

U4A IDE HELP

Table of Contents

1. Instruction.....	10
2. 개발 환경 구성.....	12
2.1 U4A IDE PAM (Product Available Matrix).....	13
2.2 SAP GUI 버전 별 요구사항.....	14
2.3 ABAP Developer 권한 요구.....	15
3. U4A IDE Platform Beginning.....	17
3.1 UI5란 무엇인가?.....	18
3.2 U4A IDE Platform System Architecture.....	19
3.3 SAP Fiori vs U4A IDE Platform 비교.....	20
3.4 내/외 망 연계 구성도.....	21
4. U4A IDE Workbench 사용 방법.....	22
4.1 U4A Workbench 설치 및 실행.....	23
4.1.1 호스트 정보 체크.....	24
4.1.2 U4A WorkSpace Toolkit 설치.....	29
4.1.3 U4A Workbench 실행.....	35
4.2 U4A IDE Workbench 기능 영역 개요.....	38
4.3 추가 기능 메뉴 - Extras.....	43
4.3.1 App. Package Change.....	44
4.3.2 App Import/Export.....	47
4.3.3 Shortcut Manager.....	50
4.3.4 U4A Workbench 버전 확인.....	58
4.4 추가 기능 메뉴 - Utilities.....	61
4.4.1 Select Browser Type.....	62
4.4.2 Video Record.....	65

4.5 추가 기능 메뉴 - System	69
4.5.1 New Window	70
4.5.2 Close Window	72
4.5.3 Options.....	74
4.5.4 Logoff.....	77
4.5.5 Release Notes.....	78
4.5.6 Error Log.....	80
4.5.7 Administrator.....	83
4.6 추가 기능 메뉴 - Help	87
4.6.1 Help Document	88
4.7 도구 모음	90
4.7.1 Hide	91
4.7.2 T-Code.....	93
4.7.3 Browser Pin.....	95
4.7.4 Zoom.....	97
4.7.5 Window Text Search.....	99
4.8 Application 생성 하기	100
4.8.1 Application Create Option 설명.....	102
4.9 Application 수정 하기	105
4.10 Application 삭제 하기.....	108
4.11 Application 복사 하기.....	109
4.12 Application 조회 하기.....	112
4.13 Application 실행 하기.....	114
4.14 App Mult Preview	116
4.15 Example Open.....	120
4.16 ROOT(Document)	125

U4A IDE HELP

4.16.1 Touch style	130
4.16.2 CSS Link Add	132
4.16.3 JS Link Add	138
4.16.4 UI Theme	144
4.16.5 Wait Type.....	147
4.17 U4A Workbench UX Design 기능 영역 개요.....	149
4.18 UI 디자인 영역 개인화.....	152
4.19 UX Design Area	157
4.20 Attributes	159
4.21 Attribute Helper.....	162
4.22 UI Template Wizard	164
4.22.1 Forms UI Create.....	165
4.22.2 Table UI Create.....	170
4.22.3 Report Template Create	175
4.23 Table Binding	184
4.24 Table unBinding.....	191
4.25 Structure Binding	195
4.26 Structure Unbinding	199
4.27 Binding Options	203
4.27.1 Property - Bind Type.....	204
4.27.2 Property - Nozero	206
4.27.3 Property - Is number format.....	207
4.27.4 Property - Conversion Routine.....	209
4.28 Events	210
4.29 CheckBox Select	212
4.30 Java Script	214

5. APP 개발 하기 Part 1.....	216
5.1 추가 기능 메뉴 - Style Class.....	217
5.1.1 Font Style Wizard.....	218
5.1.2 UI5 Predefined CSS.....	240
5.2 추가 기능 메뉴 - Utilities.....	248
5.2.1 Select Browser Type.....	249
5.2.2 OTR F4 Request.....	252
5.2.3 Video Record.....	254
5.3 추가 기능 메뉴 - Edit.....	258
5.3.1 StyleCSS Editor.....	259
5.3.2 ScriptCode Editor.....	265
5.3.3 HTMLCode Editor.....	273
5.3.4 Error Page Editor.....	278
5.3.5 Skeleton Scr Setting.....	282
5.4 추가 기능 메뉴 - System.....	284
5.4.1 New Window.....	285
5.4.2 Close Window.....	287
5.4.3 Options.....	289
5.4.4 Logoff.....	292
5.4.5 Release Notes.....	293
5.4.6 Error Log.....	295
5.4.7 Administrator.....	298
5.5 추가 기능 메뉴 - Help.....	302
5.5.1 Help Document.....	303
5.5.2 Document.....	305
5.6 도구 모음.....	308

5.6.1 Hide	309
5.6.2 T-Code.....	311
5.6.3 Browser Pin.....	313
5.6.4 Zoom.....	315
5.6.5 Window Text Search.....	317
5.7 Button 사용 방법.....	318
5.8 Label 사용 방법.....	322
5.9 ComboBox 사용 방법	325
5.10 RadioButton 사용 방법	331
5.11 Find 기능	335
5.12 Help Document 기능.....	340
5.13 Display-Change 기능.....	341
5.14 Syntax Check 기능	344
5.15 Activate 기능	347
5.16 UI Template Wizard	349
5.16.1 Forms UI Create.....	350
5.16.2 Table UI Create.....	355
5.16.3 Report Template Create	360
5.17 MIME Repository 기능	369
5.18 Controller(Class builder) 기능.....	377
5.19 Icon list 기능	378
5.20 Add Event Method 기능	384
5.21 RunTime Class Navigator 기능	386
5.22 Binding Popup 기능.....	389
5.23 UI Obj의 생성,삭제,위치변경.....	392
5.24 UI Obj의 Attributes 구성 요소	406

5.25 Event method 기초	409
5.26 Skeleton Screen	412
6. APP 개발 하기 Part 2.....	416
6.1 ABAP Class /U4A/CL_UTILITIES	417
6.1.1 Methodes FILE_UPLOAD.....	418
6.1.2 Methodes FILE_DOWN.....	427
6.1.3 Methodes FILE_DOWN2	432
6.1.4 Methodes M_LETTERMSG	436
6.1.5 Methodes SET_FREESTYLE_SCRIPT	440
6.1.6 Methodes LOCATION_HREF.....	442
6.1.7 Methodes SET_MODEL_SORTER	445
6.1.8 Methodes SET_M_TABLE_UNSELECT	448
6.1.9 Methodes SET_MODEL_FILTER	450
6.1.10 Methodes CREATE _SAP_SHORT_CUT	453
6.1.11 Methodes UI_PRINT	458
6.1.12 Methodes VOICE_MESSAGE.....	461
6.1.13 Methodes GET_UI_SID	463
6.1.14 Methodes SET_UI_TABLE_UNSELECT	465
6.1.15 Methodes SET_UI_TABLE_FIRST_VISIBLE_ROW	468
6.1.16 Methodes SET_UI_TABLE_FOCUS	471
6.1.17 Methodes RESET_UITABLE_SORT	474
6.1.18 Methodes RESET_UITABLE_FILTER	476
6.1.19 Methodes UITABLE_AUTORESIZE_COLUMN	479
6.1.20 Methodes GET_SELECTED_INDEX_TABLE_UI	482
6.1.21 Methodes APP_LOG_OFF.....	485
6.1.22 Methodes RESET_USAGE_APP.....	489

6.1.23 Methodes MESSAGE_STRIP491

6.1.24 Methodes PAGE NAVIGATION495

6.1.25 Methodes M_MESSAGEBOX.....498

6.1.26 Methodes GET_SELECTED_INDEX.....505

6.1.27 Methodes DIALOG_OPEN.....508

6.1.28 Methodes DIALOG_CLOSE510

6.1.29 Methodes M_MESSAGETOAST.....513

6.1.30 Methodes EXT_WIN_OPEN516

6.1.31 Methodes SET_STYLE520

6.1.32 Methodes RENDERING_USAGE_APP525

6.1.33 Methodes ADD_COACHMARKS528

6.1.34 Methodes GET_SELECTED_INDEX DEP530

6.1.35 Methodes MESSAGE_POPOVER.....533

6.1.36 Methodes USAGE_DIALOG536

6.1.37 Methodes RUN_ASYNC_ON_REQUEST541

6.1.38 Methodes ATTACH_LANDSCAPE.....548

6.1.39 Methodes DETACH_LANDSCAPE554

6.1.40 Methodes CLONE_UI_RUNTIME_INST558

6.1.41 Methodes MOBILE_VIBRATE562

6.1.42 Methodes CHANGE_UI5_THEME.....565

6.1.43 Methodes SET_BROWSER_TITLE.....568

6.1.44 Methodes PARSE_BASE64_IMAGE570

6.1.45 Methodes CONV_HTML2IMG.....573

6.1.46 Methodes SPLIT_FILENAME.....576

6.1.47 Methodes JS_FILES_LOAD578

6.1.48 Methodes CHK_DATE_VALUE.....582

U4A IDE HELP

6.1.49 Methodes GET_MIME_REPOSITORY_API.....	587
6.1.50 Methodes EXCEL_UPDATA_BIN_TO_ABAP.....	589
6.1.51 Methodes DELETE_SAME_BROWSER_SESSION.....	593
6.1.52 Methodes CLOSE_MESSAGE_TOAST_ALL.....	597
6.1.53 Methodes CLOSE_MESSAGE_BOX_ALL.....	600
6.2 ABAP Class /U4A/CL_F4_HNADL.....	604
6.2.1 Methodes F4_HNADL_SH.....	605
6.2.2 Methodes F4_HNADL_SHDYN.....	614
6.3 ABAP Class /U4A/CL_EXTUI_EVENTS.....	624
6.3.1 Methodes EXTUI PASTE.....	625
6.3.2 Methodes EXTUI SWIPE.....	631
6.3.3 Methodes EXTUI KEYDOWN.....	635
6.3.4 Methodes EXTUI CLICK LAYOUT.....	639
6.3.5 Methodes EXTUI FOCUSIN.....	643
6.3.6 Methodes EXTUI CLICK.....	646
6.3.7 Methodes EXTUI CONTEXTMENU.....	649
6.3.8 Methodes EXTUI BROWSER_EVENT.....	652
7. APP 개발 하기 Part 3.....	656
7.1 WorldMap.....	657
7.2 Angulargauge.....	664
7.3 SingleDonutChart.....	673
7.4 AppContain.....	678
7.5 Iframe.....	680
7.6 BadgeButton.....	683
7.7 TabContainer.....	687
7.8 SessionWorker.....	690

7.9 Presstrigger	693
7.10 Richeditor	696
7.11 CountdownTimer	699
7.12 CurrentTimer	702
7.13 VerticalTimeLine	704
7.14 UnitedStateMap.....	709
7.15 RichTextEditor	716
8. APP 개발 하기 Part 4.....	719
8.1 sapUiMargin.....	720
8.2 sapUiSizeCompact and Cozy	726
9. APP 개발 하기 Part 5.....	728
9.1 AmSerialChartComposite.....	729
9.1.1 Properties.....	730
9.1.2 FieldCat	743
9.1.3 Axis.....	761
9.1.4 Guide.....	775
10. APP 개발 하기 Part 6.....	781
10.1 Dynamic ComboBox Create.....	782
10.2 Dynamic Column Create	790
10.3 Dynamic Table Create.....	794

1. Instruction

■ U4A IDE Platform 의 특징

-  SAP Interface
-  WEB Developer
-  WEB Designer
-  Fiori Development
-  Additional H/W

➔ **필요없이,**



멀티 브라우저/디바이스 지원

ABAP개발자가 UI5기반 UX를 손쉽게 개발할 수 있습니다.

■ U4A IDE Platform 개발의 편의성

UI5 개발 환경의 일원화
(화면 및 비즈니스 Logic 통합 개발환경 제공)



No Interface를 통한 UI5 Application 개발 가능
(※ ERP DB 기존 No Interface Web 개발 가능)



One Application, One Developer Assign 가능
(개발 시간 단축 효과)



단순하고(simple), 직관적인(Intuitive) 개발환경 제공
(Toolkit 교육 수강 기간 어플 소모)



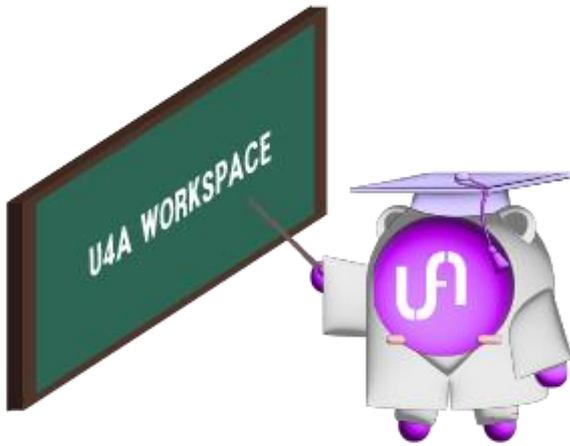
■ U4A IDE Platform 개발의 생산성

구분		SAP UI5 구축	U4A IDE Platform 구축
개발 방법	개발 방법	<p>ABAP, EAI, WEB Data 요건 협의(요청/결과) 화면디자인 화면제어(Logic개발) Data 추출 (Logic 개발), Data 연계 (Config 설정), Data 추출 (결과 조회) 기능 TEST</p>	<p>ABAP 화면디자인 Data 추출 (Logic 개발) 기능 TEST</p>
	특징	<p>개발인력 Platform 의존도</p> <p>3개 영역(ABAP+EAI+WEB) Skill Set을 보유한 인력이 동시 필요 의존적 : Platform별로 별도 소스 개발</p>	<p>ABAP Skill Set 보유 인력으로 해결 독립적 : 동일 소스를 모든 Platform에 동시 적용 가능</p>

Only ABAP개발자만 참여 개발 가능



2. 개발 환경 구성



2.1 U4A IDE PAM (Product Available Matrix)

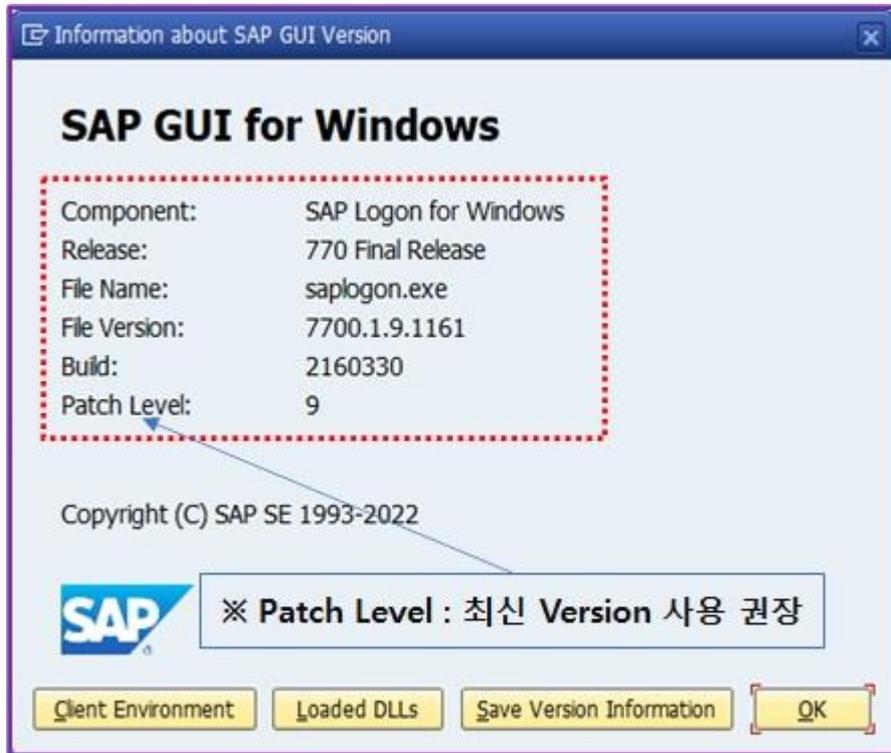
Classification	Minimum Specification	Recommended Specification														
SAP System	SAP S/4 HANA 1610(BASIS Component 7.51) * Must be a Unicode system!!	SAP S/4 HANA 2021(BASIS Component 7.56) ~ [On-premises or PCE environments available] * Must be a Unicode system!!														
SAP GUI (Win GUI) For ABAP Controller Class Coding	SAP GUI 770 (Patch Lev. 1 or Higher) ~															
User OS For U4A Workspace 3.0 Toolkit	MS Windows 10 ~															
Browser	<table border="1"> <thead> <tr> <th>Laptop (Or Desktop)</th> <th>Mobile (Or Tablet)</th> </tr> </thead> <tbody> <tr> <td>IE11(Not Support)</td> <td>Chrome for Android 49 (or Higher)</td> </tr> <tr> <td>IE Edge Chromium</td> <td>Android webview 49 (or Higher)</td> </tr> <tr> <td>Chrome 49 (or Higher)</td> <td>Safari on ios 10</td> </tr> <tr> <td>Safari 10</td> <td>Firefox for Android 44 (or Higher)</td> </tr> <tr> <td>Firefox 44 (or Higher)</td> <td>Edge Mobile 12</td> </tr> <tr> <td>Opera 17</td> <td>Opera for Android 17</td> </tr> </tbody> </table>		Laptop (Or Desktop)	Mobile (Or Tablet)	IE11(Not Support)	Chrome for Android 49 (or Higher)	IE Edge Chromium	Android webview 49 (or Higher)	Chrome 49 (or Higher)	Safari on ios 10	Safari 10	Firefox for Android 44 (or Higher)	Firefox 44 (or Higher)	Edge Mobile 12	Opera 17	Opera for Android 17
	Laptop (Or Desktop)	Mobile (Or Tablet)														
	IE11(Not Support)	Chrome for Android 49 (or Higher)														
	IE Edge Chromium	Android webview 49 (or Higher)														
	Chrome 49 (or Higher)	Safari on ios 10														
	Safari 10	Firefox for Android 44 (or Higher)														
	Firefox 44 (or Higher)	Edge Mobile 12														
Opera 17	Opera for Android 17															

2.2 SAP GUI 버전 별 요구사항

■ 사용자 SAP GUI Version 정보 확인은 해당 로그인 패드(Logon Pad) 좌측상단  Icon 을 Click 하면 Menu가 표시 되고, 항목 중 "About SAP Logon ..." 을 Click 하면

대상 SAP GUI Version 및 Patch level을 확인 할 수 있으며, "SAP GUI Version 770" 부터 U4A WorkSpace 사용 가능

< Figure : SAP Version Information (Verified Version) >



< SAP GUI 770 >

2.3 ABAP Developer 권한 요구

U4A IDE Platform은 SAP SE ABAP Technology 기반으로 개발 되었으며, 일반적인 ABAP 개발할 때와 마찬가지로 다음과 같이 Developer 권한이 필요하다.

1. ABAP Developer Access Key (※ 단, S/4 HANA는 해당 없음)

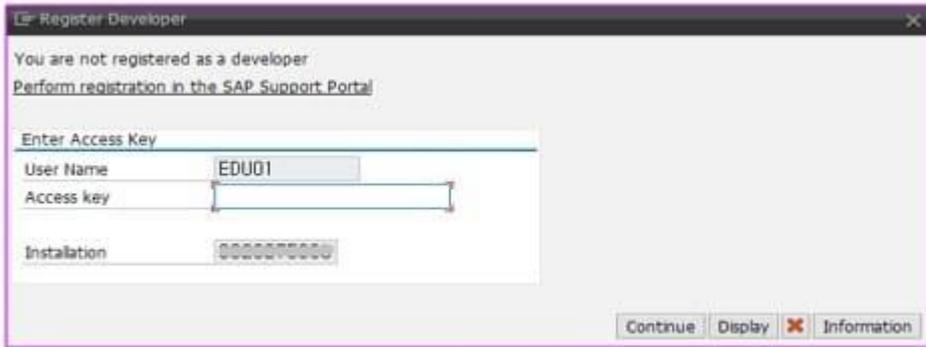


Figure : Register Developer (시스템 관리자에게 요청)

2. 필요 T-Code 권한 정보

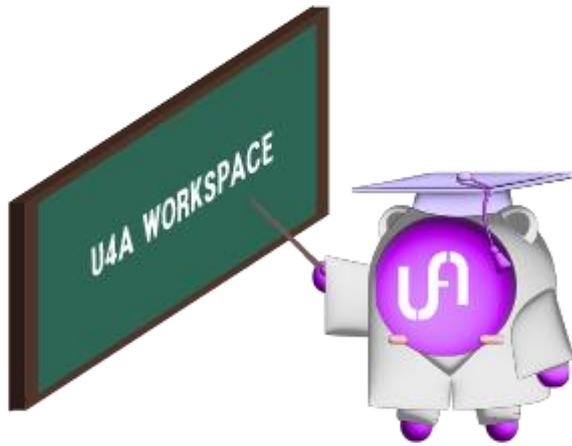
T-CODE	Descriptions
SE80	Object Navigator
SE11	ABAP Dictionary Maintenance
SE24	ABAP Class Builder
SICF	HTTP Service Hierarchy Maintenance
SE37	ABAP Function Modules
SE38	ABAP Editor
SE30	ABAP Objects Runtime Analysis
ST22	ABAP Dump Analysis
SMW0	SAP Web Repository
ZU4A_WB	[U4A] UI5 Application Workbench

3. 필요 권한 오브젝트 정보

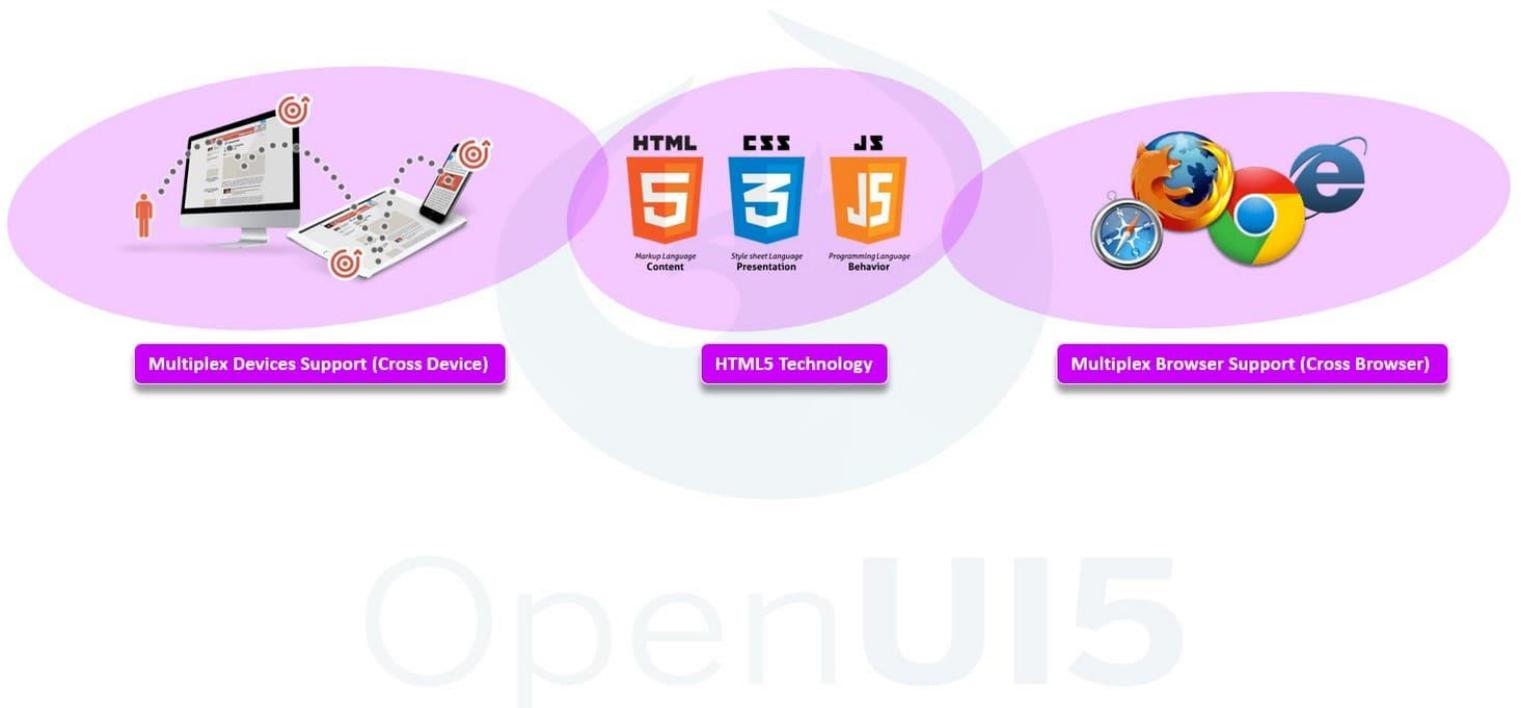
U4A IDE HELP

Authority Object	Descriptions
S_ICF_ADM	Administration for Internet Communication Framework
S_DATASET	Authorization for file access
S_DEVELOP	ABAP Workbench
S_DOKU_AUT	SE61 Documentation Maintenance Authorization
S_PROGRAM	ABAP: Program Flow Checks
S_TRANSPRT	Transport Organizer

3. U4A IDE Platform Beginning



3.1 UI5란 무엇인가?



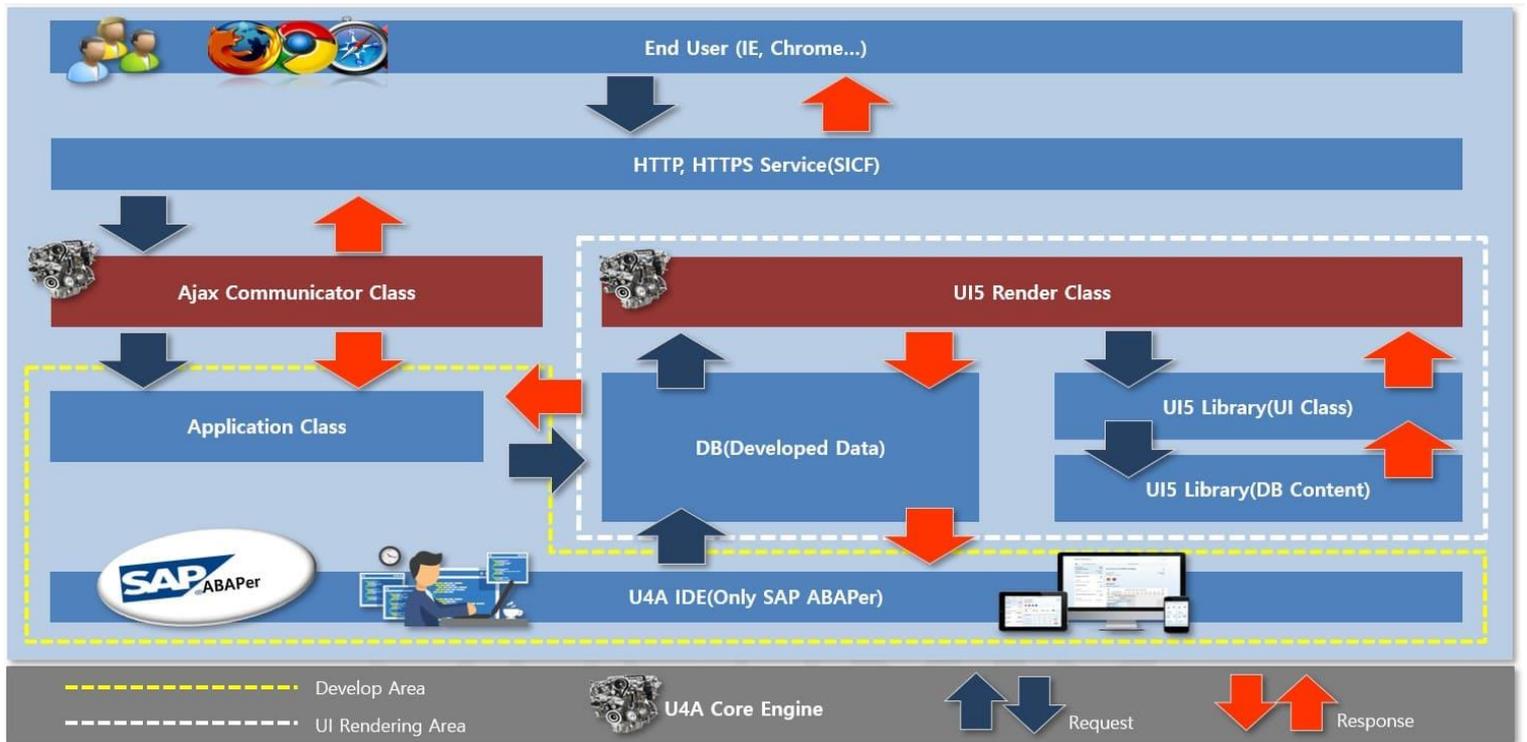
■ UI5란 HTML5 기반 UI Contents를 개발 하기 위한 Java-script Development Framework 를 의미 하며, 개발된 Contents 는

다양한 Device / Browser 를 통해 사용자와 Interaction하는 표준 UI 개발 Toolkit 임.

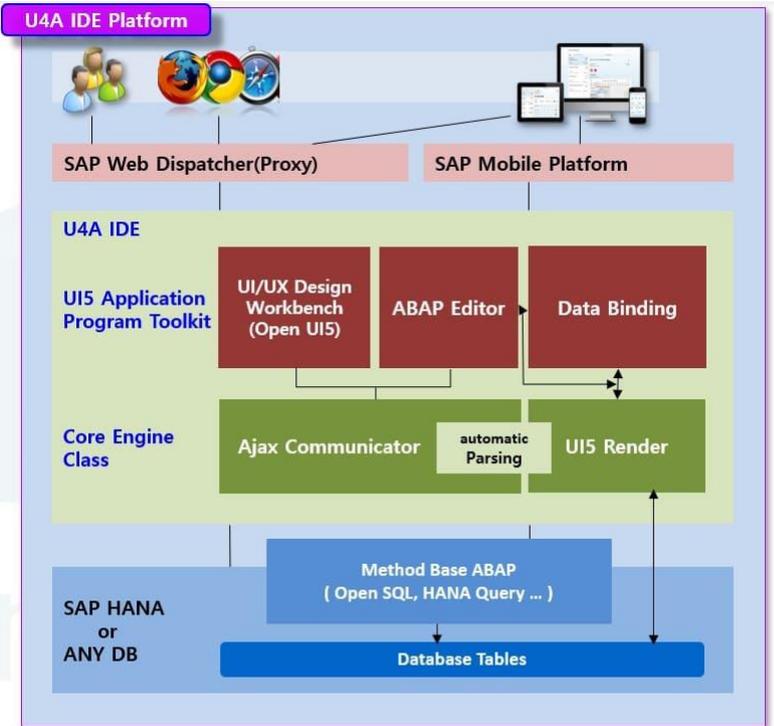
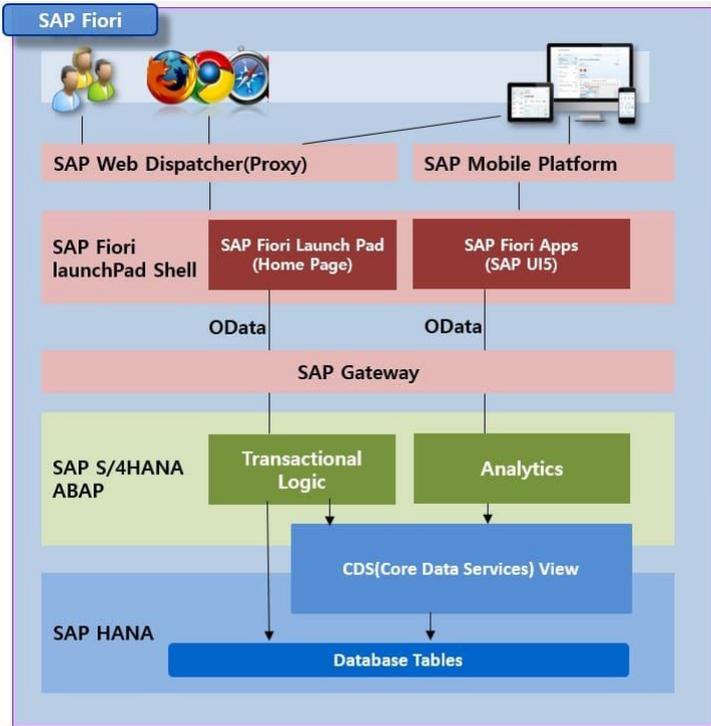
■ 주요기능

Laptop, Tablet, Mobile 등 다양한 Device 환경을 위한 풍부한 UI Object Library 제공
 Open Source Framework 와 Web 표준 수용
 Cross Device, Cross Browsing 환경 지원

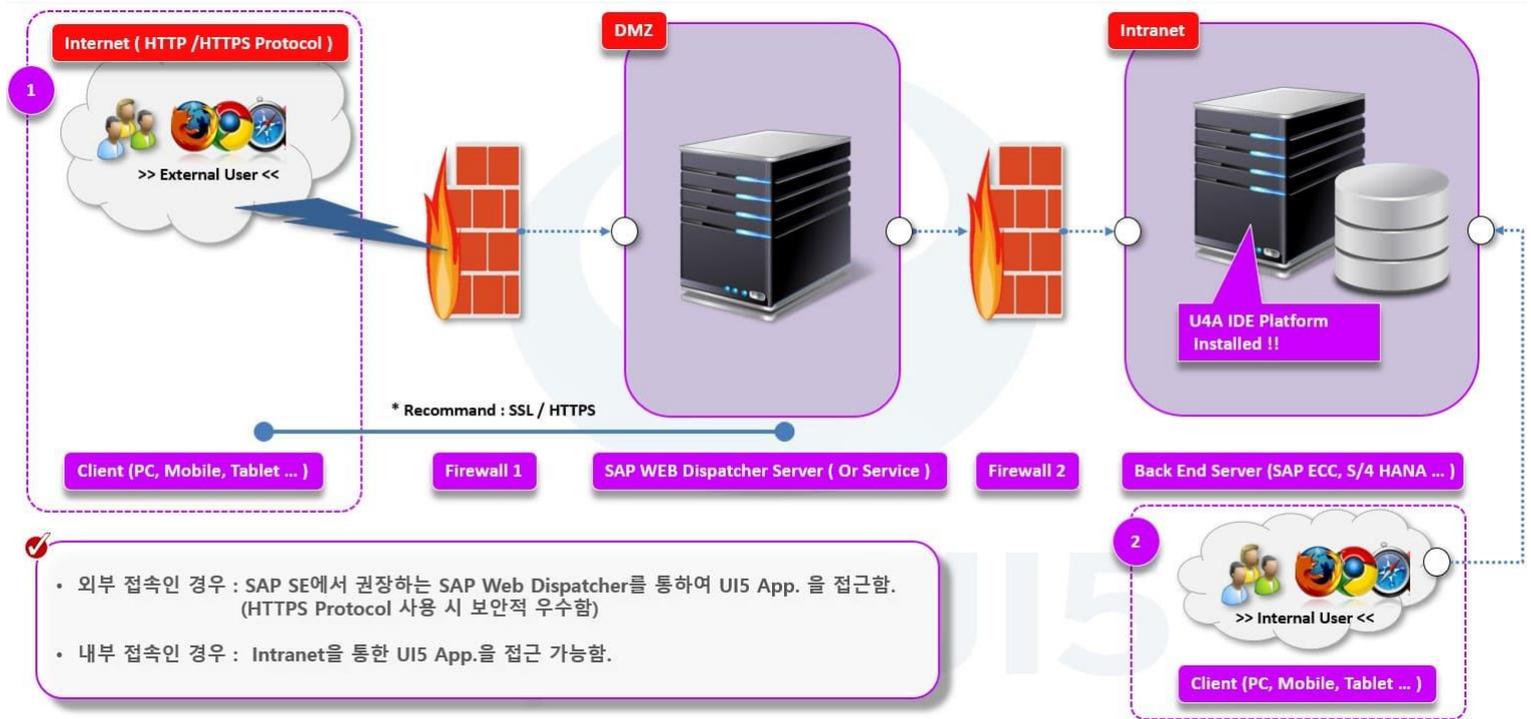
3.2 U4A IDE Platform System Architecture



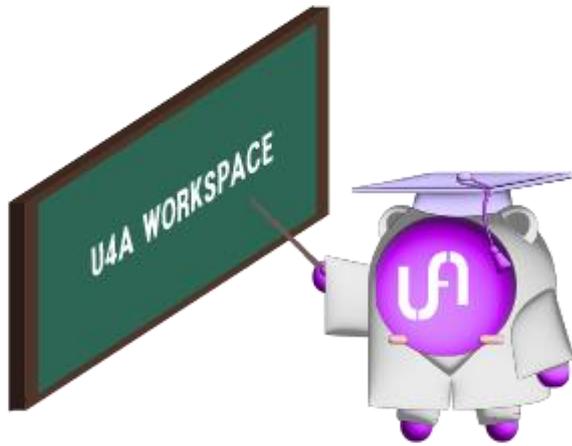
3.3 SAP Fiori vs U4A IDE Platform 비교



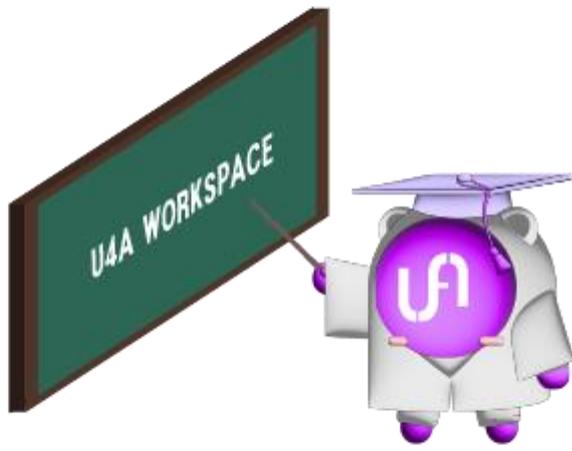
3.4 내/외 망 연계 구성도



4. U4A IDE Workbench 사용 방법



4.1 U4A Workbench 설치 및 실행



4.1.1 호스트 정보 체크

▣ 목적 : U4A WorkSpace 설치하기 전 "Host File 등록 점검 및 Host File 미등록시 설정 방법 가이드.

1. SAP GUI 실행 => T-Code : "/N/U4A/CHOST" 입력 후 실행



2. U4A WorkSpace를 설치 및 사용하기 위한 현재 Host 정보 창 상태 화면

※ Port, Host Name 정보를 본 단원에서만 블록처리 (정상적으로 설정된 상태임)

[U4A] Check Current Host

1) System Host Informations

Protocol	Active	Time Out	Port	Host Name(URL)
HTTP	🔧	600		
HTTPS	🔧	600		
SMTP	🔧	600		

2) Check Your "hosts" File Status

※ File Location => C:\Windows\system32\drivers\etc\hosts

* Result: 🛠️ Registration ok!

① "System Host Informations" : Protocol, Time Out, Port, Host Name 정보 창

※ Port, Host Name 정보를 본 단원에서만 블록처리 (정상적으로 설정된 상태임)

1) System Host Informations

Protocol	Active	Time Out	Port	Host Name(URL)
HTTP		600		
HTTPS		600		
SMTP		600		

※ 만약 "Active" 컬럼의 아이콘이 위의 사진과 다르거나(= 비활성화) "Port" 및 "Host Name"이 없거나 잘못 설정되어있다면 시스템 관리자(BC)에게 문의

② "Check Your "hosts" File Status"

2) Check Your "hosts" File Status

※ File Location => C:\Windows\system32\drivers\etc\hosts

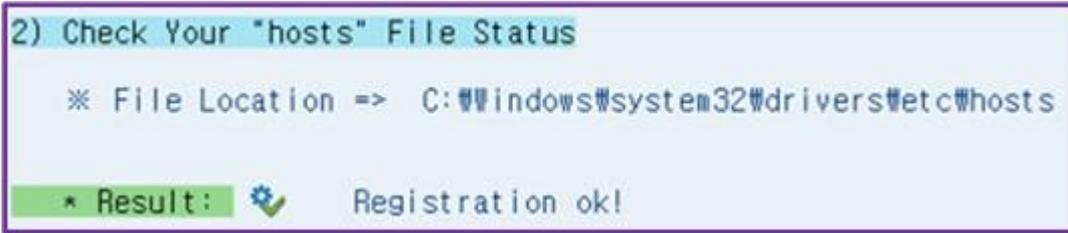
* Result:  Value setting required.

[Recommend] Please add the following to your "hosts" file.

(1) File Location : "Host 정보"를 등록할 위치

(2) Result : Host 정보가 정상적으로 등록 되었는지 상태 확인

※ 정상적으로 등록된 화면

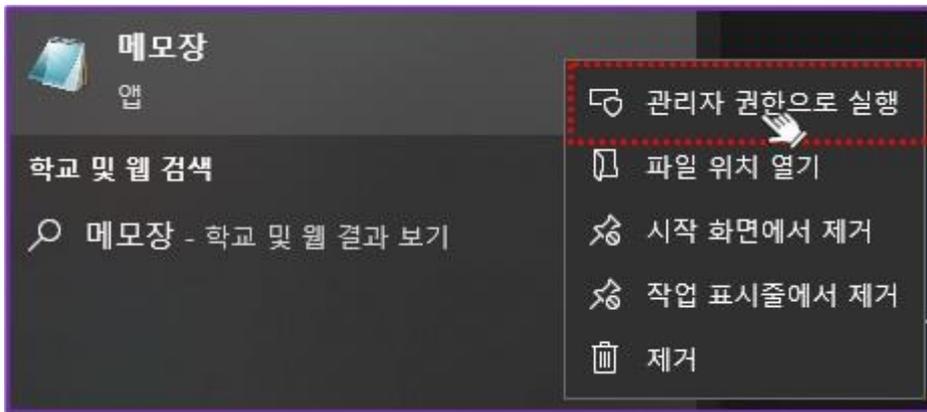


(3) Result 영역이 위의 사진과 같이 "빨강색(= 오류)"이면 Host File에 등록할 "Host 정보"를 나타냄. (Host 정보를 본 단원에서만 블록처리)

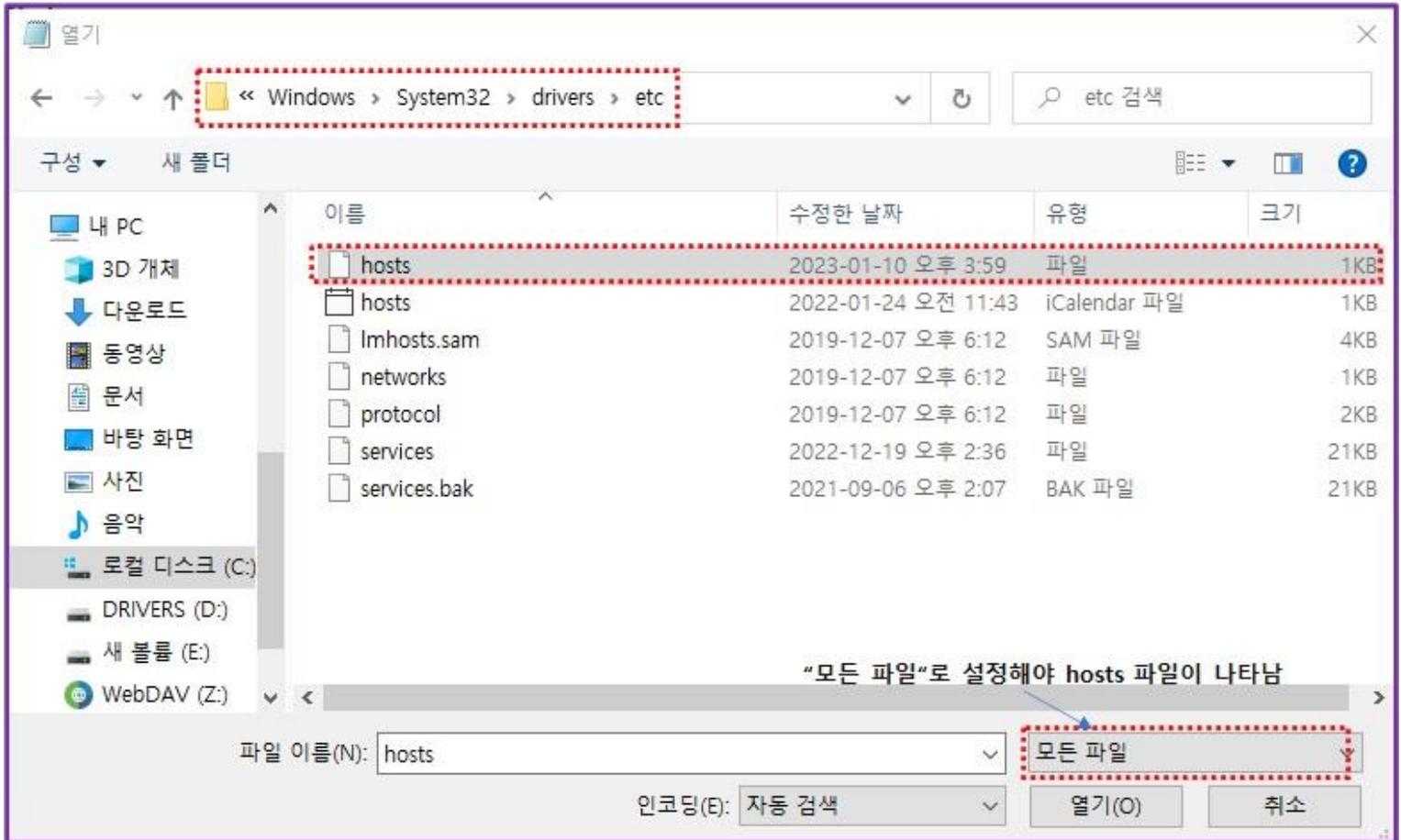
※ (3)의 Host 정보를 참고하여 (1) File Location에 나타난 경로에 들어가서 Host 정보 등록 할 것.

※ Host File 등록 및 수정 방법

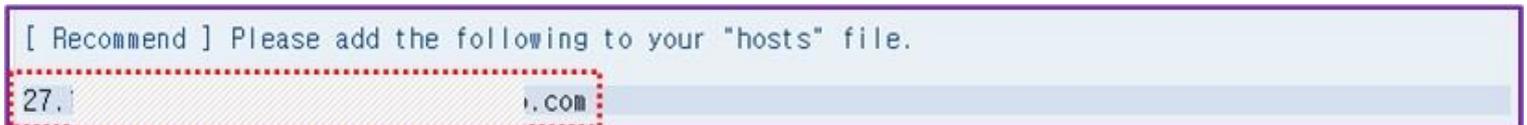
① Windows "메모장(notepad.exe)" 실행 (※ 반드시 관리자 권한으로 실행 해야 함)



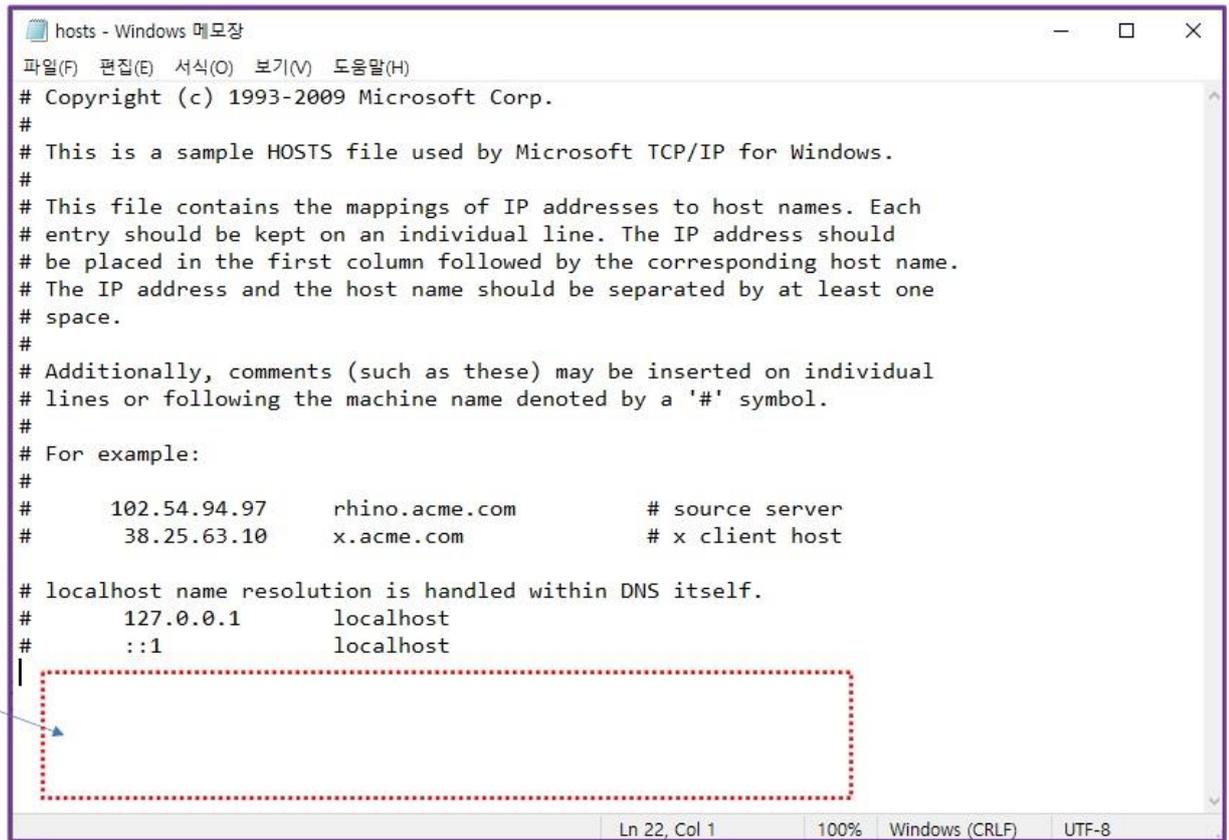
② 메모장에서 "C:\Windows\System32\drivers\etc"폴더 안의 "hosts" File 열기



③ 첫 등록시 "Check Your "hosts" File Status" = > Recommend 영역에 있는 IP, HostName 정보를 "hosts" 파일에 추가 후 저장 종료한다.



U4A IDE HELP



```
hosts - Windows 메모장
파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)
# Copyright (c) 1993-2009 Microsoft Corp.
#
# This is a sample HOSTS file used by Microsoft TCP/IP for Windows.
#
# This file contains the mappings of IP addresses to host names. Each
# entry should be kept on an individual line. The IP address should
# be placed in the first column followed by the corresponding host name.
# The IP address and the host name should be separated by at least one
# space.
#
# Additionally, comments (such as these) may be inserted on individual
# lines or following the machine name denoted by a '#' symbol.
#
# For example:
#
#       102.54.94.97       rhino.acme.com       # source server
#       38.25.63.10      x.acme.com         # x client host
#
# localhost name resolution is handled within DNS itself.
#       127.0.0.1        localhost
#       ::1             localhost
|
```

위의 단계에서 찾은 IP(IPv4), Host Name 정보를 입력한다.

 "hosts" File 수정 후 변경된 내용을 반영하기 위해서 PC 다시 시작 권장.

4.1.2 U4A WorkSpace Toolkit 설치

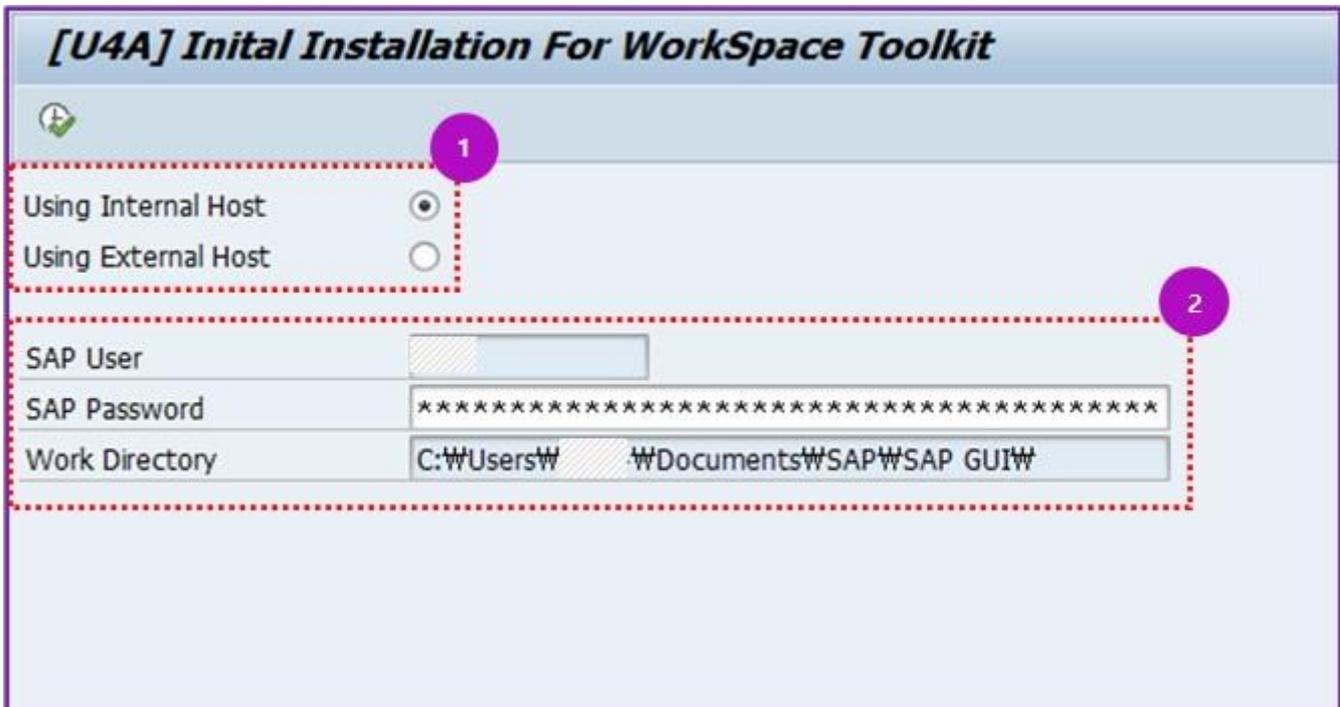
1. SAP GUI 실행 => T-Code : "/N/U4A/WST_INST" 입력 후 실행



2. /U4A/WST_INST 실행화면.

※ Internal Host, External Host 중 무엇을 지정해야하는지 모른다면 시스템관리자 또는 U4A Solution Team 에
게 문의

① "Using Internal Host"



1) Internal Host : 서버 내부 호스트(Intranet) => 동일한 네트워크망에 연결 되어있는 경우

※ 아래의 Host Name(URL)을 이용.

1) System Host Informations

Protocol	Active	Time Out	Port	Host Name(URL)
HTTP		600		
HTTPS		600		
SMTP		600		

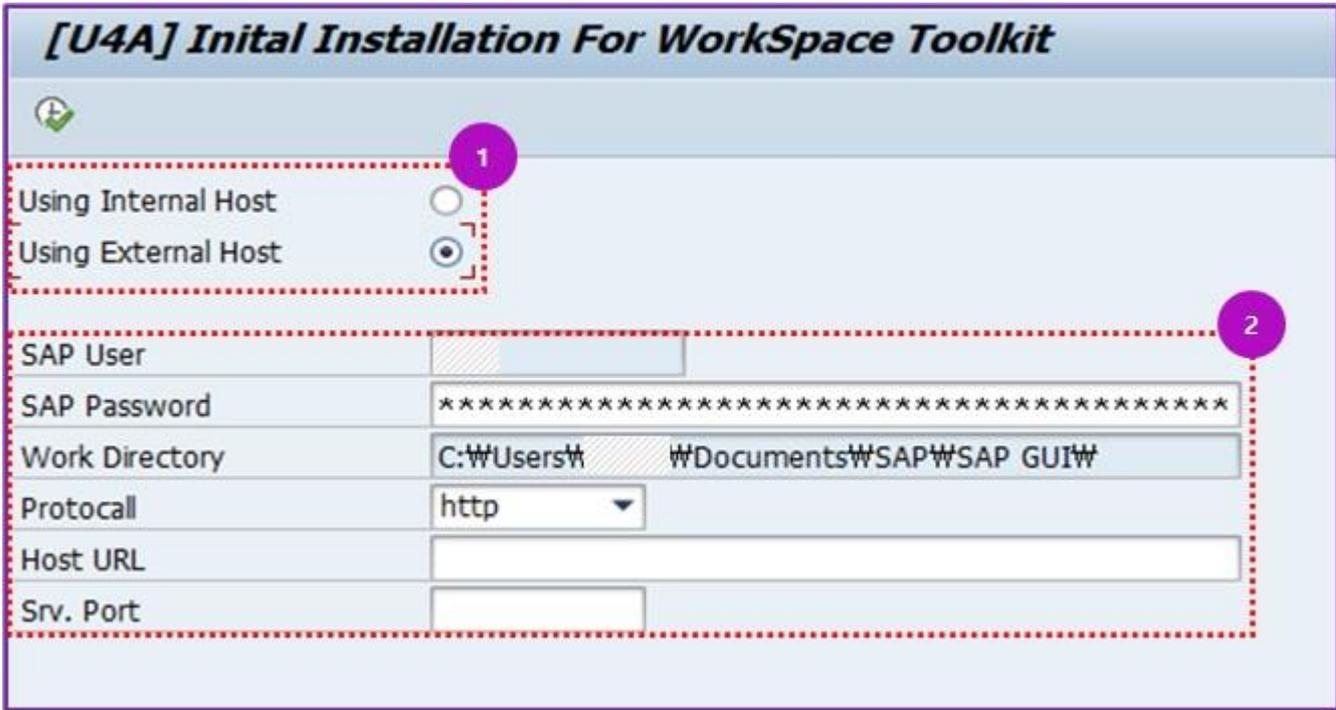
2) SAP GUI ID,PW 및 설치 경로

SAP User : "SAP GUI ID" 입력

SAP Password : "SAP GUI Password" 입력

Work Directory : "U4A WorkSpace" 설치 경로(수정불가)

② "Using External Host"



1) External Host : 서버 외부 호스트 => 동일한 네트워크망에 연결 되어있지 않은 경우 (외부 접근 가능 용도)

2) SAP GUI ID,PW 및 설치 경로

SAP User : "SAP GUI ID" 입력

SAP Password : "SAP GUI Password" 입력

Work Directory : "U4A WorkSpace" 설치 경로(수정불가)

Protocol : "http" 또는 "https" 설정 가능 => 시스템관리자(BC)에게 문의 후 설정

Host URL : 시스템관리자(BC)에게 문의 후 설정

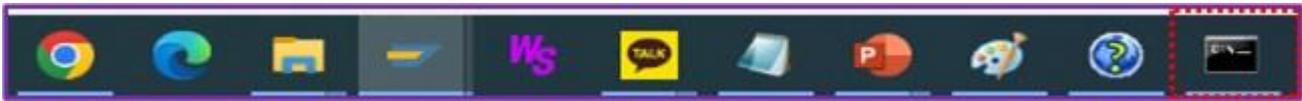
Srv.Port : 시스템관리자(BC)에게 문의 후 설정

3. 위의 2번 단계 설정을 마친뒤 실행 버튼(F8) 클릭하여 "U4A WorkSpace" 설치 진행

[U4A] Inital Installation For WorkSpace Toolkit

(1) U4A WorkSpace 설치 파일 다운로드 단계 화면 ("SAP GUI" + "작업표시줄")

Downloading WST installation file ...



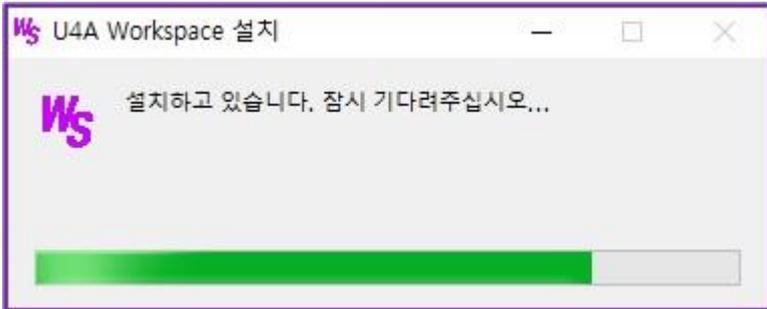
```

C:\Windows\SysWOW64\CMD.EXE
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
100 2929k 100 2929k 100 52 11.7M 213 --:--:-- --:--:-- --:--:-- 11.8M
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
100 2929k 100 2929k 100 52 10.8M 196 --:--:-- --:--:-- --:--:-- 10.8M
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
100 2929k 100 2929k 100 52 13.3M 242 --:--:-- --:--:-- --:--:-- 13.4M
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
100 2929k 100 2929k 100 52 13.3M 242 --:--:~ --:~:~ --:~:~ 13.4M
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
100 2929k 100 2929k 100 52 11.6M 212 --:~:~ --:~:~ --:~:~ 11.6M
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
100 2929k 100 2929k 100 52 12.5M 227 --:~:~ --:~:~ --:~:~ 12.6M
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
100 2929k 100 2929k 100 52 10.8M 197 --:~:~ --:~:~ --:~:~ 10.9M
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
100 2929k 100 2929k 100 52 11.3M 205 --:~:~ --:~:~ --:~:~ 11.3M
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
100 2929k 100 2929k 100 52 10.6M 193 --:~:~ --:~:~ --:~:~ 10.6M
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
0 0 0 0 0 0 0 0 --:~:~ --:~:~ --:~:~ 0
    
```

(위의 화면은 다운로드 속도, 정상적으로 다운로드 진행되는지 상태를 나타낸다.)

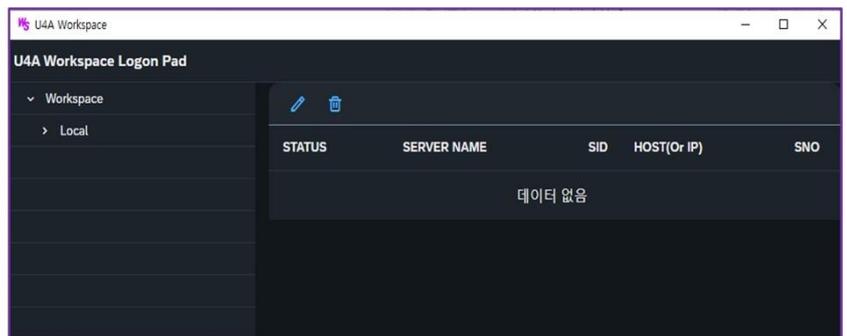
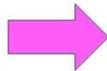
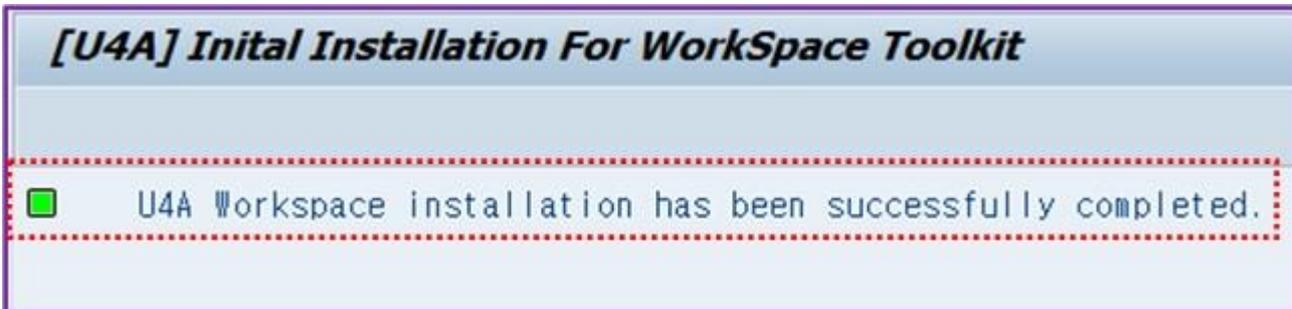
※ 다운로드 진행시 위의 화면과 다르게 진행된다면 "U4A Solution Team"에 문의 바람

(2) U4A WorkSpace 설치 단계 화면 ("SAP GUI" + "U4A WorkSpace")



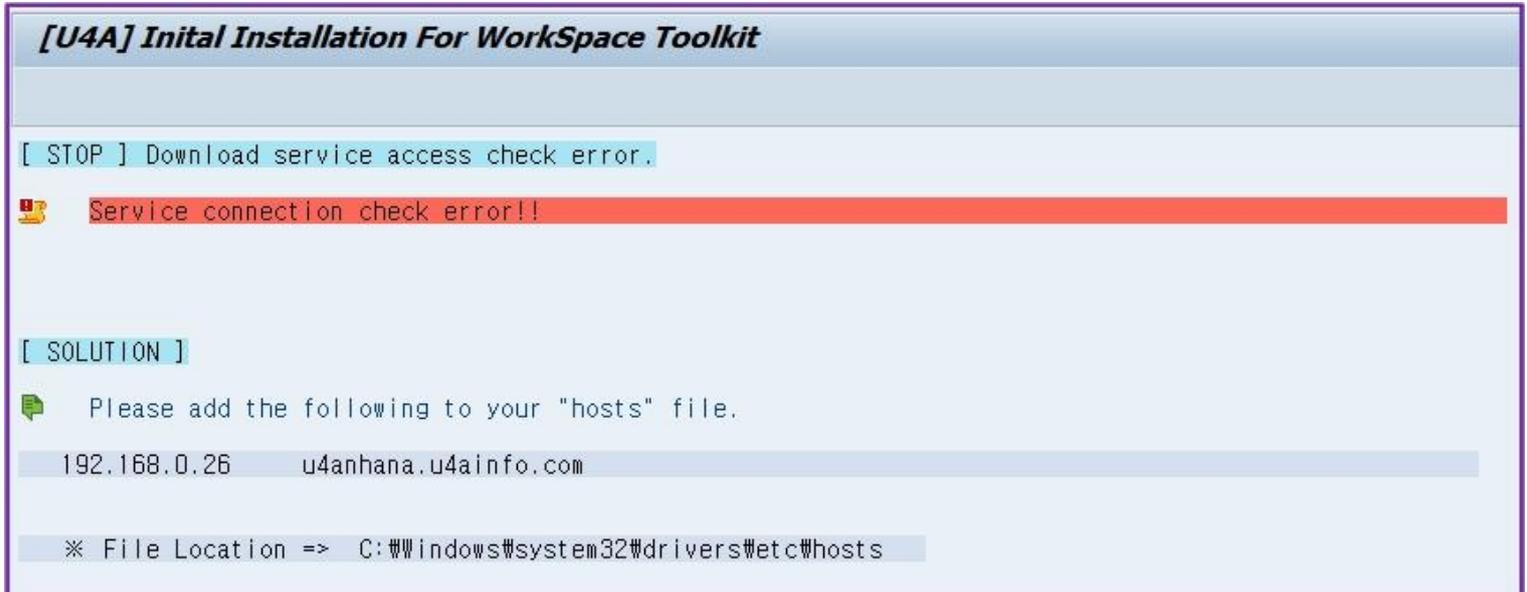
(3) U4A WorkSpace 설치 완료 화면 ("SAP GUI" + "U4A WorkSpace")

※ U4A WorkSpace가 정상적으로 설치되면 자동 실행.



4) U4A WorkSpace 설치 에러 화면

- ① Host 정보 미등록 ("호스트 정보 체크" 단원 참조하여 Host 정보 등록 할것)



<Figure : "Using Internal" + "Host 정보 미등록" : U4A WorkSpace 설치 오류 화면>



- 1) U4A WorkSpace 설치 시간은 평균 1분이며 사용자 네트워크 상태에 따라 달라짐
- 2) 설치를 진행하면서 위의 사례와 다른 케이스로 에러 발생 시 "U4A Solution Team" 에게 문의

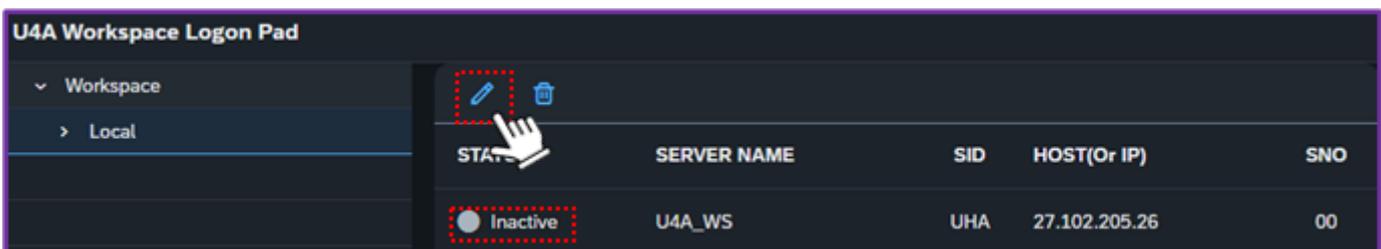
4.1.3 U4A Workbench 실행



② SAP GUI T-CODE 실행 : "/N/U4A/WS" 입력



2. 초기 서버 STATUS : "Inactive" => Edit 버튼 실행함



3. Host 및 Port 입력 후 확인 (Protocol => 디폴트값은 "http" 변경 가능함)

U4A_WS

Protocol: *

http

Host: *

Port:

✓ ✕

4. 서버 STATUS : "Active" 확인 => 더블 클릭으로 실행함

U4A Workspace Logon Pad

Workspace

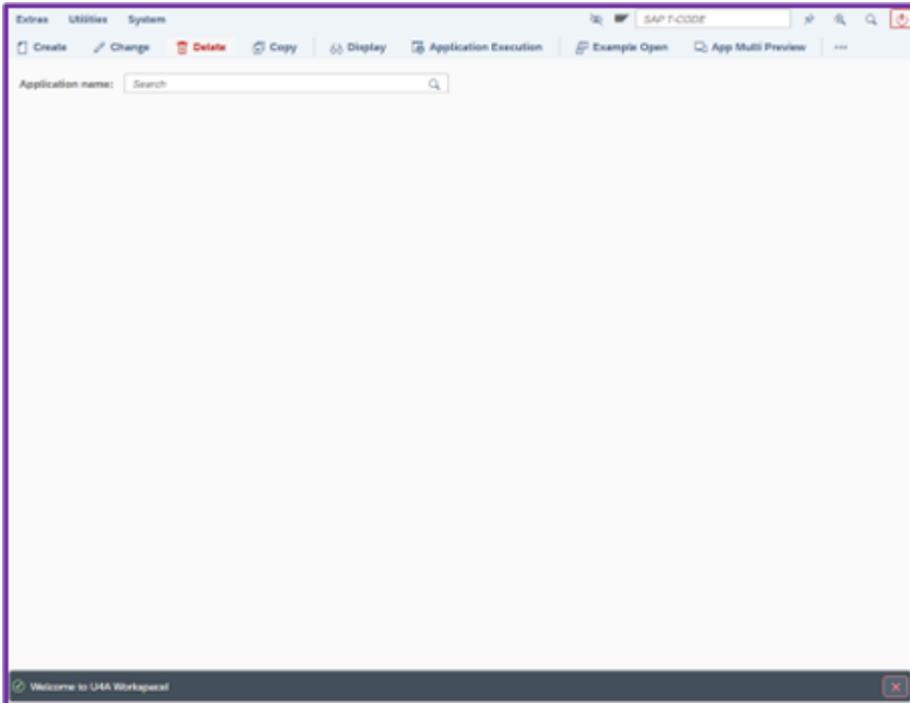
Local

STATUS	SERVER NAME	SID	HOST(Or IP)	SNO
Active	U4A_WS			

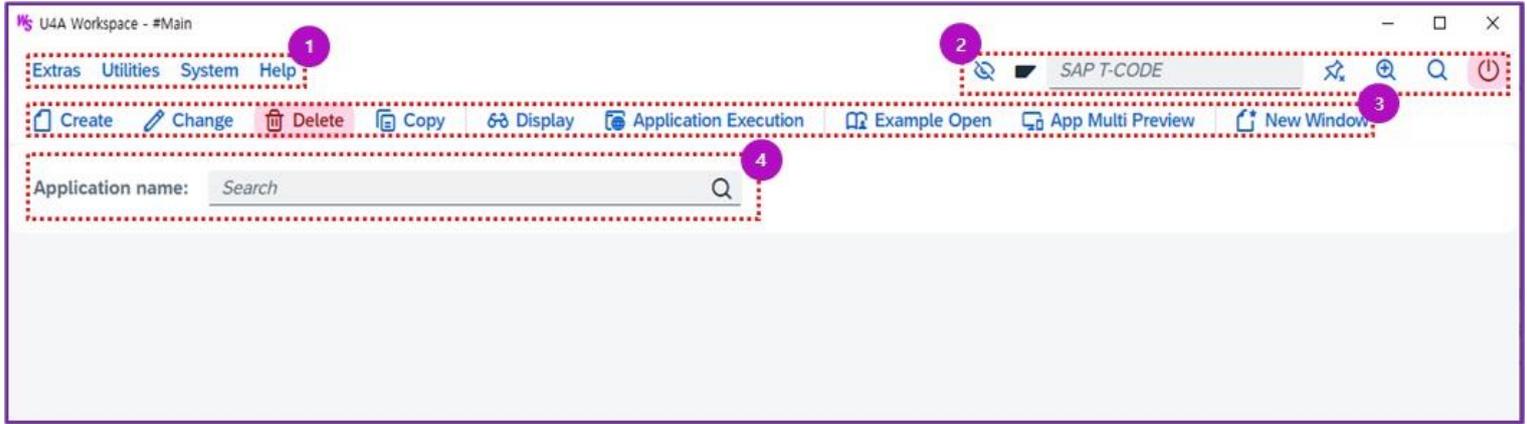
5. SAP ID & PW 입력 후 로그인

The image shows the 'U4A Workspace Login' form. It includes fields for 'CLIENT' (with '800' entered), 'ID', 'PASSWORD', 'LANGUAGE' (with 'EN' entered), and a 'Remember' checkbox (checked). A blue 'LOGIN' button is at the bottom. Annotations include a speech bubble labeled '"SAP ID"' pointing to the ID field and another labeled '"SAP PW"' pointing to the password field. Red dashed boxes highlight the ID and password fields.

6. 정상적으로 실행된 U4A_WorkSpace 화면



4.2 U4A IDE Workbench 기능 영역 개요



1. 추가 기능 모음 영역

“Extras” Menu 영역에는 다음의 기능으로 구성되어 있음.

① App. Package Change

Application 생성 시 지정된 개발 패키지(Development Package)를 변경을 하기 위한 기능

② App. Import/Export

개발된 Application을 Download 또는 Upload 하기 위한 기능

③ U4A Help Document

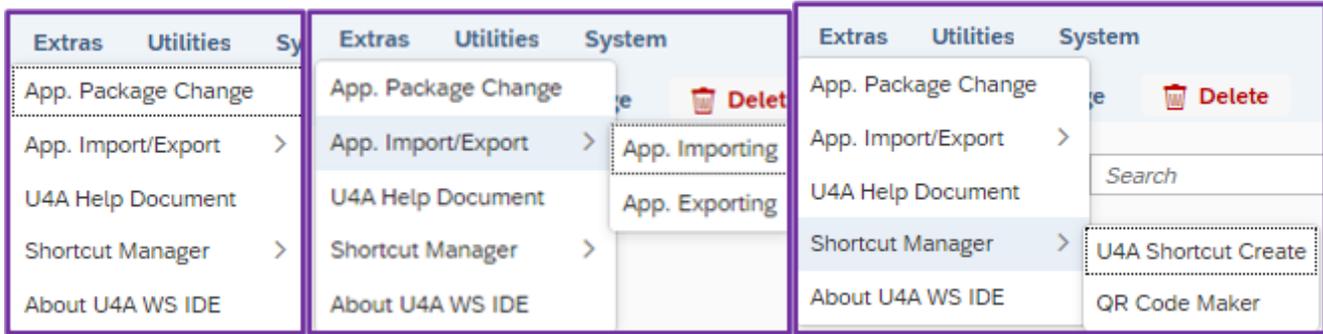
U4A IDE Platform 도움말 문서를 실행 함

④ Shortcut Manager

U4A Shortcut 및 QR Code Marker 실행 함

⑤ About U4A IDE

U4A IDE Platform Version 정보 및 추가 정보를 확인 할 수 있음.



< Figure : "Extras" 메뉴 구성 내역 확인 >

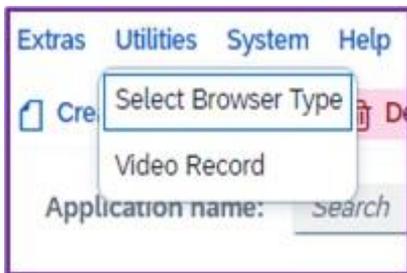
“Utilities” Menu 영역에는 다음의 기능으로 구성되어 있음.

① Select Browser Type

Application 실행시 기본 브라우저 설정

② Video Record

U4A WorkSpace 사용 중 화면 녹화 기능



< Figure : "Utilities" 메뉴 구성 내역 확인 >

“System” Menu 영역에는 다음의 기능으로 구성되어 있음.

① New Window

U4A WorkSpace 새 화면 열기

② Close Window

U4A WorkSpace 화면 종료 (해당 화면만 적용)

③ Options

U4A WorkSpace Application "Theme(테마)" 와 "CDN (U4A WS 3.0 Application Upgrade Setting)" 설정

④ Logoff

U4A WorkSpace 화면 종료 (띄어놓은 모든 화면 종료)

⑤ Release Notes

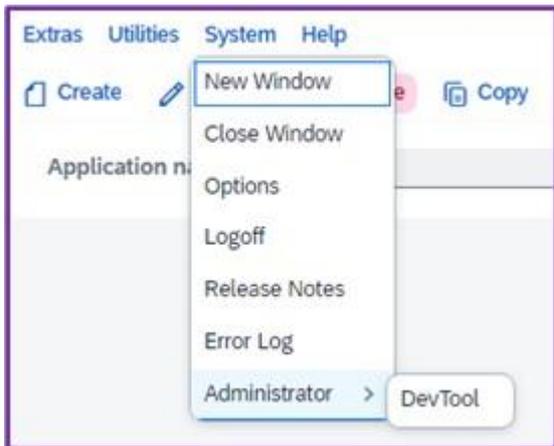
U4A WorkSpace Version Update 내역 제공 (추가, 변경, 에러 수정 사항 등)

⑥ Error Log

U4A WorkSpace 사용시 Log를 남겨 에러 발생시 Log를 확인하여 문제 해결.

⑦ DevTool

U4A WorkSpace 3.0 사용시 문제가 발생할 경우 Developer Tool을 활용하여 Support 하기 위함.

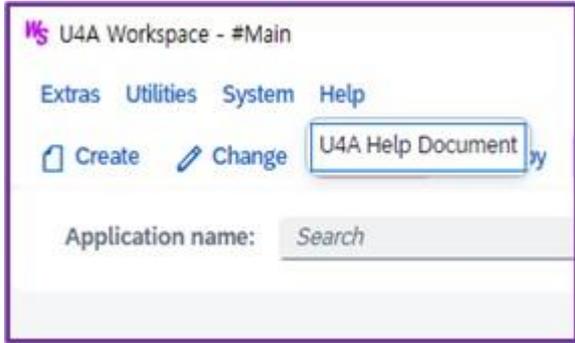


< Figure : "System" 메뉴 구성 내역 확인 >

“Help” Menu 영역에는 다음의 기능으로 구성되어 있음.

① Help Document

U4A Solution에 대한 상세 설명 제공



< Figure : "Help" 메뉴 구성 내역 확인 >

2. 도구 모음 영역에는 다음의 기능으로 구성되어 있음.

- ① Hide : U4A WorkSpace 화면 투명도를 조절하는 기능
- ② T-Code : SAP GUI T-Code 입력창과 동일한 기능
- ③ Browser Pin : U4A WorkSpace 화면 고정하는 기능
- ④ Zoom : U4A WorkSpace 화면 확대 및 축소 기능
- ⑤ Window Text Search : U4A WorkSpace 화면에 존재하는 텍스트 검색 기능
- ⑥ Logoff : U4A WorkSpace 화면 종료 (띄어놓은 모든 화면 종료)

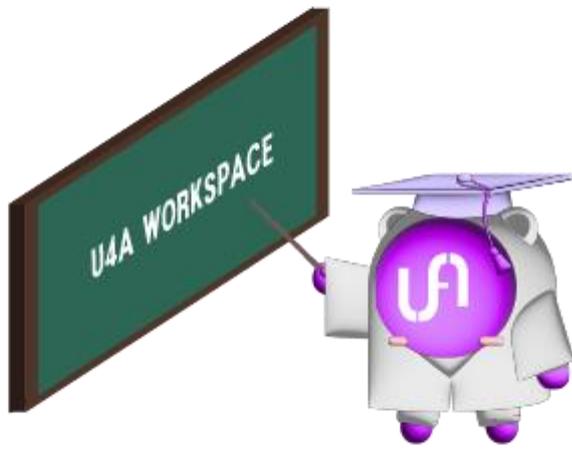
3. Workbench Toolbar 영역에는 다음의 기능으로 구성되어 있음.

- ① Create : 신규 Application 생성 기능

- ② **Change** : 기존 개발된 Application 변경 기능
- ③ **Delete** : 기존 개발된 Application 삭제 기능
- ④ **Copy** : 기존 개발된 Application 복사 기능
- ⑤ **Display** : 기존 개발된 Application 개발 사항 조회 기능
- ⑦ **Application Execution** : 기존 개발된 Application Browser를 통한 실행 기능
- ⑧ **Example Open** : Application 예제 조회 기능
- ⑩ **App Multi Preview** : 기존 개발된 Application Browser를 통한 Simulation 실행 기능

4. 신규, 변경, 삭제, 복사, Package 변경 등 (1)항과 (3)항에서 기능 수행 시 필수 입력 사항인 Application ID 화면 Field 임.

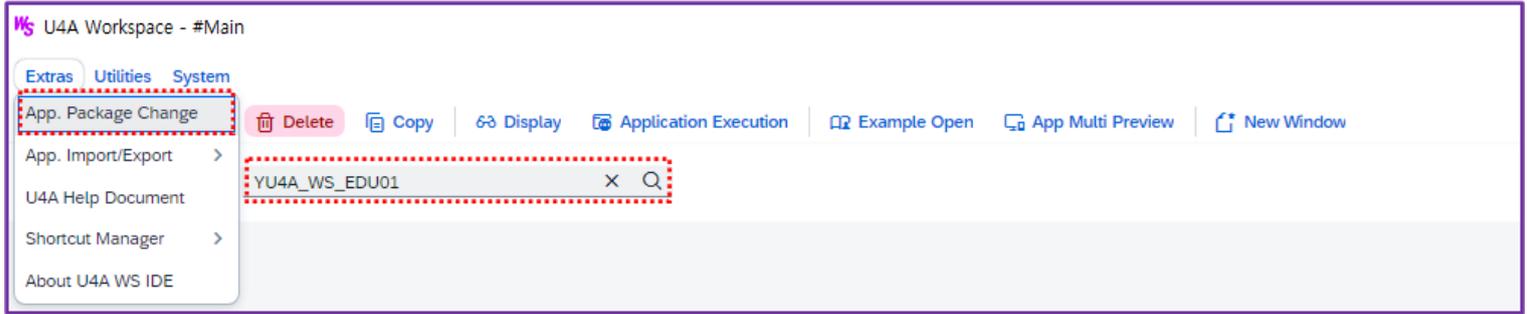
4.3 추가 기능 메뉴 - Extras



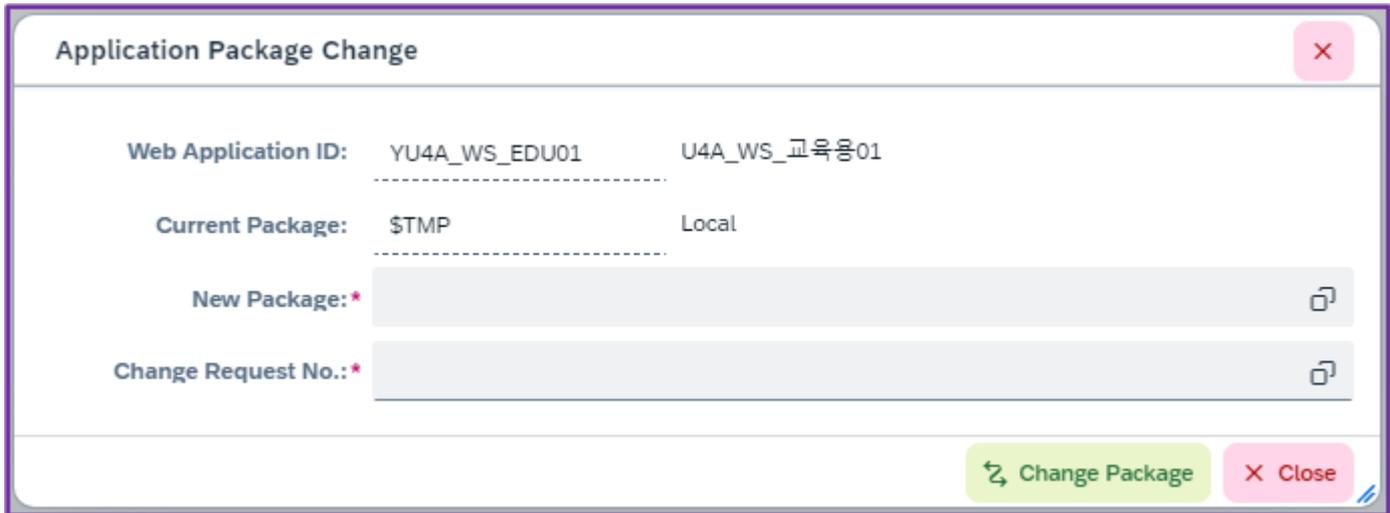
4.3.1 App. Package Change

■ 목적 : Application의 Package를 변경하기 위한 목적.

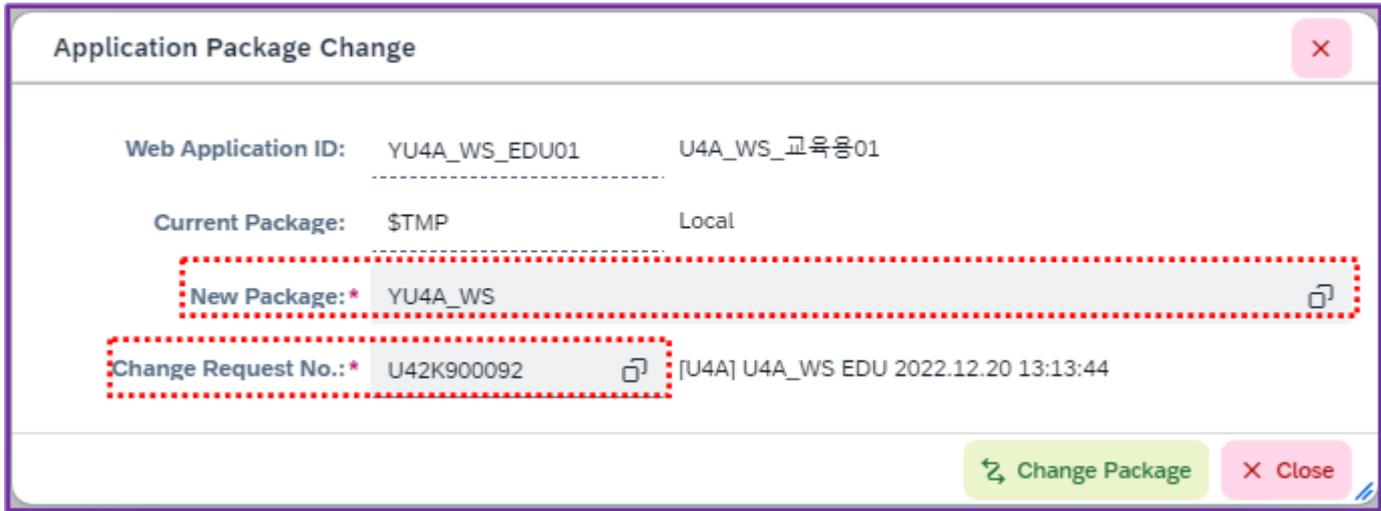
1. Package 변경 할 Application ID(Name)을 입력 후 " Extras -> App. Package Change" 선택 함.



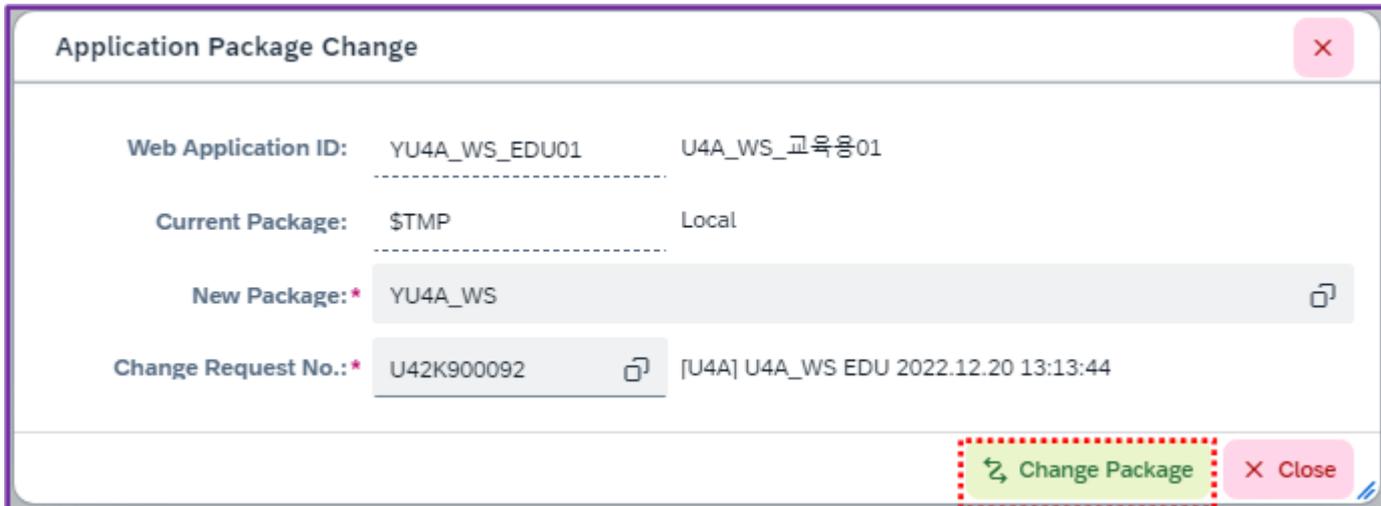
2. Package 변경 정보를 입력할 수 있는 Popup 창 생성 .



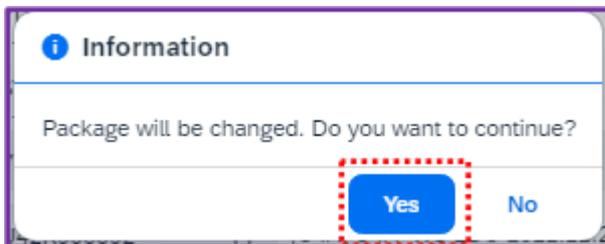
3. 변경 하려는 Package Name, CTS 번호 입력



4. Package Name, CTS 번호 확인 후 "Change Package" Button Click.



5. 확인 Popup에서 "Yes" Button Click.



6. Object Navigator에서 Package 변경 된 Application 확인

Package

YU4A_WS

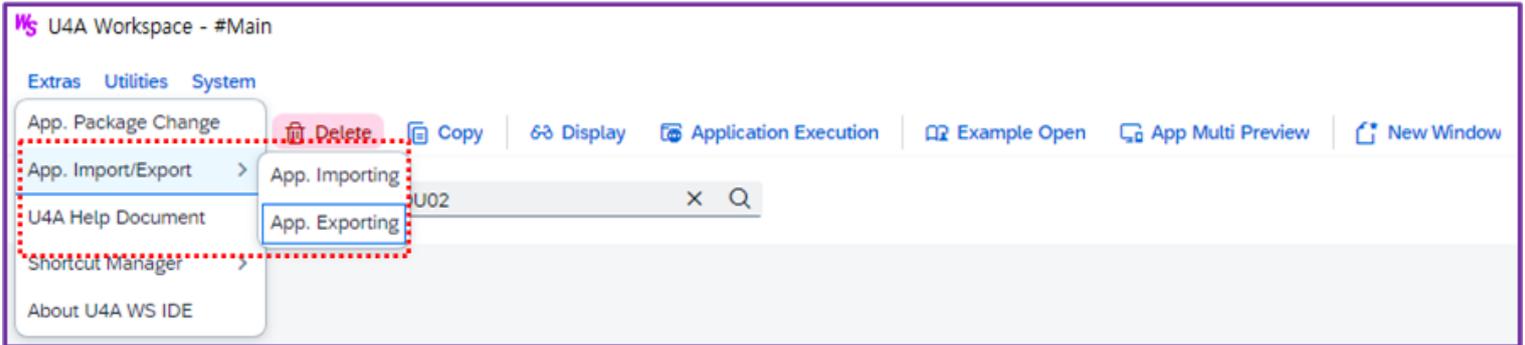
Object Name | Description

- YU4A_WS [U4A] PACKAGE
 - Class Library
 - Classes
 - YCL_U4A_APP_U4A_WS_EDU01 U4A_WS 교육용01

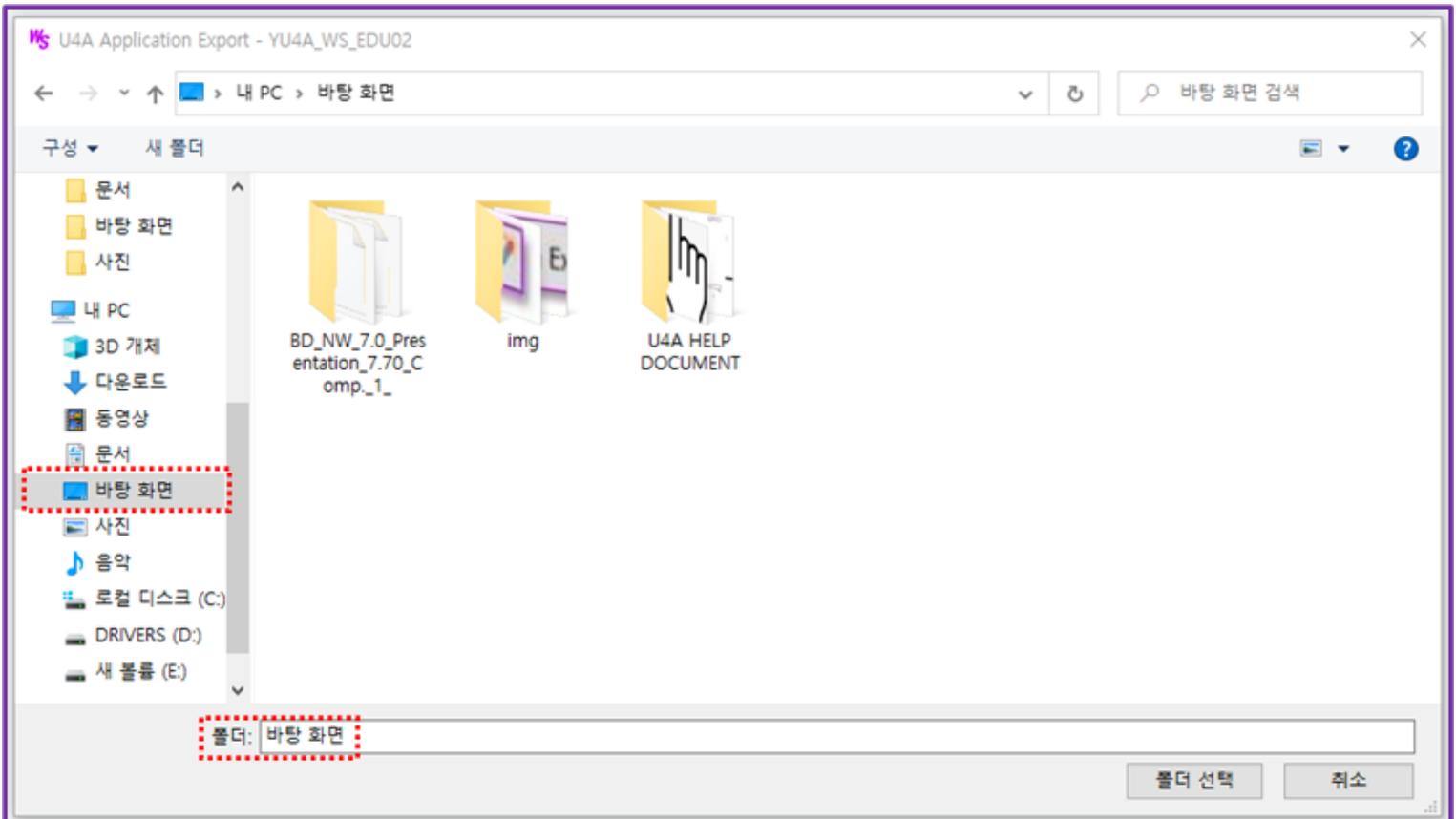
4.3.2 App Import/Export

■ 목적 : Application을 U4A File로 저장 및 불러오기를 하기 위한 목적 .

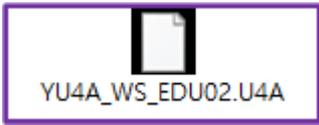
1. Exporting 할 Application ID(Name)을 입력 후 " Extras -> App. Import/Export -> App. Exporting" 선택 함.



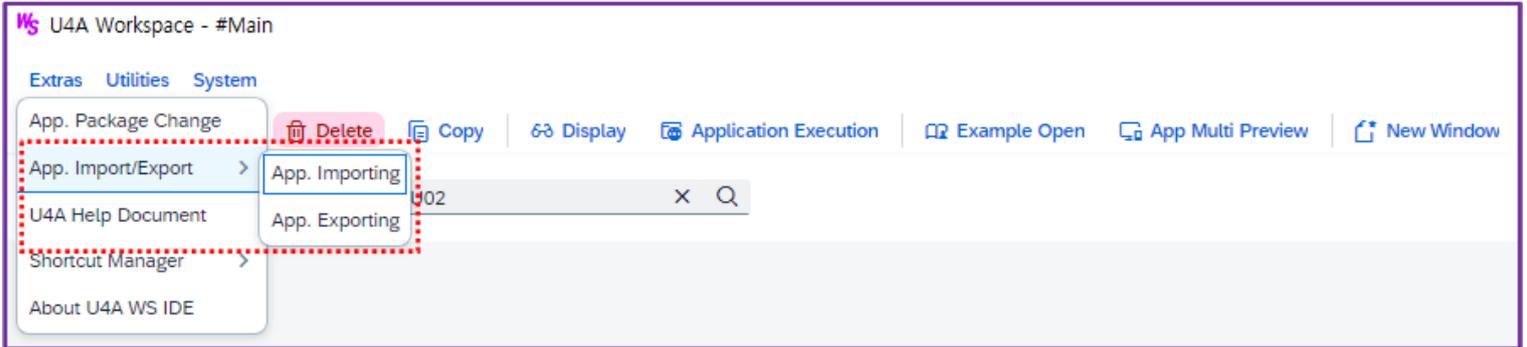
2. Exporting하여 저장 할 위치를 선택한 후 폴더 선택 Button Click.



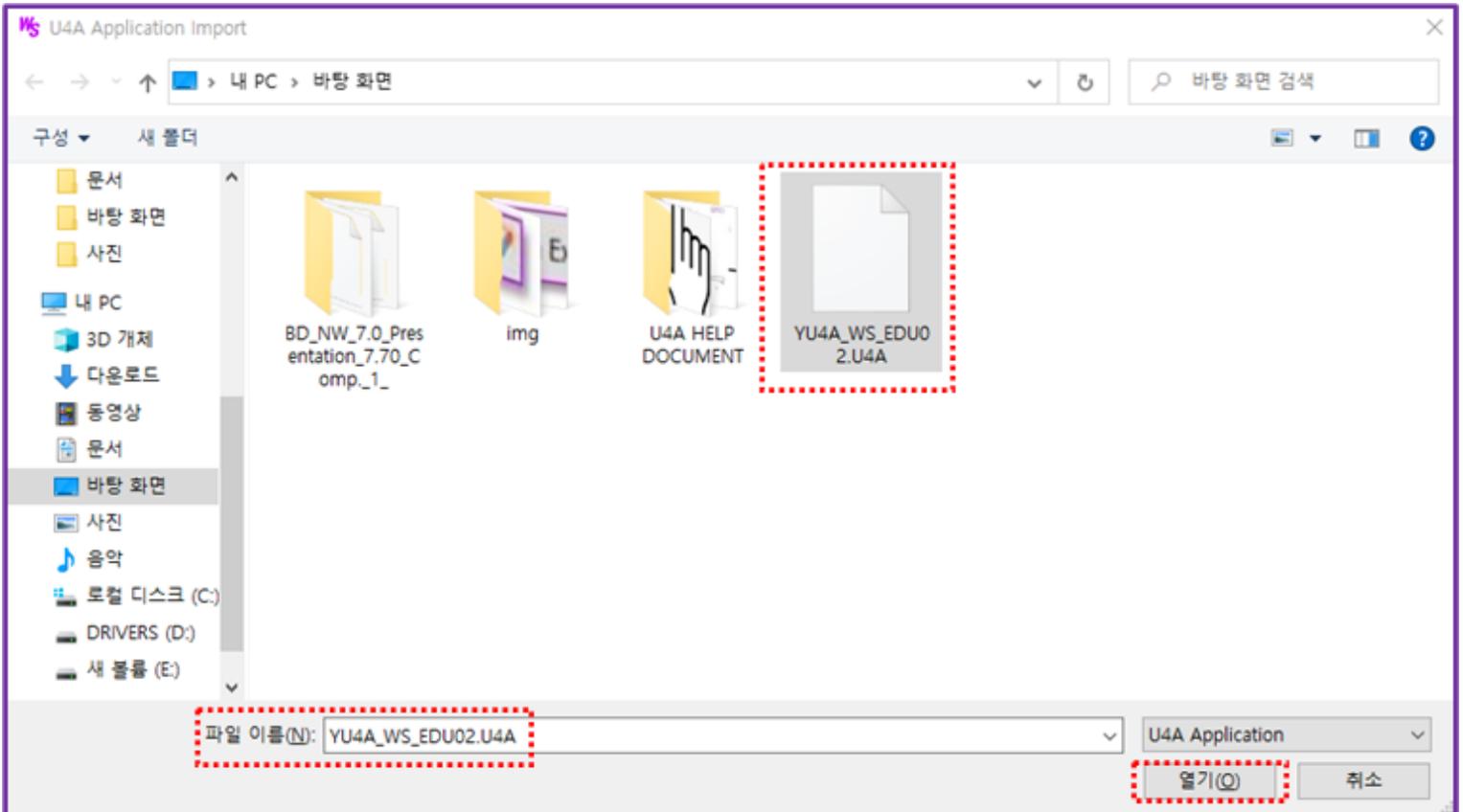
3. 저장한 위치에서 Exporting 된 Application 확인 (확장자 : U4A)



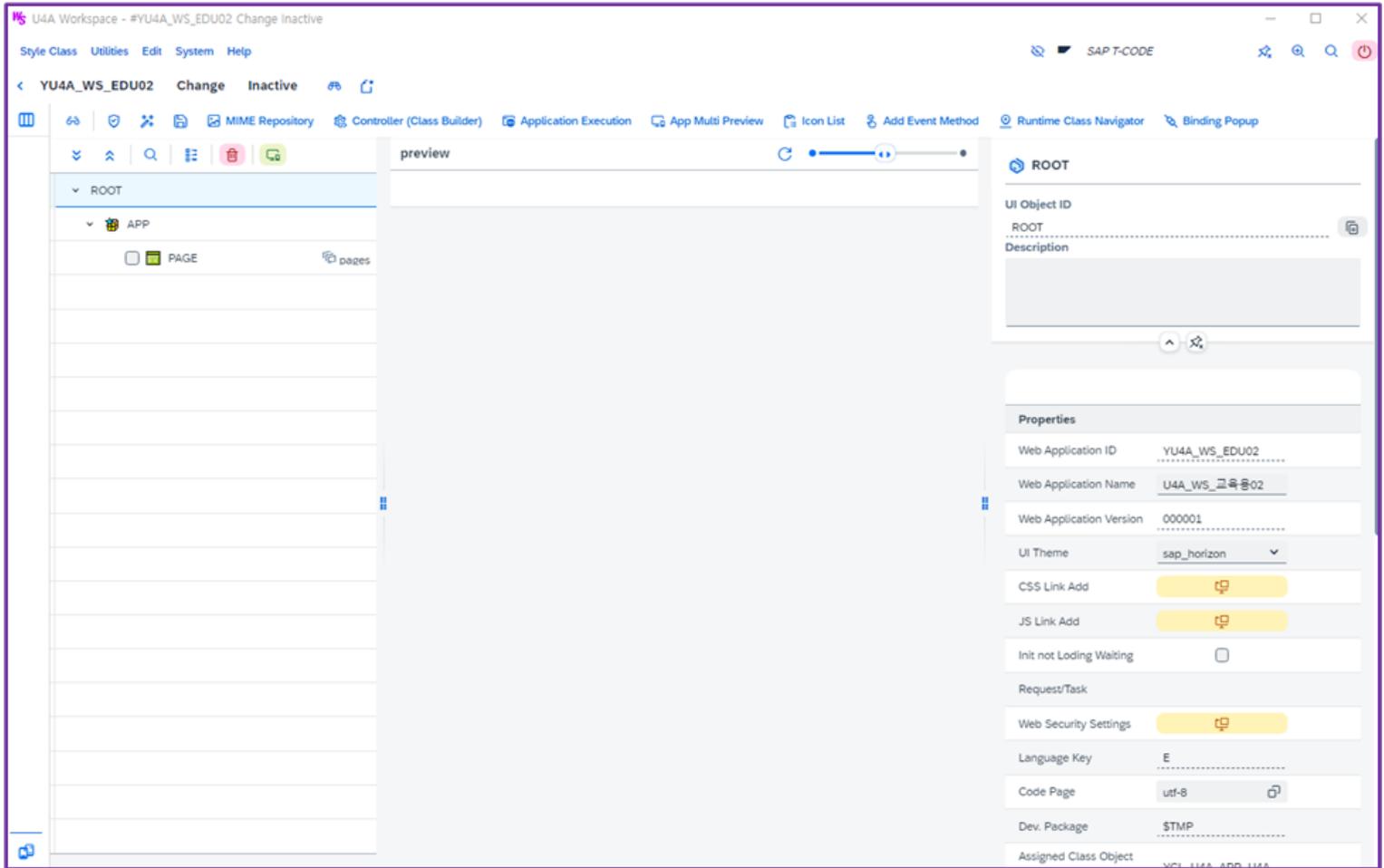
4. " Extras -> App. Import/Export -> App. Importing" 선택 함.



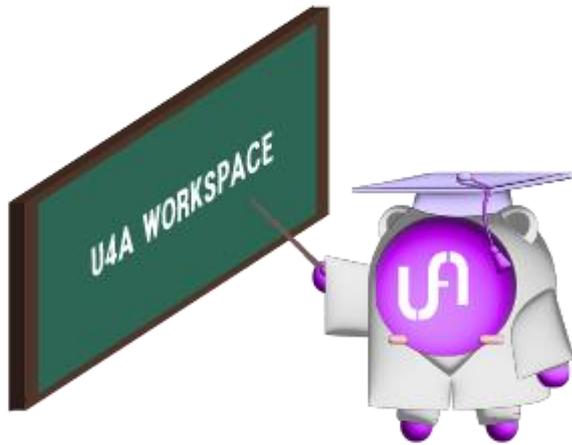
5. Importing 할 Application 선택 후 열기 Button Click .



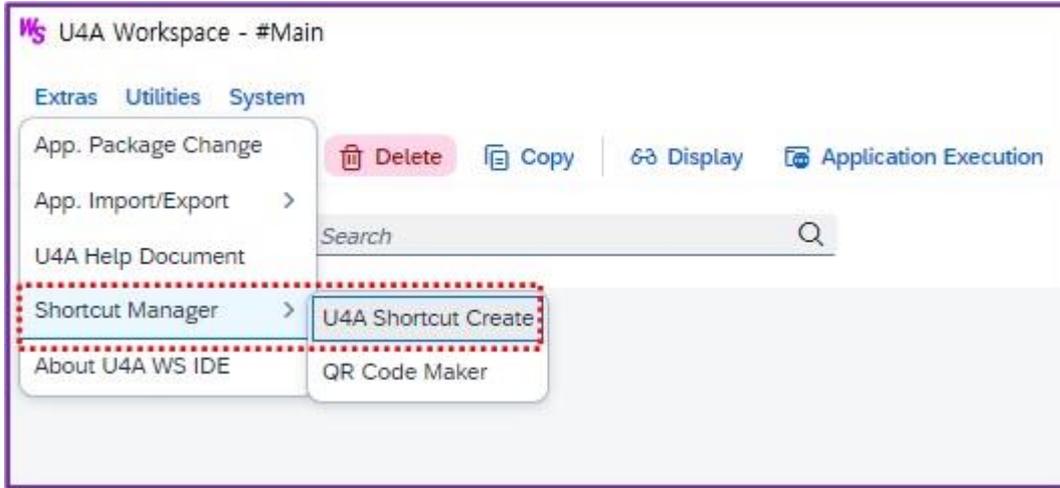
6. Importing 한 Application 확인 .



4.3.3 Shortcut Manager



4.3.3.1 U4A Shortcut Create



2. "Shortcut Creator" Popup이 표시됨 .

The screenshot shows the 'APP Shortcut Creator' window with the following sections:

- General Information:**
 - APP ID: * U4A Application ID (with a copy icon)
 - APP Name:
 - File Name: Shortcut File Name
 - Save As Path: * Shortcut Download Path (with a copy icon)
 - Icon Path: Shortcut ICON Path (with a copy icon)
- Target Host URL:**
 - Radio buttons for Internal Host URL (selected) and External Host URL.
 - A text input field with a blacked-out area.
- Browser Type:**
 - Radio buttons for Chrome Browser (selected) and IE edge Browser.
- Browser Option(Chrome):**
 - Radio buttons for App Mode (selected), Full Screen Mode, and Kiosk Mode.
 - Checkboxes for Displible Translate Mode and Secret Mode.
- Additional Parameter(Optional):**
 - Buttons for adding (+) and removing (-) parameters.
 - A table with columns 'Name' and 'Value'.
 - Current state: No data

At the bottom right, there are 'Save ShortCut' and 'Cancel' buttons.

3. "APP ID" 입력란에 Shortcut을 생성 할 App ID 입력 또는 가능엔트리 선택.

General Infomation

APP ID: * YU4A_WS_EDU01 

APP Name:

File Name: *Shortcut File Name*

Save As Path: * *Shortcut Download Path* 

Icon Path: *Shortcut ICON Path* 

4. Shortcut File을 저장할 로컬 경로 설정 .

General Infomation

APP ID: * YU4A_WS_EDU01 

APP Name:

File Name: *Shortcut File Name*

Save As Path: * C:\TEMP1 

Icon Path: *Shortcut ICON Path* 

5. 만약 Shortcut File의 심볼 Icon 설정 필요 시, "Icon Path"에 지정함. (※ 확장자는 반드시 ico여야 함) .

General Information

APP ID: * YU4A_WS_EDU01 

APP Name:

File Name: *Shortcut File Name*

Save As Path: * C:\TEMP1 

Icon Path: C:\TEMP1\Folder.ico 

6. Application Host URL 설정 .

- ① Internal Host URL : Default 값은 해당 로그인 서버 SAP Host URL 임.
- ② External Host URL : 연결 가능한 외부 Domain URL을 직접 입력.

Target Host URL

Internal Host URL External Host URL



7. Application 실행 시 Browser 선택. (※ Default : Chrome Browser)

7-1 Chrome Browser And Option.

Browser Type

Chrome Browser IE edge Browser

Browser Option(Chrome)

App Mode Full Screen Mode Kiosk Mode

Displble Translate Mode Secret Mode

7-2. IE edge Browser And Option

Browser Type

Chrome Browser IE edge Browser

Browser Option(IE edge)

App Mode Normal Mode

8. 만약 Application 실행 시 추가 파라미터가 필요하면 다음과 같이 지정 가능함 .

Additional Parameter(Optional)

<input type="checkbox"/> Name	Value
<input checked="" type="checkbox"/> USER	U4A

9. Shortcut 생성을 위한 정보를 모두 입력 하였다면 "Save Shortcut" Button 클릭함 .

APP Shortcut Creator

General Infomation

APP ID: * YU4A_WS_EDU01

APP Name:

File Name: Shortcut File Name

Save As Path: * C:\TEMP1

Icon Path: C:\TEMP1\Folder.ico

Target Host URL

Internal Host URL External Host URL

██████████

Browser Type

Chrome Browser IE edge Browser

Browser Option(Chrome)

App Mode Full Screen Mode Kiosk Mode

Displbie Translate Mode Secret Mode

Additional Parameter(Optional)

+ -

Name	Value
<input type="checkbox"/> USER	U4A

Save ShortCut Cancel

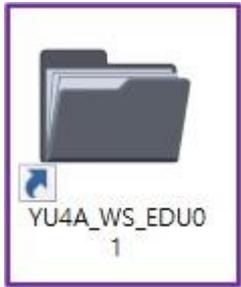
10. Shortcut Create 확인 팝업 "OK" Button Click.

Confirmation

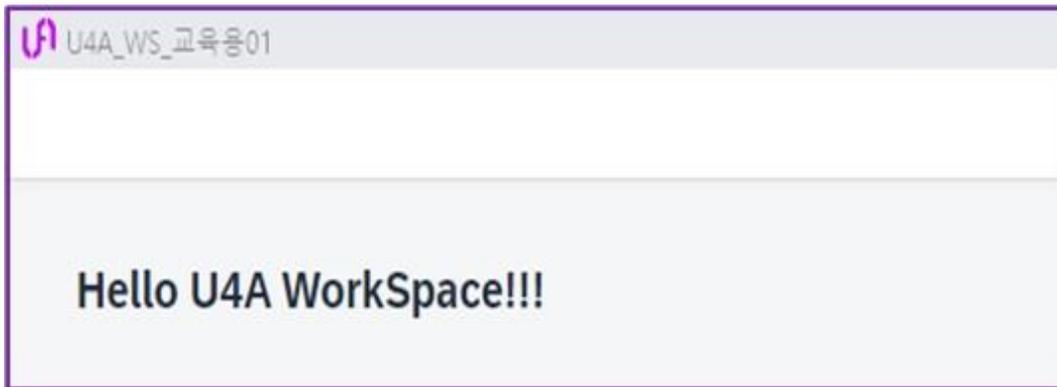
Are you sure you want to proceed with creating a shortcut?

OK Cancel

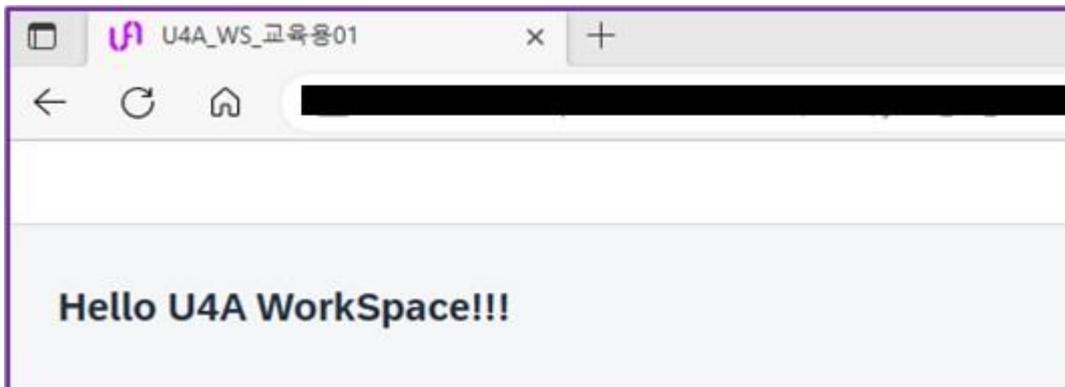
11. 그림(4)에서 지정한 로컬 경로에 생성된 Shortcut File 확인 할수 있으며, 해당 아이콘을 Double Click하여 실행 .



12. Chrome Browser에서 정상적으로 실행된 화면. (※ 옵션 : App Mode, Full Screen Mode, Kiosk Mode, Disable Translate Mode, Secret Mode,)



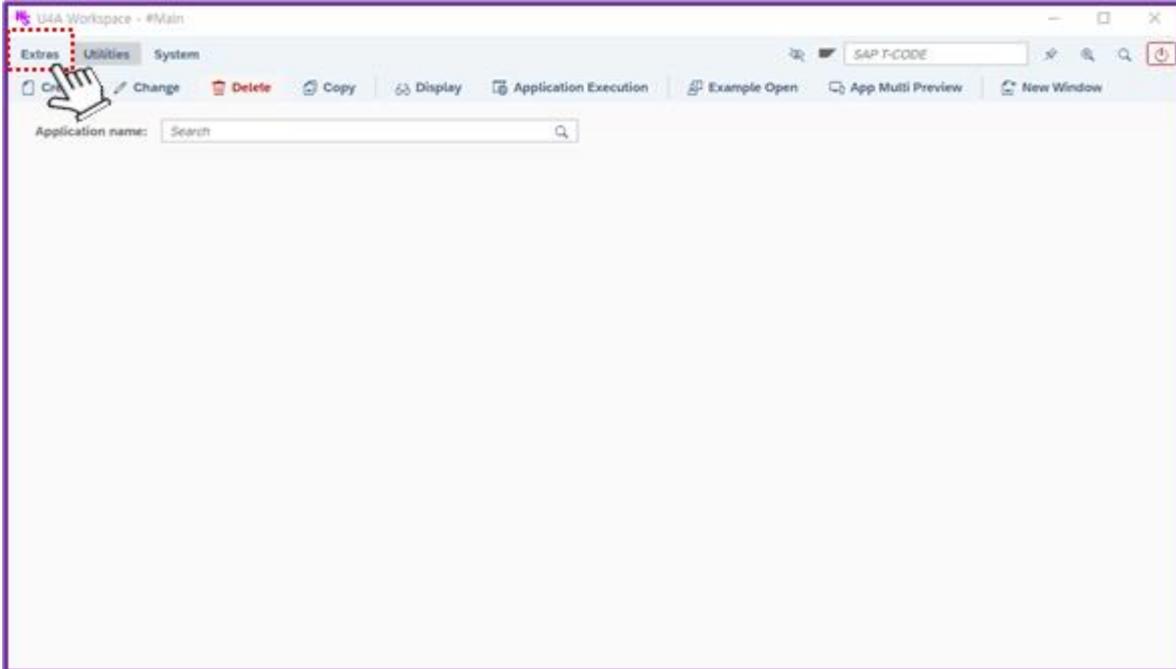
13. IE edge Browser 에서 정상적으로 실행된 화면. (※ 옵션 : Normal Mode)



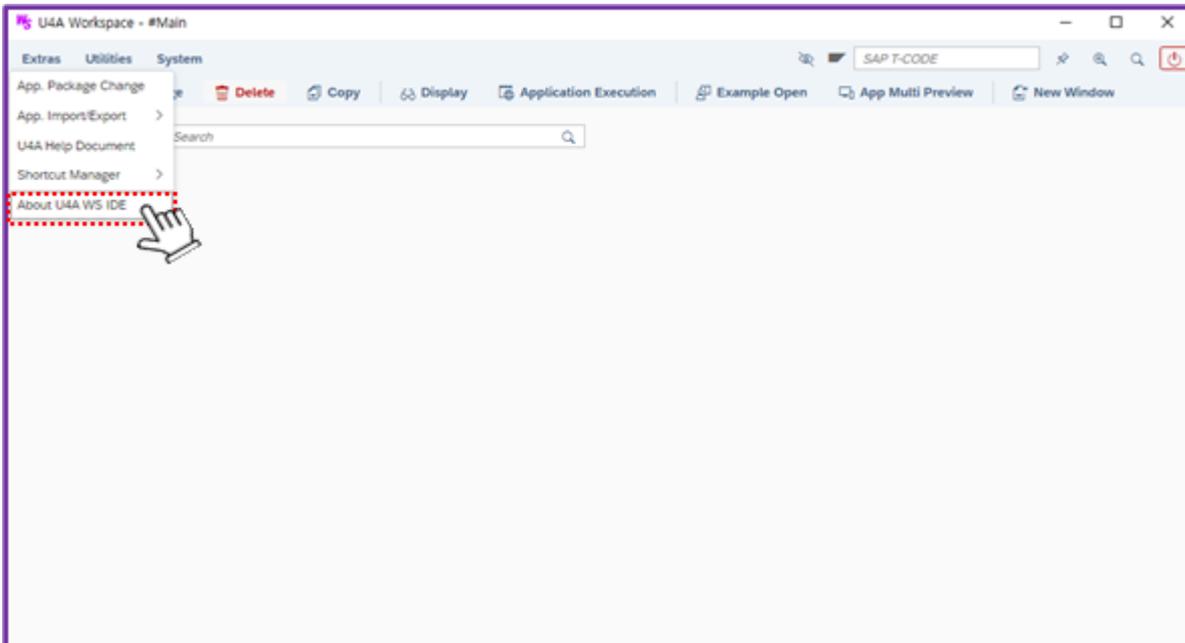
 U4A Shorcut 기능은 Microsoft Windows OS만 지원함.

4.3.4 U4A Workbench 버전 확인

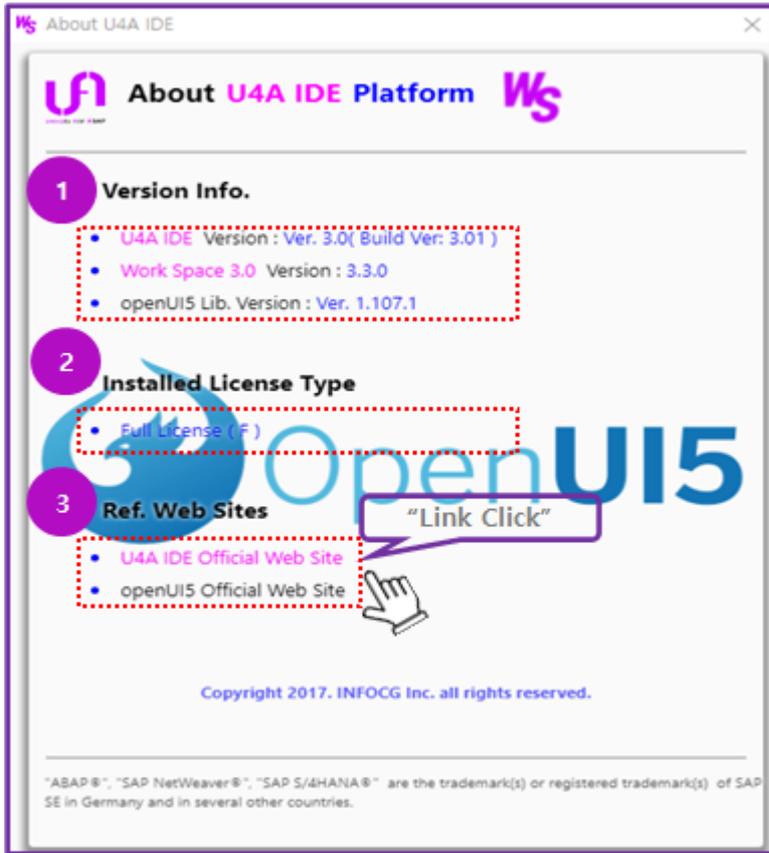
1. 좌측 상단 메뉴 중 “Extras” 항목 Click.



2. “Extras” Server Menu 중 “About U4A WS IDE” 항목 Click.



3. Popup으로 U4A IDE Platform 정보가 표시 됨.



(1) U4A IDE “Version Info.”는(은) 하기와 같이 세가지 의미로 표기됨.

① U4A IDE Version

U4A IDE Platform 설치 Version을 의미 함.

※ U4A IDE Version Rules

- . Major Release : x .y (Build Ver: z.zz)
- . Minor Release : x.y (Build Ver: z.zz)
- . Patch Release : x.y (Build Ver: z.zz)

② Work Space 3.0

Work Space 3.0 설치 Version을 의미 함.

③ openUI5 Lib. Version

UI5 Application 화면 개발 시 사용되는 UI5 Library Version을 의미 함.

(2) 설치된 U4A IDE 라이선스 유형을 의미함

(3) 추가 정보 Web Site(Ref. Web Sites)는 다음과 같음 (Link Click 가능).

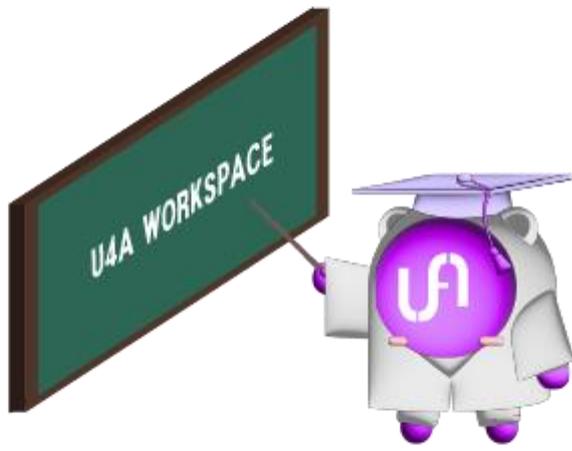
- ① U4A IDE Official Web Site (www.u4ainfo.com)

U4A IDE Platform 기술자료 및 App. Template 등을 제공함.

② openUI5 Official Web Site (www.openui5.org)

openUI5 공식 Site 이며, 관련 기술의 상세 정보 및 예제 정보를 확인 할 수 있음.

4.4 추가 기능 메뉴 - Utilities

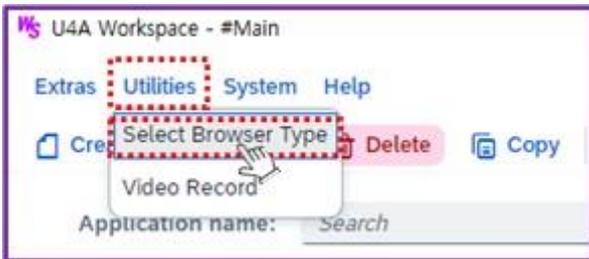


4.4.1 Select Browser Type

▣ 목적 : U4A Application 실행 시 기본 Browser를 지정 하기 위함.

1. 그림과 같이 추가 기능 메뉴 영역에서 "Utilities -> Select Browser Type"을 선택함. (선택 방법은 2가지 이지만 사용방법은 동일하다)

① U4A WorkSpace 메인 화면에서 선택

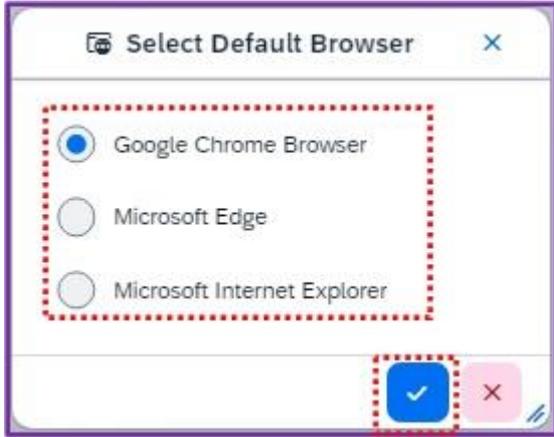


② Application 안에서 선택

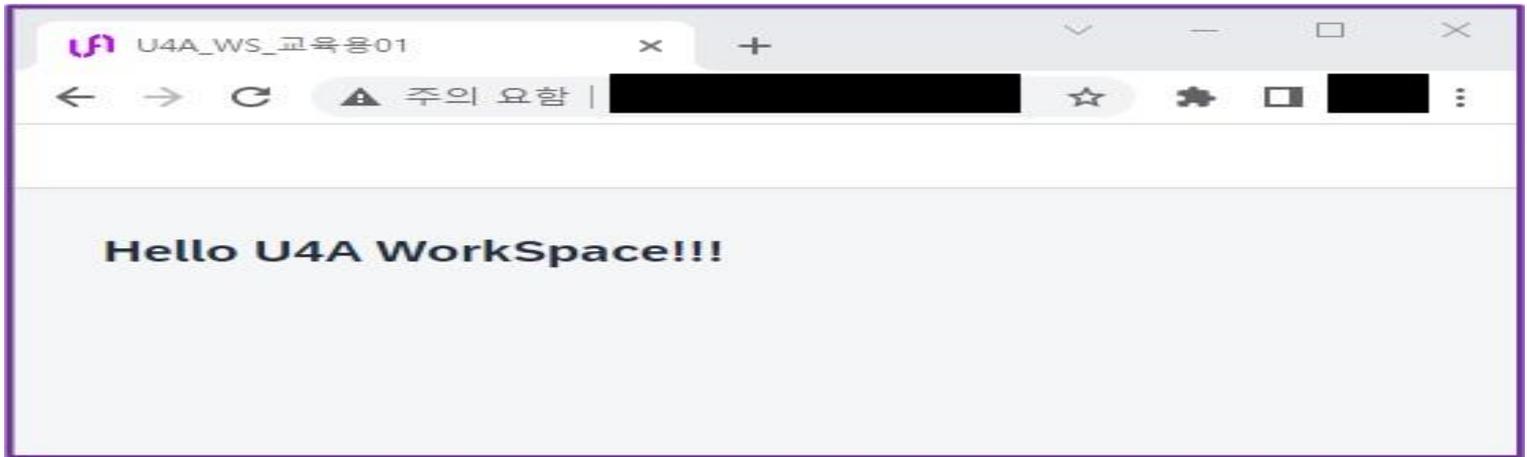


2. 활성화된 Popup 에서 표시된 Browser 중 기본 Browser 설정을 원하는 것을 선택 후 "√" Button을 클릭함.

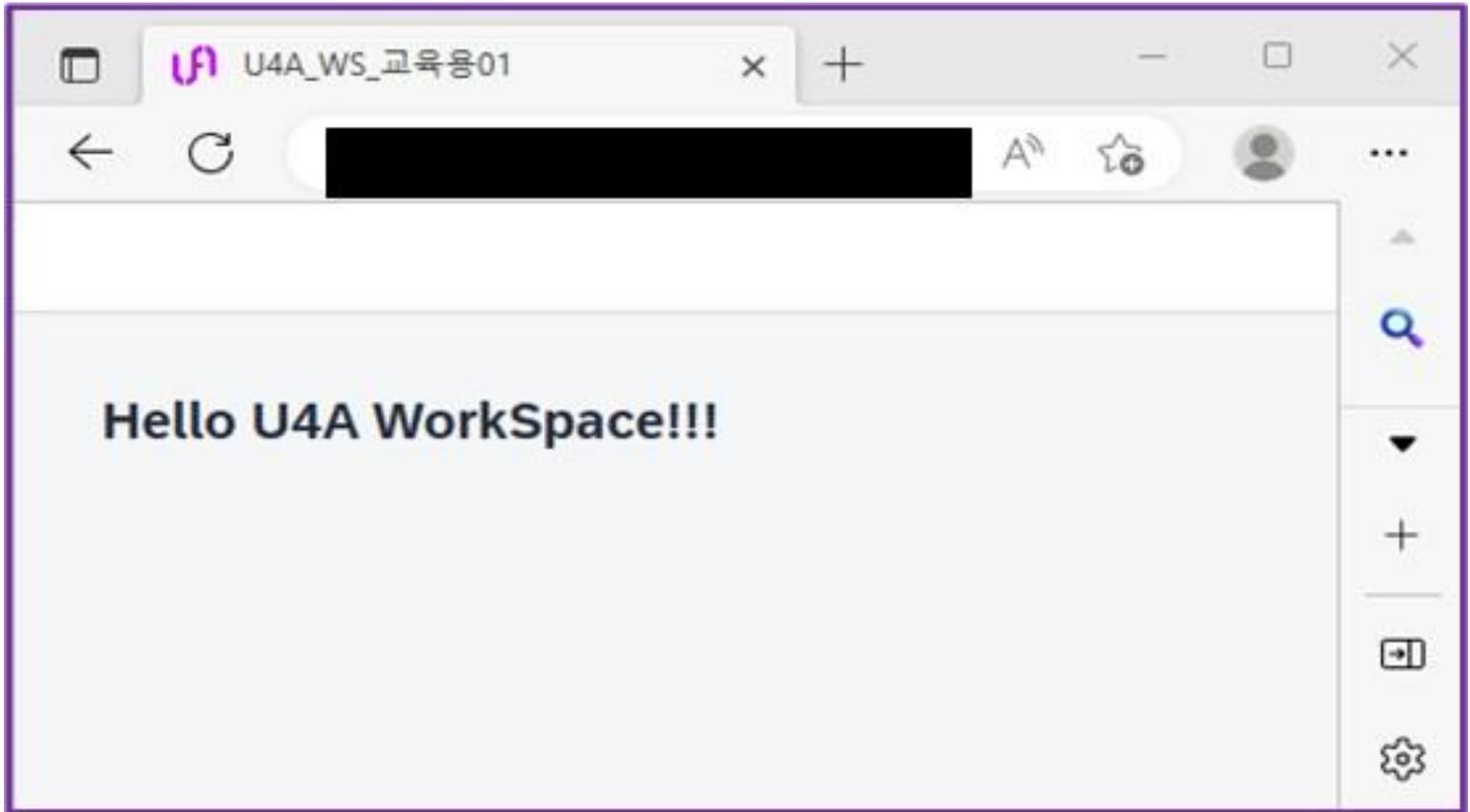
초기 Default Browser => "Google Chrome Browser".



2-1 "Google Chrome" Browser로 실행된 경우



2-2 "Microsoft Edge" 로 실행된 경우



2-3 "Microsoft Internet Explorer" 로 실행된 경우

즉, 해당 Application의 Cross Browsing 테스트 시 실행 URL 복사 후 각각의 Browser 수동 실행 및 복사된 URL을 붙여넣기 후 실행 하는 번거로운 작업을 대체하는 편리한 기능.

단, 한번 기본 Browser로 지정되면 별도 다른 Browser로 변경 하기 전 까지 계속해서 해당 속성이 지속됨. (PC를 재부팅 해도 지속됨)

4.4.2 Video Record

■ 목적 : U4A WorkSpace 사용 중 화면 녹화 기능.

1. 상단의 Utilities = > Video Record 선택함 (선택 방법은 2가지 이지만 사용방법은 동일하다)

① U4A WorkSpace 메인 화면에서 선택



② Application 안에서 선택



2. Recording Controller Pop-Up Open

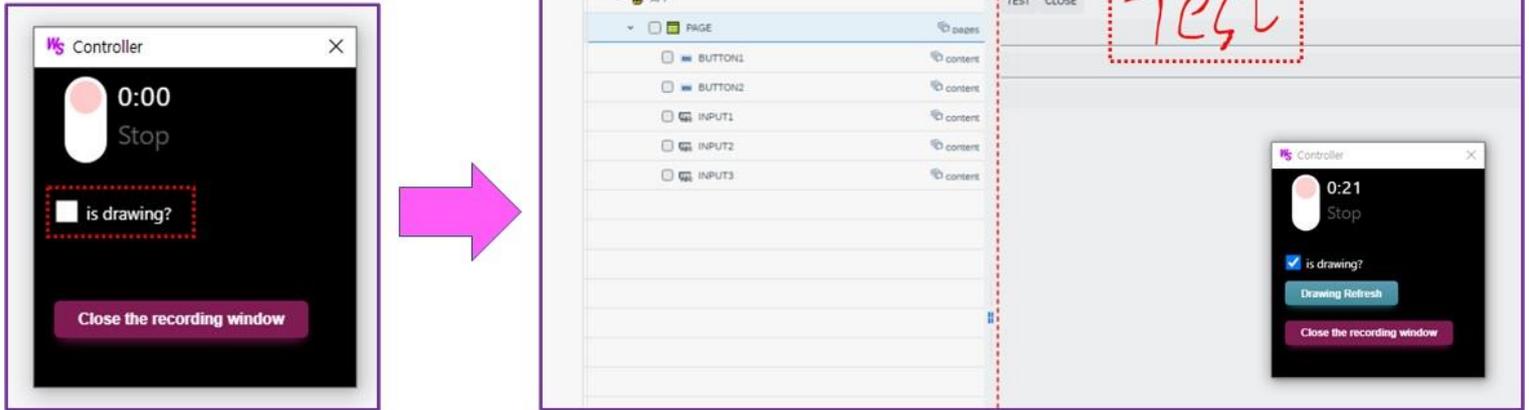


1) 스위치 전환을 통해 Record / Stop 설정

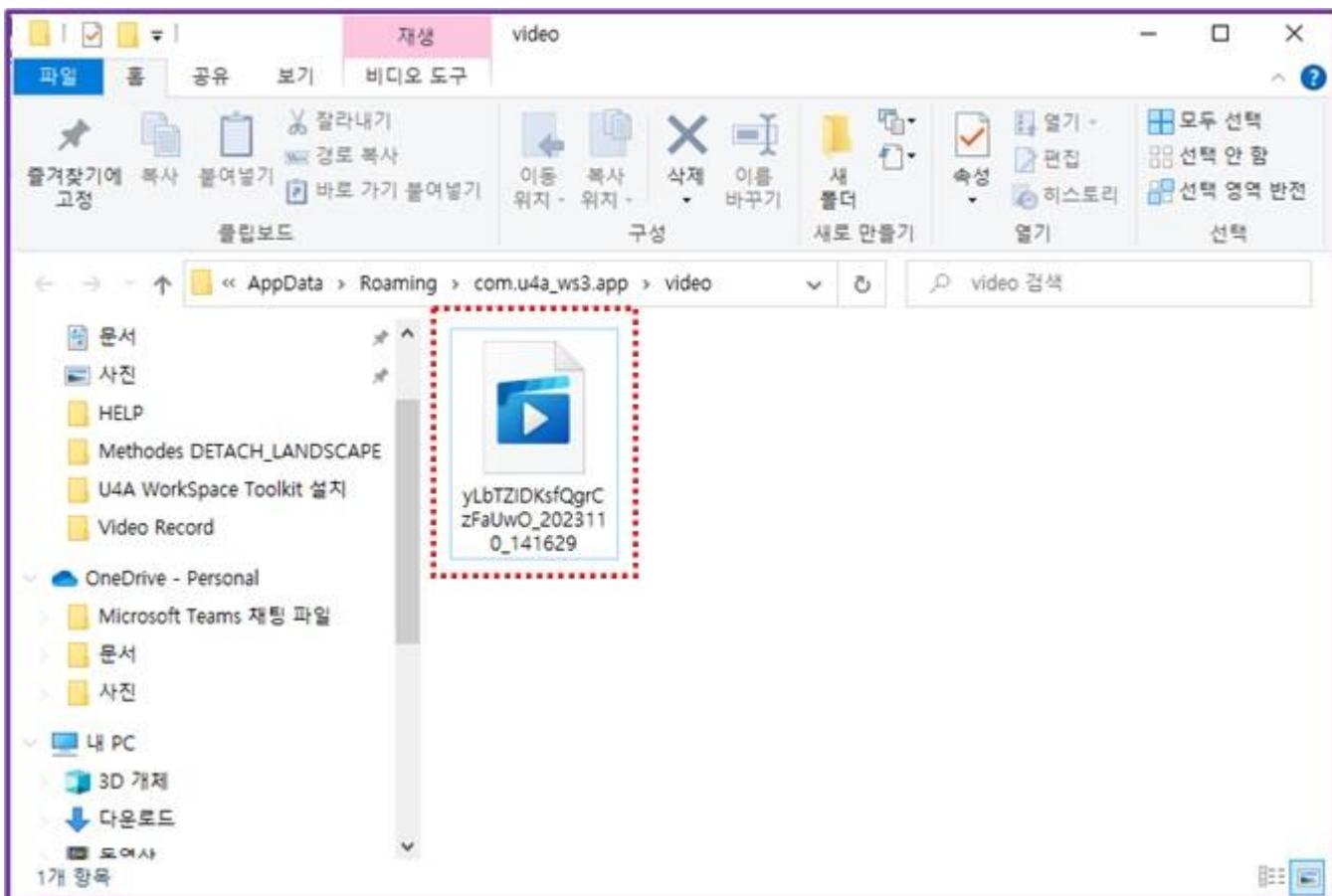
2) Recording Controller 팝업창 종료

3. (1) 스위치 "Record"로 전환하여 Recording 시작

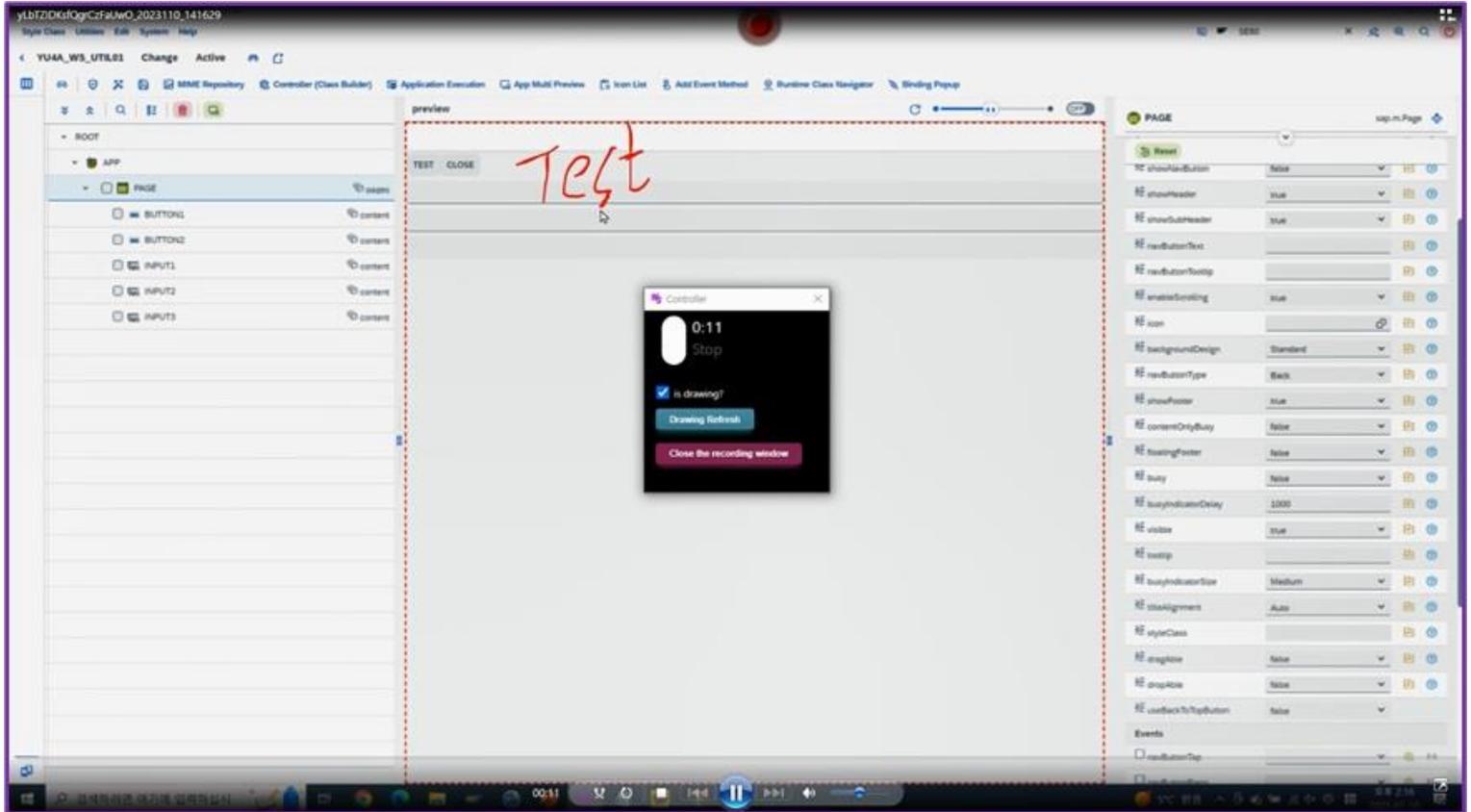
① Stop -> Record 변경시 "is drawing?" 체크박스에 체크시 녹화중 화면에 Drawing 가능



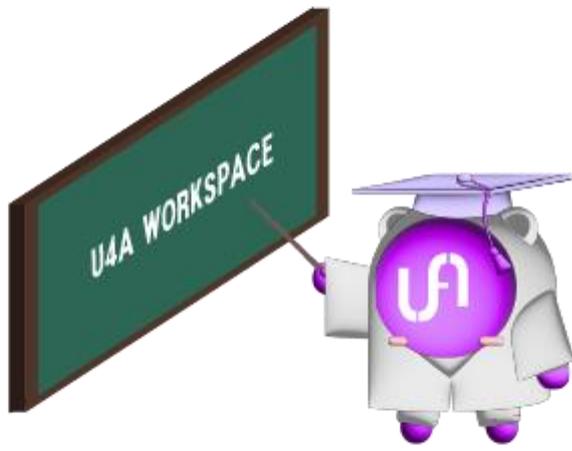
4. 다시 스위치 "Stop"으로 전환하면 녹화가 중지되며 녹화 Video 파일이 담긴 폴더 자동 실행



5. Video 파일 실행하여 녹화 화면 확인

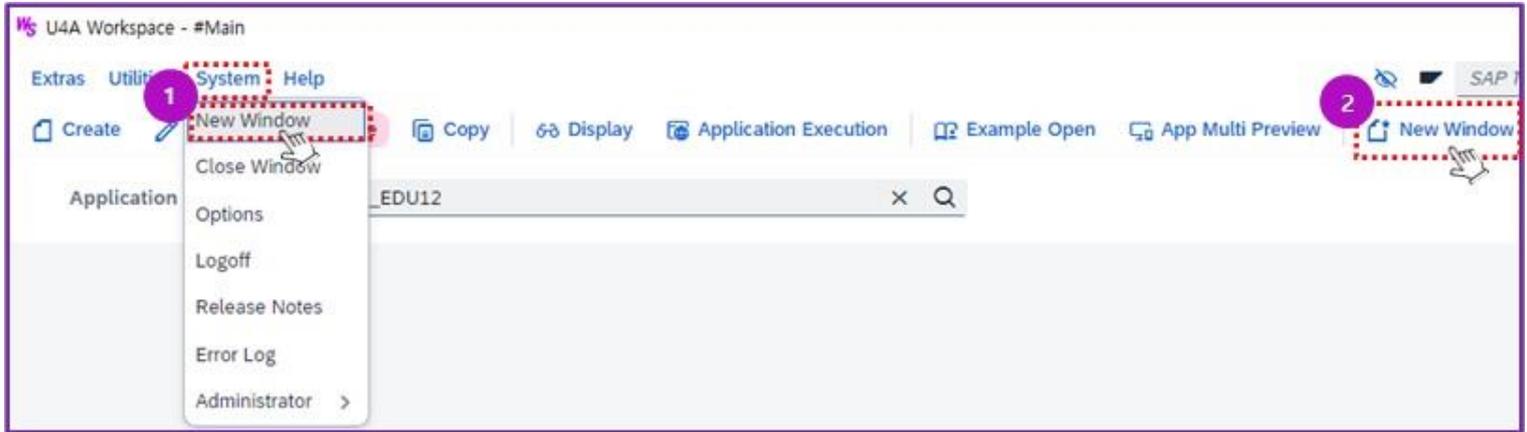


4.5 추가 기능 메뉴 - System

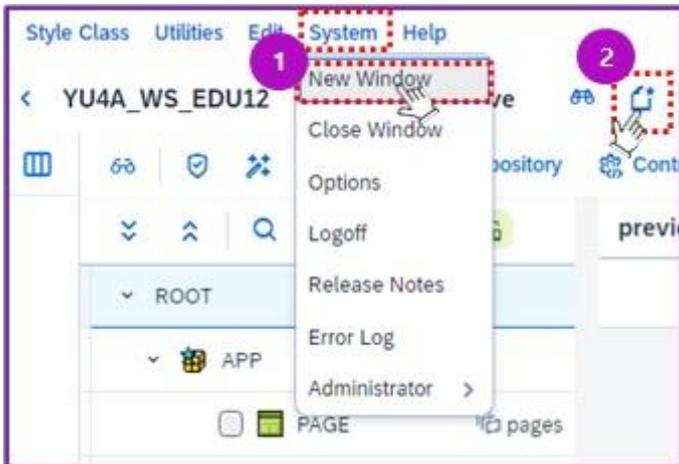


4.5.1 New Window

① U4A WorkSpace 메인 화면에서 선택

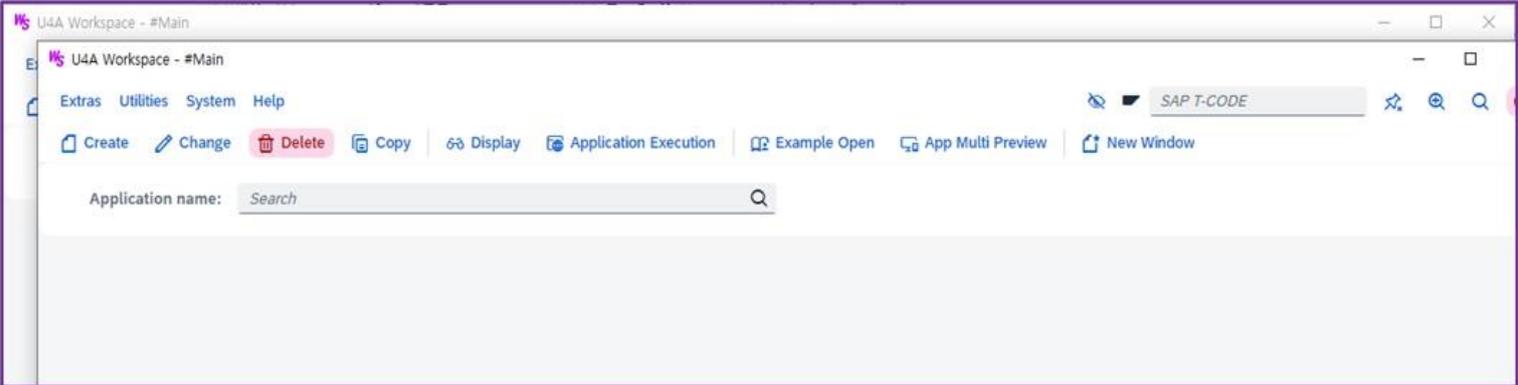


② Application 안에서 선택



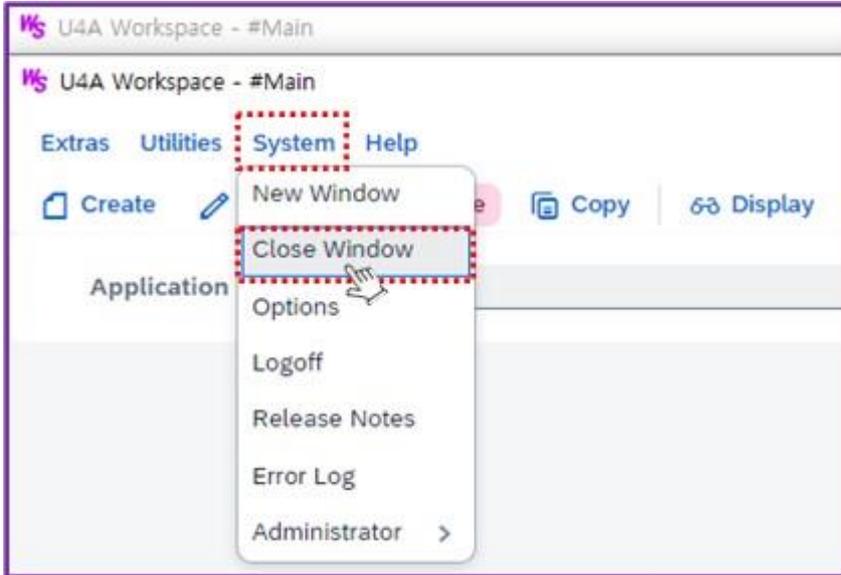
2. U4A WorkSpace 새 화면 확인.

U4A IDE HELP

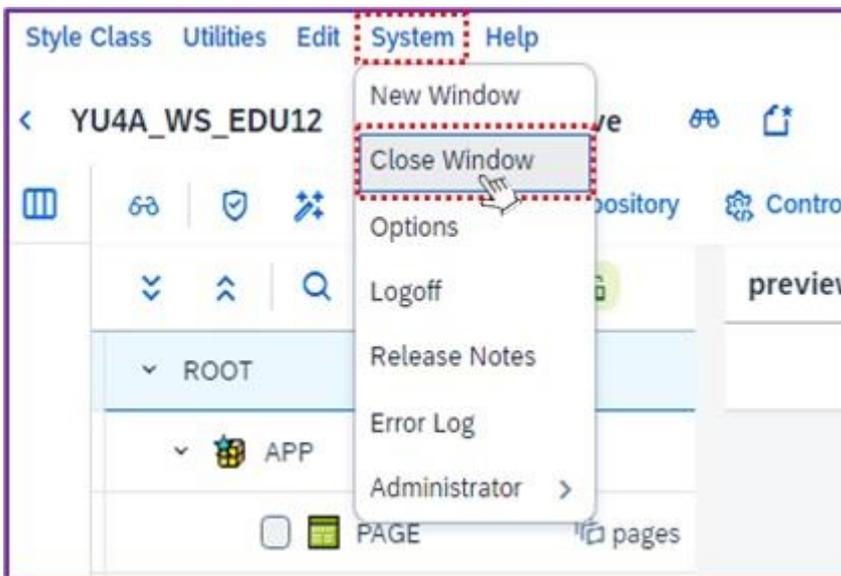


4.5.2 Close Window

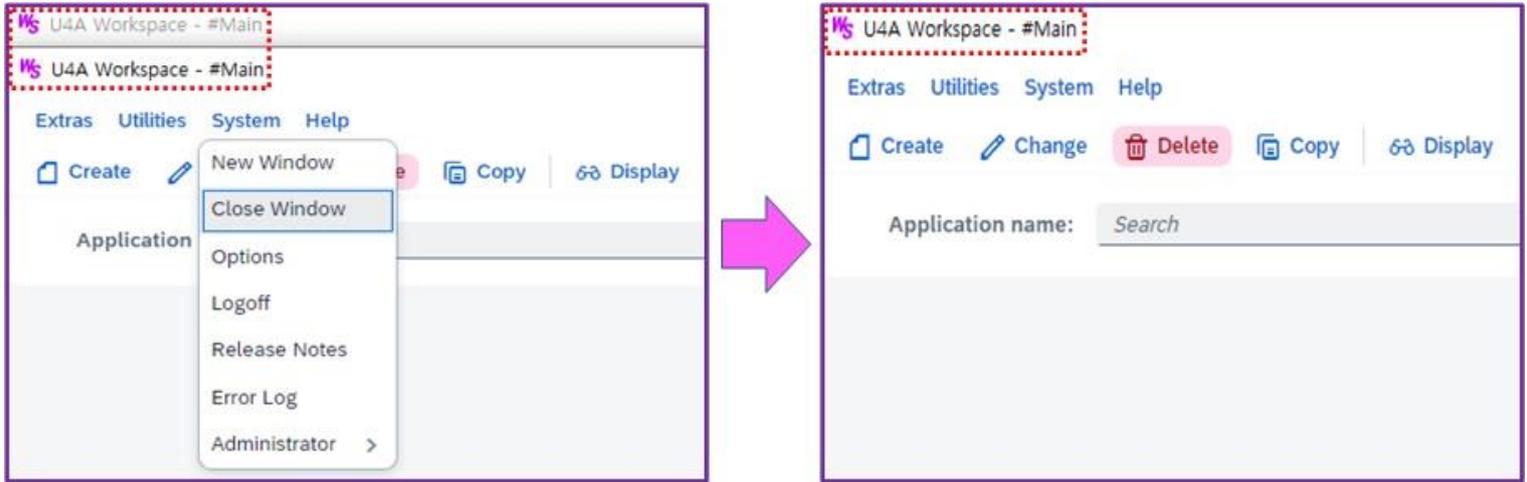
① U4A WorkSpace 메인 화면에서 선택



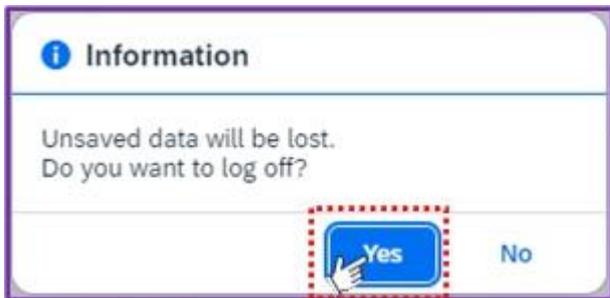
② Application 안에서 선택



2. U4A WorkSpce창 종료 확인(2개 -> 1개로 줄어듦)

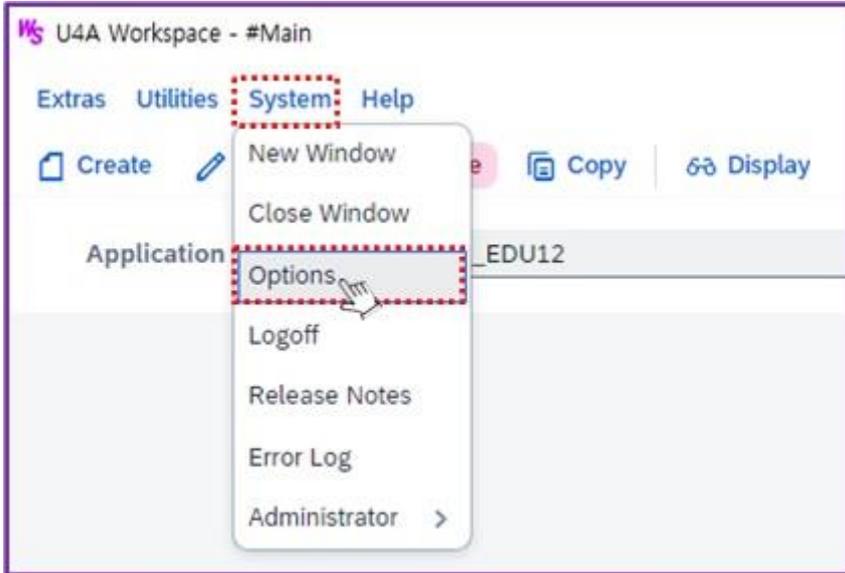


※ U4A WorkSpace창 1개인 상태에서 Close Window 선택시 아래와 같은 Pop-Up 발생하고 "Yes"버튼 누르면 종료

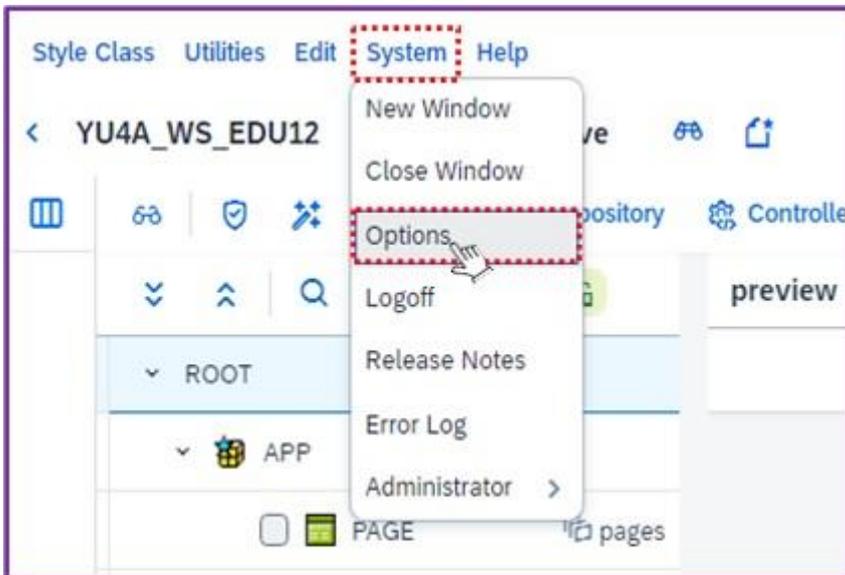


4.5.3 Options

① U4A WorkSpace 메인 화면에서 선택

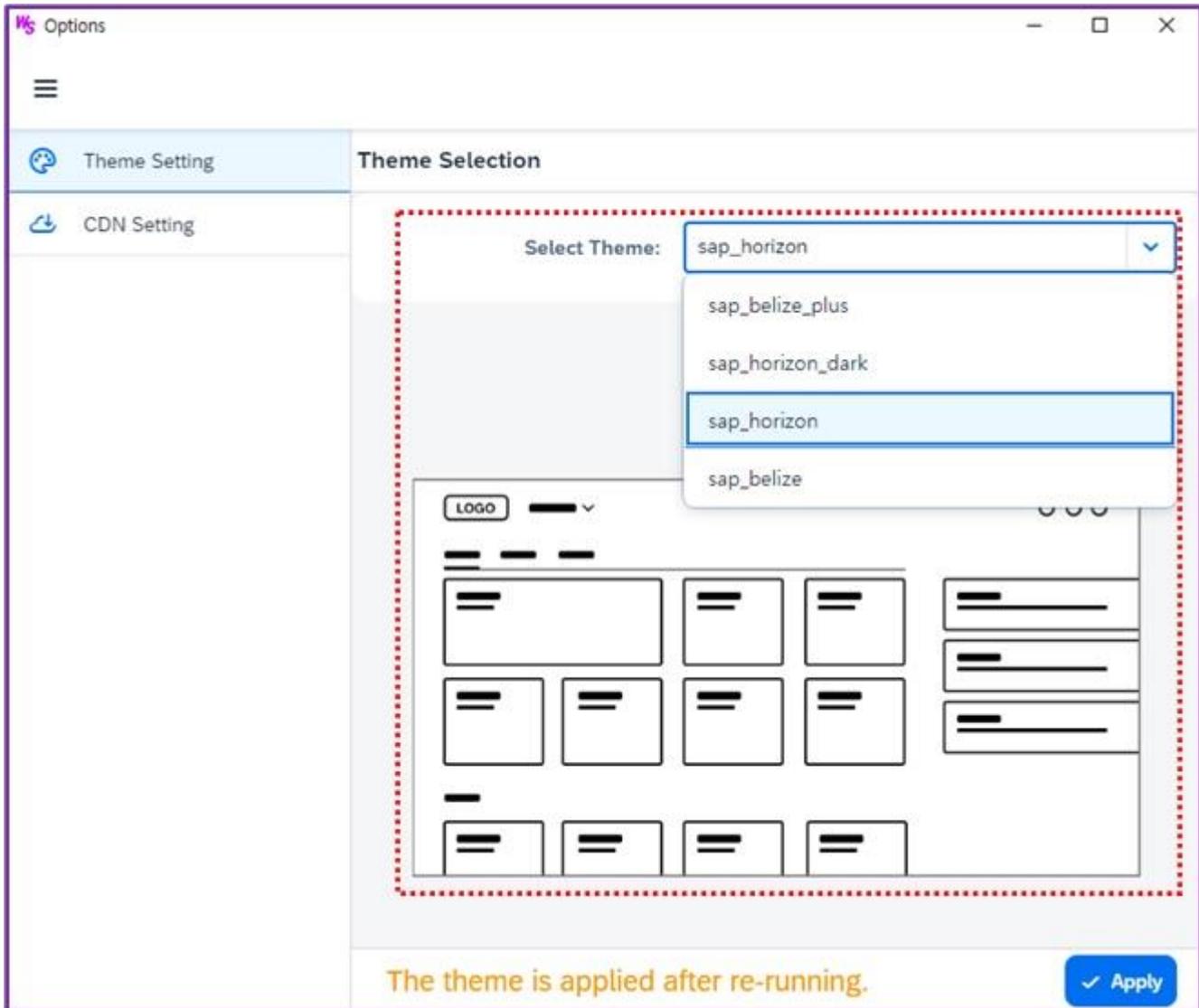


② Application 안에서 선택

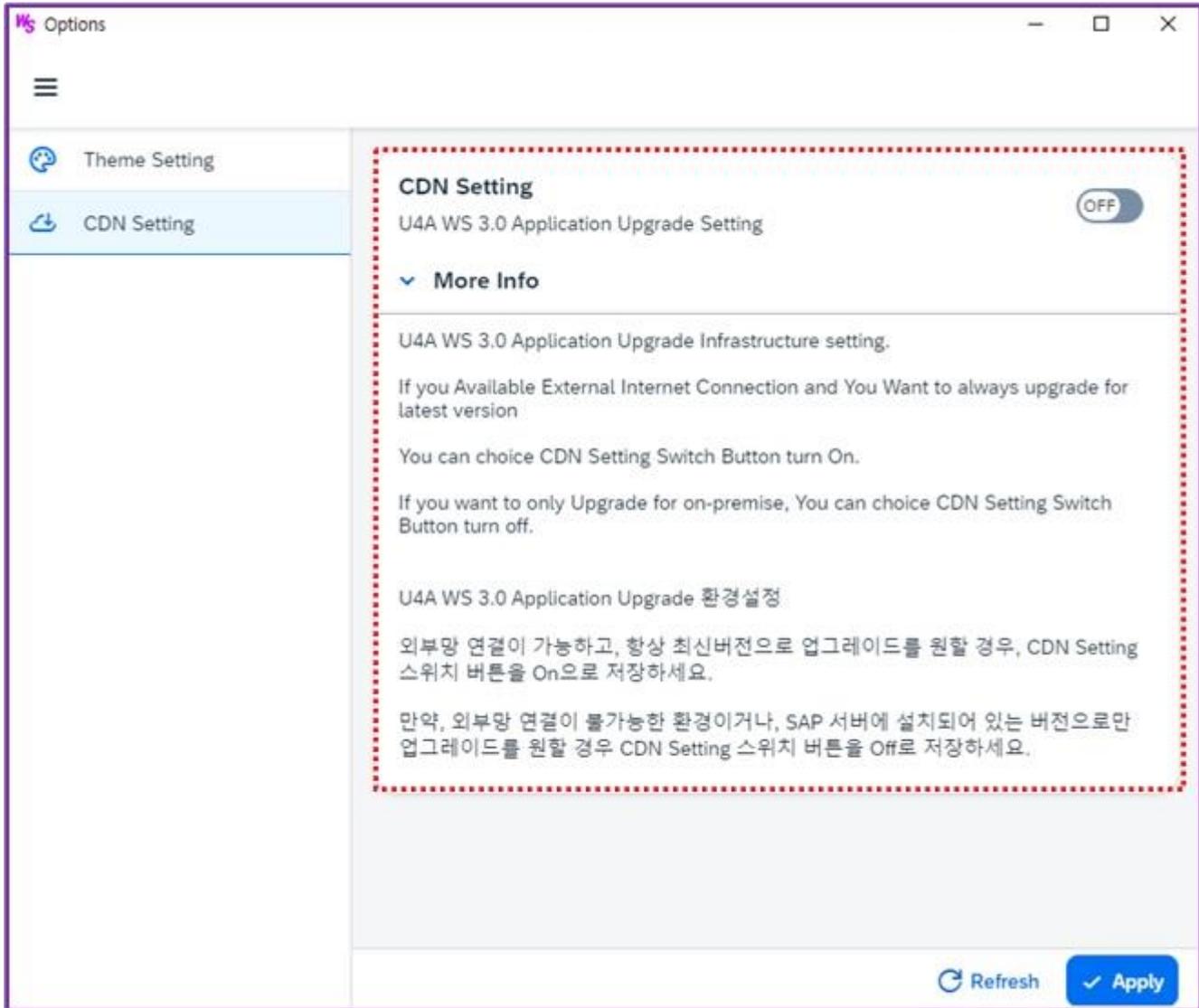


2. Options 개요

① Theme Setting (테마) => Theme는 현재 4개이며 원하는것을 선택한 후 Apply 버튼 클릭



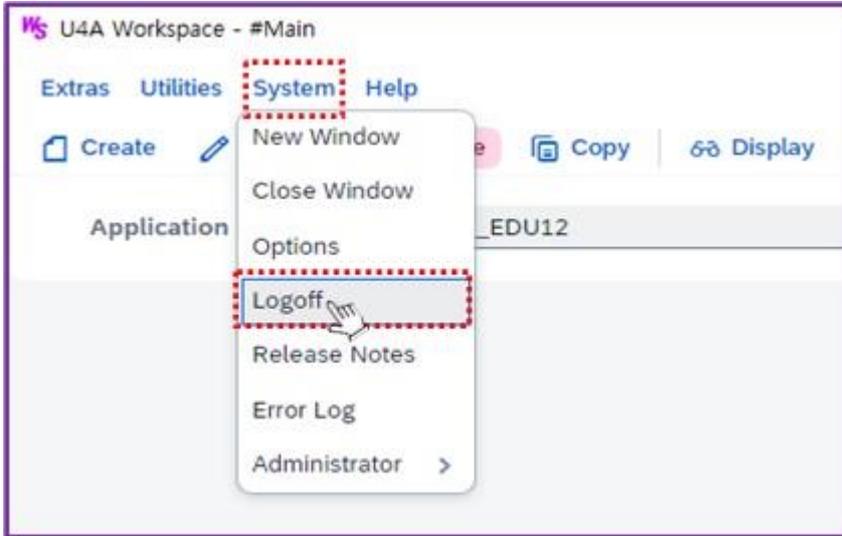
② CDN Setting(U4A WS 3.0 Application Upgrade Setting) = > ON / OFF 스위치 설정 후 Apply 버튼 클릭



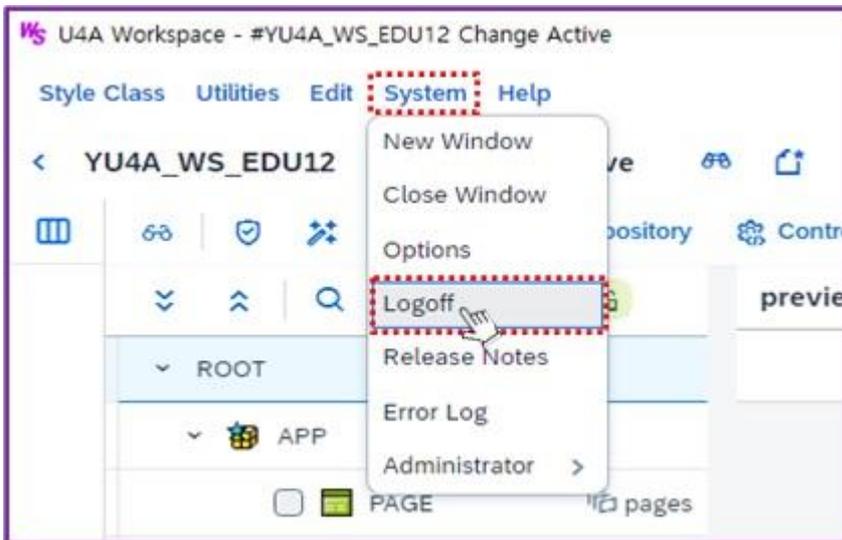
 설정 변경 후 U4A WorkSpace 재 시작을 권장함.

4.5.4 Logoff

① U4A WorkSpace 메인 화면에서 선택



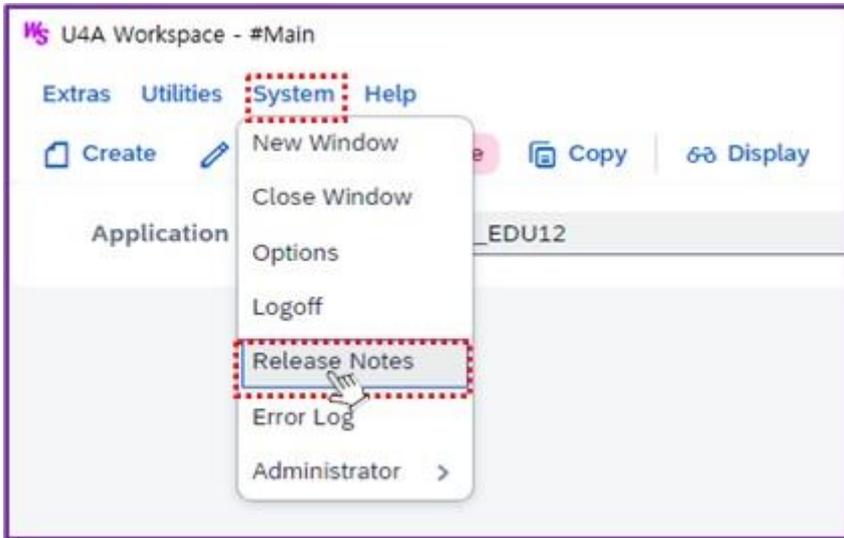
② Application 안에서 선택



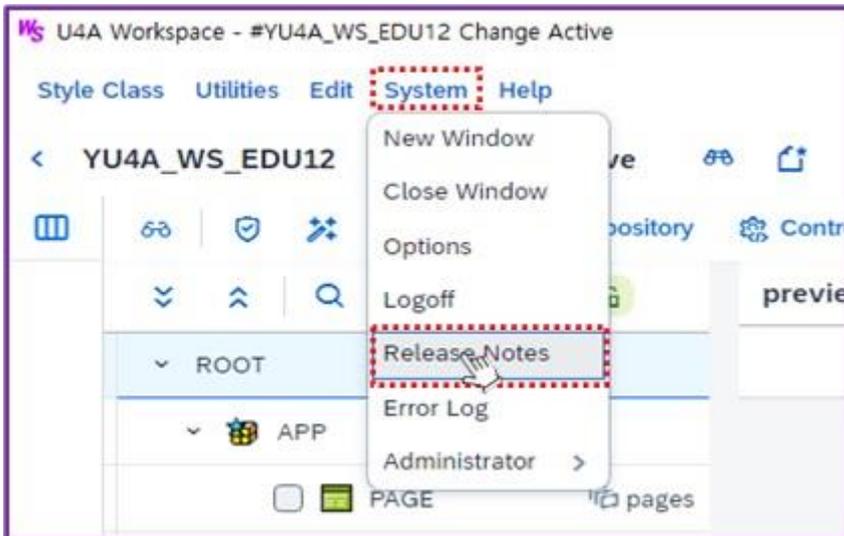
2. Logoff Pop-Up에서 Yes 버튼 클릭하면 모든 U4A WorkSpace 종료.(SAP GUI Logoff 와 동일)

4.5.5 Release Notes

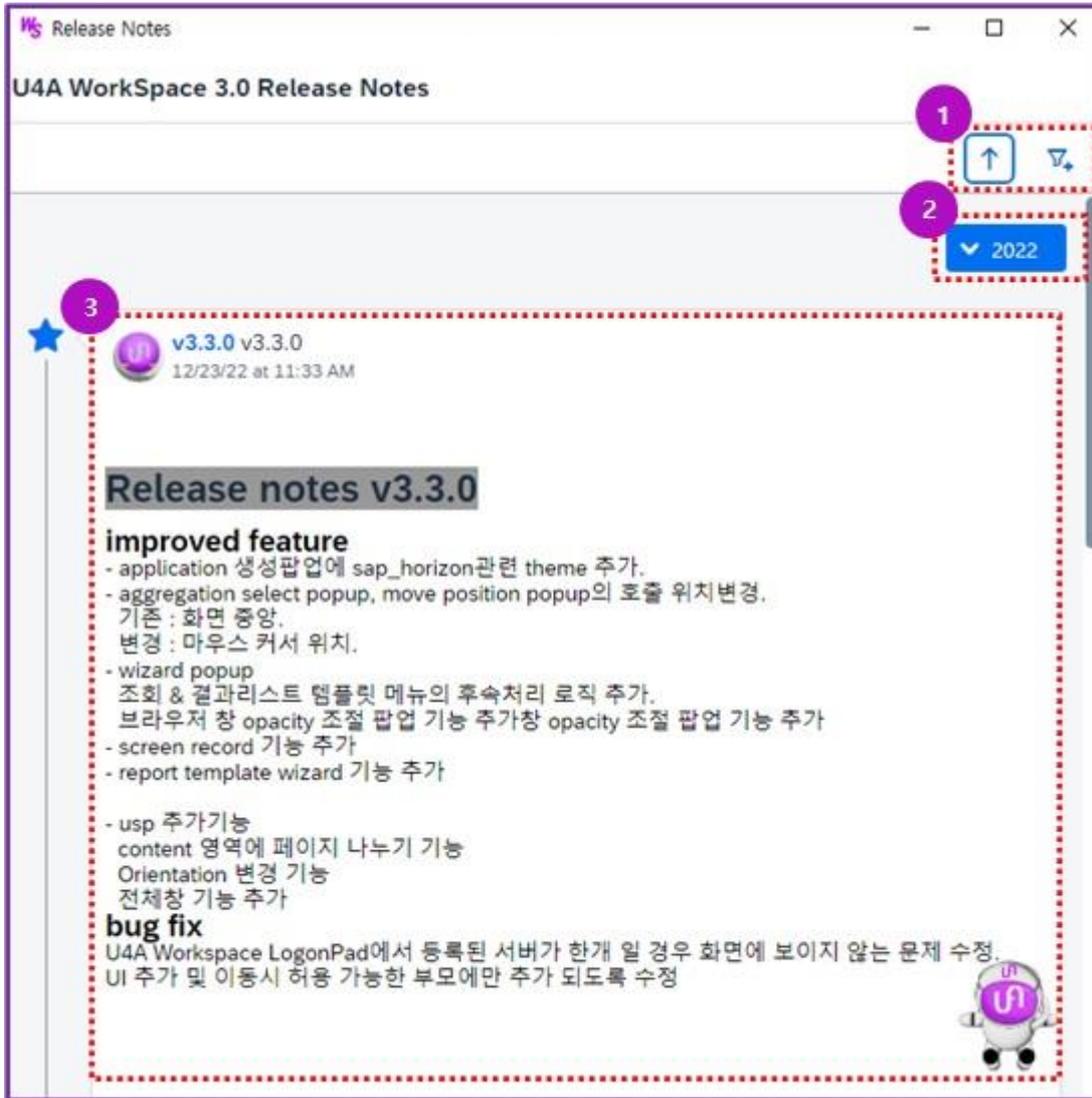
① U4A WorkSpace 메인 화면에서 선택



② Application 안에서 선택



2. Release Notes 개요.



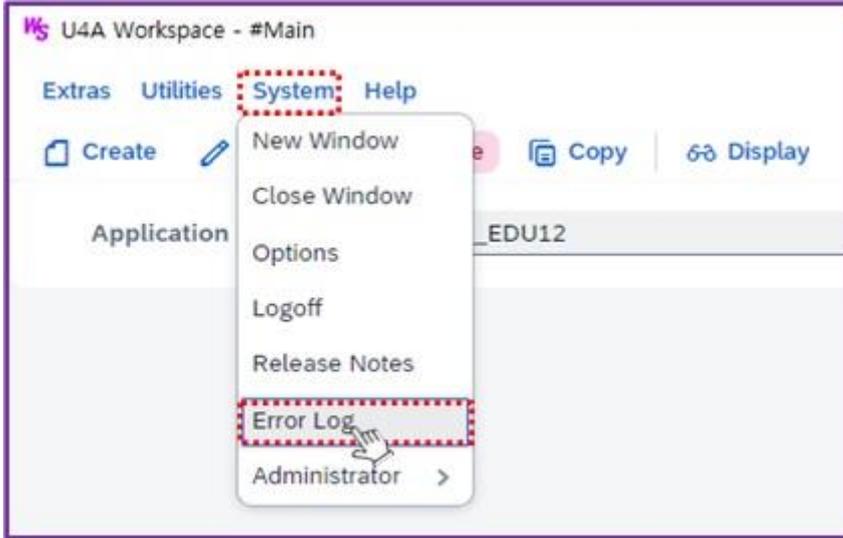
(1) Sort(Patch Version), Filter(Patch Version or 날짜) 기능

(2) 연도별 U4A WorkSpace Release Notes 확인 기능

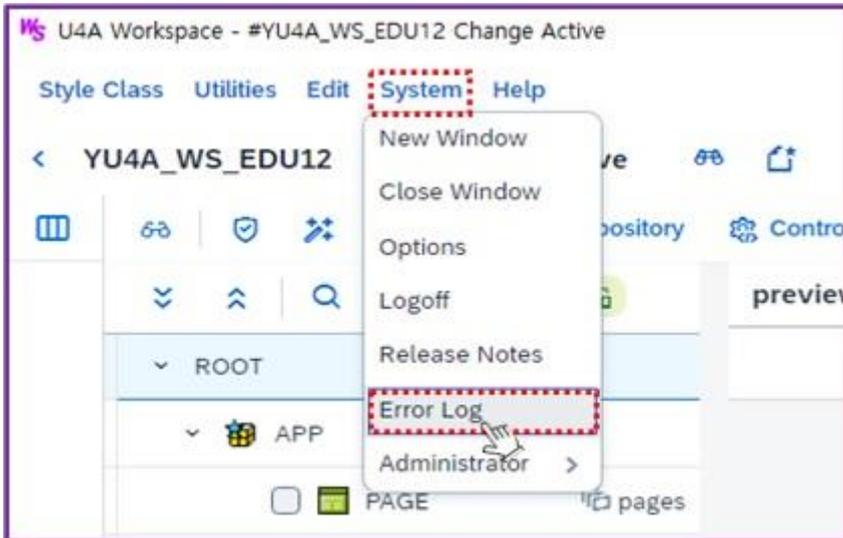
(3) U4A WorkSpace Release Notes 내역

4.5.6 Error Log

① U4A WorkSpace 메인 화면에서 선택

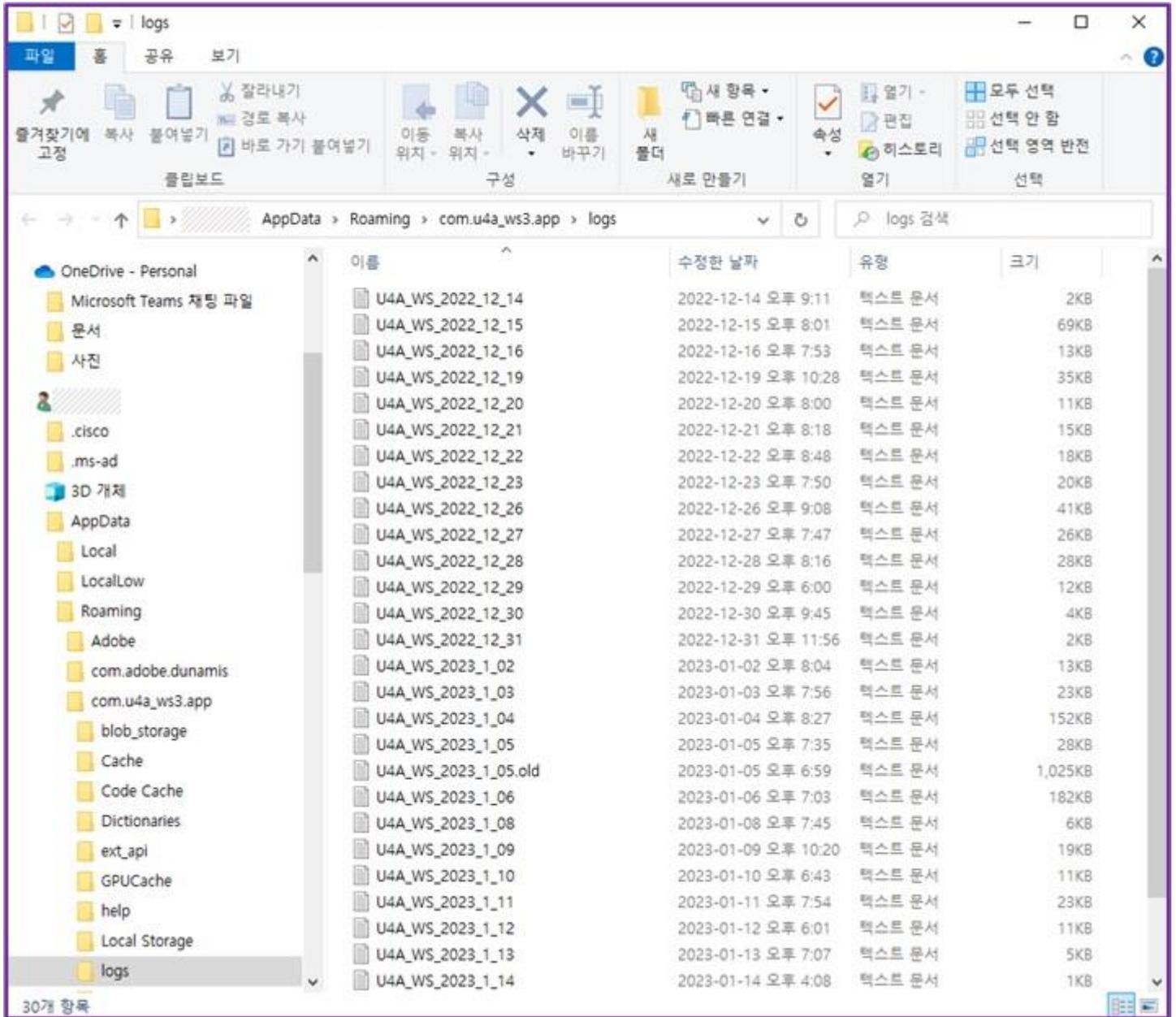


② Application 안에서 선택



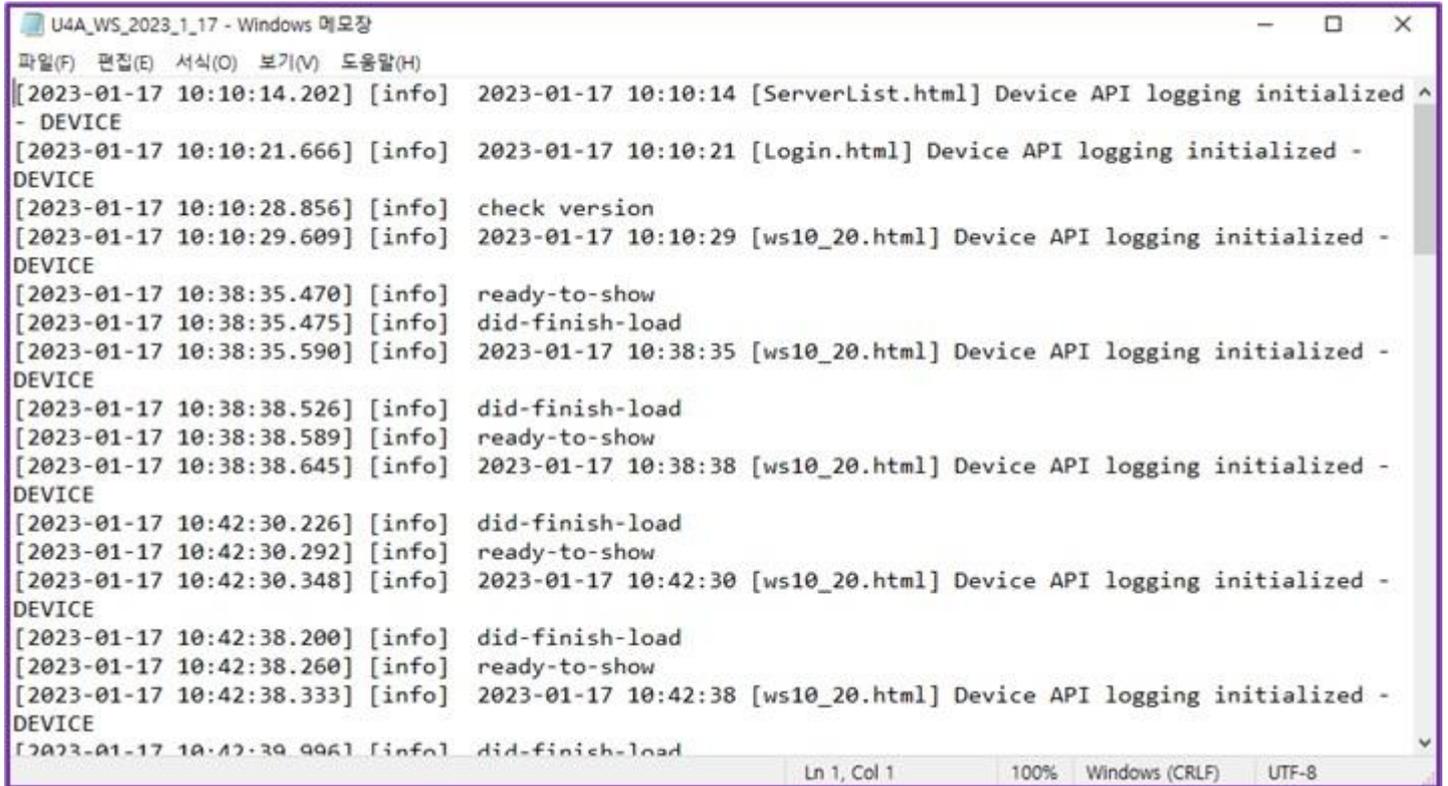
2. Error Log 선택시 날짜별 Log 모음 폴더 호출.

※ Log 파일을 U4A Solution Team 에게 전달하면 에러 발생시 신속한 대처 가능



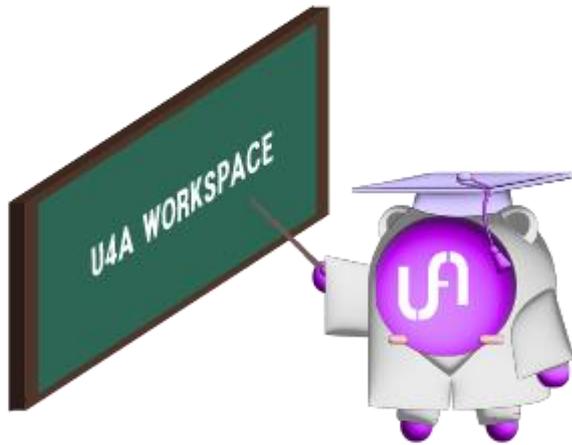
※ Log 개요.

U4A IDE HELP



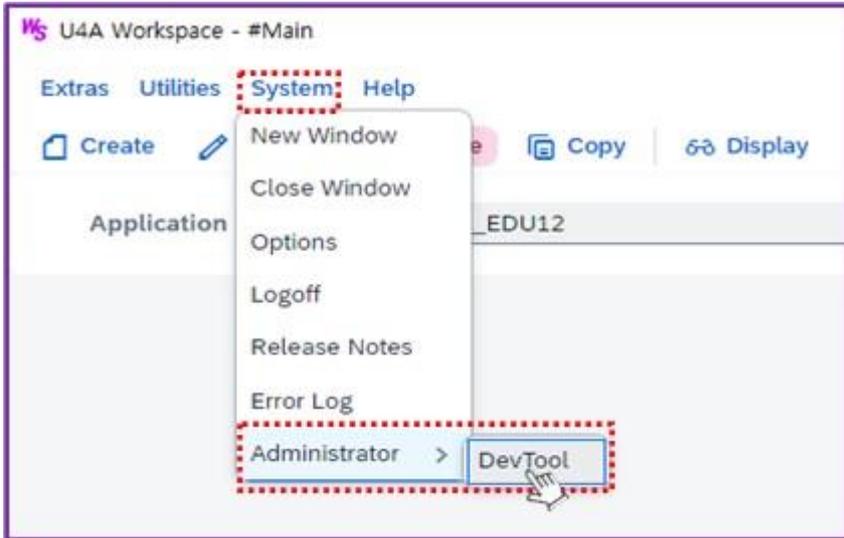
```
U4A_WS_2023_1_17 - Windows 메모장
파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)
[[2023-01-17 10:10:14.202] [info] 2023-01-17 10:10:14 [ServerList.html] Device API logging initialized -
- DEVICE
[2023-01-17 10:10:21.666] [info] 2023-01-17 10:10:21 [Login.html] Device API logging initialized -
DEVICE
[2023-01-17 10:10:28.856] [info] check version
[2023-01-17 10:10:29.609] [info] 2023-01-17 10:10:29 [ws10_20.html] Device API logging initialized -
DEVICE
[2023-01-17 10:38:35.470] [info] ready-to-show
[2023-01-17 10:38:35.475] [info] did-finish-load
[2023-01-17 10:38:35.590] [info] 2023-01-17 10:38:35 [ws10_20.html] Device API logging initialized -
DEVICE
[2023-01-17 10:38:38.526] [info] did-finish-load
[2023-01-17 10:38:38.589] [info] ready-to-show
[2023-01-17 10:38:38.645] [info] 2023-01-17 10:38:38 [ws10_20.html] Device API logging initialized -
DEVICE
[2023-01-17 10:42:30.226] [info] did-finish-load
[2023-01-17 10:42:30.292] [info] ready-to-show
[2023-01-17 10:42:30.348] [info] 2023-01-17 10:42:30 [ws10_20.html] Device API logging initialized -
DEVICE
[2023-01-17 10:42:38.200] [info] did-finish-load
[2023-01-17 10:42:38.260] [info] ready-to-show
[2023-01-17 10:42:38.333] [info] 2023-01-17 10:42:38 [ws10_20.html] Device API logging initialized -
DEVICE
[2023-01-17 10:42:39.996] [info] did-finish-load
Ln 1, Col 1 100% Windows (CRLF) UTF-8
```

4.5.7 Administrator

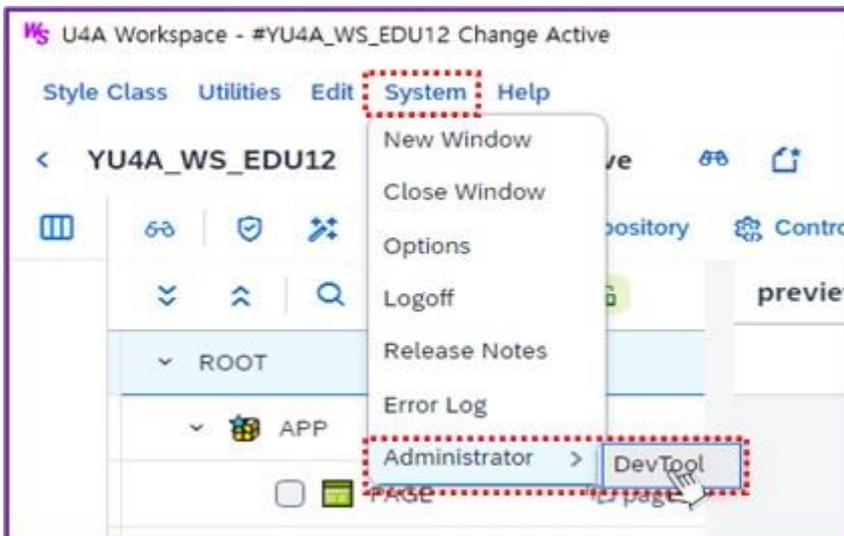


4.5.7.1 DevTool

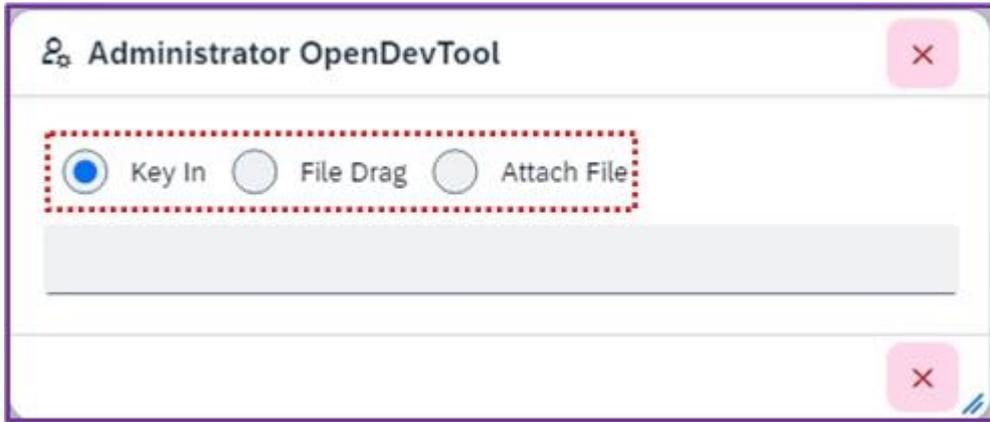
① U4A WorkSpace 메인 화면에서 선택



② Application 안에서 선택



2. DevTool 개요.



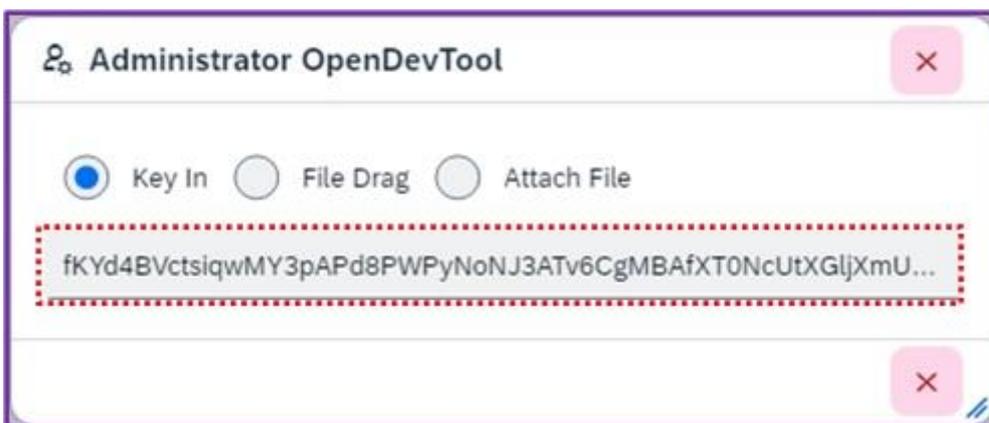
(1) Key In : U4A Solution Team에게 받은 Key를 입력.

(2) File Drag : U4A Solution Team에게 받은 Key file을 Drag & Drop을 통해 등록.

(3) Attach File : U4A Solution Team에게 받은 Key file을 업로드를 통해 등록.

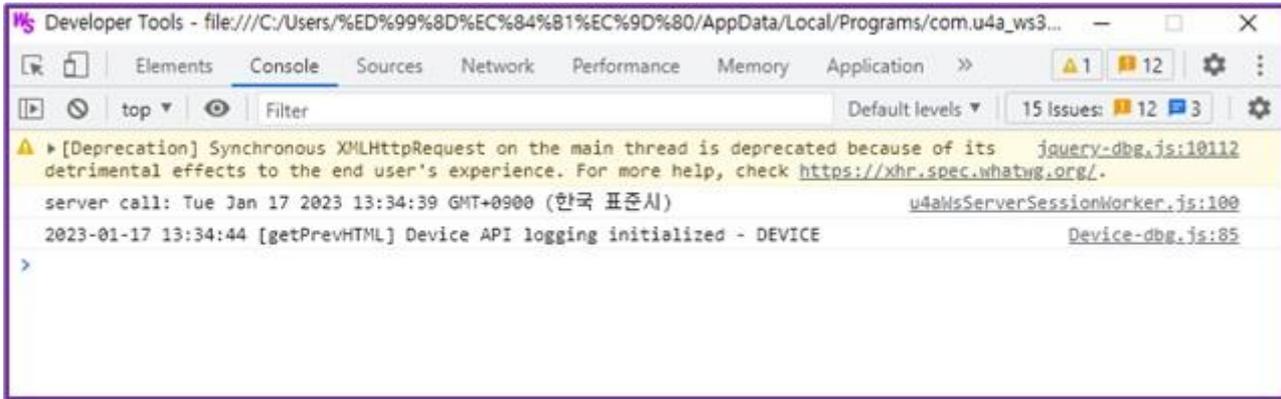
3. 본 단원에서는 Key In 방식을 통하여 DevTool를 호출.

① U4A Solution Team에게 받은 "Key" 입력 후 Enter.

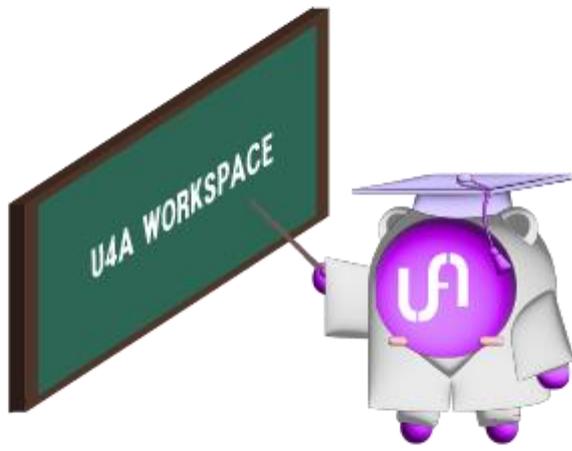


② Developer Tool 호출

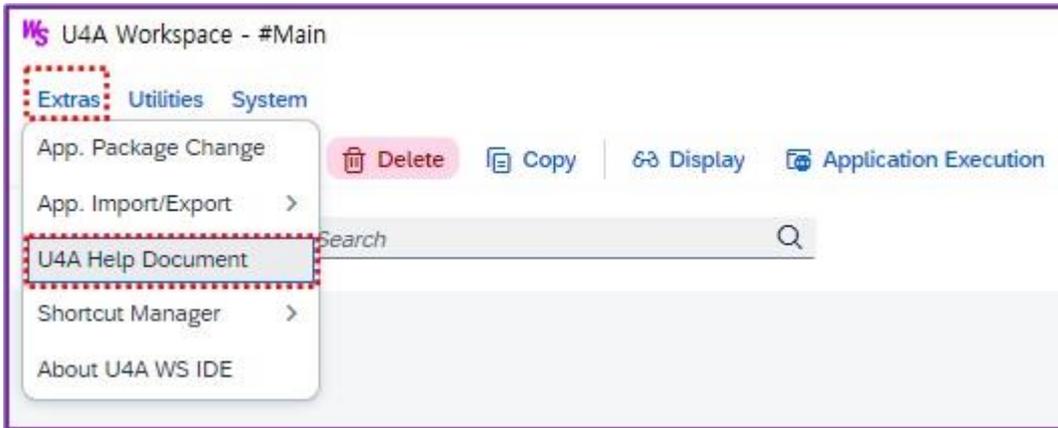
U4A IDE HELP



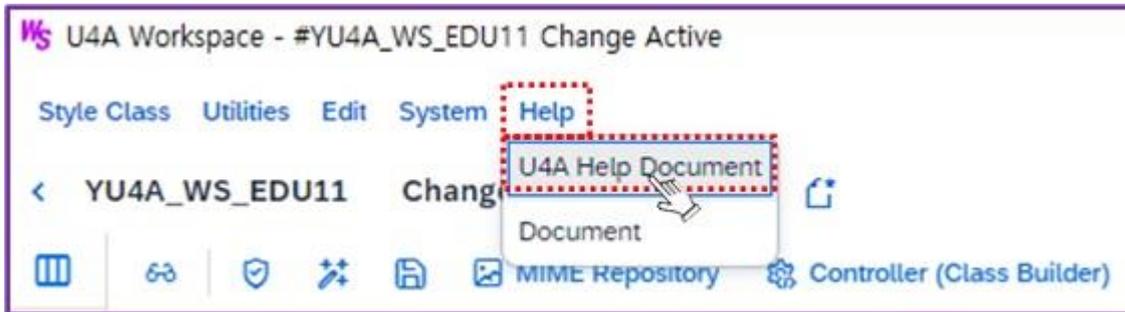
4.6 추가 기능 메뉴 - Help



4.6.1 Help Document



② Application 화면에서 Help -> U4A Help Document 선택.



2. U4A의 가이드에 대한 Help Document 실행.

The screenshot shows a web browser window titled "U4A IDE HELP". The interface includes a top toolbar with icons for home, search, back, forward, stop, refresh, and print. Below the toolbar is a navigation pane on the left with a tree view containing the following items:

- Instruction
- 개발 환경 구성
- U4A IDE Platform Beginning
- U4A IDE Workbench 사용 방법

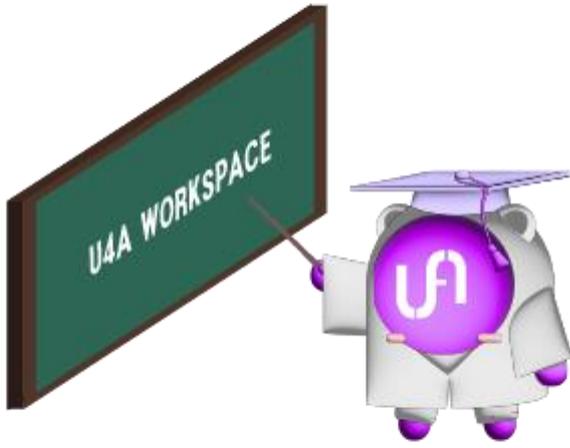
The main content area displays the "Cover Page" of the help documentation. The page features a background image of a laptop and a smartphone on a desk. A central text box contains the following text:

SAP UI5기반 시스템 구축 개발을 위한
U4A IDE Platform Help
 openUI5 Application Development Solution for SAP Developers

Below this, the text "Part - 1" is prominently displayed. At the bottom of the page, there is contact information for INFOCG Inc. and a logo for SAP Certified.

Copyright 2017. INFOCG Inc. all rights reserved.
 "ABAP®", "SAP NetWeaver®", "SAP S/4HANA®" are the trademark(s) or registered trademark(s) of SAP SE in Germany and in several other countries.

4.7 도구 모음



4.7.1 Hide

▣ 목적 : U4A WorkSpace 화면 투명도를 조절하여 WorkSpace가 실행된 상태에서 다른 작업 수행하기 위한 용도.

1. "Hide" 실행 방법에는 2가지가 존재한다.

① U4A WorkSpace 메인 화면에서 붉은 영역 아이콘 선택



② Application 안 붉은 영역 아이콘 선택



2. 본 단원에서는 U4A WorkSpace 메인 화면에서 "Hide" 기능 수행

① 화면 투명도 조절 Pop-Up

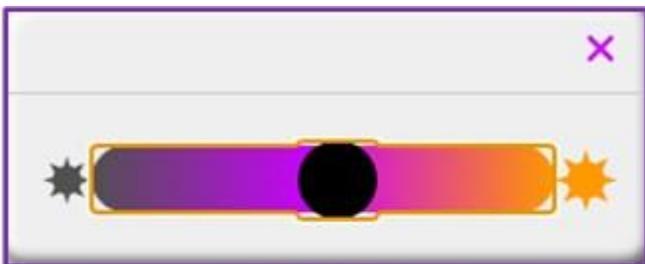


Figure : < 화면 투명도 조절 Pop-Up >



② 화면 투명도를 조절하여 U4A WorkSpace 실행 중 다른 작업 수행

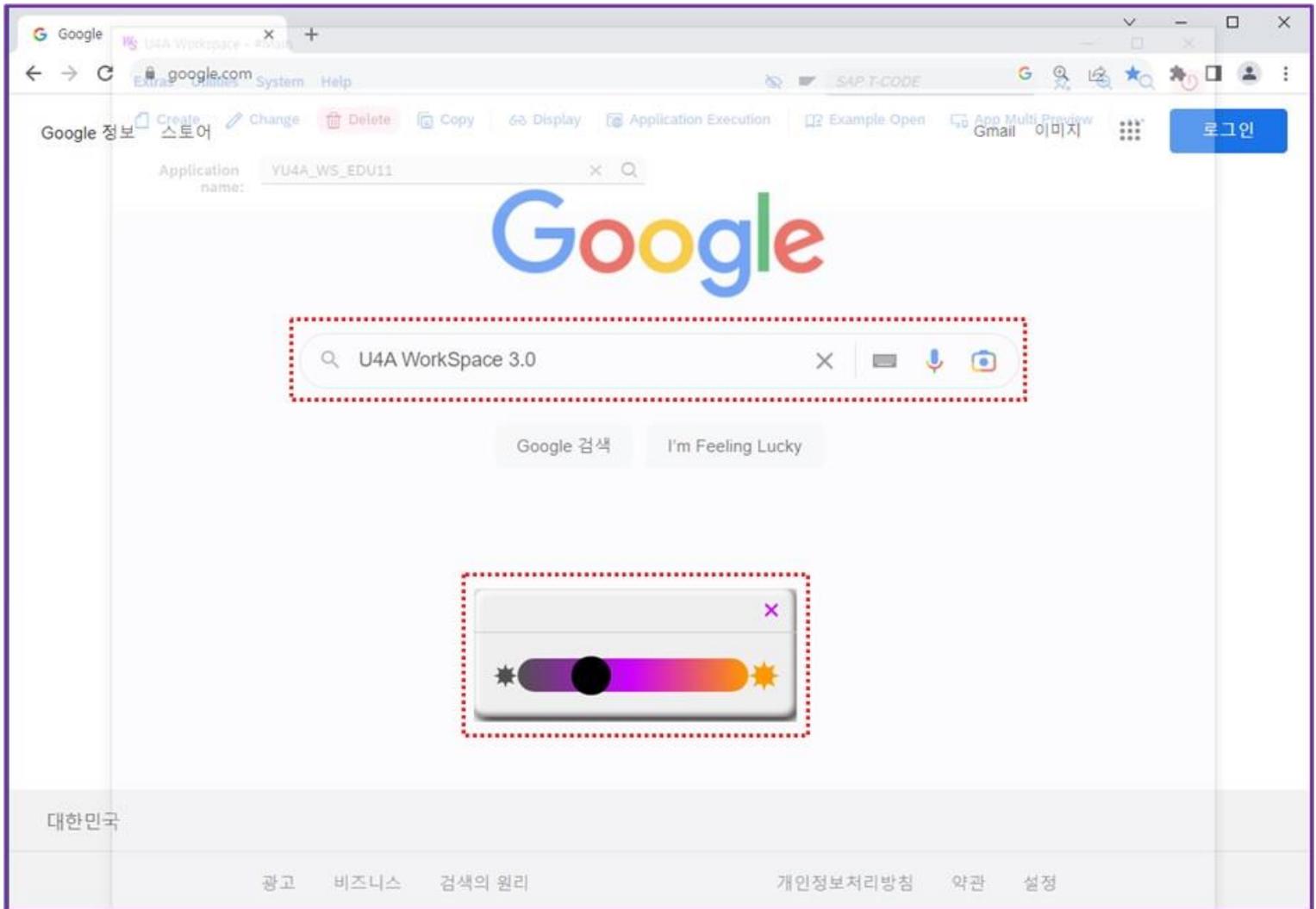


Figure : < 화면 투명도 조절 하여 WorkSpace 창 뒤에있는 브라우저 이용한 모습 >

4.7.2 T-Code

- ① U4A WorkSpace 메인 화면에서 붉은 영역에 T-Code 입력



- ② Application 안 붉은 영역에 T-Code 입력



2. 본 단원에서는 U4A WorkSpace 메인 화면에서 "T-Code" 입력 후 Enter = > 프로그램 실행.

U4A IDE HELP

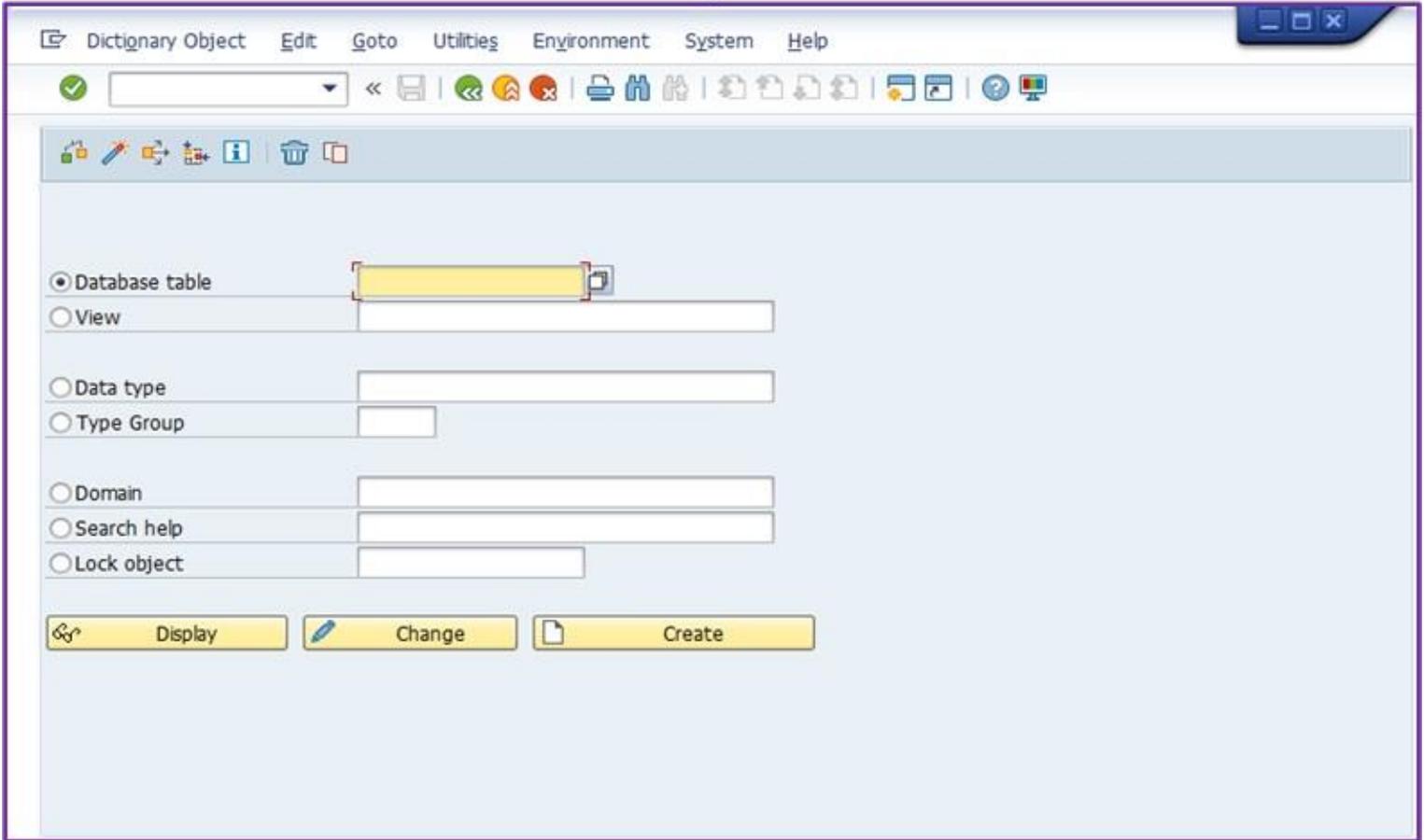
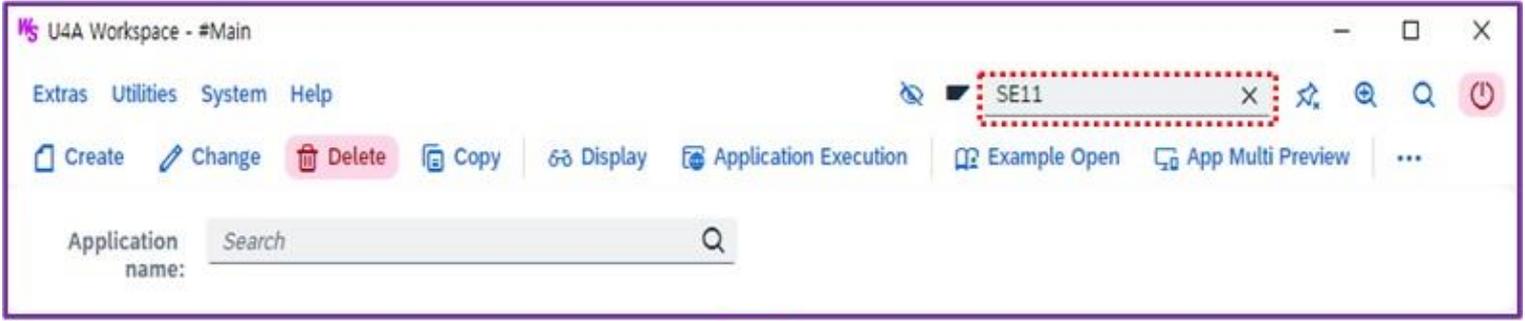


Figure : < ABAP Dictionary 호출 >

4.7.3 Browser Pin

① U4A WorkSpace 메인 화면에서 붉은 영역 아이콘 선택



② Application 안 붉은 영역 아이콘 선택



2. 본 단원에서는 U4A WorkSpace 메인 화면에서 "Browser Pin" 기능 수행

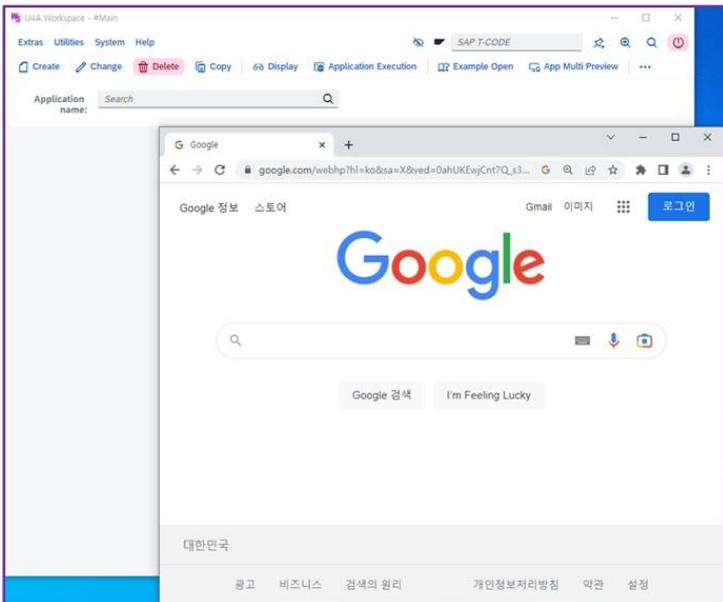


Figure : < Browser Pin 기능 미적용 상태 >

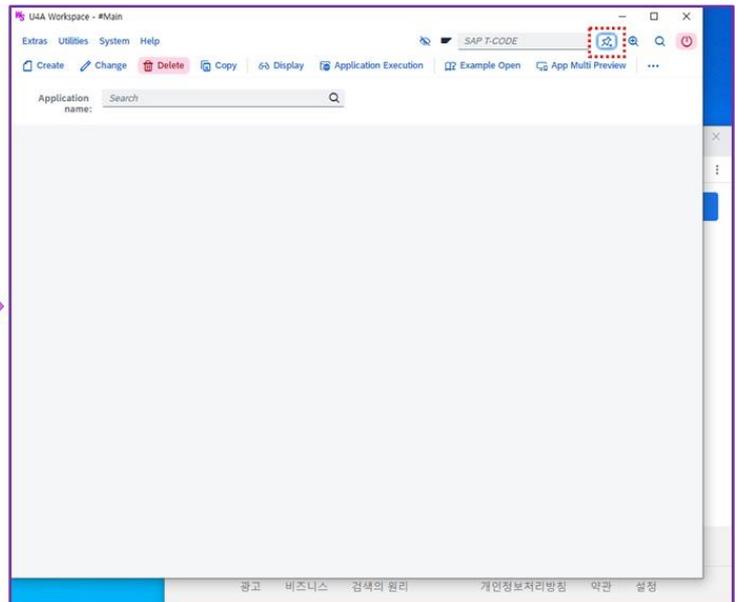


Figure : < Browser Pin 기능 적용 상태 >

4.7.4 Zoom

■ 목적 : U4A WorkSpace 화면 확대 및 축소

1. "Zoom" 실행 방법에는 2가지가 존재한다.

① U4A WorkSpace 메인 화면에서 붉은 영역 아이콘 선택



② Application 안 붉은 영역 아이콘 선택



2. 본 단원에서는 U4A WorkSpace 메인 화면에서 "Zoom" 기능 수행

① Zoom(확대 및 축소) Pop-Up



Figure : < Zoom(확대 및 축소) Pop-Up >



② "확대" 한 모습

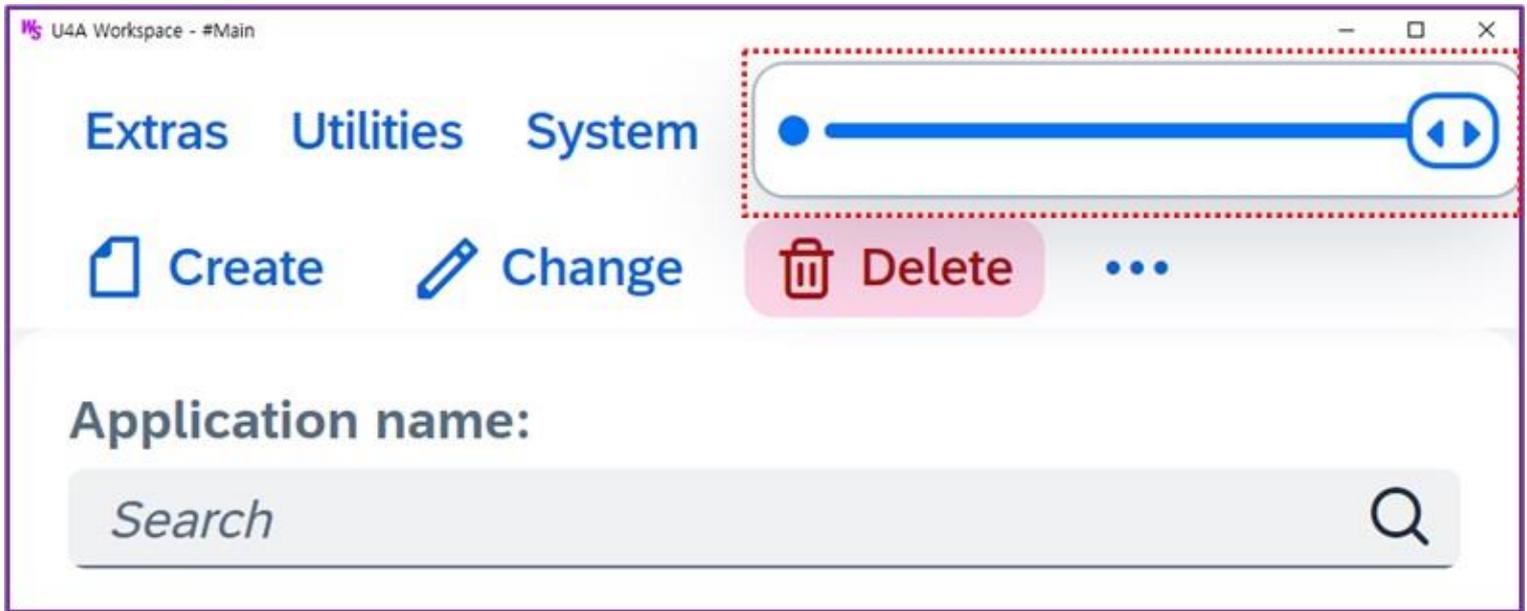


Figure : < 화면 최대로 확대 >

② "축소" 한 모습

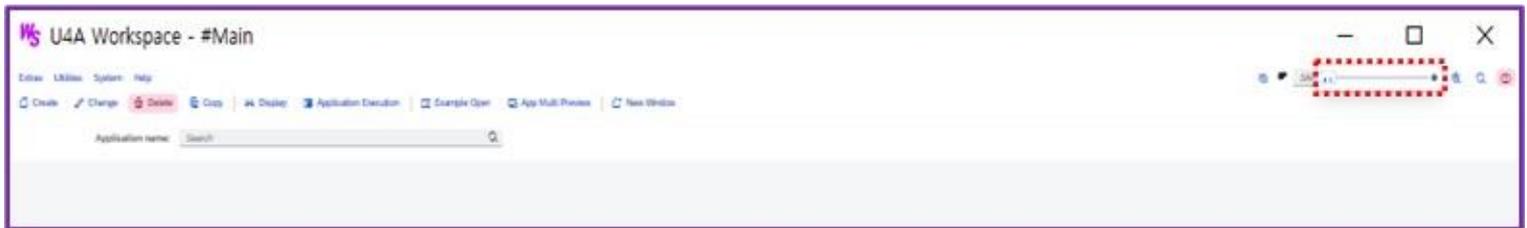


Figure : < 화면 최대로 축소 >

4.7.5 Window Text Search

■ 목적 : 키워드를 입력하여 U4A WorkSpace 화면에 존재하는 Text를 빠르게 찾기 위함

1. "Window Text Search" 실행 방법에는 2가지가 존재한다. (단축키 = > CTRL + SHIFT + R)

① U4A WorkSpace 메인 화면에서 붉은 영역 아이콘 선택



② Application 안 붉은 영역 아이콘 선택



2. 본 단원에서는 U4A WorkSpace 메인 화면에서 "Window Text Search" 기능 수행

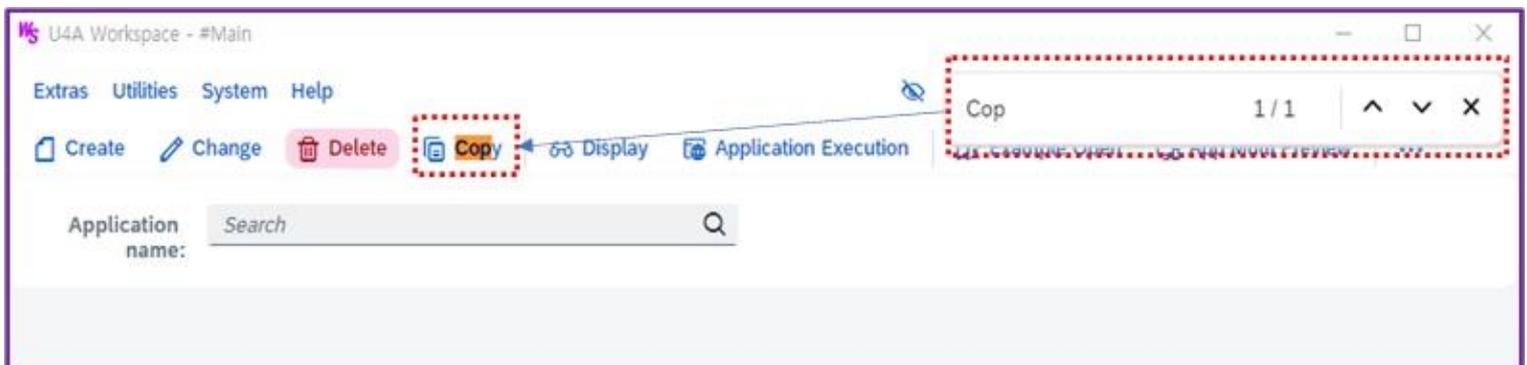
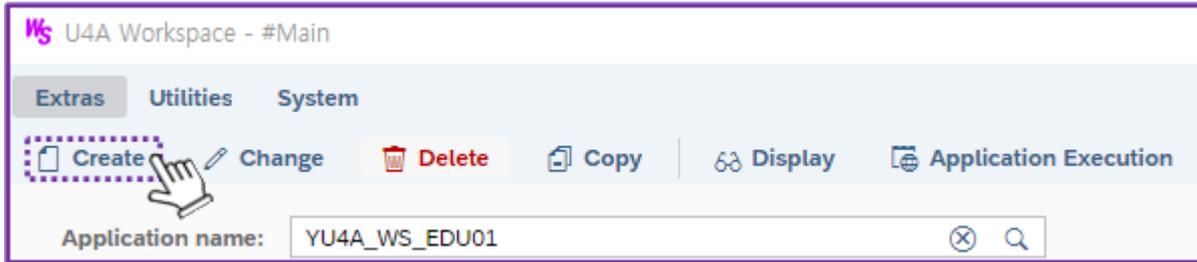


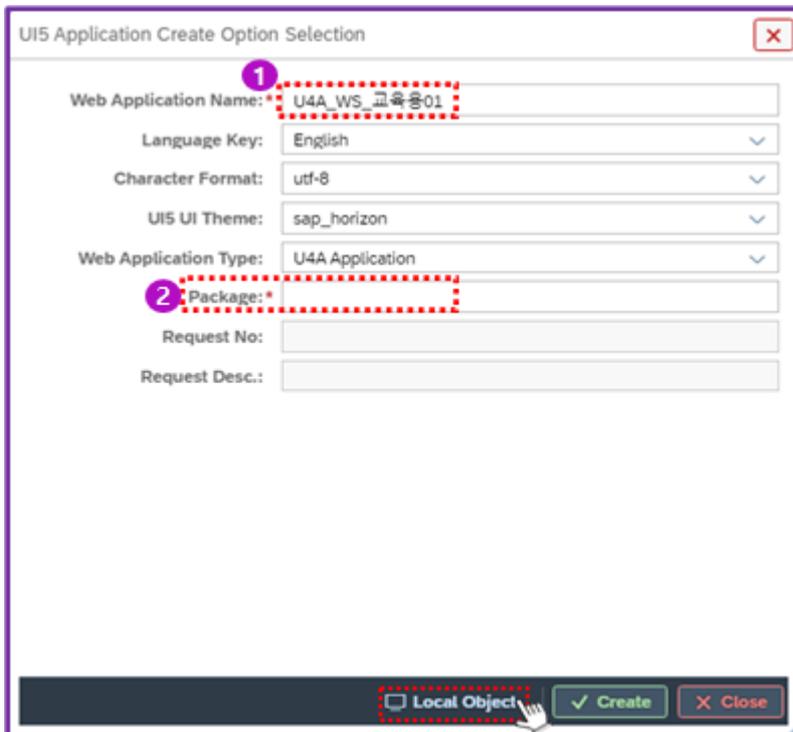
Figure : < Window Text Search에 Text 입력시 해당 하는 Text 백그라운드 색상 변경 >

4.8 Application 생성 하기

1. 신규 생성 할 Application ID(Name)을 입력 후 “Create” Button을 Click 함.



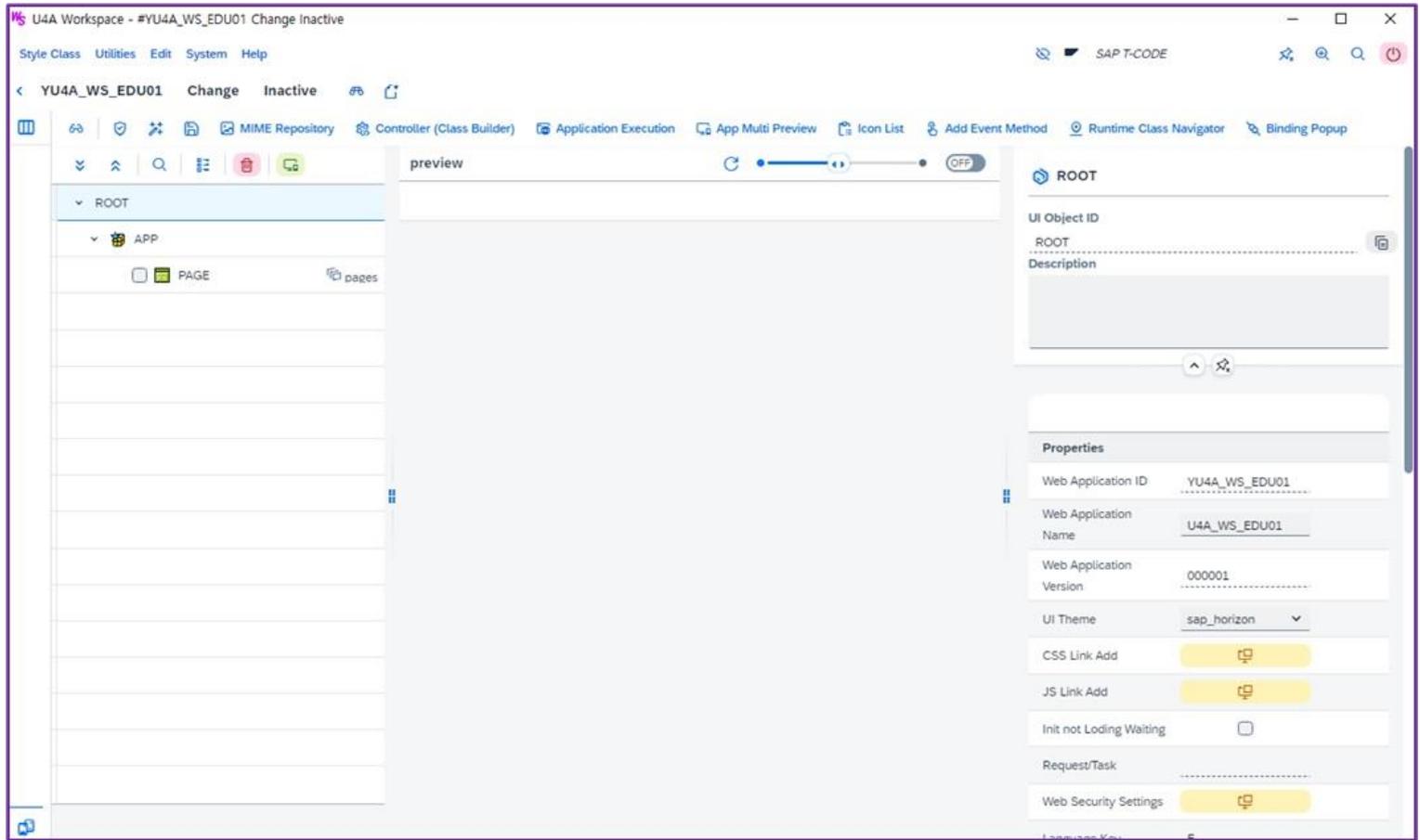
2. Application Create Option Selection 입력, 본 단원에선 Package 지정 없이 “Local Object” Button을 Click 하여 진행 함.



(1) Web Application Name = > 생성할 Application Descrtion 작성

(2) Application의 개발 Package 지정

3. 정상적으로 생성 된 Application Designer Toolkit 초기 화면



생성 시 사용될 Application ID(Name)은 반드시 "Y" 또는 "Z"로 시작해야 함.

생성 시 Application ID(Name)의 길이(Length)는 반드시 15 Character 이하 여야 함.

4.8.1 Application Create Option 설명

■ 목적 : Application을 생성할때 설정할 상세 항목 정의

1. Application 생성 시 보여 지는 Popup

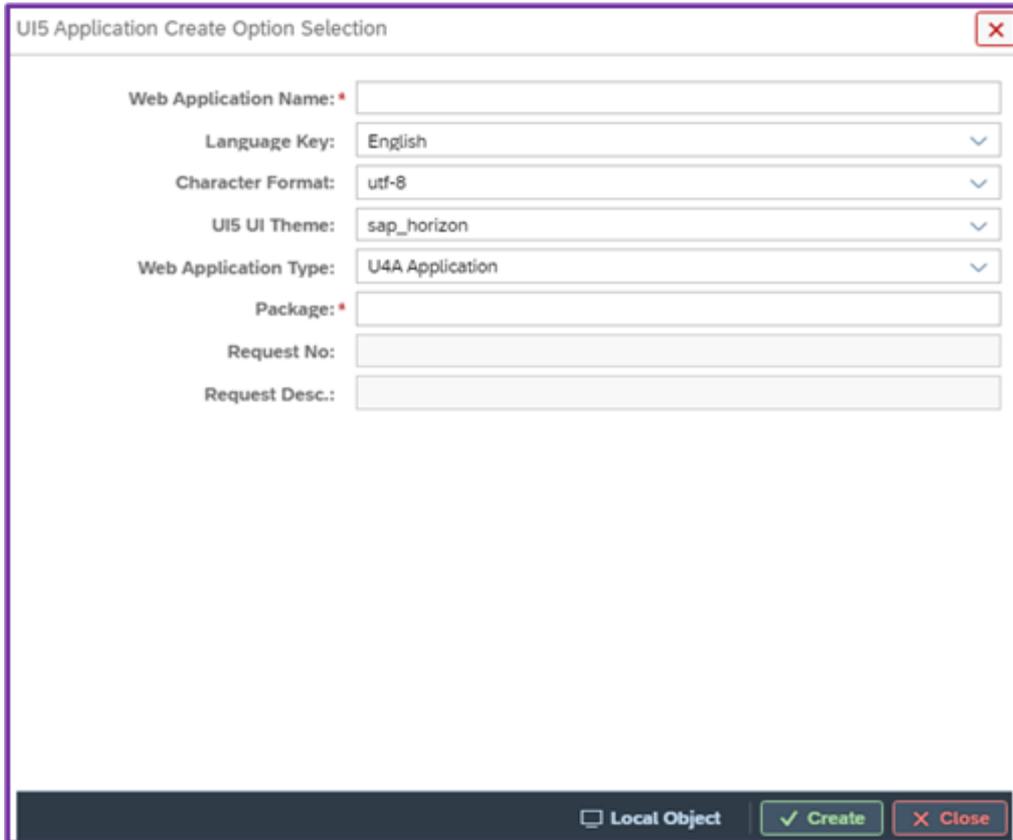


Figure : 신규생성할 Application의 속성 정의 팝업

Option Property	Means
Web Application Name	대상 Application의 내역 기술
Language Key	대상 Application Object 관리 Language Key (기본값 : SAP Logon Language Key) 용도는 Application OTR(Online Text Repository)의 선택 및 SAP 서버 시스템 변수 "SY-LANGU" 값에 영향을 줌
Character Format	UI5 Web Meta Tag Character Encoding (기본값 : utf-8 유니코드(Unicode))

	※ 상세 정보 참조 "Character encoding"
UI5 UI Theme	UI5 표준에서 제공 되는 UI Theme 값 지정 (기본값 : sap_horizon)
Web Application Type	U4A Application으로 기본값 설정
Package	개발 Package 설정
Request No:	CTS No
Request Desc:	CTS Description

2. UI5 UI Theme 별 Preview



Figure : UI5 UI Theme 별 적용 Metrix
(Sample UI : sap.m.button)



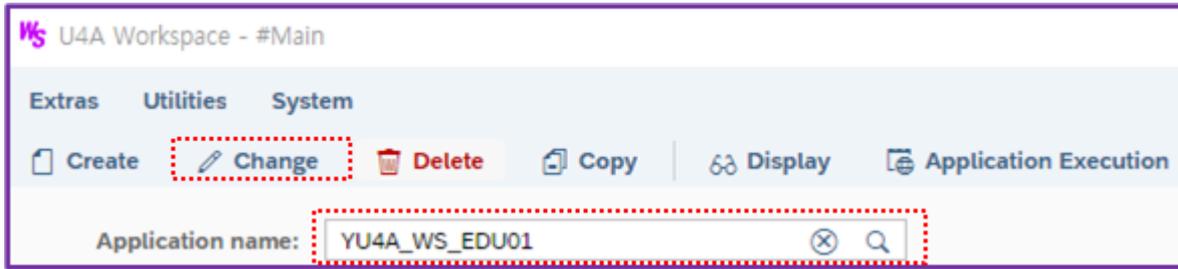
표준에서 제공되는 UI5 UI Theme를 원하는 Design Concept에 맞게 Customizing 가능 하며,

해당 작업은 **Web Designer**를 통해 진행 되어야 함. **UI5 Design guide**에 대한 정보를 얻기 원한다면 ["Fiori Design Guidelines"](#) 를 참조 하기 바람.

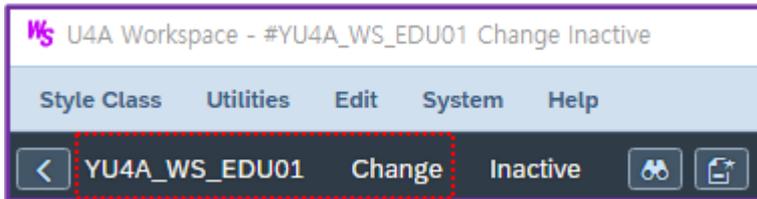
4.9 Application 수정 하기

■ 목적 : Application을 변경하기 위한 Button.

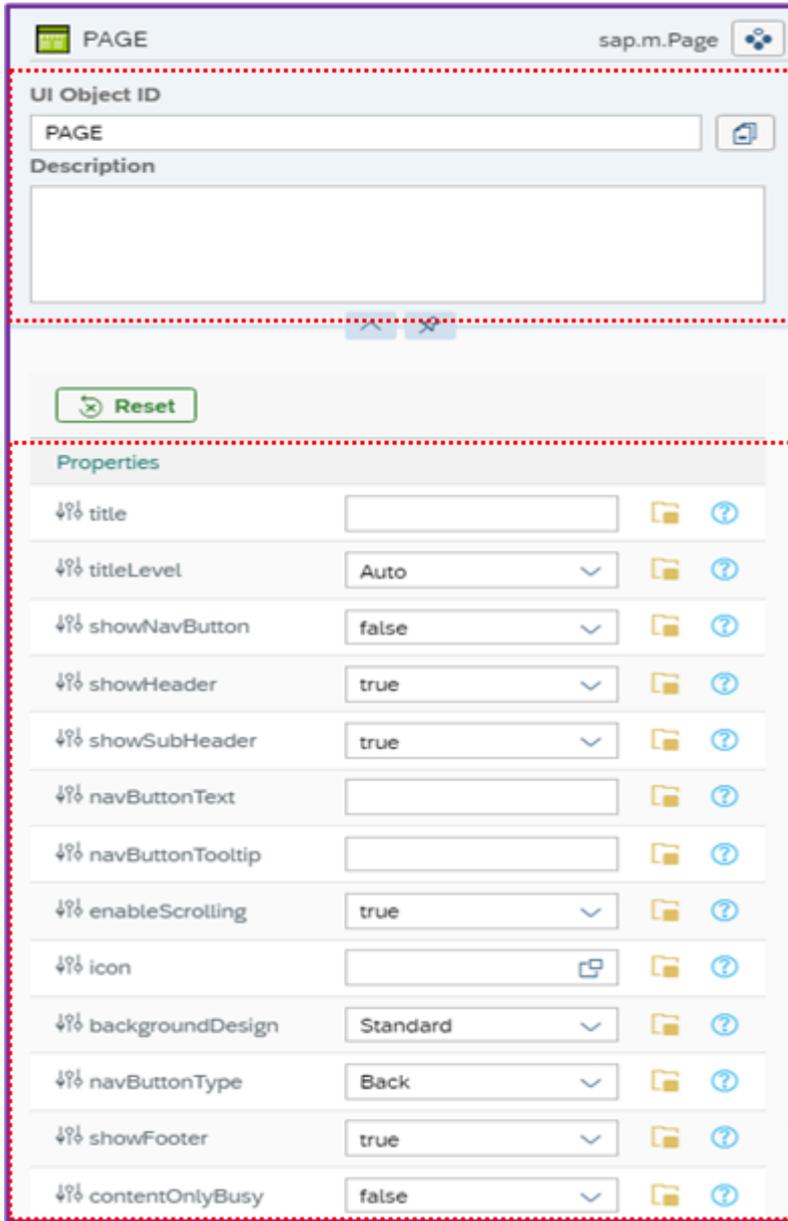
1. 수정 하려는 Application ID(Name)을 입력 후 "Change" Button Click 함.



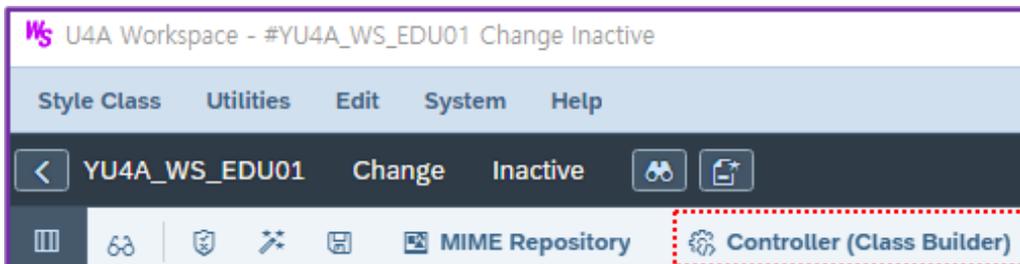
2. Application 좌측 상단에 Change 상태인 것을 확인할 수 있음



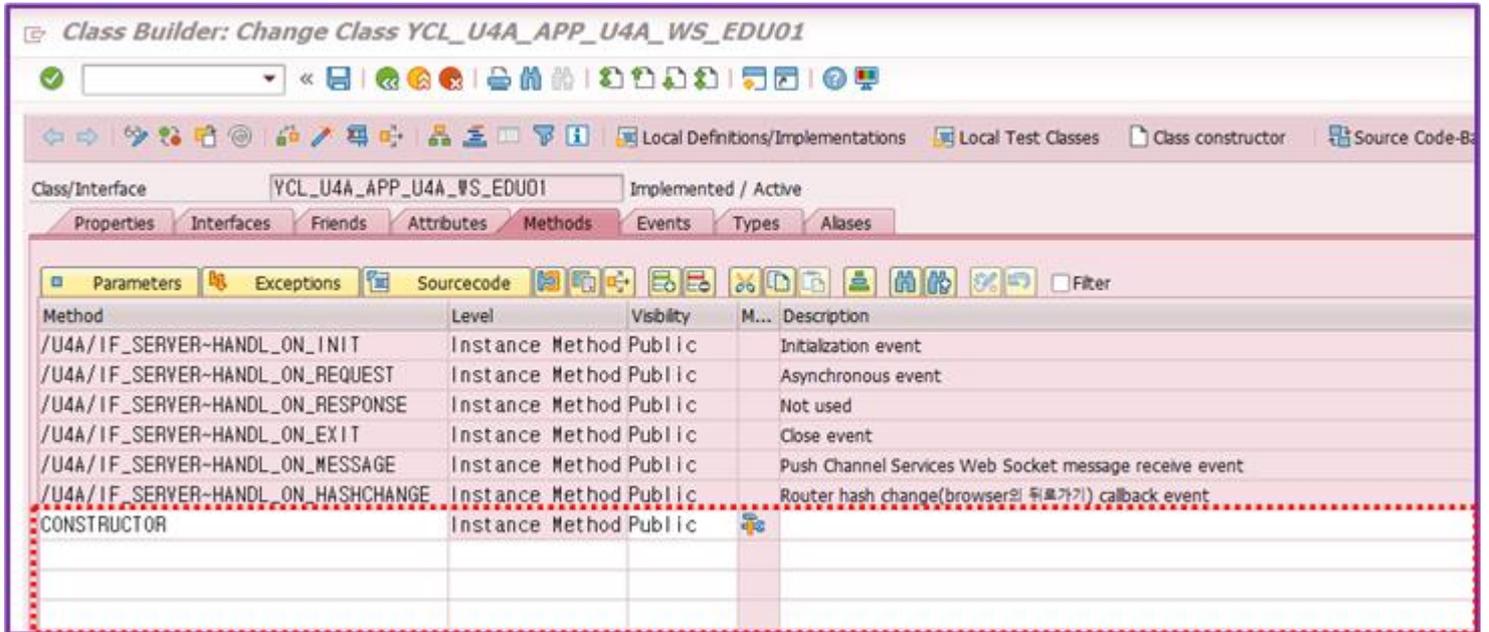
3. 초기 화면에서 Object id, Descriptions, Attributes 등을 수정할 수 있음.



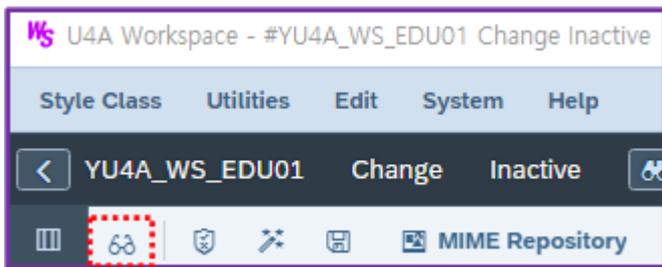
4. "Controller(Class builder)" Button Click.



5. Methods 등을 생성하고 수정할 수 있음.



6. 상단에서 해당 Icon을 클릭하거나 Ctrl + F1 단축키를 통해 조회 상태로 변경할 수 있음.

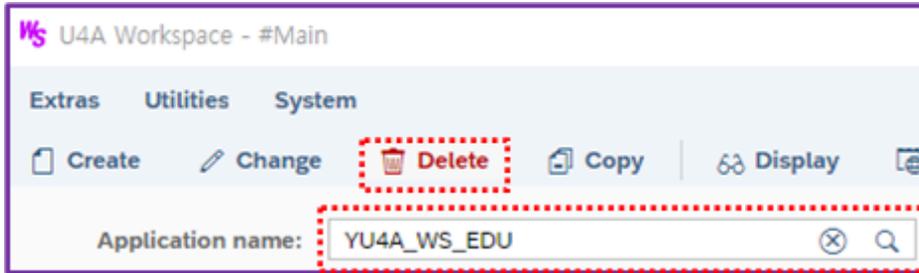


7. 만약 다른 사용자가 해당 Application을 Change 상태로 수정 중이라면 해당 Application을 Active 하거나 수정할 수 없음.

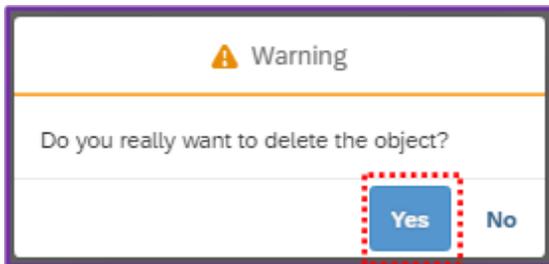


4.10 Application 삭제 하기

1. 삭제 할 Application ID(Name)을 입력 후 "Delete" Button Click 함.



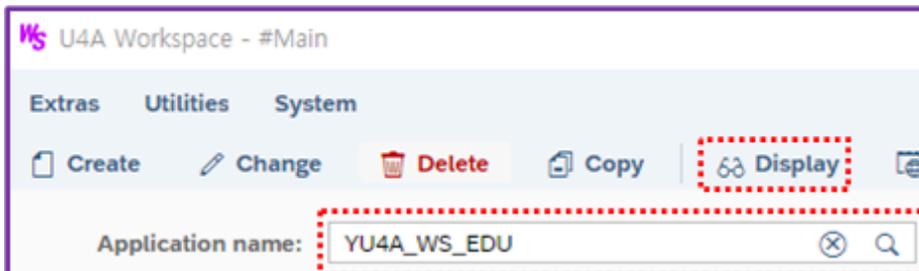
2. 삭제 확인 Popup이 표시되며 삭제할 시 "Yes" Button을 클릭함.



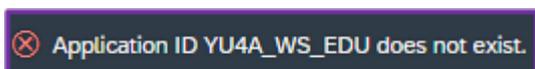
3. 왼쪽 하단 Message 확인 가능 - 해당 Application 존재하지 않음.



4. 삭제한 Application을 조회하여 삭제되었는지 확인.

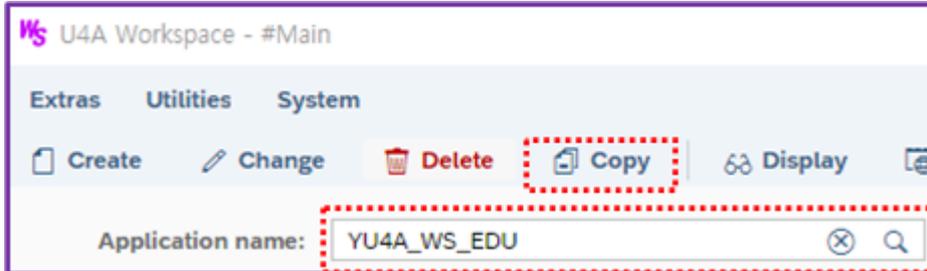


5. 왼쪽 아래에 해당 Application은 삭제 되어 존재하지 않음을 알려주는 Message 확인 할 수 있음.

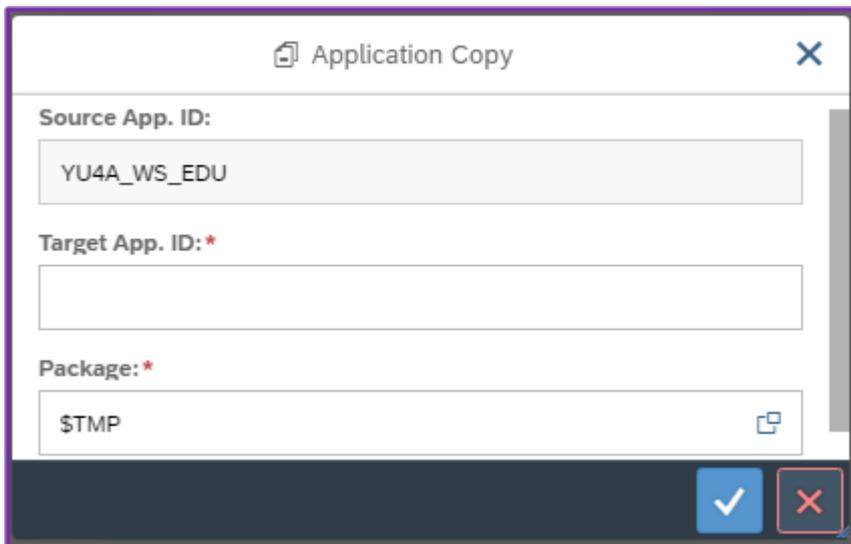


4.11 Application 복사 하기

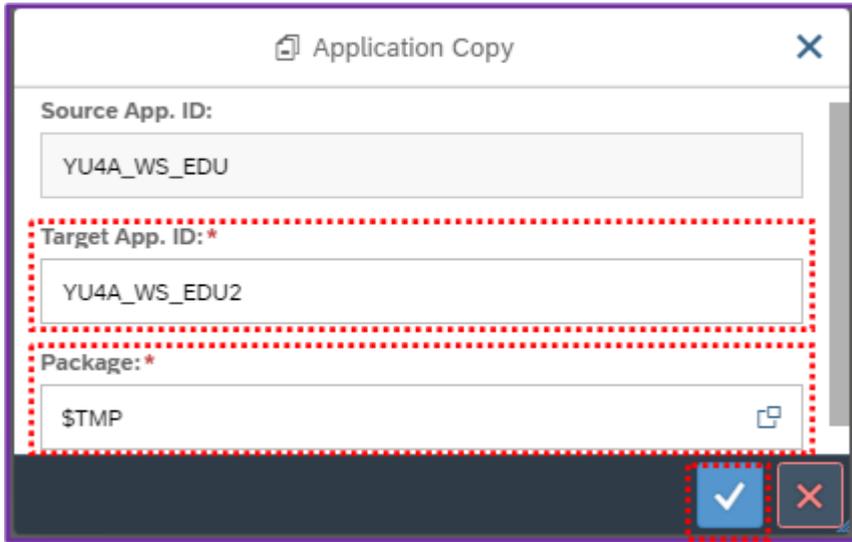
1. 복사 하려는 Application ID(Name)을 입력 후 “Copy“ Button Click 함.



2. 복사 하려는 Application 명을 확인할 수 있음.



3. 복사한 내용을 붙여 넣을 Application ID(Name), 개발 Package를 입력 후 실행 Button Click



- ① 본 단원에선 Package 지정 없이 "Local Object"로 진행함
- ② 복사하려는 Application이 "Inactive" 상태일 경우의 Error Pop-Up

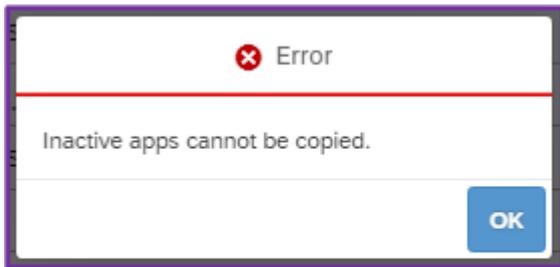
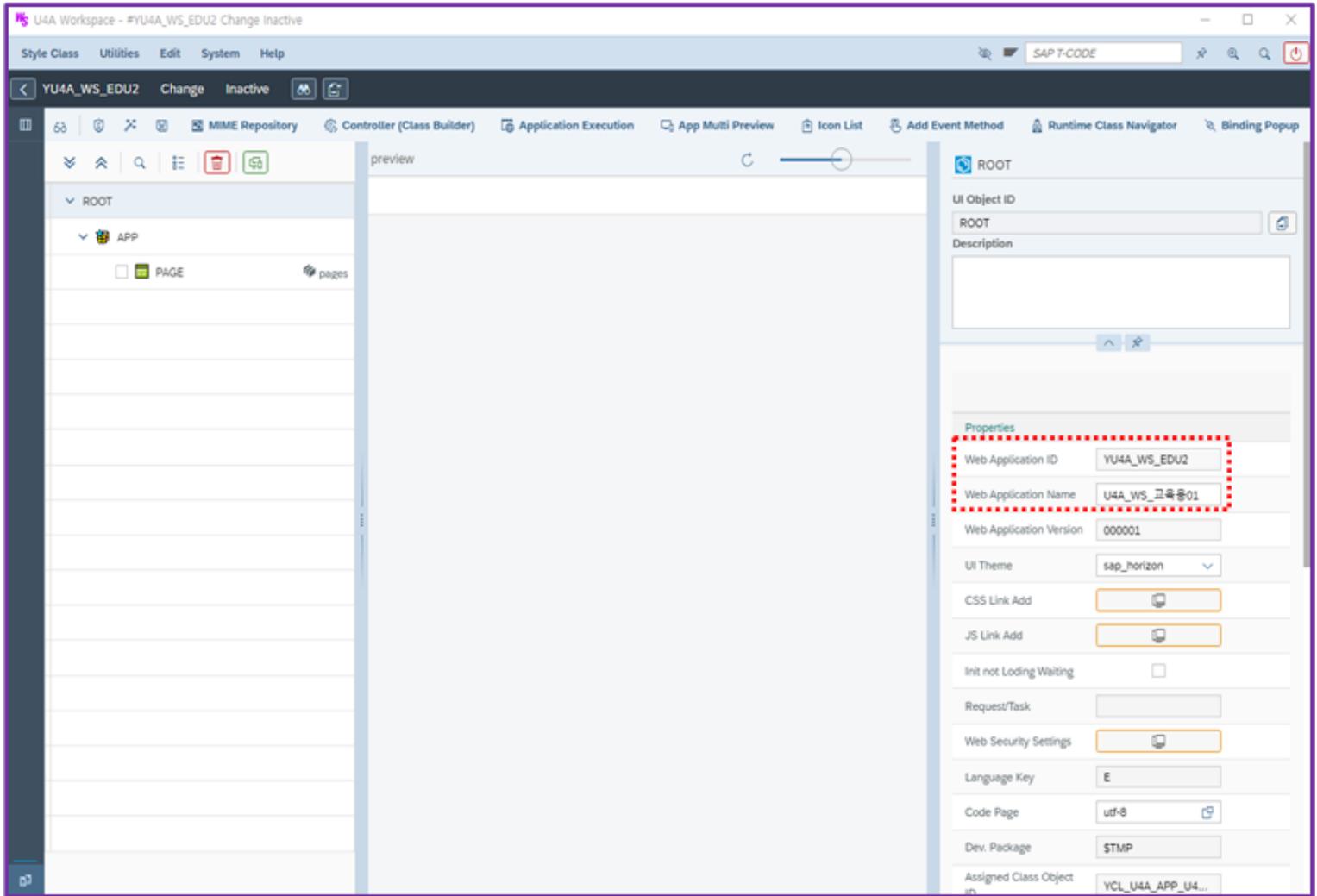


Figure : Active 되지 않은 Application 복사 불가

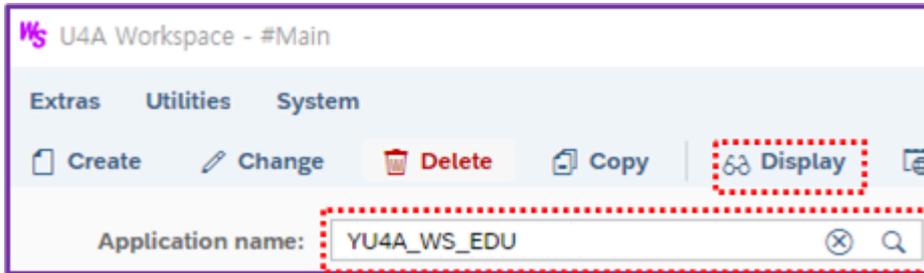
4. 정상적으로 복사 된 Application Designer Toolkit 초기 화면.

U4A IDE HELP

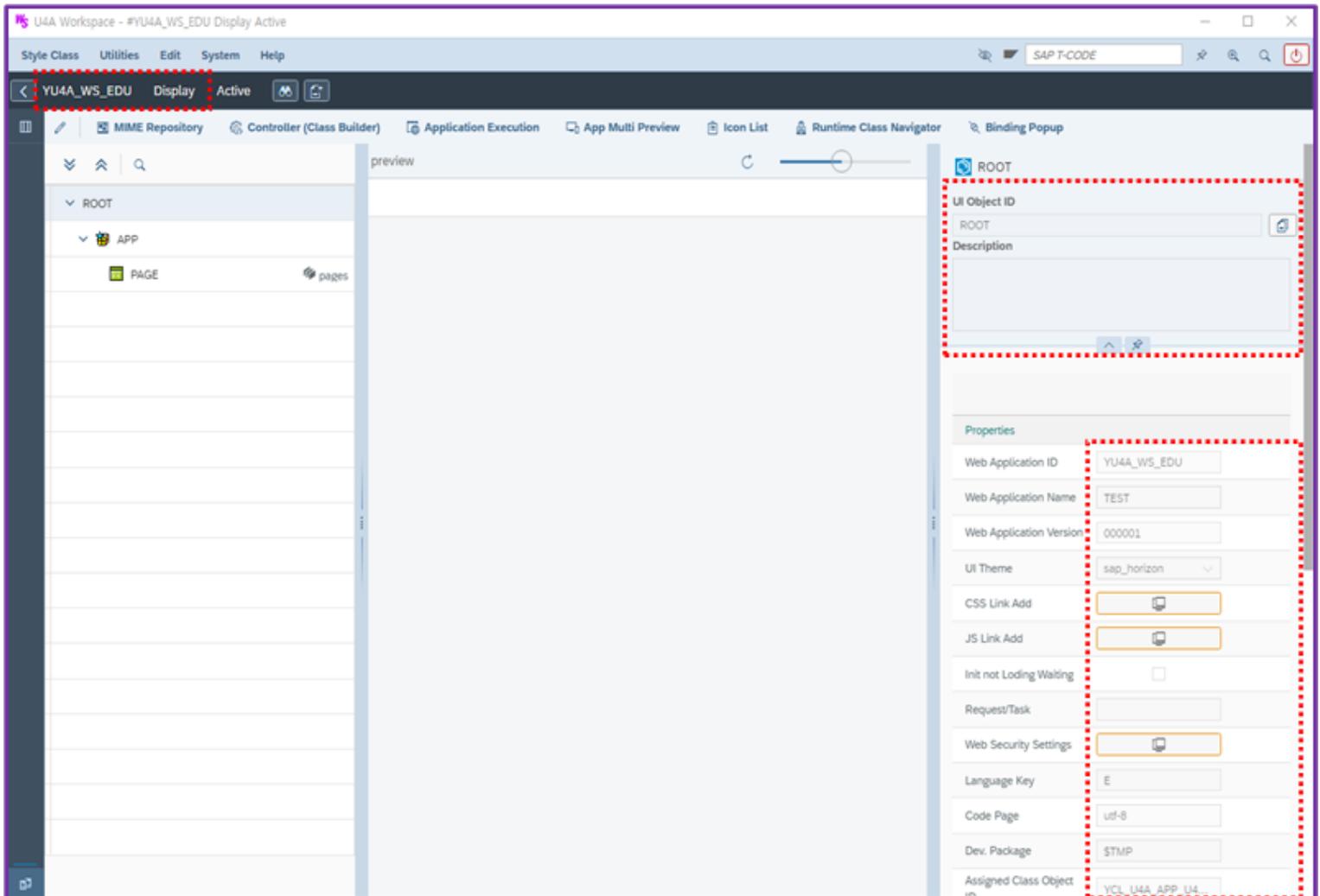


4.12 Application 조회 하기

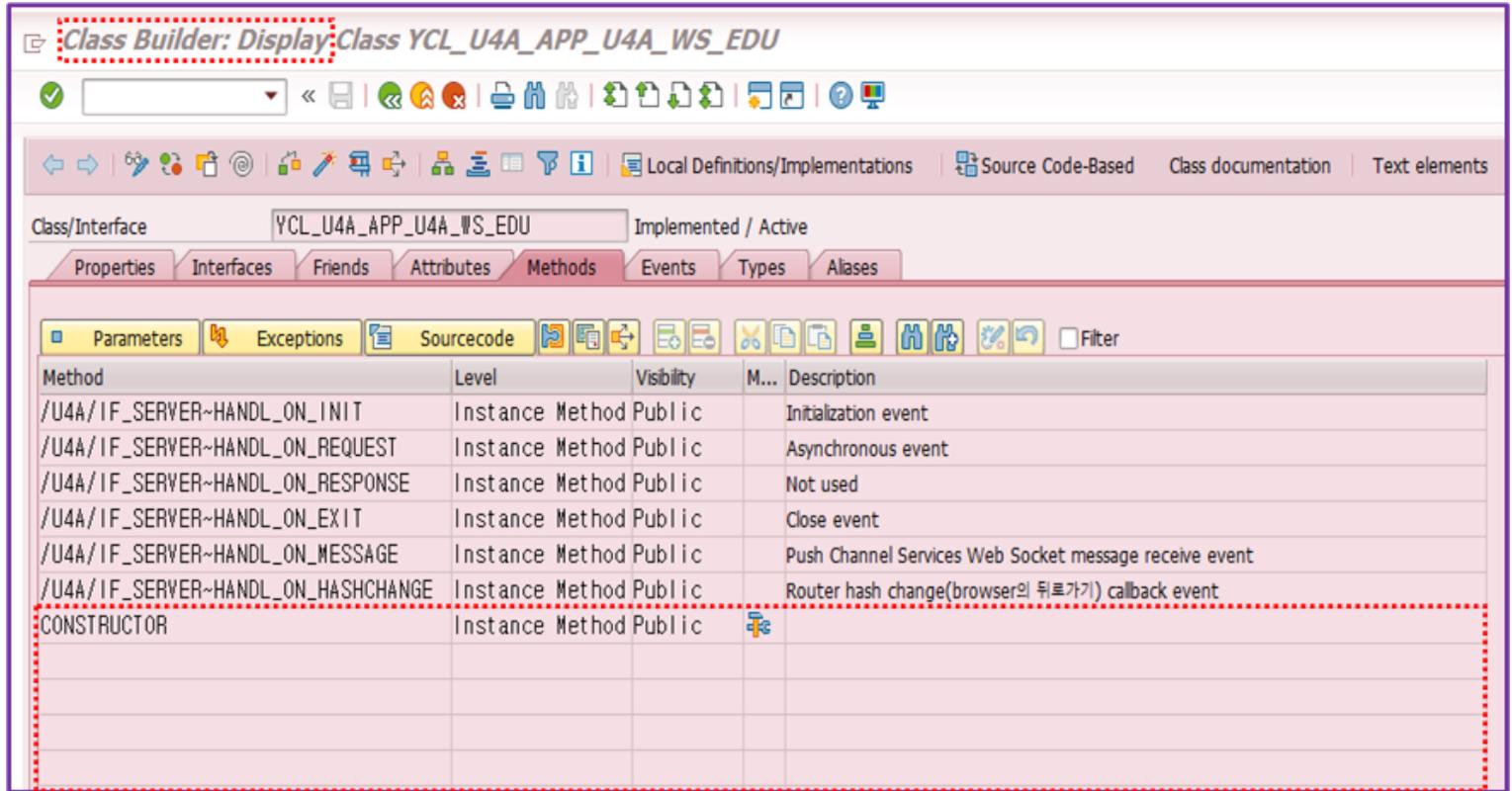
1. 조회하려는 Application ID(Name)을 입력 후 “Display” Button Click 함.



2. 화면에서 좌측 상단에 Display 상태인 것을 확인할 수 있으며 Object id, Description, Attributes 등 수정이 불가능함.

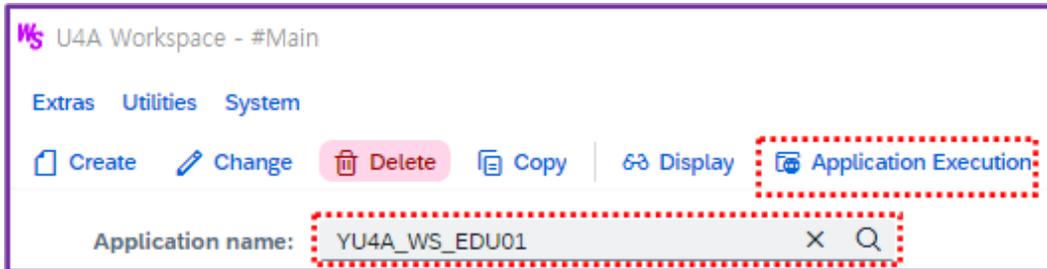


3. Class Builder의 상단에 Display 상태를 나타내며 Method 등 생성하거나 수정이 불가능함.

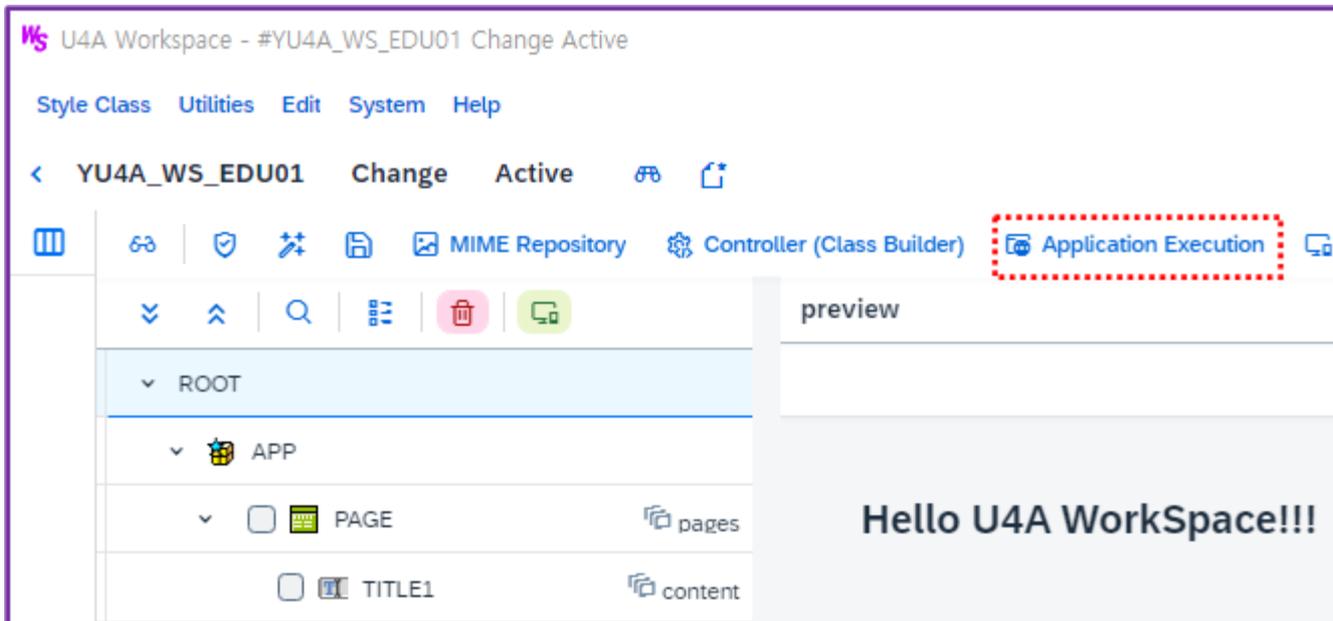


4.13 Application 실행 하기

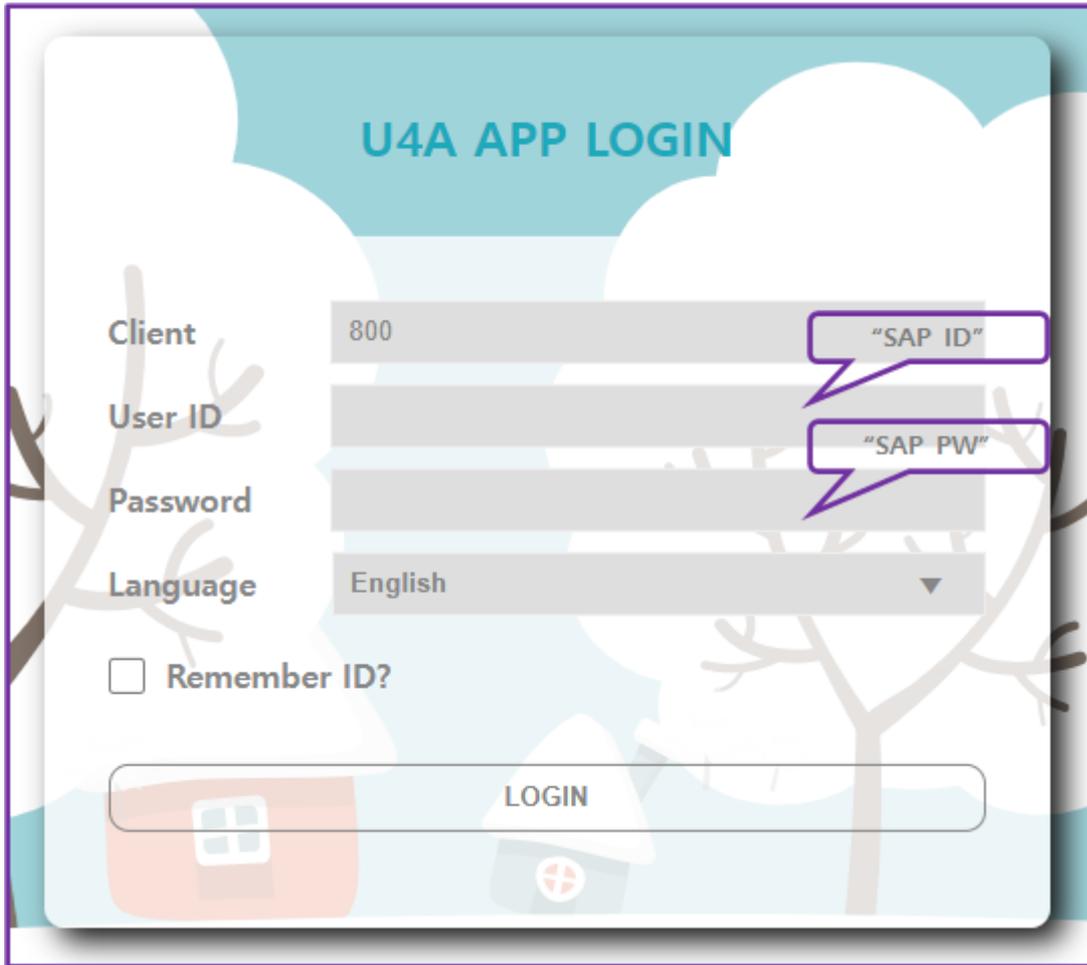
1. 실행 하려는 Application ID(Name)을 입력 후 “Application Execution” Button Click 함.



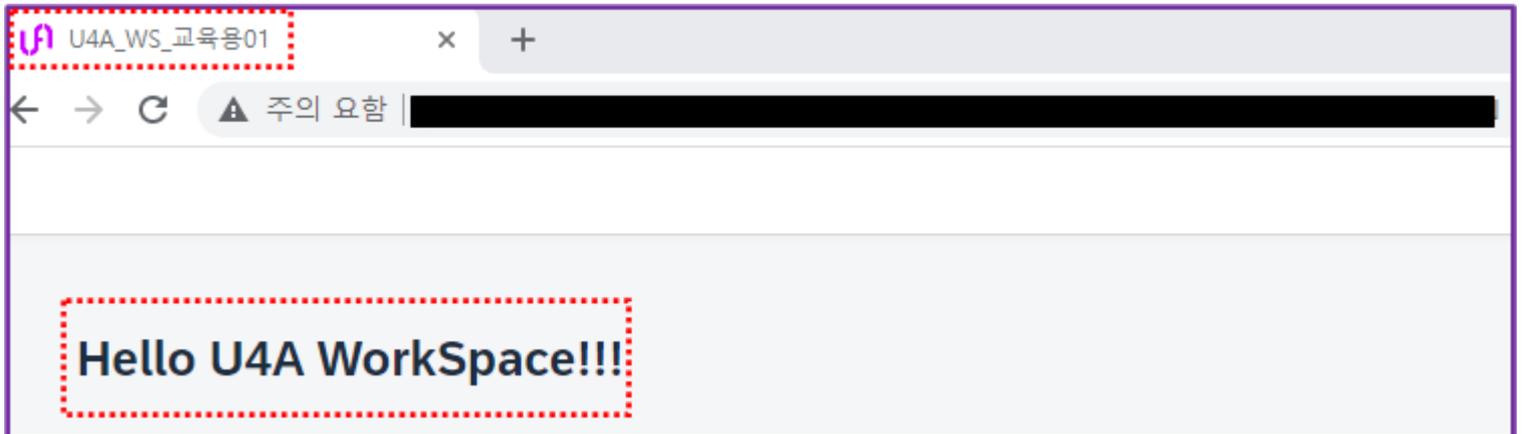
2. Display 또는 Change 중인 상태에서도 Application 실행이 가능함.



3. Login Popup이 뜨면 ID와 Password를 쓰고 Login Button을 누름.



4. Web Page에 실행한 Application 화면이 나옴.



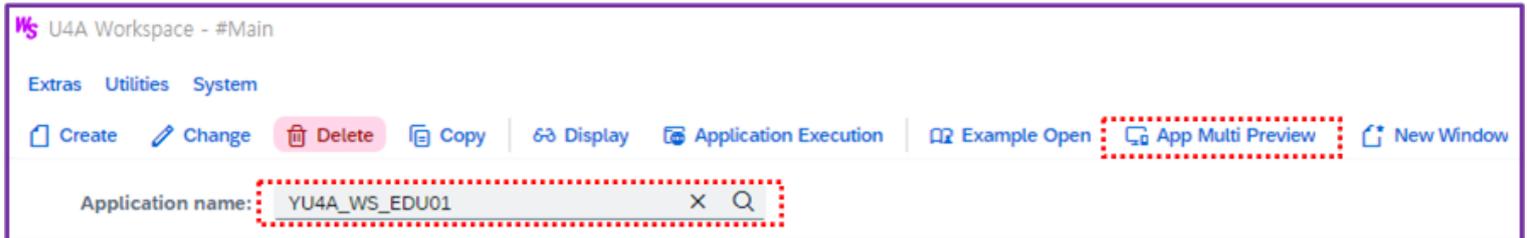
 Application을 실행하려면 반드시 Active 상태여야 함.

 Only in activity state !!!

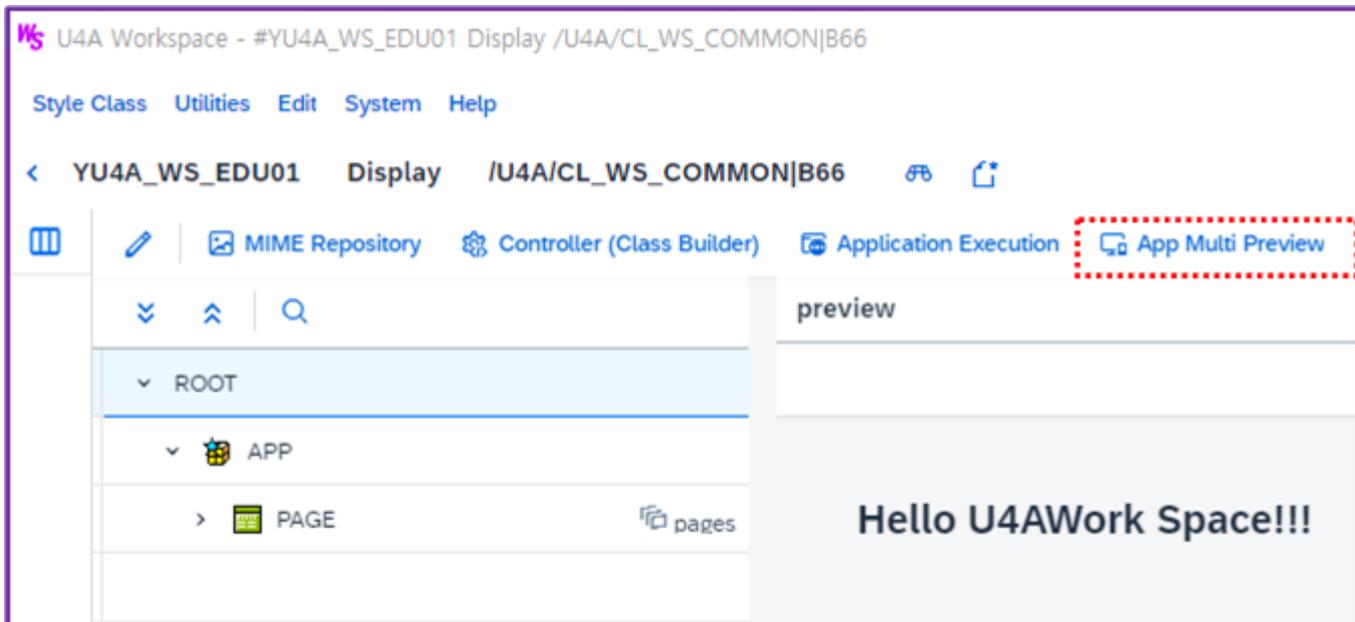
4.14 App Mult Preview

■ 목적 : 생성 된 Application을 다양한 Device에서 실행했을 때 화면을 보기 위함 .

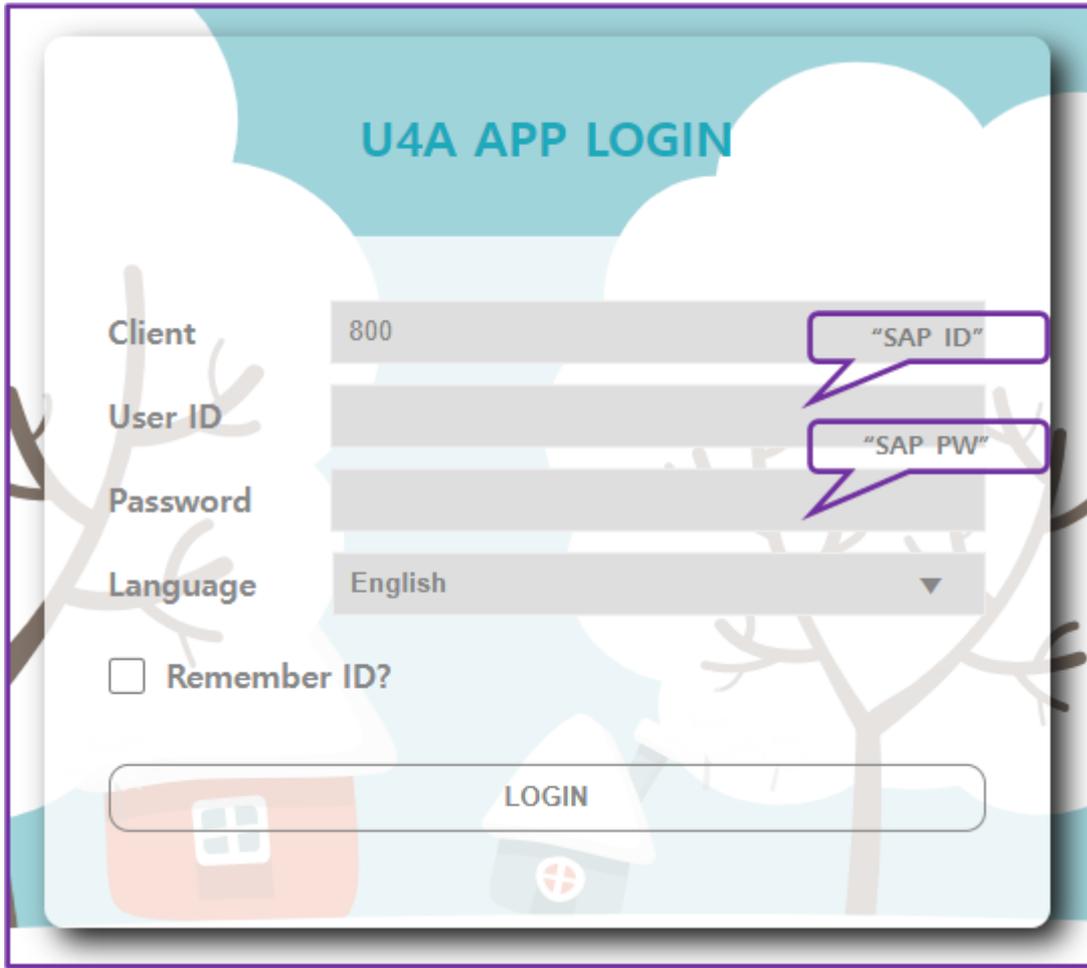
1. App Mult Preview를 실행 할 Application ID(Name)을 입력 후 " App Mult Preview" Button Click 함.



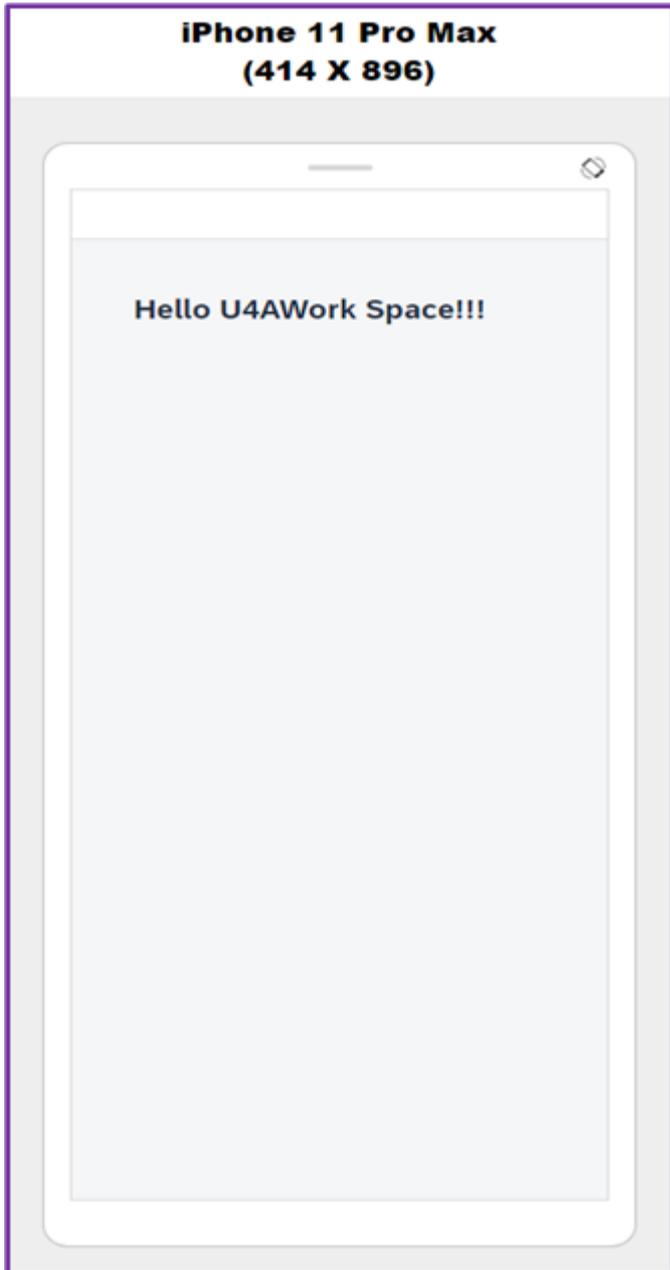
2. Application 내에서 Display 또는 Change 중인 상태에서도 App Mult Preview 실행이 가능함.



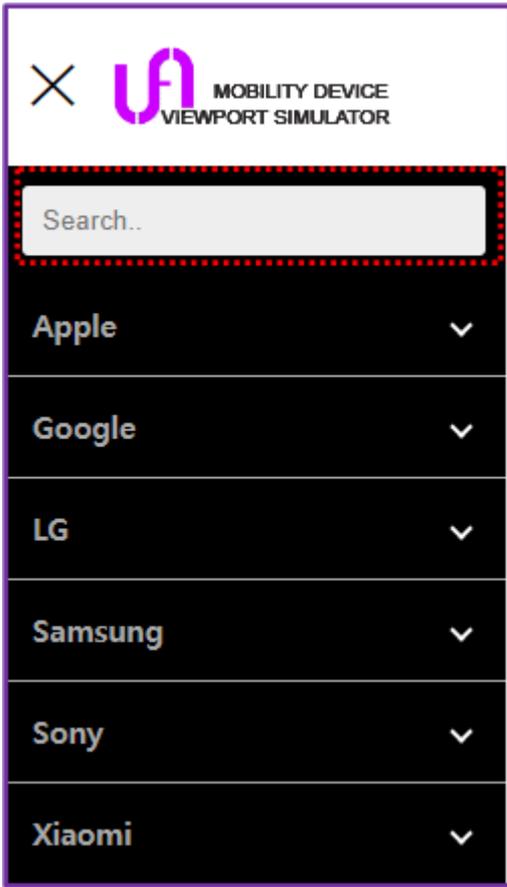
3. Login Popup이 뜨면 ID와 Password를 쓰고 Login Button을 누름.



4) 다른 Device에서 동작 했을 시 실행 화면을 Preview로 볼 수 있음 .



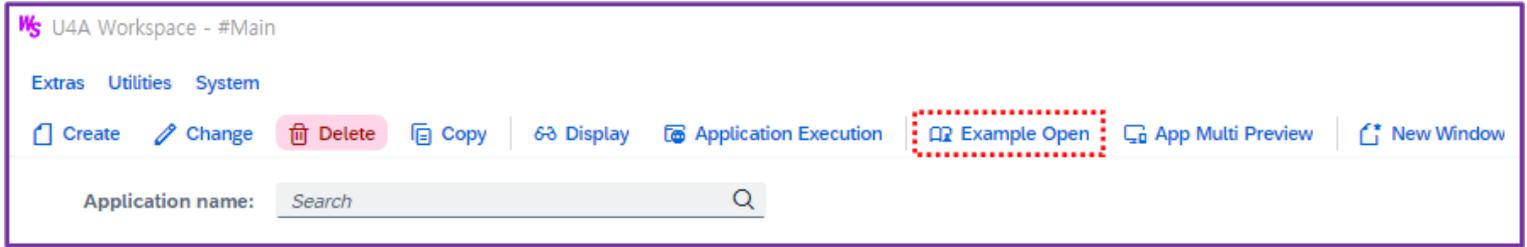
5) 왼쪽 Menu에서 Preview를 보고 싶은 Device를 검색하거나 선택할 수 있음 .



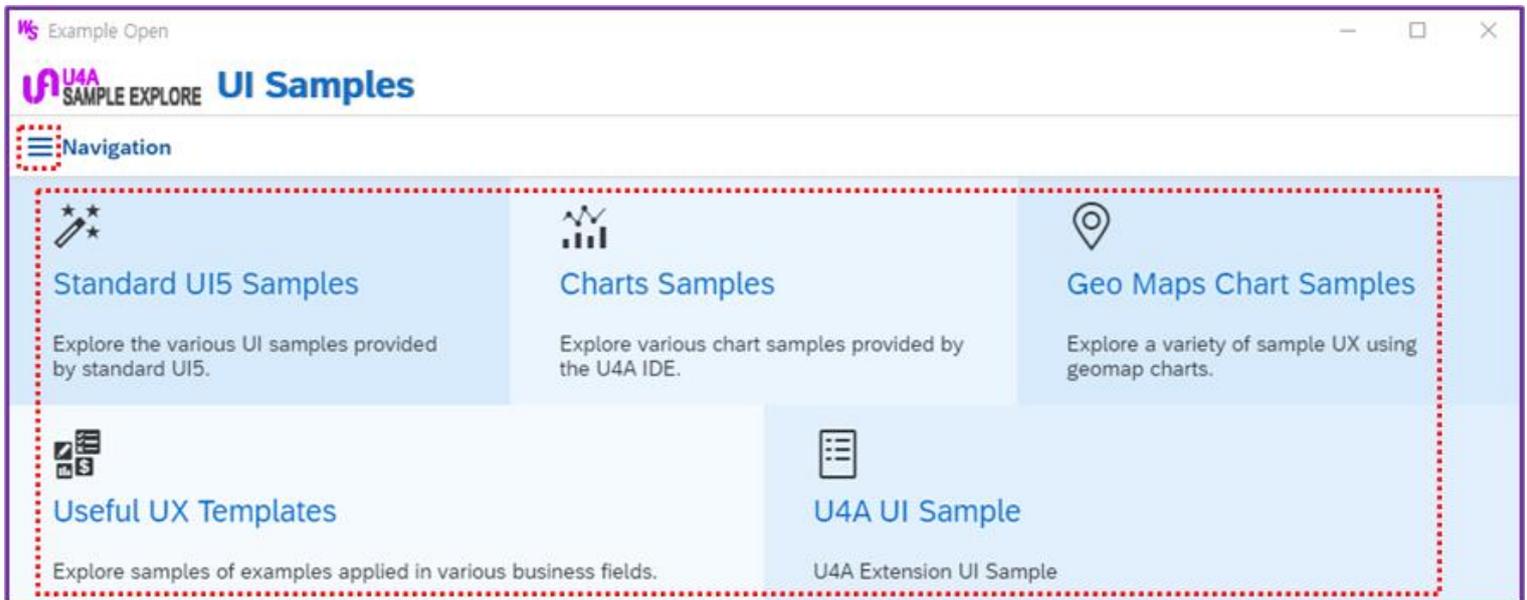
6) 우측 상단에서 원하는 Width, Height 길이를 Preview 화면에 적용시킬 수 있음 .



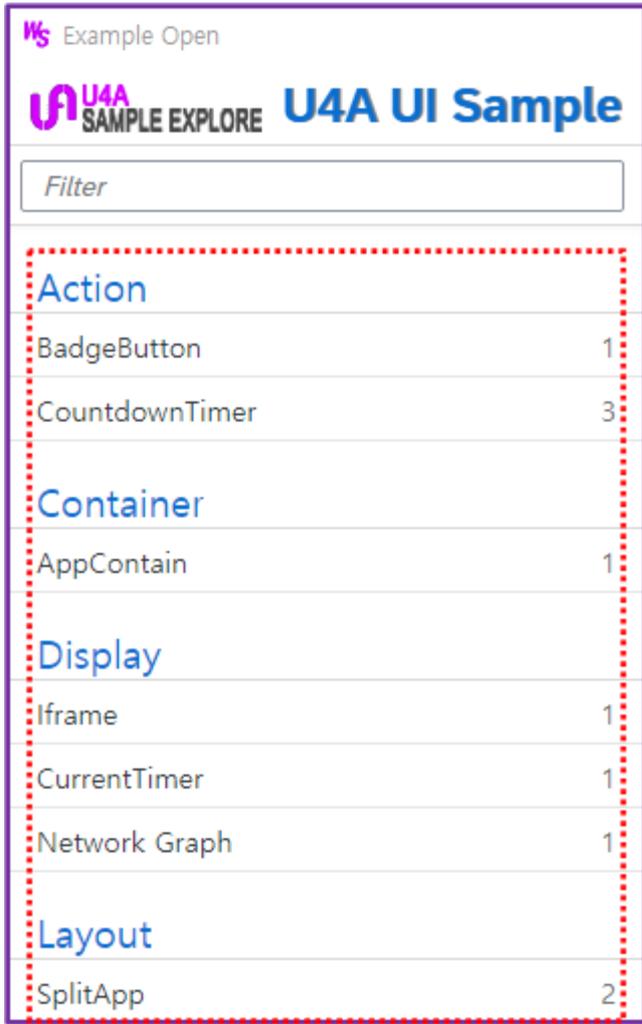
4.15 Example Open



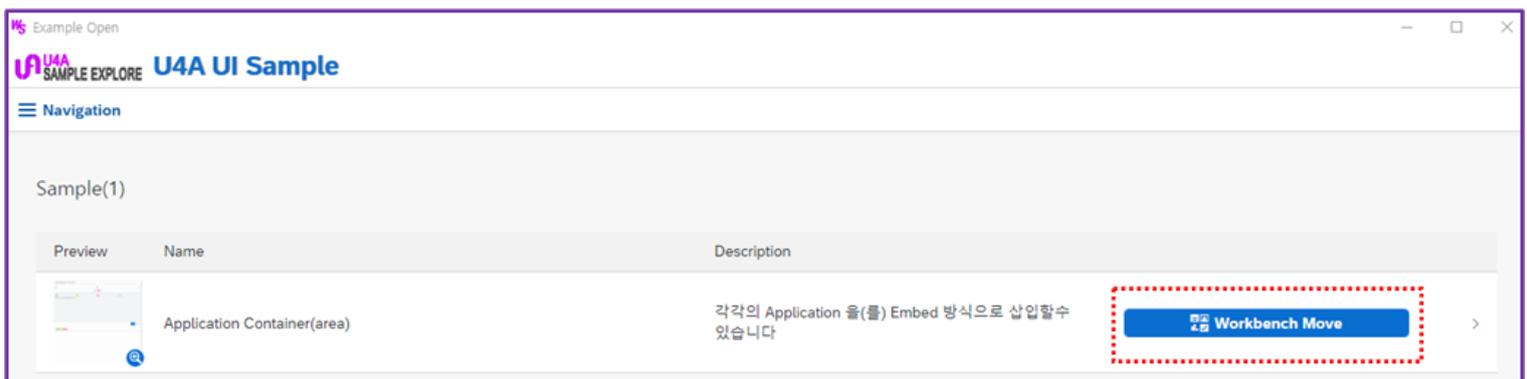
2. 해당 Menu에서 원하는 Sample Category를 선택하거나 Side Menu Click 함.



3. 선택한 Category에 해당하는 목록에서 원하는 Sample 선택함 .

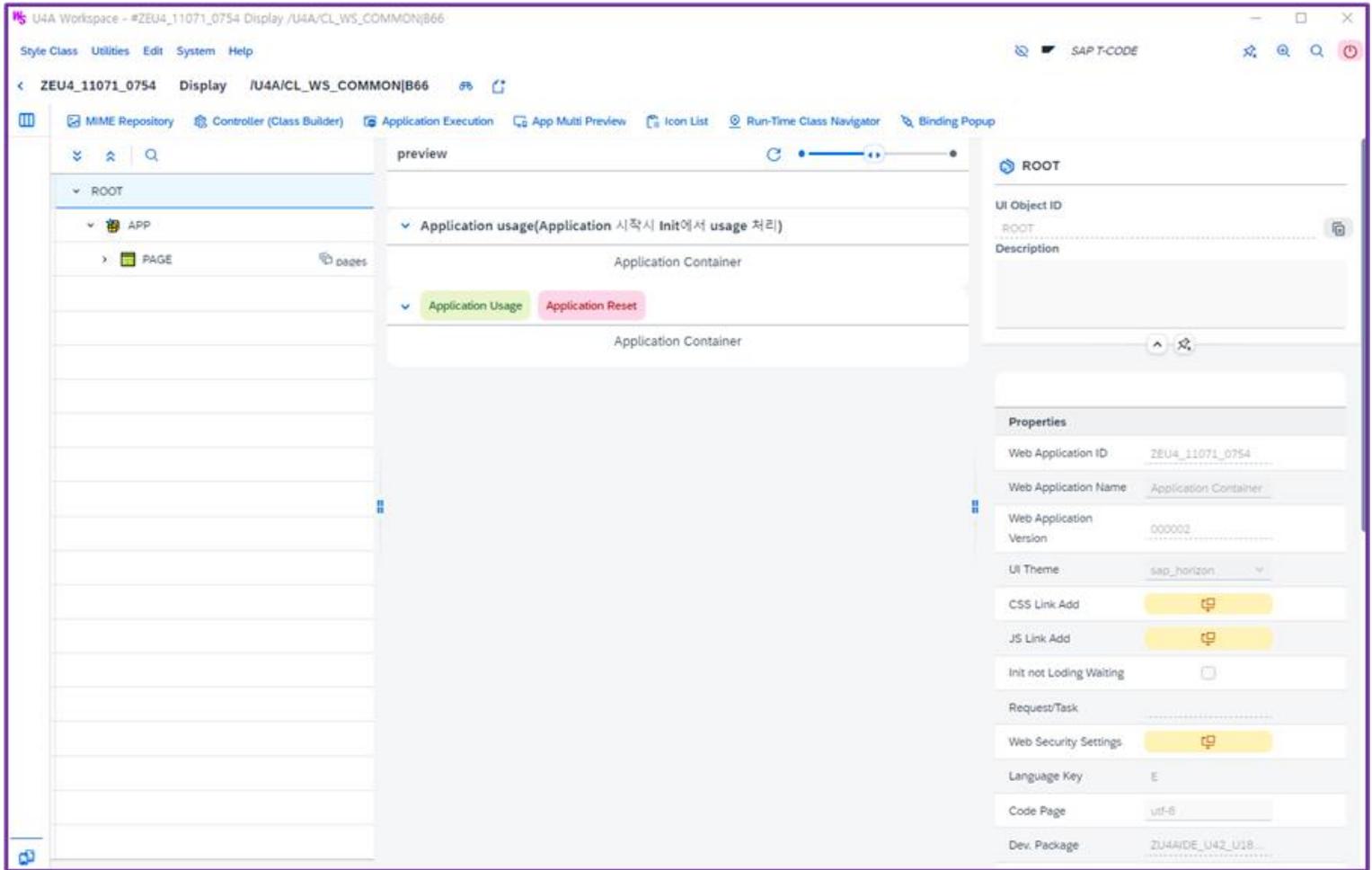


4. "Workbench Move" Button을 Click .

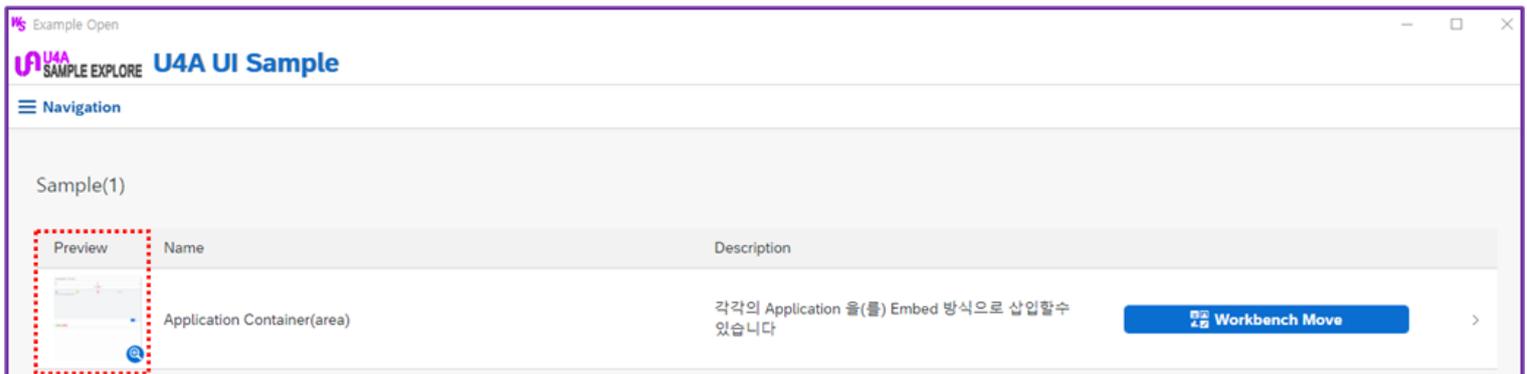


5. 선택한 Sample Workbench로 이동 .

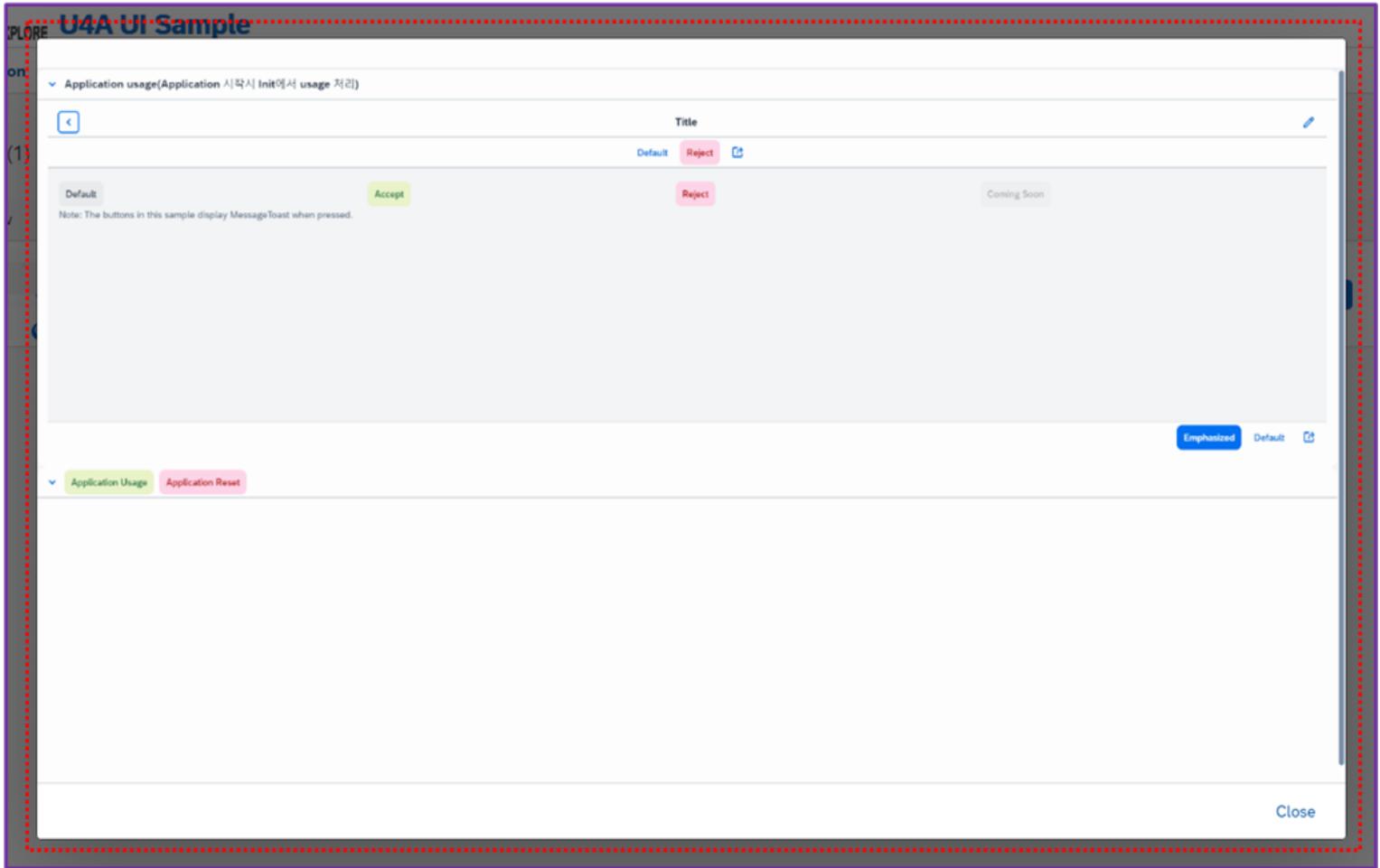
U4A IDE HELP



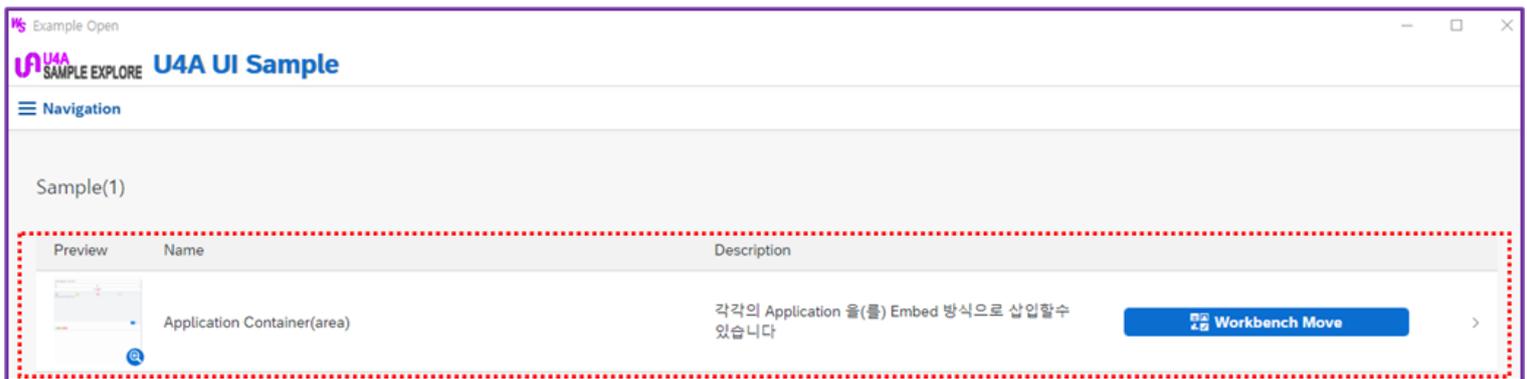
6. Sample Privity Click .



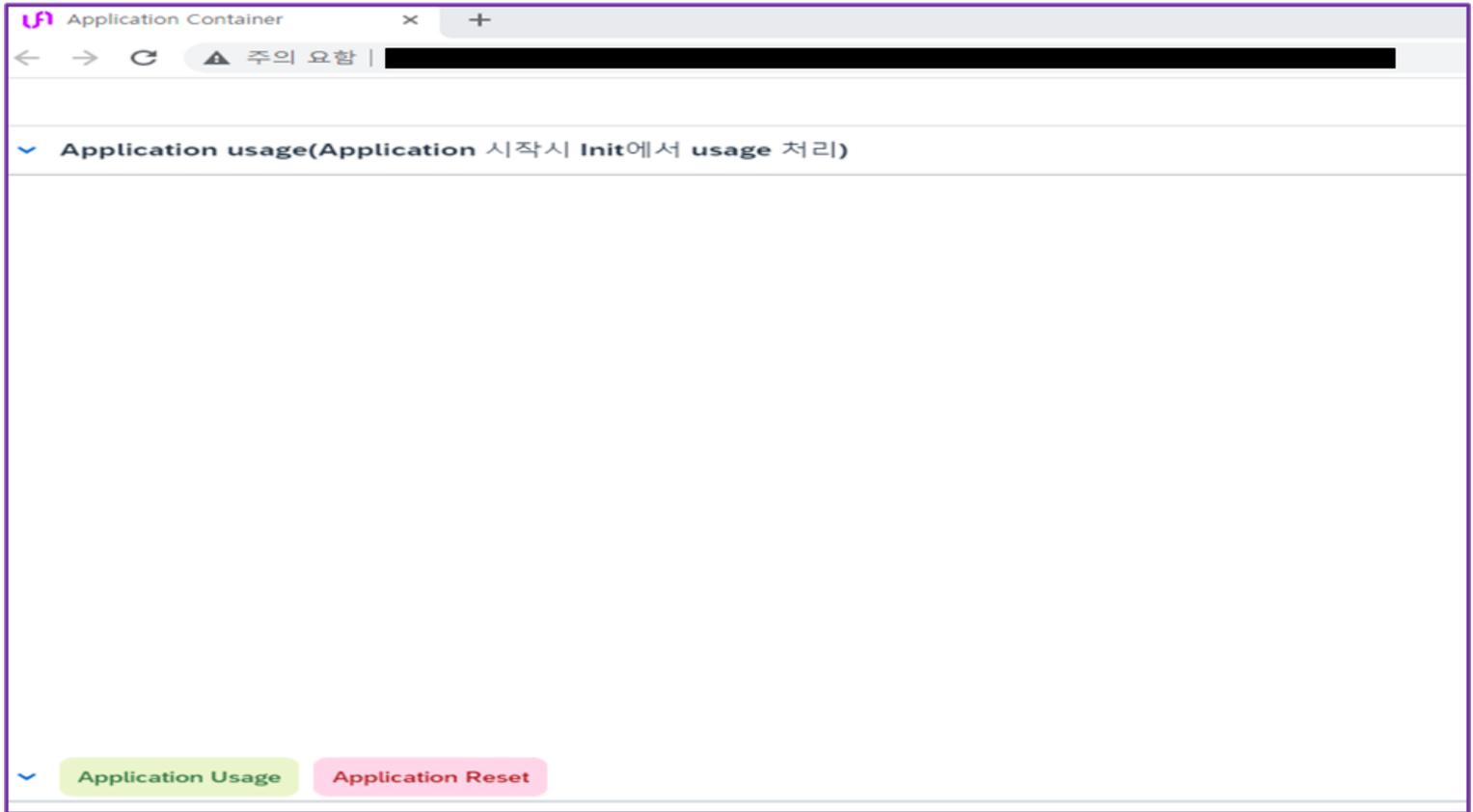
7. Sample Privity Image 확인 .



8. Sample 여백 Click.



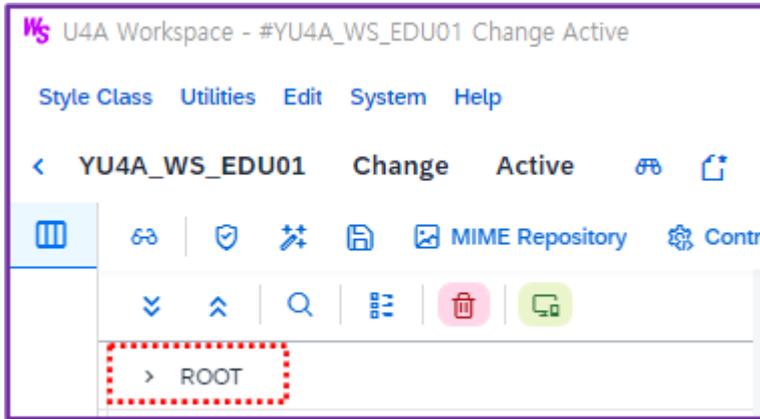
9. Sample Web 실행 화면 확인.



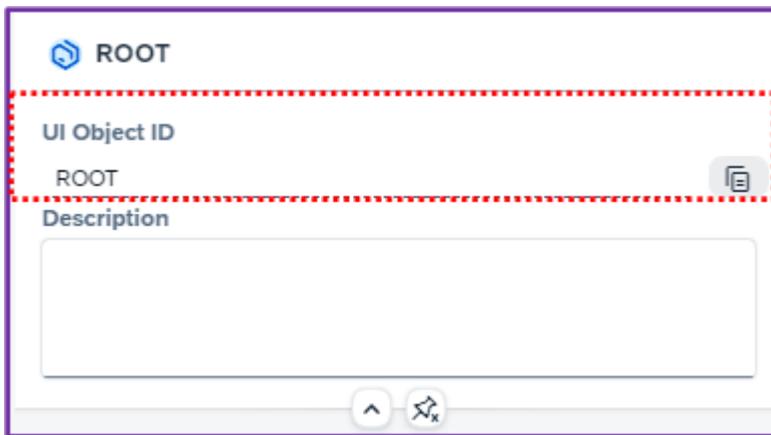
4.16 ROOT(Document)

■ 목적 : 현재 Application에 대한 상세정보 확인 가능 .

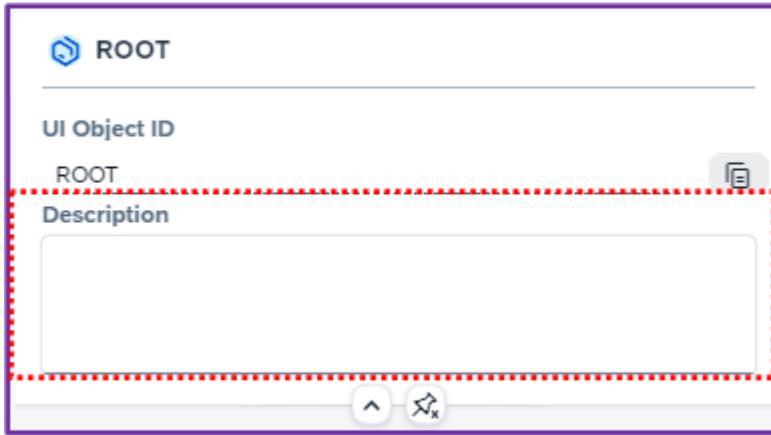
1. UX Design Area에서 ROOT(Document) 정보 확인을 위해 Click .



2. 현재 선택된 Document의 명칭.



3. 해당 Application에 대한 상세 Description을 적을 수 있는 항목.



4. 해당 Application에 대한 상세속성 정보

U4A IDE HELP

Properties	
Web Application ID	YU4A_WS_EDU01
Web Application Name	U4A_WS_교육01
Web Application Version	000001
UI Theme	sap_horizon ▾
CSS Link Add	<input type="text" value=""/>
JS Link Add	<input type="text" value=""/>
Init not Loading Waiting	<input type="checkbox"/>
Request/Task	
Web Security Settings	<input type="text" value=""/>
Language Key	E
Code Page	utf-8 <input type="text" value=""/>
Dev. Package	\$TMP
Assigned Class Object ID	YCL_U4A_APP_U4A_WS_EDU01
Program ID in Requests and Tasks	R3TR
Object Type	UXSD
Authorization Group	<input type="text" value=""/>
Whether to use Router	<input type="checkbox"/>
Whether to use Skeleton Screen	<input type="checkbox"/>
Whether to use Mobile Zoom	<input type="checkbox"/>
Wait Type	▾
Touch style	▾
Shortcut Icon URL	<input type="text" value=""/>
Body Background Color	<input type="text" value=""/>
Create User	U4A_TM01
Create Date	20221219
Create Time	175433
Change User	<input type="text" value=""/>
Change Date	<input type="text" value=""/>
Change Time	<input type="text" value=""/>

Properties	Means
Web Application ID	Application ID
Web Application Name	Application에 대한 설명 및 실행 시 타이틀 명
Web Application Version	Application Version

UI Theme	<p>UI 테마 설정</p> <ul style="list-style-type: none"> • base • sap_belize • sap_belize_hcb • sap_belize_hcw • sap_belize_plus • sap_bluecrystal • sap_hcb • sap_fiori_3 • sap_fiori_3_dark • sap_fiori_3_hcb • sap_fiori_3_hcw • sap_horizon • sap_horizon_dark • sap_horizon_hcb • sap_horizon_hcw
Css Link Add	마임에서 등록한 CSS 파일 등록
Js Link Add	마임에서 등록한 JS 파일 등록
Init not Loading Waiting	실행 했을 때 로딩 화면을 보여줄 지 여부
Request/Task	Application과 연결되어 있는 CTS 번호
Web Security Settings	Web 접근에 대한 권한 설정
Language Key	Application 생성 시 설정한 언어
Code Page	Code set
Dev. Package	개발 패키지 명
Assigned Class Object ID	컨트롤러 클래스 명
Program ID in Requests and Tasks	R3TR (Only displaid)
Object Type	Web Service Component (Only displaid)
Authorization Group	
Whether to use Router	Browser History Back Control (X = Control)

U4A IDE HELP

Whether to use Skeleton Screen	Skeleton Screen 적용 여부 체크
Whether to use Mobile Zoom	모바일 확대/축소 사용 여부 (X = 사용)
Wait Type	Busy Indicator Loading Bar
Touch style	클릭 시 스타일 설정 :Circle_ripple
Shortcut Icon URL	Application 실행 시 Shortcut Icon 생성되는 Image Url
Body Background Color	Default HTML Body Color (RGB CODE #FFFFFF)
Create User	Application 생성 유저
Create Date	Application 생성 일자
Create Time	Application 생성 시간
Change User	Application 변경 유저
Change Date	Application 변경 일자
Change Time	Application 변경 시간

4.16.1 Touch style

■ 목적 : Click 했을 때 Style 적용 가능.

1. 해당 Application에서 Click 시 효과를 주는 것이 가능하며 종류는 1가지.

ROOT => Properties => TouchStyle.

Properties	
Web Application ID	YU4A_WS_EDU03
Web Application Name	U4A_WS_교육용03
Web Application Version	000001
UI Theme	sap_horizon
CSS Link Add	
JS Link Add	
Init not Loading Waiting	<input type="checkbox"/>
Request/Task	
Web Security Settings	
Language Key	E
Code Page	utf-8
Dev. Package	\$TMP
Assigned Class Object ID	YCL_U4A_APP_U4A_WS_EDU03
Program ID in Requests and Tasks	R3TR
Object Type	UXSD
Authorization Group	
Whether to use Router	<input type="checkbox"/>
Whether to use Skeleton Screen	<input type="checkbox"/>
Whether to use Mobile Zoom	<input type="checkbox"/>
Wait Type	
Touch style	Circle
Shortcut Icon URL	

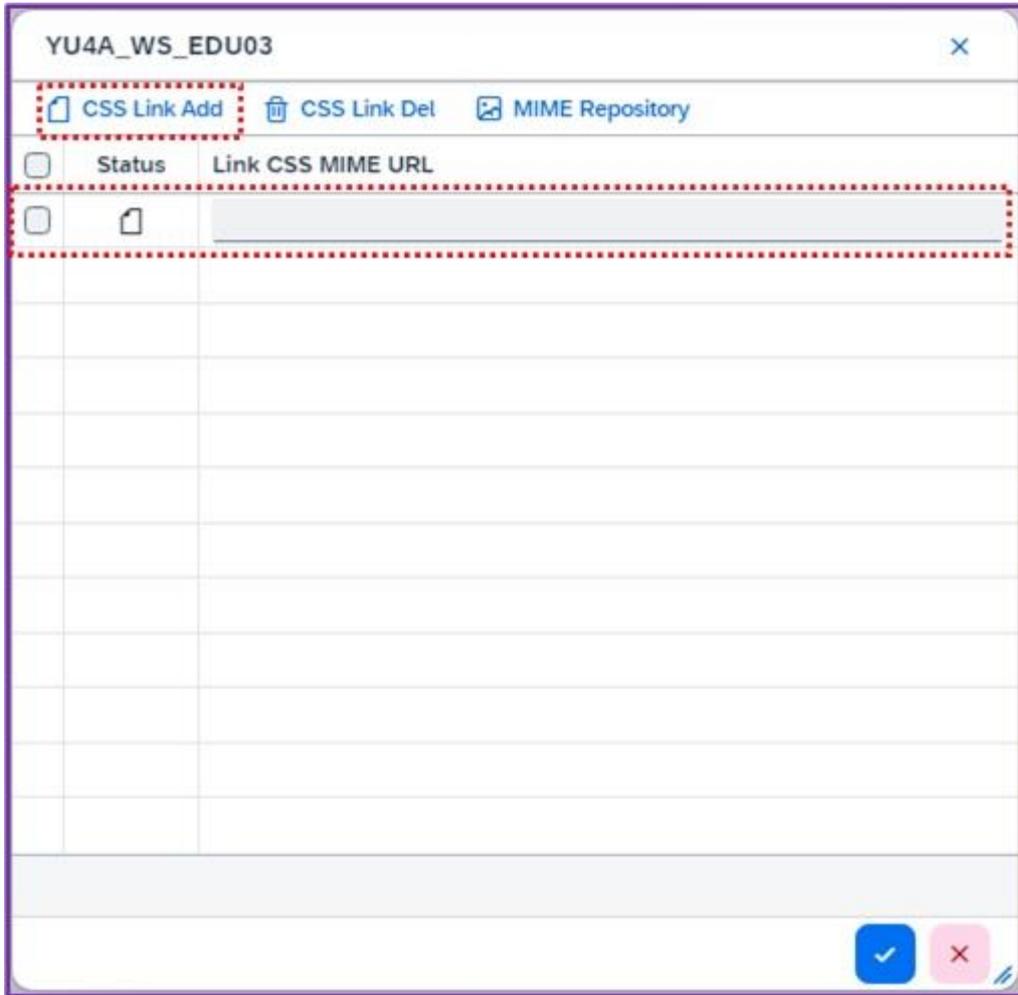
2. Circle 적용 화면으로 Button Click 시 원형 효과가 생김.



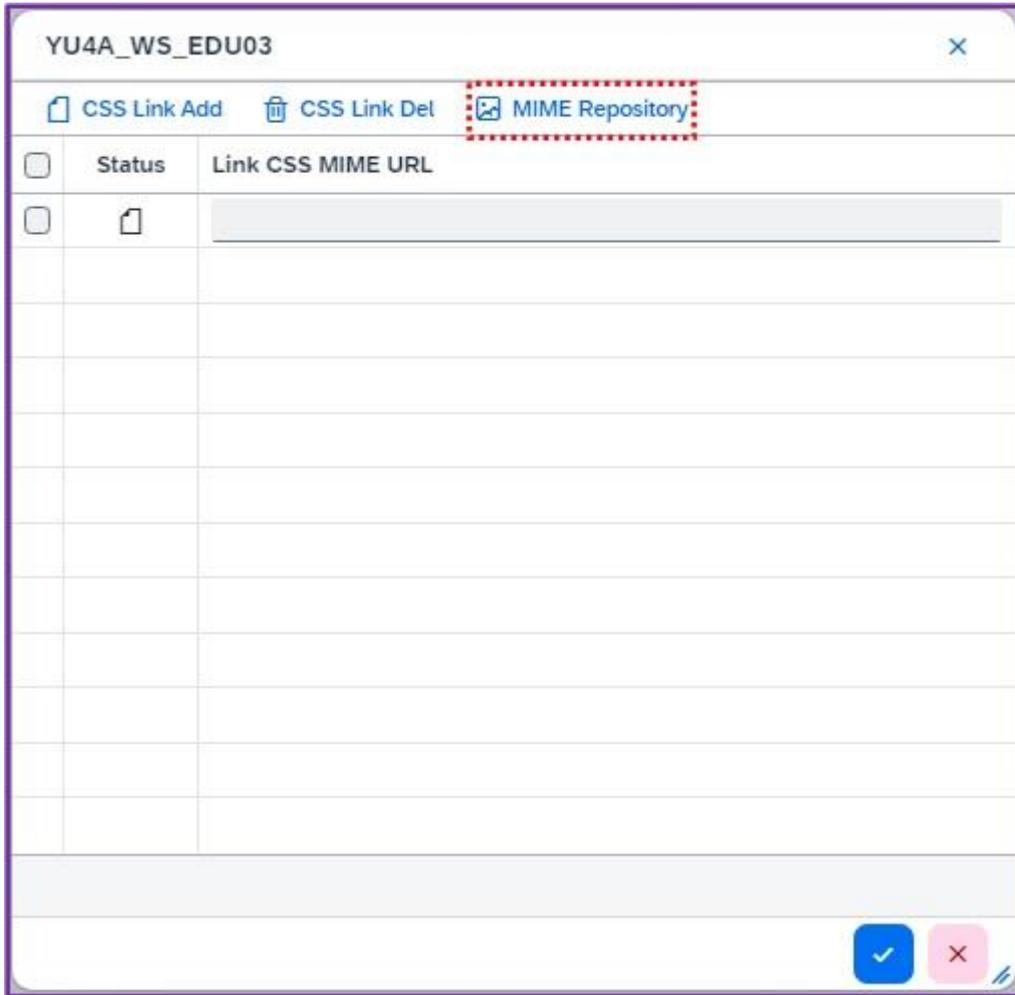
4.16.2 CSS Link Add

Properties	
Web Application ID	YU4A_WS_EDU03
Web Application Name	U4A_WS_교육용03
Web Application Version	000001
UI Theme	sap_horizon
CSS Link Add	
JS Link Add	
Init not Loading Waiting	<input type="checkbox"/>
Request/Task	
Web Security Settings	
Language Key	E
Code Page	utf-8 
Dev. Package	\$TMP
Assigned Class Object ID	YCL_U4A_APP_U4A_WS_EDU03
Program ID in Requests and Tasks	R3TR
Object Type	UXSD
Authorization Group	
Whether to use Router	<input type="checkbox"/>
Whether to use Skeleton Screen	<input type="checkbox"/>
Whether to use Mobile Zoom	<input type="checkbox"/>
Wait Type	
Touch style	Circle
Shortcut Icon URL	

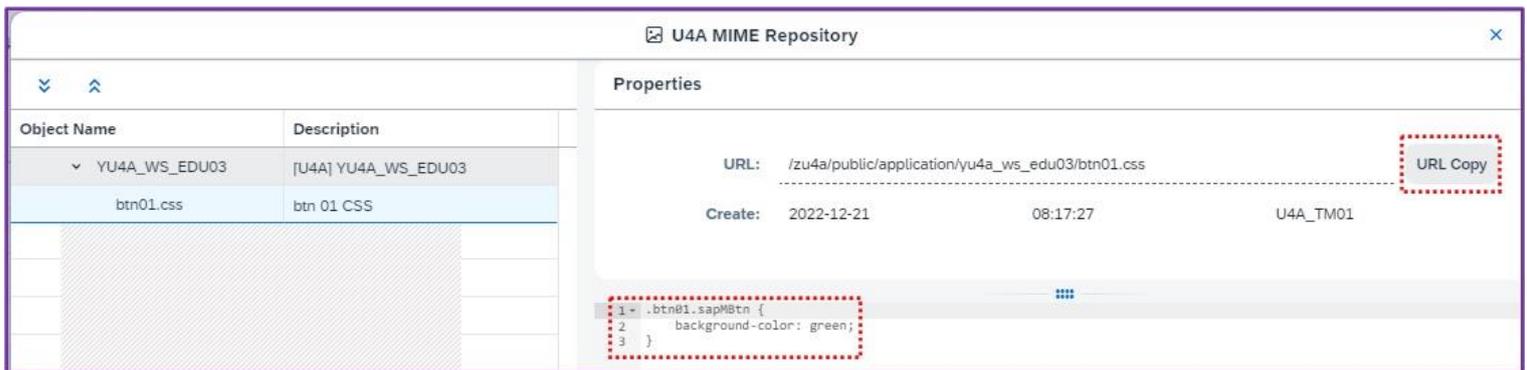
2. "CSS Link Add" Button을 눌러 원하는 만큼 URL 등록창 생성.



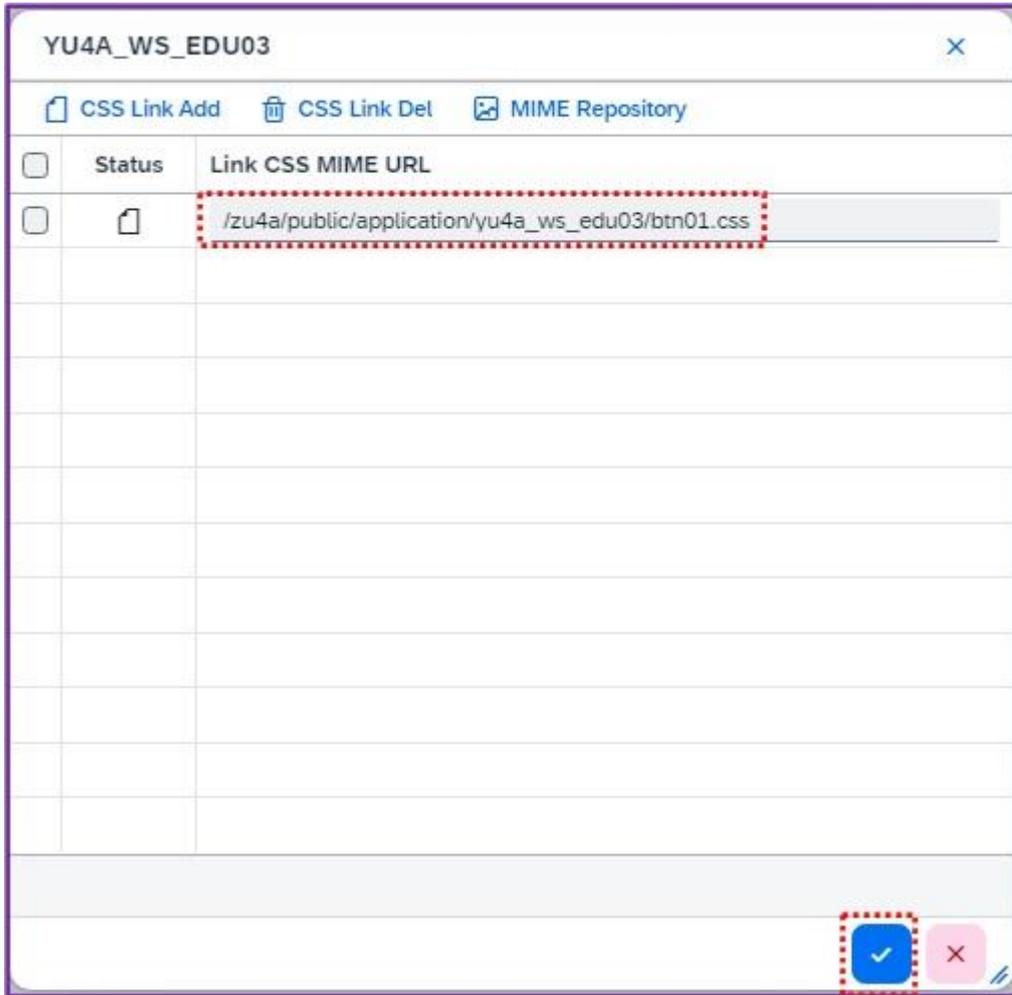
3. "MIME Repository" Button을 눌러 원하는 CSS File 등록창 생성.



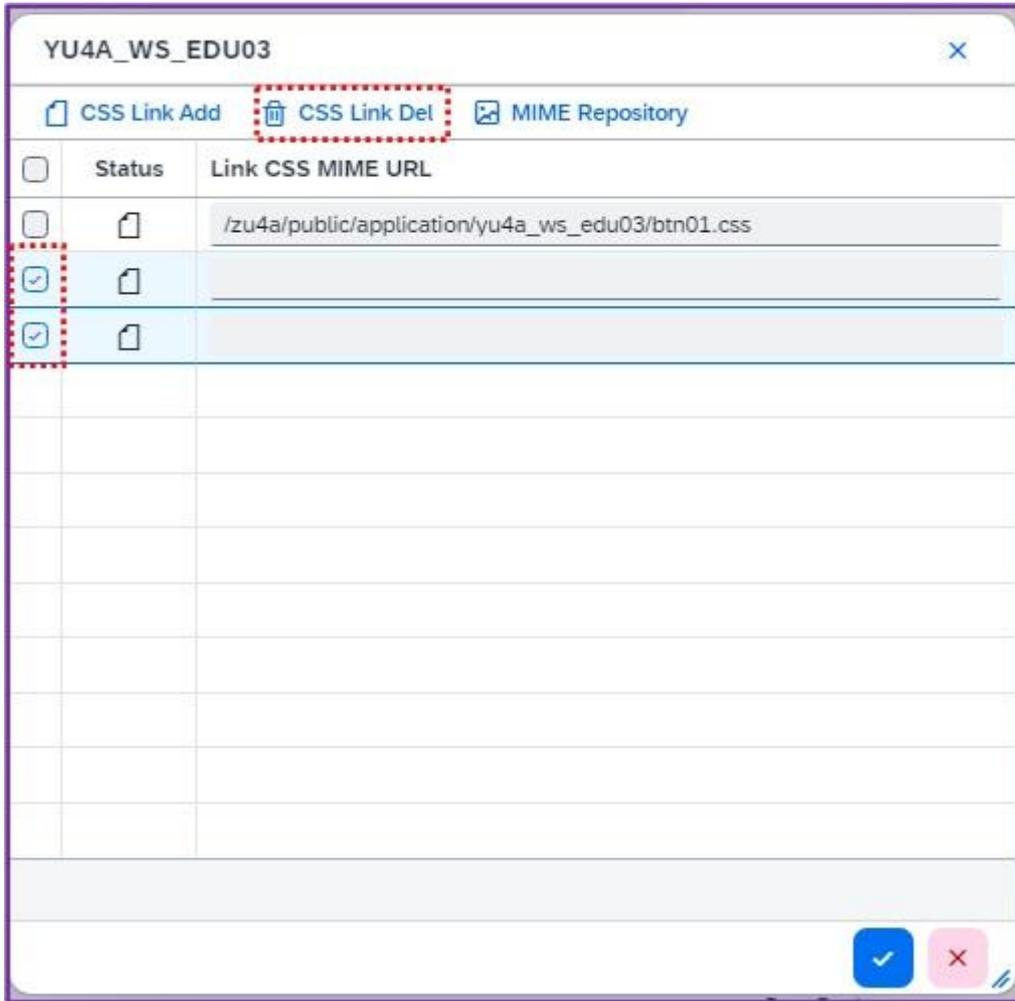
4. 해당 화면에서 원하는 CSS File을 선택 한 후 "URL Copy" Button을 눌러 URL 복사.



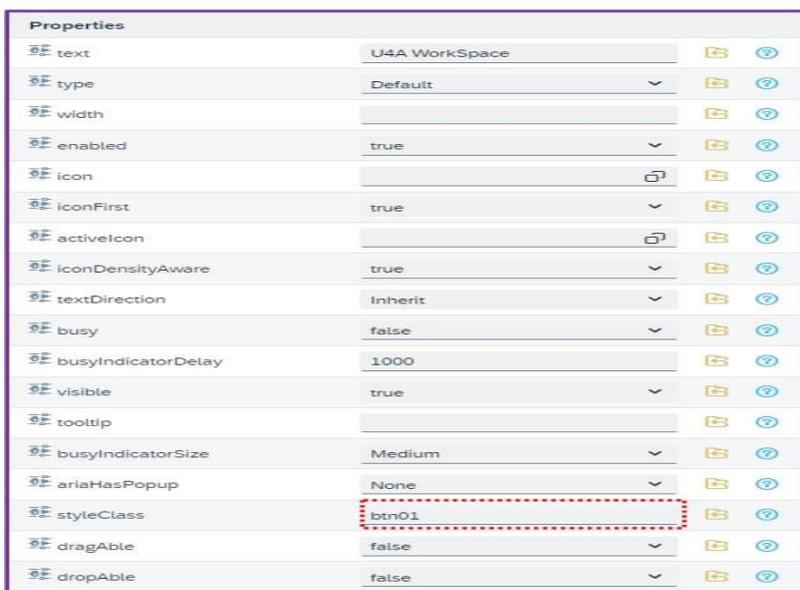
5. 복사한 URL을 Link CSS MIME URL 등록창에 붙여넣기.



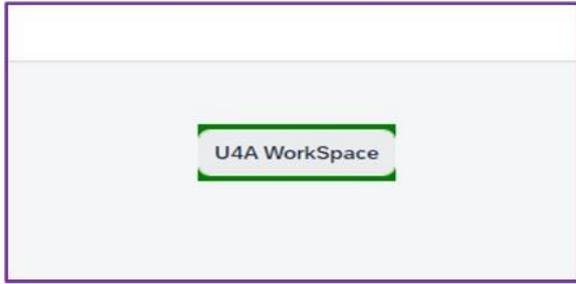
5-1 삭제하려는 URL을 원하는 만큼 선택한 후 "CSS Link Del" Button을 눌러 삭제.



6. CSS를 적용 할 UI의 Property에서 styleClass에 CSS File 입력(btn01).



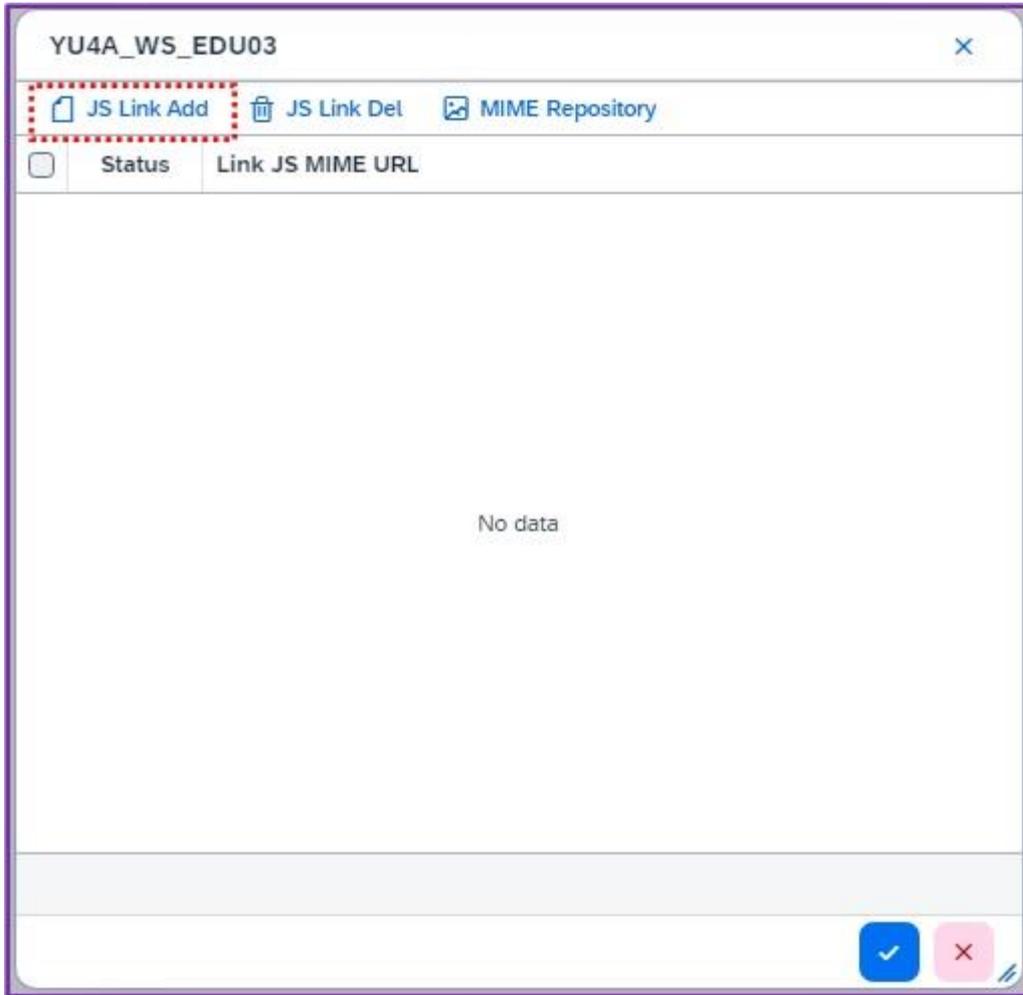
7. Application 실행하여 CSS 적용된 것을 확인.



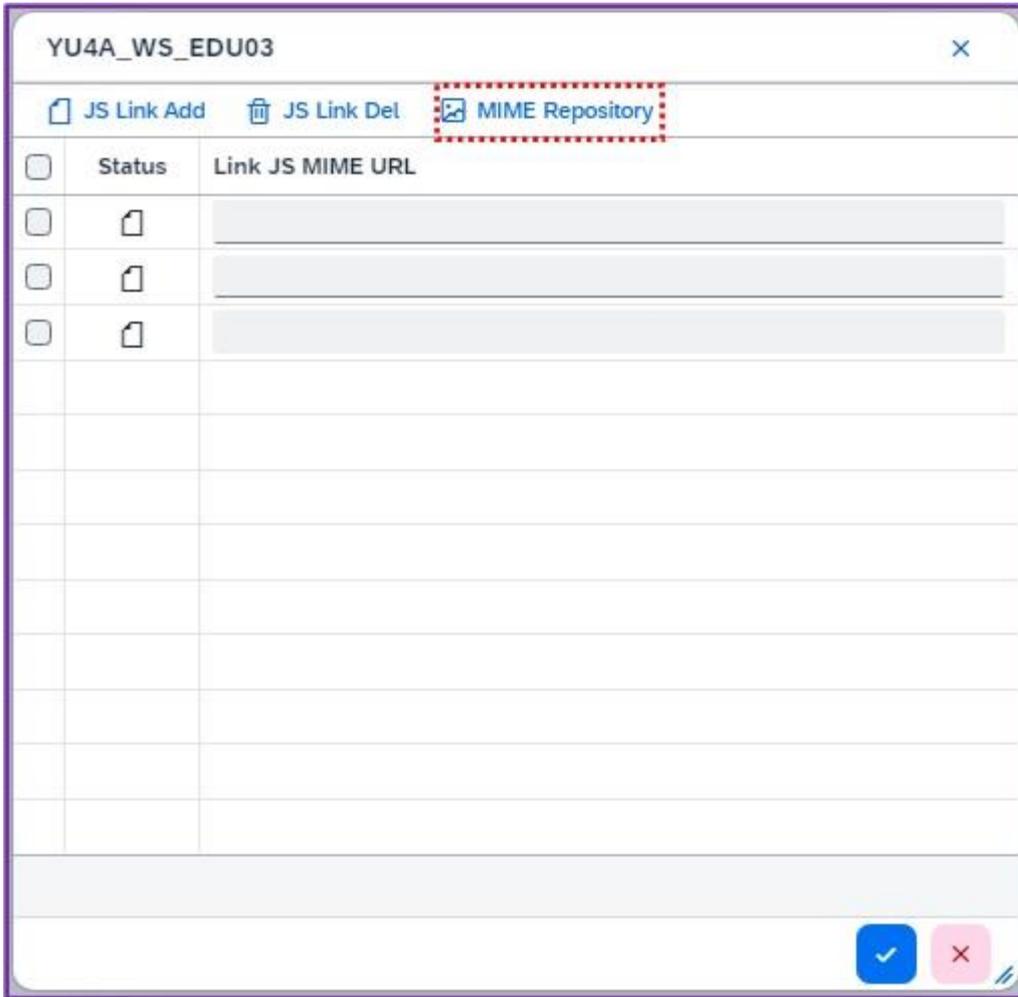
4.16.3 JS Link Add

Properties	
Web Application ID	YU4A_WS_EDU03
Web Application Name	U4A_WS_교육용03
Web Application Version	000001
UI Theme	sap_horizon
CSS Link Add	
JS Link Add	
Init not Loading Waiting	<input type="checkbox"/>
Request/Task	
Web Security Settings	
Language Key	E
Code Page	utf-8 
Dev. Package	\$TMP
Assigned Class Object ID	YCL_U4A_APP_U4A_WS_EDU03
Program ID in Requests and Tasks	R3TR
Object Type	UXSD
Authorization Group	
Whether to use Router	<input type="checkbox"/>
Whether to use Skeleton Screen	<input type="checkbox"/>
Whether to use Mobile Zoom	<input type="checkbox"/>
Wait Type	
Touch style	Circle
Shortcut Icon URL	

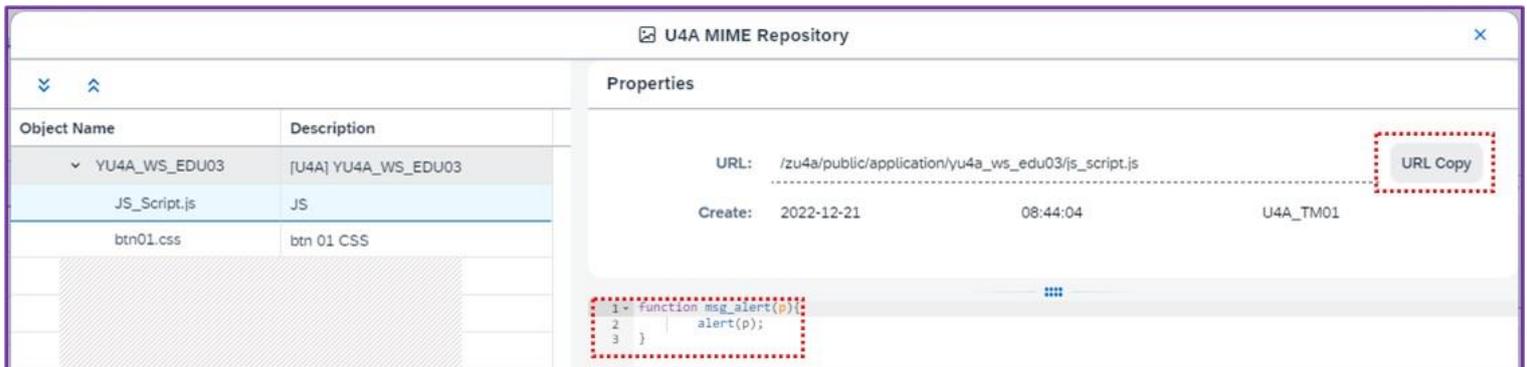
2. "JS Link Add" Button을 눌러 원하는 만큼 URL 등록창 생성.



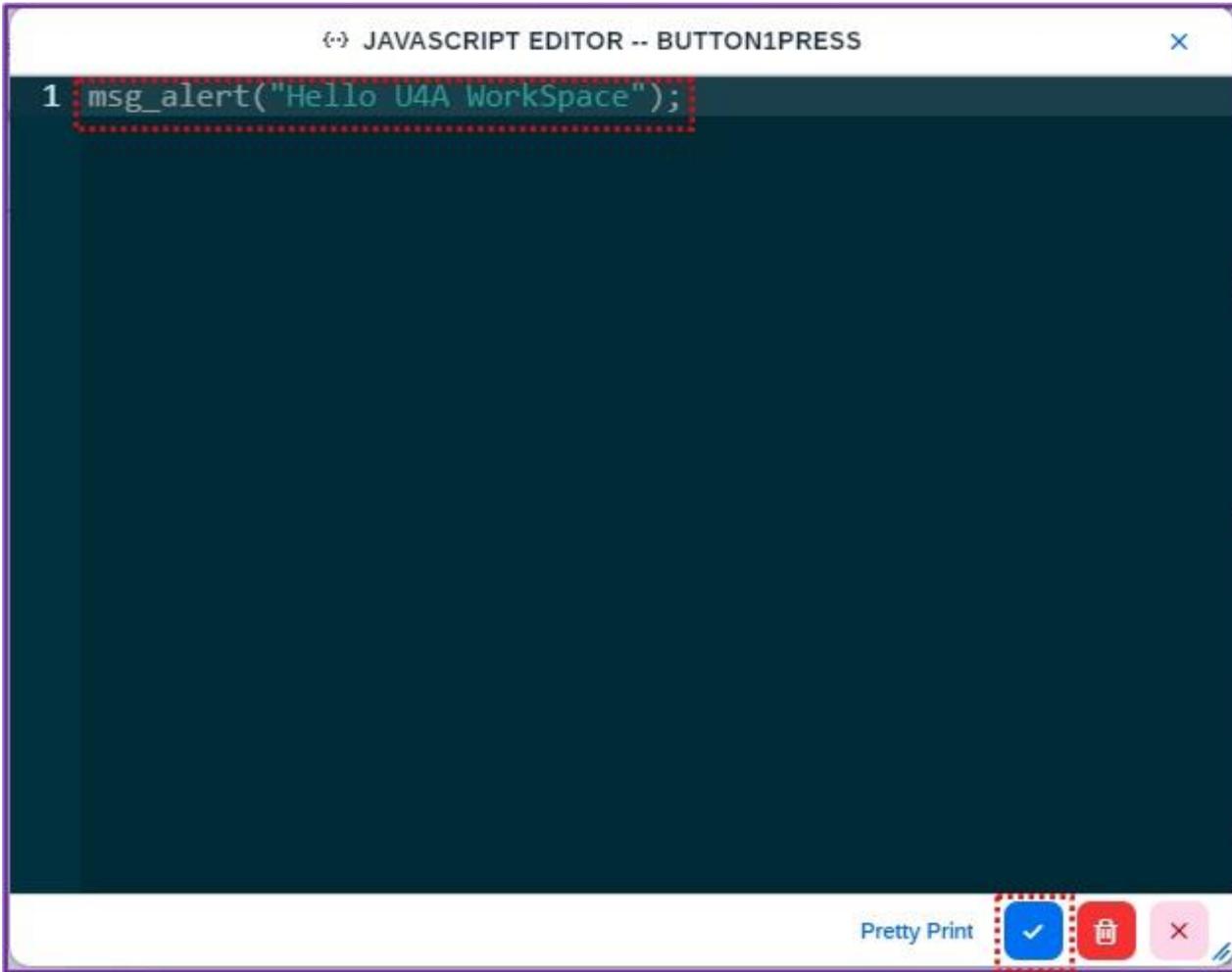
3. "MIME Repository" Button을 눌러 원하는 JS File 등록창 생성.



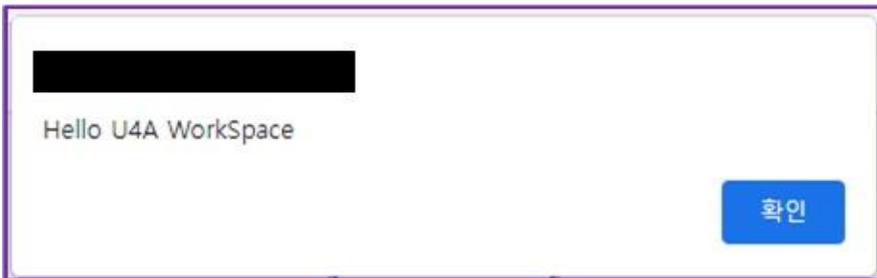
4. 해당 화면에서 원하는 JS File을 선택 한 후 "URL Copy" Button을 눌러 URL 복사.



5. 복사한 URL을 "Link JS MIME URL" 등록창에 붙여넣기.



8. Application 실행한 후 EVENT 발생 시켜 JS 적용된 것을 확인.



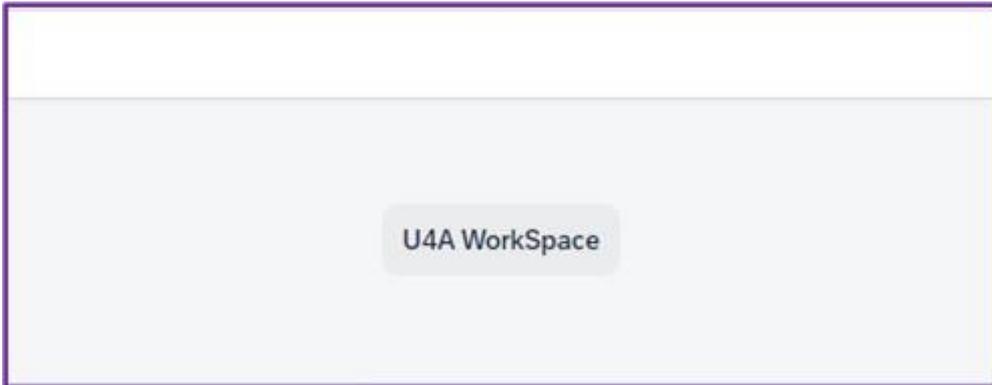
4.16.4 UI Theme

Properties	
Web Application ID	YU4A_WS_EDU03
Web Application Name	U4A_WS_교육용03
Web Application Version	000001
UI Theme	sap_horizon
CSS Link Add	
JS Link Add	
Init not Loading Waiting	<input type="checkbox"/>
Request/Task	
Web Security Settings	
Language Key	E
Code Page	utf-8 
Dev. Package	\$TMP
Assigned Class Object ID	YCL_U4A_APP_U4A_WS_EDU03
Program ID in Requests and Tasks	R3TR
Object Type	UXSD
Authorization Group	
Whether to use Router	<input type="checkbox"/>
Whether to use Skeleton Screen	<input type="checkbox"/>
Whether to use Mobile Zoom	<input type="checkbox"/>
Wait Type	
Touch style	Circle
Shortcut Icon URL	

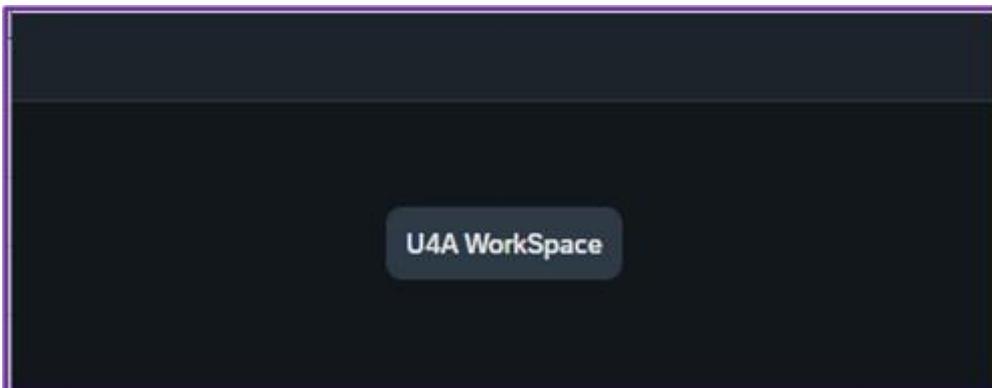
2. UI Theme 종류는 총 15가지 이며 본 단원에선 4가지 Example 소개.

ITMCD	FLD01
U0001	base
U0002	sap_belize
U0003	sap_belize_hcb
U0004	sap_belize_hcw
U0005	sap_belize_plus
U0006	sap_bluecrystal
U0007	sap_hcb
U0008	sap_fiori_3
U0009	sap_fiori_3_dark
U0010	sap_fiori_3_hcb
U0011	sap_fiori_3_hcw
U0012	sap_horizon
U0013	sap_horizon_dark
U0014	sap_horizon_hcb
U0015	sap_horizon_hcw

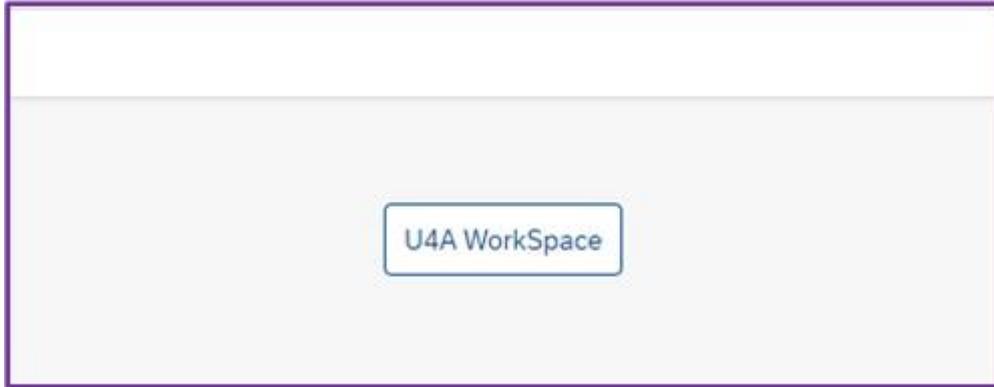
3. sap_horizon Theme 적용한 UI.



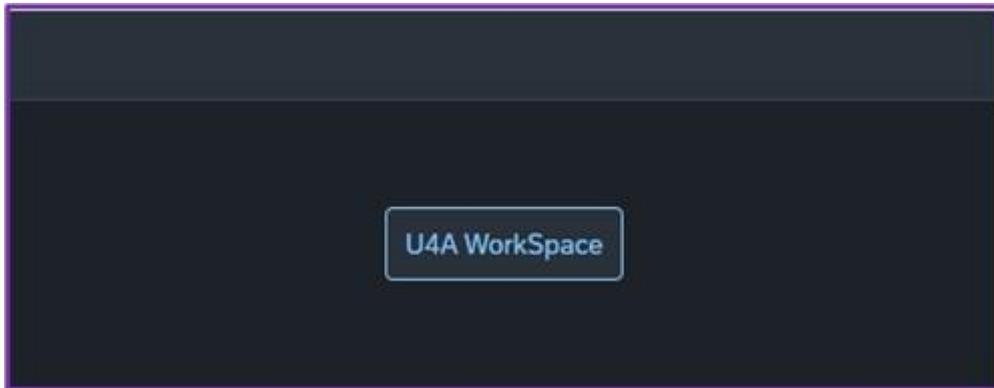
4. sap_horizon_dark Theme 적용한 UI.



5. sap_fiori_3 Theme 적용한 UI.



6. Sap_fiori_3_dark Theme 적용한 UI.



4.16.5 Wait Type

■ 목적 : Application의 Loading Type을 설정하기 위해 사용.

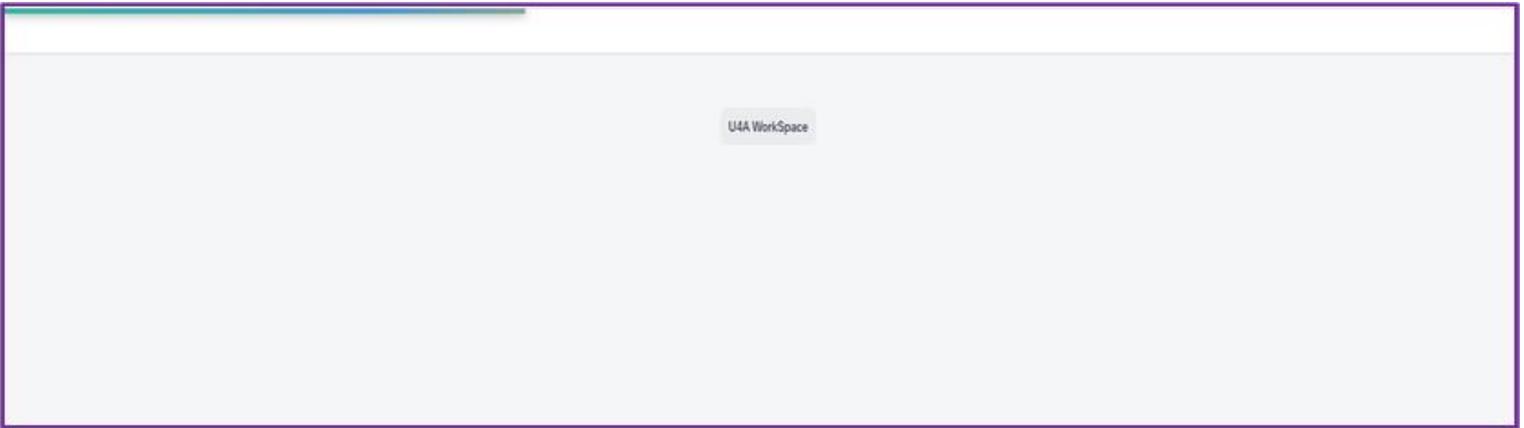
1. 해당 Application에서 Loading Type을 설정하는 것이 가능하며 총 2가지.

Properties	
Web Application ID	YU4A_WS_EDU03
Web Application Name	U4A_WS_교육용03
Web Application Version	000001
UI Theme	sap_horizon
CSS Link Add	
JS Link Add	
Init not Loading Waiting	<input type="checkbox"/>
Request/Task	
Web Security Settings	
Language Key	E
Code Page	utf-8 
Dev. Package	\$TMP
Assigned Class Object ID	YCL_U4A_APP_U4A_WS_EDU03
Program ID in Requests and Tasks	R3TR
Object Type	UXSD
Authorization Group	
Whether to use Router	<input type="checkbox"/>
Whether to use Skeleton Screen	<input type="checkbox"/>
Whether to use Mobile Zoom	<input type="checkbox"/>
Wait Type	
Touch style	Circle 
Shortcut Icon URL	

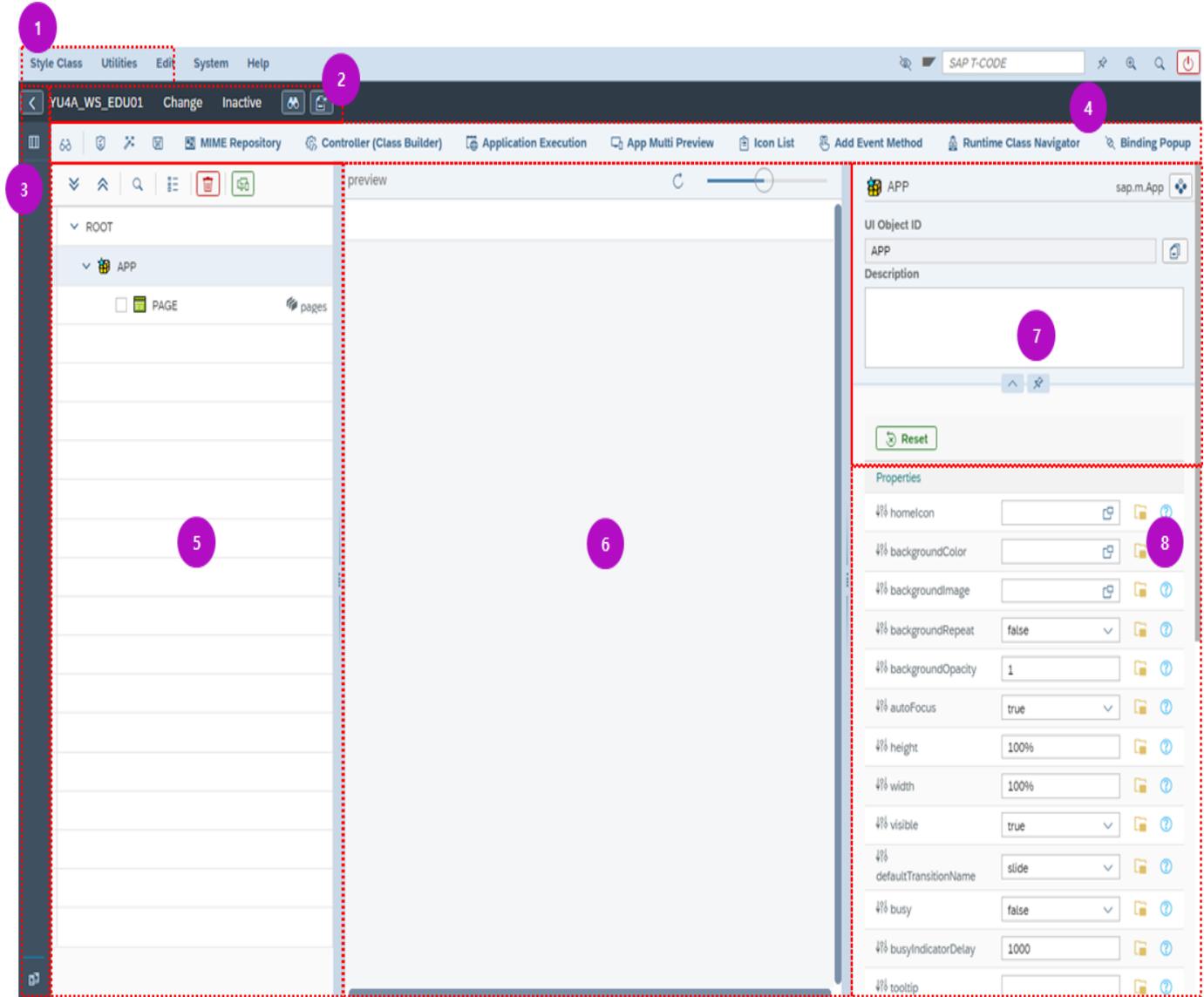
2. [UI5]-Busy_Indicator Type : Loading 화면으로 화면 중앙에 Loading Image 생김.



3. [UI5]-Loading_bar Type : Loading화면으로 화면 최상단에 색이 있는 Bar가 생김.



4.17 U4A Workbench UX Design 기능 영역 개요



Sectors	Means
1. 추가 기능 Menu	<p>App. 개발 시 경우에 따라 필요한 확장 기능이 제공된 Menu 영역이며 다음과 같은 기능이 포함되어 있음.</p> <ul style="list-style-type: none"> Style Class Theme-designer(추후 제공예정) Font Style Wizard UI5 Predefined CSS

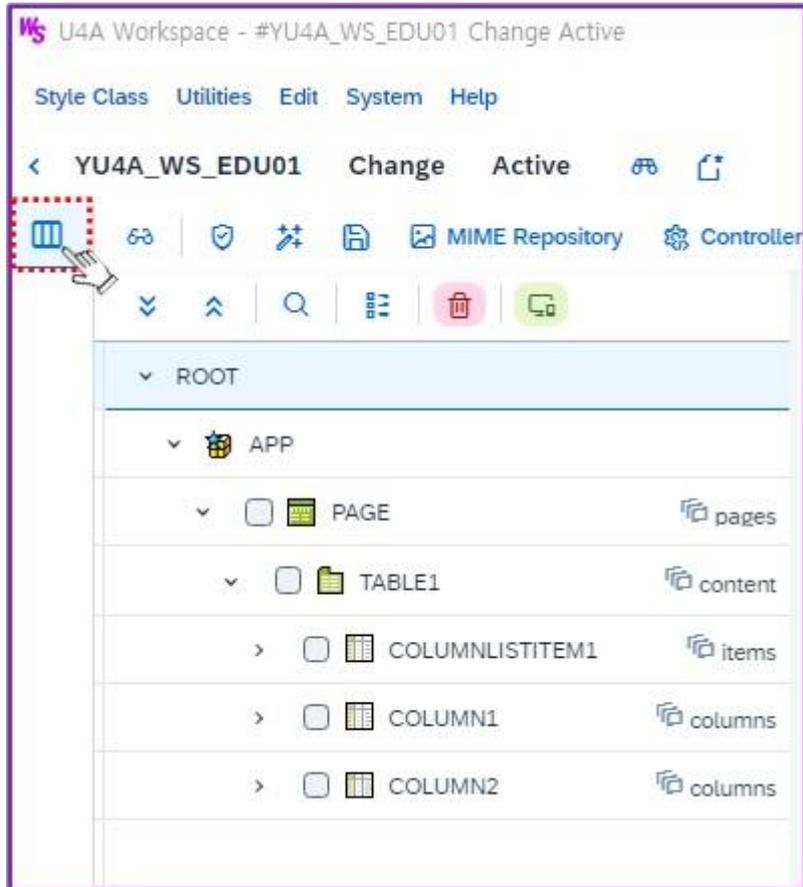
	<ul style="list-style-type: none"> • Utilities <ul style="list-style-type: none"> Select Browser Type OTR Manager Video Record • Edit <ul style="list-style-type: none"> CSS Editor Javascript Editor HTML Editor Error Page Editor Skeleton Scr Setting
<p>2. 보조 Toolbar 영역</p>	<p>상태 영역은 다음과 같이 3가지 정보를 조합하여 보여 줌.</p> <ul style="list-style-type: none"> . 현재 작업중인 App. ID . 작업상태 (Change or Display) . App. 상태 (Inactive or Active) <div style="display: flex; align-items: center; margin-bottom: 10px;">  <p>Event 사용처, Model Biding, CSS Style Class 사용처 찾기</p> </div> <div style="display: flex; align-items: center;">  <p>U4A WorkSpace 새 창 열기</p> </div>
<p>3. 사이드 Toolbar 영역</p>	<div style="display: flex; align-items: center; margin-bottom: 10px;">  <p>Application 종료</p> </div> <div style="display: flex; align-items: center; margin-bottom: 10px;">  <p>⑤(UX Design 영역), ⑥(미리보기 영역), ⑦+⑧(선택 UI Object영역 + 속성) 순서 변경</p> </div> <div style="display: flex; align-items: center;">  <p>WorkSpace Version, Client, Host, System, User, Language 정보</p> </div>
<p>4. 메인 Toolbar 영역</p>	<p>U4A IDE Design Toolkit에서 자주 사용하는 기능에 대한 영역임.</p>
<p>5. UX Design 영역</p>	<p>App. 화면 구성에 대한 Hierarchy 형태로 UI 정보를 관리 하는 영역임.</p>

	 UX Design 영역 => UI 옆 체크된 체크박스 모두 해제  UX Design 영역 => UI 옆 체크된 체크박스 UI 모두 삭제
<p>6. 미리보기 영역</p>	<p>UX Design 영역에서 UI 구성 시 위지위그(WYSIWYG) 개념으로 실시간으로 App. 화면을 보여줌.</p>
<p>선택 UI Object 영역</p>	<p>UX Design 영역에서 UI 선택 시 대상 Object의 개요를 관리 또는 추가 정보 조회를 할 수 있음.</p>  UI Samples 실행  UI Object ID 복사
<p>선택 UI Object 속성</p>	<p>UX Design 영역에서 UI 선택 시 대상 Object의 Property, Event, Aggregation 등을 관리 하는 영역임.</p>

4.18 UI 디자인 영역 개인화

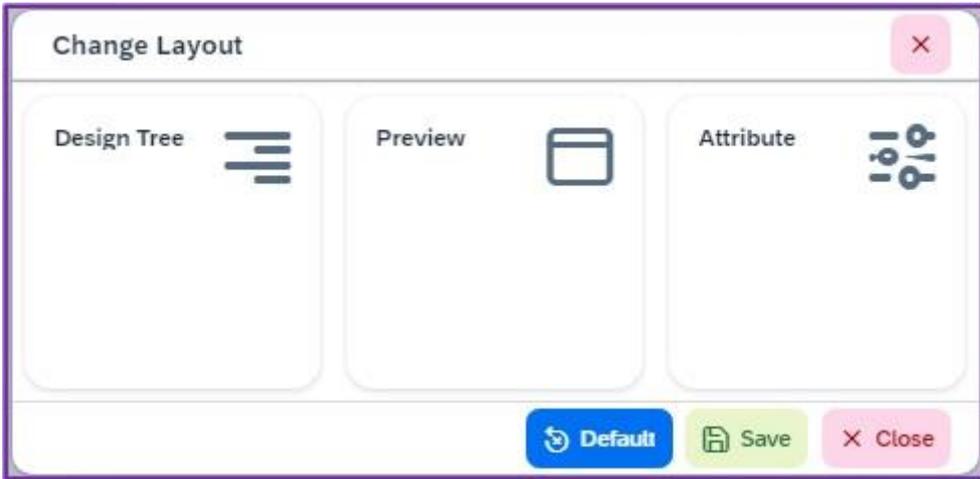
■ 목적 : UI 디자인 영역(UI Design Area, Preview, Attribute) 레이아웃 개인화.

1. WorkBench 좌측 상단에서 "Split Position Change" Button Click.



2. UI 디자인 영역 변경 가능한 Pop-Up Open.

※ Default Layout : Design Tree - Preview - Attribute 순서.



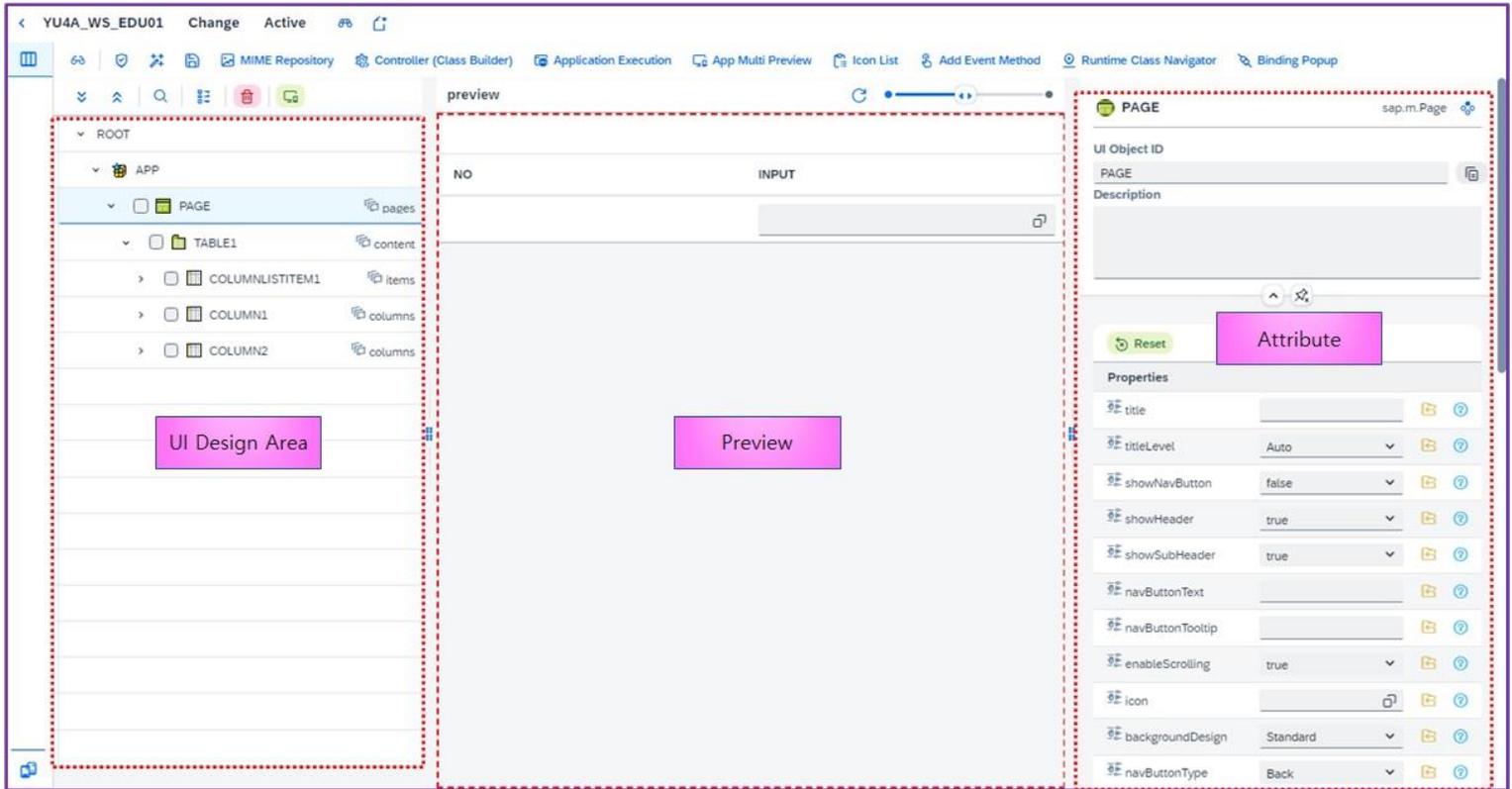
1) Default : 초기값 (Design Tree - Preview - Attribute) 순서로 원위치

2) Save : 변경한 Layout 순서 저장

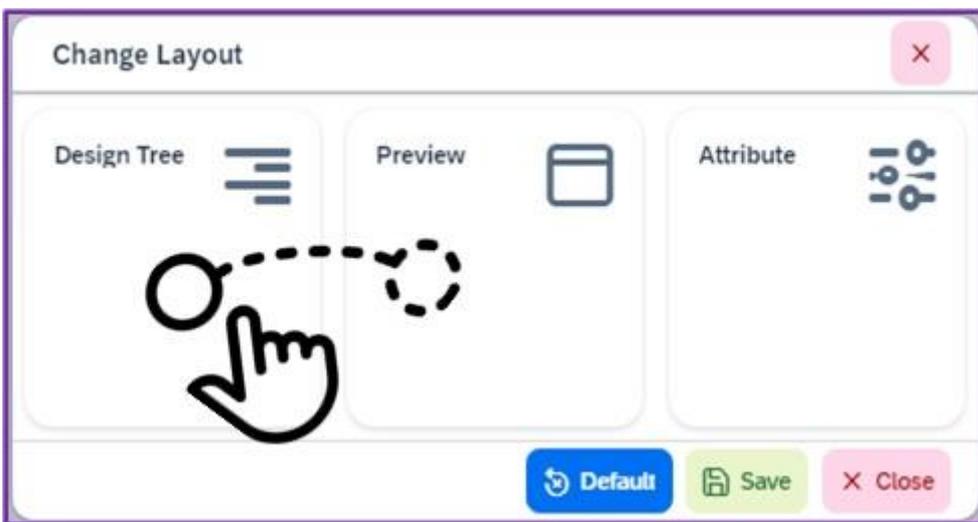
3) Close : Change Layout Pop-Up 창 종료

3. UI 디자인 영역 변경 방법.

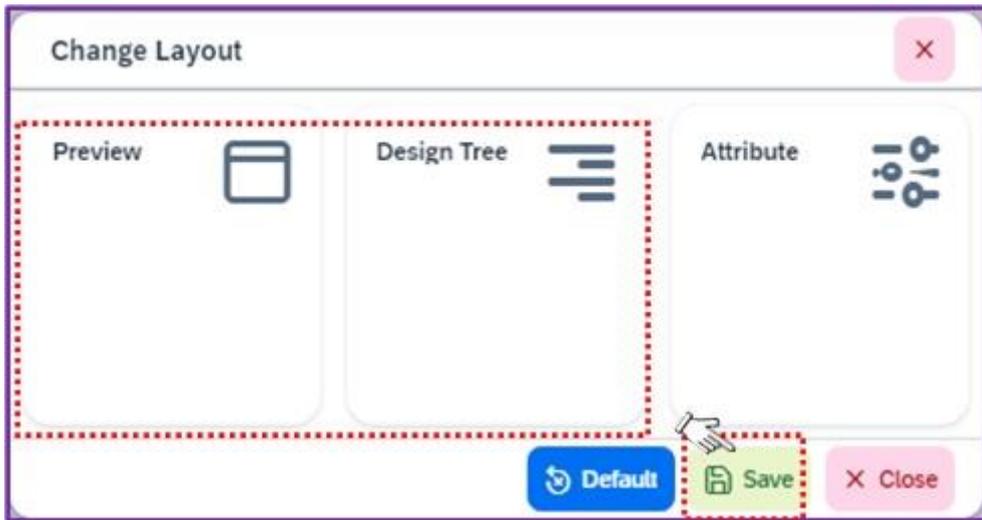
※ 현재 Application Layout (Design Tree(= UI Design Area) - Preview - Attribute)



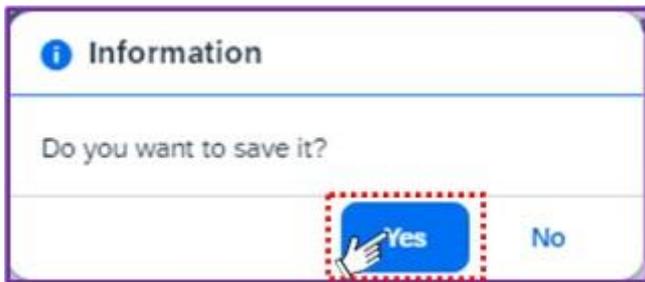
① Design Tree, Preview, Attribute 타일 Drag & Drop (본 단원에서는 Design Tree <-> Preview 위치만 변경)



② Design Tree 와 Preview 타일 위치 변경 확인 후 "Save" 버튼 클릭.

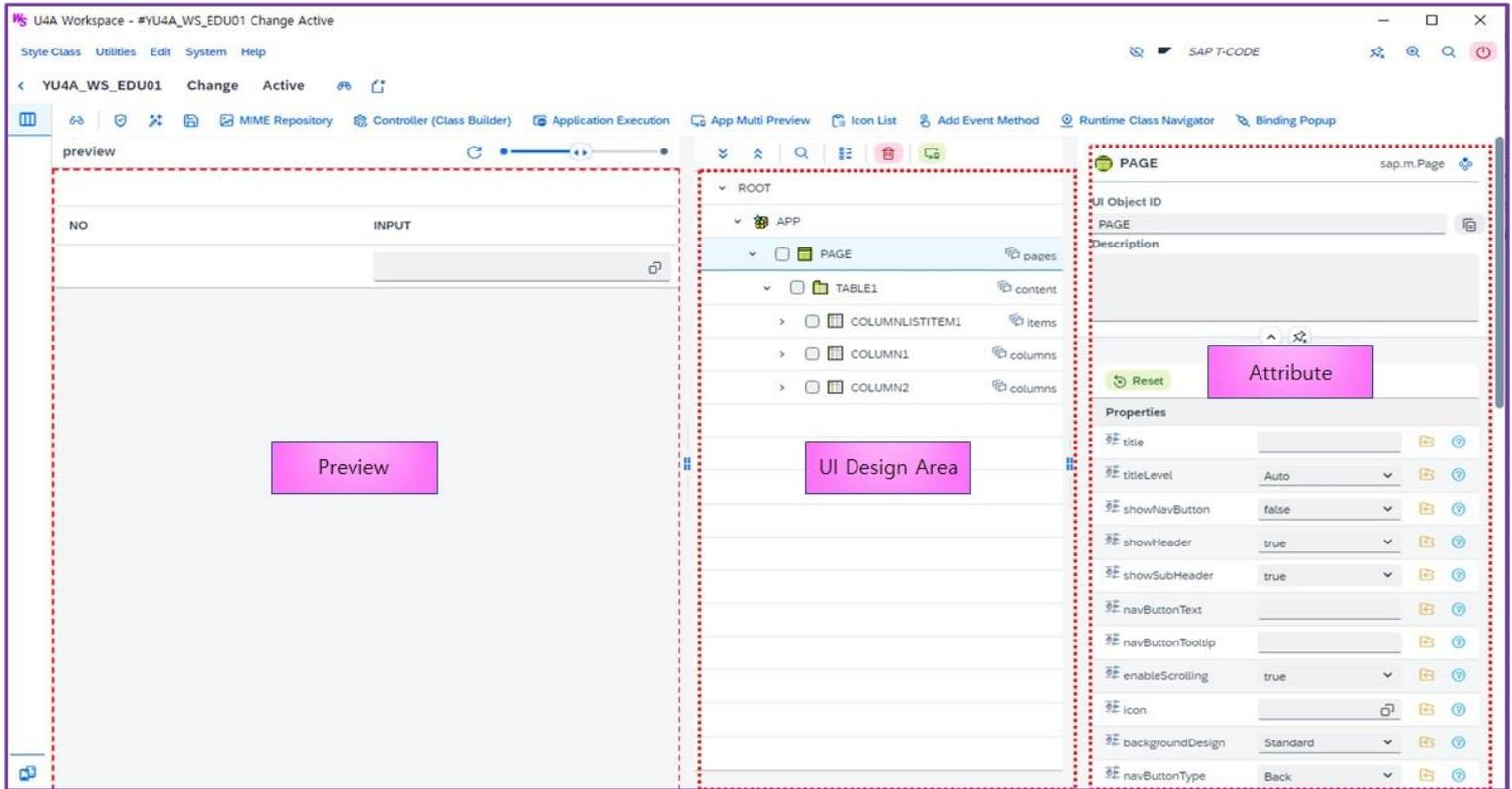


③ 확인 Popup에서 "Yes" Button Click.



4. 변경 된 UI 디자인 영역 변경 확인.

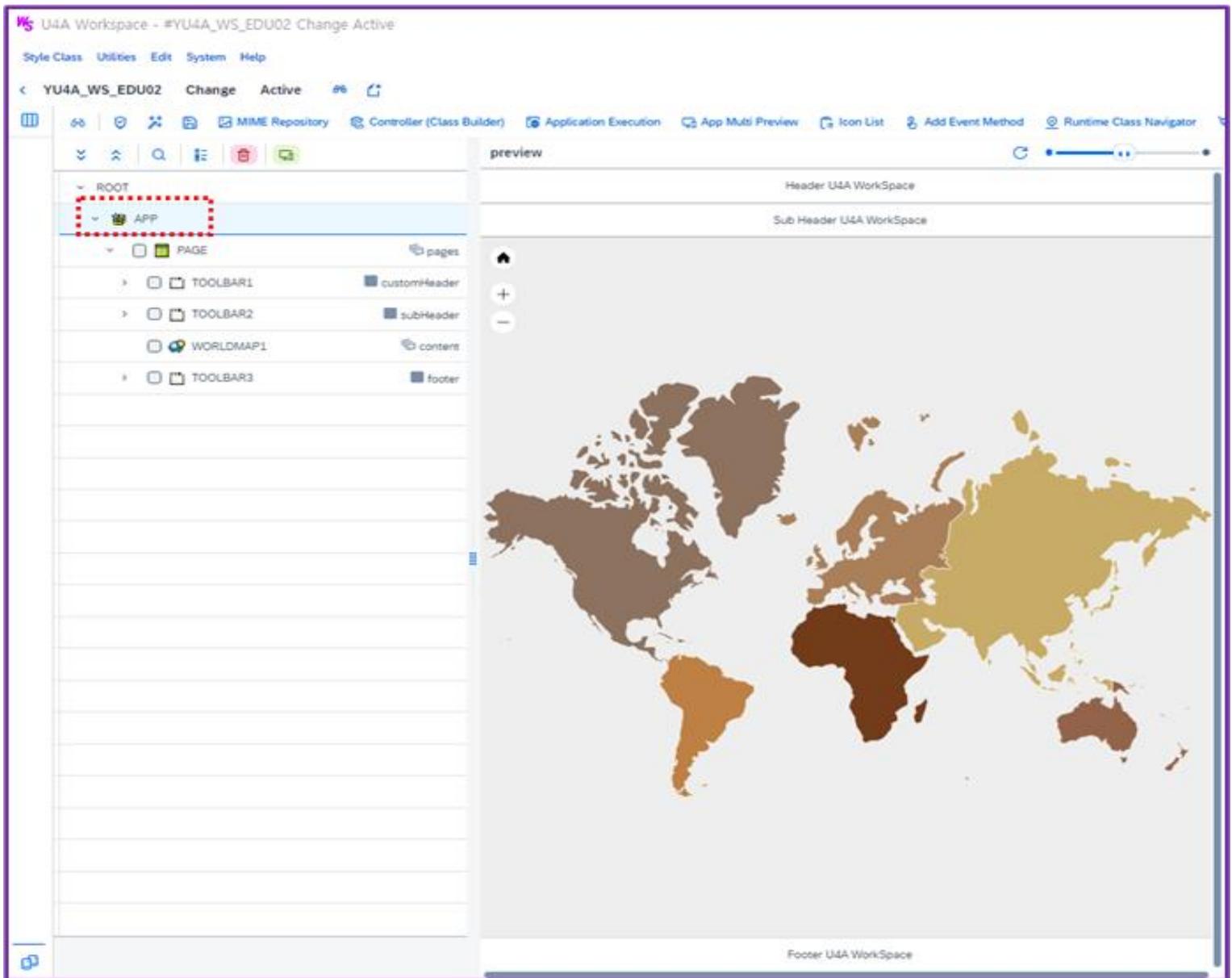
U4A IDE HELP



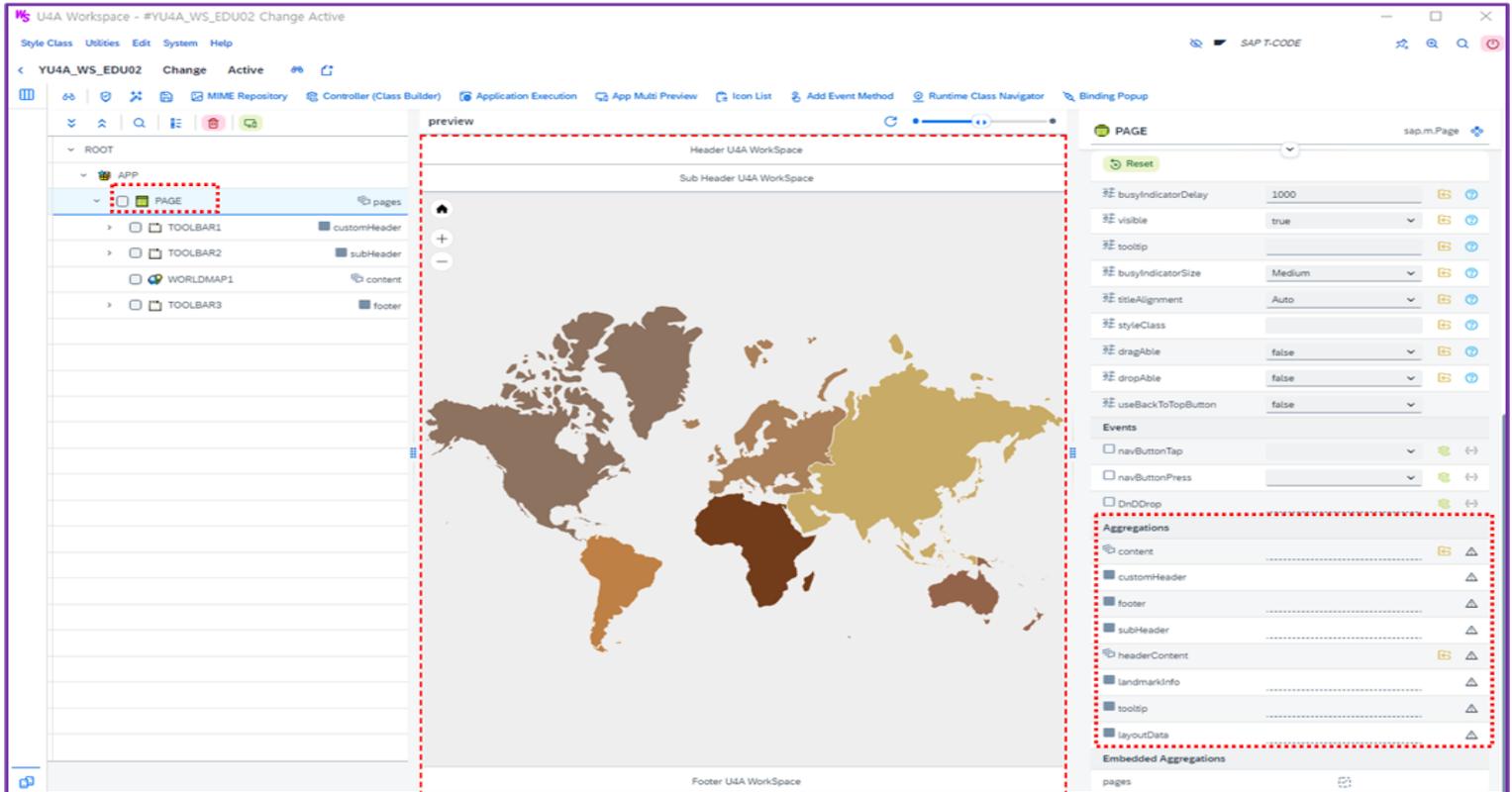
4.19 UX Design Area

■ 목적 : Application의 UX Design 영역에 대한 정의.

1. APP은 최상위 UI를 의미하며 화면을 그릴 수 있는 Page를 Aggregation으로 가짐.



2. Page는 ABAP Program에서 Screen과 같은 역할을 하며 크게 Header, Body, Footer 3개 영역으로 구분.



※ Page는 아래 항목들을 Aggregation으로 가지며 이는 해당 항목으로 UI를 그릴 수 있음을 의미.

▣ Aggregations : Table Type의 Data를 Binding 시킬 수 있는 영역으로 ++ 표시가 되어 있는 항목만 Binding 가능.

- ① HeaderContent : 화면 최상단 영역을 의미하며 Table Type Binding이 가능.
- ② customHeader : Header와 같은 영역이며 Header에서 기능이 충분하지 않은 경우 사용.
- ③ subHeader : Header 바로 아래 영역을 의미 .
- ④ Content : Header와 Footer의 중간인 Body 영역에 해당하는 부분으로 Table Type Binding이 가능 .
- ⑤ Footer : 화면 최하단 영역을 의미.

4.20 Attributes

■ **목적** : UI가 가지고 있는 Property, Aggregation, Events가 정의되어 있는 영역.

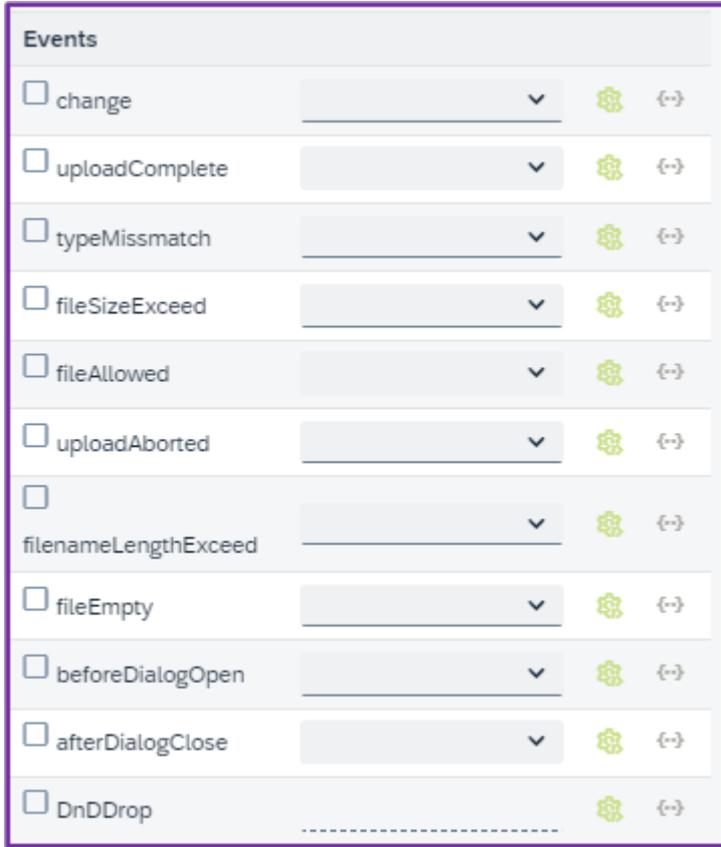
1. UI가 가질 수 있는 다양한 Option들을 나열한 영역.

※ Properties를 통해 해당 UI에 다양한 Option을 주거나 값을 Binding, Property에 대한 Help Document를 볼 수 있음.

Properties			
 backgroundDesign	Translucent	▼	 
 fixedLayout	true	▼	 
 showOverlay	false	▼	 
 inset	false	▼	 
 headerText			 
 headerDesign	Standard	▼	 
 footerText			 
 mode	None	▼	 
 width	100%		 
 includeItemInSelection	false	▼	 
 showUnread	false	▼	 
 noDataText			 
 showNoData	true	▼	 
 enableBusyIndicator	true	▼	 
 modeAnimationOn	true	▼	 
 showSeparators	All	▼	 
 swipeDirection	Both	▼	 
 growing	false	▼	 
 growingThreshold	20		 

2. UI가 특정 동작을 취했을 때 동작하는 Event를 설정하는 영역.

※ Events에서 Method를 설정하거나 Java Script를 통해 Event를 설정함.



3. Table Type의 Data를 Binding 할 수 있는 영역.



표시가 되어 있는 항목만 Binding을 할 수 있음

Aggregations		
 columns	 
 items		 
 swipeContent	
 headerToolbar	
 infoToolbar	
 tooltip		
 layoutData	
 contextMenu	
 noData		

4.21 Attribute Helper

■ **목적** : Attribute에 대한 상세한 설명이 필요할 때 Document 호출하기 위한 목적.

1. 특정 Property의 이름, Type 등의 상세 정보가 필요할 때 해당 Property의 Attribute Help Button Click.

Properties			
 backgroundDesign	Translucent	▼	
 fixedLayout	true	▼	
 showOverlay	false	▼	
 inset	false	▼	
 headerText			
 headerDesign	Standard	▼	
 footerText			
 mode	None	▼	
 width	100%		
 includeItemInSelection	false	▼	
 showUnread	false	▼	
 noDataText			
 showNoData	true	▼	
 enableBusyIndicator	true	▼	
 modeAnimationOn	true	▼	
 showSeparators	All	▼	
 swipeDirection	Both	▼	
 growing	false	▼	
 growingThreshold	20		

2. Open UI5에서 제공하고 있는 Property에 대한 Help Document를 똑같이 제공 .

선택한 Property에 대한 설명은 하늘색으로 표시.

The screenshot shows a 'Property Help' window with three entries. The first two are 'alternateRowColors' and 'autoPopinMode', both of type 'boolean' and value 'false'. The third is 'backgroundDesign', of type 'sap.m.BackgroundDesign' and value 'Translucent'. The 'backgroundDesign' entry is highlighted in light blue. Each entry includes a 'Description' section with detailed text and a 'Visibility' section.

Property Name	Type	Value
alternateRowColors	boolean	false
autoPopinMode	boolean	false
backgroundDesign	sap.m.BackgroundDesign	Translucent

alternateRowColors **boolean** **false**

Description:

Enables alternating table row colors. **Note:** This property can only be used with the Belize and Belize Deep themes. Alternate row coloring is not available for the High Contrast Black/White themes.

Since: 1.52.

Visibility : public

autoPopinMode **boolean** **false**

Description:

Enables the auto pop-in behavior for the table control.

If this property is set to true, the table control overwrites the demandPopin and the minScreenWidth properties of the sap.m.Column control. The pop-in behavior depends on the importance property of the sap.m.Column control. Columns configured with this property are moved to the pop-in area in the following order:

- With importance High: moved last
- With importance Medium or None: moved second
- With importance Low: moved first

Note: If this property is changed from true to false, the application must reconfigure the demandPopin and minScreenWidth properties of the sap.m.Column control by itself. There is no automatic mechanism that restores the old values if autoPopinMode was set from false to true before.

Since: 1.76.

Visibility : public

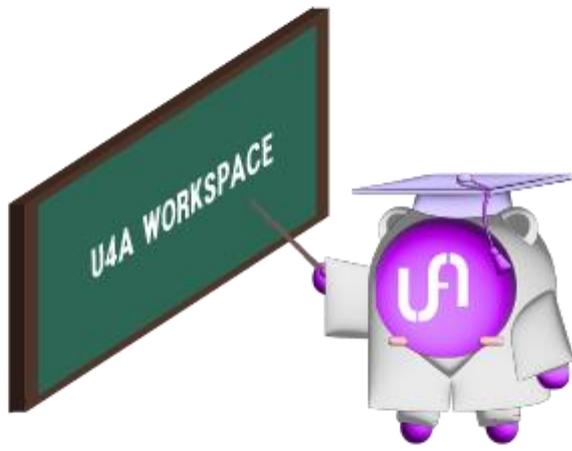
backgroundDesign **sap.m.BackgroundDesign** **Translucent**

Description:

Sets the background style of the table. Depending on the theme, you can change the state of the background from Solid to Translucent or to Transparent.

Visibility : public

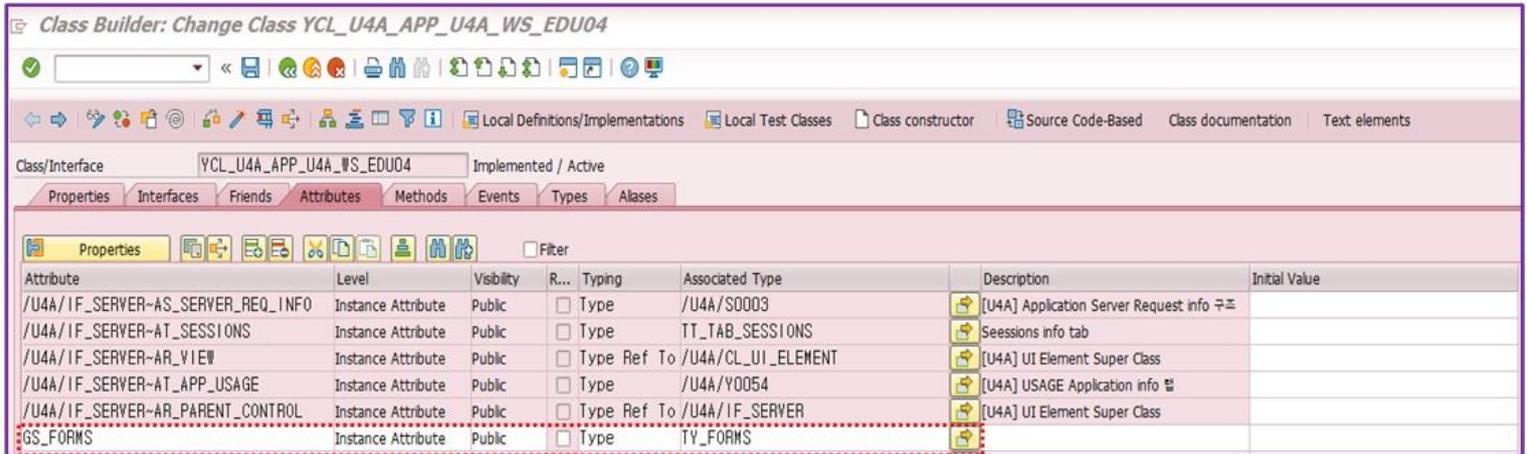
4.22 UI Template Wizard



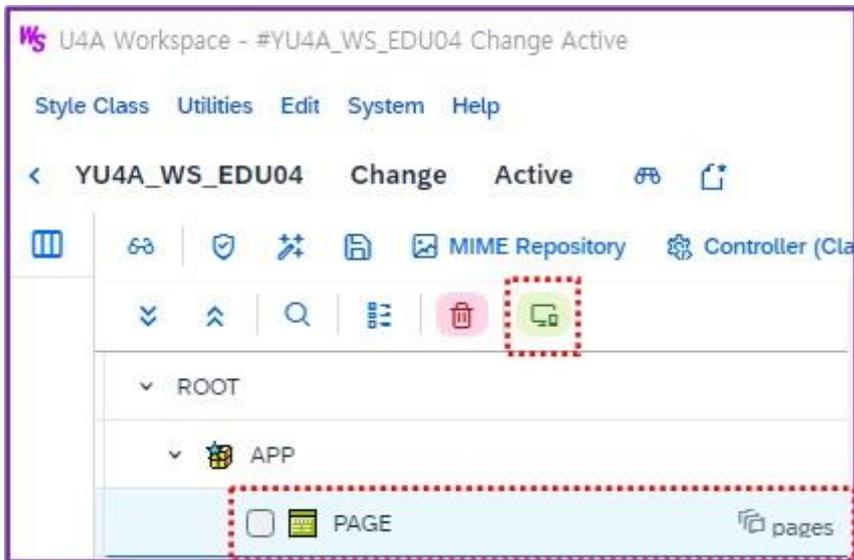
4.22.1 Forms UI Create

▣ 목적 : 자주 사용하는 Form UI를 Sample로 만들어 쉽게 생성하기 위한 목적.

1. Form의 Field 값을 Binding 시킬 Public Structure를 생성.

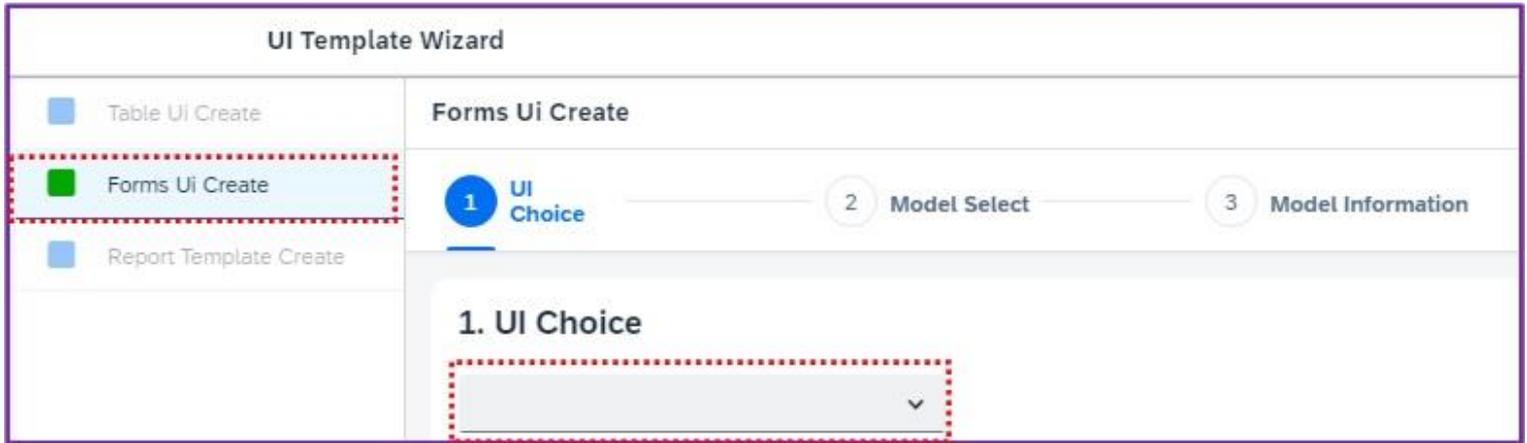


2. Form UI를 삽입 할 위치를 선택한 후 UI Template Wizard Button Click.

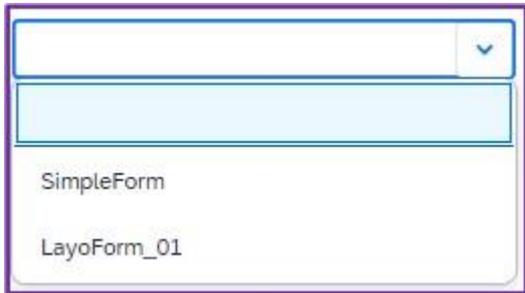


3. UI Template Wizard 화면에서 Form UI Create Tab을 선택. (녹색 Box는 Form Data를 Binding 시킬 Model이 존재함을 의미.)

※ 첫번째 단계인 UI Choice 박스 선택.

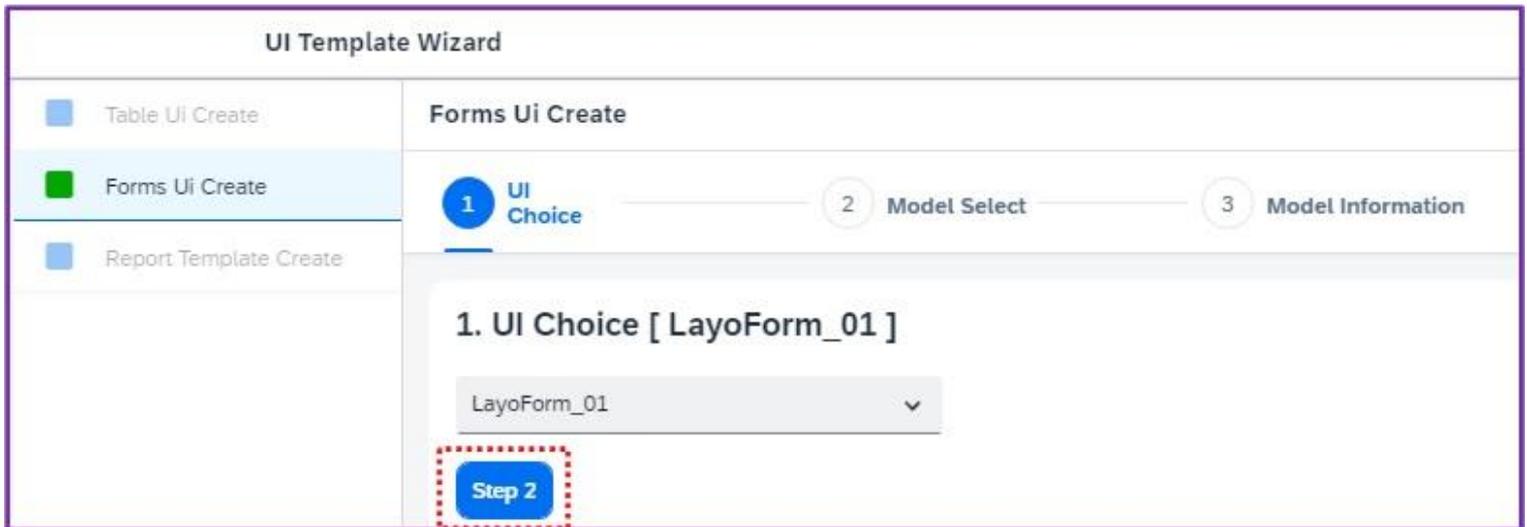


4. 생성하려는 Form UI 선택 (1. LayoForm 2. SimpleForm)

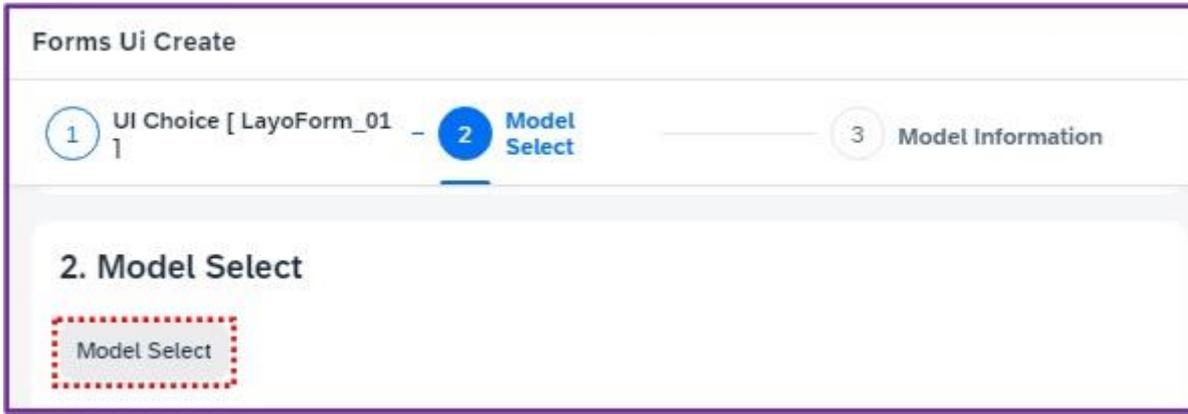


② Model Name : From UI에 Binding 시키는 Model 명 (Select Model Button Click)

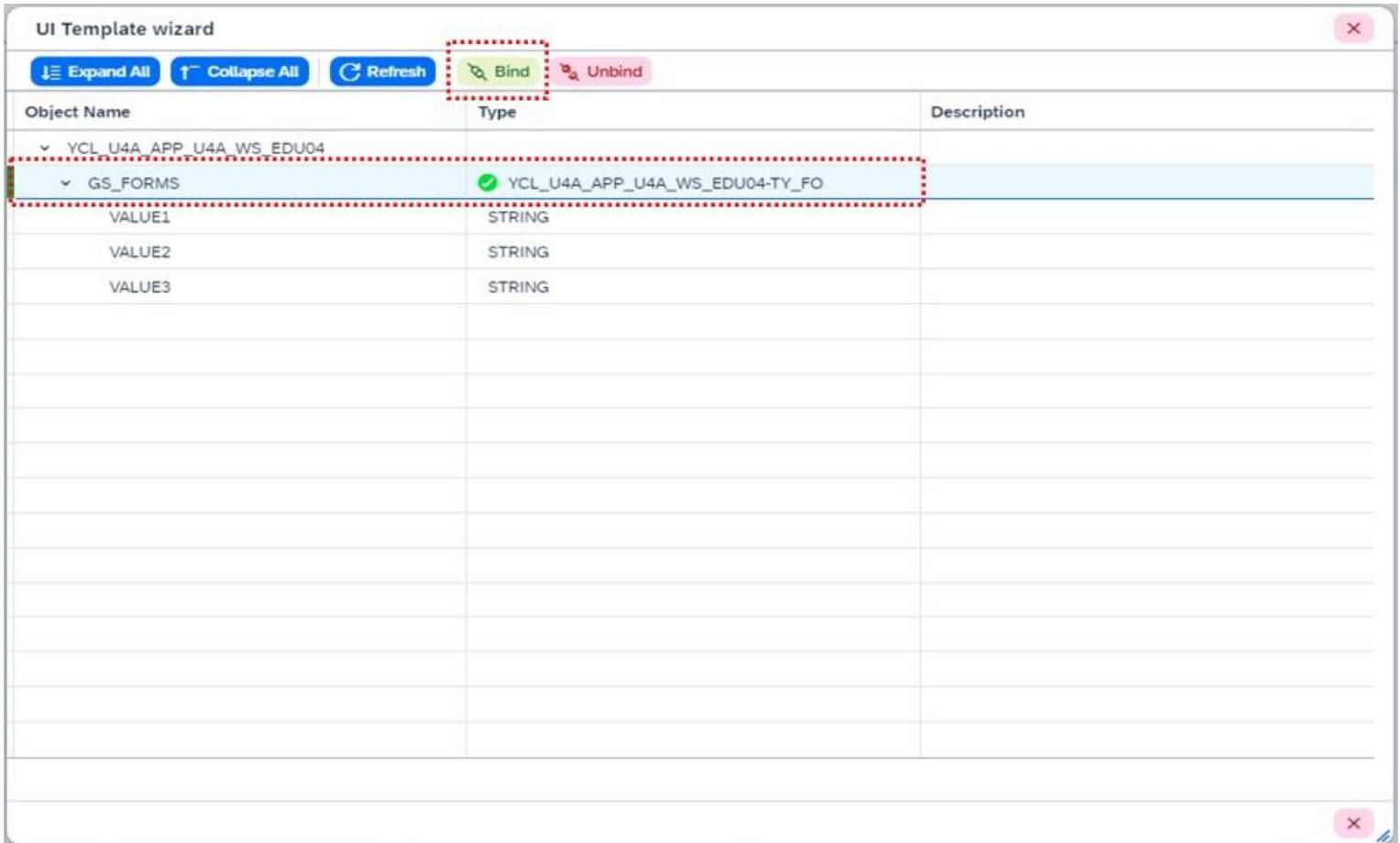
5. Form UI 선택 후 다음 Step으로 진행.



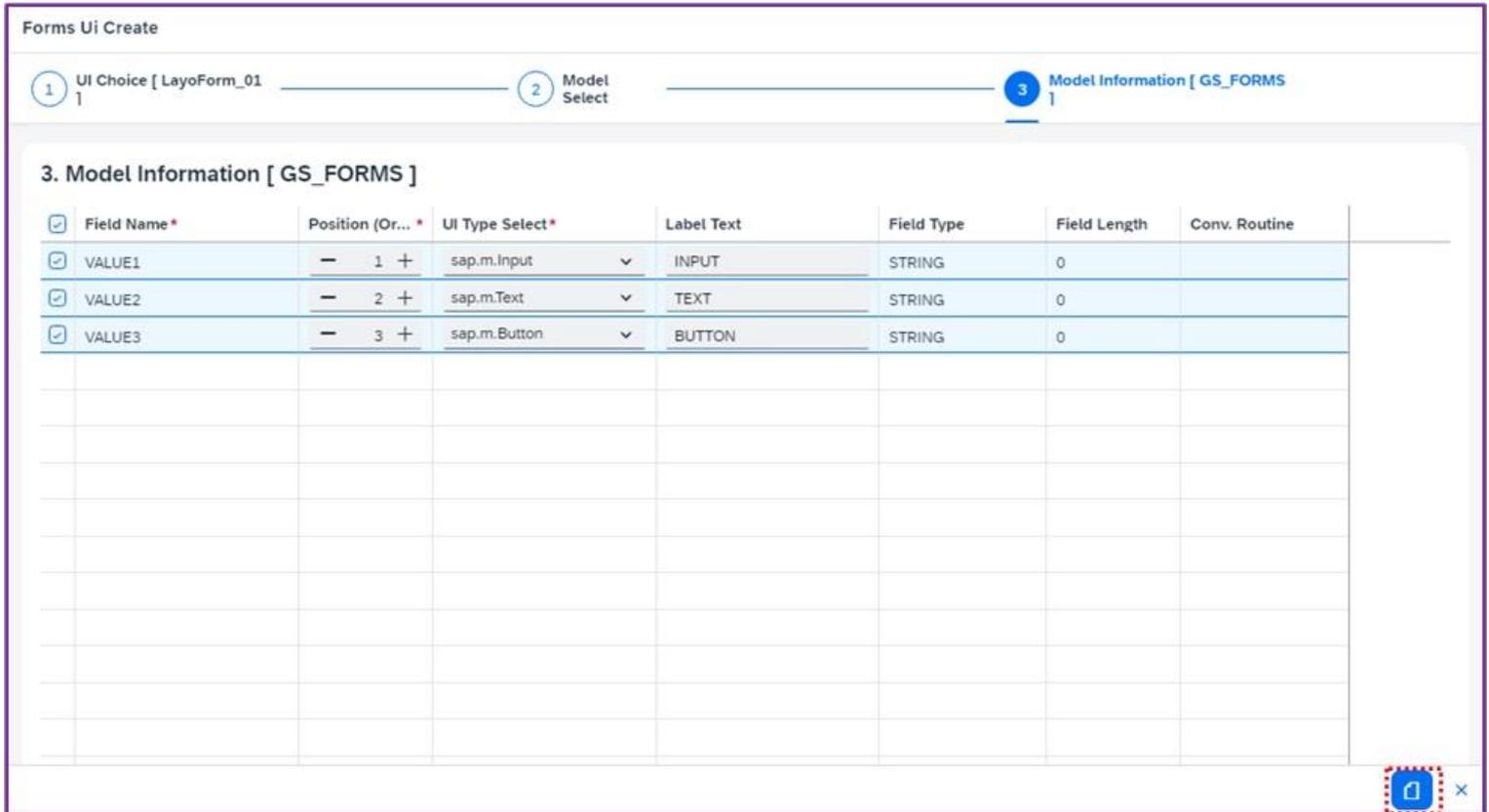
6. Form UI에 Binding 시키는 Model 명 (Model Select Button Click)



7. Binding할 Model Click 후 "Bind" Button Click.



8. Binding 되어있는 Structure의 Field 정보 및 Option 설정.



① √ : 선택하면 UI에 포함 되고 선택하지 않으면 UI에 포함되지 않음.

② Position : Form 안에서의 UI의 위치 설정.

② Field Name : Binding 되어있는 Structure Field 명.

③ UI type select : Structure Field UI Type.

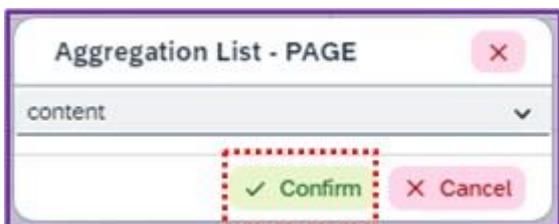
④ Label Text : Field Description.

⑤ Field type : Field Value Type.

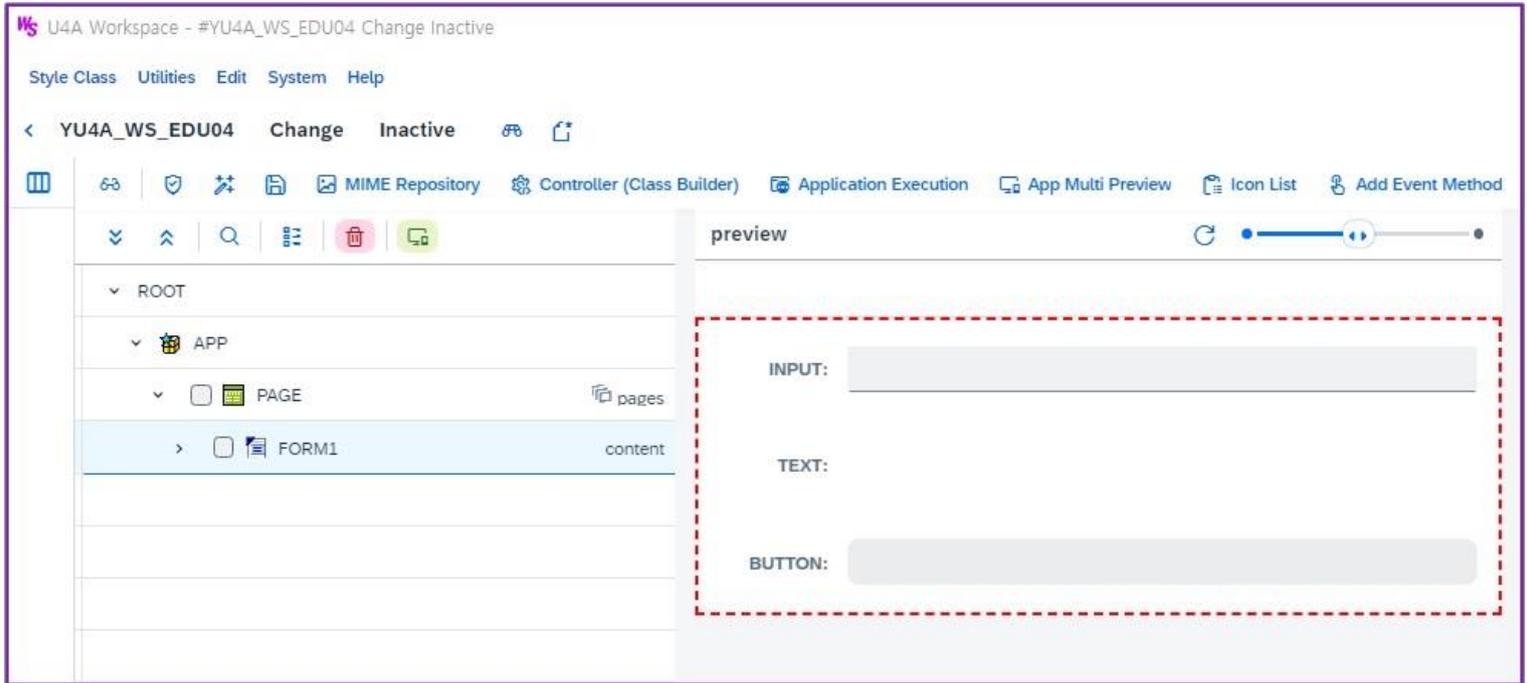
⑥ field Length : Field Value 길이.

⑦ Conversion Routine : Field에 적용 된 Conversion Routine.

9. UI가 어떤 Aggregation으로 생성 될 지 선택(선택지가 1개면 Popup은 생성 되지 않음)



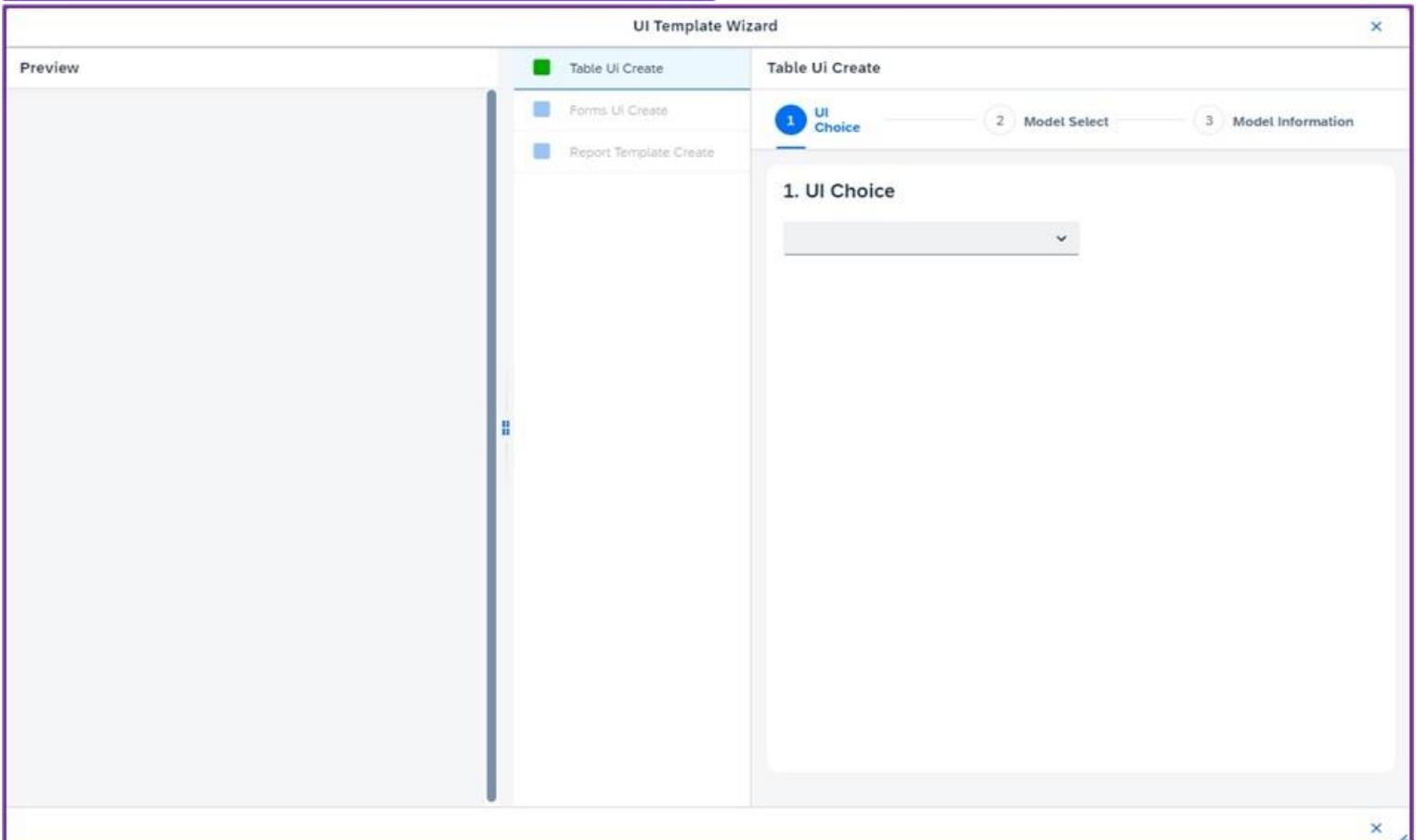
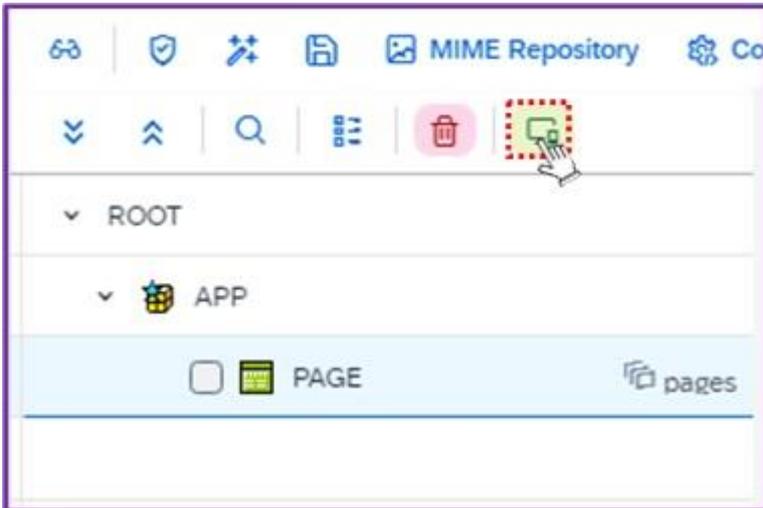
10. 해당 Application에 Form UI가 생성된 것을 확인.



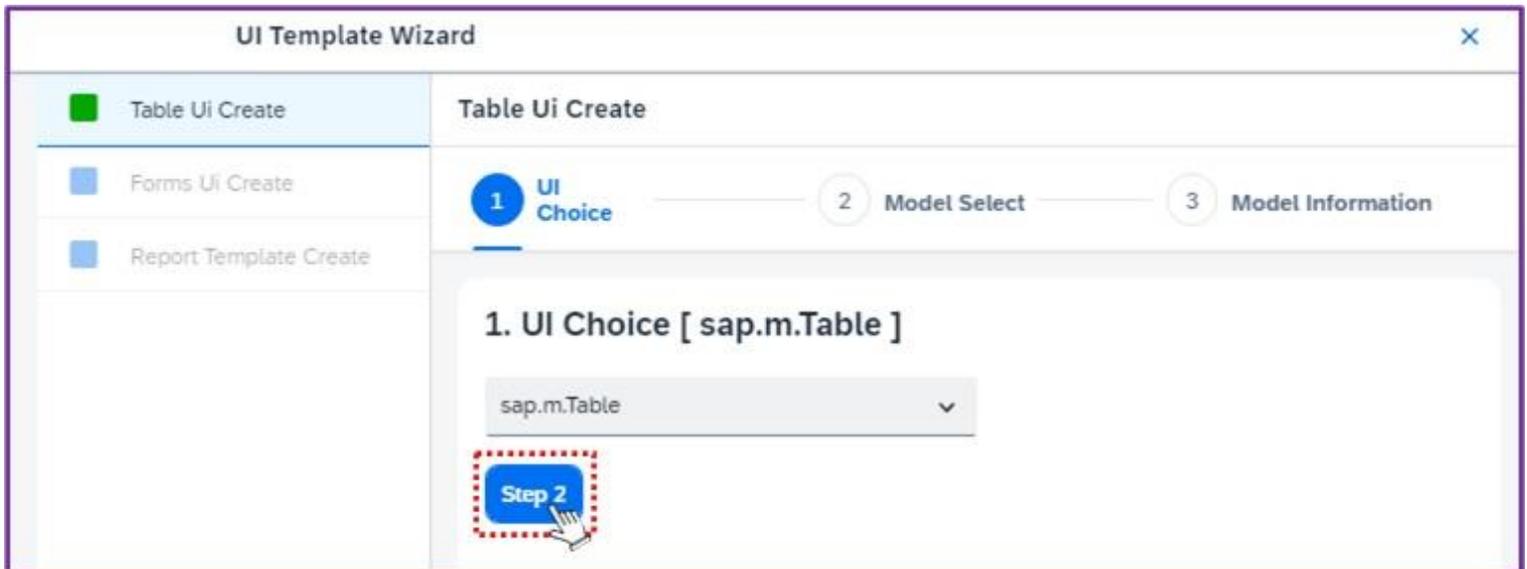
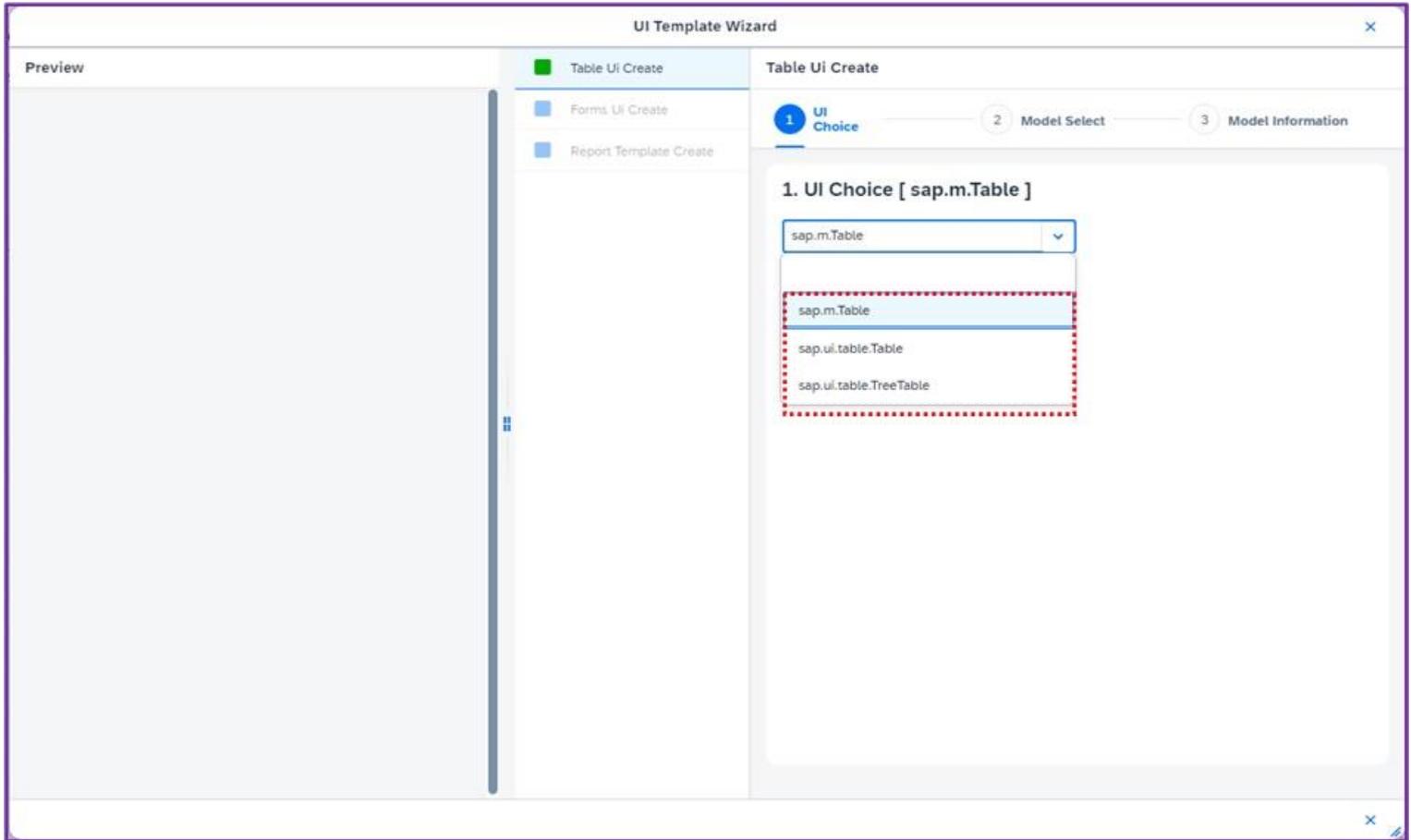
4.22.2 Table UI Create

■ 목적 : TABLE 형태의 UI Object를 편하게 생성 할 수 있는 기능이다.

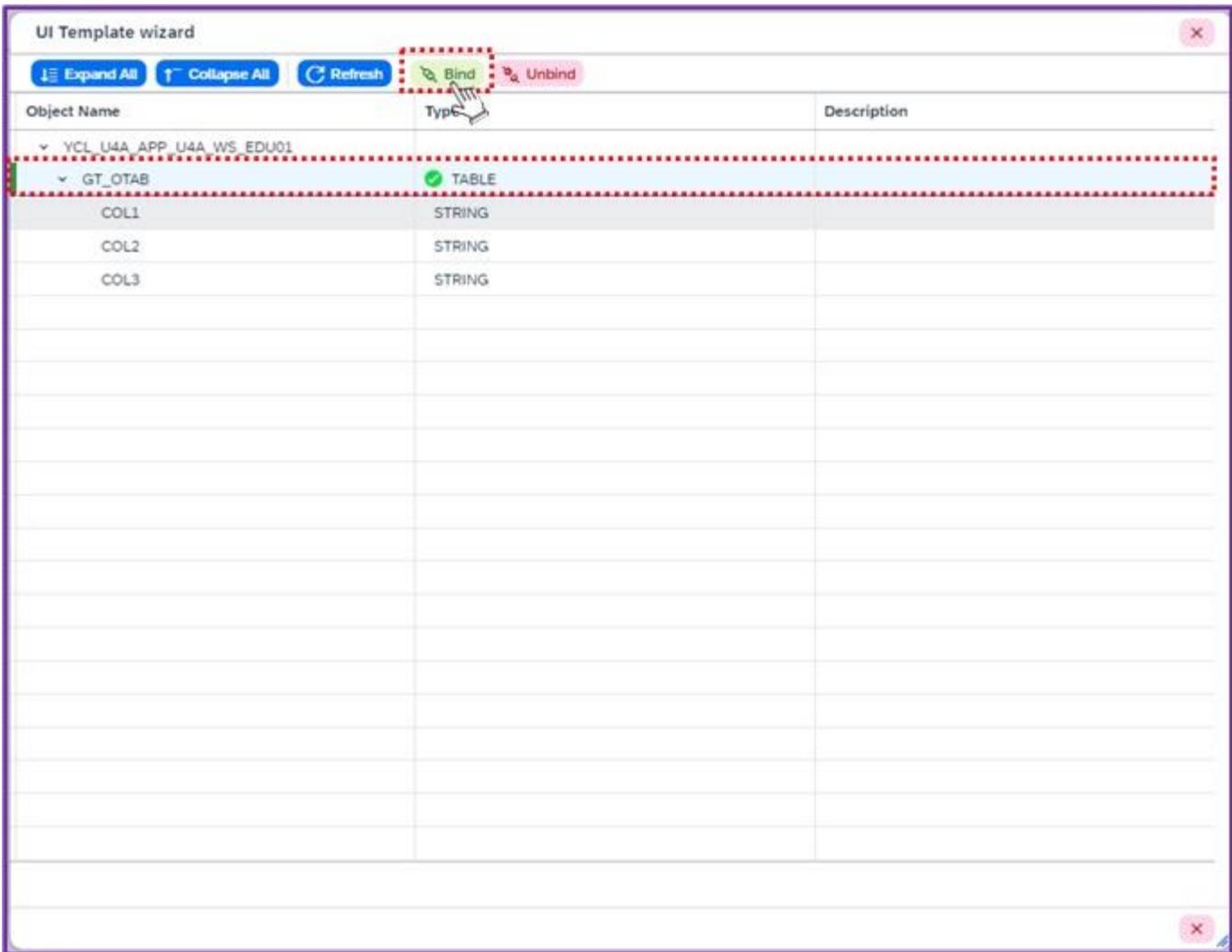
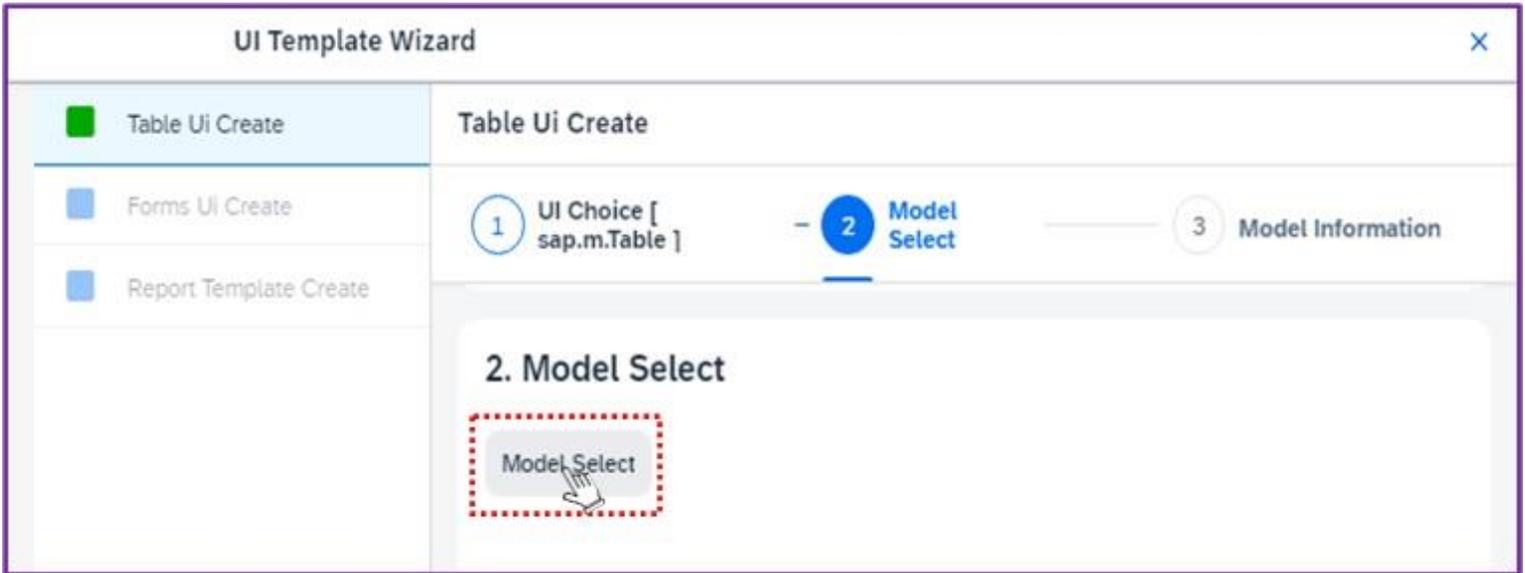
1. U4A는 application내에서 Table의 기본 형태에 대한 틀을 제공한다.



2. 원하는 Table의 UI를 선택하고 "Step 2" 버튼 클릭(m.table, ui.table, Treetable 제공)



3. "Model Select" 버튼을 클릭하여 Table의 Model을 binding 한다.



U4A IDE HELP

The screenshot displays the SAP U4A IDE interface. On the left, the project tree shows a hierarchy: ROOT > APP > PAGE > TABLE1 > COLUMNLISTITEM1 > TEXT1. The 'TEXT1' object is highlighted with a red dashed box. The central preview window shows a blank area. On the right, the properties panel for 'TEXT1' is visible, with the 'text' property highlighted by a red dashed box. The 'text' property is set to 'GT_OTAB-COL1'. Other visible properties include 'textDirection' (Inherit), 'wrapping' (true), 'textAlign' (Begin), 'width', 'maxLines', and 'busy' (false).

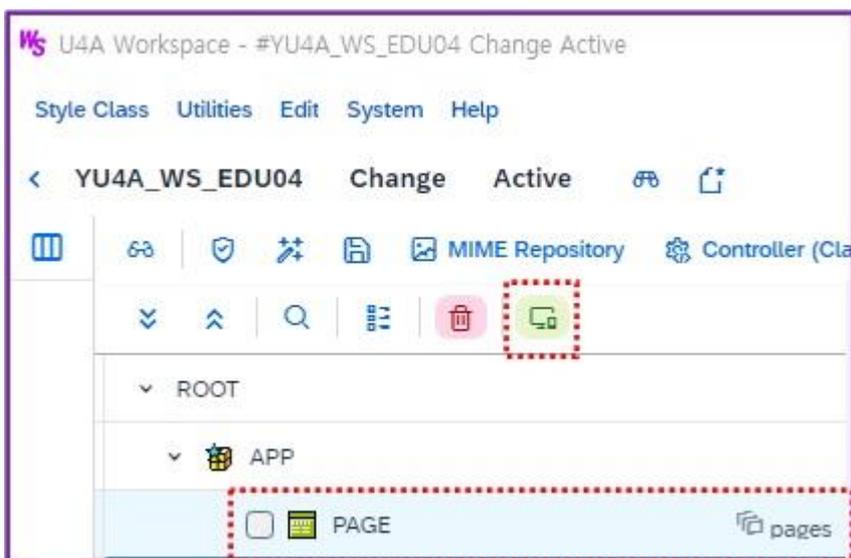
4.22.3 Report Template Create

▣ 목적 : 자주 사용하는 Form UI, TABLE UI 를 Sample로 만들어 쉽게 생성하기 위한 목적.

1. "Form"과 "Table"의 Field 값을 Binding 시킬 Public Structure 및 Table을 생성.

Attribute	Level	Visibility	R...	Typing	Associated Type	Description
/U4A/IF_SERVER~AS_SERVER_REQ_INFO	Instance Attribute	Public	<input type="checkbox"/>	Type	/U4A/S0003	[U4A] Application Server Request info 구조
/U4A/IF_SERVER~AT_SESSIONS	Instance Attribute	Public	<input type="checkbox"/>	Type	TT_TAB_SESSIONS	Seessions info tab
/U4A/IF_SERVER~AR_VIEW	Instance Attribute	Public	<input type="checkbox"/>	Type Ref To	/U4A/CL_UI_ELEMENT	[U4A] UI Element Super Class
/U4A/IF_SERVER~AT_APP_USAGE	Instance Attribute	Public	<input type="checkbox"/>	Type	/U4A/Y0054	[U4A] USAGE Application info 탭
/U4A/IF_SERVER~AR_PARENT_CONTROL	Instance Attribute	Public	<input type="checkbox"/>	Type Ref To	/U4A/IF_SERVER	[U4A] UI Element Super Class
GS_STRU	Instance Attribute	Public	<input type="checkbox"/>			
GT_OTAB	Instance Attribute	Public	<input type="checkbox"/>			
			<input type="checkbox"/>	Type		

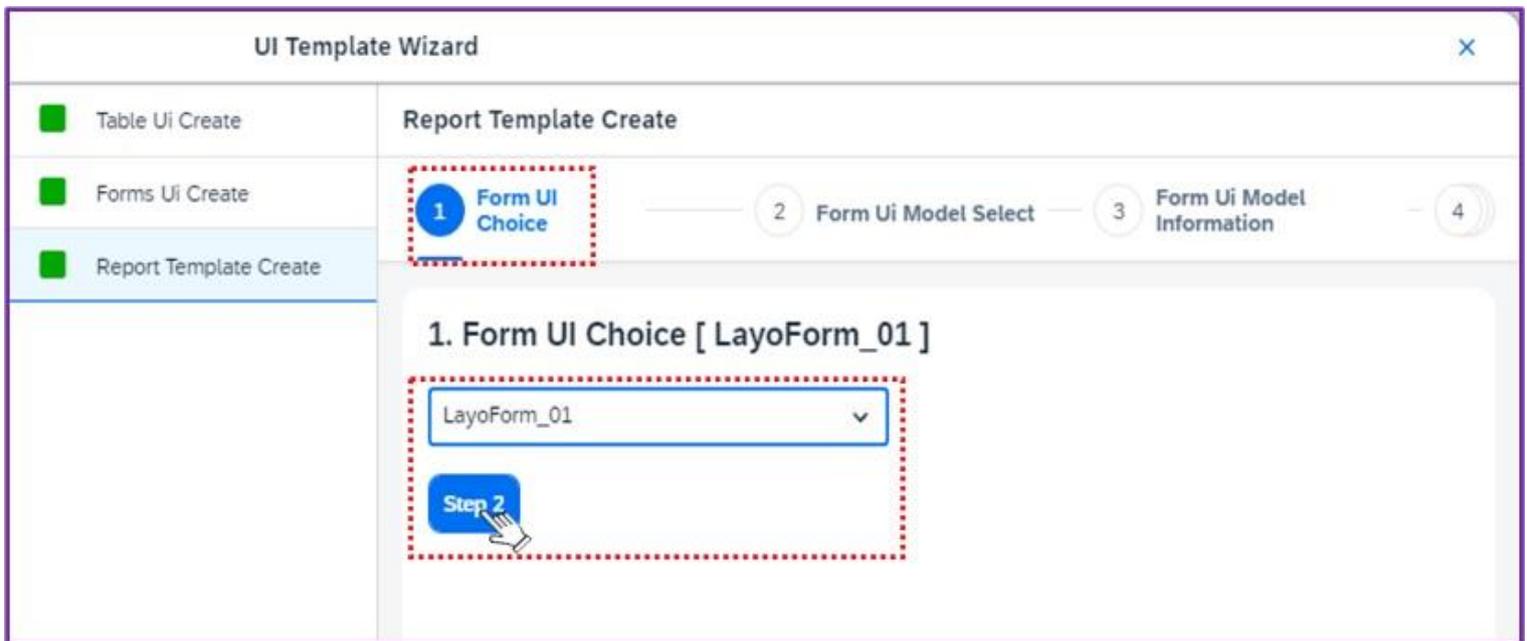
2. "Form"과 "Table" UI를 삽입 할 위치를 선택한 후 UI Template Wizard Button Click.



3. UI Template Wizard 화면에서 "Report Template Create" Tab을 선택. (녹색 Box는 Data를 Binding 시킬 Model이 존재함을 의미.)

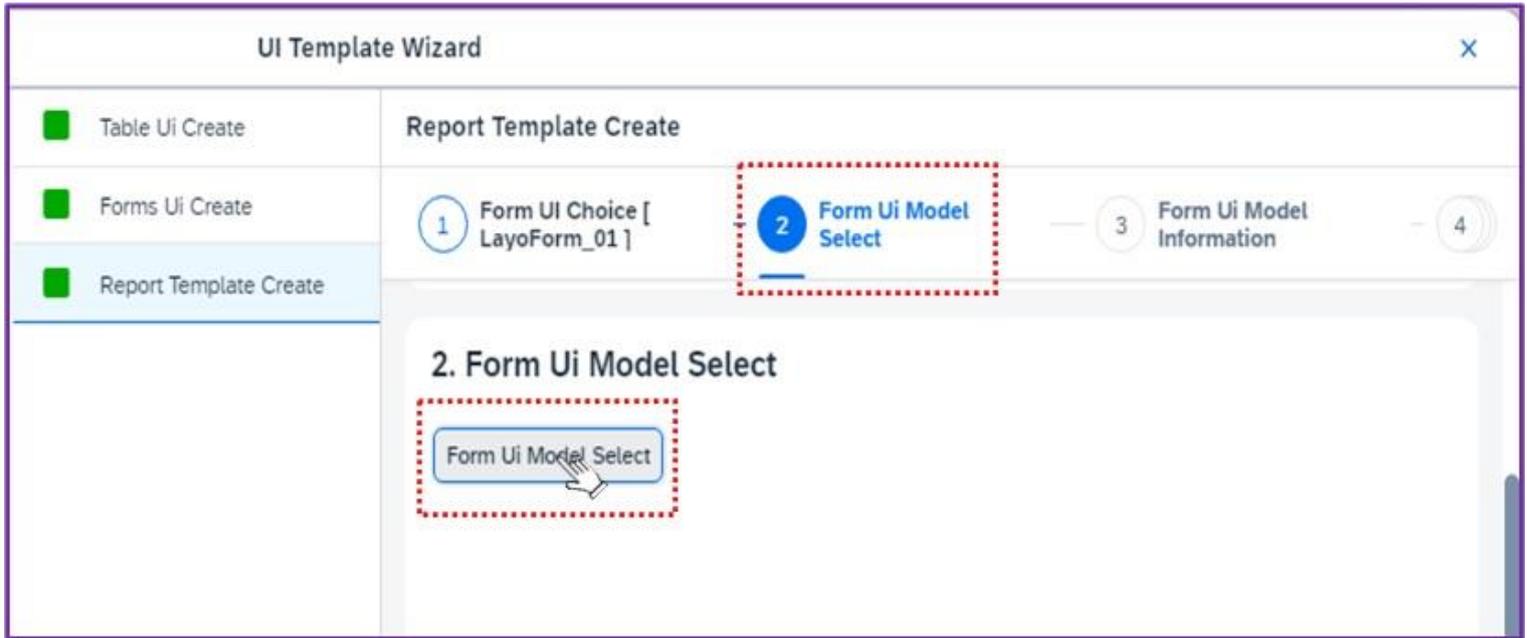
① Form UI Choice (본단원에서는 LayoForm_01로 진행)

※ 생성가능한 Form 종류 : SimpleForm, LayoForm_01

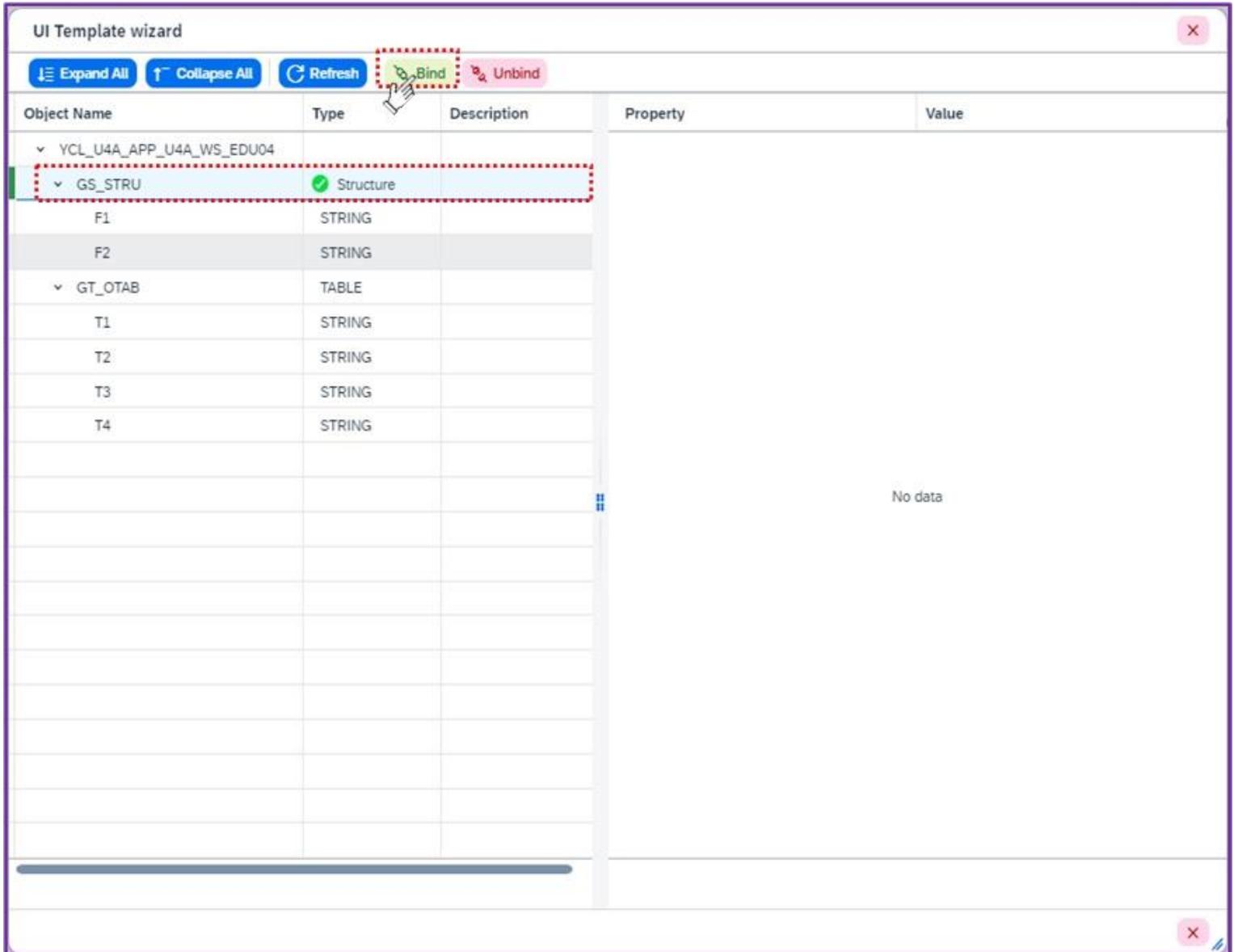


② Form UI Model Select

※ Form UI에 Binding 시키는 Model 명 (Model Select Button Click)

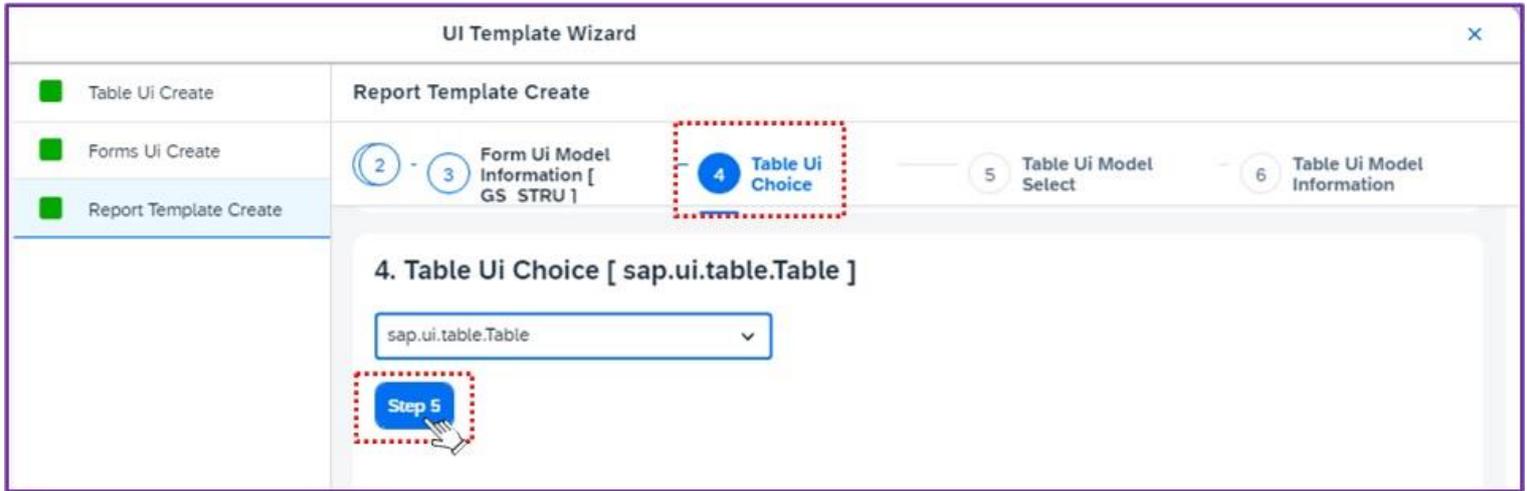


※ Binding할 Model Click 후 "Bind" Button Click.



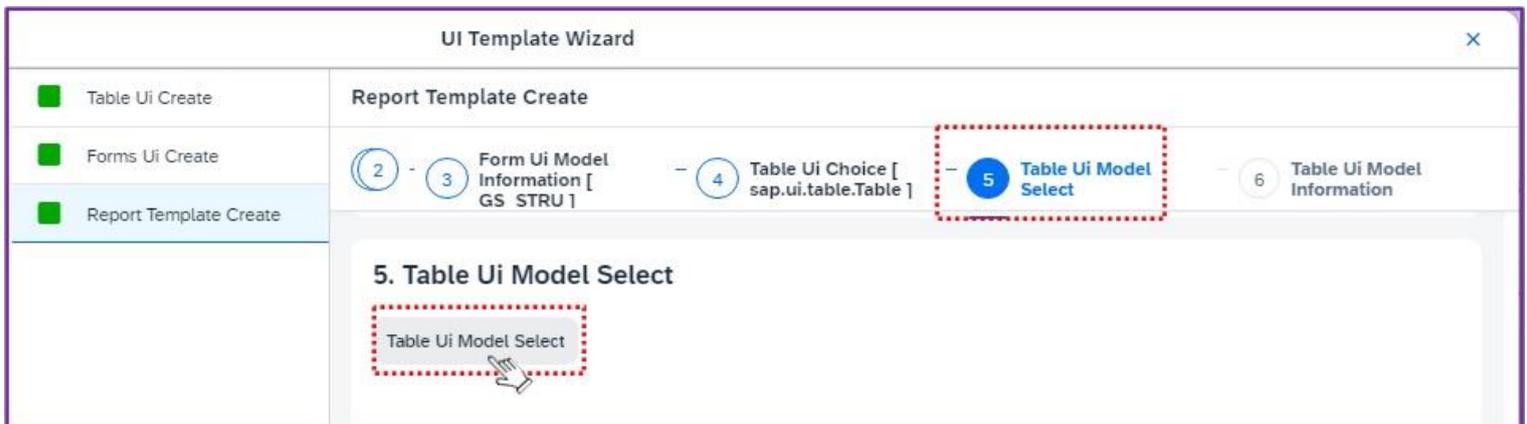
③ Form Ui Model Information (Binding 되어있는 Structure의 Field 정보 및 Option 설정).

※ Ui type, Label 등을 설정하고 Step 4 버튼 클릭.

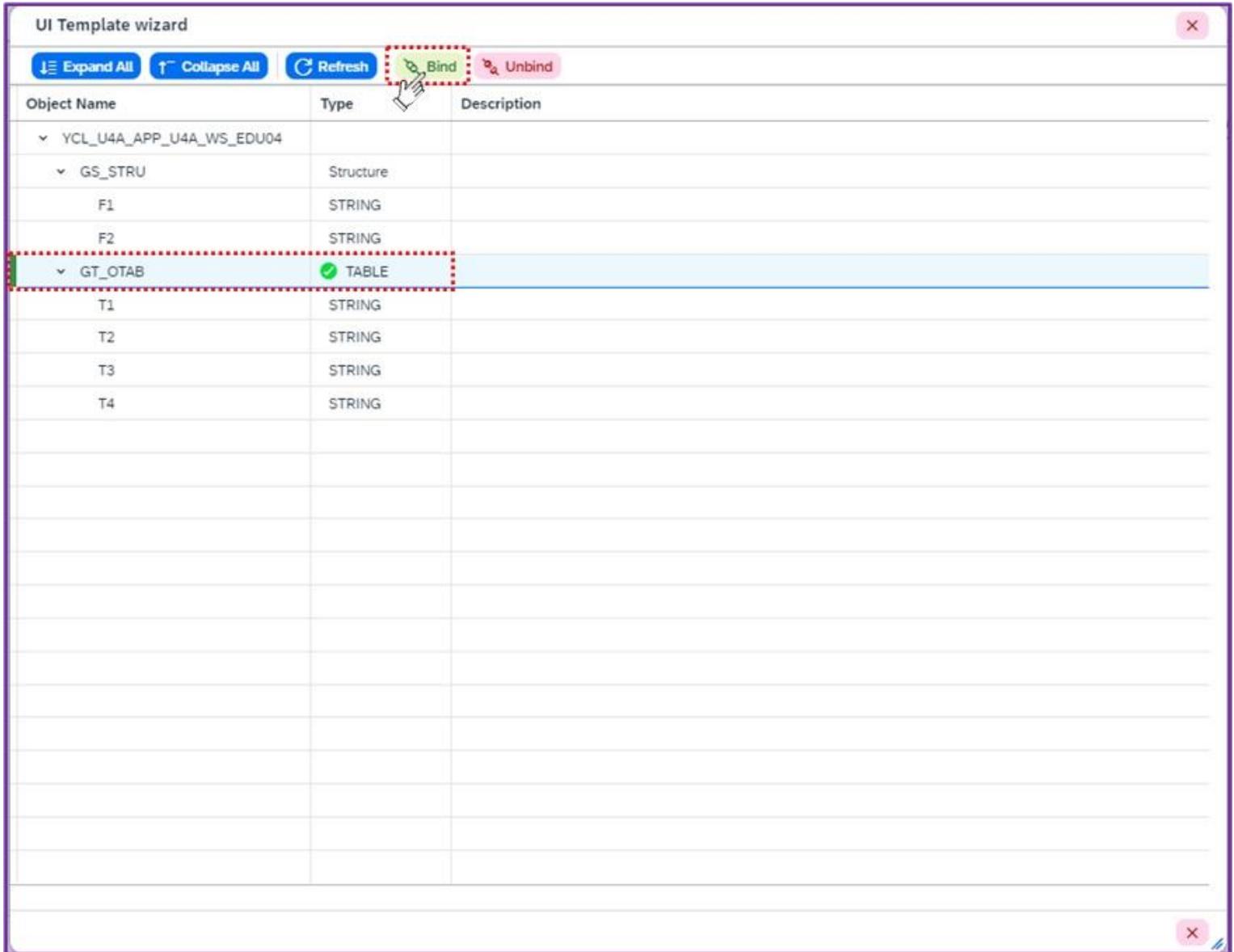


⑤ Table Ui Model Select

※ Table UI에 Binding 시키는 Model 명 (Model Select Button Click)



※ Binding할 Model Click 후 "Bind" Button Click.



⑥ Table Ui Model Information (Binding 되어있는 Table의 Field 정보 및 Option 설정).

※ Ui type, Label 등을 설정하고 생성 아이콘 클릭.

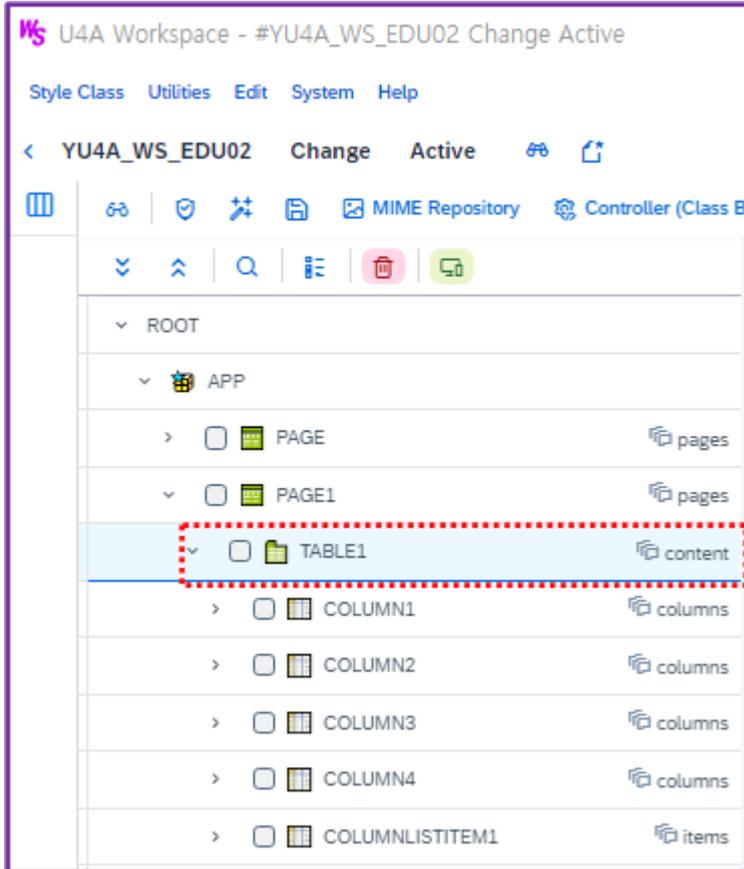
The screenshot displays the U4A IDE interface. On the left is a project tree with the following structure:

- ROOT
 - APP
 - PAGE
 - PANEL1 (content)
 - OVERFLOWTOOLBAR1 (headerToolbar)
 - FORM1 (content)
 - PANEL2 (content)
 - TABLE1 (content)

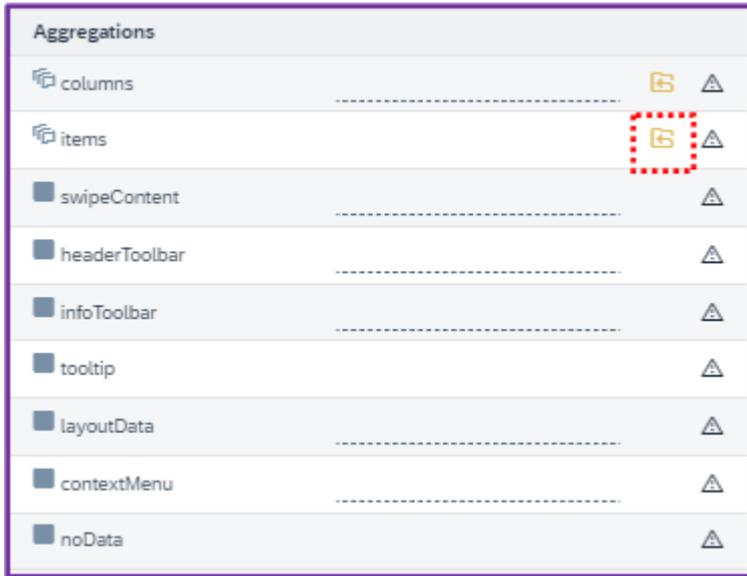
On the right is a preview window titled "preview" with a search interface. The "Search Condition" section includes an "INPUT:" field and a "TEXT:" field. Below it, the "Search Result" section shows a table with four columns: COL1, COL2, COL3, and COL4. The table is currently empty, displaying "No data".

4.23 Table Binding

1. Table Type의 Data를 Binding 할 수 있는 UI를 선택 .

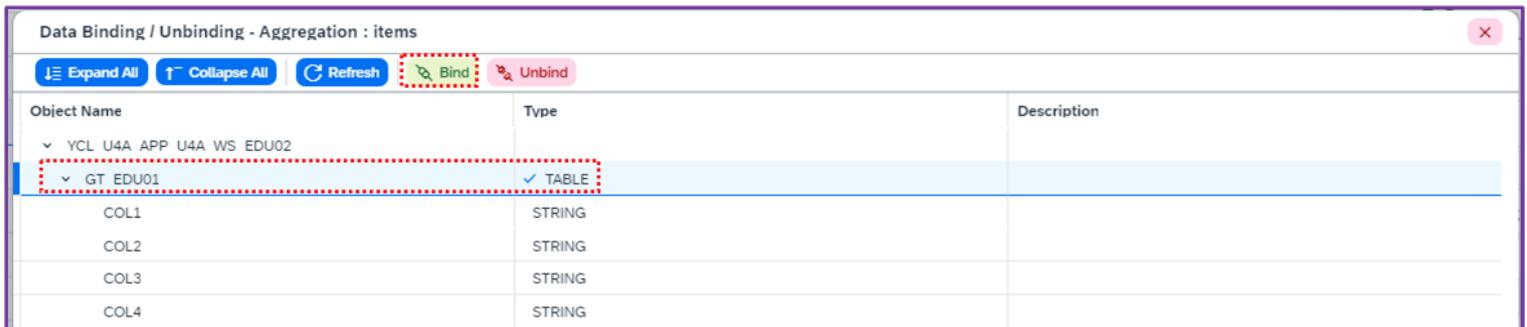


2. 가능한 Aggregation의 folder Icon을 Click.

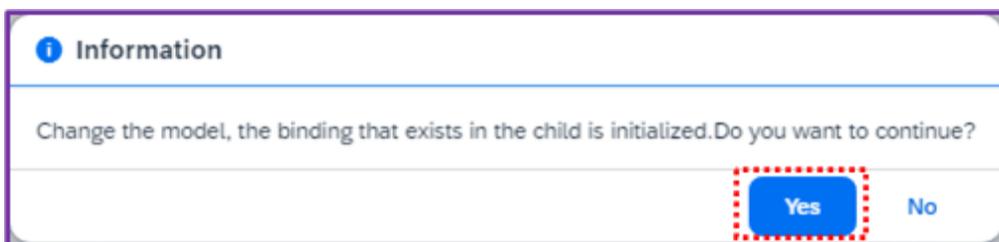


 : Table Type이 Binding 가능함을 나타내는 아이콘.

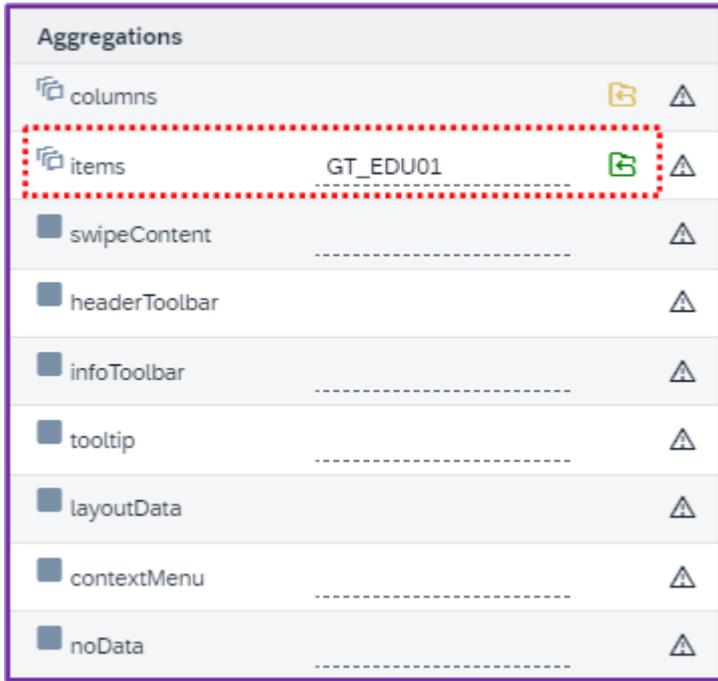
3. Table Type의 Object를 선택한 후 "Bind" Button Click.



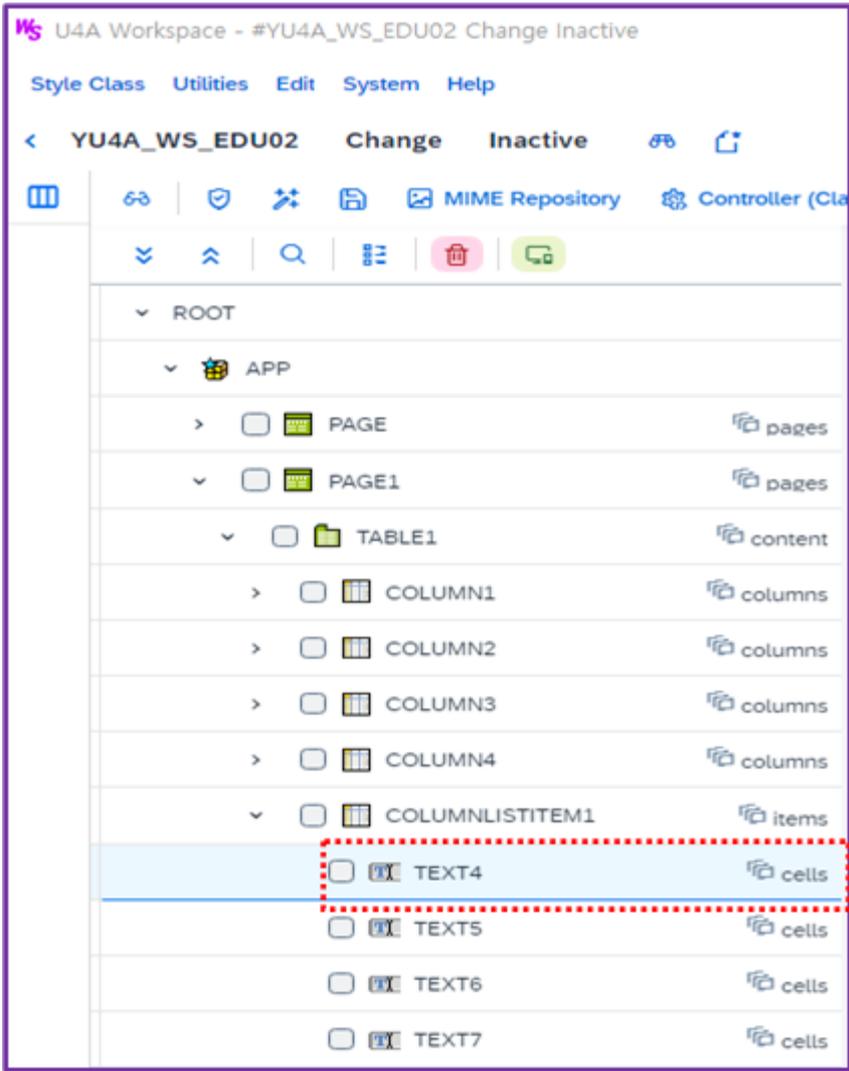
4. Binding이 걸려 있다면 현재 시도하는 Binding으로 변경된다는 경고, "Yes" Button Click.



5. Aggregations에서 Table Type의 Object가 Binding 된 것을 확인.



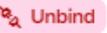
6. Binding 시킨 Table Type Object의 Item을 Binding 시킬 하위 Object를 선택.



7. Binding 시킬 Property에 해당하는 Folder Icon을 Click.

Properties			
text	<input type="text"/>		
textDirection	Inherit		
wrapping	true		
textAlign	Begin		
width	<input type="text"/>		
maxLines	<input type="text"/>		
busy	false		
busyIndicatorDelay	1000		
visible	true		
tooltip	<input type="text"/>		
busyIndicatorSize	Medium		
renderWhitespace	false		
wrappingType	Normal		
emptyIndicatorMode	Off		
styleClass	<input type="text"/>		

8. Binding 되어 있는 Table Type Object의 Item 중에서 Binding 하려는 항목을 선택한 후 "Bind" Button Click.

Data Binding / Unbinding - Property : text		
 Expand All  Collapse All  Refresh  Bind  Unbind		
Object Name	Type	Description
YCL_U4A_APP_U4A_WS_EDU02		
GT_EDU01	TABLE	
COL1	STRING	
COL2	STRING	
COL3	STRING	
COL4	STRING	

9. 해당 Property에 Table의 Item이 Binding 된 것을 확인, 같은 방법으로 나머지 Item들도 Binding 함.

Properties			
text	GT_EDU01-COL1		
textDirection	Inherit		
wrapping	true		
textAlign	Begin		
width			
maxLines			
busy	false		
busyIndicatorDelay	1000		
visible	true		
tooltip			
busyIndicatorSize	Medium		
renderWhitespace	false		
wrappingType	Normal		
emptyIndicatorMode	Off		
styleClass			

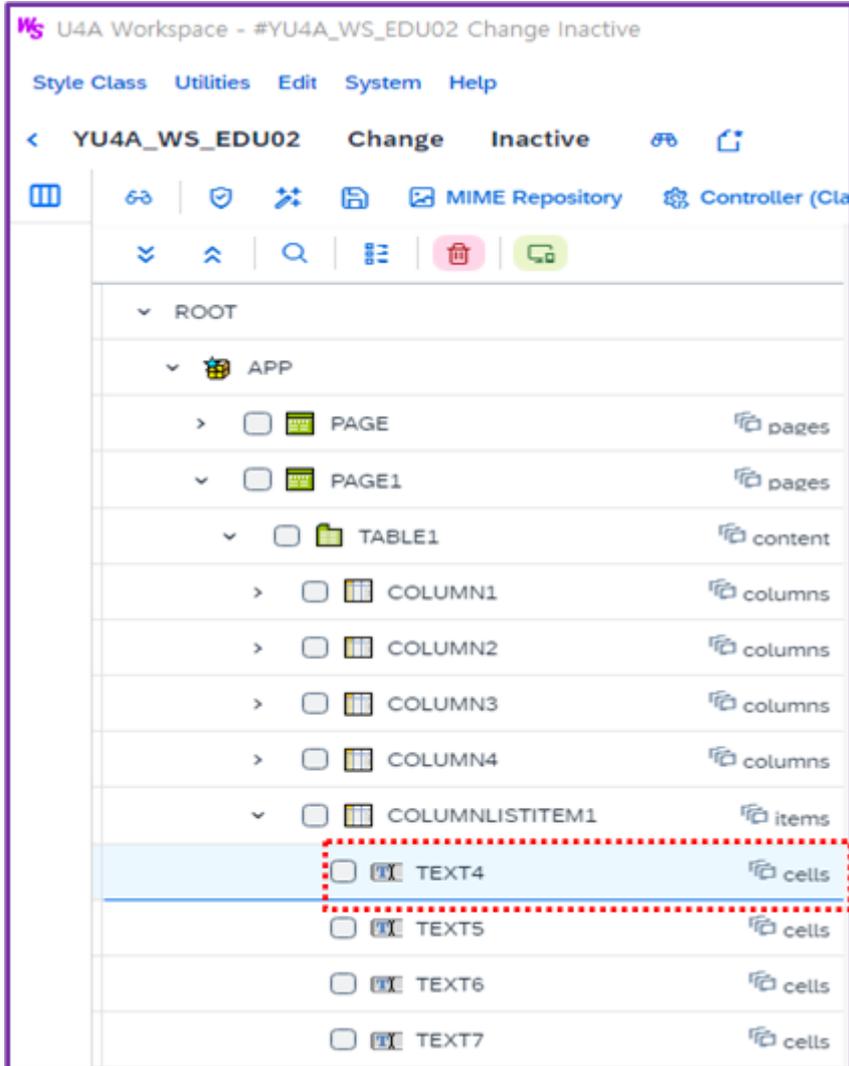
10. Application을 실행하여 Table이 Binding 된 결과 조회.

COL1	COL2	COL3	COL4
A01	A02	A03	A04
B01	B02	B03	B04
C01	C02	C03	C04
D01	D02	D03	D04

 **Bind** 하려는 Table은 반드시 **Public** 유형의 **Internal Table**이어야 함.

4.24 Table unBinding

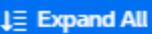
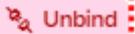
1. Table의 특정 Item을 Unbind 하고 싶을 때 해당하는 UI Click.



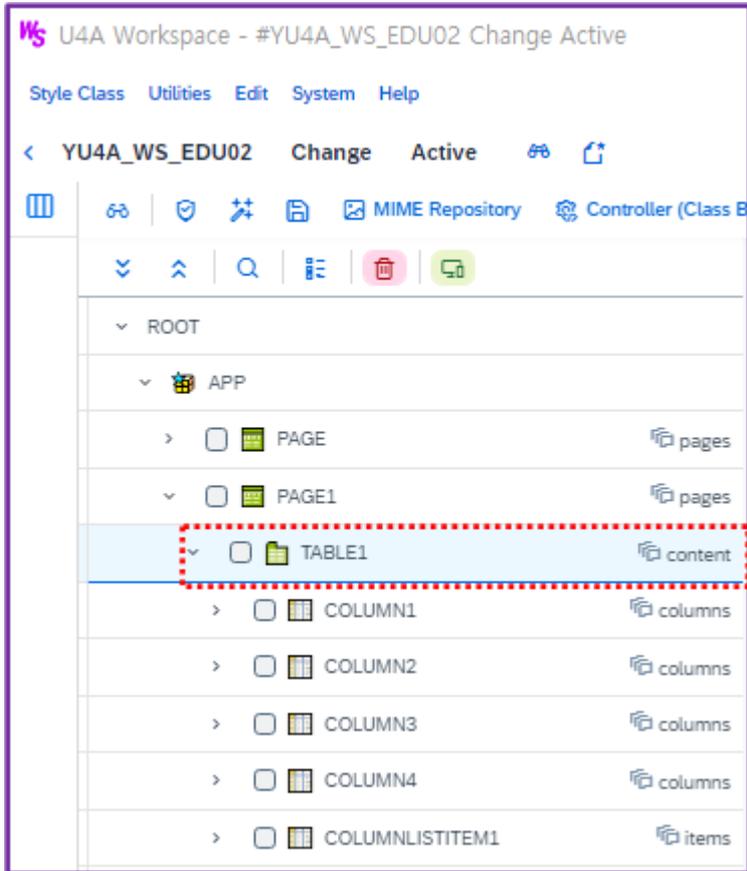
2. Binding 되어 있는 Property의 우측 Folder Icon Click.

Properties			
text	GT_EDU01-COL1		
textDirection	Inherit		
wrapping	true		
textAlign	Begin		
width			
maxLines			
busy	false		
busyIndicatorDelay	1000		
visible	true		
tooltip			
busyIndicatorSize	Medium		
renderWhitespace	false		
wrappingType	Normal		
emptyIndicatorMode	Off		
styleClass			

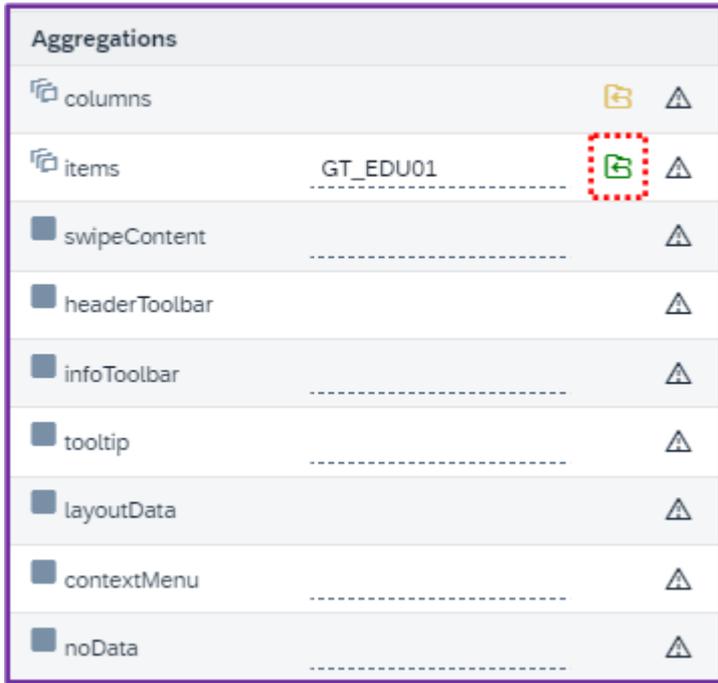
3. "Unbind" Button을 Click하여 Binding 해제 (해당 창에서 어느 Item을 선택하는지 상관 없이 Check 표시 되어 있는 Item이 Unbind).

Data Binding / Unbinding - Property : text		
 Expand All  Collapse All  Refresh  Bind  Unbind		
Object Name	Type	Description
YCL_U4A_APP_U4A_WS_EDU02		
GT_EDU01	TABLE	
COL1	✓ STRING	
COL2	✓ STRING	
COL3	✓ STRING	
COL4	✓ STRING	

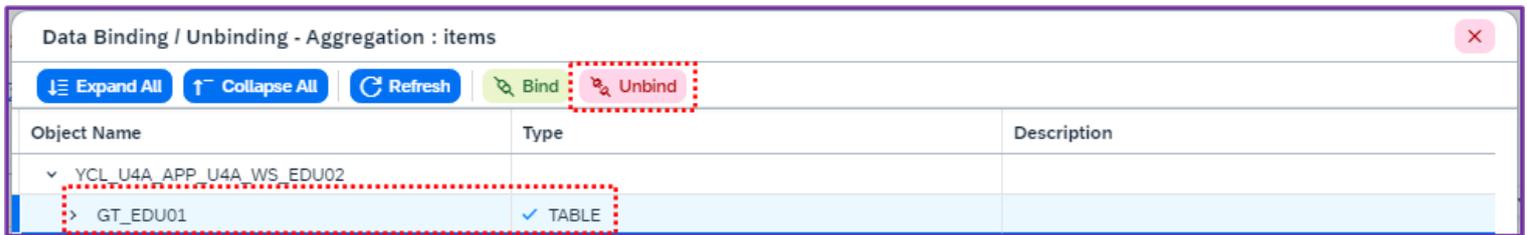
4. Table Type의 Object 자체를 Unbind 하고 싶을 때 해당하는 UI Click.



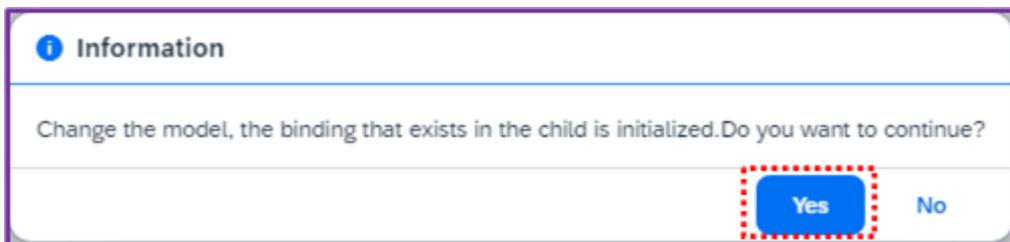
5. Binding 되어 있는 Aggregation의 우측 Folder를 Click.



6. Unbind Button을 Click. (어느 Object를 선택하는 지는 상관 없이 Binding 되어 있는 Object가 Unbind).

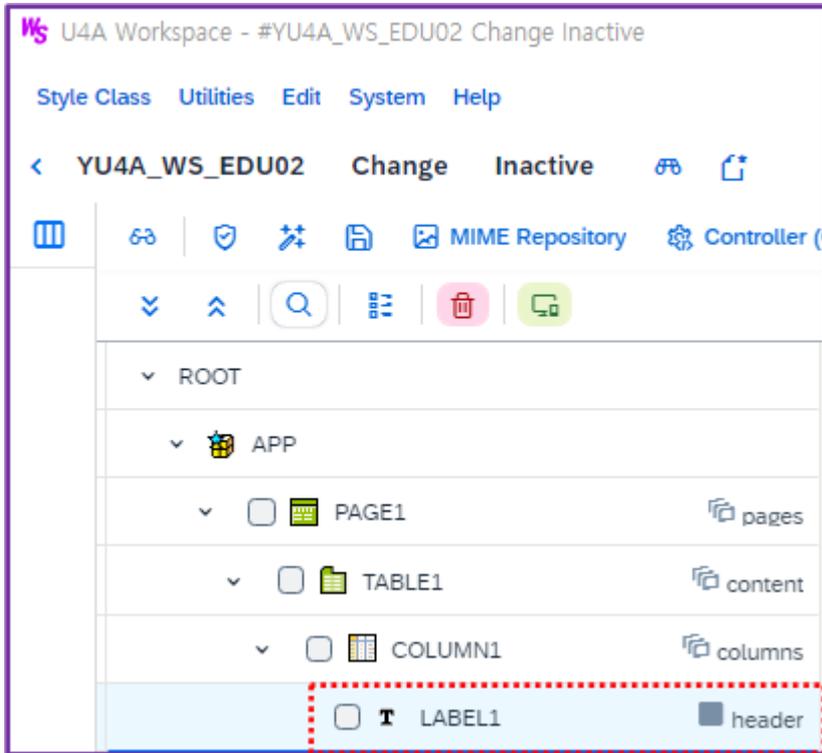


7. Table Object 자체를 Unbind 하는 경우 하위 Binding 정보도 모두 초기화 되는 것을 경고, "Yes" Button Click하여 Unbind.



4.25 Structure Binding

1. Structure Type의 Data를 Binding 할 수 있는 UI 선택 (Aggregation에 + 표시가 되어 있는 UI).



2. Binding 하려는 Property에 해당하는 Folder Icon Click.

Properties			
design	Bold	▼	 
text			 
textAlign	Begin	▼	 
textDirection	Inherit	▼	 
width			 
required	false	▼	 
busy	false	▼	 
busyIndicatorDelay	1000		 
visible	true	▼	 
tooltip			 
busyIndicatorSize	Medium	▼	 
displayOnly	false	▼	 
vAlign	Inherit	▼	 
wrapping	false	▼	 
wrappingType	Normal	▼	 
showColon	false	▼	 
styleClass			 

3. 선택한 UI에 Binding 시키고 싶은 Structure의 특정 Field를 선택한 후 "Bind" Button Click.

Data Binding / Unbinding - Property : text			
    			
Object Name	Type	Description	
YCL_U4A_APP_U4A_WS_EDU02			
GS_STRU	YCL_U4A_APP_U4A_WS_EDU02-TY_ST		
COL1	✓ STRING		
COL2	✓ STRING		
COL3	✓ STRING		
COL4	✓ STRING		

4. 해당 UI의 Property에 선택한 Structure의 Field가 Binding 된 것을 확인, 나머지 Field도 같은 방식으로 Binding.

Properties			
design	Bold		
text	GS_STRU-COL1		
textAlign	Begin		
textDirection	Inherit		
width			
required	false		
busy	false		
busyIndicatorDelay	1000		
visible	true		
tooltip			
busyIndicatorSize	Medium		
displayOnly	false		
vAlign	Inherit		
wrapping	false		
wrappingType	Normal		
showColon	false		
styleClass			

5. Application을 실행하여 Structure가 Binding 된 결과 조회.

COL1	COL2	COL3	COL4

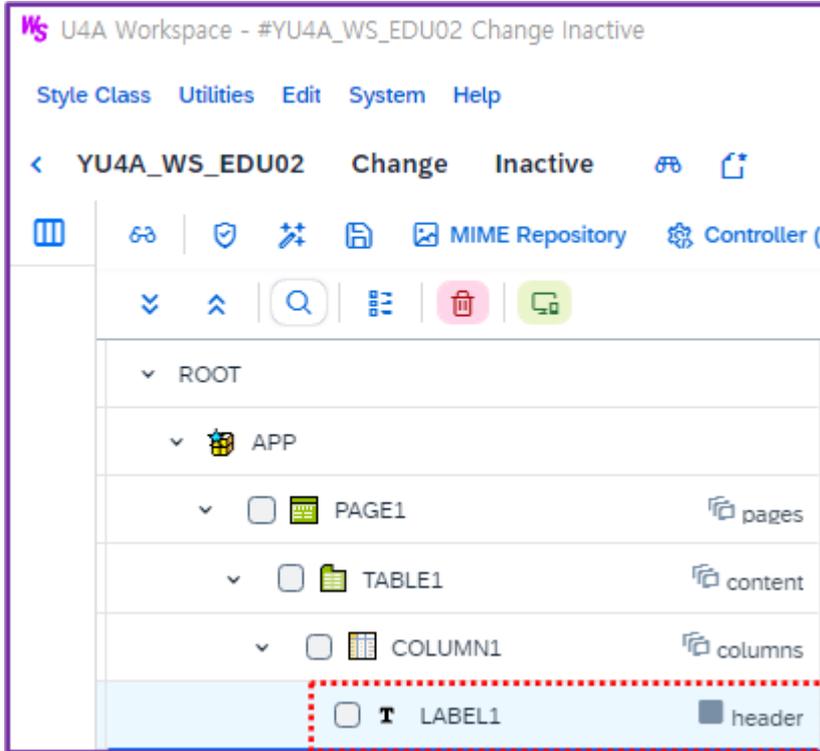


단일 값을 Binding 할 때는 Structure Type으로 Data를 생성한 후 Structure의 Field를 Binding 해야 하며 반드시 Public 해야 함.

단일 변수(Single Variable)를 Binding 하는 방식은 제공하지 않음.

4.26 Structure Unbinding

1. Structure Type의 Data가 Binding 되어 있는 UI 선택.



2. Data가 Binding 되어 있는 Property에 해당하는 Folder Icon Click .

Properties			
design	Bold	▼	 
text	GS_STRU-COL1		 
textAlign	Begin	▼	 
textDirection	Inherit	▼	 
width			 
required	false	▼	 
busy	false	▼	 
busyIndicatorDelay	1000		 
visible	true	▼	 
tooltip			 
busyIndicatorSize	Medium	▼	 
displayOnly	false	▼	 
vAlign	Inherit	▼	 
wrapping	false	▼	 
wrappingType	Normal	▼	 
showColon	false	▼	 
styleClass			 

3. "Unbind" Button을 Click하여 Binding 해제 (해당 창에서 어느 Field를 선택하는지 상관 없 이 Check 표시가 되어 있는 Field Unbind).

Data Binding / Unbinding - Property : text

Object Name	Type	Description
YCL_U4A_APP_U4A_WS_EDU02		
GS_STRU	YCL_U4A_APP_U4A_WS_EDU02-TY_ST	
COL1	✓ STRING	
COL2	✓ STRING	
COL3	✓ STRING	
COL4	✓ STRING	

4. 해당 UI를 Click하여 Properties에서 Unbind 된 것을 확인.

Properties

design	Bold	<input type="button" value="Bind"/>	<input type="button" value="Unbind"/>	<input type="button" value="Help"/>
text		<input type="button" value="Bind"/>	<input type="button" value="Unbind"/>	<input type="button" value="Help"/>
textAlign	Begin	<input type="button" value="Bind"/>	<input type="button" value="Unbind"/>	<input type="button" value="Help"/>
textDirection	Inherit	<input type="button" value="Bind"/>	<input type="button" value="Unbind"/>	<input type="button" value="Help"/>
width		<input type="button" value="Bind"/>	<input type="button" value="Unbind"/>	<input type="button" value="Help"/>
required	false	<input type="button" value="Bind"/>	<input type="button" value="Unbind"/>	<input type="button" value="Help"/>
busy	false	<input type="button" value="Bind"/>	<input type="button" value="Unbind"/>	<input type="button" value="Help"/>
busyIndicatorDelay	1000	<input type="button" value="Bind"/>	<input type="button" value="Unbind"/>	<input type="button" value="Help"/>
visible	true	<input type="button" value="Bind"/>	<input type="button" value="Unbind"/>	<input type="button" value="Help"/>
tooltip		<input type="button" value="Bind"/>	<input type="button" value="Unbind"/>	<input type="button" value="Help"/>
busyIndicatorSize	Medium	<input type="button" value="Bind"/>	<input type="button" value="Unbind"/>	<input type="button" value="Help"/>
displayOnly	false	<input type="button" value="Bind"/>	<input type="button" value="Unbind"/>	<input type="button" value="Help"/>
vAlign	Inherit	<input type="button" value="Bind"/>	<input type="button" value="Unbind"/>	<input type="button" value="Help"/>
wrapping	false	<input type="button" value="Bind"/>	<input type="button" value="Unbind"/>	<input type="button" value="Help"/>
wrappingType	Normal	<input type="button" value="Bind"/>	<input type="button" value="Unbind"/>	<input type="button" value="Help"/>
showColon	false	<input type="button" value="Bind"/>	<input type="button" value="Unbind"/>	<input type="button" value="Help"/>
styleClass		<input type="button" value="Bind"/>	<input type="button" value="Unbind"/>	<input type="button" value="Help"/>

4.27 Binding Options

Property	Value
Field name	COL1
Field path	GT_EDU01-COL1
type	STRING
Bind type	
Reference Field name	
Conversion Routine	
Nozero	false
Is number format?	false

Property	Means
Field name	현재 바인딩 하기 위해 선택한 필드 명
Field path	선택한 필드가 위치하고 있는 경로
type	선택한 필드가 참조하고 있는 Type
Bind type	숫자 타입의 필드에 적용 시킬 단위 타입 선택. <ul style="list-style-type: none"> • sap.ui.model.type.Currency • ext.ui.model.type.Quantity
Reference Field name	Bind Type 설정 시 참조 할 필드 명
Conversion Routine	선택한 필드에 Conversion Routine 적용
Nozero	숫자 타입일 경우 Initial 값으로 0을 보이게 할지 여부 결정
Is number format?	숫자 타입의 필드일 경우 숫자 포맷으로 보이게 할지 여부 결정

4.27.1 Property – Bind Type

▣ 목적 : 단위에 해당하는 Field를 Binding 하여 Format 설정 가능 .

1. 금액, 수량에 해당하는 통화키/단위에 해당하는 Field를 Binding하여 해당 Format 적용하는 것이 가능.

CARRID	CONNID	FLDATE	PRICE	CURRENCY	PLANETYPE
AA	0017	2013.05.01	42294	USD	747-400
AA	0021	2016.04.08	7567545	USD	747-400
AA	0030	2019.11.20	352523	EUR	747-400
AZ	0041	2014.07.01	12345	KRW	747-400
AZ	0055	2019.12.20	9796543	KRW	747-400

2. 숫자 Type의 Field를 선택하여 Binding 시키는 과정에서 Bind type에 Currency Type, Quantity Type 중에서 선택.

Reference Field name에 참조 할 단위에 해당하는 Field를 입력한 후 "Bind" Button Click.

Object Name	Type	Description	Property	Value
YCL_U4A_APP_U4A_WS_EDU01			Field name	PRICE
GT_SFLIGHT	TABLE		Field path	GT_SFLIGHT-PRICE
CARRID	C		type	P
CONNID	N		Bind type	sap.ui.model.type.Currency
FLDATE	D		Reference Field name	GT_SFLIGHT-CURRENCY
PRICE	P		Conversion Routine	
CURRENCY	S_CURRCODE		Nozero	false
PLANETYPE	C		Is number format?	false

3. Bind Type 설정 할 때 반드시 Reference Field name을 함께 설정해야 함, 그렇지 않으면 Error.

If Bind type is selected, Reference Field name is required.

4. Application 실행하여 Price Field의 값에 참조 Field Currency의 값이 적용 된 것을 확인.

U4A IDE HELP

CARRID	CONNID	FLDATE	PRICE	CURRENCY	PLANETYPE
AA	0017	2013.05.01	42.294,00	USD	747-400
AA	0021	2016.04.08	7.567.545,00	USD	747-400
AA	0030	2019.11.20	352.523,00	EUR	747-400
AZ	0041	2014.07.01	1.234.500	KRW	747-400
AZ	0055	2019.12.20	979.654.300	KRW	747-400

4.27.2 Property – Nozero

▣ 목적 : Number Type의 Data가 0일 때 0을 보이게 할 지 여부 설정.

1. 숫자 Type Data가 Initial 값으로 0이 입력될 경우 0이 입력되지 않도록 설정 가능.

CARRID	CONNID	FLDATE	PRICE	CURRENCY	PLANETYPE
AA	0017	2013.05.01	42.284	KRW	747-400
AA	0021	2016.04.08	7.567.545	KRW	747-400
AA	0030	2019.11.20	352.523	KRW	747-400
AZ	0041	2014.07.01	12.345	KRW	747-400
AZ	0055	2019.12.20	9.796.543	KRW	747-400
AZ	0064	2018.03.04	0	KRW	747-400

2. Data Binding 시키는 과정에서 Nozero를 true로 선택하고 "Bind" Button Click.

Object Name	Type	Description	Property	Value
YCL_U4A_APP_U4A_WS_EDU01			Field name	PRICE
GT_SFIGHT	TABLE		Field path	GT_SFIGHT-PRICE
CARRID	C		type	P
CONNID	N		Bind type	sap.ui.model.type.Currency
FLDATE	D		Reference Field name	GT_SFIGHT-CURRENCY
PRICE	P		Conversion Routine	Nozero
CURRENCY	S_CURRCODE		Nozero	true
PLANETYPE	C		Is number format?	false

3. Application 실행하여 Price Field의 값이 없는 경우 0이 입력 되지 않은 것을 확인 .

CARRID	CONNID	FLDATE	PRICE	CURRENCY	PLANETYPE
AA	0017	2013.05.01	42.284	KRW	747-400
AA	0021	2016.04.08	7.567.545	KRW	747-400
AA	0030	2019.11.20	352.523	KRW	747-400
AZ	0041	2014.07.01	12.345	KRW	747-400
AZ	0055	2019.12.20	9.796.543	KRW	747-400
AZ	0064	2018.03.04		KRW	747-400

4.27.3 Property – Is number format

☐ 목적 : Number Type의 Data에 Number Format 설정 가능 .

1. Data가 숫자 Type과 문자 Type의 구분이 되지 않을 시 구분 가능하도록 format 변경 가능 .

CARRID	CONNID	FLDATE	PRICE	CURRENCY	PLANETYPE
AA	0017	2013.05.01	42294	KRW	747-400
AA	0021	2016.04.08	7567545	KRW	747-400
AA	0030	2019.11.20	352523	KRW	747-400
AZ	0041	2014.07.01	12345	KRW	747-400
AZ	0055	2019.12.20	9796543	KRW	747-400

2. 숫자 Type Data Binding 시 Is number format을 true로 선택.

Data Binding / Unbinding - Property : text ✕

Expand All
Collapse All
Refresh
Bind
Unbind

Object Name	Type	Description	Property	Value
YCL_U4A_APP_U4A_WS_EDU01			Field name	PRICE
GT_SFLIGHT	TABLE		Field path	GT_SFLIGHT-PRICE
CARRID	C		type	P
CONNID	N		Bind type	
FLDATE	D		Reference Field name	
PRICE	P		Conversion Routine	
CURRENCY	S_CURRCODE		Nozero	false
PLANETYPE	C		Is number format?	true

3. Price Field의 값이 숫자 Type format으로 변경된 것을 확인.

CARRID	CONNID	FLDATE	PRICE	CURRENCY	PLANETYPE
AA	0017	2013.05.01	42.294	USD	747-400
AA	0021	2016.04.08	7.567.545	USD	747-400
AA	0030	2019.11.20	352.523	EUR	747-400
AZ	0041	2014.07.01	12.345	KRW	747-400
AZ	0055	2019.12.20	9.796.543	KRW	747-400

Is number format을 true로 설정하려면 Bind type이 Unbinding 상태여야 함.

4.27.4 Property – Conversion Routine

▣ 목적 : 저장될 때와 Display 될 때 Format 변경하기 위한 목적.

1. Conversion Routine이 설정 되기 전 실제로 Table에 들어 있는 CONNID Field의 값 .

CARRID	CONNID	FLDATE	PRICE	CURRENCY	PLANETYPE
AA	0017	2013.05.01	42.294,00	USD	747-400
AA	0021	2016.04.08	7.567.545,00	USD	747-400
AA	0030	2019.11.20	352.523,00	EUR	747-400
AZ	0041	2014.07.01	1.234.500	KRW	747-400
AZ	0055	2019.12.20	979.654.300	KRW	747-400

2. Data Binding 시에 Conversion Routine 명을 입력한 후 "Bind" Button Click .

Data Binding / Unbinding - Property : text

Expand All Collapse All Refresh Bind Unbind

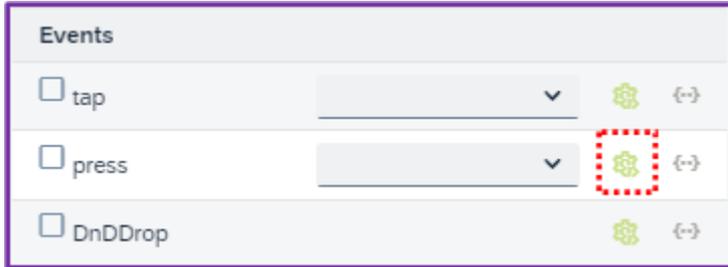
Object Name	Type	Description	Property	Value
YCL_U4A_APP_U4A_WS_EDU01			Field name	CONNID
GT_SFLIGHT	TABLE		Field path	GT_SFLIGHT-CONNID
CARRID	C		type	N
CONNID	N		Bind type	
FLDATE	D		Reference Field name	
PRICE	P		Conversion Routine	ALPHA
CURRENCY	S_CURRCODE		Nozero	false
PLANETYPE	C		Is number format?	false

3. Data 조회시에 2자리 정수값으로 보이지만 실제 Table에는 4자리 Numeric 형태로 저장되도록 Conversion Routine이 설정된 것을 확인 .

CARRID	CONNID	FLDATE	PRICE	CURRENCY	PLANETYPE
AA	17	2013.05.01	42.294,0000	USD	747-400
AA	21	2016.04.08	7.567.545,0000	USD	747-400
AA	30	2019.11.20	352.523,0000	EUR	747-400
AZ	41	2014.07.01	12.345,0000	KRW	747-400
AZ	55	2019.12.20	9.796.543,0000	KRW	747-400

4.28 Events

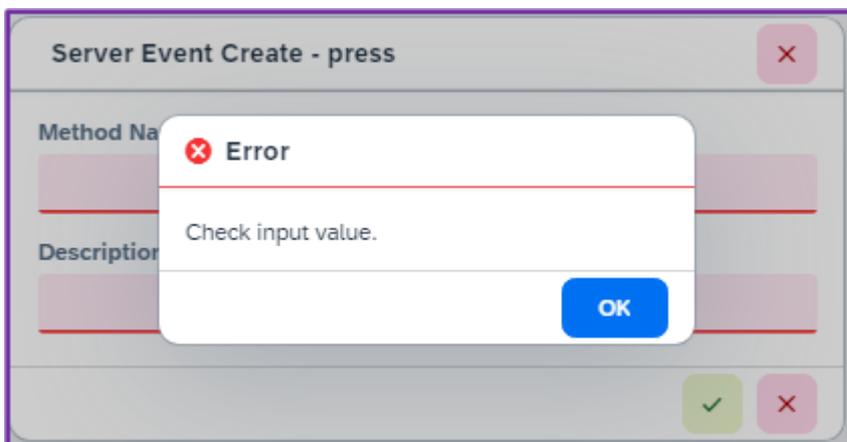
1. Events에서 Server Event(Method) Button을 Click.



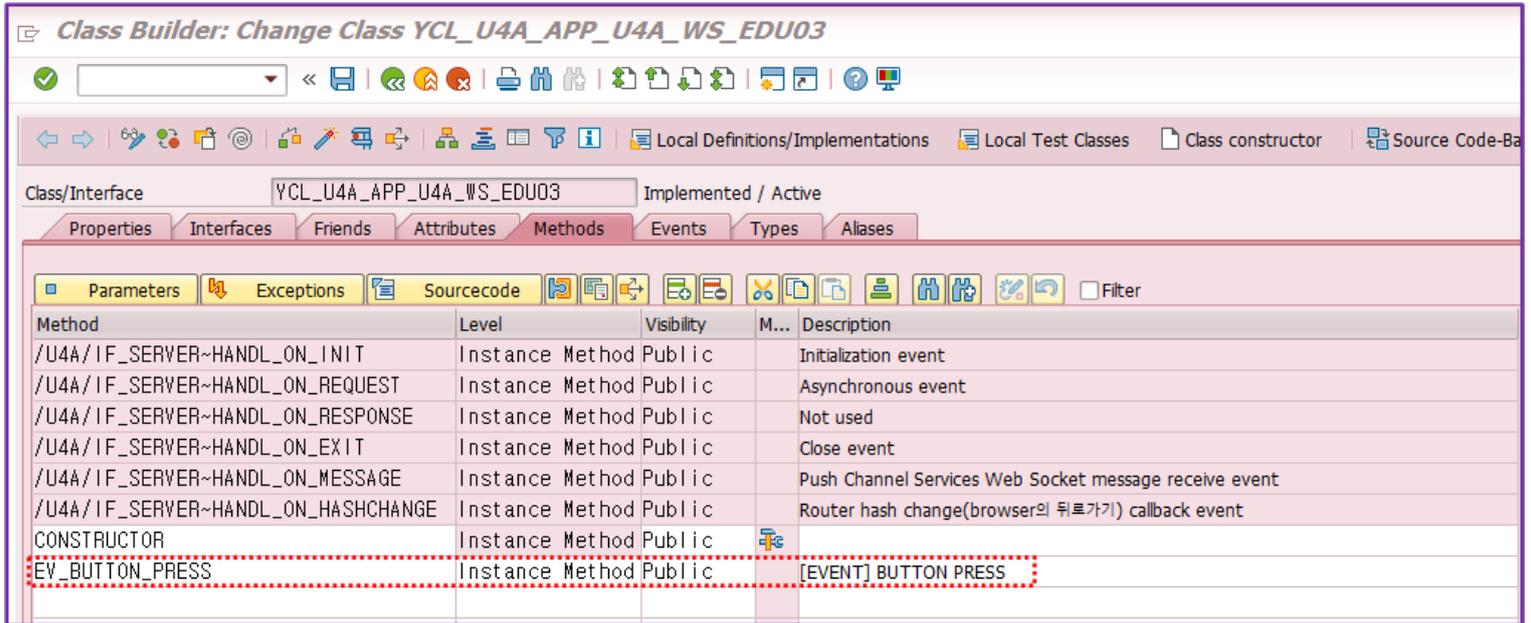
2. Method Name과 Description을 작성한 후 확인 Button Click.



3. Method Name이나 Description 중 하나라도 입력하지 않았을 경우, Error 표시 .



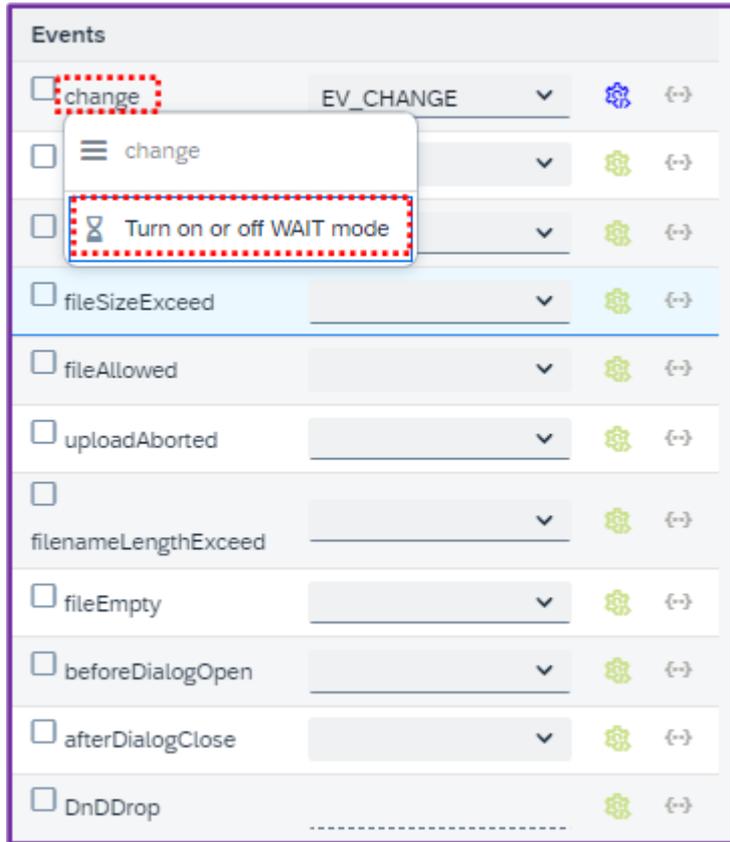
4. 정상적으로 진행 되었다면 EV_Method Name 으로 이름이 생성 되며 Instance Method이면서 Public 상태로 생성됨.



4.29 CheckBox Select

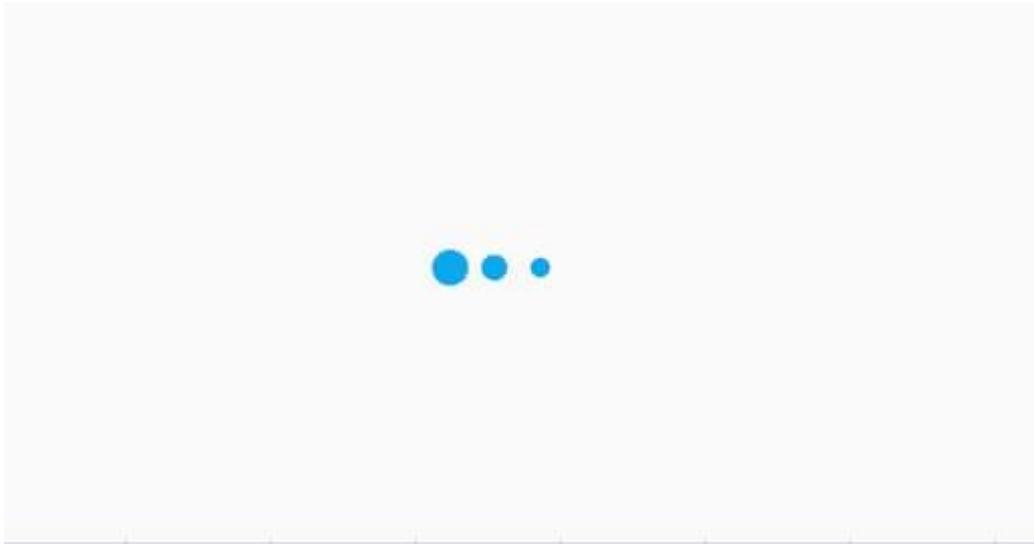
■ 목적 : Event 실행 시 Loading 중에 다른 작업을 동시에 가능하도록 설정 여부 .

1. UI의 Events 종류(현재 change)에서 마우스 우클릭 후 나타난 Pop-Up "Turn on or off WAIT mode" 선택.



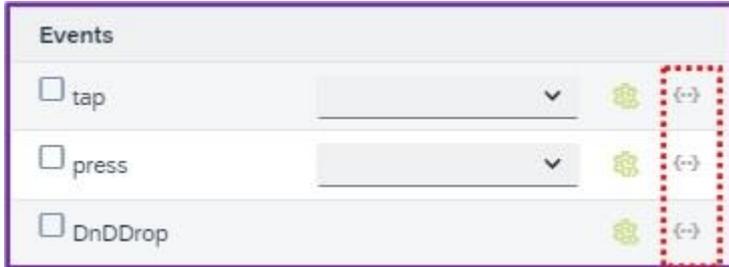
2. Event의 Method가 동작할 때 아래의 Loading 화면이 생기지 않고 Loading 중에 다른 작업을 할 수 있음.

하지만 Loading 중에 다른 작업이 가능할 경우 심각한 Lisk를 초래할 수 있으므로 기본적으로 Check Box는 Initial로 설정.

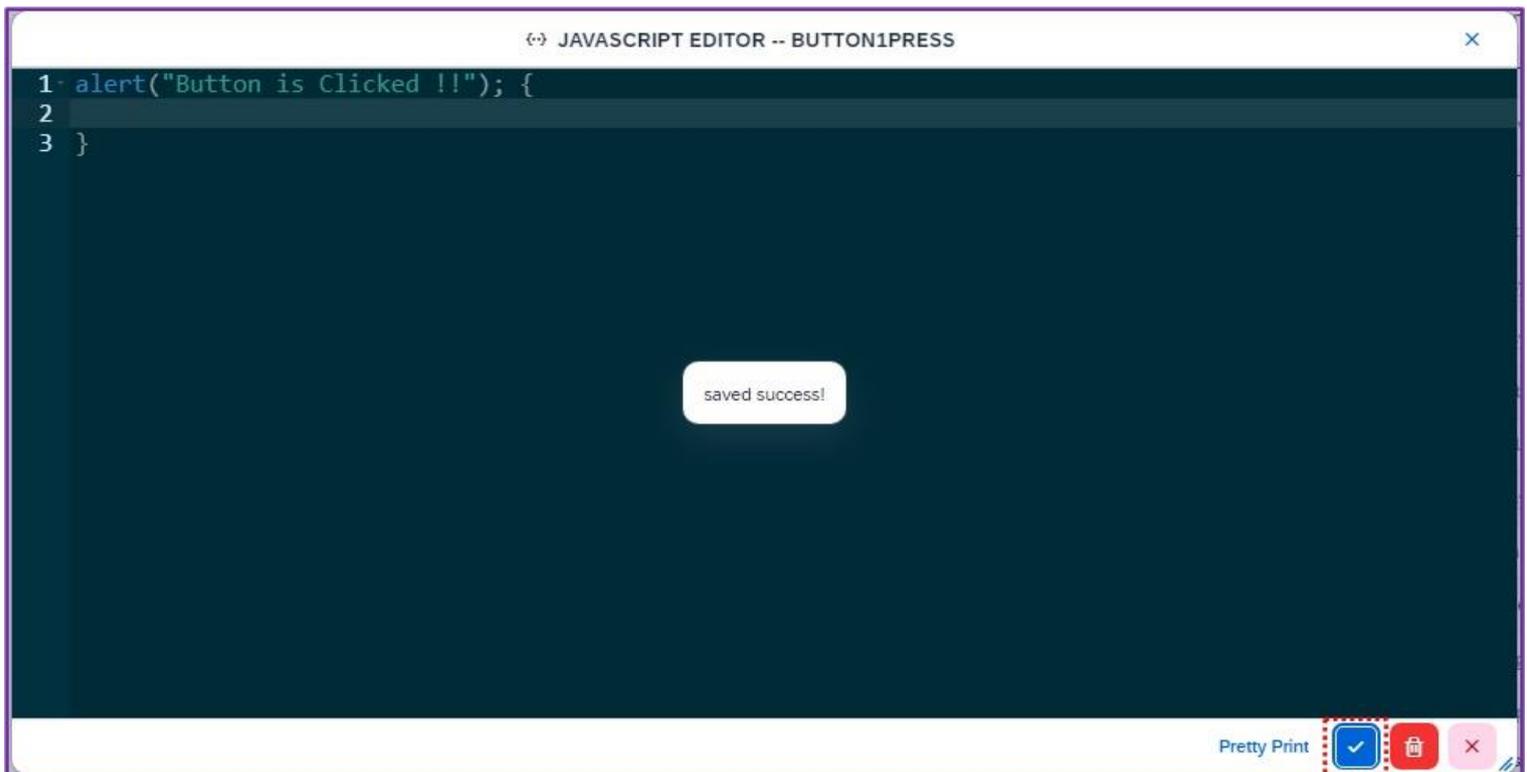


4.30 Java Script

1. JS를 적용시키고 싶은 Event에 해당하는 Script Content Button Click.



2. JS를 자유롭게 입력할 수 있는 Editor가 생성, User가 JS를 적용 시킨 후 "Save" Button을 누르면 성공 Message와 함께 Event에 JS 적용.



3. Java Script Editor Button.

Pretty Print : 작성된 소스코드 정렬.

Save : 현재까지 작성된 Java Script를 Editor에 저장하고 Event에 적용시킴.

Clear : 해당 UI Event에 적용된 모든 Java Script 를 초기화 시킴.

Close : Editor를 종료하고 가장 최근에 저장한 상태로 RollBack.



4. JS가 성공적으로 저장되었다면 Script Content Button이 적색으로 변경.

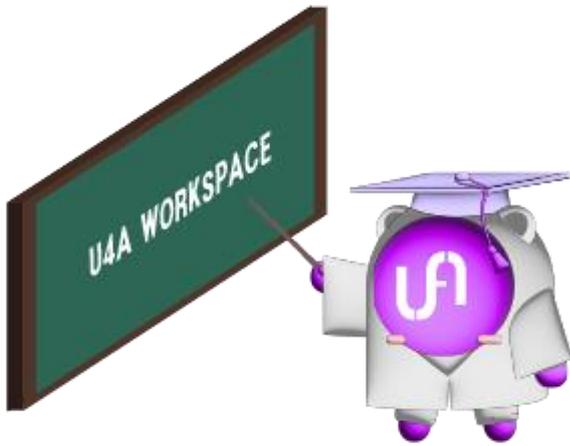


5. Application을 실행하여 JS가 성공적으로 적용된 것을 확인.



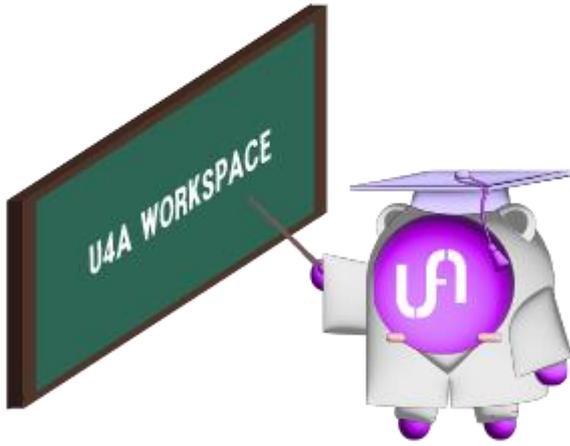
 불가피한 경우를 제외하고 필요 이상으로 Java Script를 사용하는 것은 권하지 않음.

5. APP 개발 하기 Part 1



5.1 추가 기능 메뉴 - Style Class

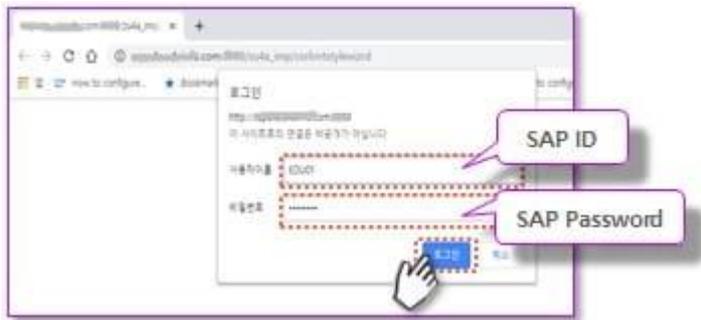
5.1.1 Font Style Wizard



5.1.1.1 Font Style Wizard 1



2) 실행된 Browser에서 만약 Login 정보 요구 시 자신의 SAP ID, SAP Password를 입력함.



3) 본 Wizard 화면에서 원하는 속성을 설정 후 얻어진 "CSS Source" 영역의 Code를 마우스로 Block Copy 함.

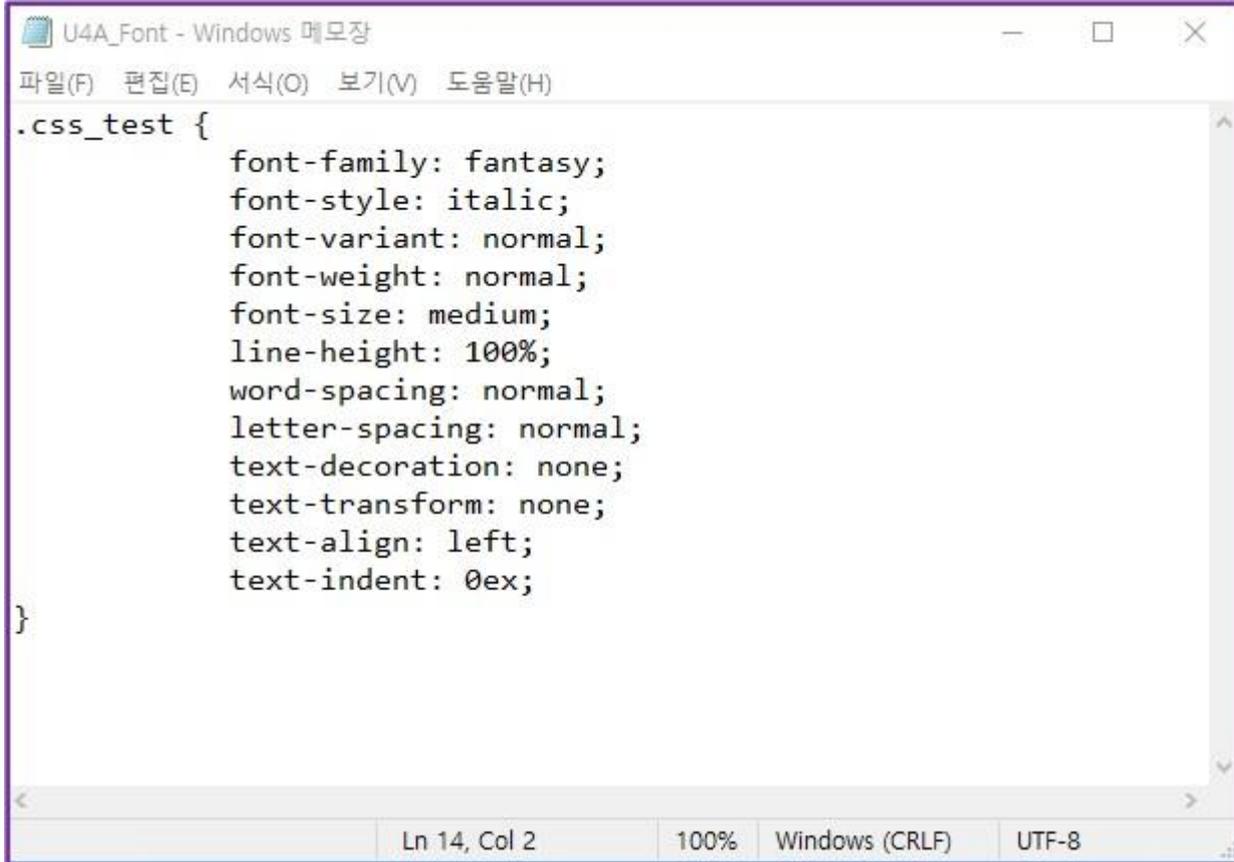
단 "CSS Source"에서 code 복사 시 예시와 같이 "p.sample(...)" 영역의 CSS Code 값만 복사 해야 함.

```
p.sample {
    font-family: fantasy;
    font-style: italic;
    font-weight: normal;
    font-size: medium;
    line-height: 100%;
    word-spacing: normal;
    text-decoration: none;
    text-align: left;
    text-indent: 0ex;
}
※ "p.sample" CSS Class Code
```

 CSS 기술에 대한 추가 정보를 원할 시 ["CSS Tutorial - W3Schools"](https://www.w3schools.com/css/) Web Site를 참고 하기 바람.

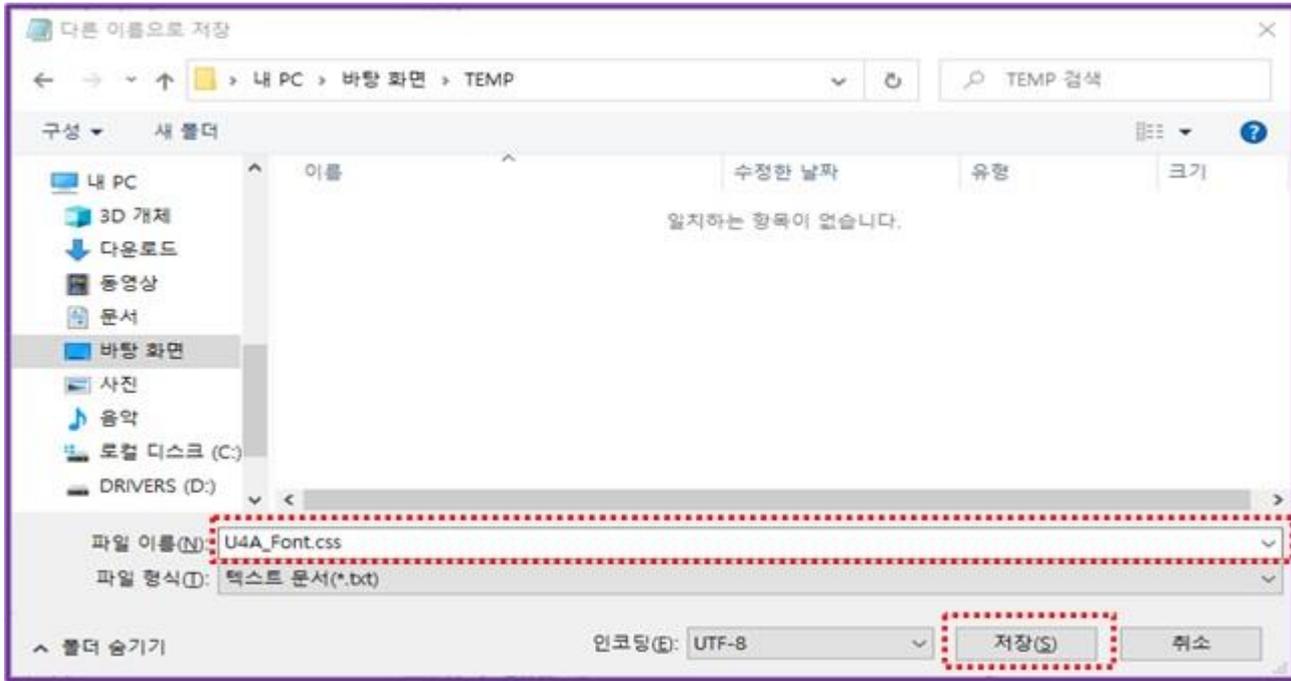
5.1.1.2 Font Style Wizard 2

- 1) 메모장(notepad.exe) 실행 후 복사된 Code 붙여 넣은 후 예제와 같이 CSS Class ID를 “.css_test”로 변경함.

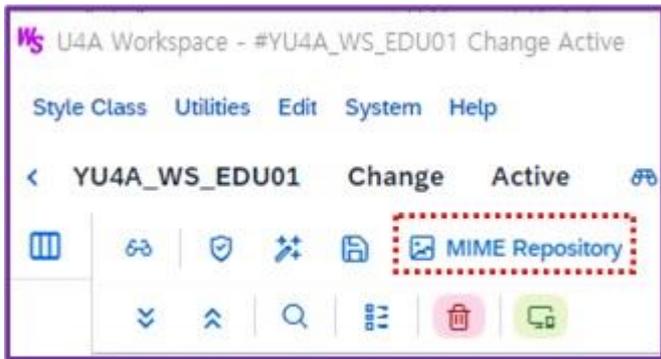


```
.css_test {  
    font-family: fantasy;  
    font-style: italic;  
    font-variant: normal;  
    font-weight: normal;  
    font-size: medium;  
    line-height: 100%;  
    word-spacing: normal;  
    letter-spacing: normal;  
    text-decoration: none;  
    text-transform: none;  
    text-align: left;  
    text-indent: 0ex;  
}
```

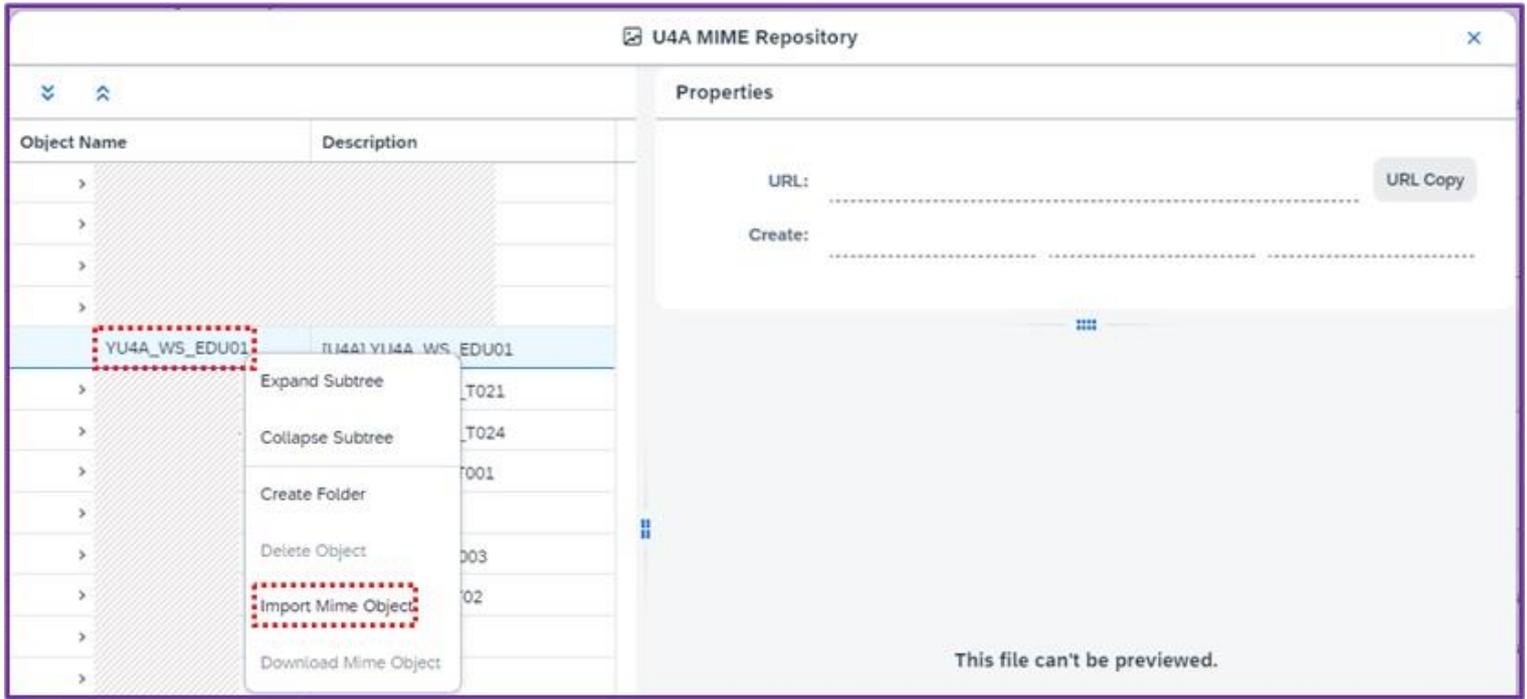
- 2) “다른 이름으로 저장” 으로 적절한 위치에 “.css” 확장자 File로 저장함. (ex: U4A_Font.css)



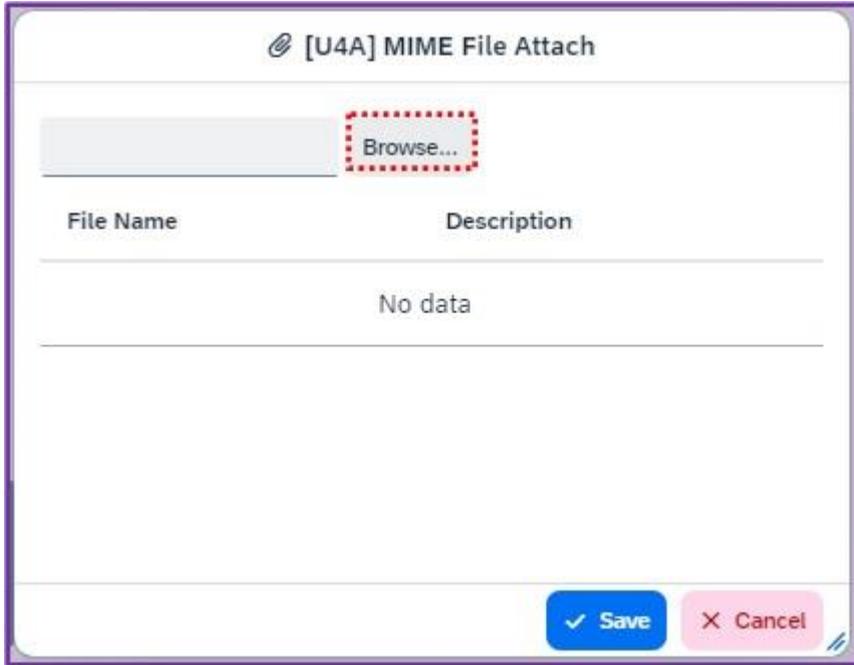
3) U4A Workbench 메인 Toolbar 영역 Button 중 “MIME Repository”를 Click함.



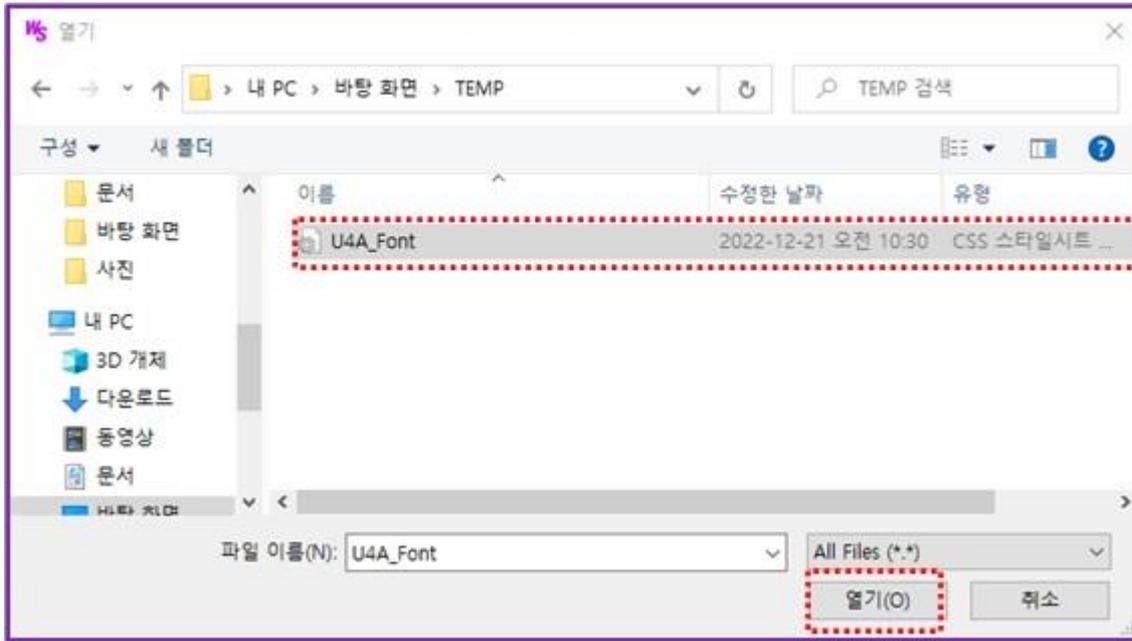
4) 활성화된 “MIME Repository” Popup에서 현재 수정 중인 App. ID를 마우스 우클릭 하여 Sub menu 중 “Import MIME Object”를 Click함.



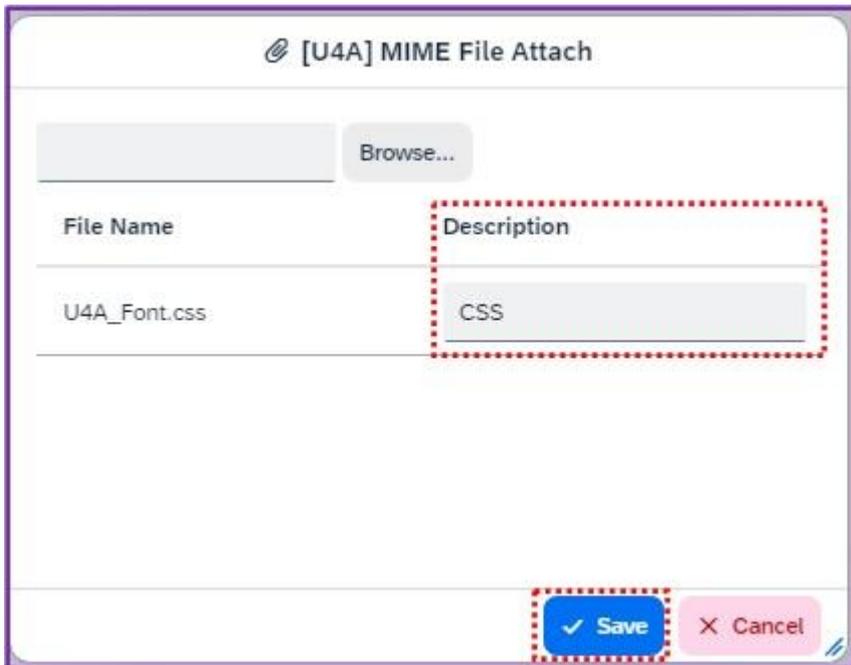
5) "Browser" Button을 Click함.



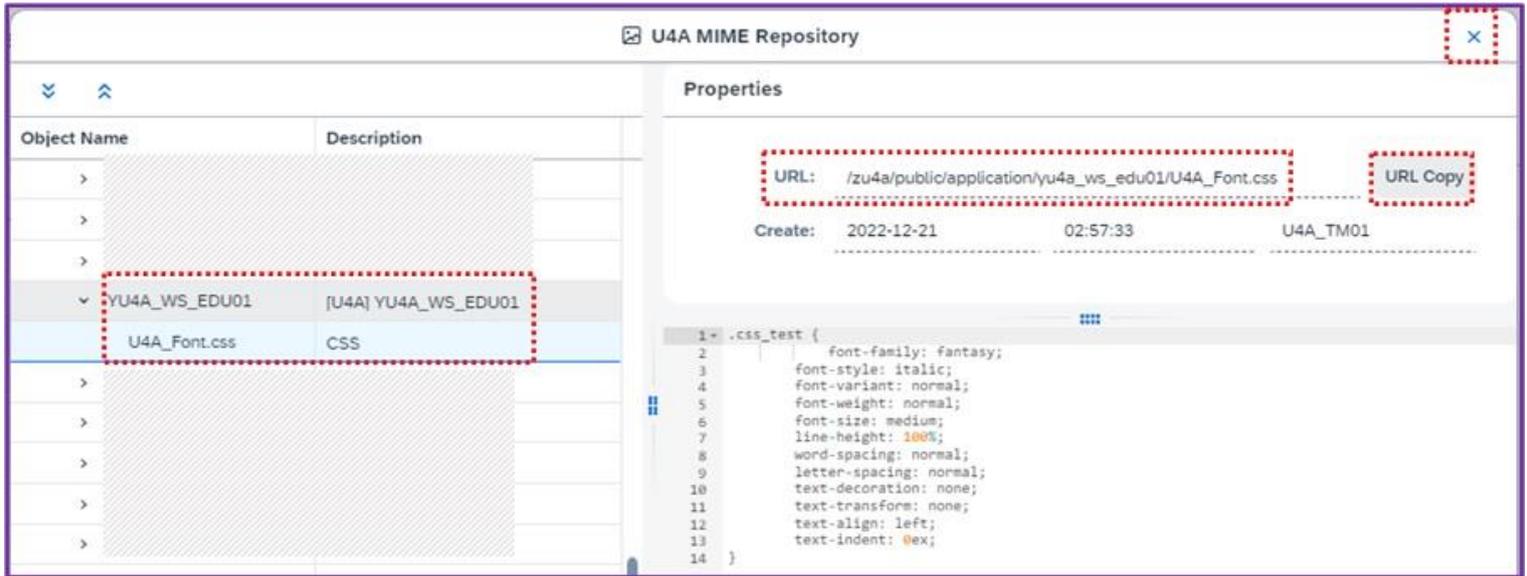
6) (2)항에서 저장된 CSS File을 선택함.



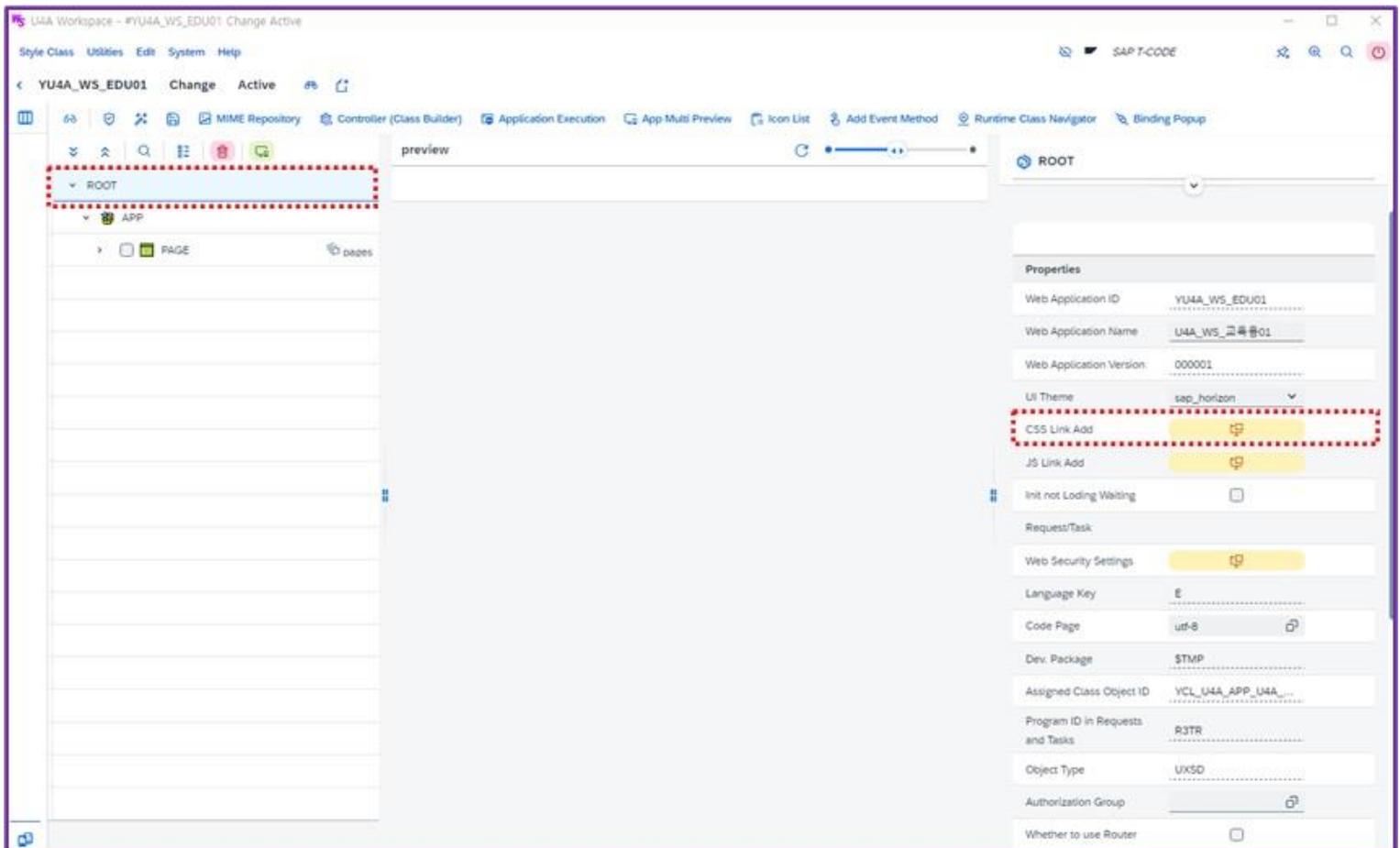
7) 해당 CSS File이 정상적으로 Load 되었다면 Description을 입력하고 "Save" Button을 Click함.



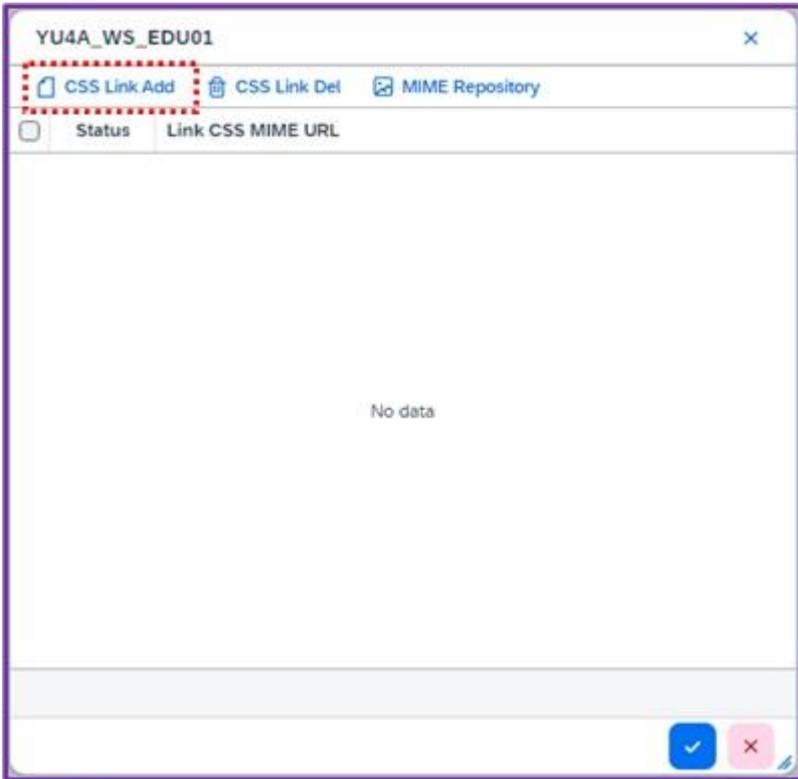
8) 정상적으로 저장되면 표시된 영역처럼 화면이 갱신됨. "URL Copy" Button을 Click한 후 현재 Popup을 종료함.



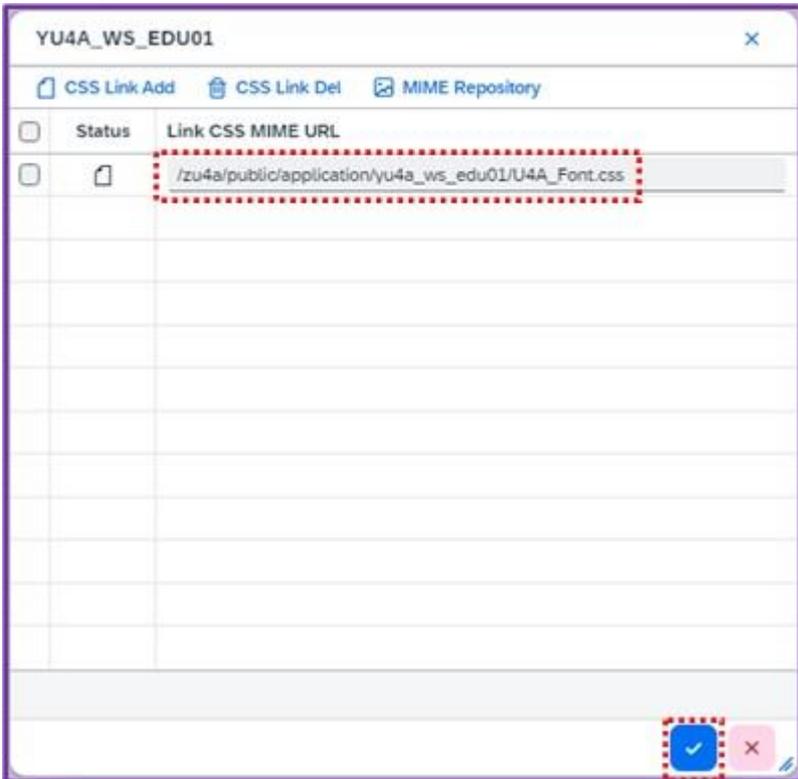
9) Workbench "UX Design Area" 내 최상위 Node "ROOT"를 Click 한 후 우측 속성 Panel 항목 중 "CSS Link Add"를 Click함.



10) 표시된 Popup에서 "CSS Link Add" Button을 Click함.



11) (8)항에서 복사된 MIME URL을 추가된 Row에 붙여 넣기 후 “확인” Button을 Click하여 등록 종료함.



5.1.1.3 Font Style Wizard 3

2. U4A IDE “StyleClass editor” 이용하여 CSS Code 등록 방법

1) CSS Font Wizard 화면에서 원하는 속성을 설정 후 얻어진 “CSS Source” 영역의 Code를 마우스로 Block Copy 함.

단 “CSS Source”에서 code 복사 시 예시와 같이 “p.sample(...)” 영역의 CSS Code 값만 복사 해야 함.

```

p.sample {
font-family: fantasy;
font-style: italic;
font-variant: normal;
font-weight: normal;
font-size: medium;
line-height: 100%;
word-spacing: normal;
letter-spacing: normal;
text-decoration: none;
text-transform: none;
text-align: left;
text-indent: 0ex;
}
    
```

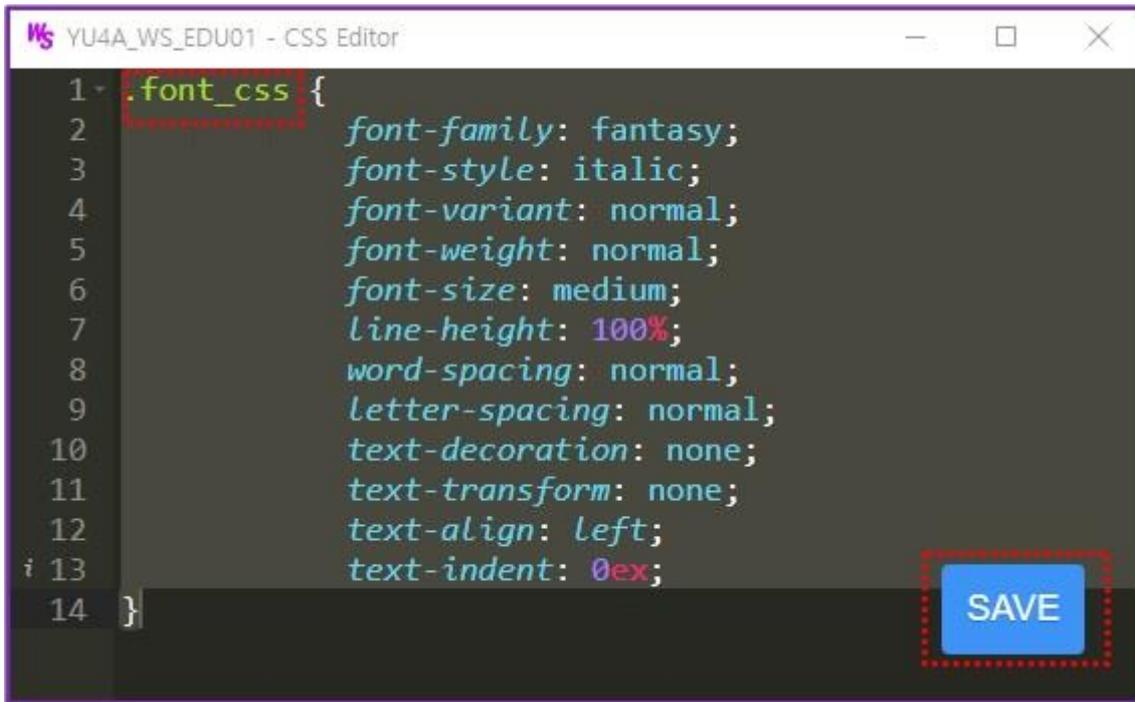
※ “p.sample” CSS Class Code

마우스 Block Copy!

2) Workbench 추가기능 메뉴 “Edit -> StyleCSS editor” Click함

3) 활성화된 “Editor - Css” Popup에서 (1)항에서 복사된 CSS 소스를 붙여 넣는다. 단, 표시된 것과 같이 CSS Class ID를

“p.sample” -> “.font_css”로 변경 후 저장 해야 함.



The screenshot shows a window titled "YU4A_WS_EDU01 - CSS Editor". The main content is a CSS rule for ".font_css" with the following properties: font-family: fantasy; font-style: italic; font-variant: normal; font-weight: normal; font-size: medium; line-height: 100%; word-spacing: normal; letter-spacing: normal; text-decoration: none; text-transform: none; text-align: left; text-indent: 0ex;. A blue "SAVE" button is located in the bottom right corner of the editor area. Red dashed boxes highlight the ".font_css" selector and the "SAVE" button.

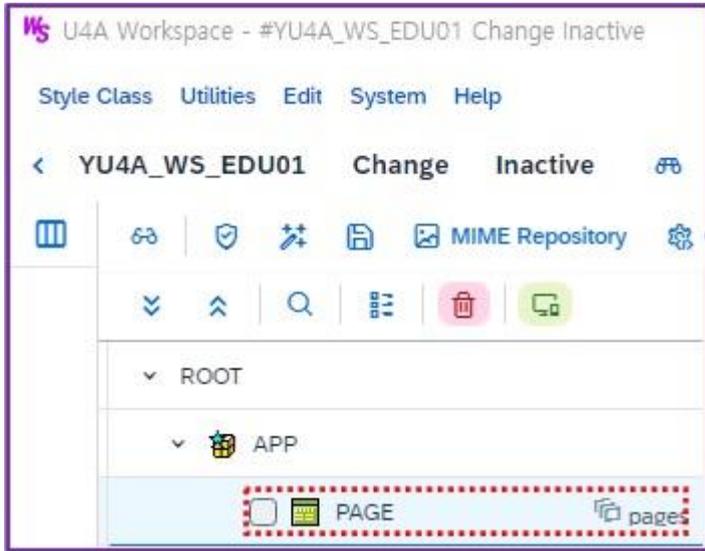
```
1 .font_css {  
2     font-family: fantasy;  
3     font-style: italic;  
4     font-variant: normal;  
5     font-weight: normal;  
6     font-size: medium;  
7     line-height: 100%;  
8     word-spacing: normal;  
9     letter-spacing: normal;  
10    text-decoration: none;  
11    text-transform: none;  
12    text-align: left;  
13    text-indent: 0ex;  
14 }
```

 단, “CSS Source” 에서 Code 복사 시 예시와 같이 “p.sample{...}” 영역의 CSS Code 값만 복사 해야 함.

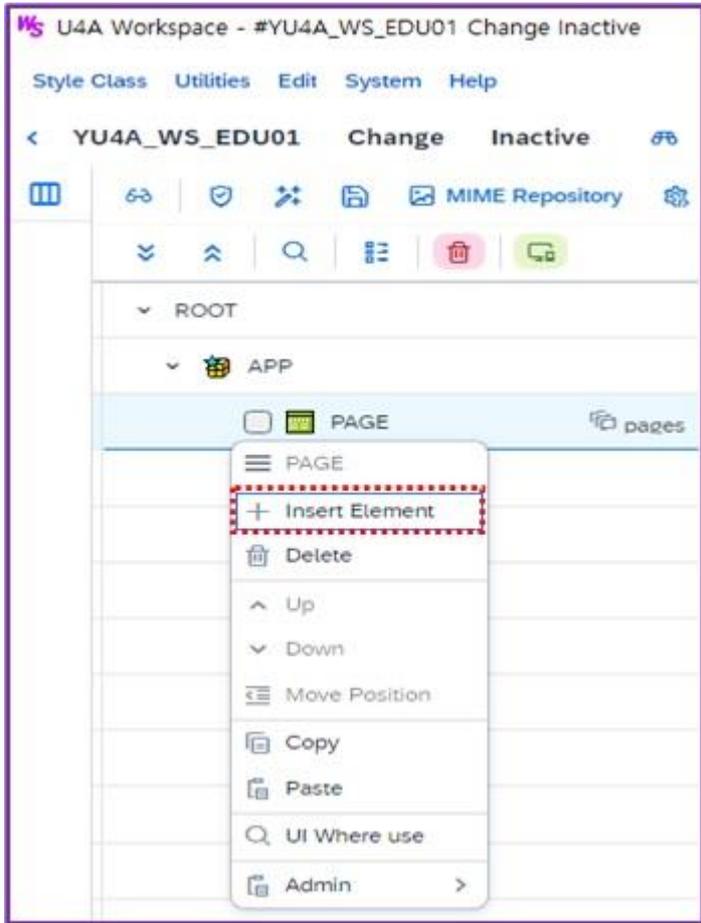
5.1.1.4 Text UI에 Font Style CSS 적용 Test 1

Test를 하기 위하여 “Text” UI를 다음과 같이 등록/추가 함. (※ (1)항 ~ (6)항 총 3회 진행 필요)

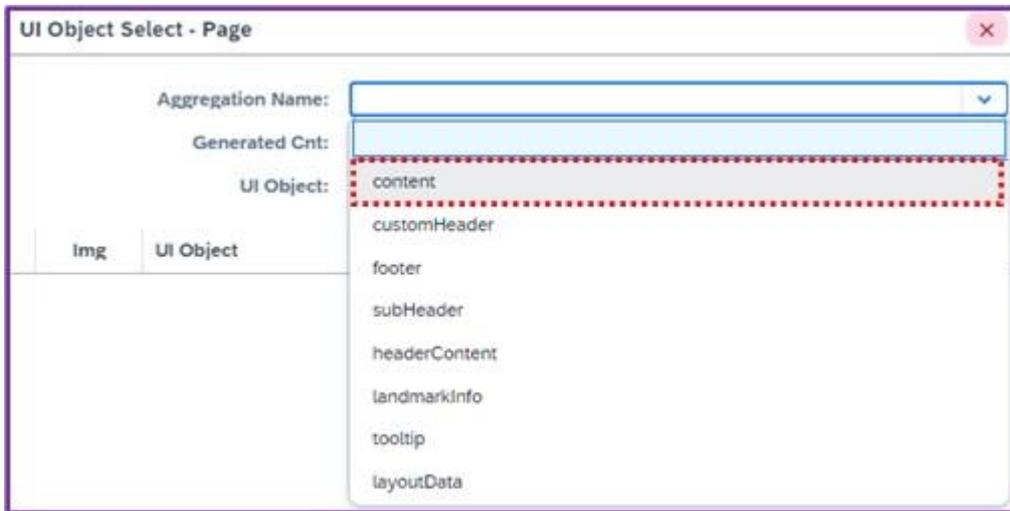
1) Workbench “좌측 “UX Design Area” Tree 영역 내 “PAGE” UI Object를 선택함.



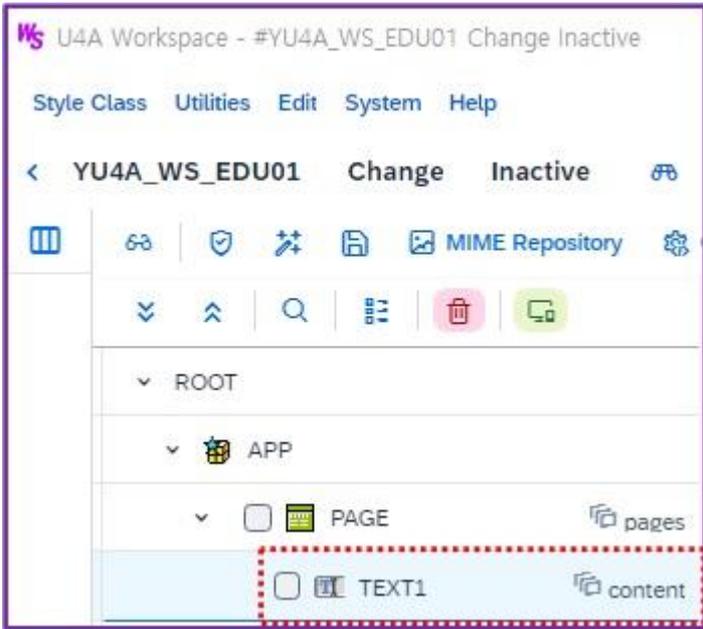
2) 선택한 “PAGE” UI Object를 마우스 우 Click하면 Menu Panel이 보여 지며, 항목 중 “Insert Elements”를 마우스로 선택함.



3) 활성화된 "UI Object Select" Popup 내 "Aggregations" List 항목 중 "content"를 선택함.

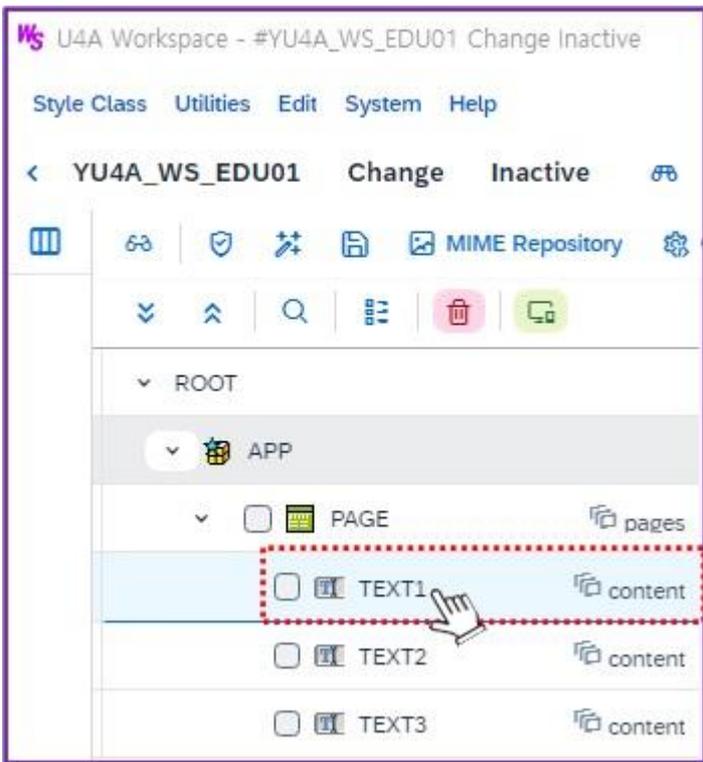


4) 표시된 영역의 "UI Object" 입력란에 "text"란 Keyword를 입력 후 Enter.

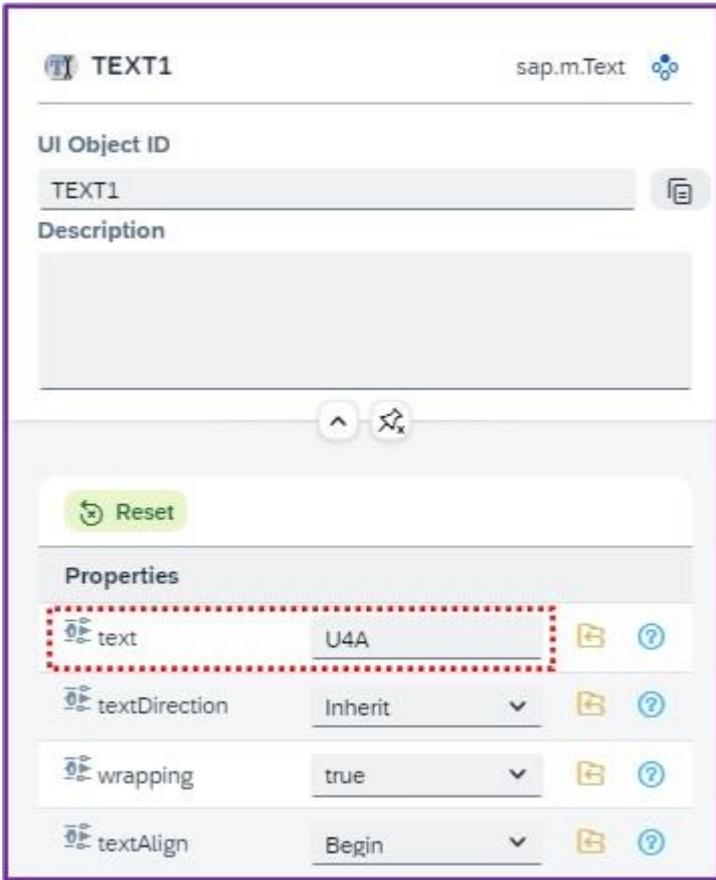


<--- 여기까지 총 3회 진행 필요

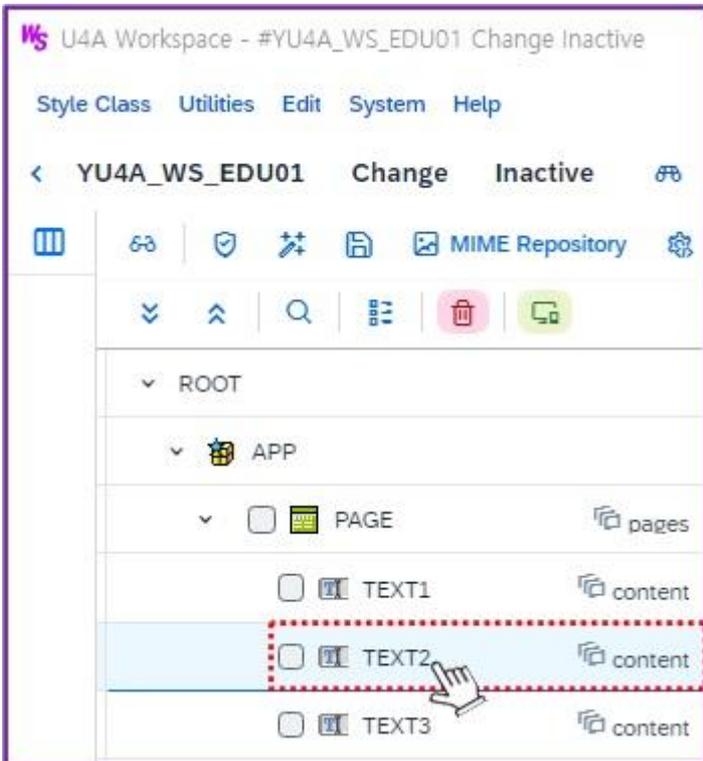
7) 표시된 것과 같이 "TEXT1" Object를 선택함.



8) "Properties"항목 중 "text"에 "U4A"를 입력함.



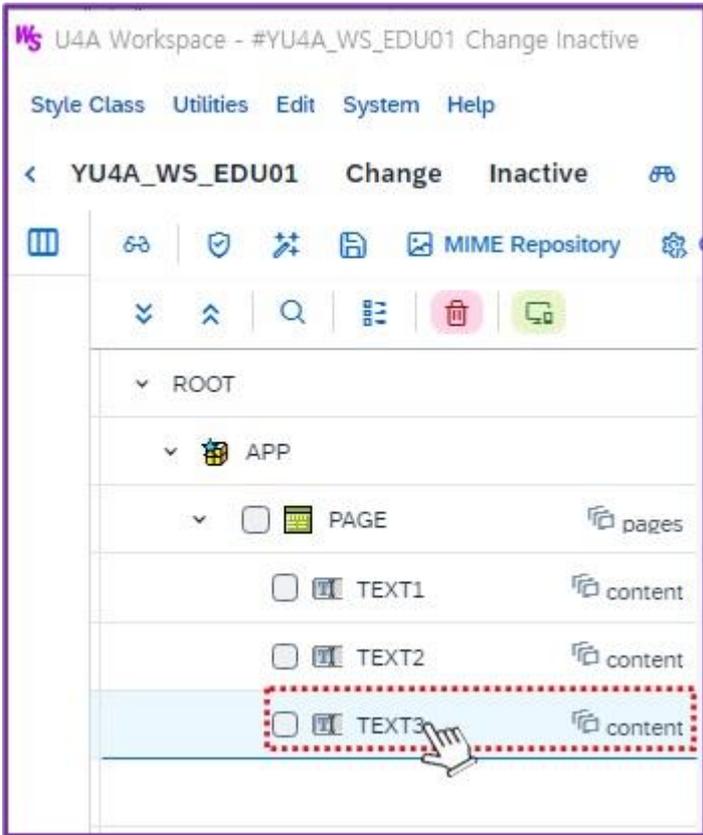
9) 표시된 것과 같이 "TEXT2" Object를 선택함.



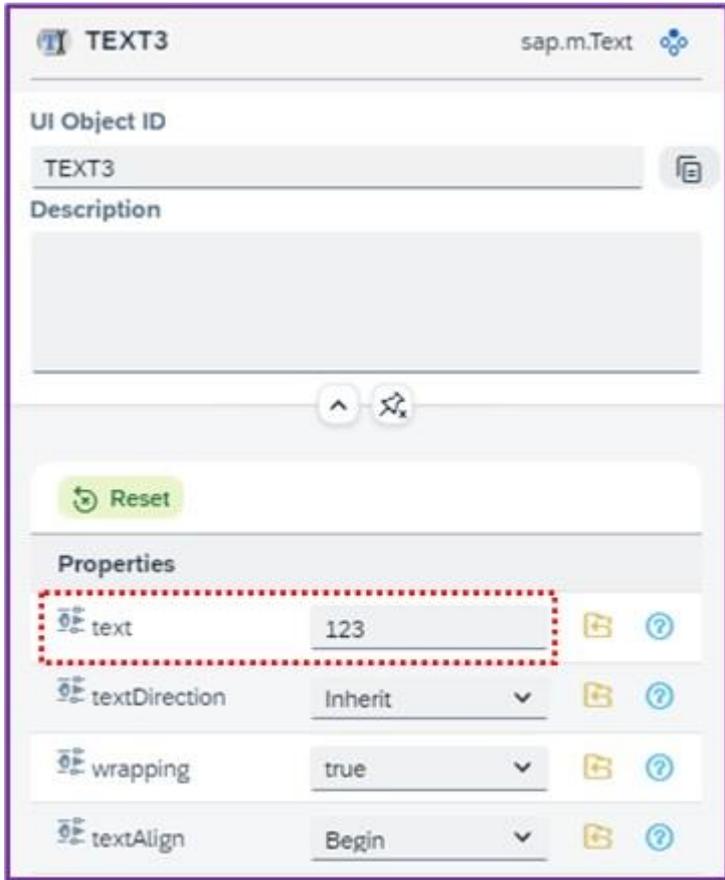
10) "Properties"항목 중 "text"에 "유포에이"를 입력함.



11) 표시된 것과 같이 "TEXT3" Object를 선택함.



12) "Properties"항목 중 "text"에 "123"를 입력함.



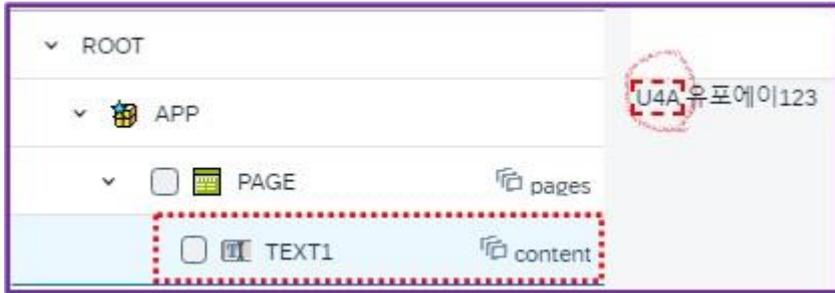
13) 3개의 Text UI Object에 기본 Style로 출력됨.



5.1.1.5 Text UI에 Font Style CSS 적용 Test 2

1. "MIME Repository"에 등록된 "fontcss.css" Style Sheet로 적용 하는 방법

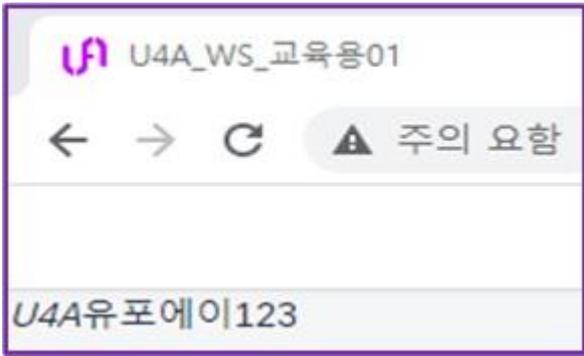
- 1) "[Font Style Wizard2](#)" 단원에서 소개된 내용을 참고하여 Style Sheet를 등록함.
- 2) 그림에서 표시된 것과 같이 "UX Design Area" 또는 "미리보기 영역"에서 "TEXT1" UI Object를 선택함.



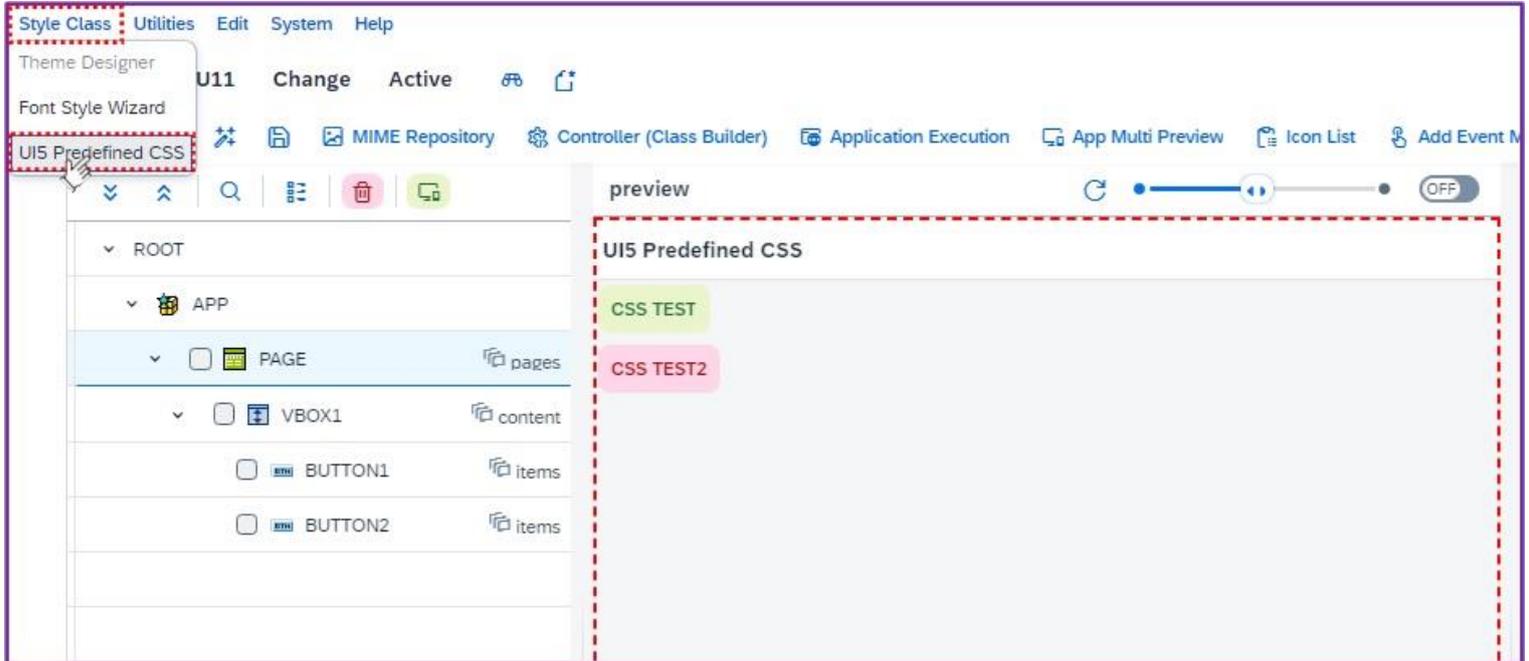
- 3) 우측에 표시된 Attributes 내역 중 "styleClass" Properties에 "css_test" 라고 입력함. (※ 대/소문자 입력 주의)

Properties		
text	U4A	
textDirection	Inherit	
wrapping	true	
textAlign	Begin	
width		
maxLines		
busy	false	
busyIndicatorDelay	1000	
visible	true	
tooltip		
busyIndicatorSize	Medium	
renderWhitespace	false	
wrappingType	Normal	
emptyIndicatorMode	Off	
styleClass	css_test	
dragAble	false	
dropAble	false	

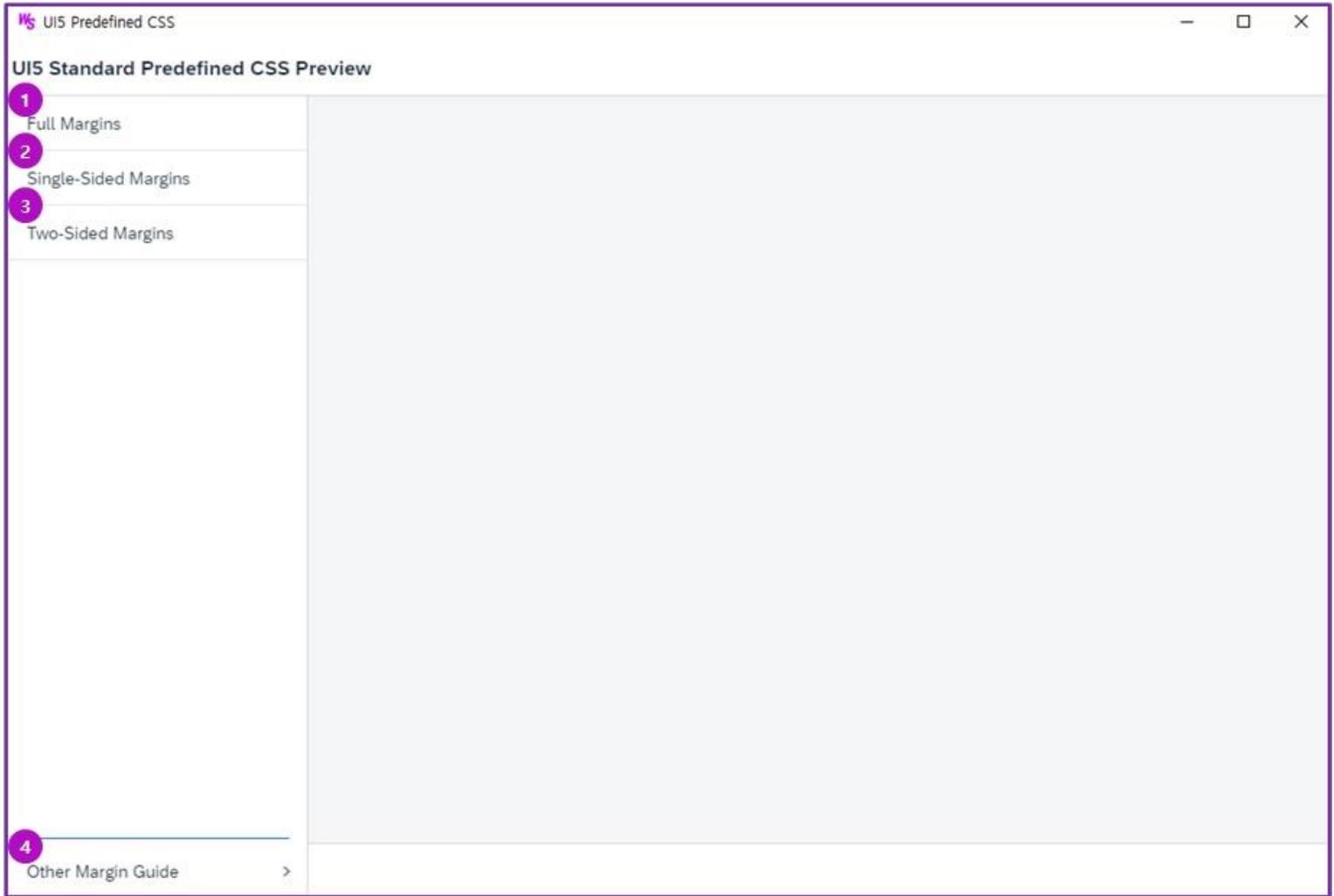
- 4) 입력이 완료되면 Main Toolbar 영역 내  "Active" Button(Ctrl+F3)을 Click하여 Application을 활성화 함.
- 5) Main Toolbar 영역 내  "Application Execution" Button(F8)을 Click하여 실행함.
- 6) 그림과 같이 실행된 Browser 내 출력된 Text 내용 중 Style Class가 지정된 "TEXT1" UI Object Text값 "U4A"가 미적용된 Text와 다른 글꼴 및 "Italic" 효과 확인 가능.
즉, Style Sheet를 다양하게 만들어 각각 UI별 Style Class 다르게 지정 가능.



5.1.2 UI5 Predefined CSS



① UI5 Predefined CSS 실행 시 첫 Pop-Up 화면



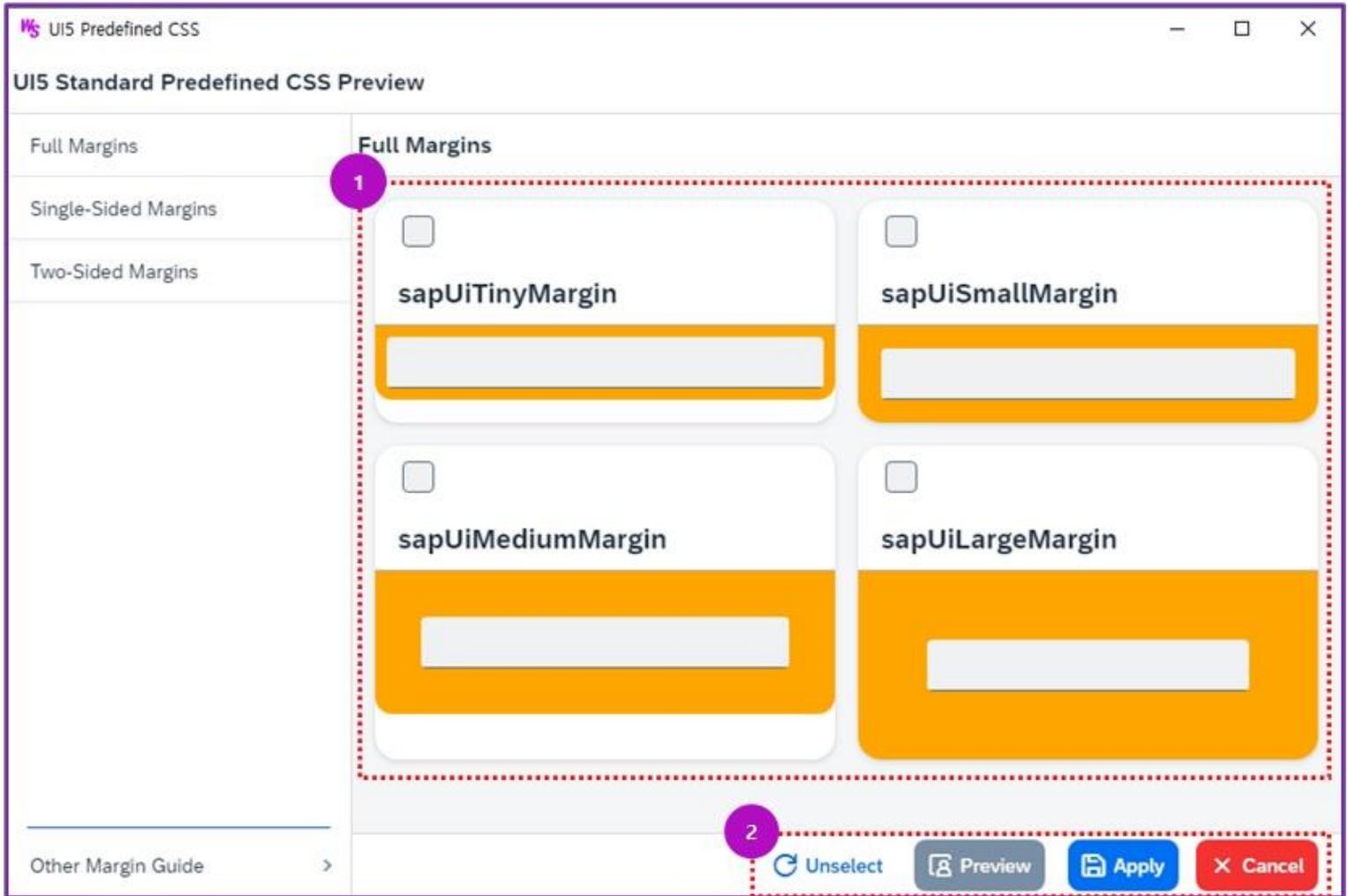
1) Full Margins : UI의 Top, Bottom, Begin, End에 모두 Margin 적용

2) Single-Sided Margins : UI의 Top, Bottom, Begin, End 중 "한 지점"에 Margin 적용

3) Two-Sided Margins : UI의 "두 지점"에 Margin 적용 가능 ("Top and Bottom" 또는 "Begin and End" 만 가능하다)

4) Other Margin Guide : 1,2,3 이외에 Margin 적용에 대한 가이드

② UI5 Predefined CSS 실행 시 상세 Pop-Up 화면

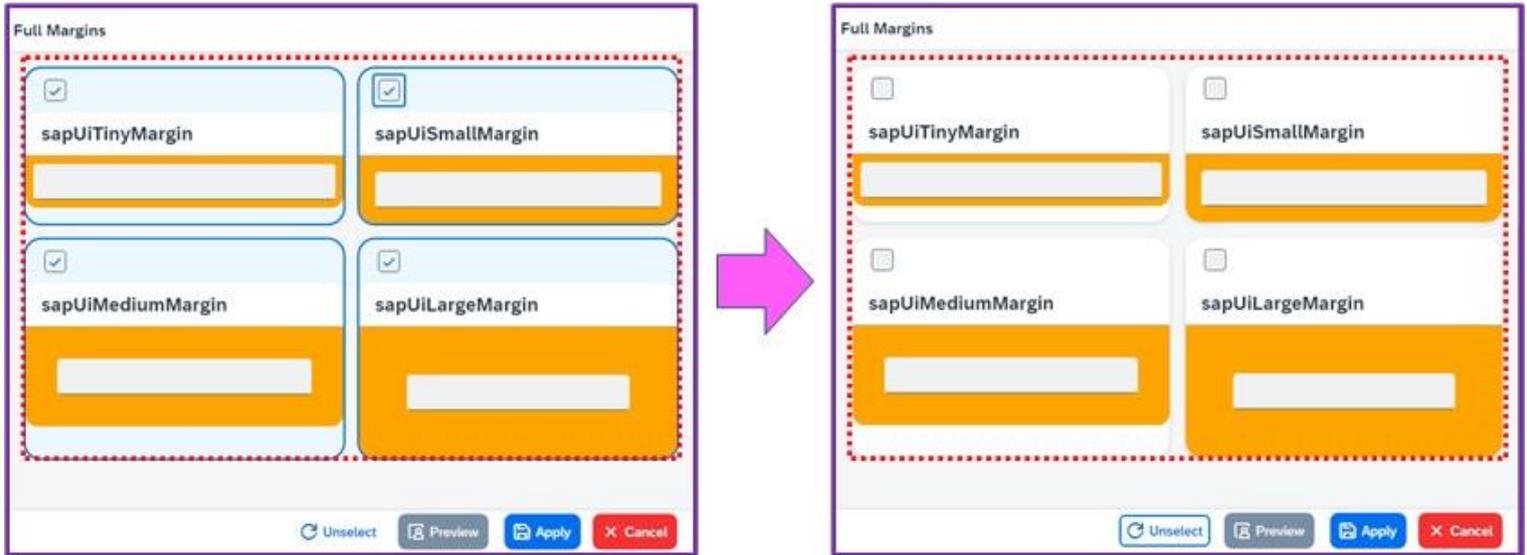


1) UI에 적용 가능한 Margin CSS 종류 및 적용 시 UI Margin 미리보기 Sample 영역.

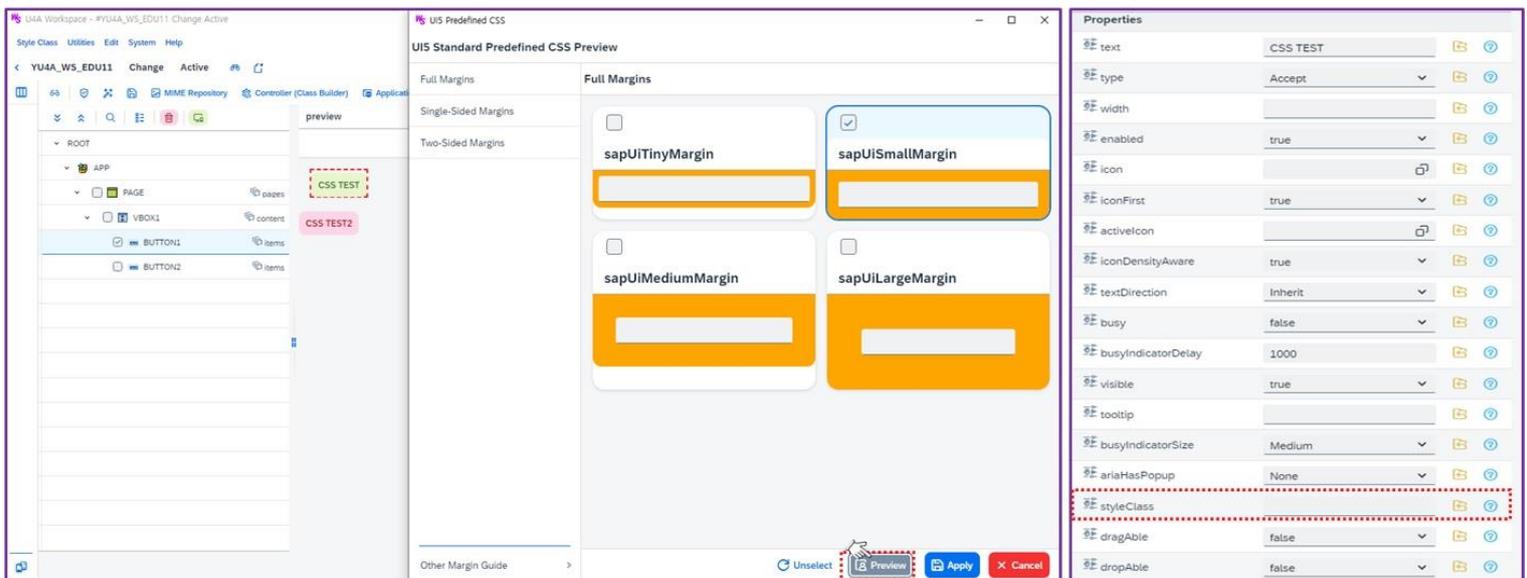
2) 하단 Footer 영역

"Candle" : UI5 Predefined CSS Pop-up 종료

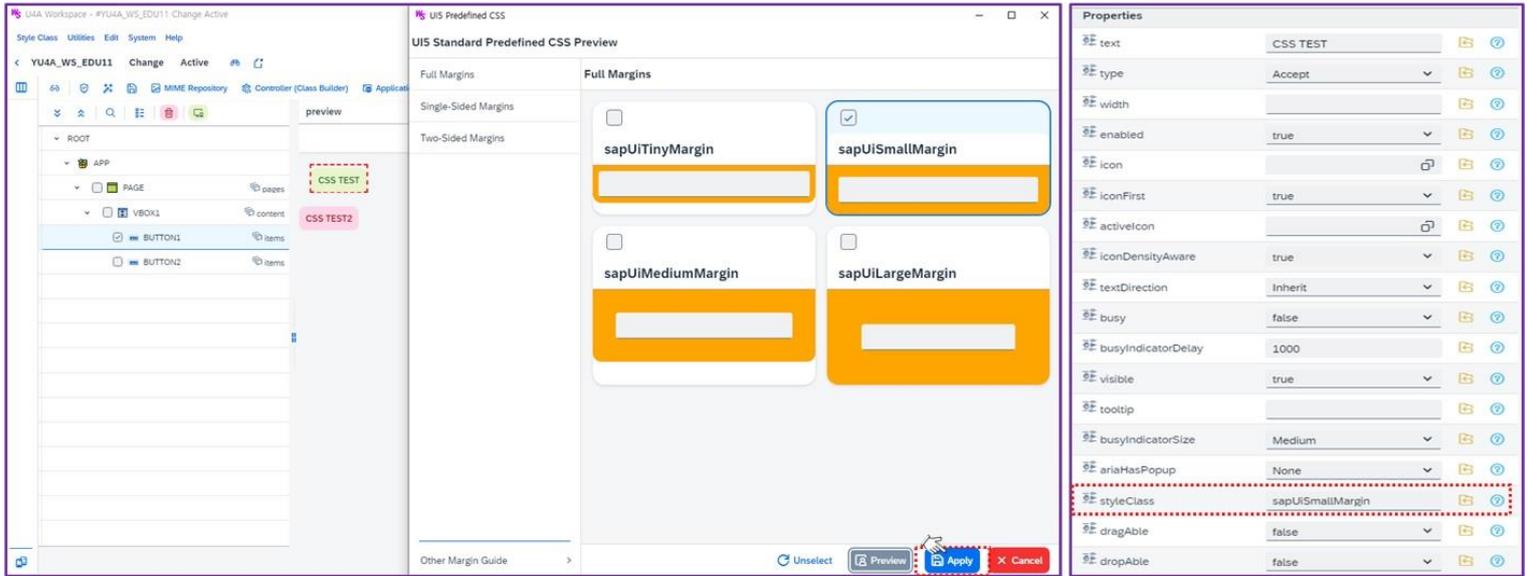
"Unselect" : 선택된 체크박스 모두 해제



"Preview" : 선택한 CSS 적용시 UI 변화 미리보기 (해당 UI의 style Class에는 적용이 되지 않음)



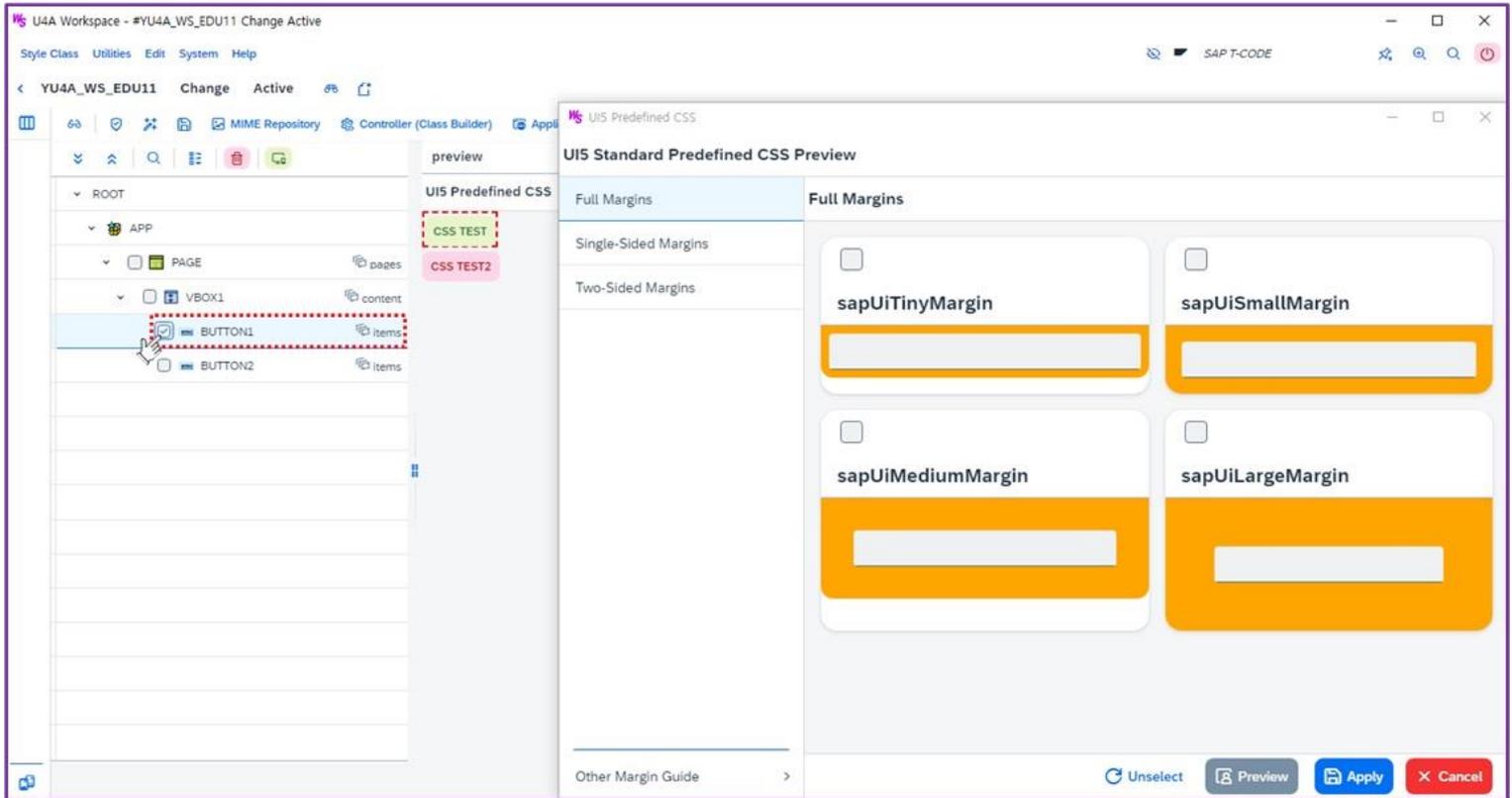
"Apply" : 선택한 CSS UI에 적용 (해당 UI의 style Class에도 선택한 CSS 적용)



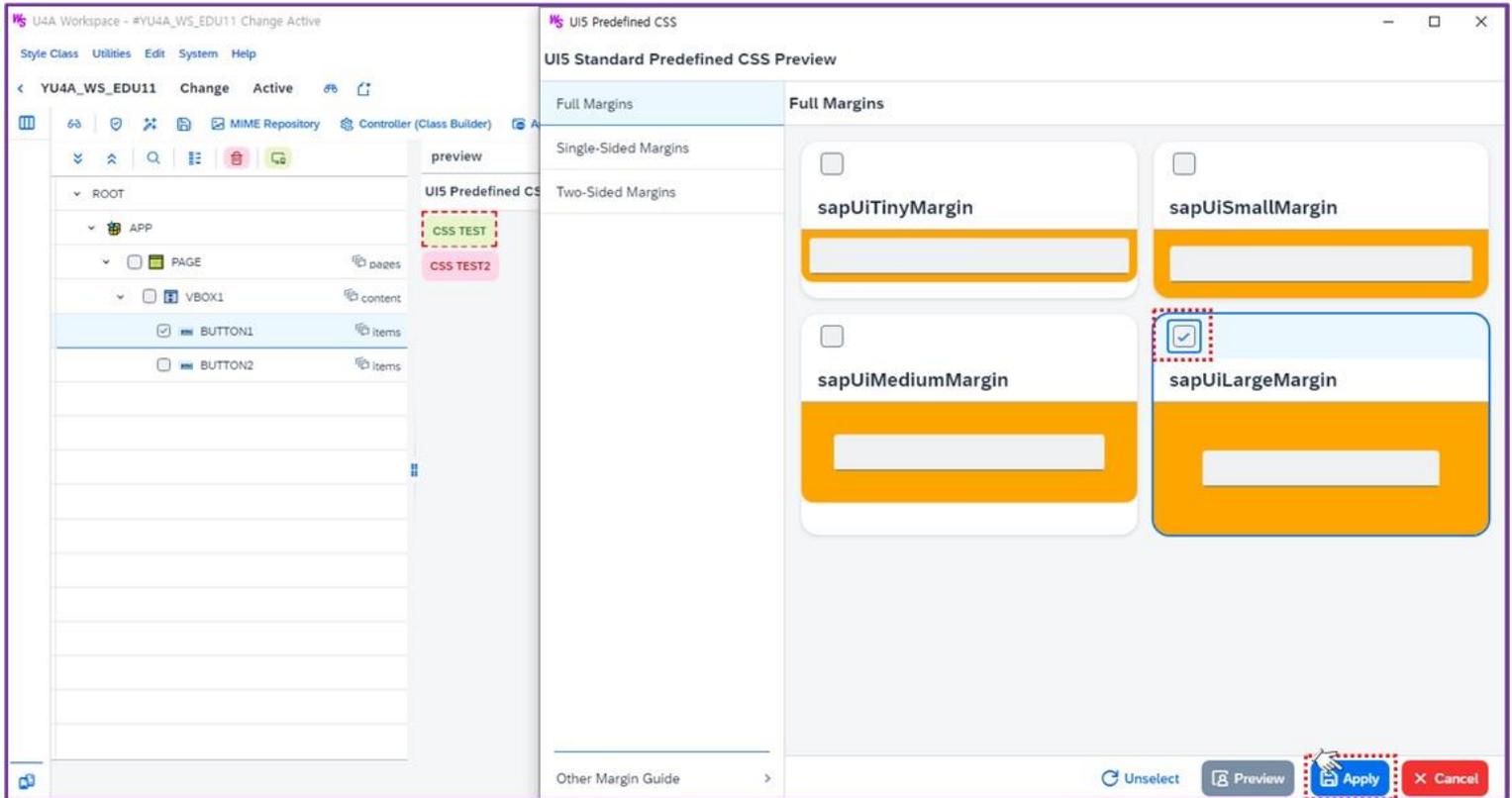
2. UI5 Predefined CSS를 UI에 적용하기.

※ 본 단원에서는 "Full Margins"을 이용하여 CSS TEST 버튼에 적용한다.

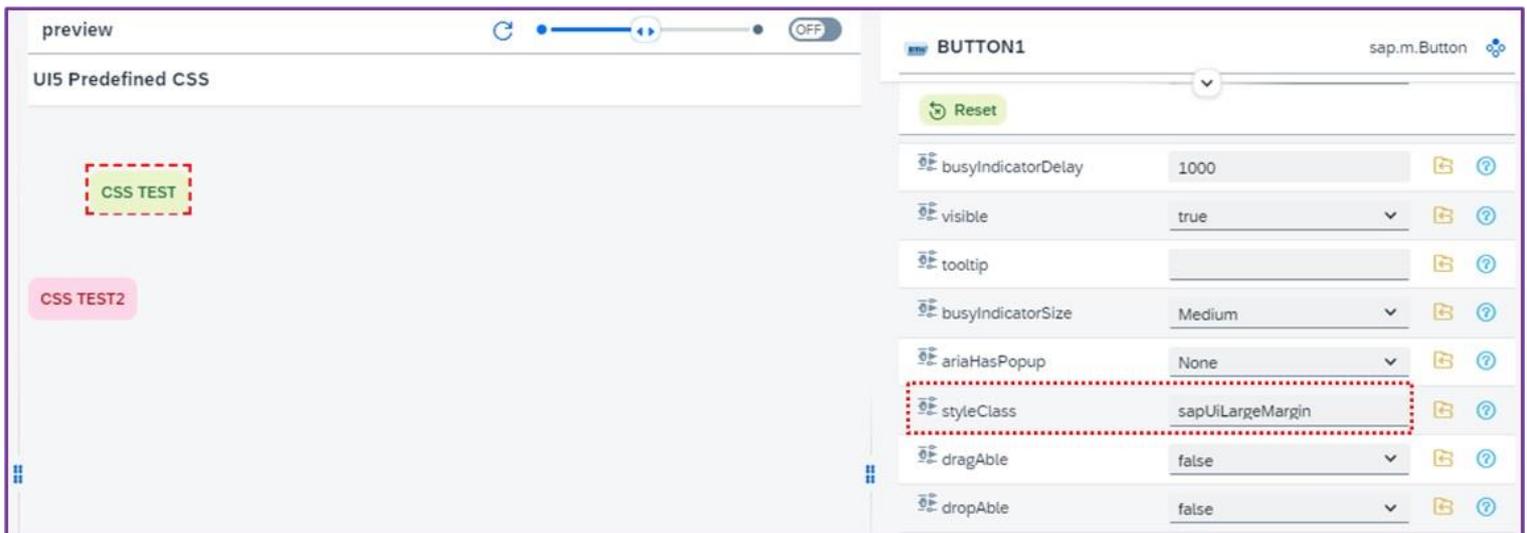
① Margin CSS 적용할 UI의 체크박스에 체크한다.



② 적용할 Margin CSS를 체크한 뒤 "Apply" Button을 클릭하고 Pop-Up Close



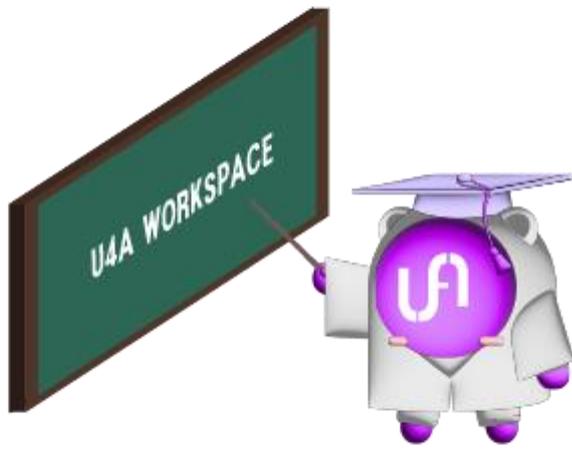
③ Preview 영역에서 Margin CSS 적용 확인 가능 + Properties의 styleClass에 "sapUiLargeMargin" 적용 확인



3. Application을 실행 하여 UI CSS 적용 확인.



5.2 추가 기능 메뉴 - Utilities

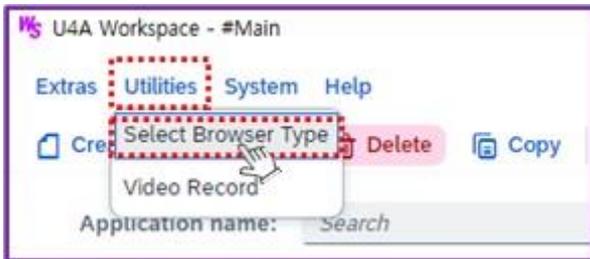


5.2.1 Select Browser Type

▣ 목적 : U4A Application 실행 시 기본 Browser를 지정 하기 위함.

1. 그림과 같이 추가 기능 메뉴 영역에서 "Utilities -> Select Browser Type"을 선택함. (선택 방법은 2가지 이지만 사용방법은 동일하다)

① U4A WorkSpace 메인 화면에서 선택

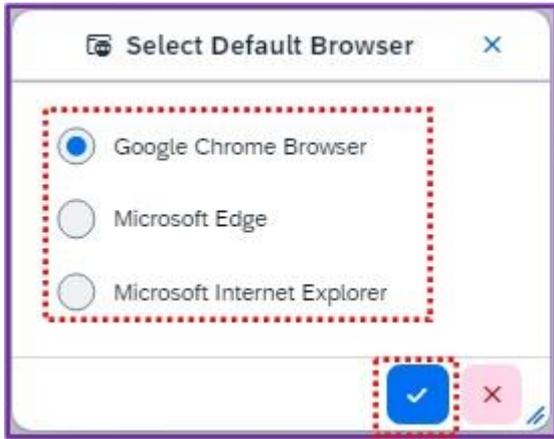


② Application 안에서 선택

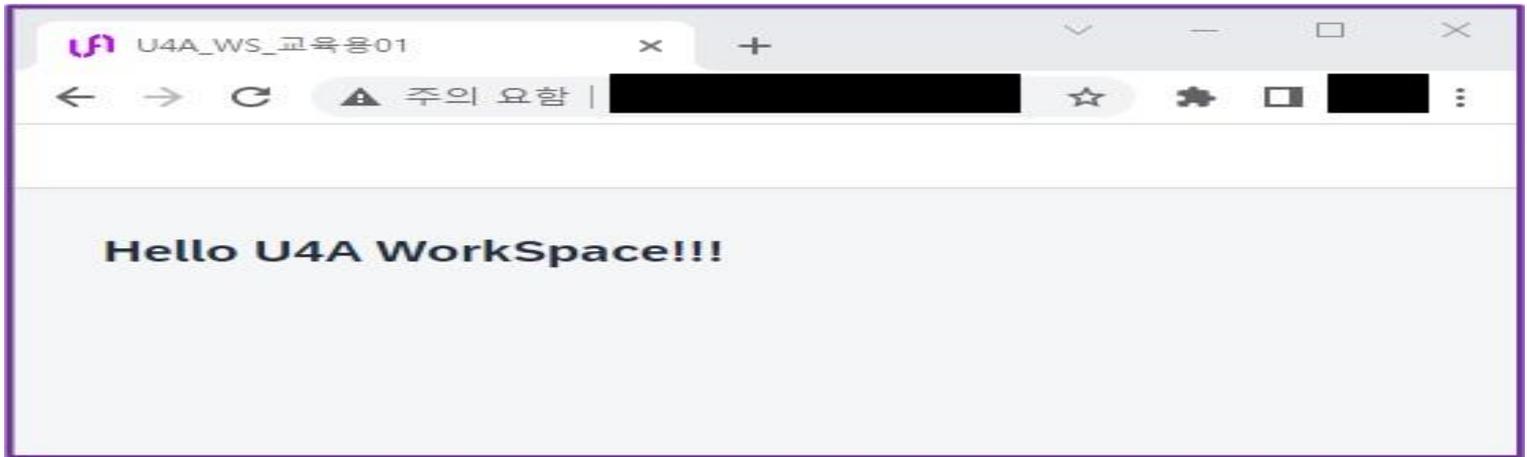


2. 활성화된 Popup 에서 표시된 Browser 중 기본 Browser 설정을 원하는 것을 선택 후 "√" Button을 클릭함.

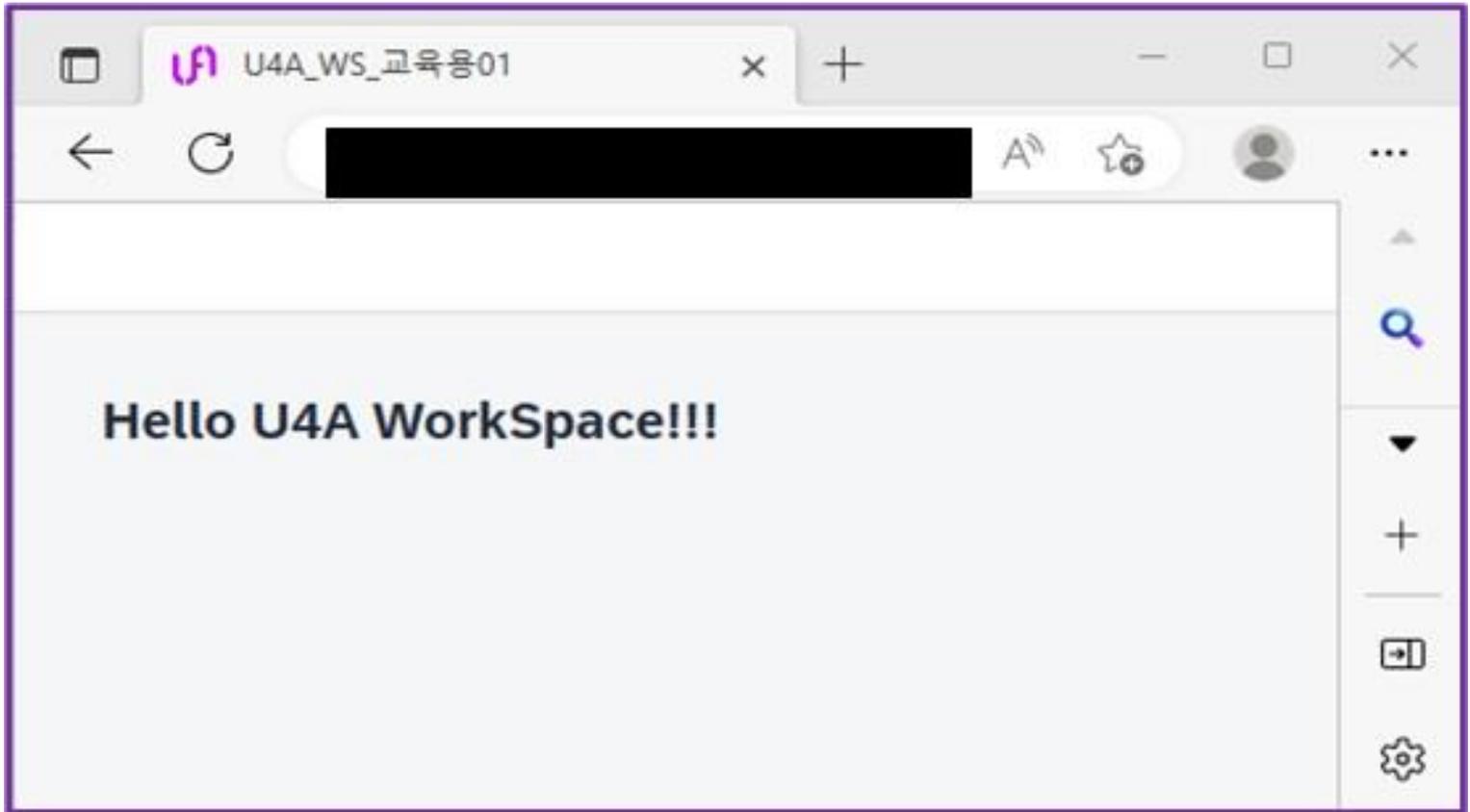
초기 Default Browser => "Google Chrome Browser".



2-1 "Google Chrome" Browser로 실행된 경우



2-2 "Microsoft Edge" 로 실행된 경우



2-3 "Microsoft Internet Explorer" 로 실행된 경우

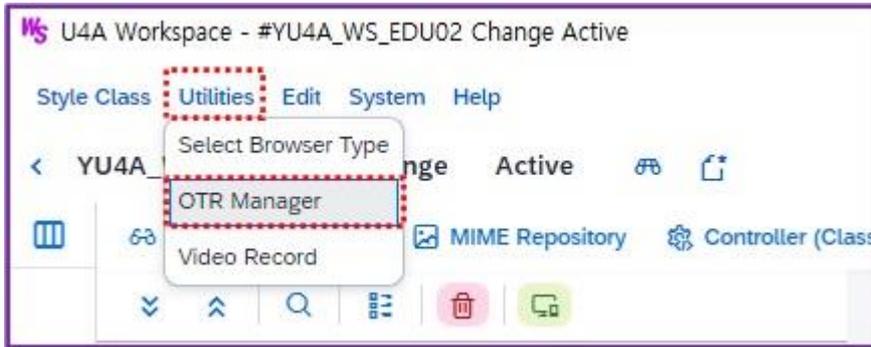
즉, 해당 Application의 Cross Browsing 테스트 시 실행 URL 복사 후 각각의 Browser 수동 실행 및 복사된 URL을 붙여넣기 후 실행 하는 번거로운 작업을 대체하는 편리한 기능.

단, 한번 기본 Browser로 지정되면 별도 다른 Browser로 변경 하기 전 까지 계속해서 해당 속성이 지속됨. (PC를 재부팅 해도 지속됨)

5.2.2 OTR F4 Request

■ 목적 : 다양한 언어 설정에 맞는 OTR Search 목적.

1. 상단의 Utilities = > OTR Manager 선택함.



2) Pop-Up 형식으로 찾고자 하는 OTR 검색이 가능.



(1) Package, Language, Alias Name, OTR Text 조건으로 OTR 검색 가능

(2) 검색된 결과를 더블 클릭하여 Alias 복사

3) 검색 결과 리스트를 더블 클릭 했을때 Alias 복사 메시지 팝업

\$OTR:YPERF01/U4A000001
copied.

4) 복사한 Alias를 Properties에 등록함.

위와 같이 검색을 통해 복사 등록 했다면 \$OTR: 이 자동으로 생성되어 OTR임을 표시.

사용자가 직접 Properties에 입력하여 등록한다면 반드시 \$OTR: 을 앞에 입력하여 OTR임을 표시해야 함.

Properties			
backgroundDesign	Translucent	▼	🗑️ ?
fixedLayout	true	▼	🗑️ ?
showOverlay	false	▼	🗑️ ?
inset	false	▼	🗑️ ?
headerText	\$OTR:YPERF01/U4A000001		🗑️ ?
headerDesign	Standard	▼	🗑️ ?
footerText			🗑️ ?

5.2.3 Video Record

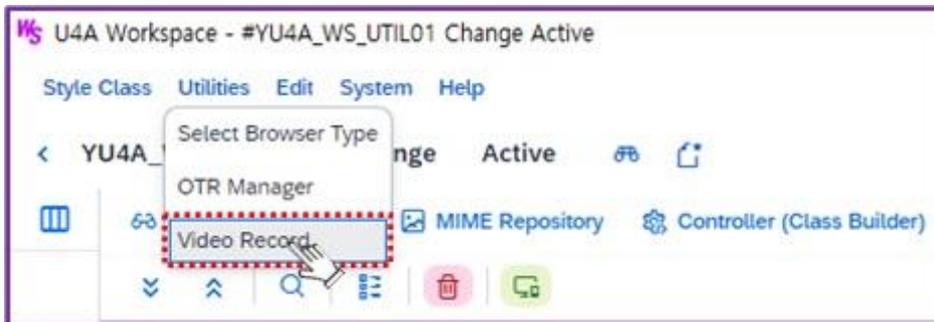
■ 목적 : U4A WorkSpace 사용 중 화면 녹화 기능.

1. 상단의 Utilities = > Video Record 선택함 (선택 방법은 2가지 이지만 사용방법은 동일하다)

① U4A WorkSpace 메인 화면에서 선택



② Application 안에서 선택



2. Recording Controller Pop-Up Open

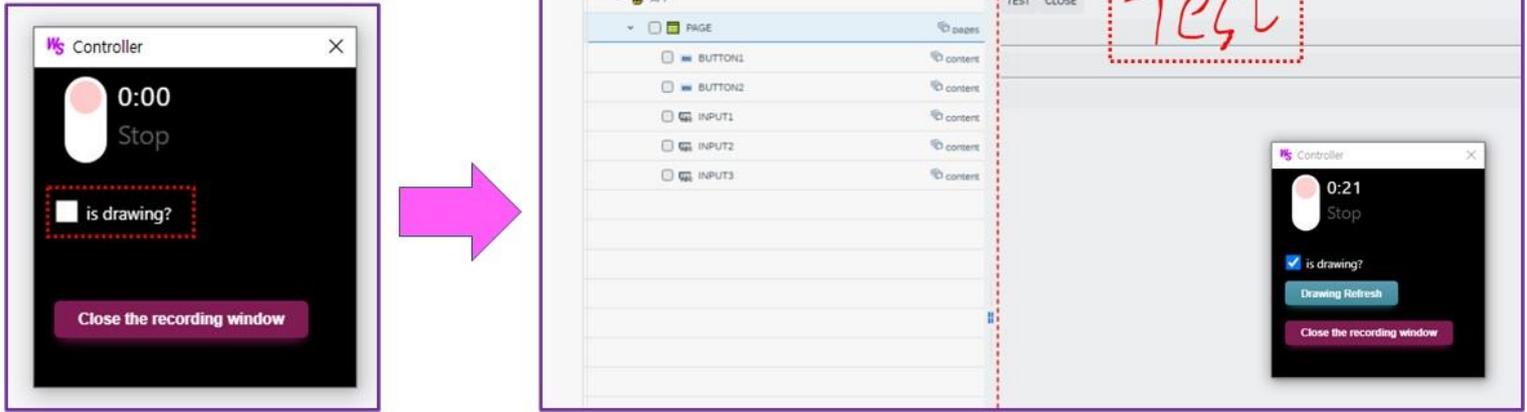


1) 스위치 전환을 통해 Record / Stop 설정

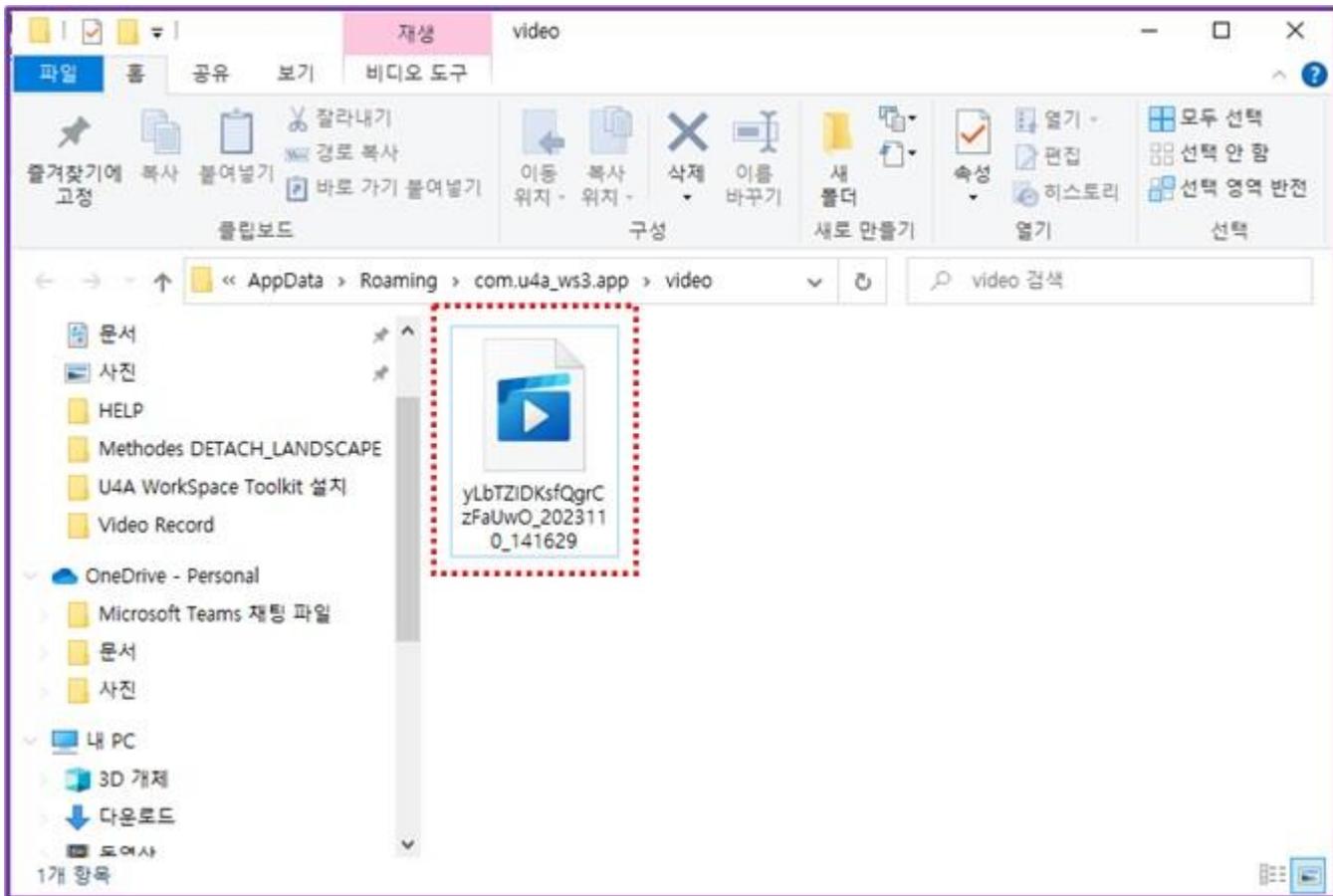
2) Recording Controller 팝업창 종료

3. (1) 스위치 "Record"로 전환하여 Recording 시작

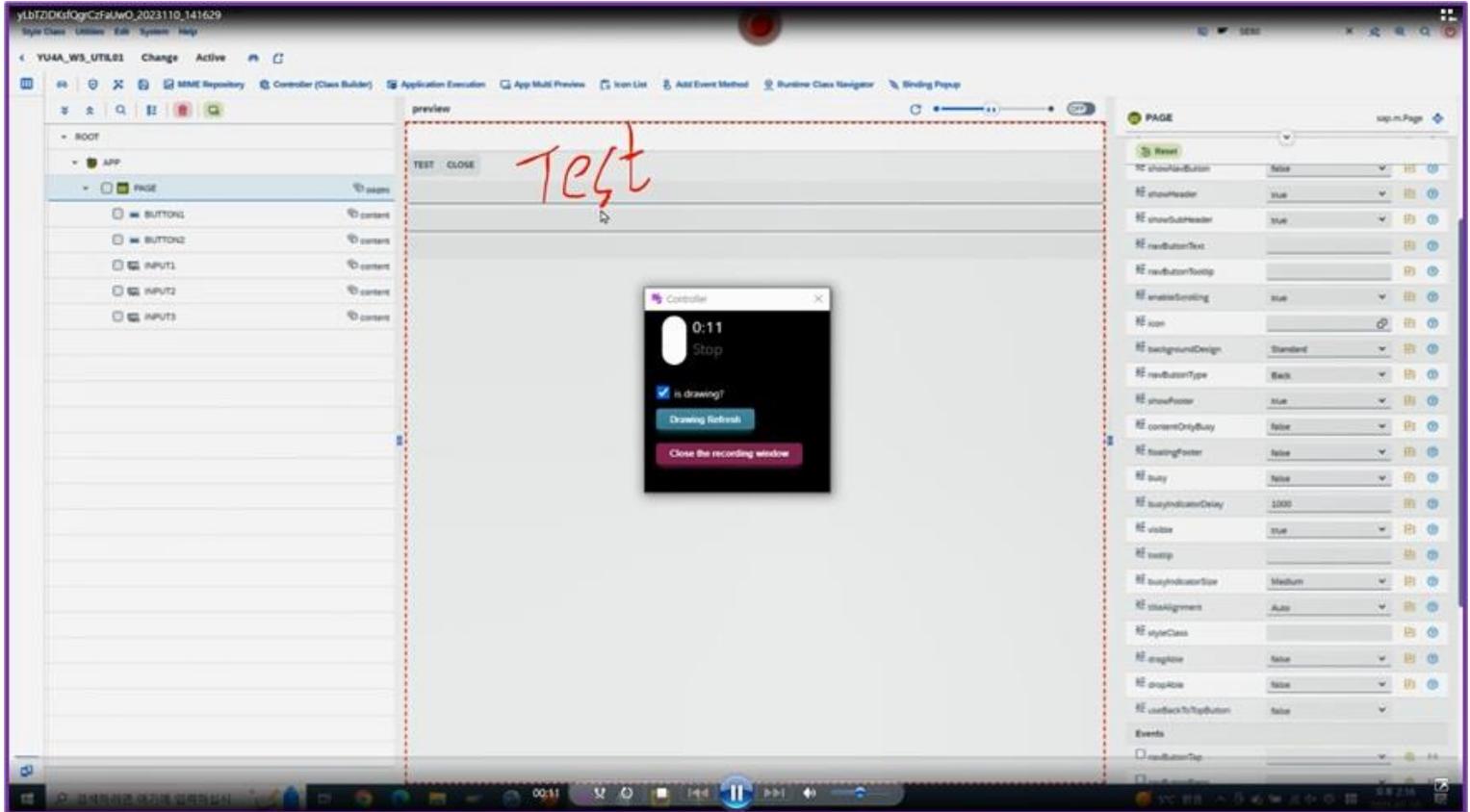
① Stop -> Record 변경시 "is drawing?" 체크박스에 체크시 녹화중 화면에 Drawing 가능



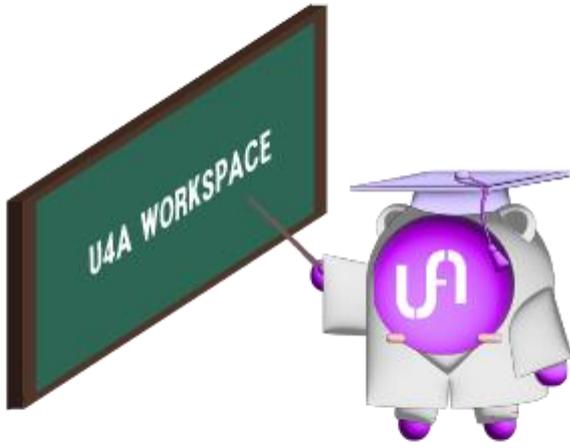
4. 다시 스위치 "Stop"으로 전환하면 녹화가 중지되며 녹화 Video 파일이 담긴 폴더 자동 실행



5. Video 파일 실행하여 녹화 화면 확인



5.3 추가 기능 메뉴 - Edit



5.3.1 StyleCSS Editor

▣ 목적 : 대상 Application 내 사용될 JAVA Script를 정의하는 외부 Editor 임.

1. Editor 실행은 Workbench 상단 추가 기능 메뉴 영역에서 "Edit -> StyleCSS editor" Click함.

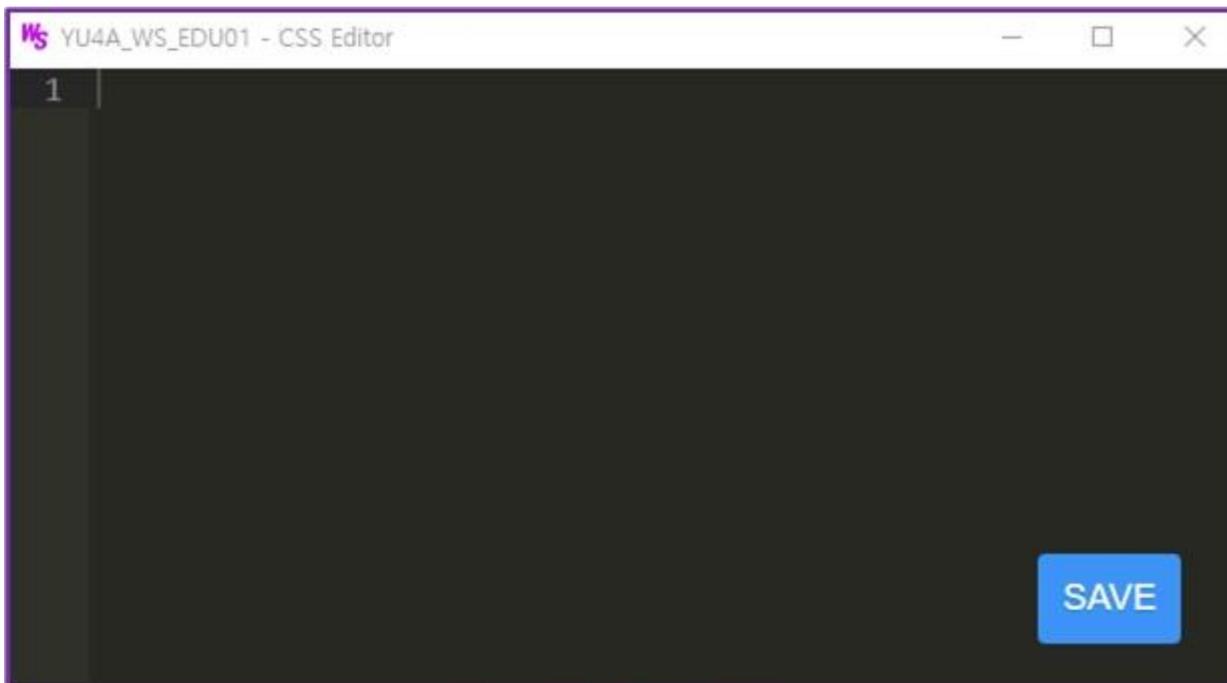
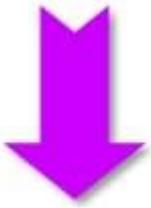
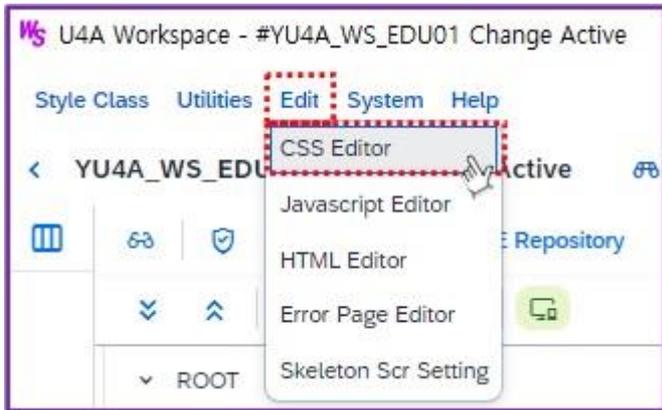
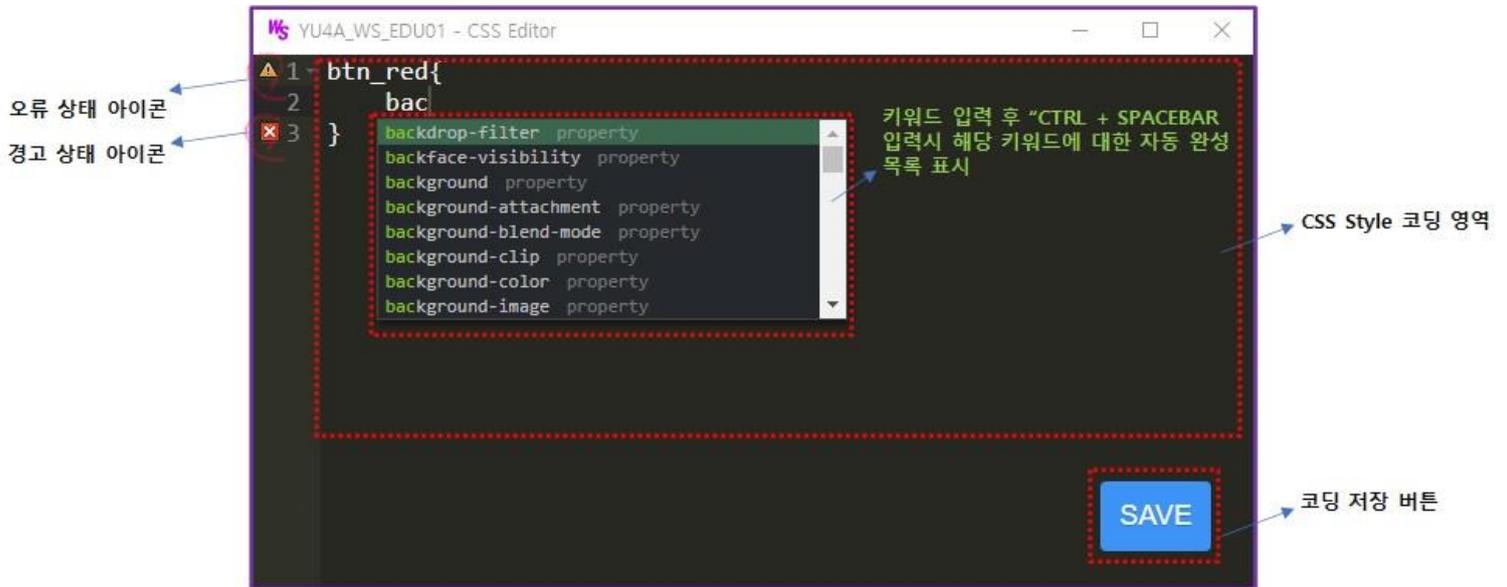


Figure : Style CSS Editor

2. Editor 기능 설명



3. 기능 테스트 예제

1) Editor 코딩 영역에서 다음과 같이 입력 함.

```
.btn_red{
background-color: red;
}
```

2) Editor Popup 우측 하단 "SAVE" Button을 Click하여 코딩 내용을 저장함.

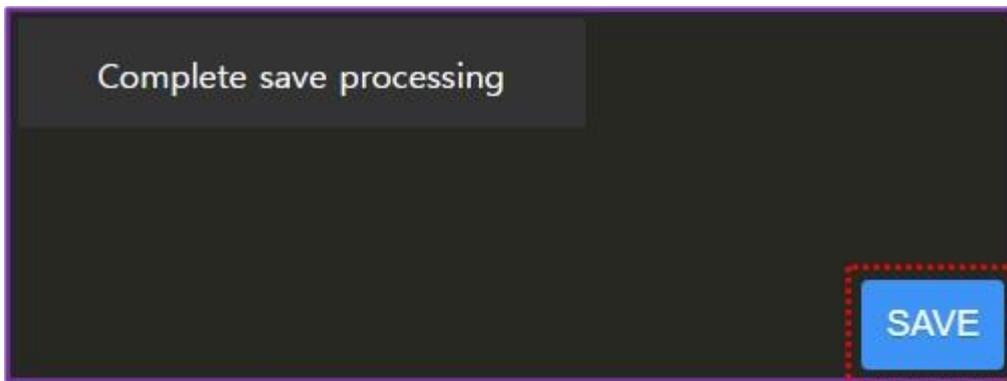
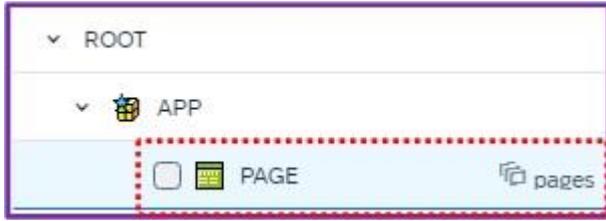
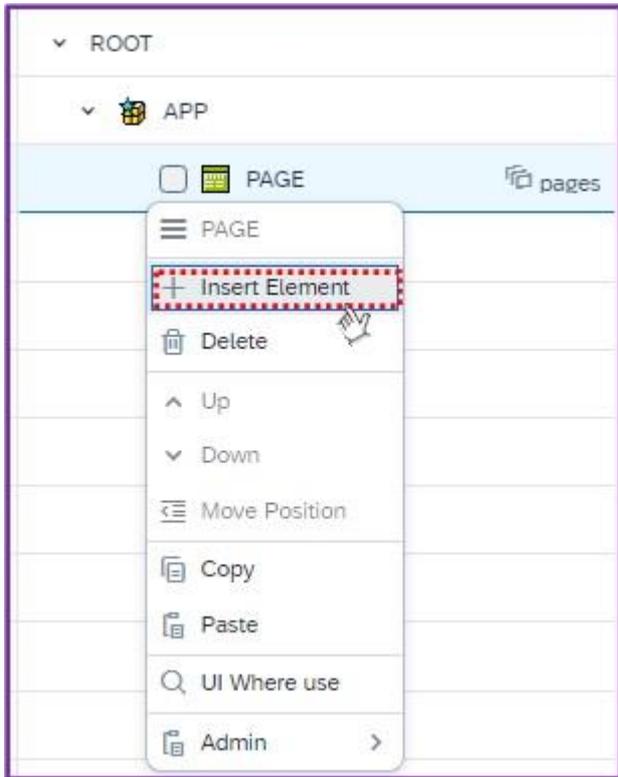


Figure : "SAVE" 버튼 클릭 시 처리에 대한 정상 메시지 표시

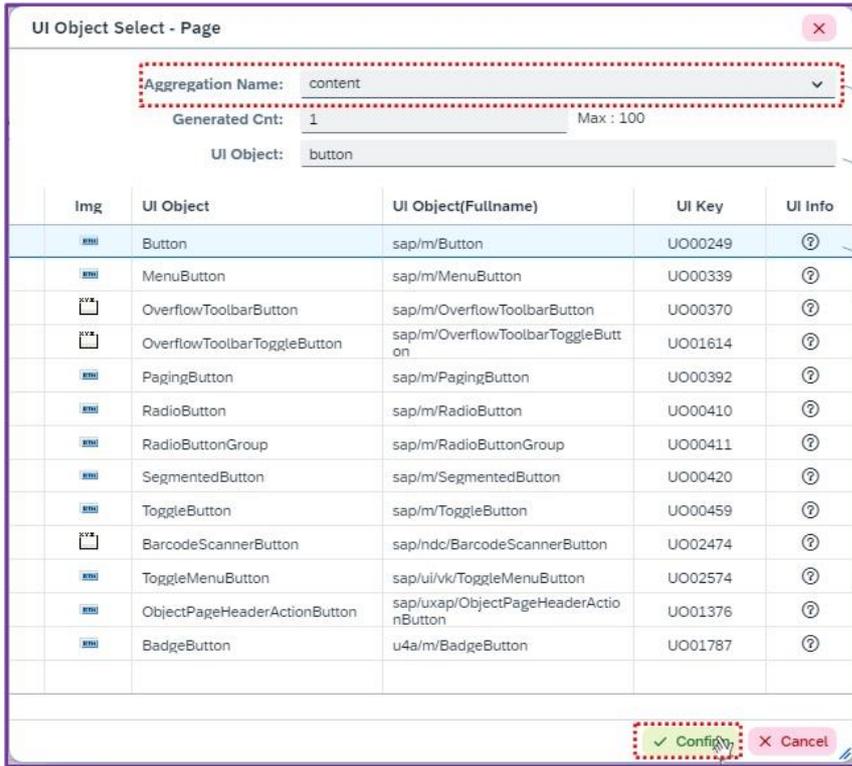
3) Workbench "UX Design Area"에서 "PAGE" Object를 선택함.



4) 선택한 "PAGE"Object에서 마우스 우클릭 후 메뉴 Panel에서 "Insert Elements"를 Click함.

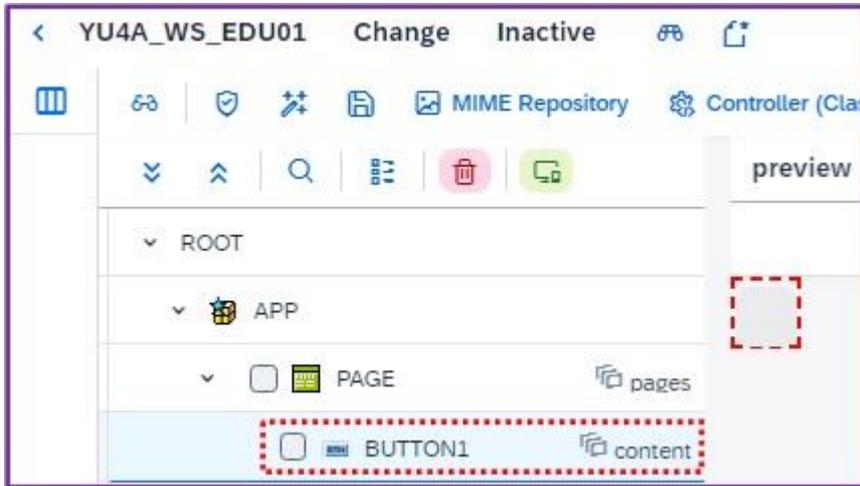


5) "UI Object Select" Popup에서 다음과 같이 처리함.



- (1) "Aggregation Name" 에서 "content" 선택
- (2) "UI Object" 에서 "button" 키워드 입력 후 Enter
- (3) 표시된 항목 중 "Button" 마우스로 선택 후 아래의 "Confirm" Click

6) (5)항에서 정상적으로 처리 되었다면 Workbench "UX Design Area" 및 "Preview Area"에 Button UI 한개가 등록됨을 확인할 수 있음.



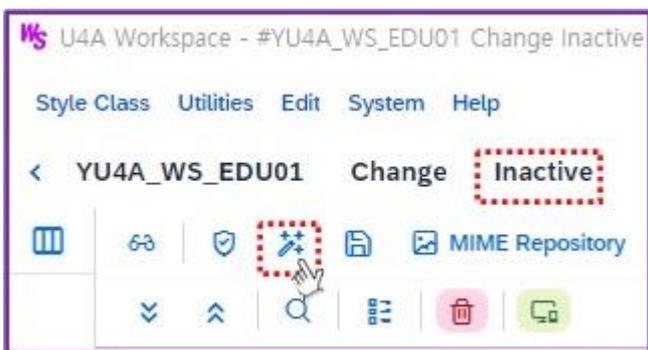
7) 생성된 Buttons UI("BUTTON1")를 선택 후 Workbench 우측 속성(Attributes) 영역 중 "Properties"에 다음과 같이 지정(입력) 함.

“text” 속성에 표시된 텍스트 입력

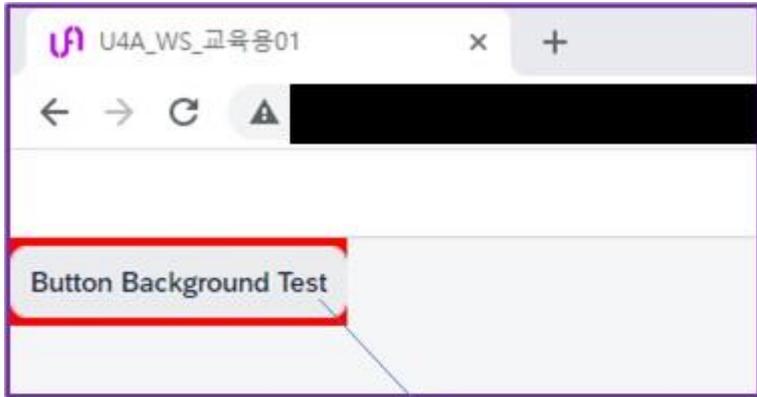
Properties		
text	Button Background Test	
type	Default	
width		
enabled	true	
icon		
iconFirst	true	
activeIcon		
iconDensityAware	true	
textDirection	Inherit	
busy	false	
busyIndicatorDelay	1000	
visible	true	
tooltip		
busyIndicatorSize	Medium	
ariaHasPopup	None	
styleClass	btn_red	
dragAble	false	
dropAble	false	

“styleClass” 속성에 Style Editor 에서 정의한 “btn_red” Class ID 입력

8) 비활성(Inactive) 상태 APP을 Main Toolbar 영역 표시된 활성화(Activate) Button을 Click함.



9) 메인 Toolbar 영역  Button을 클릭하여 APP을 실행함.



“Style Editor에서 배경 적색 적용 코딩이 적용된 버튼

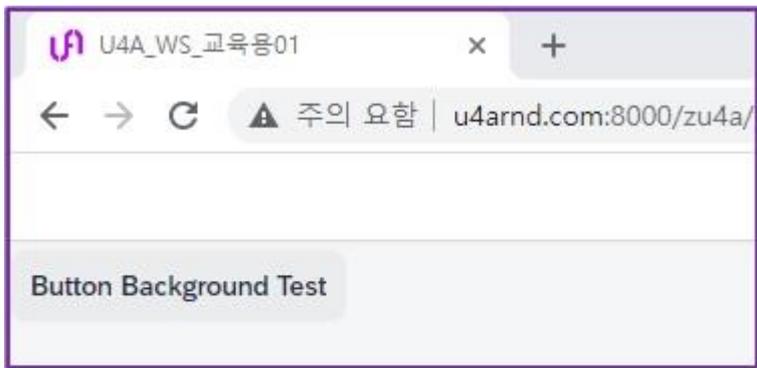


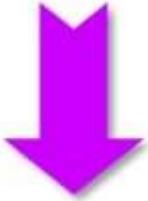
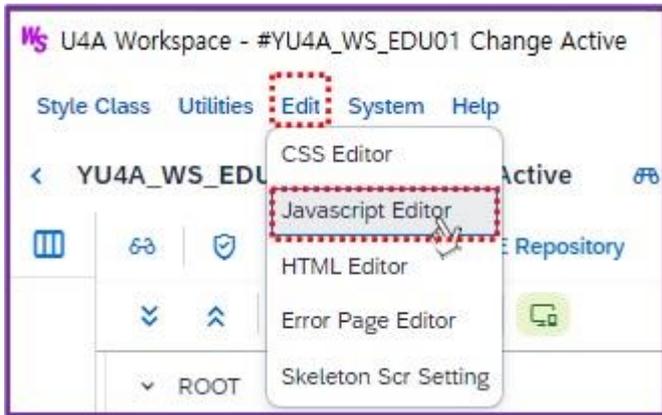
Figure : Style을 변경하지 않은 원본 Button UI

 CSS 기술에 대한 추가 정보를 원할 시 ["CSS Tutorial - W3Schools"](https://www.w3schools.com/css/) Web Site를 참고 하기 바람.

5.3.2 ScriptCode Editor

▣ 목적 : 대상 Application 내 사용될 JavaScript를 정의하는 외부 Editor 임.

1. Editor 실행은 Workbench 상단 추가 기능 Menu 영역에서 "Edit -> StyleCSS editor" Click함.



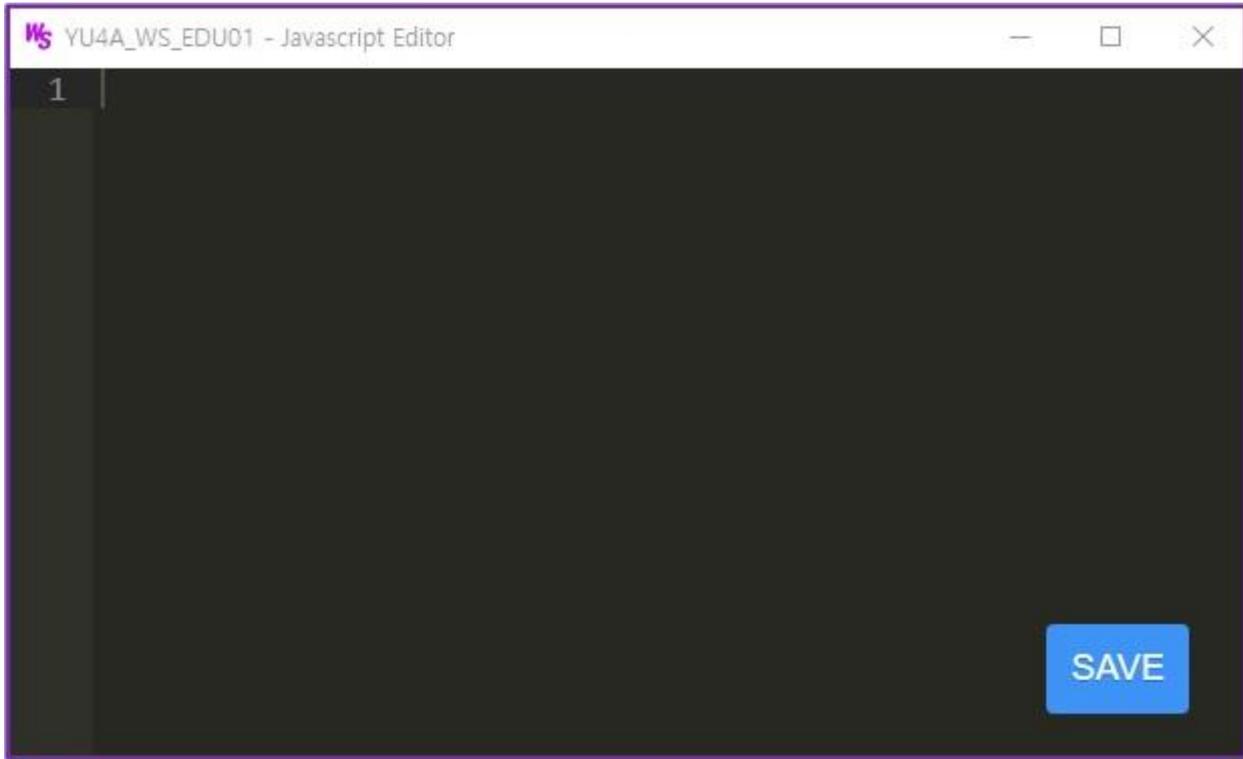
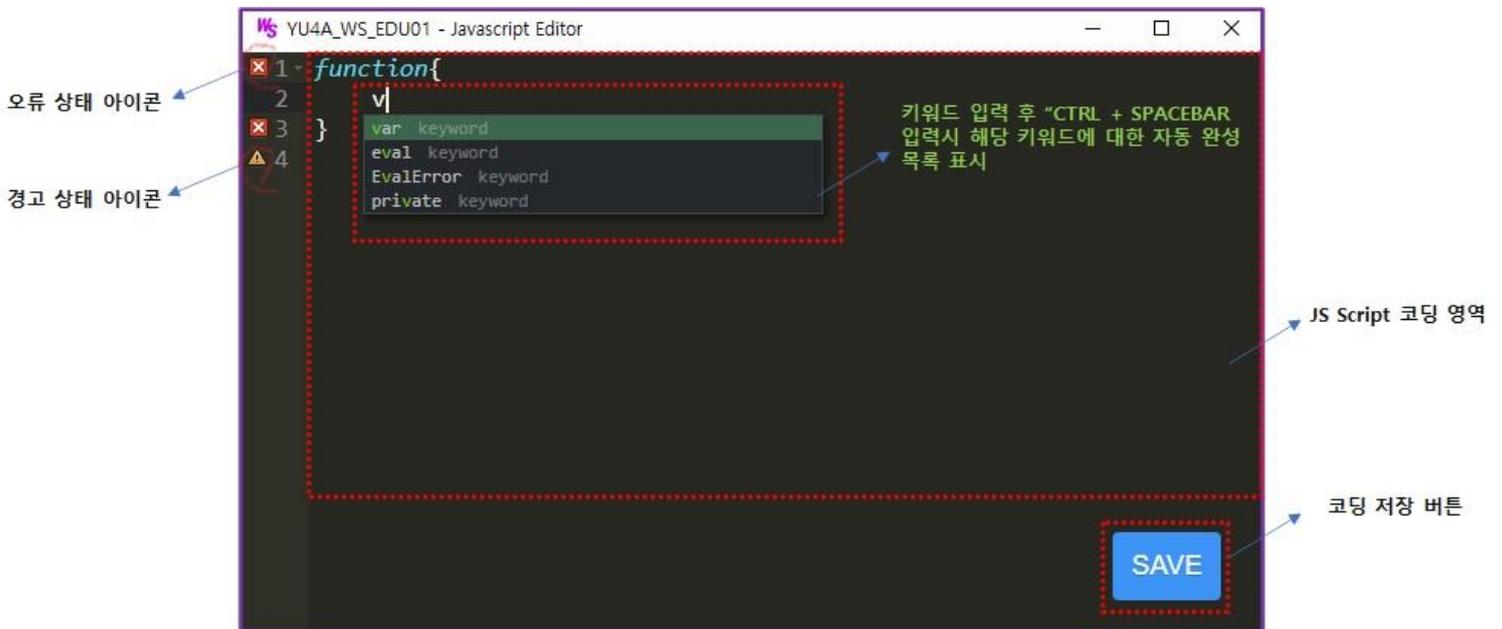


Figure : JS Script Editor

2. Editor 기능 설명



3. 기능 테스트 예제

1) Editor 코딩 영역에서 다음과 같이 입력 함.

```
function jsPopup(){
    alert("Hello U4A WorkSpace!!");
}
```

2) Editor Popup 우측 하단 "SAVE" Button을 Click하여 코딩 내용을 저장함.

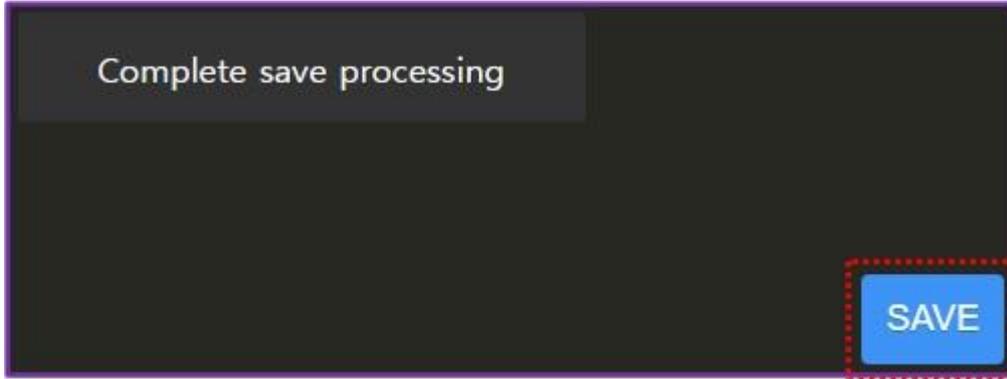
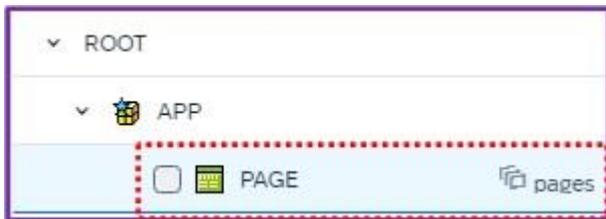
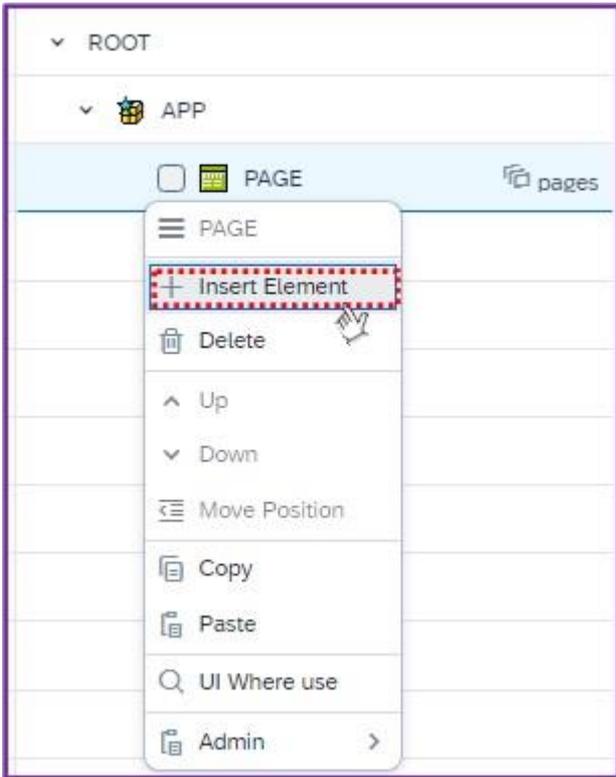


Figure : "SAVE" 버튼 클릭 시 처리에 대한 정상 메시지 표시

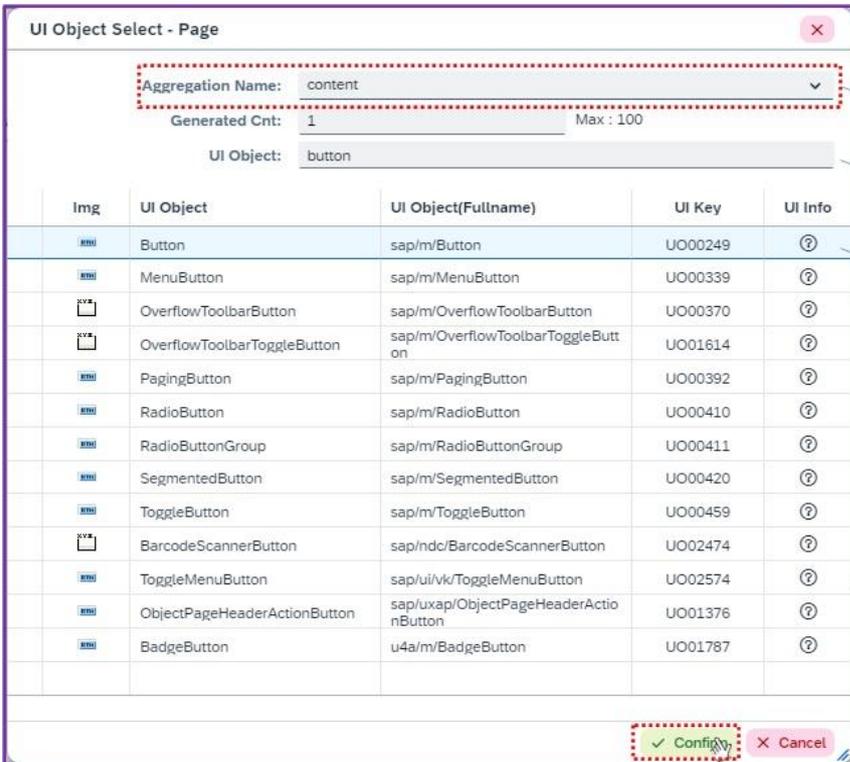
3) Workbench "UX Design Area"에서 "PAGE" Object를 선택함.



4) 선택한 "PAGE"Object에서 마우스 우클릭 후 메뉴 Panel에서 "Insert Elements"를 Click함.



5) "UI Object Select" Popup에서 다음과 같이 처리함.

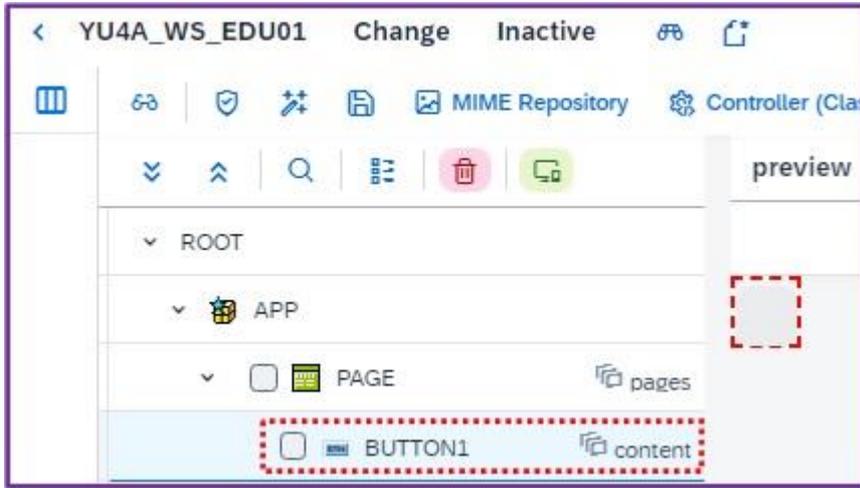


(1) "Aggregation Name" 에서 "content" 선택

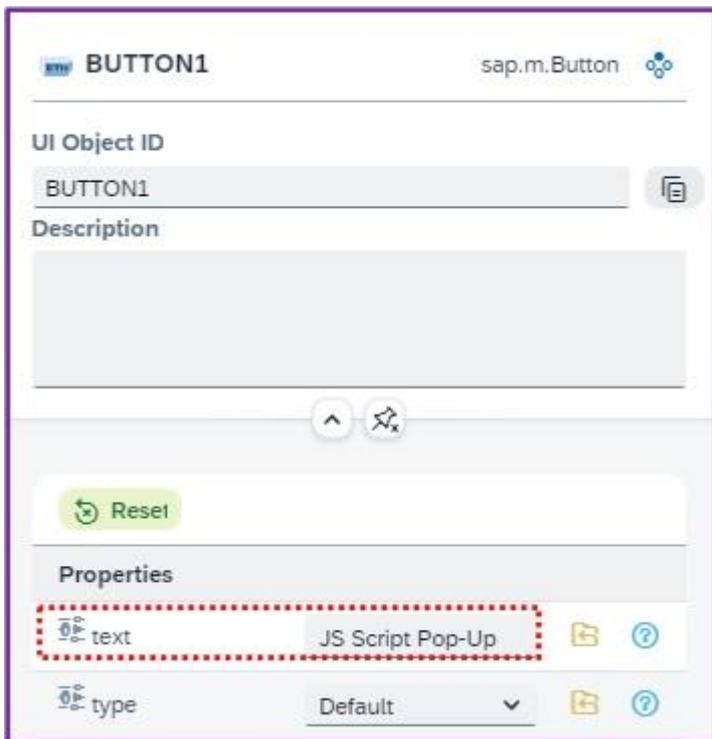
(2) "UI Object" 에서 "button" 키워드 입력 후 Enter

(3) 표시된 항목 중 "Button" 마우스로 선택 후 아래의 "Confirm" Click

6) (5)항에서 정상적으로 처리 되었다면 Workbench "UX Design Area" 및 "Preview Area"에 Button UI 한개가 등록됨을 확인할 수 있음.



7) 생성된 Button UI("BUTTON1")를 선택 후 Workbench 우측 속성(Attributes) 영역 중 "Properties"에 다음과 같이 지정(입력) 함.

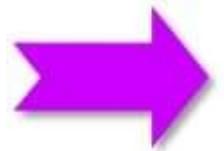
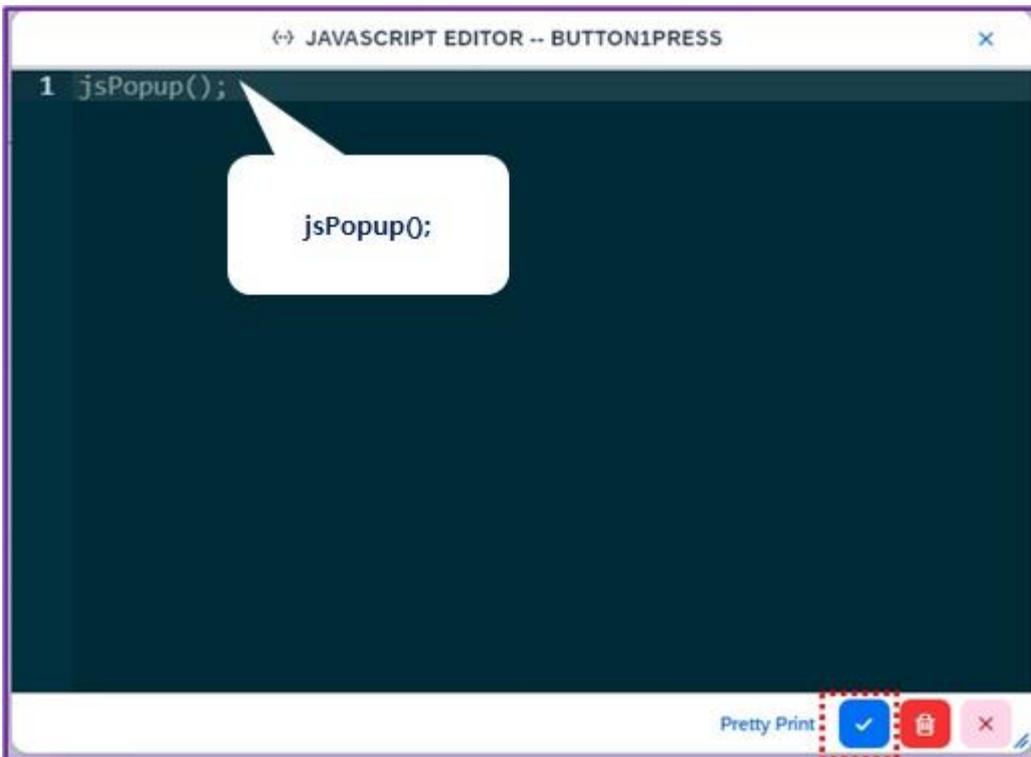


8) Workbench 우측 속성(Attributes) 영역 중 "Events"에 다음과 같이 표시된 Icon Click.

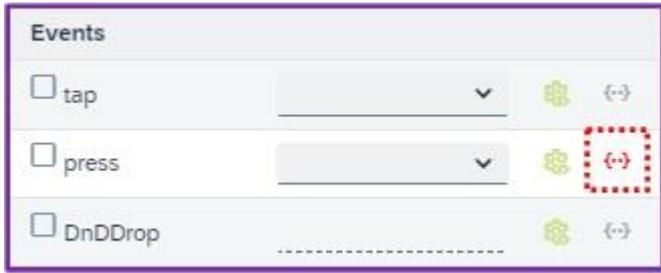


9) 활성화된 "JAVASCRIPT EDITOR" 코딩 Popup에서 (1) ~ (2)항에서 선언한 js Function을 입력 후 "Save" Button을 Click 함.

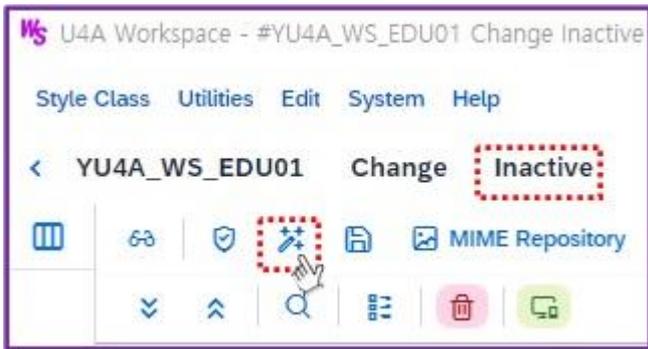
(주의: javascript는 문법상 대소문자를 구분하므로 정확하게 입력해야 함)



10) 만약 (9)항에서 정상적으로 Sript가 등록 되었다면 하기 표시된 것과 같이 적색 Icon으로 표시됨.

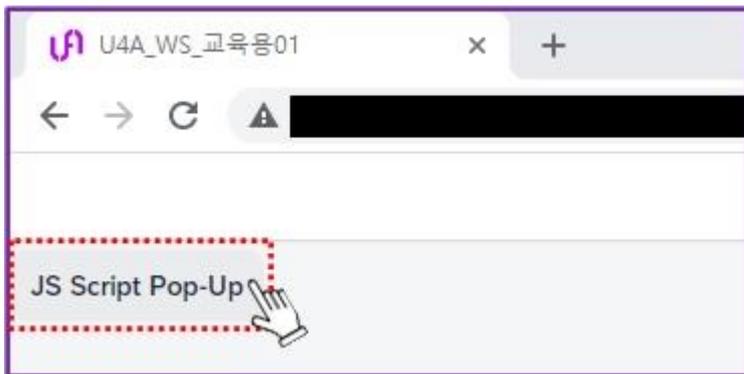


11) 비활성(Inactive) 상태 APP을 Main Toolbar 영역 표시된 활성화(Activate) Button을 Click함.



12) Main Toolbar 영역  Button을 Click하여 APP을 실행함.

13) 실행된 APP에서 표시된 "JS 스크립트 Popup 실행" Button을 Click함.



14) 만약 JavaScript Function "jsPopup"이 정상적으로 작동되었다면 아래의 Browser Popup이 표시됨.

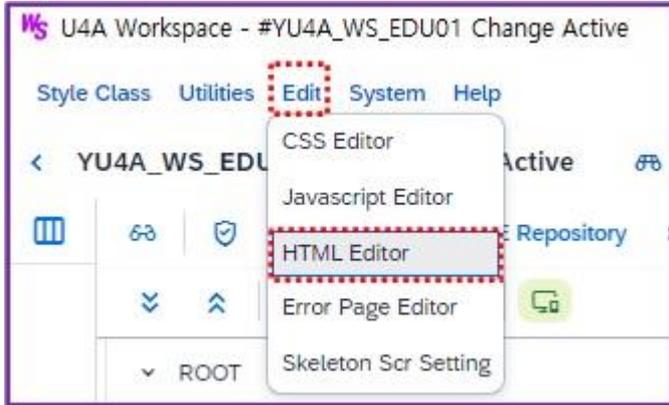


 JavaScript 기술에 대한 추가 정보를 원할 시 ["JavaScript Tutorial - W3Schools"](https://www.w3schools.com/javascript/) Web Site를 참고 하기 바람.

주의 : UI5 APP 기술은 HTML5 JavaScript 기반으로 개발되었기 때문에 무분별할 스크립트를 적용 시 실행된 APP에 심각한 오류를 발생 시킬 수 있기 때문에 가급적 특수한 상황이 아닌 경우 사용하지 않는것을 권고함.

5.3.3 HTMLCode Editor

1. Editor 실행은 Workbench 상단 추가 기능 Menu 영역에서 "Edit -> StyleCSS editor" Click함.



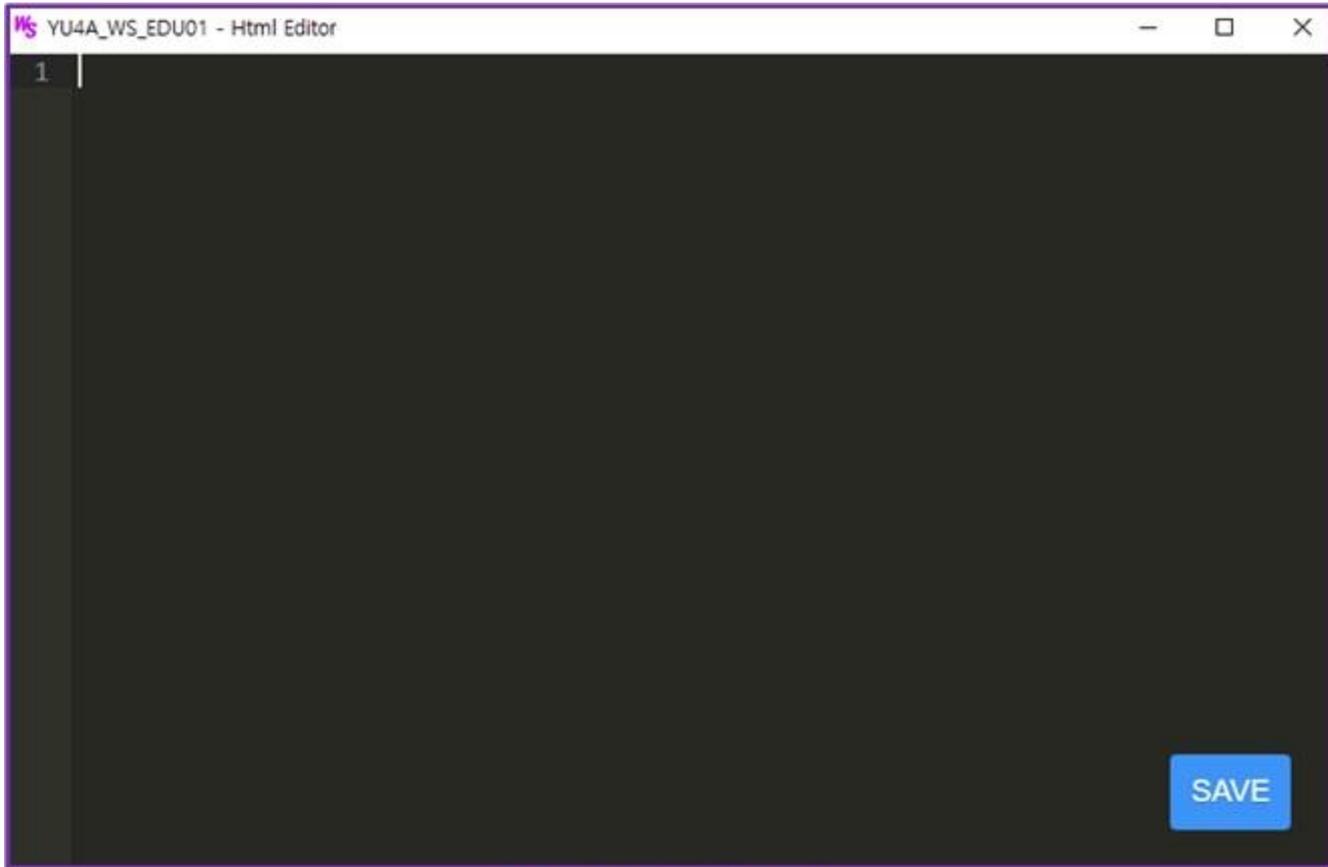
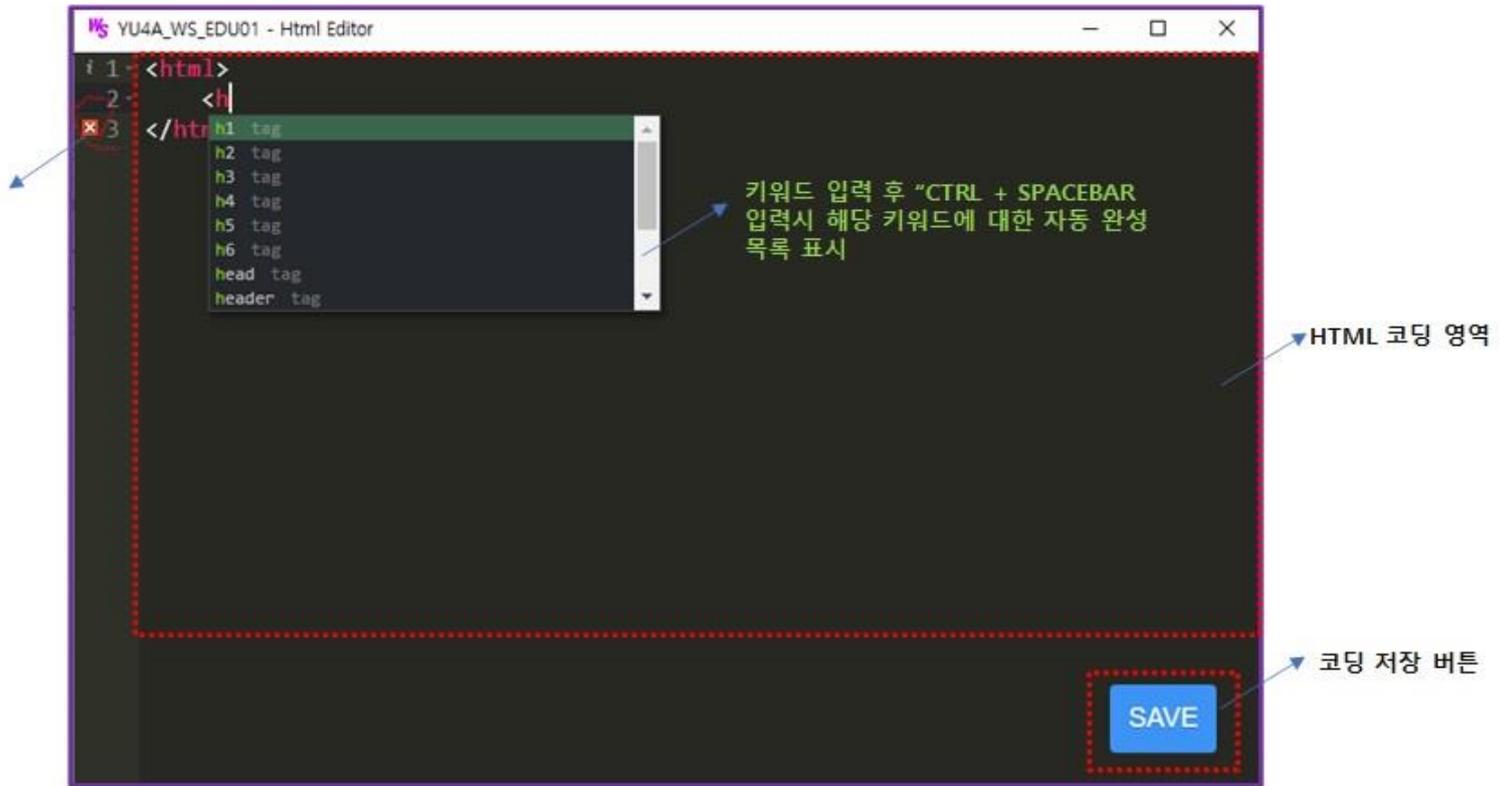


Figure : HTML Code Editor

2. Editor 기능 설명



3. 기능 테스트 예제

1) Editor 코딩 영역에서 다음과 같이 입력 함.

```

<html>
  <body>
    <table border="1">
      <tr>
        <th>col01 </th>
        <th>col02 </th>
      </tr>
      <tr>
        <td>
          Hello U4A!!
        </td>
        <td>
          How are You??
        </td>
      </tr>
    </table>

```

</body>

</html>

2) Editor Popup 우측 하단 "SAVE" Button을 Click하여 코딩 내용을 저장함.

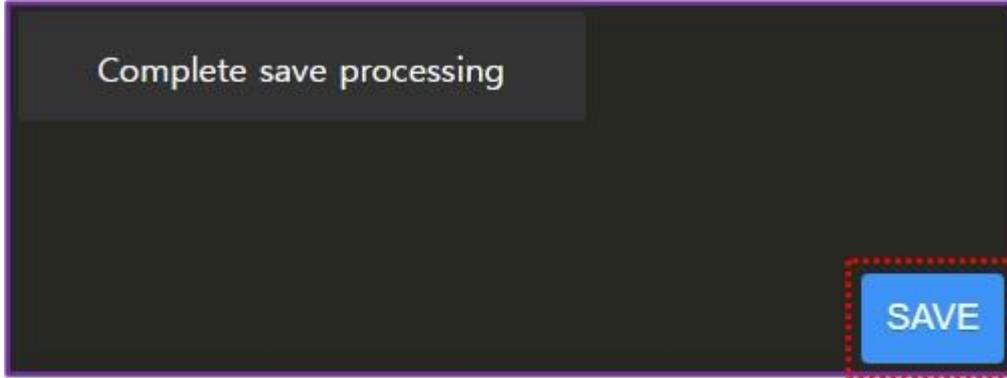
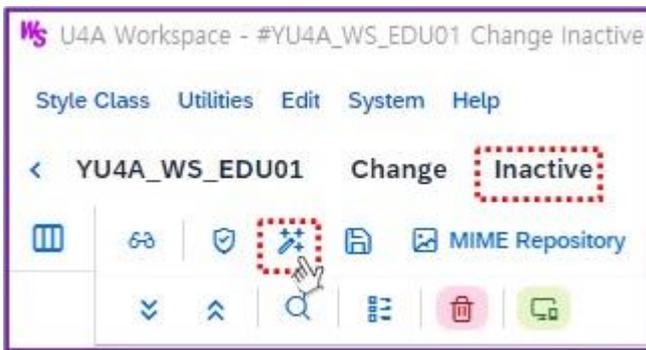


Figure : "SAVE" 버튼 클릭 시 처리에 대한 정상 메시지 표시

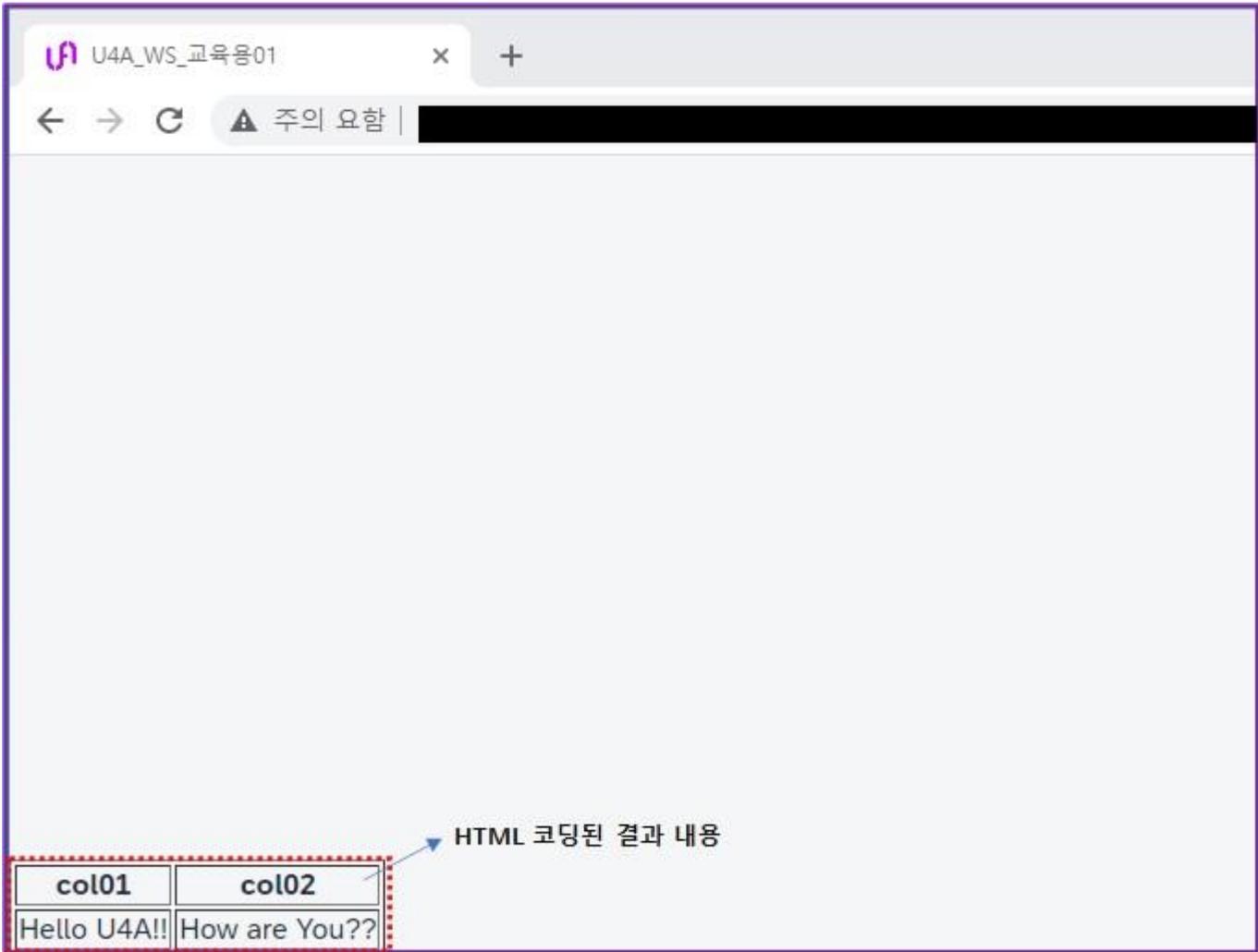
3) 비활성(Inactive) 상태 APP을 Main Toolbar 영역 표시된 활성화(Activate) Button을 Click 함.



4) Main Toolbar 영역  Button을 Click하여 APP을 실행함.

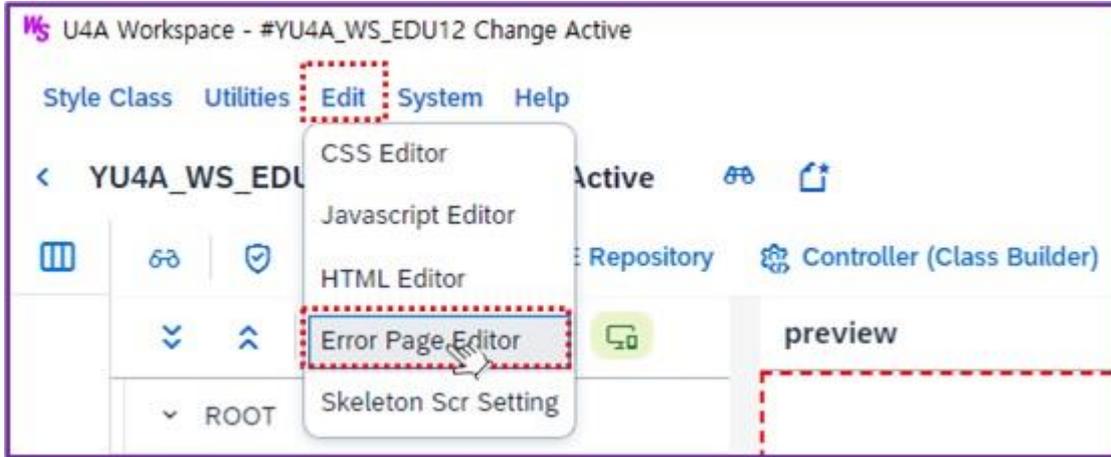
5) APP 실행 후 스크롤을 아래쪽으로 이동.

6) APP 화면 하단에 코딩된 내용의 결과가 표시됨을 확인 할 수 있음. 즉, HTML 코딩 내용 출력은 우선순위 측면에서 UI5 UI 출력이 최우선이기 때문에 UI5 영역 밖에 표시됨.

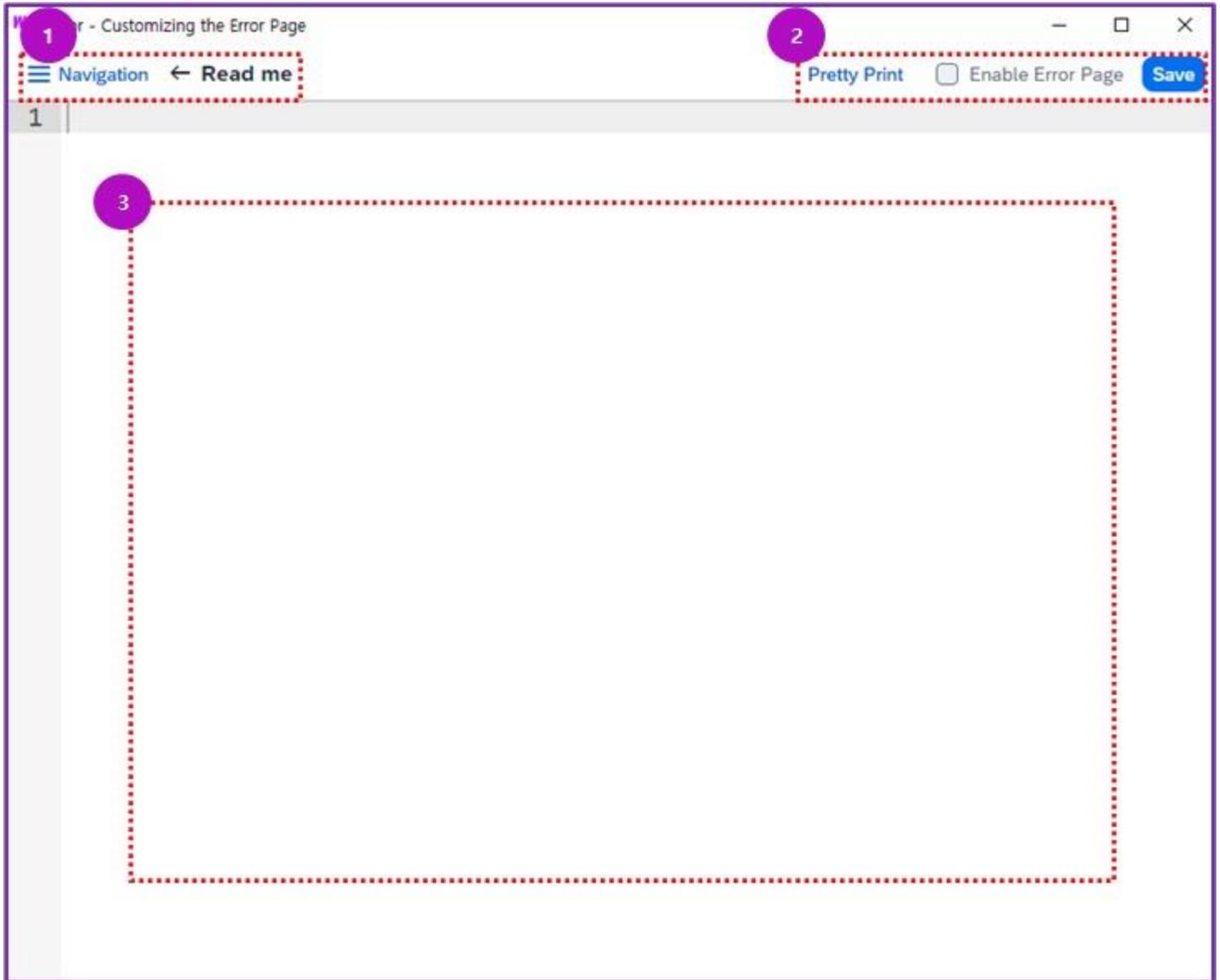


 HTML 기술에 대한 추가 정보를 원할 시 ["HTML Tutorial - W3Schools"](https://www.w3schools.com/html/) Web Site를 참고 하기 바람.

5.3.4 Error Page Editor



2. Error Page Editor 개요.



▶

Preview

[How to Use]

1. APP 실행 중 발생하는 오류(시스템 오류 포함)페이지 재구성 필요시 우측 HTML 에디터를 사용한다.
2. 만약 오류 본문 내 시스템 메시지를 출력 필요시 해당 위치에 매개변수 "`<%=MSG=%>`"를 입력한다.
3. 만약 에디터 HTML 코딩에 대한 미리보기를 원할 시 "Preview" 아이콘을 클릭한다.
4. 완성된 HTML 코딩을 시스템에 적용 시 "Enable Error Page" 체크박스 체크 후 "SAVE" 버튼을 클릭한다.

1. If you need to reorganize the page errors (including system errors) that occur while running the APP, use the HTML editor on the right.
2. If you need to output the system message in the error body, enter the parameter "`<%=MSG=%>`" in the corresponding location
3. If you want to preview the HTML coding, click the "Preview" icon.
4. When applying the completed HTML coding to the system, check the "Enable Error Page" checkbox and click the "SAVE" button.

“Navigation” 버튼 클릭시 상세화면

(1) Navigation : Error Page Editor 사용 방법 가이드 화면

(2) 코딩 소스 정렬, Customizing 한 Error Page 사용여부, 저장

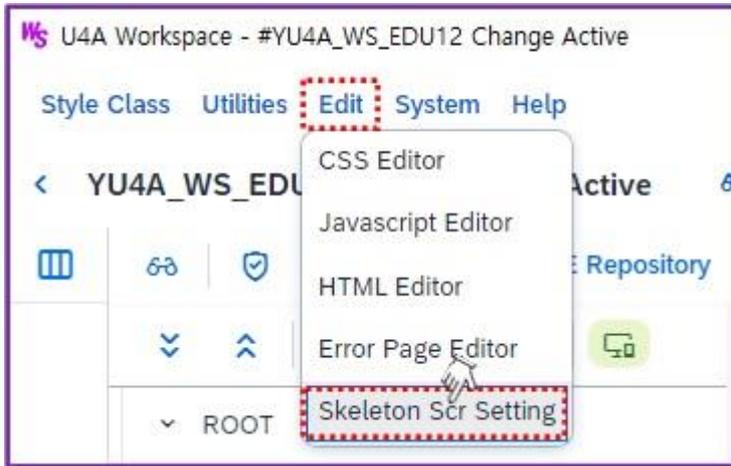
① Pretty Print : (3)영역에 작성한 코딩 소스 정렬

② Enable Error Page : Customizing 한 Error Page 사용여부 체크

③ Save : (3)영역에 작성한 코딩 소스 저장.

(3) Error Page HTML 코딩 작성 영역

5.3.5 Skeleton Scr Setting

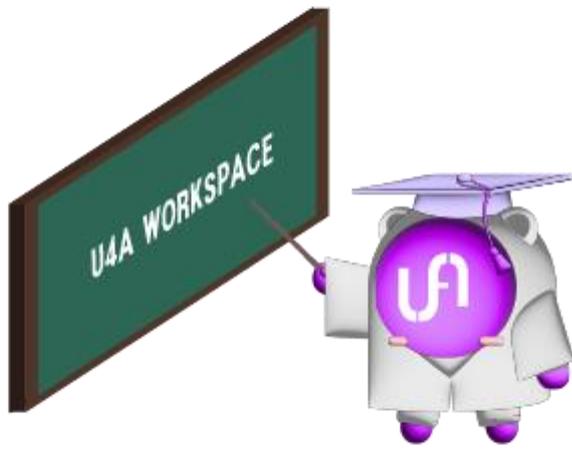


2. "Skeleton Screen Configuration" 개요.



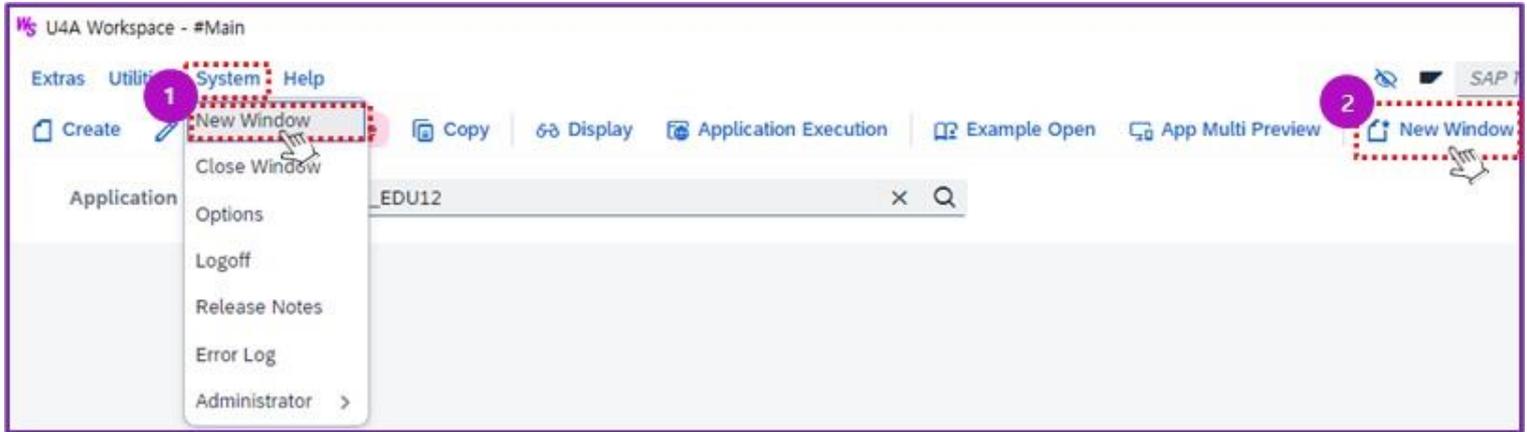
- 1) **Waiting mode effect** : 미리보기용 화면(= Skeleton Screen)이 출력된 뒤 Application 화면이 호출 되기전까지 미리보기 화면이라는 것을 나타내기 위한 효과
- 2) **Use of glass** : Skeleton Screen의 blurry(= 흐릿한) 효과 사용여부 체크
- 3) **Glass concentration** : Skeleton Screen의 blurry 농도 설정(0 ->100으로 갈수록 더욱 흐릿해진다, "Use of glass"가 미체크 되어있다면 적용되지 않음)

5.4 추가 기능 메뉴 - System

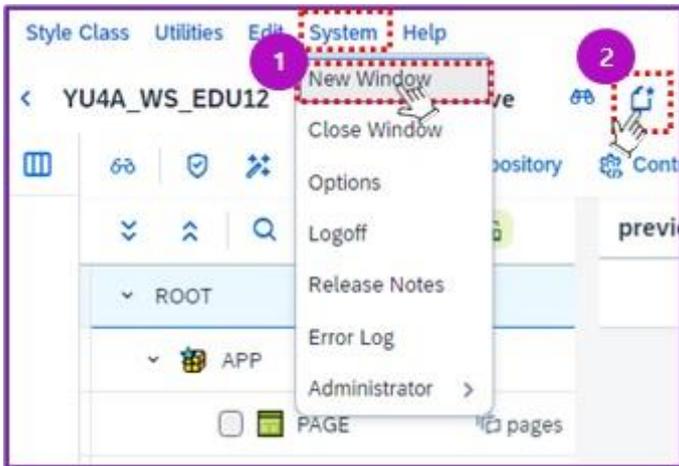


5.4.1 New Window

① U4A WorkSpace 메인 화면에서 선택

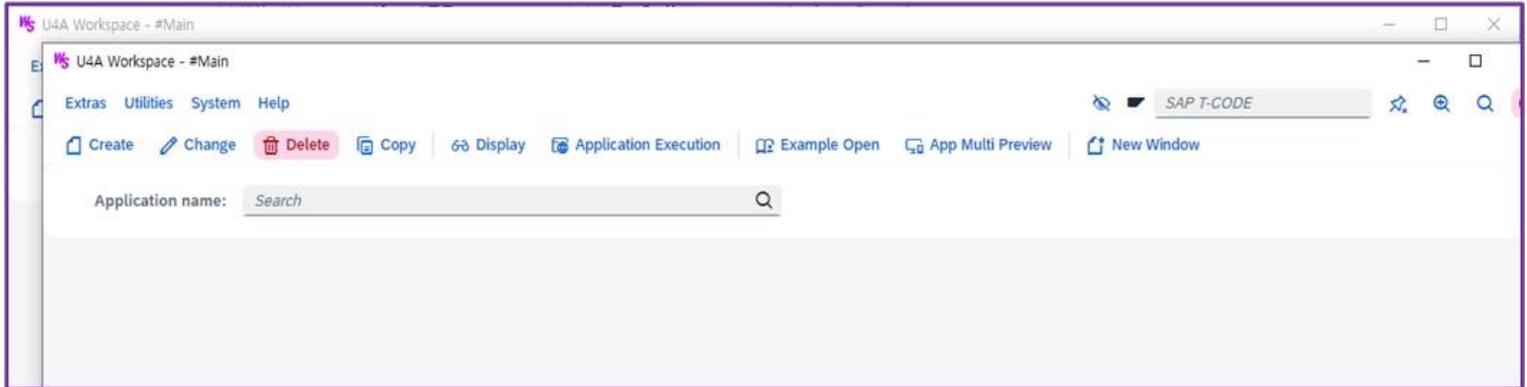


② Application 안에서 선택



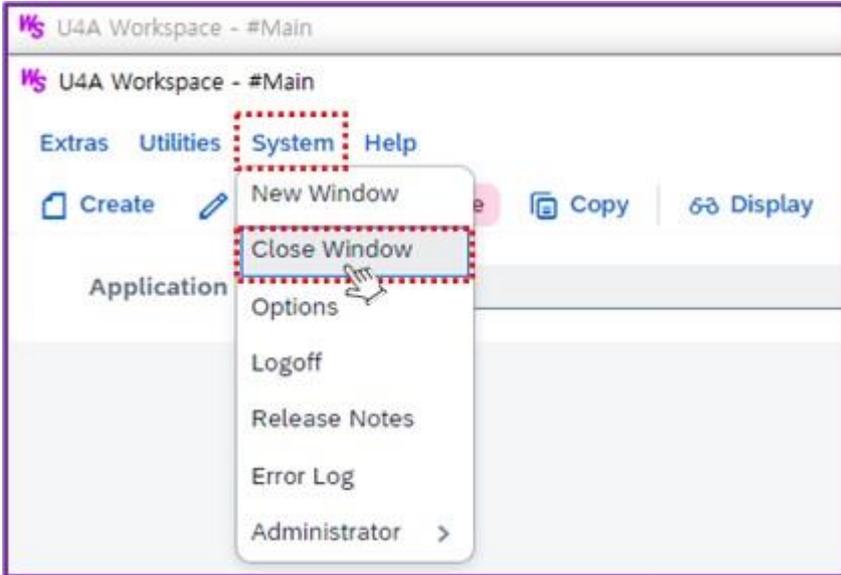
2. U4A WorkSpace 새 화면 확인.

U4A IDE HELP

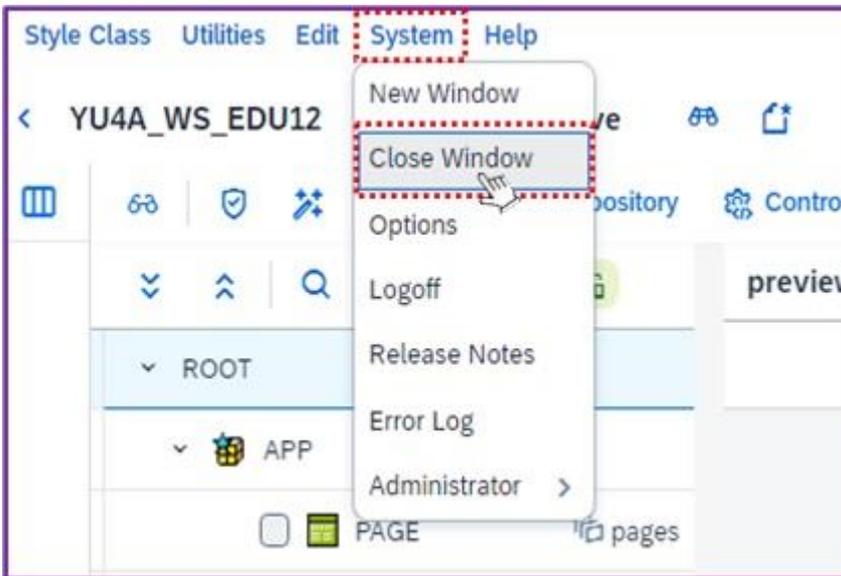


5.4.2 Close Window

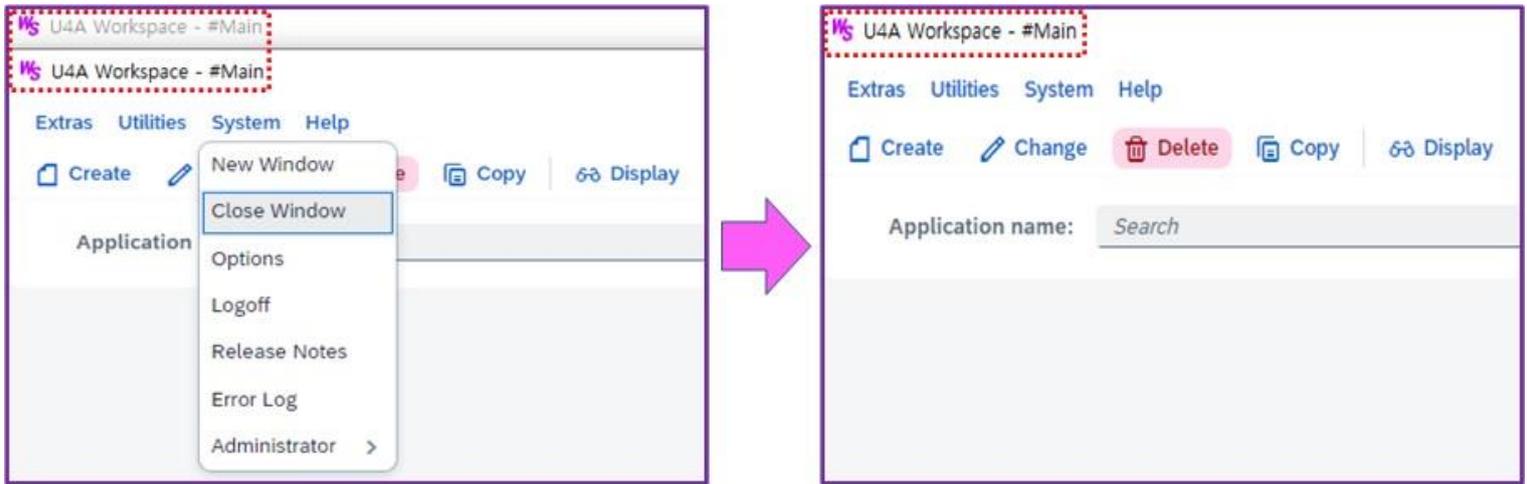
① U4A WorkSpace 메인 화면에서 선택



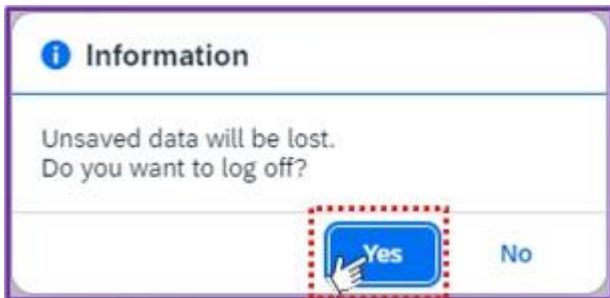
② Application 안에서 선택



2. U4A WorkSpce창 종료 확인(2개 -> 1개로 줄어듦)

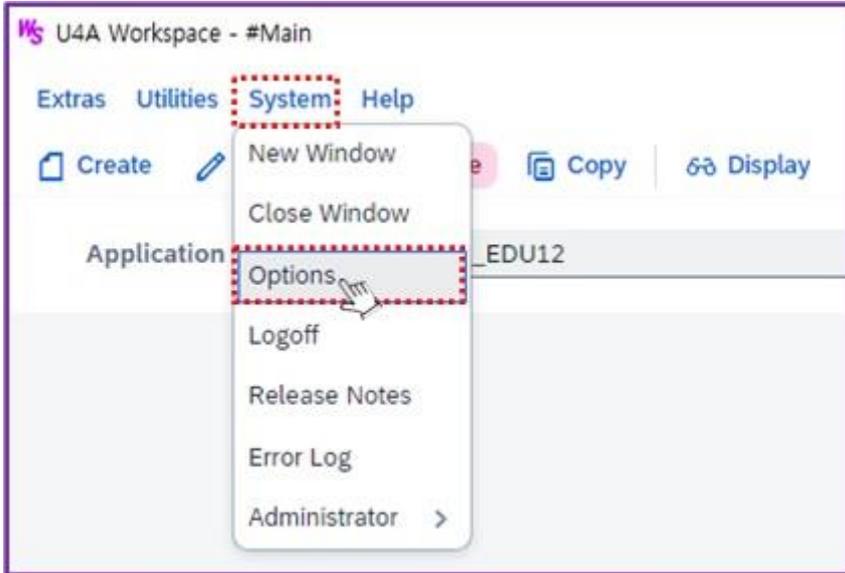


※ U4A WorkSpace창 1개인 상태에서 Close Window 선택시 아래와 같은 Pop-Up 발생하고 "Yes"버튼 누르면 종료

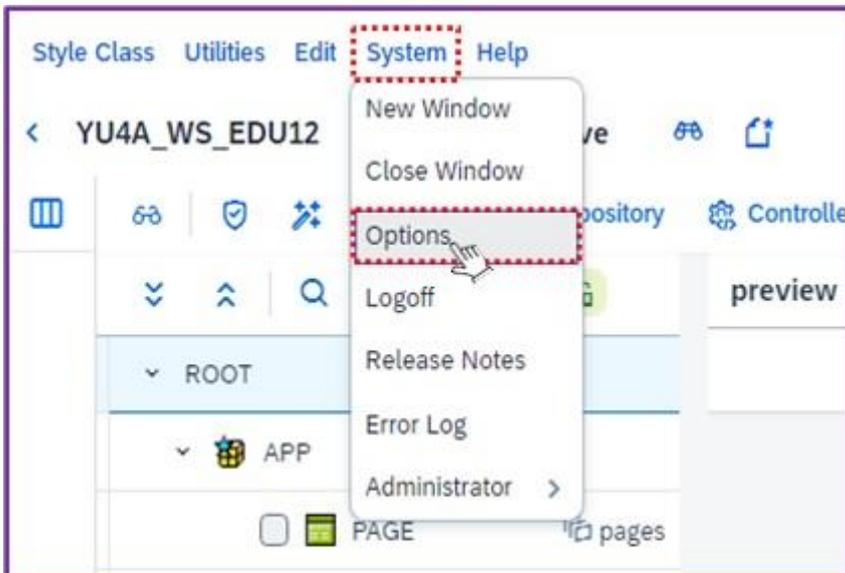


5.4.3 Options

① U4A WorkSpace 메인 화면에서 선택

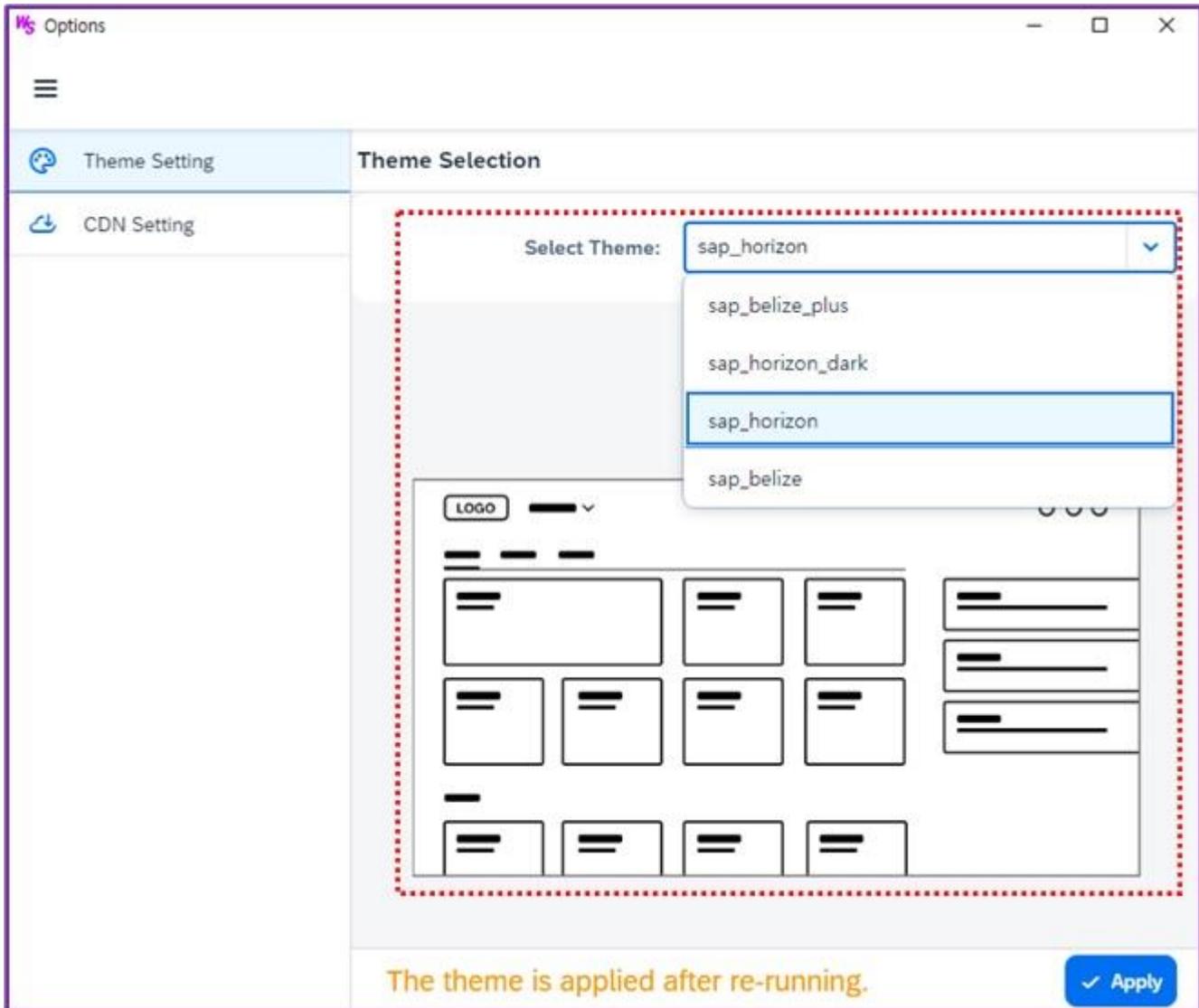


② Application 안에서 선택

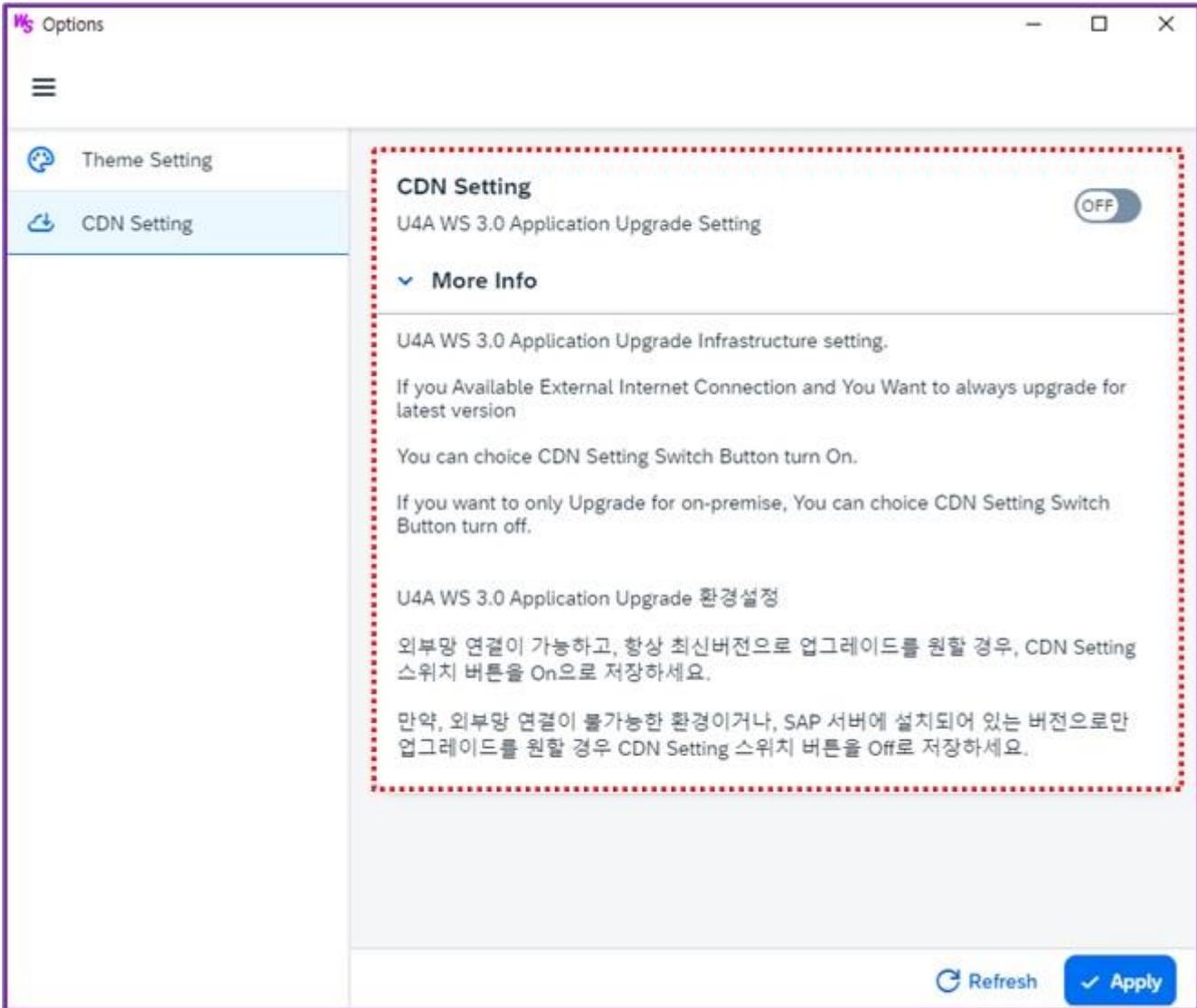


2. Options 개요

① Theme Setting (테마) => Theme는 현재 4개이며 원하는것을 선택한 후 Apply 버튼 클릭



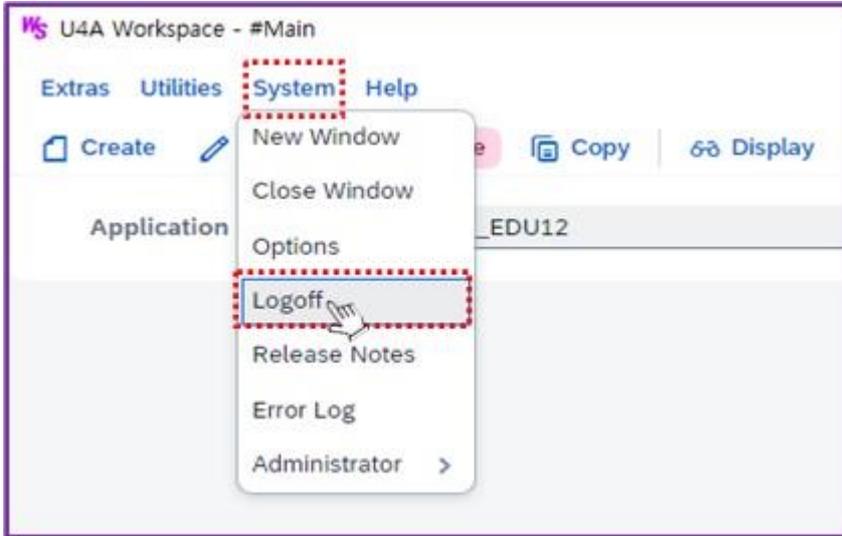
② CDN Setting(U4A WS 3.0 Application Upgrade Setting) = > ON / OFF 스위치 설정 후 Apply 버튼 클릭



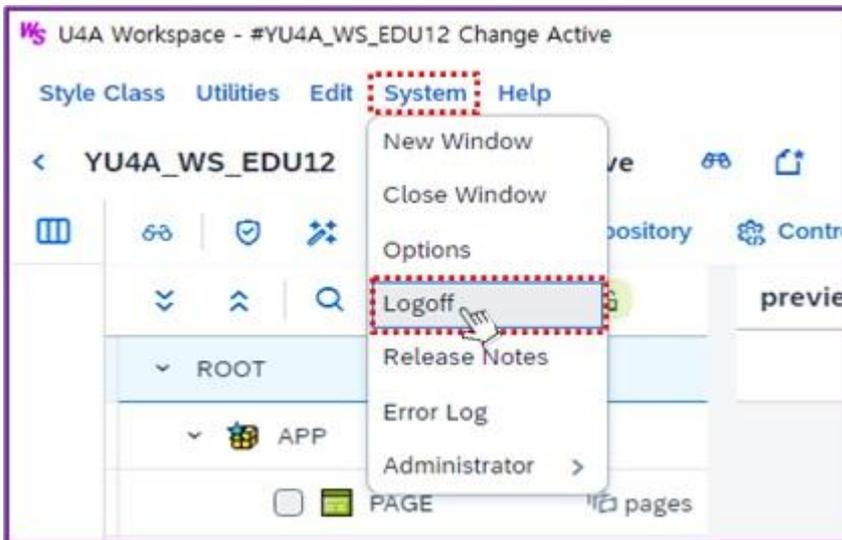
 설정 변경 후 U4A WorkSpace 재 시작을 권장함.

5.4.4 Logoff

① U4A WorkSpace 메인 화면에서 선택



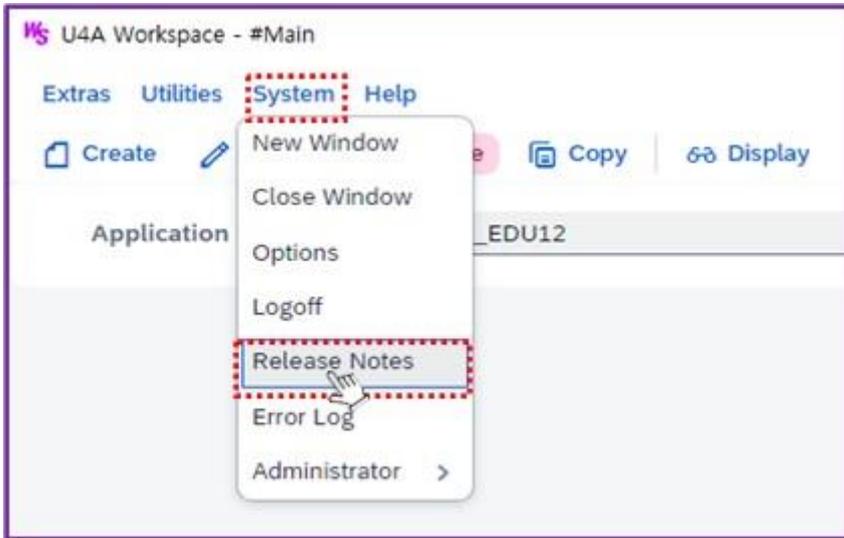
② Application 안에서 선택



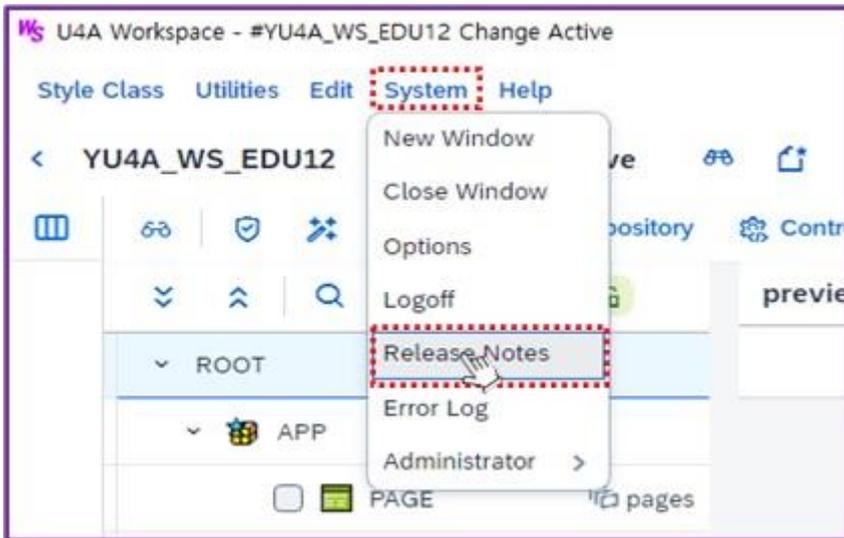
2. Logoff Pop-Up에서 Yes 버튼 클릭하면 모든 U4A WorkSpace 종료.(SAP GUI Logoff 와 동일)

5.4.5 Release Notes

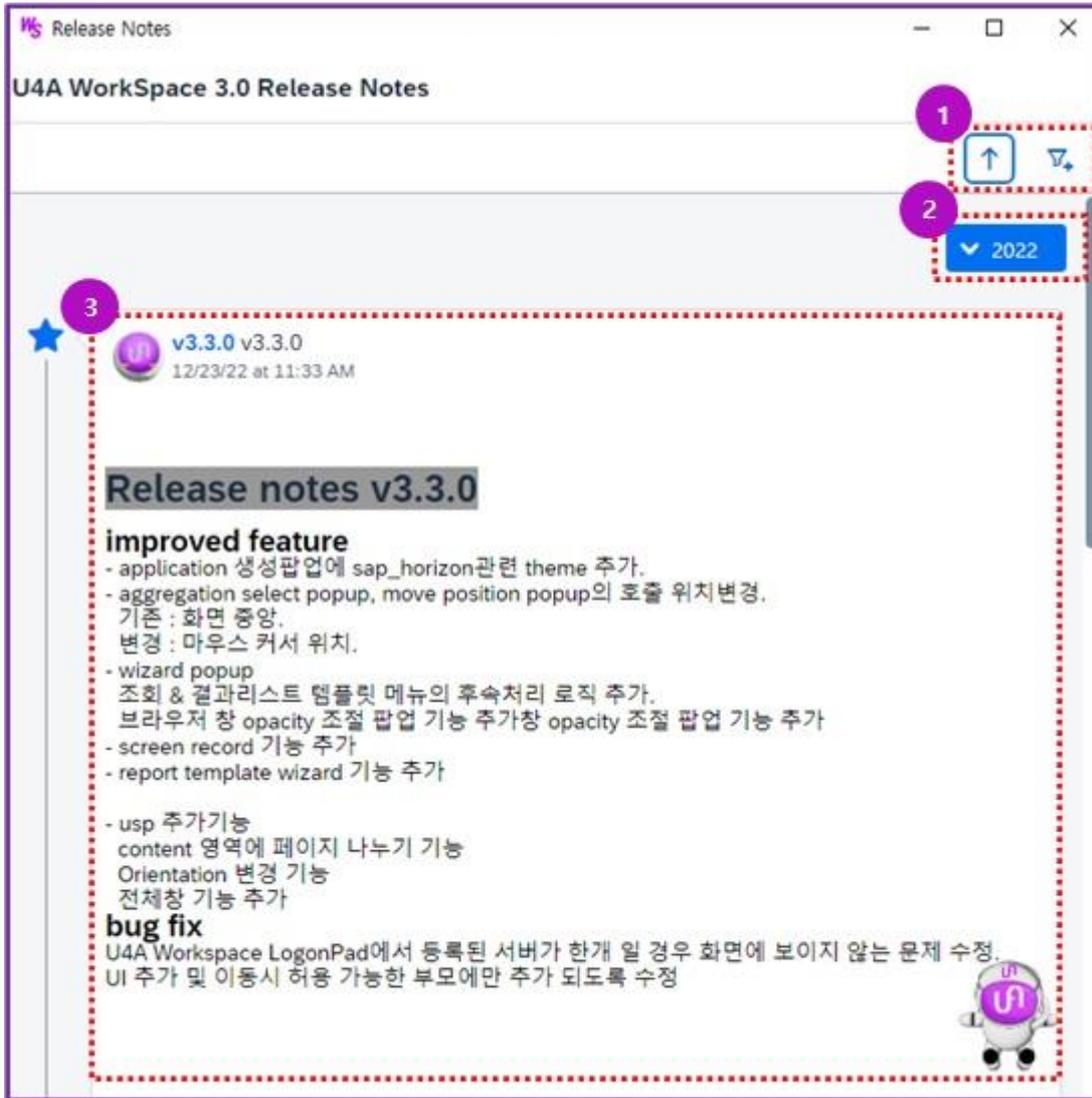
① U4A WorkSpace 메인 화면에서 선택



② Application 안에서 선택



2. Release Notes 개요.



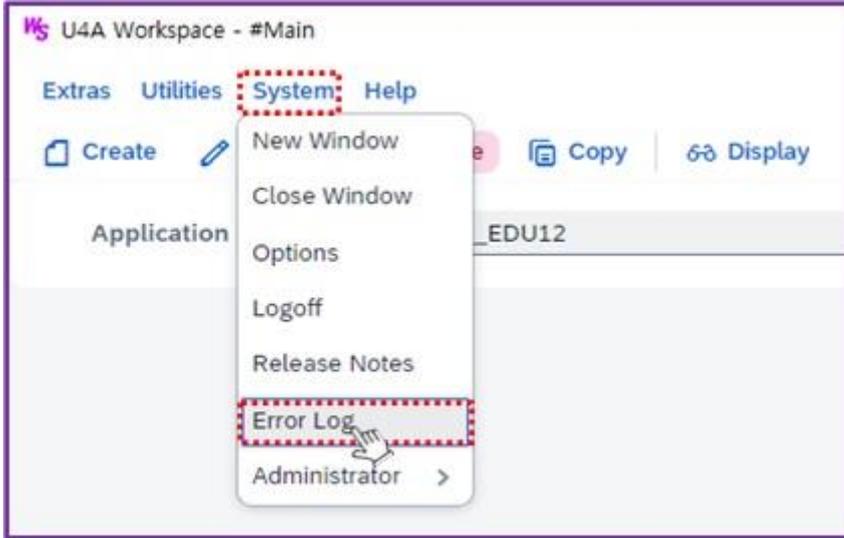
(1) Sort(Patch Version), Filter(Patch Version or 날짜) 기능

(2) 연도별 U4A WorkSpace Release Notes 확인 기능

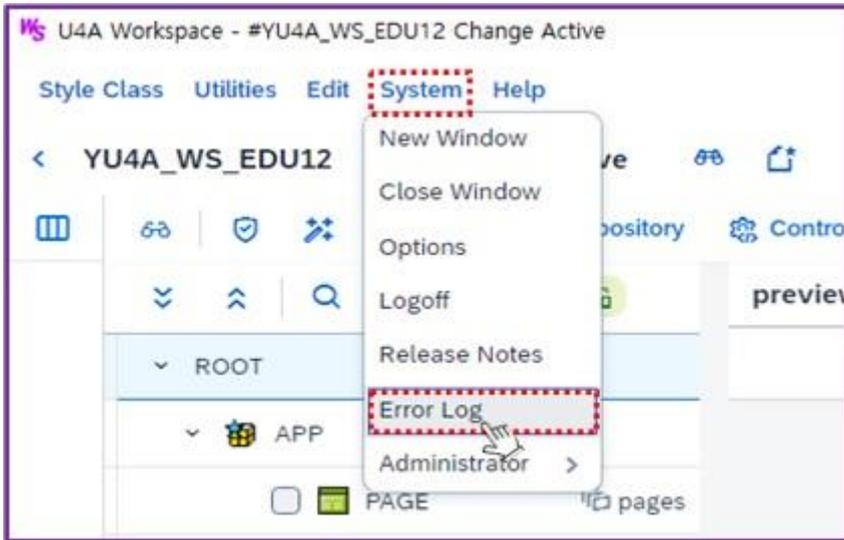
(3) U4A WorkSpace Release Notes 내역

5.4.6 Error Log

① U4A WorkSpace 메인 화면에서 선택

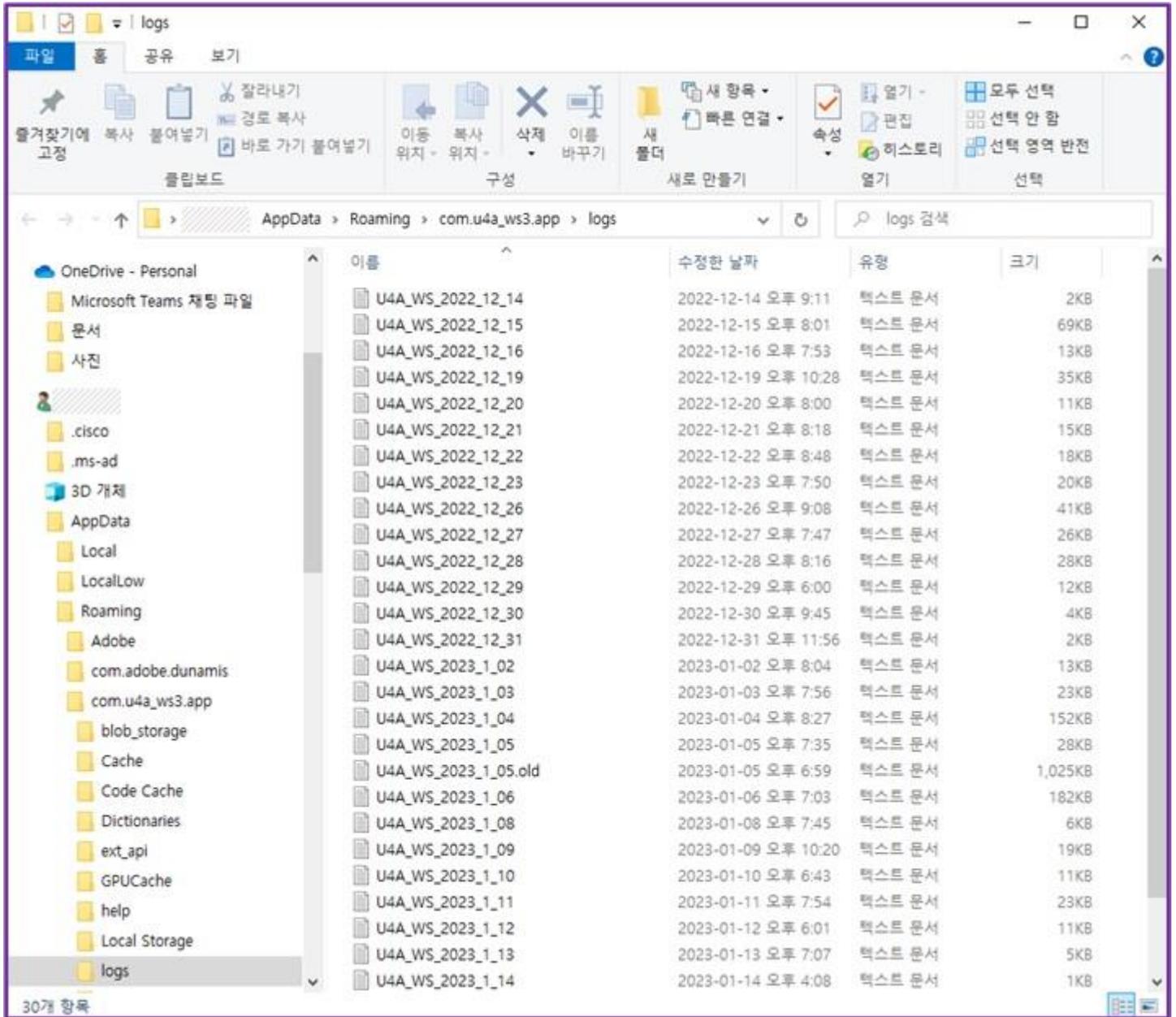


② Application 안에서 선택



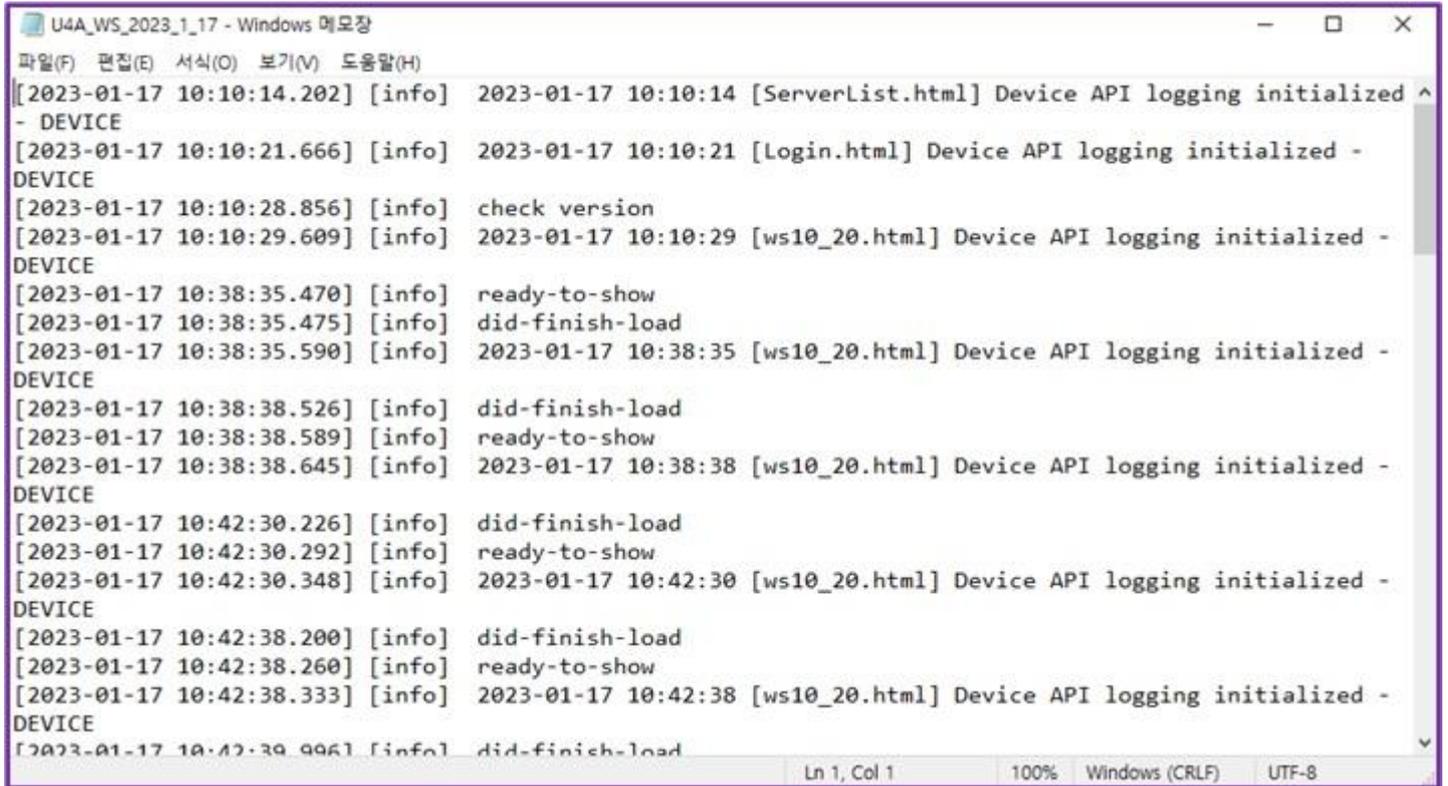
2. Error Log 선택시 날짜별 Log 모음 폴더 호출.

※ Log 파일을 U4A Solution Team 에게 전달하면 에러 발생시 신속한 대처 가능



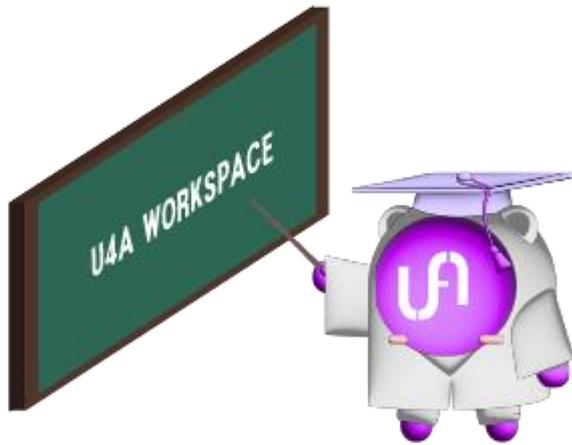
※ Log 개요.

U4A IDE HELP



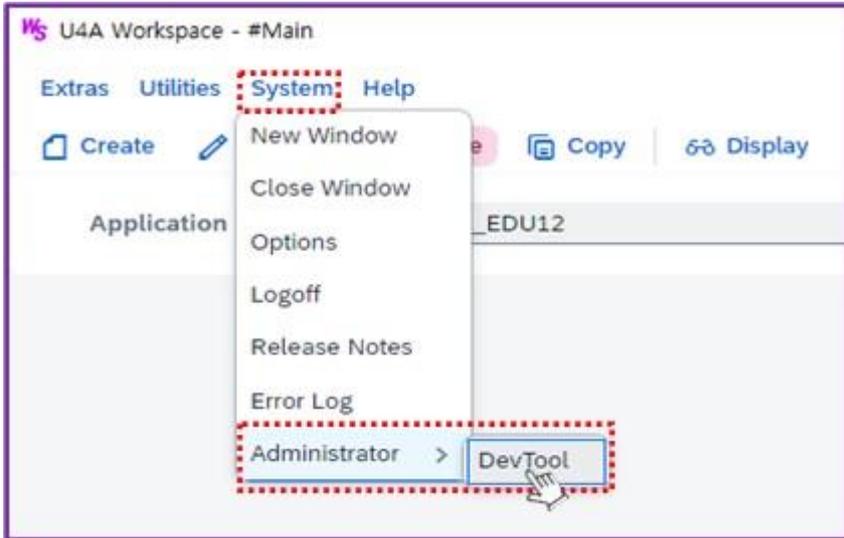
```
U4A_WS_2023_1_17 - Windows 메모장
파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)
[[2023-01-17 10:10:14.202] [info] 2023-01-17 10:10:14 [ServerList.html] Device API logging initialized -
- DEVICE
[2023-01-17 10:10:21.666] [info] 2023-01-17 10:10:21 [Login.html] Device API logging initialized -
DEVICE
[2023-01-17 10:10:28.856] [info] check version
[2023-01-17 10:10:29.609] [info] 2023-01-17 10:10:29 [ws10_20.html] Device API logging initialized -
DEVICE
[2023-01-17 10:38:35.470] [info] ready-to-show
[2023-01-17 10:38:35.475] [info] did-finish-load
[2023-01-17 10:38:35.590] [info] 2023-01-17 10:38:35 [ws10_20.html] Device API logging initialized -
DEVICE
[2023-01-17 10:38:38.526] [info] did-finish-load
[2023-01-17 10:38:38.589] [info] ready-to-show
[2023-01-17 10:38:38.645] [info] 2023-01-17 10:38:38 [ws10_20.html] Device API logging initialized -
DEVICE
[2023-01-17 10:42:30.226] [info] did-finish-load
[2023-01-17 10:42:30.292] [info] ready-to-show
[2023-01-17 10:42:30.348] [info] 2023-01-17 10:42:30 [ws10_20.html] Device API logging initialized -
DEVICE
[2023-01-17 10:42:38.200] [info] did-finish-load
[2023-01-17 10:42:38.260] [info] ready-to-show
[2023-01-17 10:42:38.333] [info] 2023-01-17 10:42:38 [ws10_20.html] Device API logging initialized -
DEVICE
[2023-01-17 10:42:39.996] [info] did-finish-load
Ln 1, Col 1 100% Windows (CRLF) UTF-8
```

5.4.7 Administrator

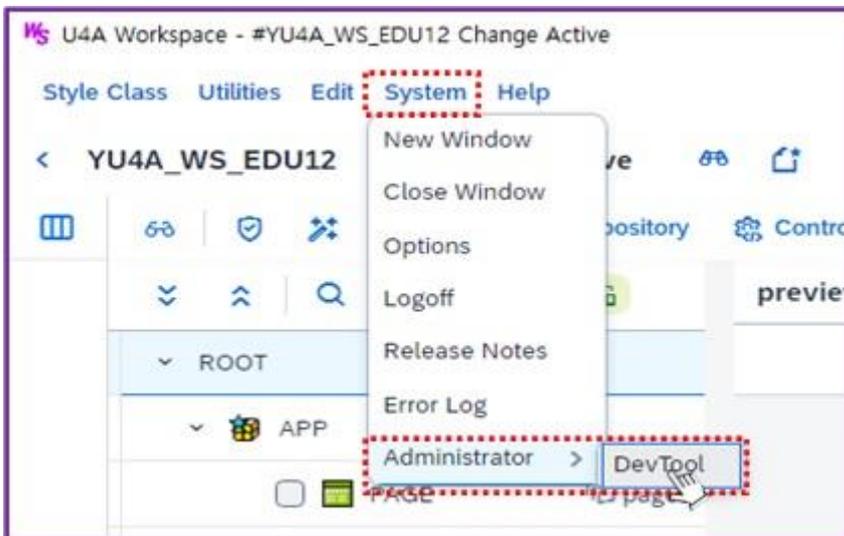


5.4.7.1 DevTool

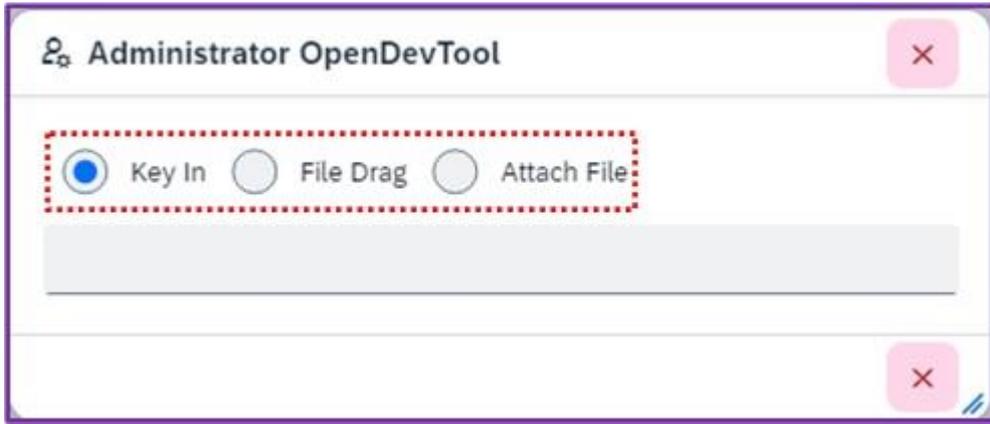
① U4A WorkSpace 메인 화면에서 선택



② Application 안에서 선택



2. DevTool 개요.



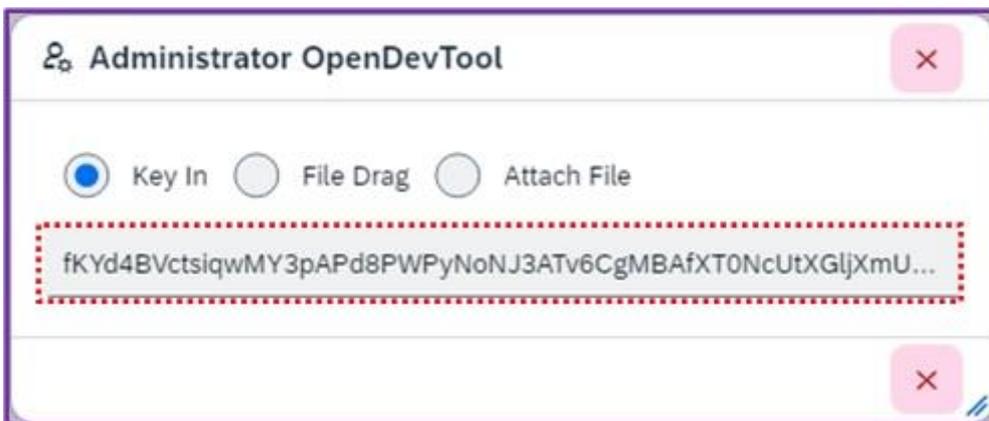
(1) Key In : U4A Solution Team에게 받은 Key를 입력.

(2) File Drag : U4A Solution Team에게 받은 Key file을 Drag & Drop을 통해 등록.

(3) Attach File : U4A Solution Team에게 받은 Key file을 업로드를 통해 등록.

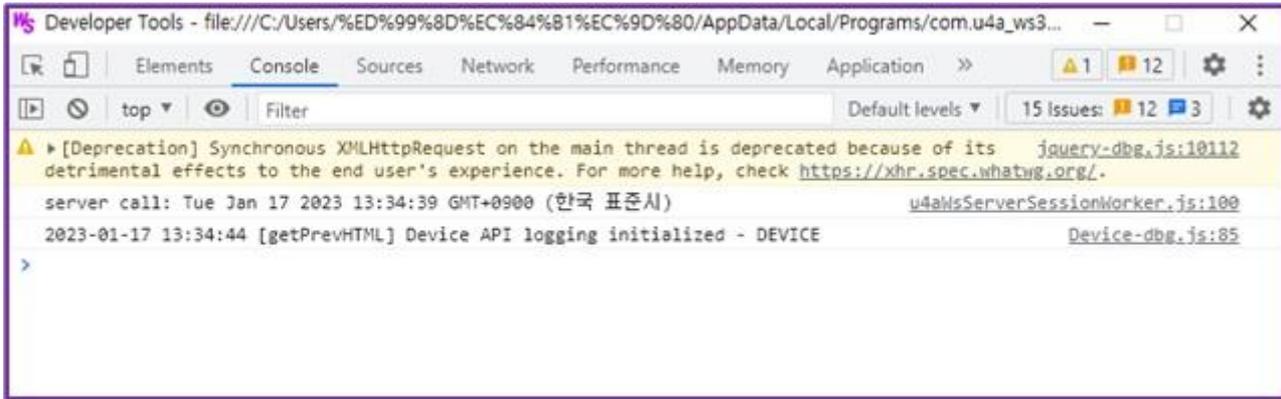
3. 본 단원에서는 Key In 방식을 통하여 DevTool를 호출.

① U4A Solution Team에게 받은 "Key" 입력 후 Enter.

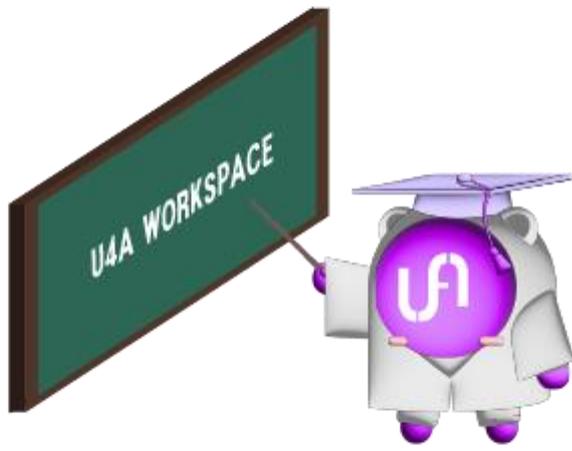


② Developer Tool 호출

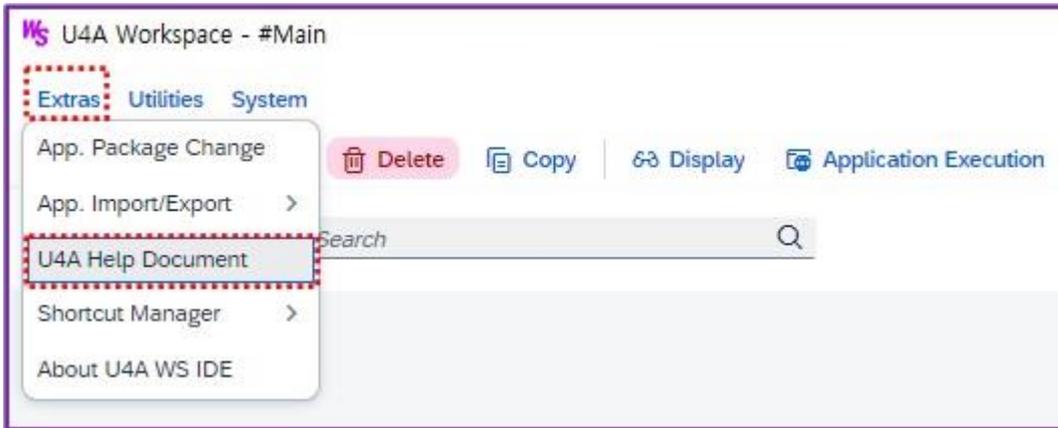
U4A IDE HELP



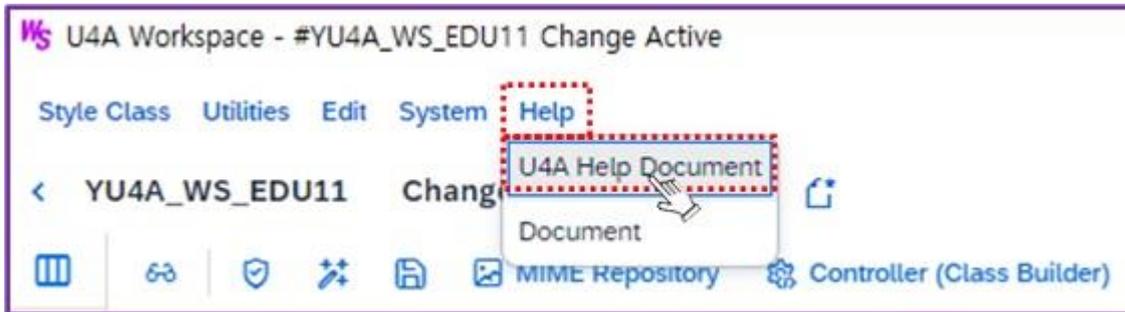
5.5 추가 기능 메뉴 - Help



5.5.1 Help Document



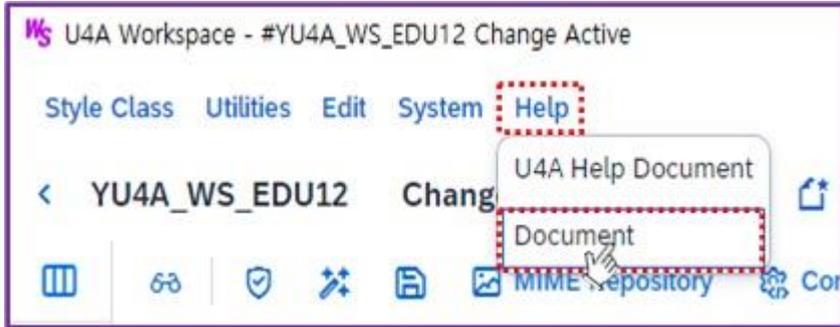
② Application 화면에서 Help -> U4A Help Document 선택.



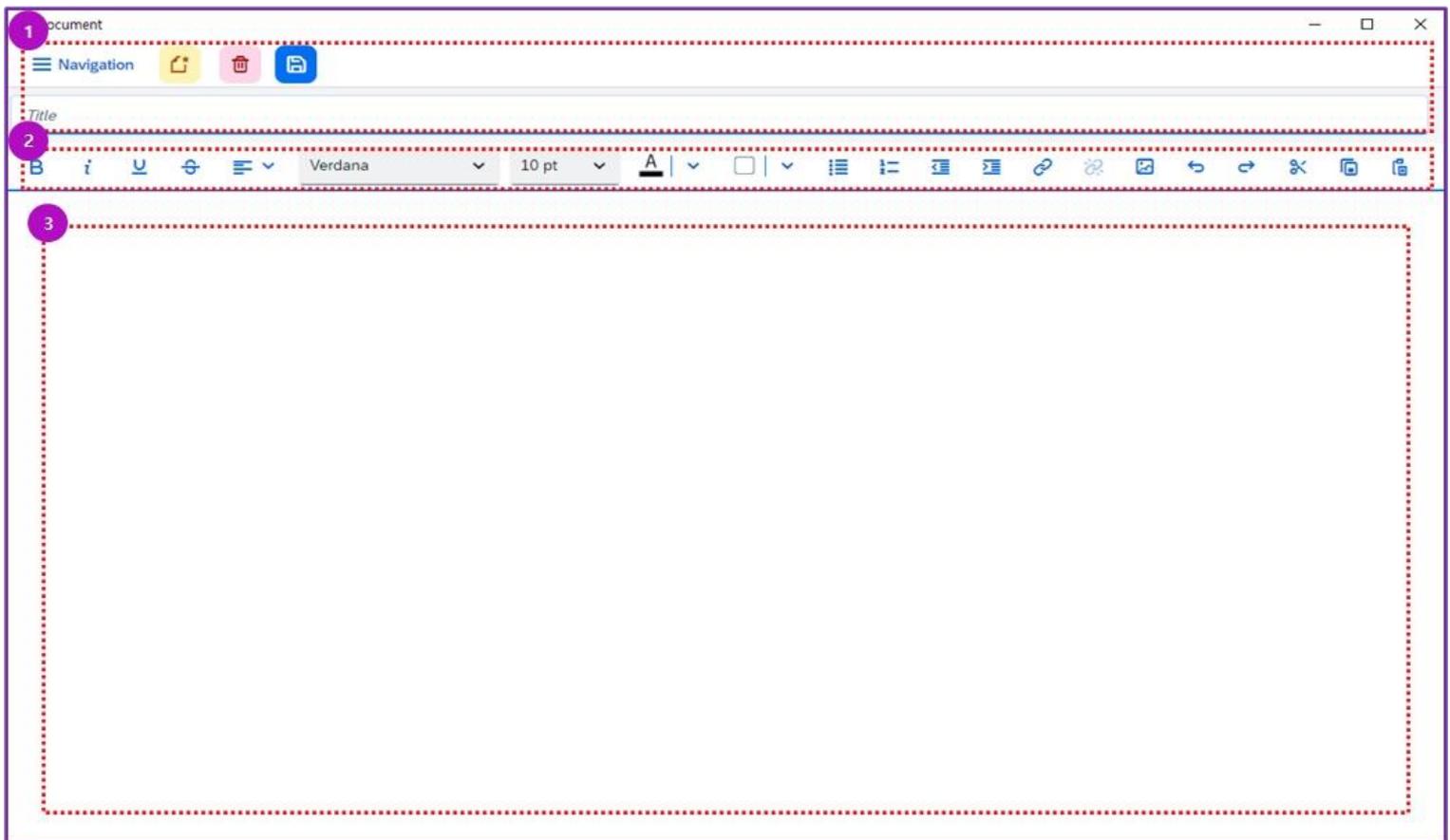
2. U4A의 가이드에 대한 Help Document 실행.

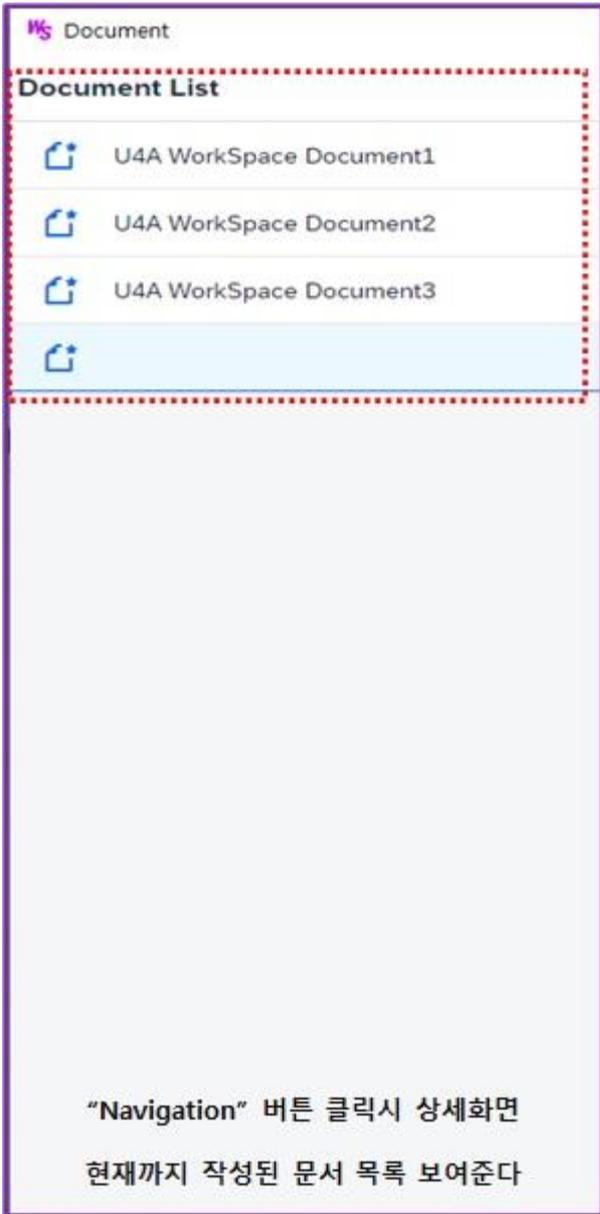
The screenshot shows a web browser window titled "U4A IDE HELP". The browser's address bar and navigation buttons are visible at the top. On the left side, there is a sidebar menu with the following items: "Instruction", "개발 환경 구성", "U4A IDE Platform Beginning", and "U4A IDE Workbench 사용 방법". The main content area displays the "Cover Page" of the "U4A IDE Platform Help". The cover page features a background image of a laptop and a smartphone on a desk. A central text box contains the following text: "SAP UI5기반 시스템 구축 개발을 위한 U4A IDE Platform Help openUI5 Application Development Solution for SAP Developers". Below this, it says "Part - 1". At the bottom of the cover page, there is contact information for "인포씨지" (INFOCG) and a "SAP Certified" logo. The footer of the page contains copyright information: "Copyright 2017. INFOCG Inc. all rights reserved. 'ABAP®', 'SAP NetWeaver®', 'SAP S/4HANA®' are the trademark(s) or registered trademark(s) of SAP SE in Germany and in several other countries."

5.5.2 Document



2. Document 개요.





(1). 문서 목록, 문서 제목, 문서 생성, 문서 삭제, 문서 저장 하는 영역

(2). (3)영역에 문서 작성시 글자 굵기, 기울기, 폰트, 폰트 사이즈 등을 설정하는 영역



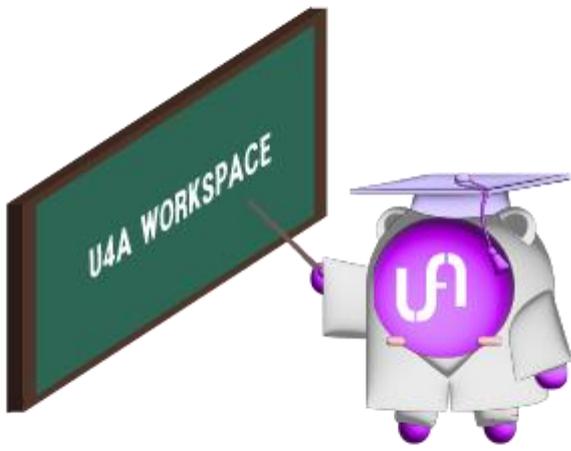
① 텍스트 굵기, 기울기, 밑줄, 취소선

② 텍스트 정렬

- ③ 텍스트 Font 및 크기 변경
- ④ 텍스트 색상 변경, 텍스트 배경색상 변경
- ⑤ 맨 앞의 줄마다 점 또는 번호 추가
- ⑥ 줄의 들여쓰기,내어쓰기
- ⑦ 링크 추가, 링크삭제, 이미지 첨부
- ⑧ 실행 취소 또는 되돌리기
- ⑨ 잘라내기, 복사, 붙여넣기

(3). 문서 내용을 작성하는 영역

5.6 도구 모음



5.6.1 Hide

■ 목적 : U4A WorkSpace 화면 투명도를 조절하여 WorkSpace가 실행된 상태에서 다른 작업 수행하기 위한 용도.

1. "Hide" 실행 방법에는 2가지가 존재한다.

① U4A WorkSpace 메인 화면에서 붉은 영역 아이콘 선택



② Application 안 붉은 영역 아이콘 선택



2. 본 단원에서는 U4A WorkSpace 메인 화면에서 "Hide" 기능 수행

① 화면 투명도 조절 Pop-Up

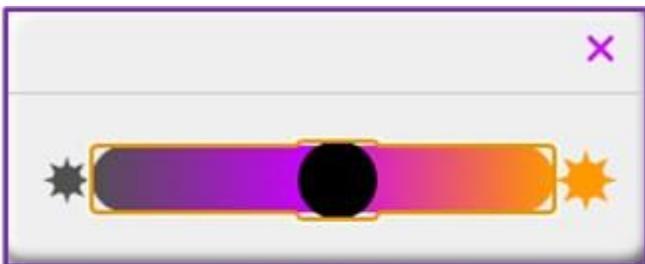


Figure : < 화면 투명도 조절 Pop-Up >



② 화면 투명도를 조절하여 U4A WorkSpace 실행 중 다른 작업 수행

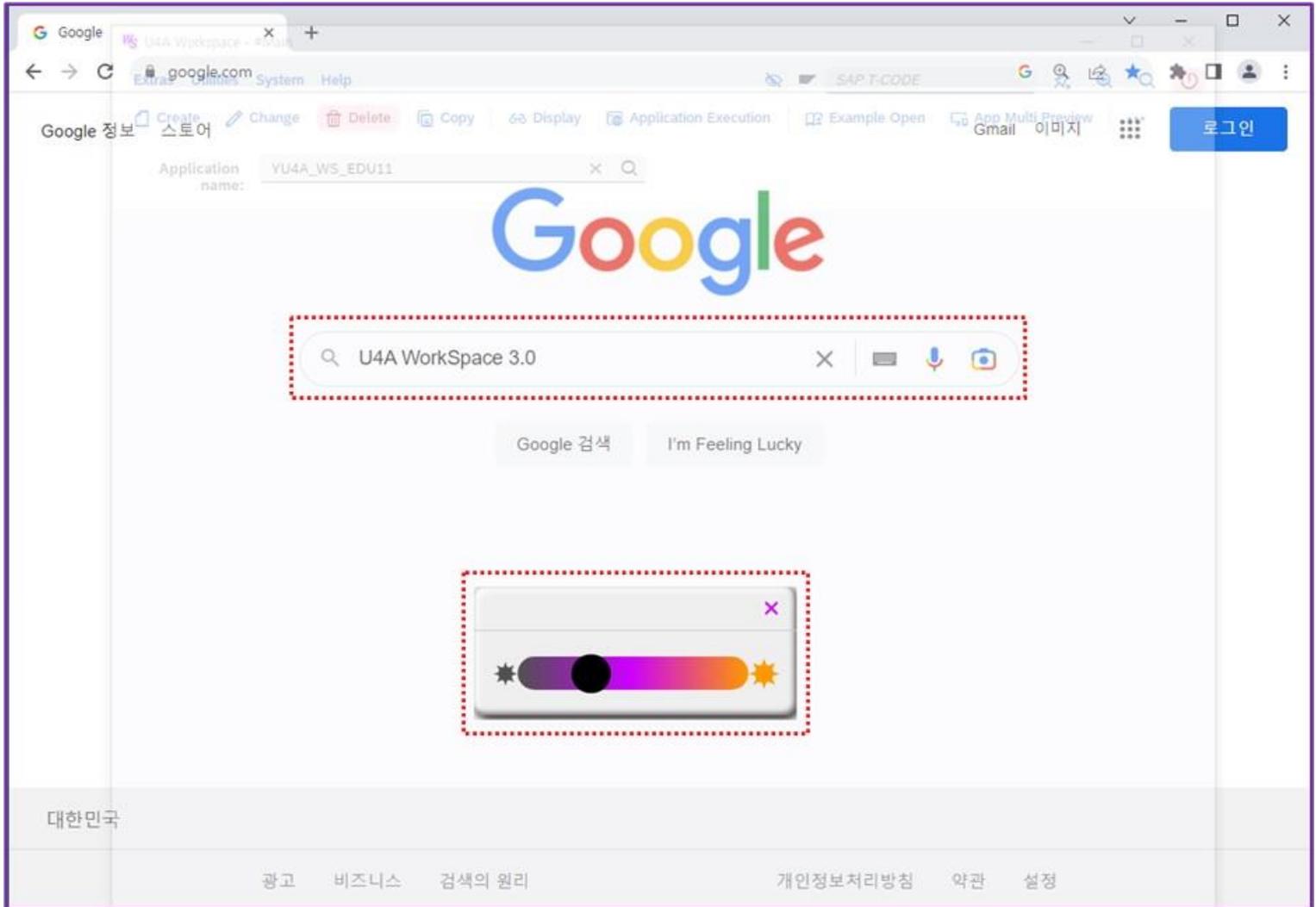


Figure : < 화면 투명도 조절 하여 WorkSpace 창 뒤에있는 브라우저 이용한 모습 >

5.6.2 T-Code

- ① U4A WorkSpace 메인 화면에서 붉은 영역에 T-Code 입력



- ② Application 안 붉은 영역에 T-Code 입력



2. 본 단원에서는 U4A WorkSpace 메인 화면에서 "T-Code" 입력 후 Enter = > 프로그램 실행.

U4A IDE HELP

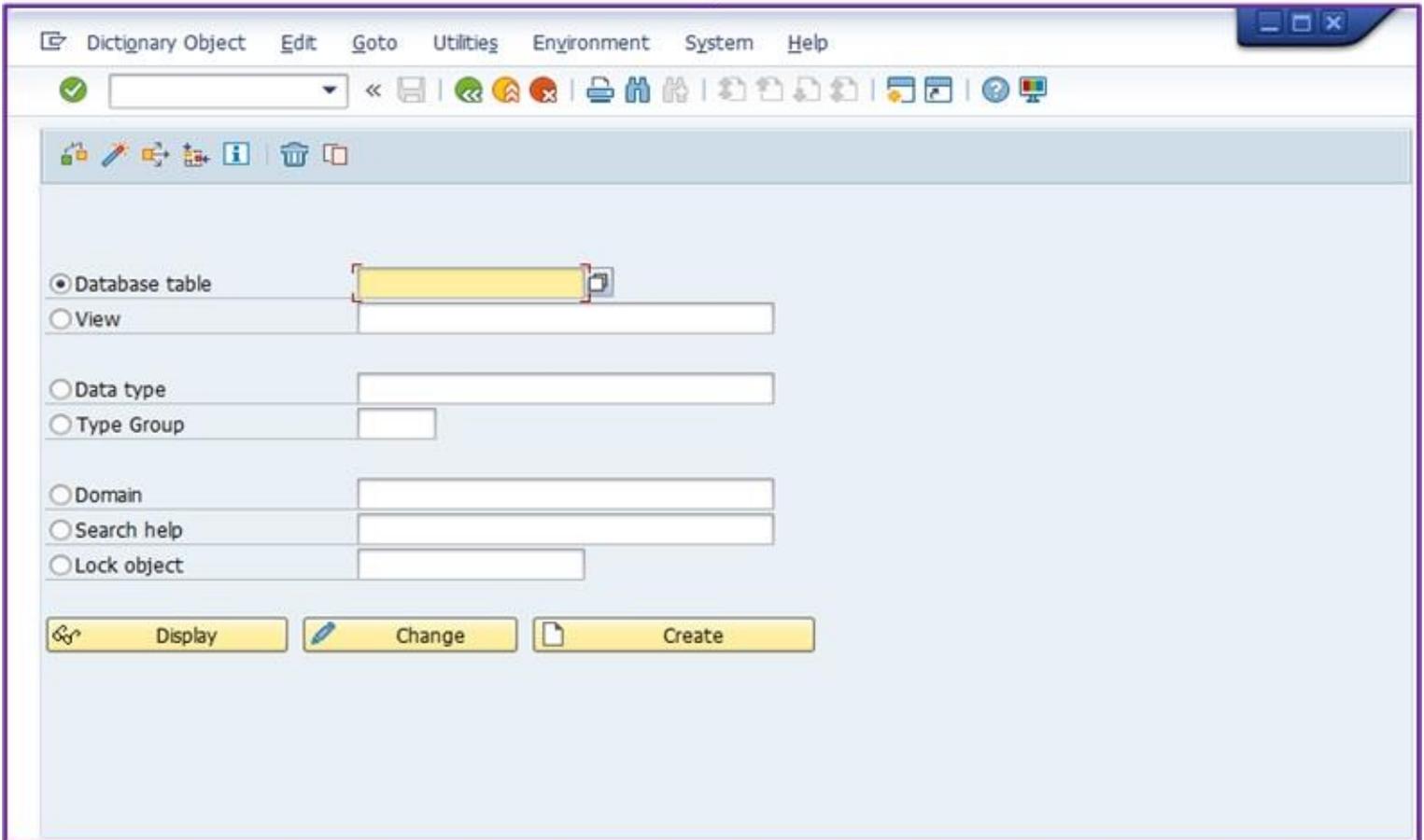
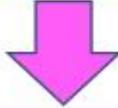
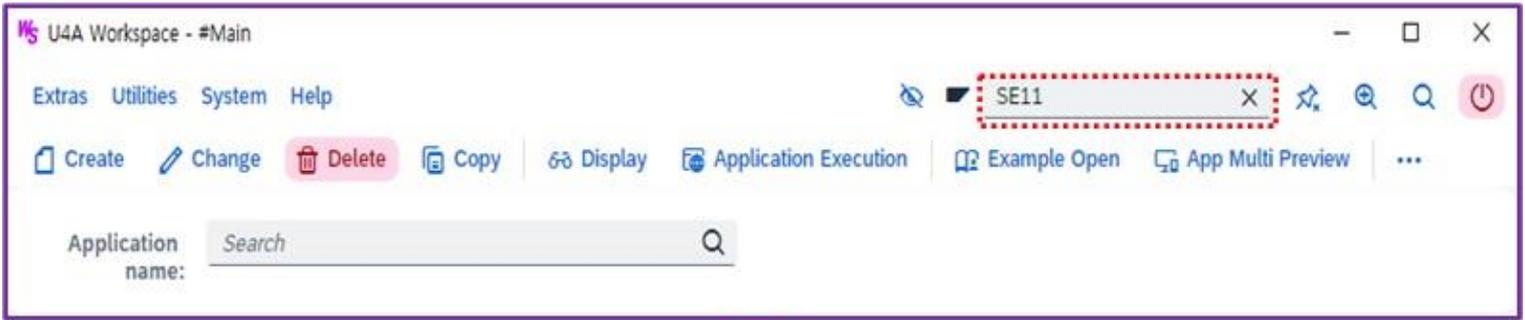


Figure : < ABAP Dictionary 호출 >

5.6.3 Browser Pin

① U4A WorkSpace 메인 화면에서 붉은 영역 아이콘 선택



② Application 안 붉은 영역 아이콘 선택



2. 본 단원에서는 U4A WorkSpace 메인 화면에서 "Browser Pin" 기능 수행

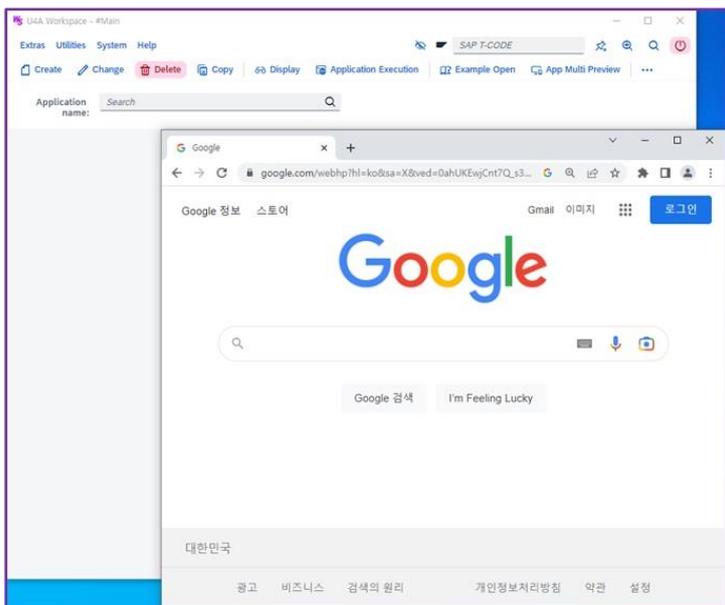


Figure : < Browser Pin 기능 미적용 상태 >

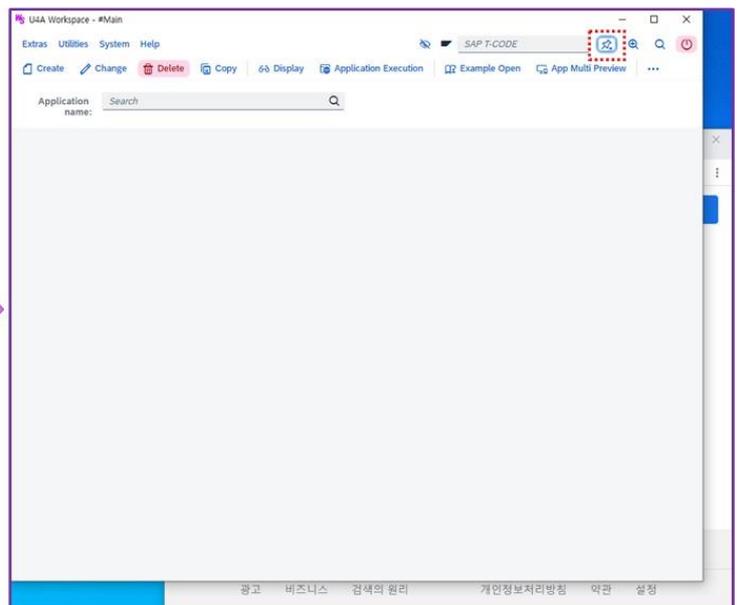


Figure : < Browser Pin 기능 적용 상태 >

5.6.4 Zoom

■ 목적 : U4A WorkSpace 화면 확대 및 축소

1. "Zoom" 실행 방법에는 2가지가 존재한다.

① U4A WorkSpace 메인 화면에서 붉은 영역 아이콘 선택



② Application 안 붉은 영역 아이콘 선택



2. 본 단원에서는 U4A WorkSpace 메인 화면에서 "Zoom" 기능 수행

① Zoom(확대 및 축소) Pop-Up



Figure : < Zoom(확대 및 축소) Pop-Up >



② "확대" 한 모습

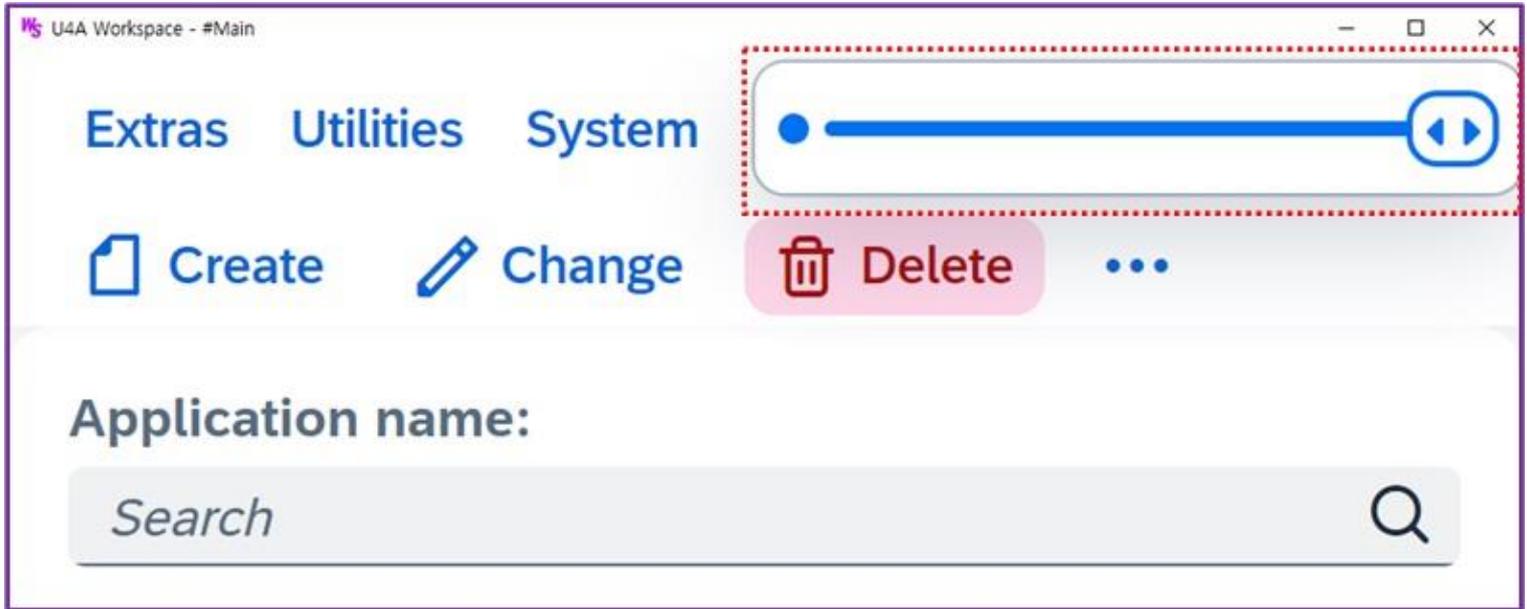


Figure : < 화면 최대로 확대 >

② "축소" 한 모습

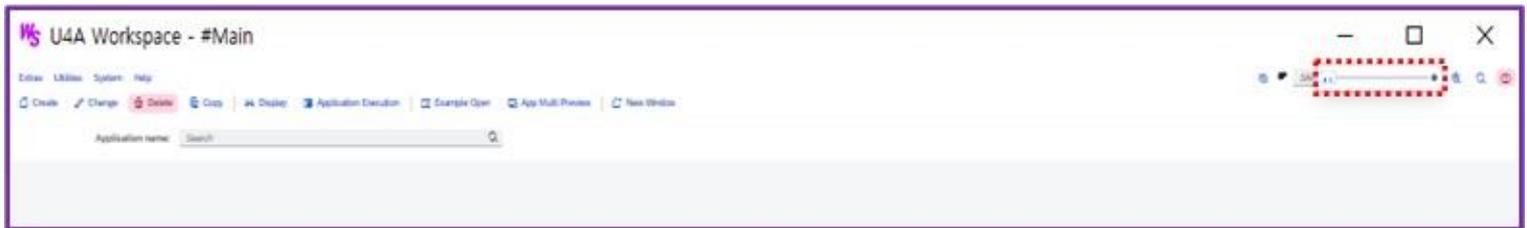


Figure : < 화면 최대로 축소 >

5.6.5 Window Text Search

■ 목적 : 키워드를 입력하여 U4A WorkSpace 화면에 존재하는 Text를 빠르게 찾기 위함

1. "Window Text Search" 실행 방법에는 2가지가 존재한다. (단축키 = > CTRL + SHIFT + R)

① U4A WorkSpace 메인 화면에서 붉은 영역 아이콘 선택



② Application 안 붉은 영역 아이콘 선택



2. 본 단원에서는 U4A WorkSpace 메인 화면에서 "Window Text Search" 기능 수행

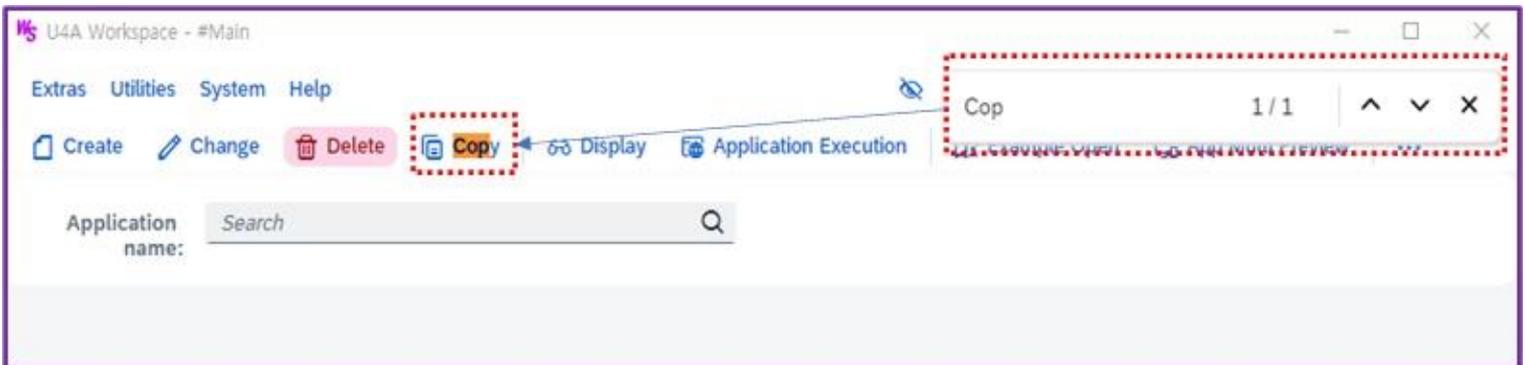
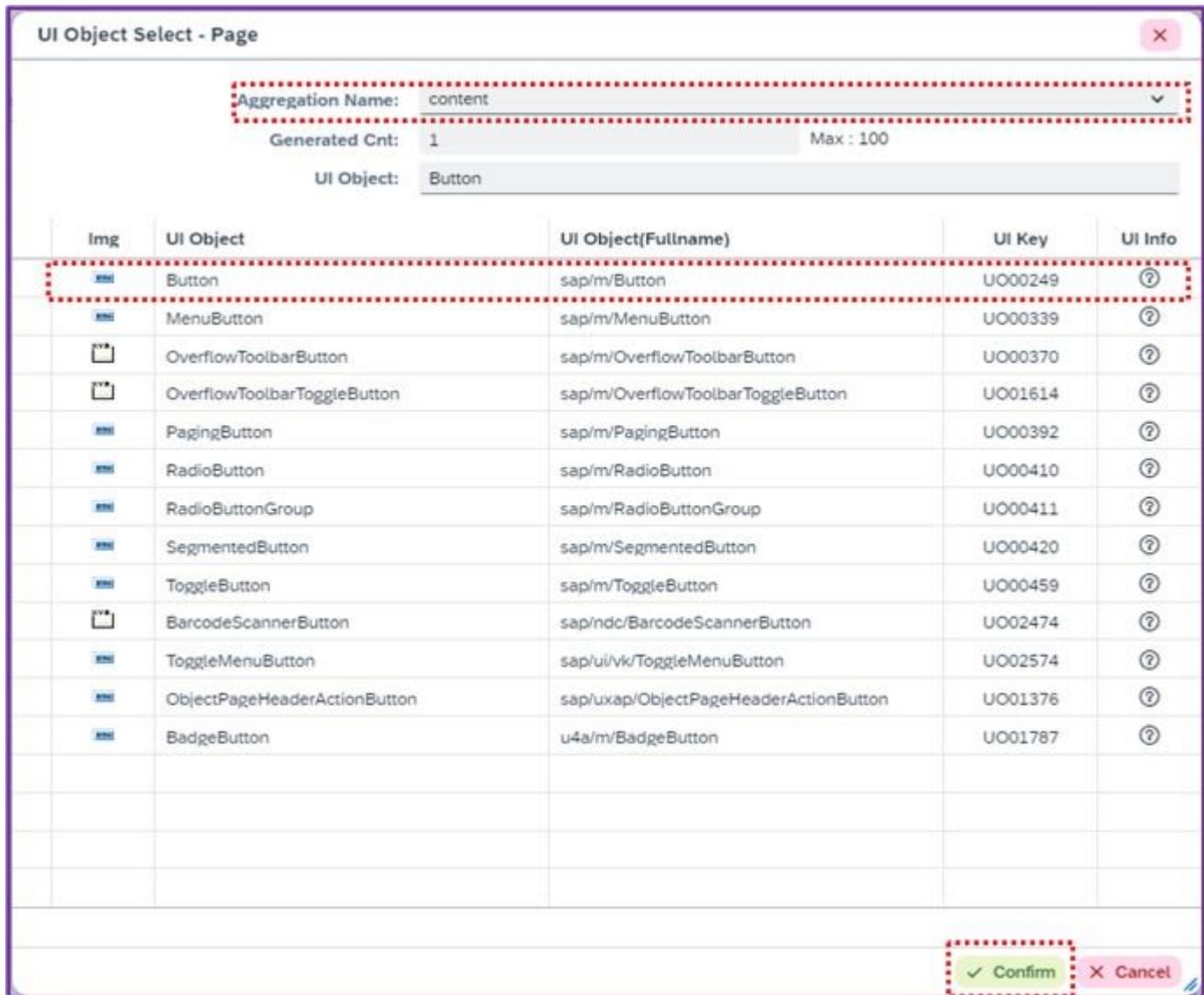


Figure : < Window Text Search에 Text 입력시 해당 하는 Text 백그라운드 색상 변경 >

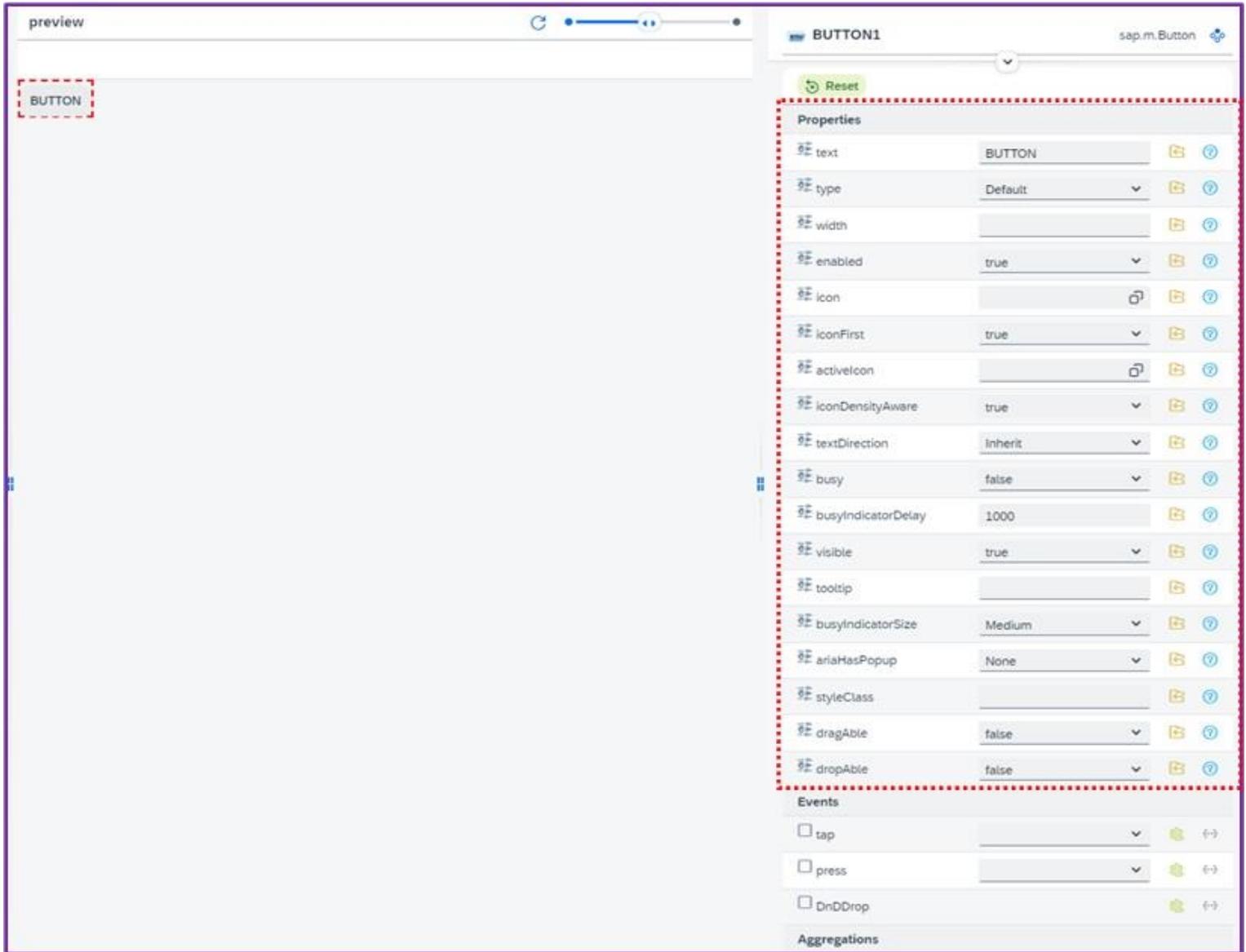
5.7 Button 사용 방법

■ 목적 : 마우스로 Click 해서 Event를 작동 시키기 위해 사용한다.

1. 신규 생성할 Button을 원하는 위치에서 생성한다.

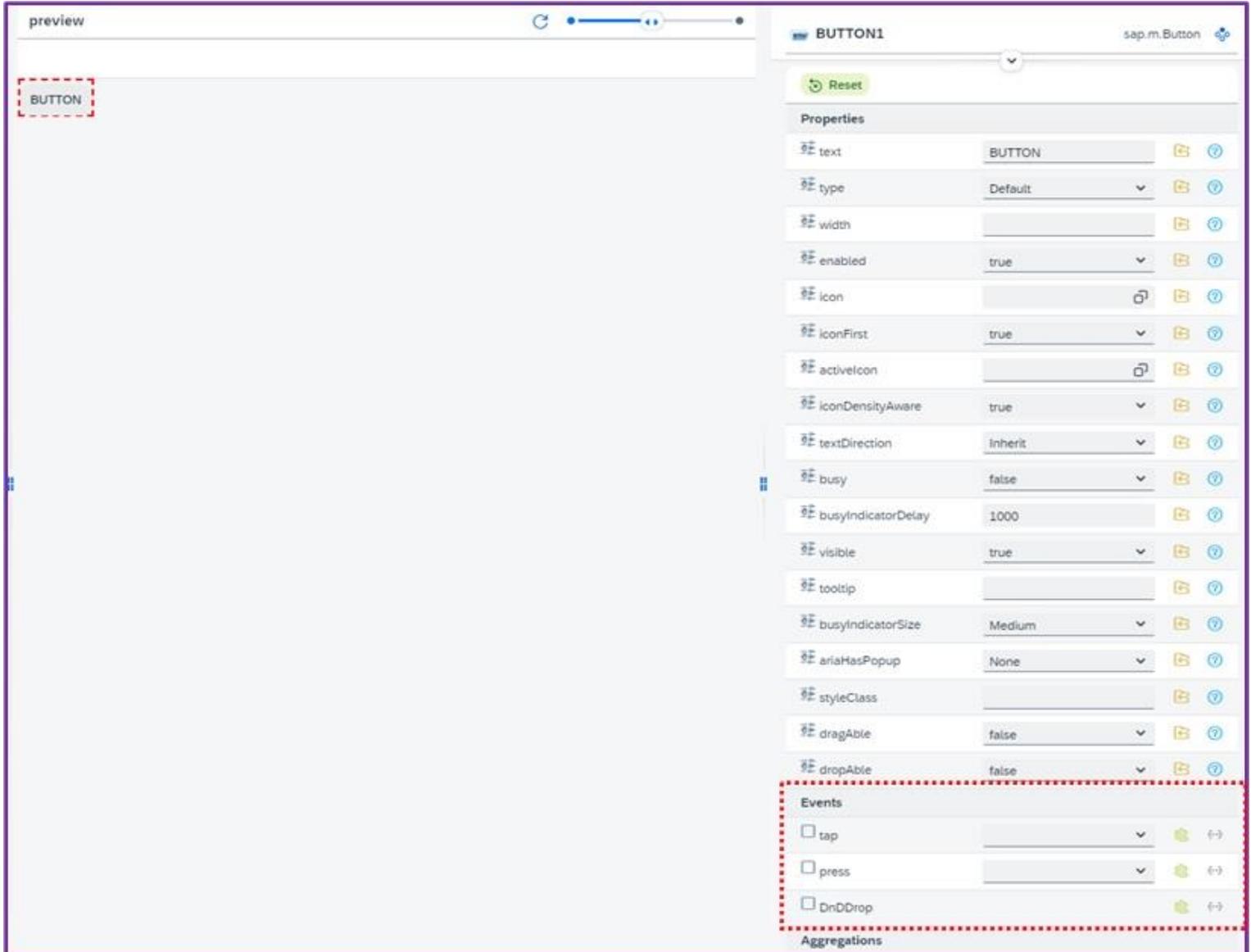


2. Button생성 후 Properties에 원하는 기능을 넣을 수 있다.



Option Property	Means
text	Button의 원하는 Text 입력
type	다른 모양의 Button으로 변경
width	Button의 가로폭 조정 단위는 "px", "em", "%"
enabled	true일 경우 Button Click 가능 상태 false일 경우 Button Click 불가능 상태
icon	Button에 원하는 Image 삽입
visible	true일 경우 Button 보이는 상태 false일 경우 Button 안보이는 상태

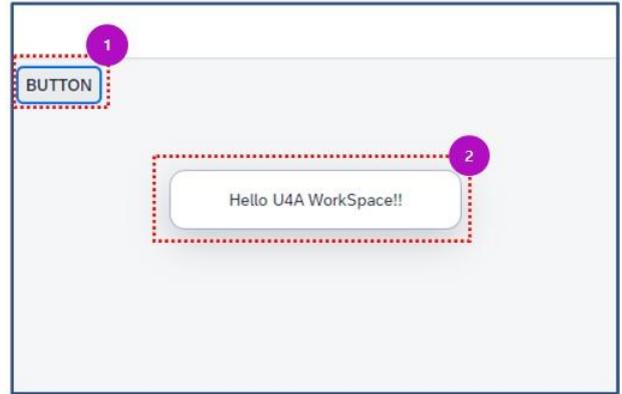
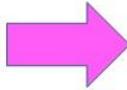
3. Button에 마우스로 Click시 특정 동작을 수행하려면 Events의 press항목에 Methods를 적용시켜준다.



4. Methods에 Event를 구성하고 Web에서 Button을 누르면 Message가 나타난다.

U4A IDE HELP

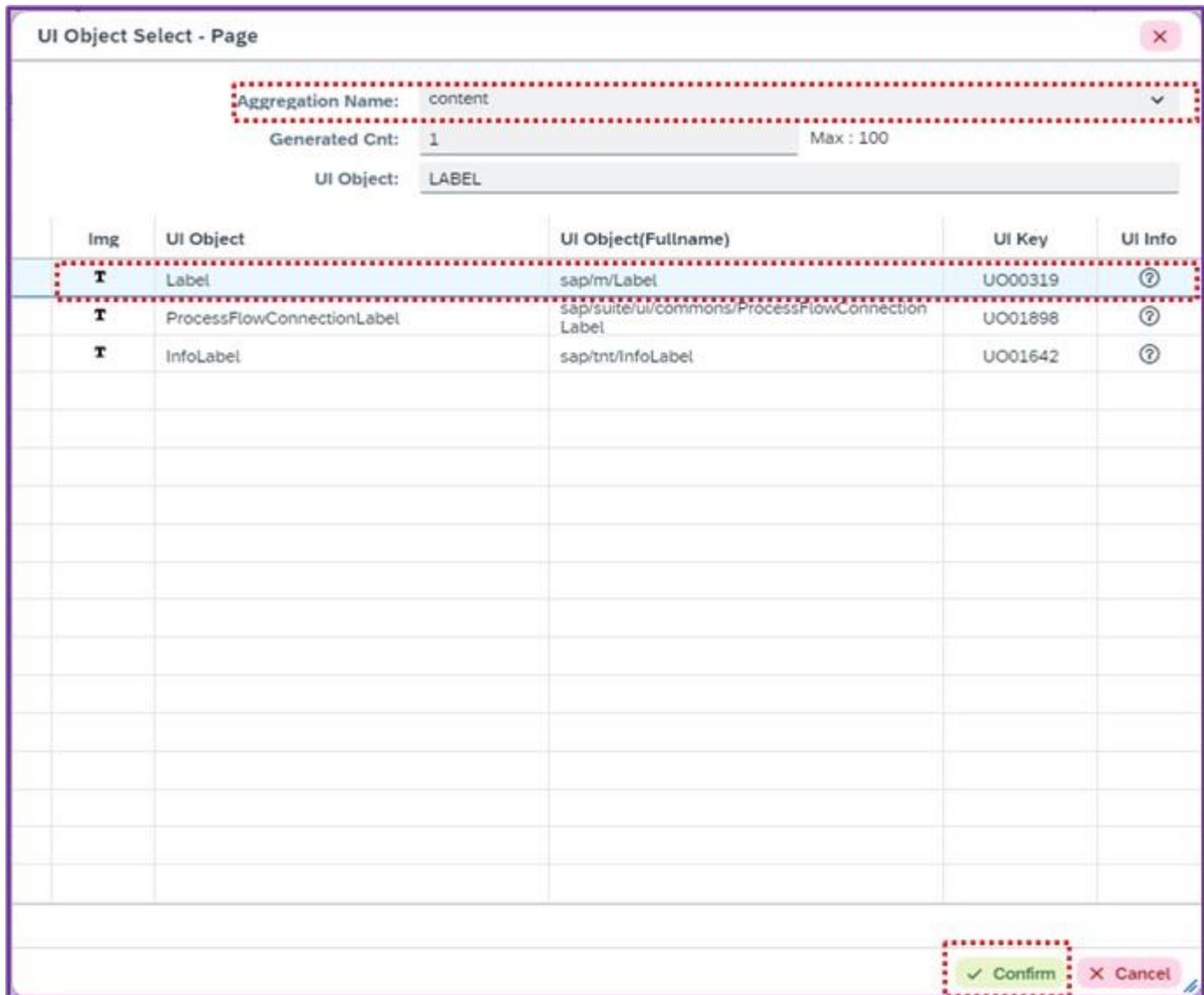
```
Method EV_BUTTON_PRESS active
1 METHOD EV_BUTTON_PRESS.
2
3 DATA LV_MSG TYPE STRING.
4
5
6 "MESSAGE 내역
7 LV_MSG = 'Hello U4A WorkSpace!!'.
8
9 "MESSAGE 호출
10 CALL METHOD /U4A/CL_UTILITIES=>M_MESSAGE_TOAST
11 EXPORTING
12 IO_VIEW = ME->/U4A/IF_SERVER-AR_VIEW " [U4A] UI Element Super Class
13 IS_MSG = " Structure of message variables
14 I_MSGTX = LV_MSG " MESSAGE TEXT
15 IT_MSGTX = " Table of Strings
16 I_VPOS = CS_M_MSG_TOS_VPOS-CENTER " Vertical position
17 I_HPOS = CS_M_MSG_TOS_HPOS-CENTER " Horizontal position
18 I_VOFF = '0' " Vertical offset
19 I_HOFF = '0' " Horizontal offset
20 I_DURATION = 10000
21
22
23 ENDMETHOD.
```



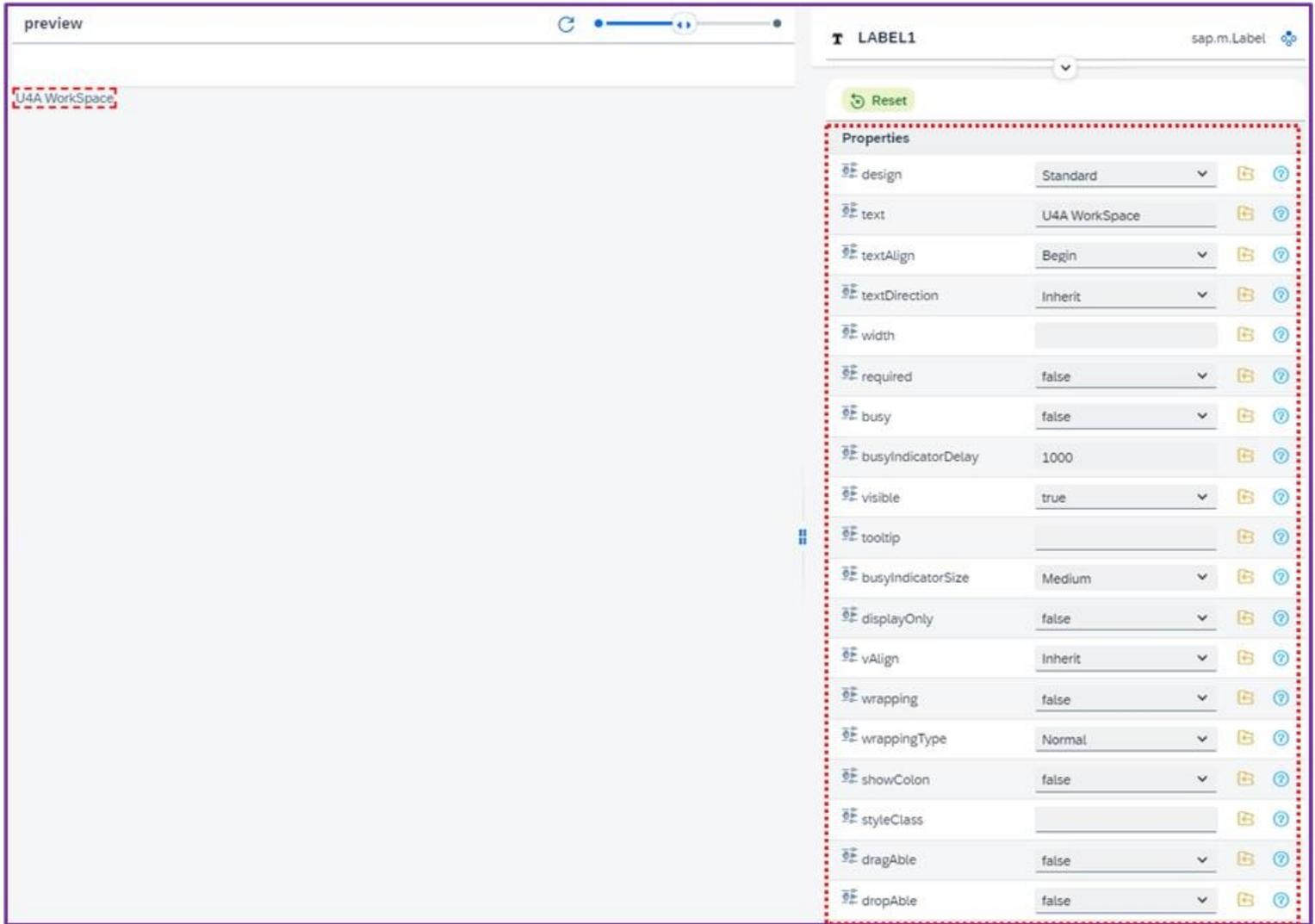
5.8 Label 사용 방법

■ 목적 : 다른 UI Obj의 Title형태로 만들어 주기 위해 사용한다.

1. 신규 생성할 Label을 원하는 위치에서 생성한다.

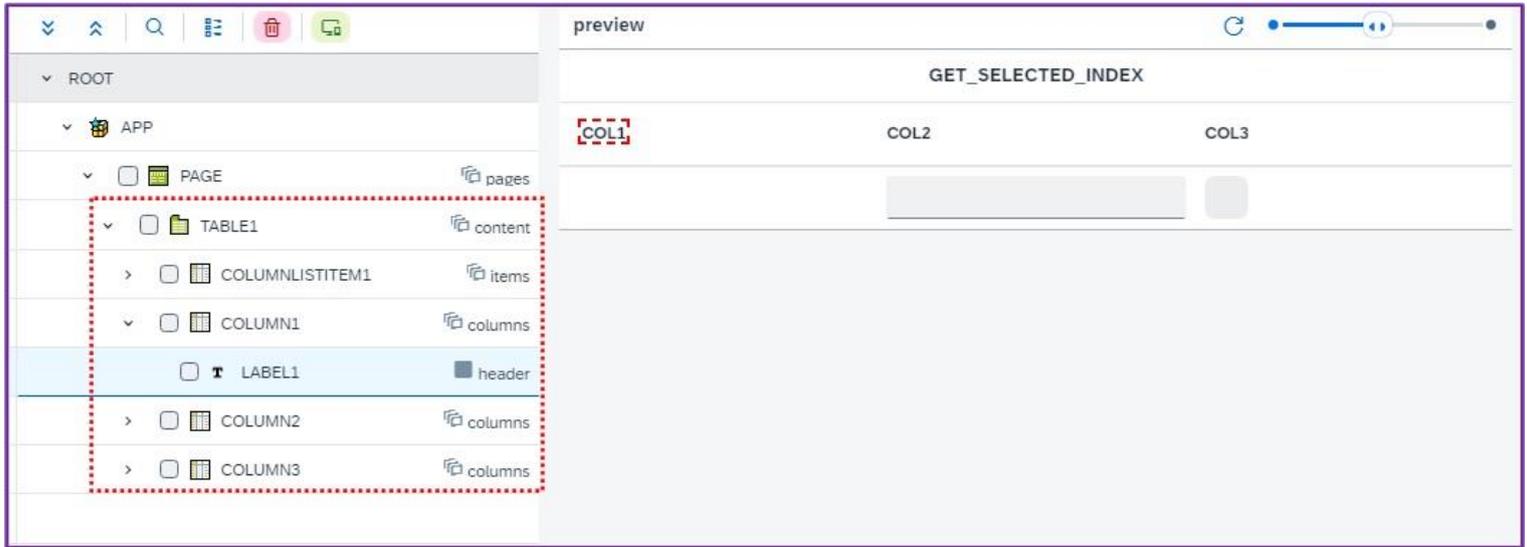


2. Label 생성 후 Properties에 원하는 기능을 사용한다.



Option Property	Means
text	Label에 표시될 Text 입력
design	Label의 Text Style을 굵은 글씨체로 변경 가능
width	Label의 가로폭 조정 단위는 "px", "em", "%"
visible	true일 경우 Label 보이는 상태, false일 경우 Label 안보이는 상태

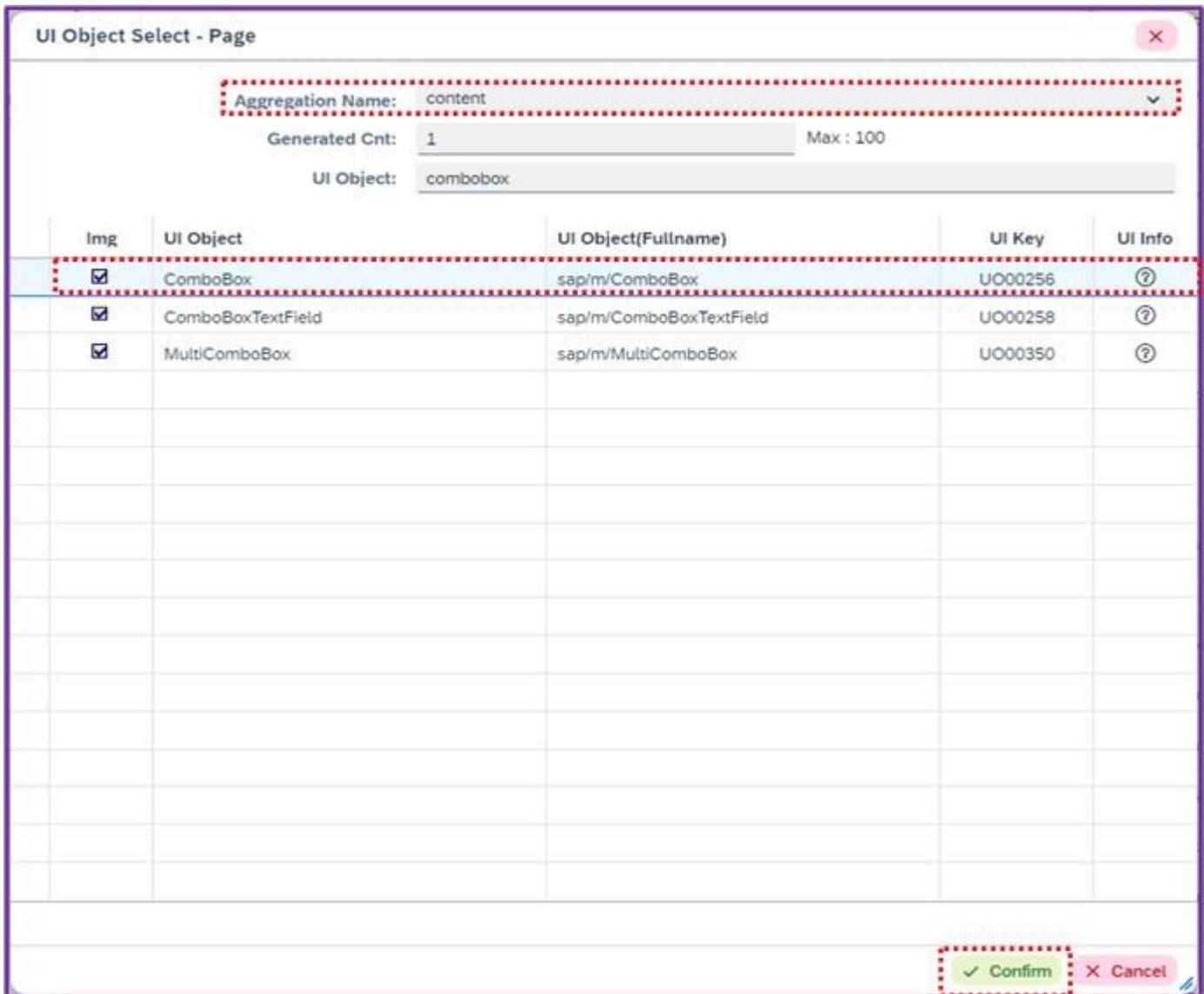
3. Label은 주로 다른 object의 제목을 입력할때 사용한다.



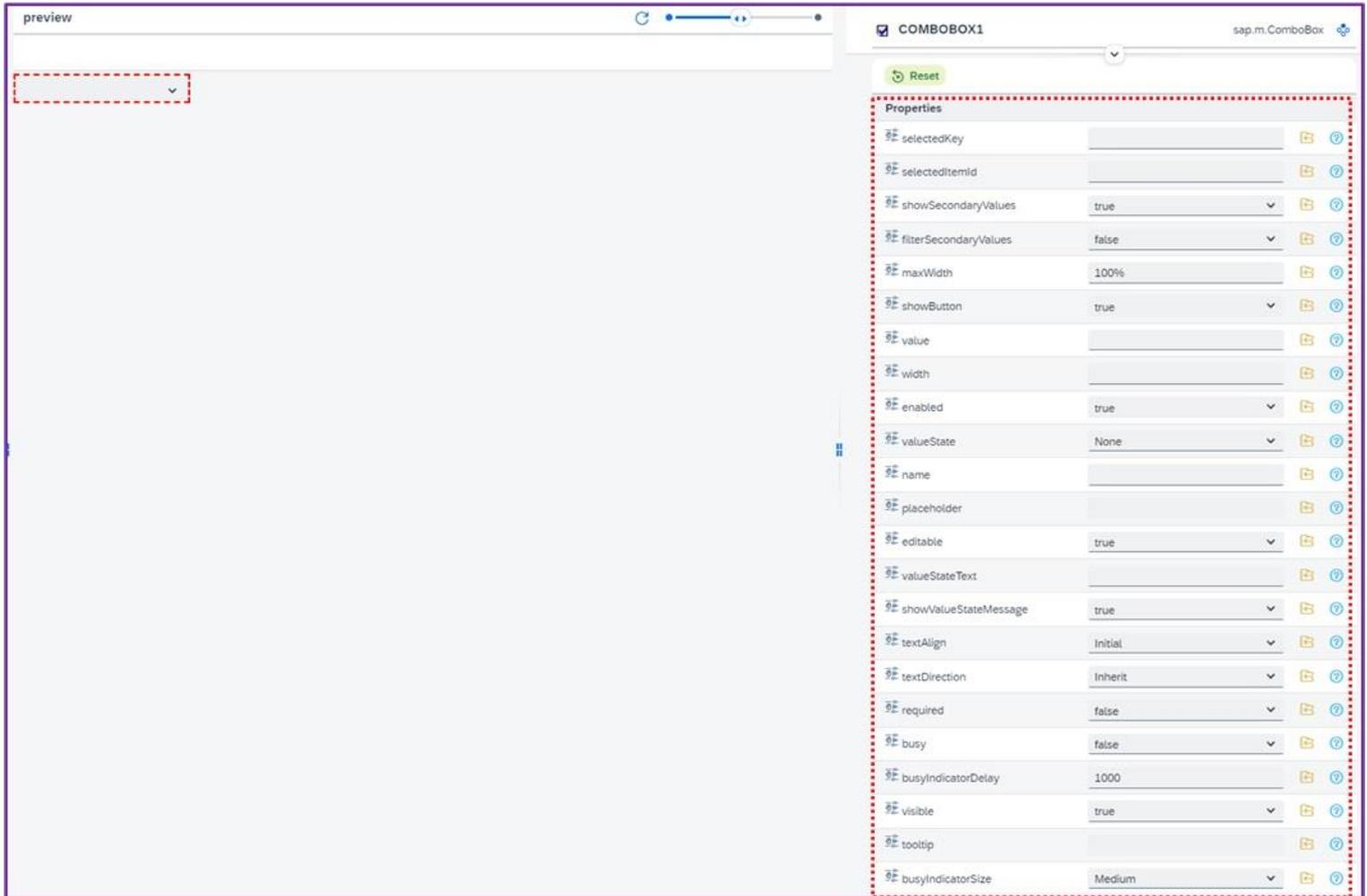
5.9 ComboBox 사용 방법

- 목적 : 정해진 값을 List 형태로 보여주고 선택 가능한 Input을 사용 할 수 있다.

1. 신규 생성할 ComboBox를 원하는 위치에 생성한다.

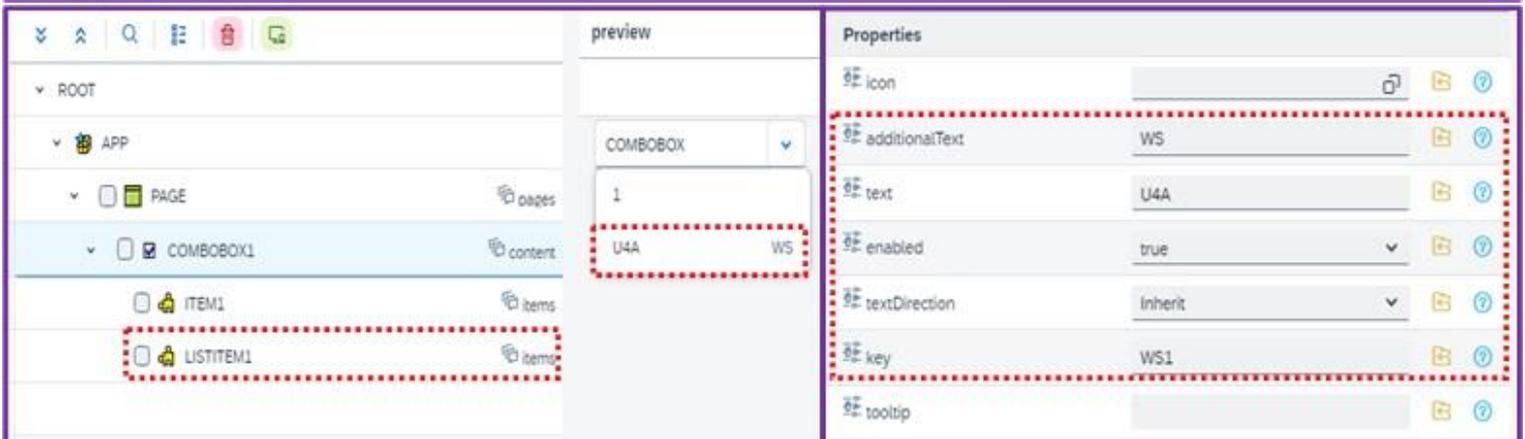
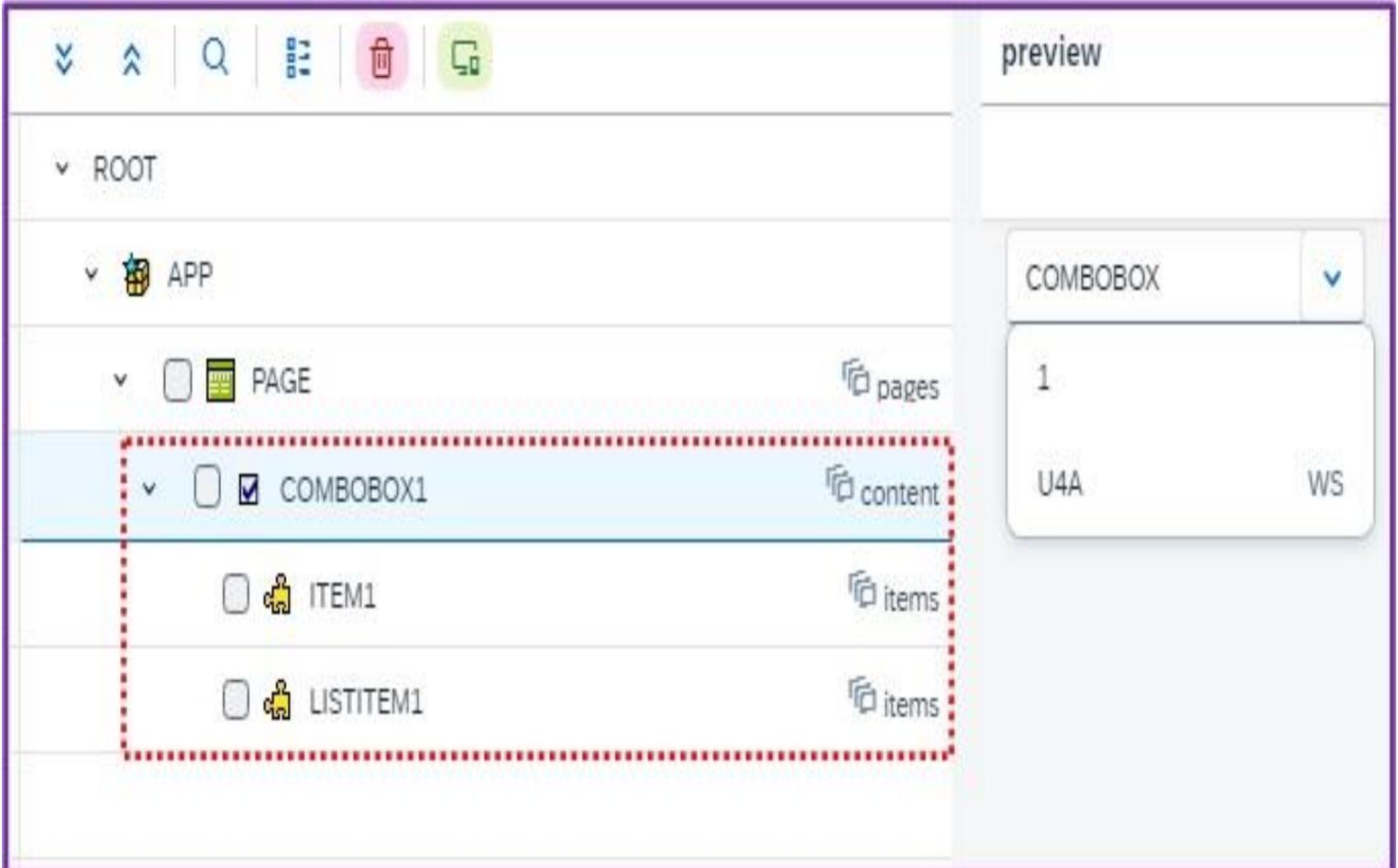


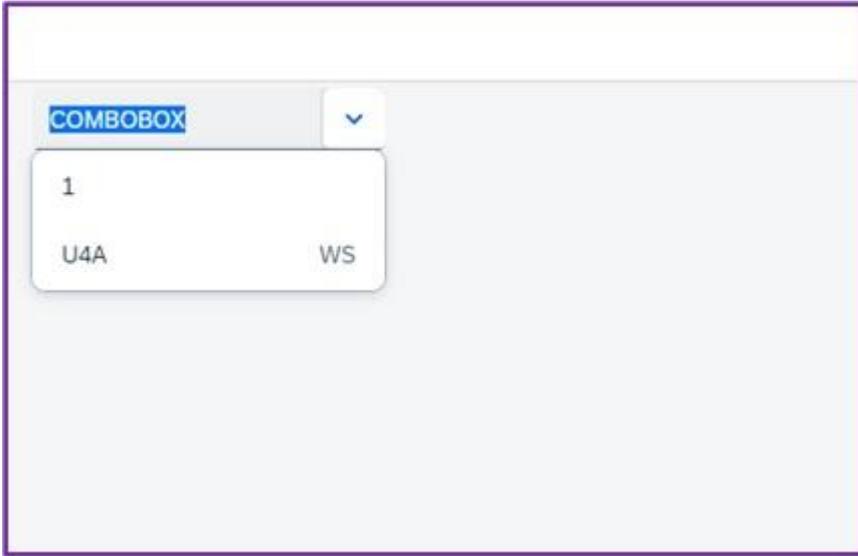
2. ComboBox 생성 후 Properties에 원하는 기능을 넣을 수 있다.



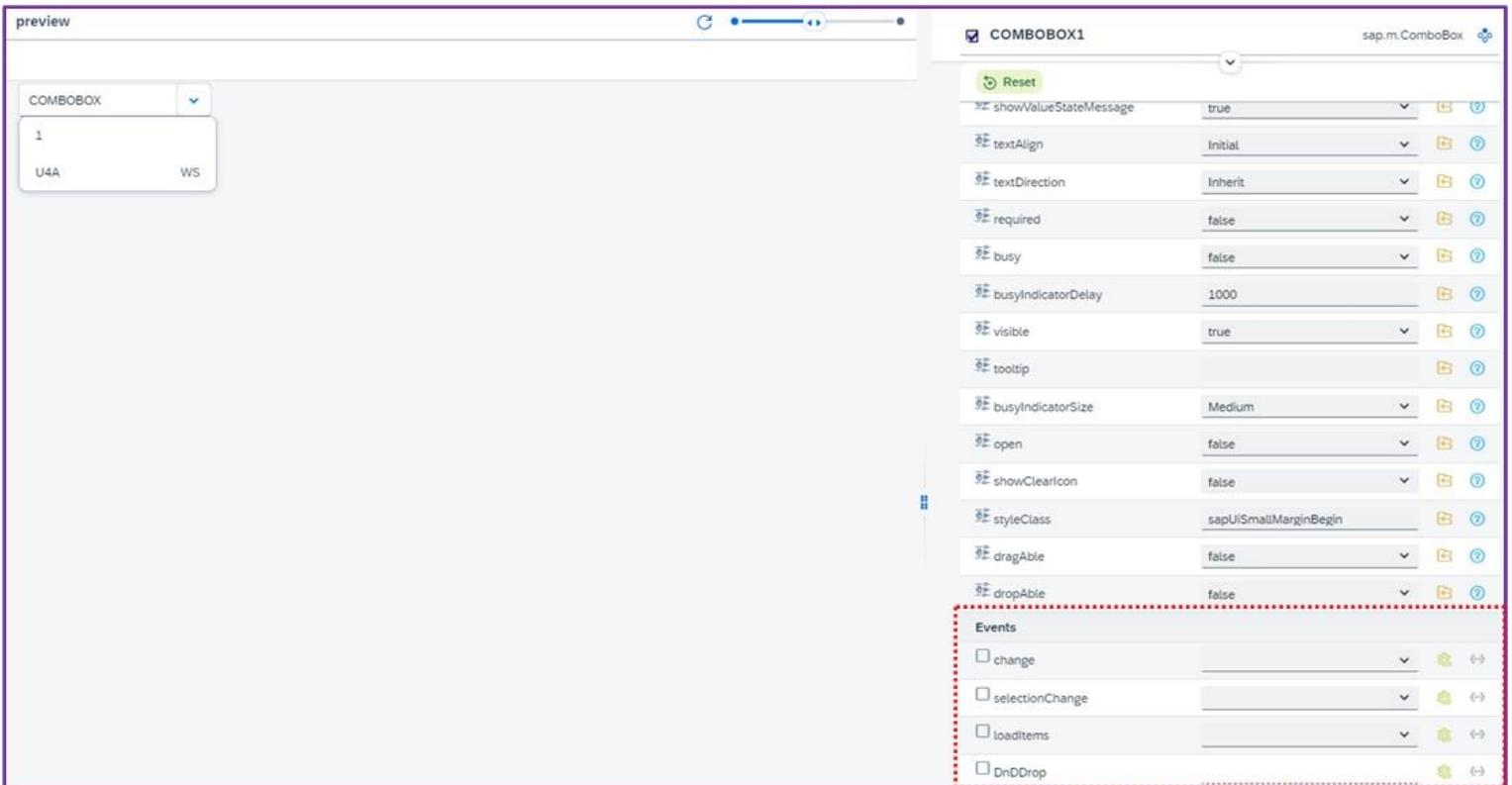
Option Property	Means
selectedKey	Key값을 Binding 해서 가져 올 때 사용
showButton	ComboBox의 item 선택 할 수 있는 오른쪽 Button 활성화/비활성
filterSecondaryValues	ComboBox에 sap.ui.core.ListItem 적용시 text, additionalText값 둘다 보여줄때 사용
value	ComboBox의 초기값 입력
width	ComboBox 가로폭 조정 단위는 "px", "em", "%"
Enabled	ComboBox의 수정 활성화/비활성 (전체적으로 반투명하게 변화)

3. ComboBox에 원하는 만큼 Item을 생성하고 key, value값을 binding 하여 값을 넣어준다.



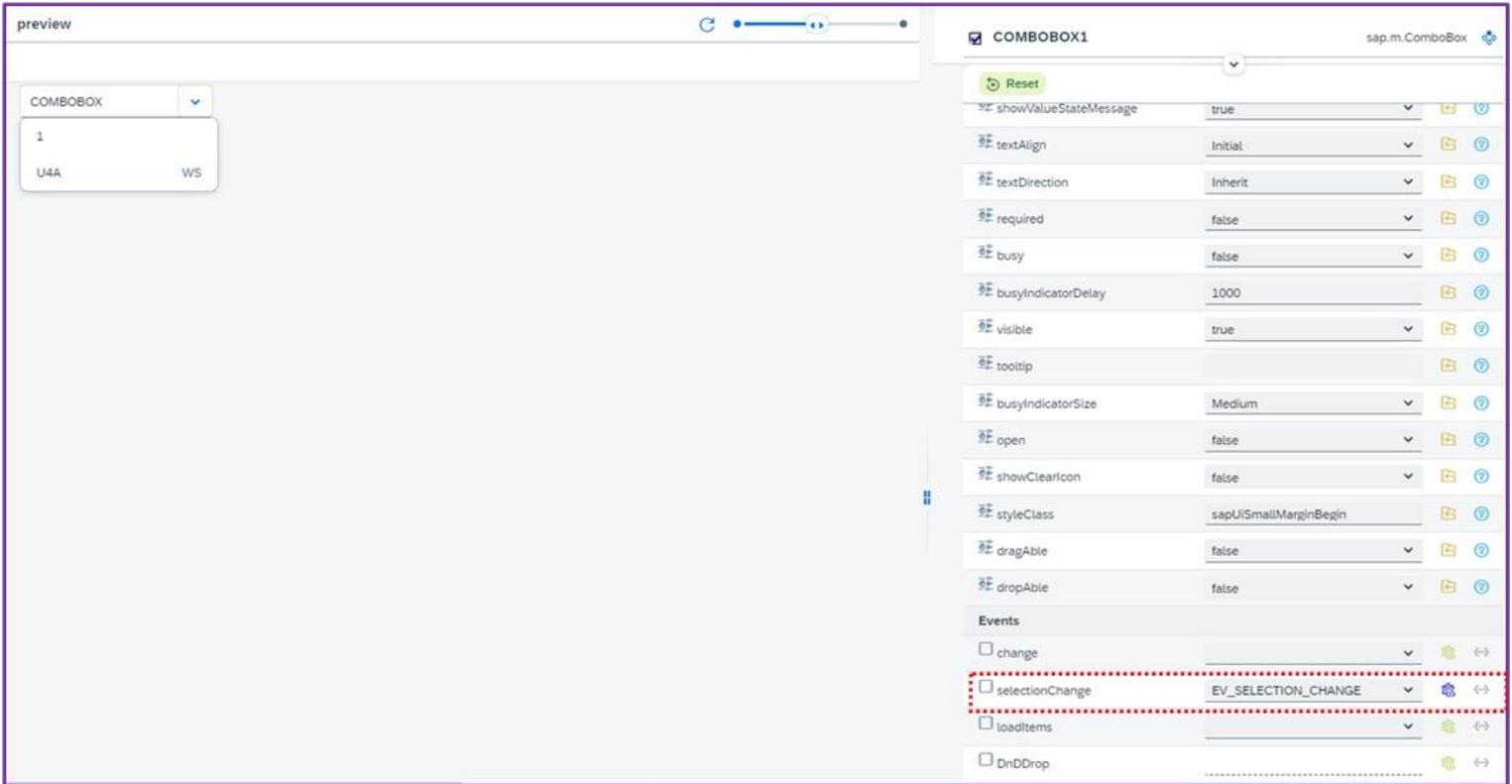


4. ComboBox에 특정 동작을 수행하기 위해 Events의 Methods를 적용시켜준다.



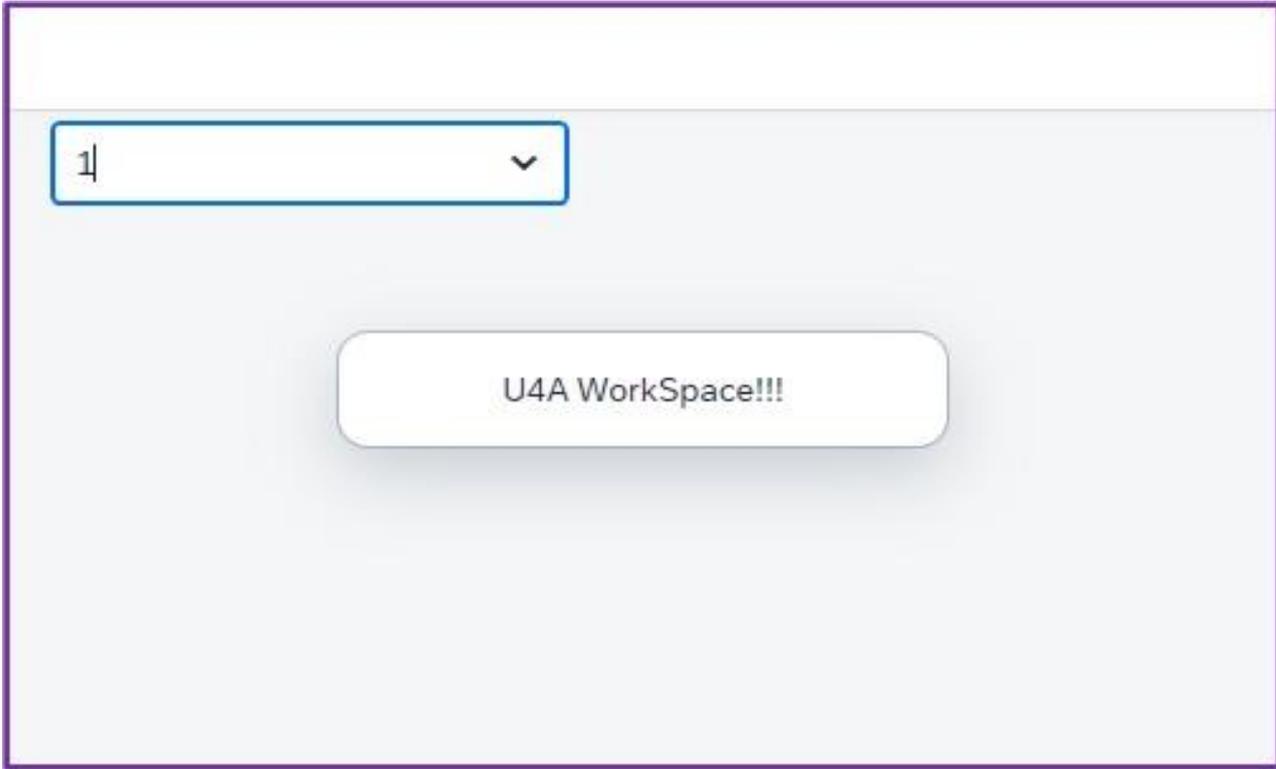
Events	Means
change	Enter를 누르거나 목록의 항목이 선택 될 시 작동
selectionChange	ComboBox의 item의 Key값이 변경될 때 작동

5. Methods에 Messagetoast를 구성하고 Web에서 자신이 정한 동작을 수행하면 Message가 나타난다.



```

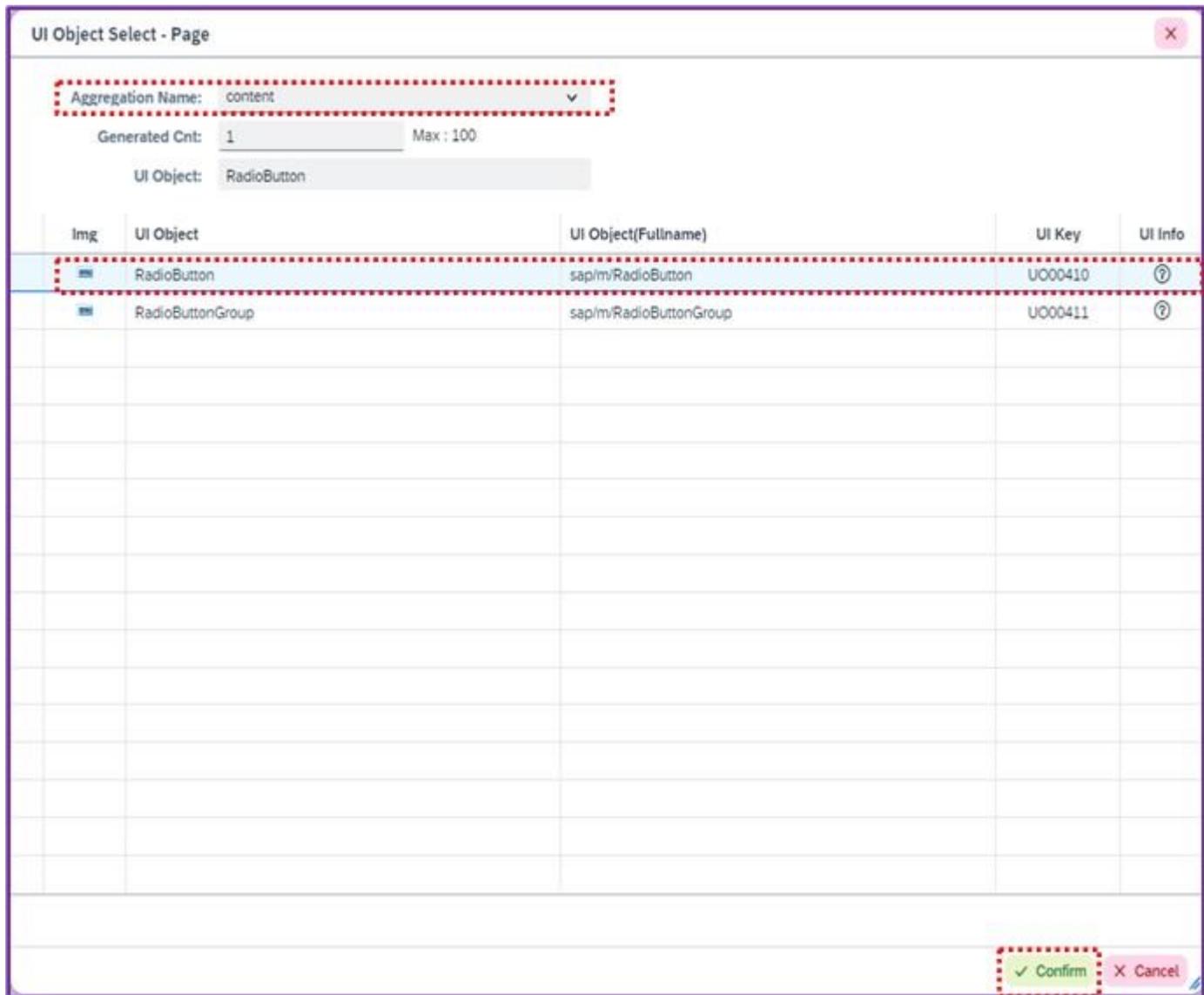
Method      EV_SELECTION_CHANGE      active
1  METHOD EV_SELECTION_CHANGE.
2
3      DATA LV_MSG TYPE STRING.
4      *-----*
5
6      "MESSAGE 내역
7      LV_MSG = 'U4A WorkSpace!!!'.
8
9      "MESSAGE 호출
10     CALL METHOD /U4A/CL_UTILITIES=>M_MESSAGE_TOAST
11         EXPORTING
12             IO_VIEW = ME->/U4A/IF_SERVER-AR_VIEW " [U4A] UI Element Super Class
13             * IS_MSG = " Structure of message variables
14             I_MSGTX = LV_MSG " MESSAGE TEXT
15             * IT_MSGTX = " Table of Strings
16             * I_VPOS = CS_M_MSG_TOS_VPOS-CENTER " Vertical position
17             * I_HPOS = CS_M_MSG_TOS_HPOS-CENTER " Horizontal position
18             * I_VOFF = '0' " Vertical offset
19             * I_HOFF = '0' " Horizontal offset
20             * I_DURATION = 10000
21
22
23     ENDMETHOD.
    
```



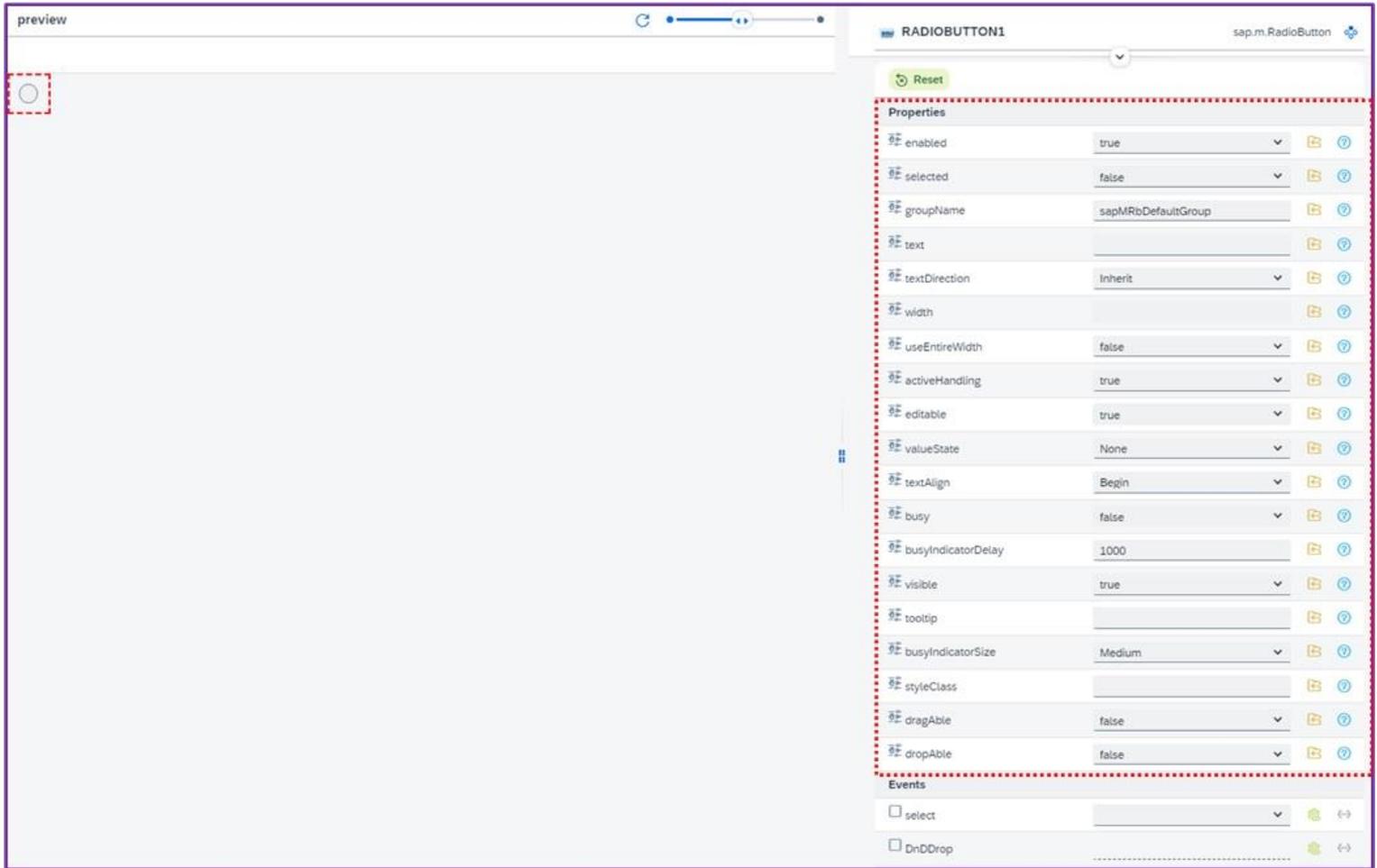
5.10 RadioButton 사용 방법

■ 목적 : 미리 정의 된 option 집합 중 하나만 선택하기 위해 사용한다.

1. 신규 생성할 RadioButton을 원하는 위치에서 생성한다.



2. RadioButton생성 후 Properties에 원하는 기능을 사용한다.



Option Property	Means
text	RadioButton의 옆쪽에 Text 입력
selected	true일 경우 첫 시작부터 RadioButton 선택, false일 경우 Radiobutton 공백
width	Radiobutton의 가로폭 조정 단위는 "px", "em", "%"
visible	true일 경우 Radiobutton 보이는 상태, false일 경우 Radiobutton 안보이는 상태

3. RadioButton은 Click 시 Event를 수행.

U4A IDE HELP

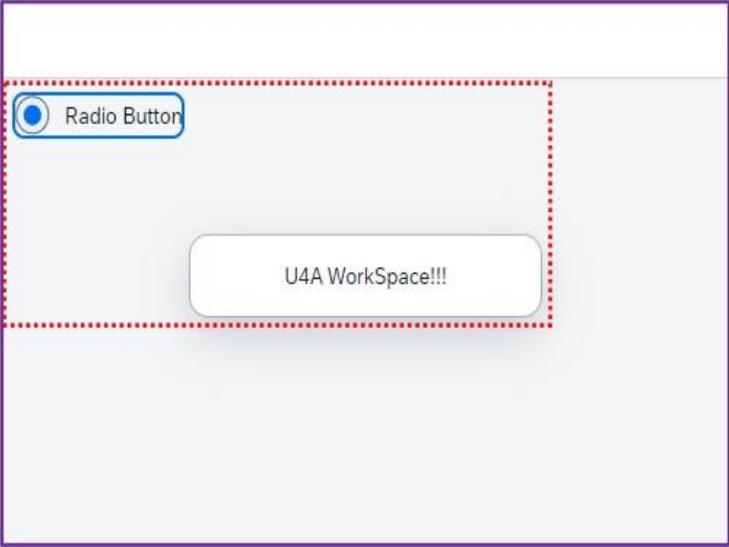
The screenshot shows the SAP IDE interface. On the left, a preview window displays a single radio button labeled "Radio Buttons". On the right, the "PROPERTIES" pane for the control "RADIOBUTTON1" (class "sap.m.RadioButton") is visible. The "Events" section at the bottom of the properties pane is highlighted with a red dashed box, showing the "select" event with the handler "EV_SELECT_RADIO".

Properties	
enabled	true
selected	false
groupName	sapMRbDefaultGroup
text	Radio Button
textDirection	Inherit
width	
useEntireWidth	false
activeHandling	true
editable	true
valueState	None
textAlign	Begin
busy	false
busyIndicatorDelay	1000
visible	true
tooltip	
busyIndicatorSize	Medium
styleClass	
dragAble	false
dropAble	false

Events	
<input type="checkbox"/> select	EV_SELECT_RADIO
<input type="checkbox"/> DnDDrop	

```

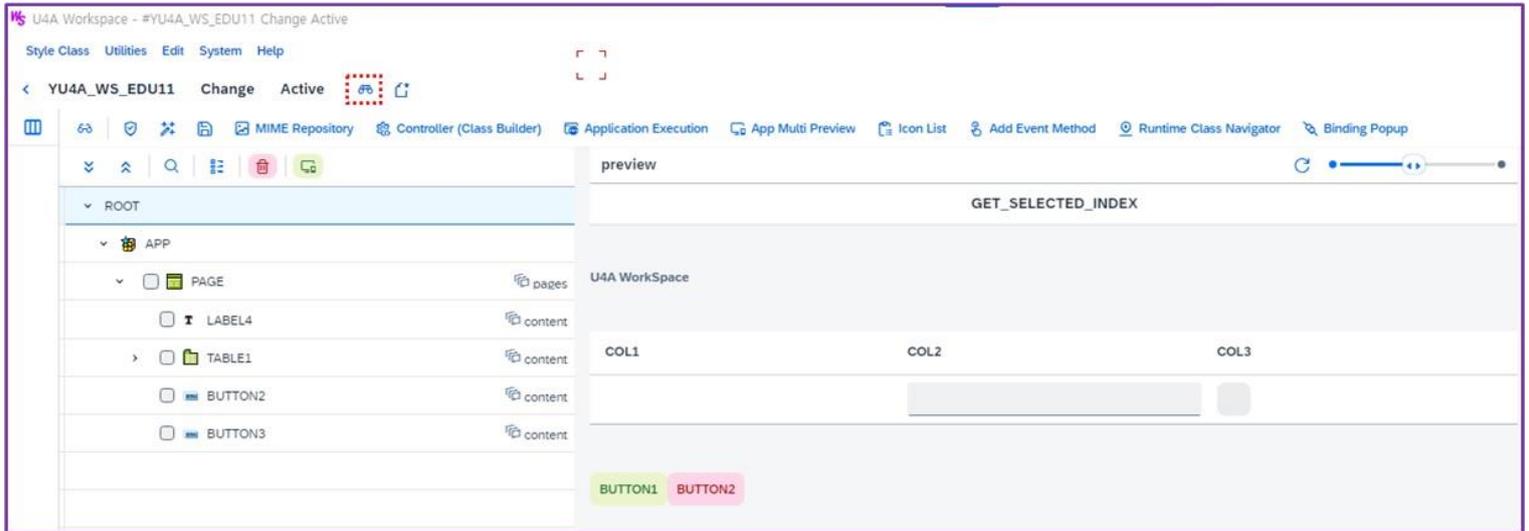
Method EV_SELECT_RADIO active
1  METHOD EV_SELECT_RADIO.
2
3  DATA LV_MSG TYPE STRING.
4  -----*
5
6  "MESSAGE 내역
7  LV_MSG = 'U4A WorkSpace!!!'.
8
9  "MESSAGE 호출
10 CALL METHOD /U4A/CL_UTILITIES=>M_MESSAGE_TOAST
11 EXPORTING
12 IO_VIEW = ME->/U4A/IF_SERVER-AR_VIEW " [U4A] UI Element Super Class
13 * IS_MSG = " Structure of message variables
14 I_MSGTX = LV_MSG " MESSAGE TEXT
15 * IT_MSGTX = " Table of Strings
16 * I_VPOS = CS_M_MSG_TOS_VPOS-CENTER " Vertical position
17 * I_HPOS = CS_M_MSG_TOS_HPOS-CENTER " Horizontal position
18 * I_VOFF = '0' " Vertical offset
19 * I_HOFF = '0' " Horizontal offset
20 * I_DURATION = 10000
21
22
23 ENDMETHOD.
  
```



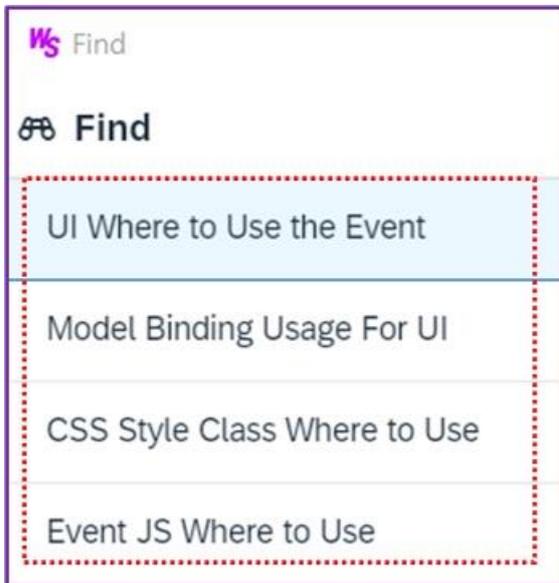
5.11 Find 기능

■ **목적** : Find는 Application 생성 후 Event, Model Binding, CSS style, JS Event등 어디에 사용되었는지 확인 할 수 있다.

1. 상단의 Application 이름과 상태를 나타낸 Line 중에 망원경 모양 ICON 클릭.

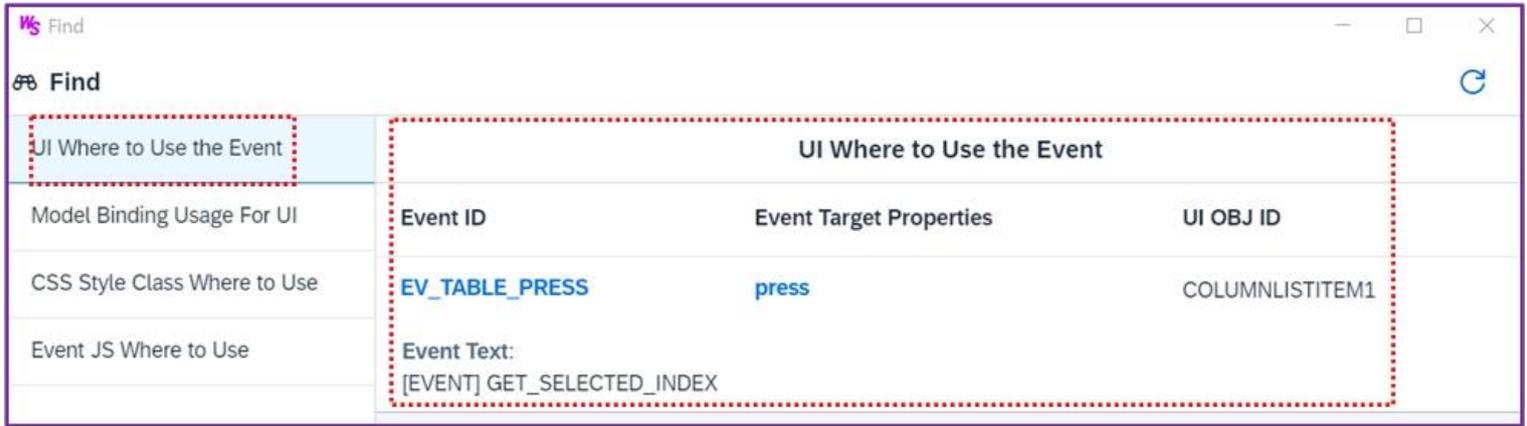


망원경 모양 아이콘을 클릭하면 하단의 이미지 처럼 팝업창이 나타난다.



2-1. UI Where to use The event

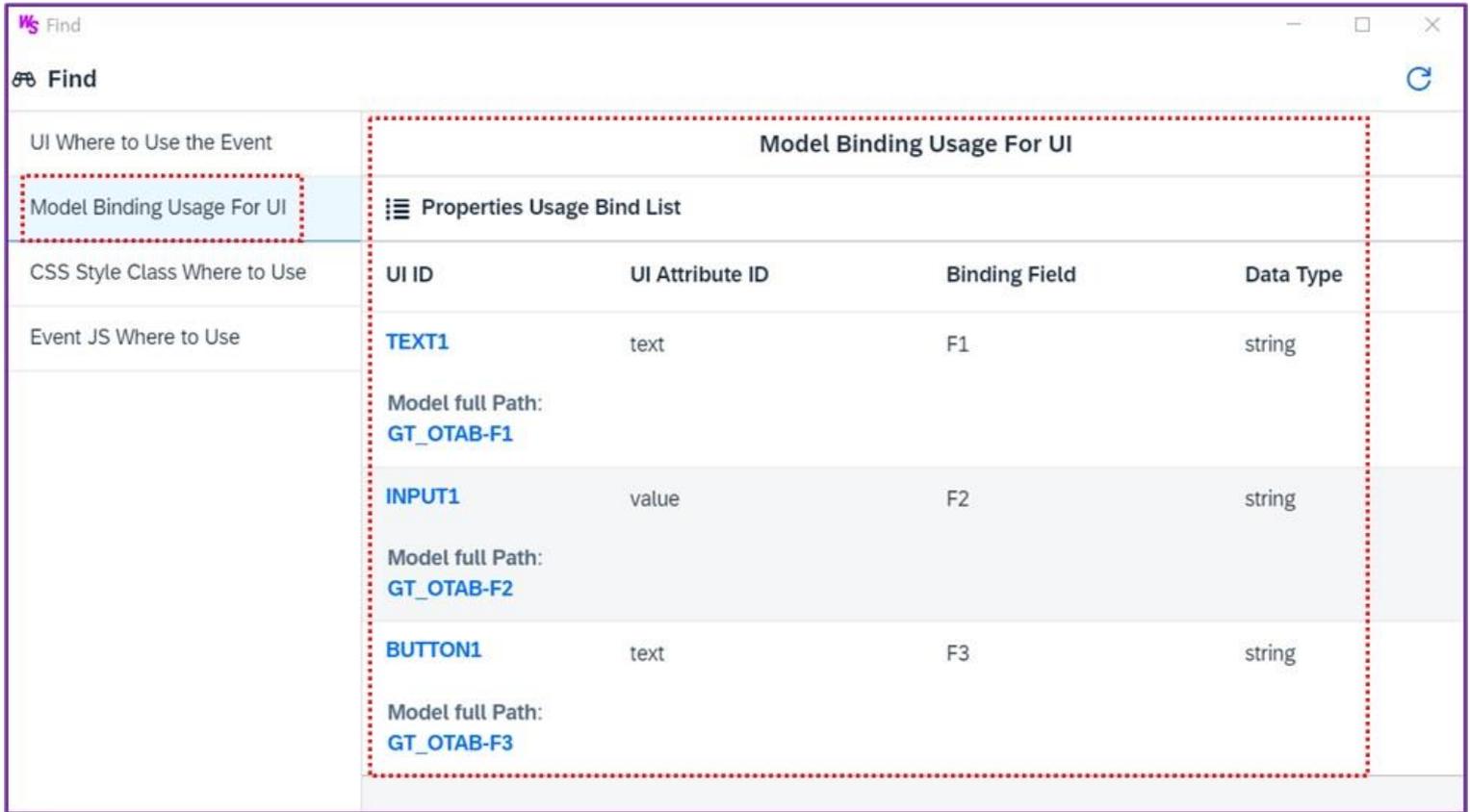
1) Application에서 사용한 Event 내역을 확인한다.



Option Name	Means
Event Id	Event 생성 시 작성한 이름(Click시 해당 Method로 이동)
Event Text	Event 생성 시 작성한 설명
Event Target Properties	적용한 Event 이름(Click시 해당 UI Object로 이동)
UI Object Id	Event가 적용된 UI Object의 id

2-2. Model Binding Usage for UI

1) application에서 Binding된 UI Object를 확인한다.



2-1) Properties Usage bind List

Option Name	Means
UI ID	Binding한 UI Object ID(Click시 해당 UI Object로 이동)
UI Attribute Id	Properties에서 어디에 Binding이 걸려있는지 위치 확인
Model full path	Binding이 걸려있는 field의 정보(Click시 Source Code-Based로 이동)
Binding field	Binding이 걸려있는 Attribute의 field
Data type	Binding이 걸려있는 field의 type

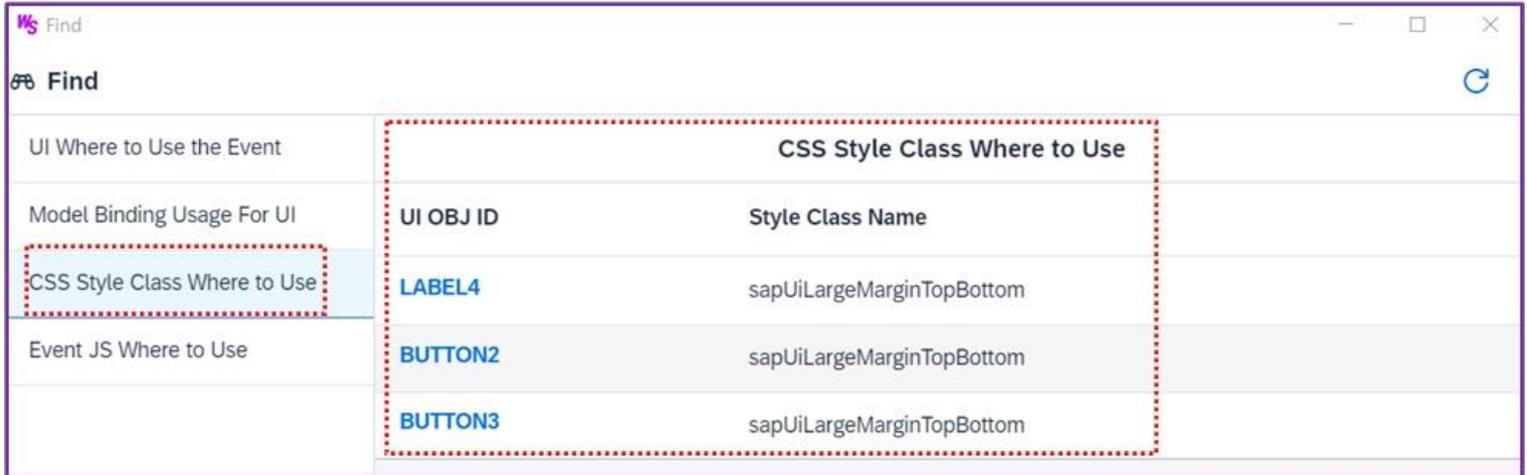
2-2) Aggregations Usage bind Models

Option Name	Means
UI ID	Binding한 UI Object ID(Click시 해당 UI Object로 이동)

Aggregations ID	UI Object 에서 Aggregations에서 어디에 Binding이 걸려있는지 위치 확인
Binding Model	Binding이 걸려있는 attribute

2-3. CSS Style Class Name

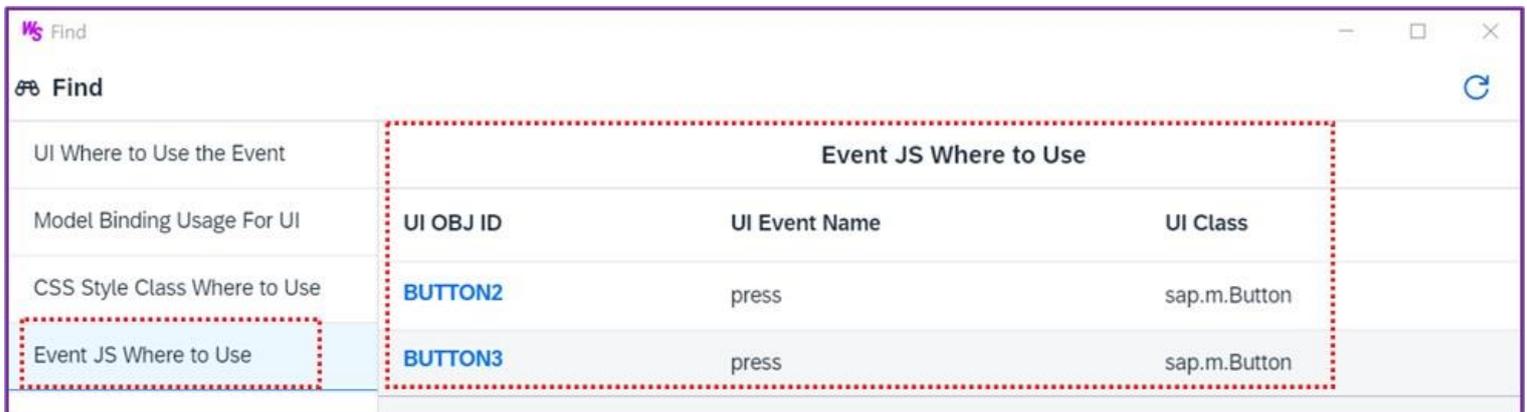
1) application에서 사용한 CSS 내역을 확인한다.



Option Name	Means
UI Obj id	Style Class가 적용된 UI Obj id(Click시 해당 UI Obj로 이동)
Style Class Name	사용한 Style Class의 이름

2-4. Event JS Where to use

1) application에서 사용한 JS(JavaScript) 내역을 확인한다.



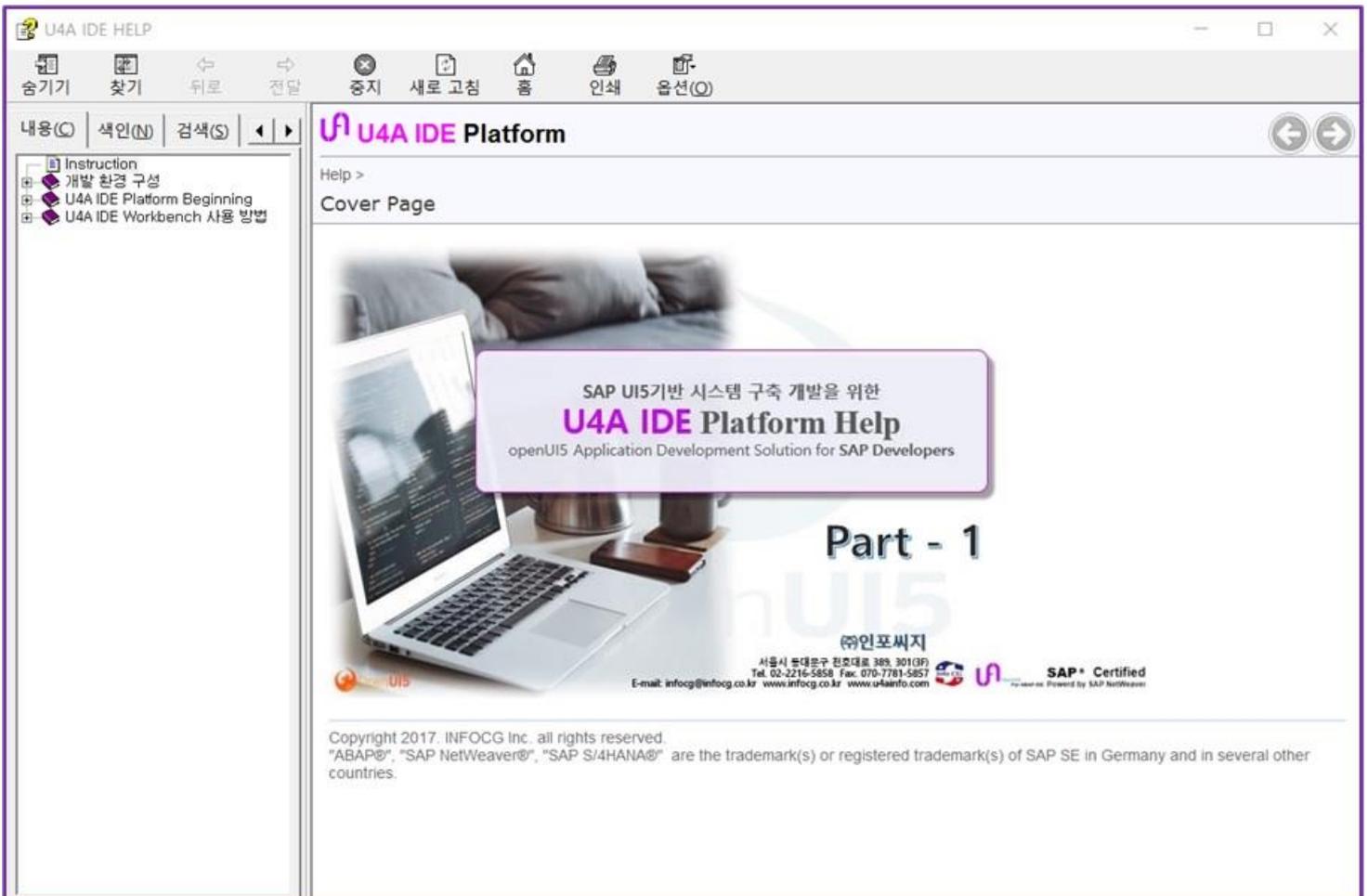
U4A IDE HELP

Option Name	Means
UI Obj id	JS(JavaScript)가 적용된 UI Obj id(Click시 해당 UI Obj로 이동)
UI Event Name	JS(JavaScript)가 적용된 Event의 항목
UI class	사용된 UI의 class 명

5.12 Help Document 기능



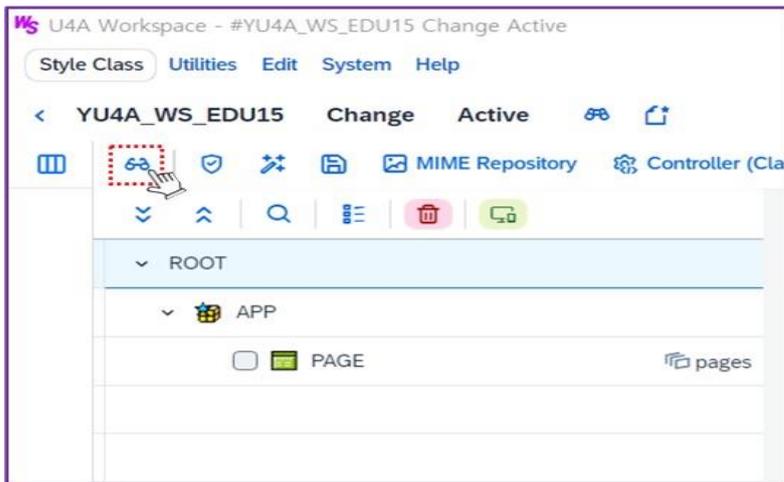
1) Help Document로 application을 만들때 필요한 정보를 확인 할 수 있다.



5.13 Display-Change 기능

■ 목적 : Application을 수정 또는 확인만 가능한 "Mode" 상태를 변경 할 수 있다.

1. Application에 들어가면 상단에 안경과 펜모양을 한 Icon이 Display<->Change이다.

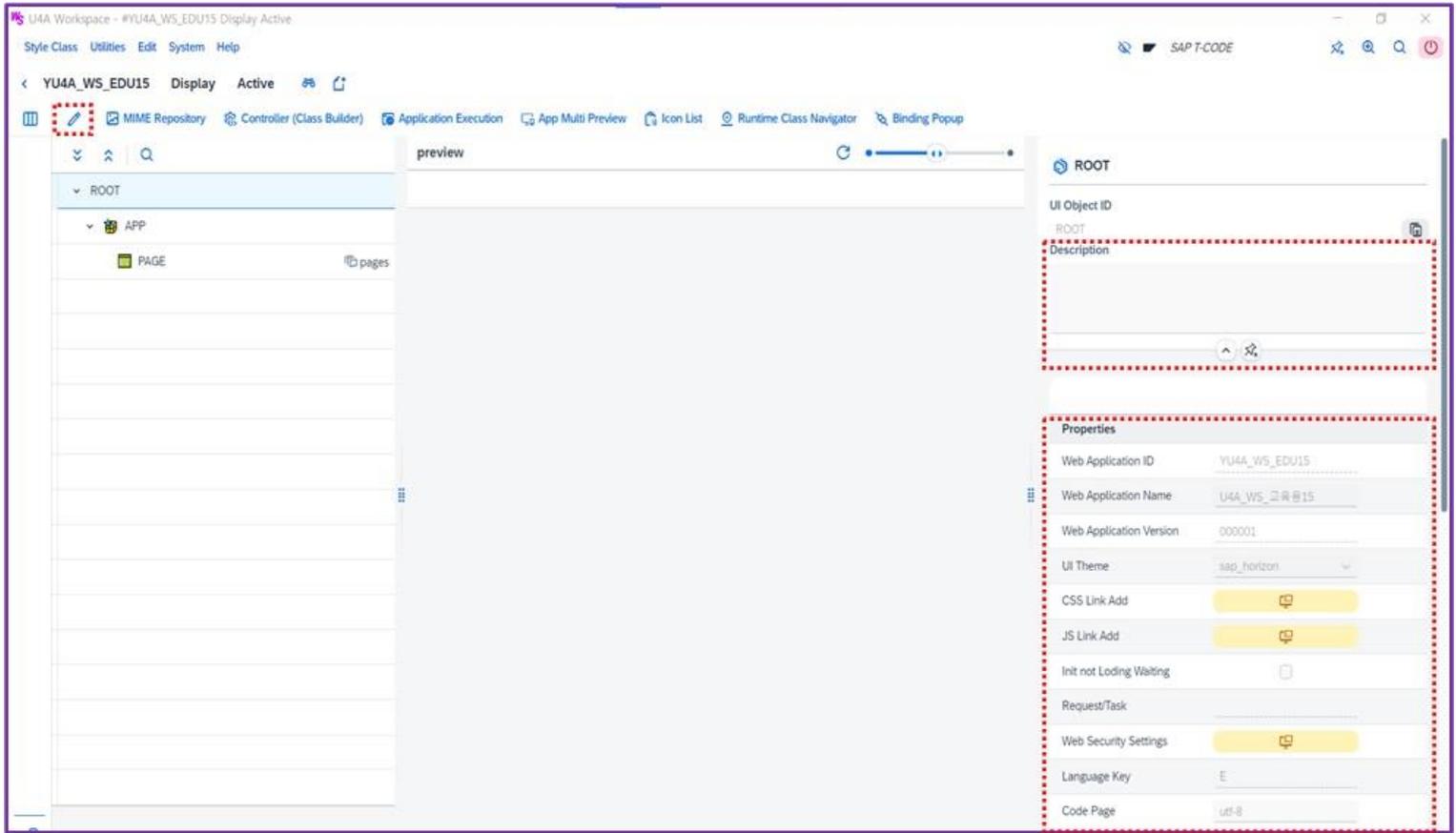


1) Display mode에서는 Change로, Change mode에서는 Display로 변환할 수 있다 ("단축키 : Ctrl + F1")

2-1. Display mode

1) Display mode에서는 변경은 불가능 하나, application 작동시 Debug확인은 가능하다.

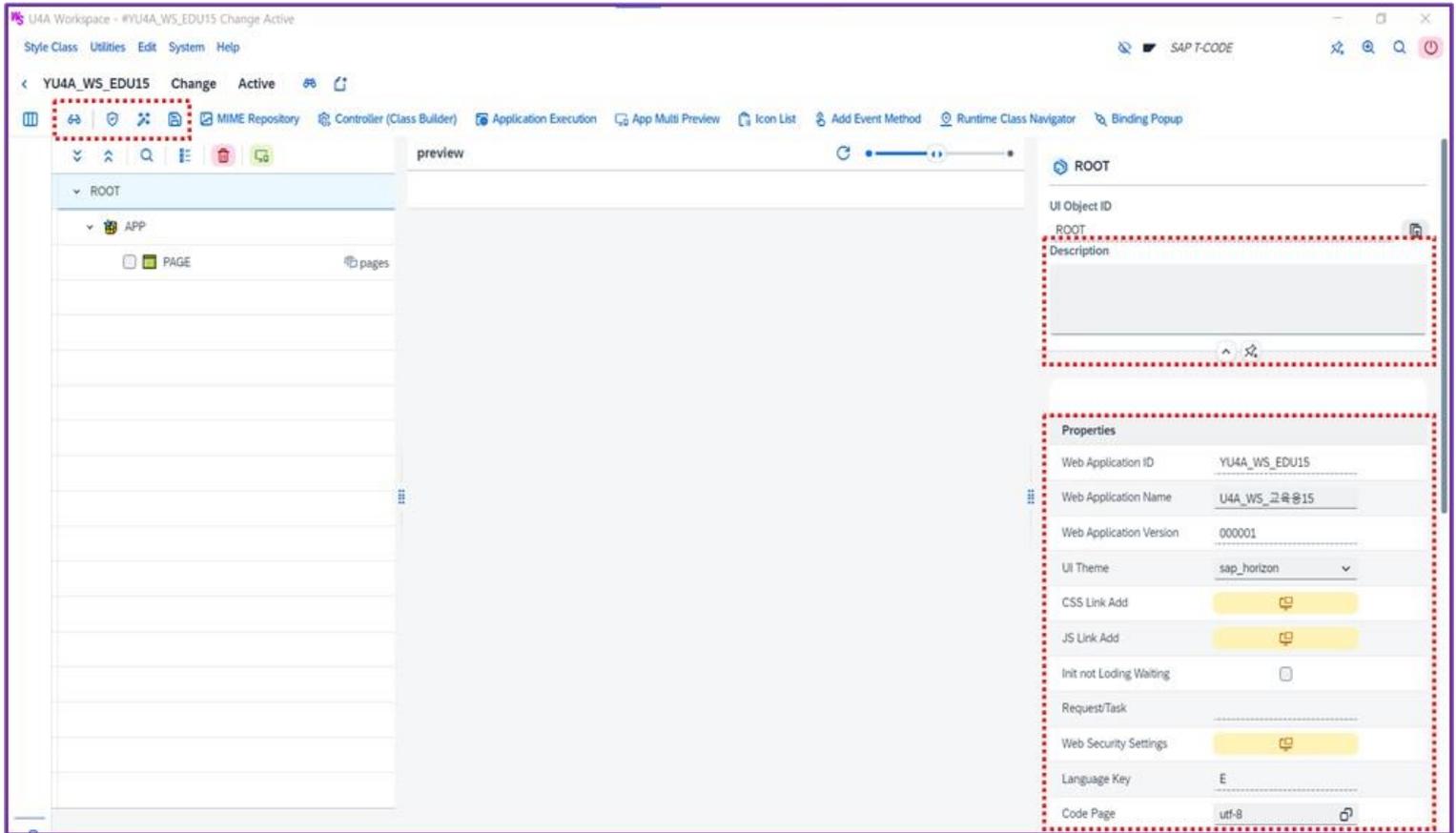
(Display mode에서 오른쪽 check와 Active Icon이 나타나지 않는다)



2-2. Change mode

1) Chagne모드에서는 UI Object 생성, Application수정 모두 가능하다.

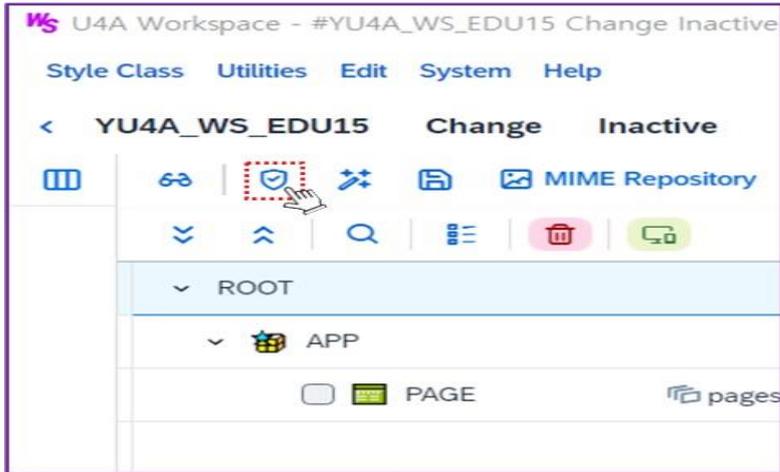
U4A IDE HELP



5.14 Syntax Check 기능

■ 목적 : 만든 Application의 Logic 구조가 정상적으로 구성되어 있는지 Syntax Check를 진행한다.

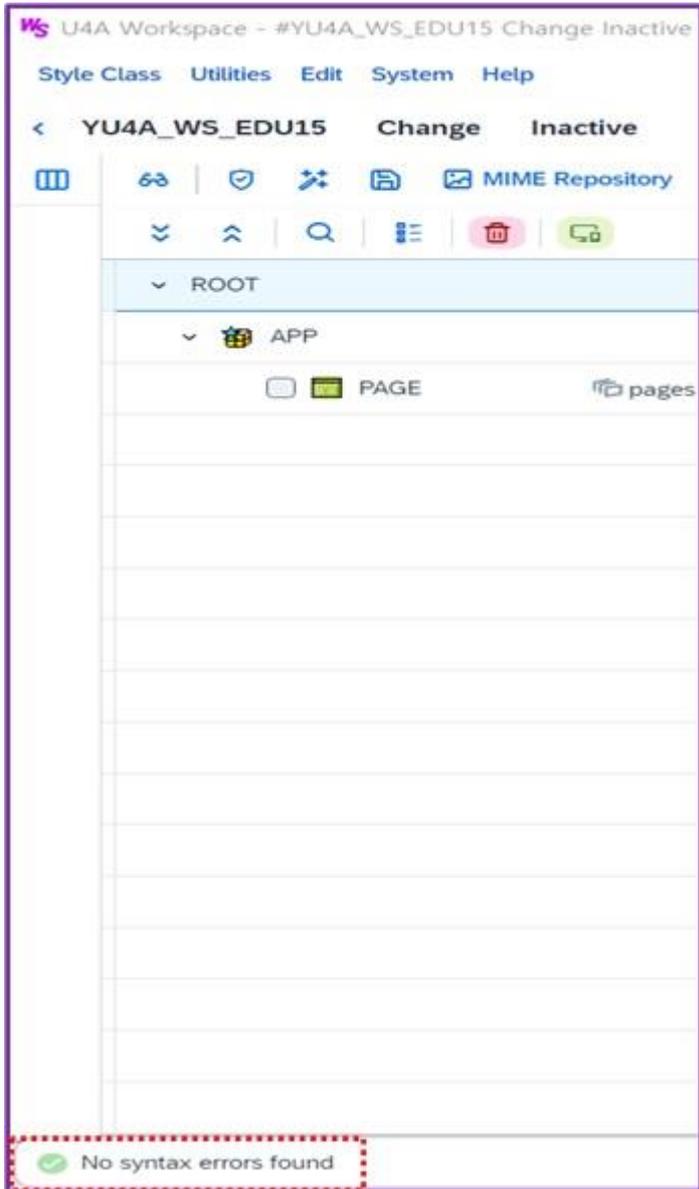
1. Application에 들어가면 하단 그림의 붉은색 Box 안에 있는 것이 Syntax Check이다.
(단, Change모드 일 시 확인 가능)



2. Syntax Check를 통해 Application이 정상적으로 작동이 가능한지 확인 할 수 있다.

1-1) 정상 작동일 시

하단에 "No syntax errors found" 라는 문구가 출력된다



1-2) 오류 발생 시

구문 점검 오류 "팝업"이 나타나면서 오류의 원인을 확인 할 수 있다.(Description 내역 Click 시 오류 발생 지점으로 이동)

U4A IDE HELP

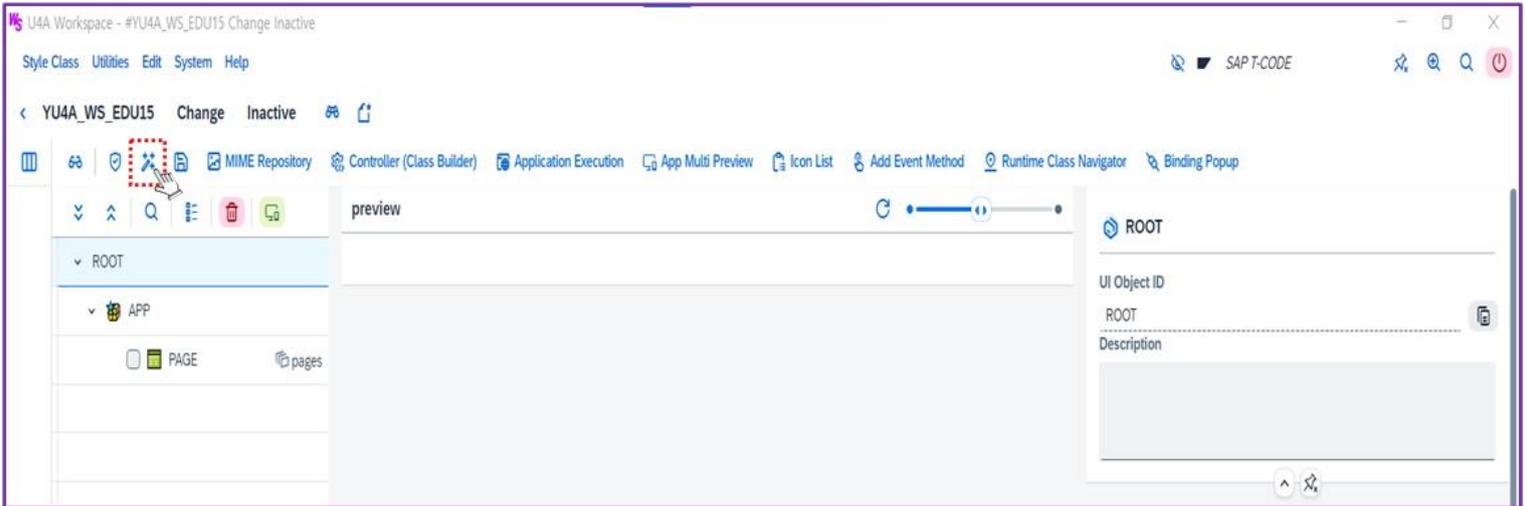
The screenshot shows the SAP U4A IDE interface. At the top, the workspace is titled "U4A Workspace - #YU4A_WS_EDU15 Change Inactive". The menu bar includes "Style Class", "Utilities", "Edit", "System", and "Help". The breadcrumb navigation shows the current path: "YU4A_WS_EDU15 > Change > Inactive". Below the breadcrumb is a toolbar with icons for "MIME Repository", "Controller (Class Builder)", "Application Execution", and "App Multi". A "preview" window is visible on the right. The bottom of the IDE features an "Error Footer Message" dialog box, which is highlighted with a red dashed border. This dialog contains a table with error details.

Error Type	Line	Description
E	0	Controller Method name : "/U4A/IF_SERVER~HANDL_ON_INIT" inactive status.

5.15 Activate 기능

■ 목적 : Syntax Check와 함께 정상적으로 기능 구현을 확인 하면 Inactive 상태를 Active로 변경한다.

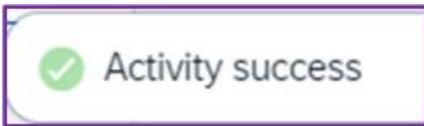
1. Application에 들어가면 하단 그림의 붉은색 Box 안에 있는 것이 Activate이다.(단, Change모드 일시 확인 가능)



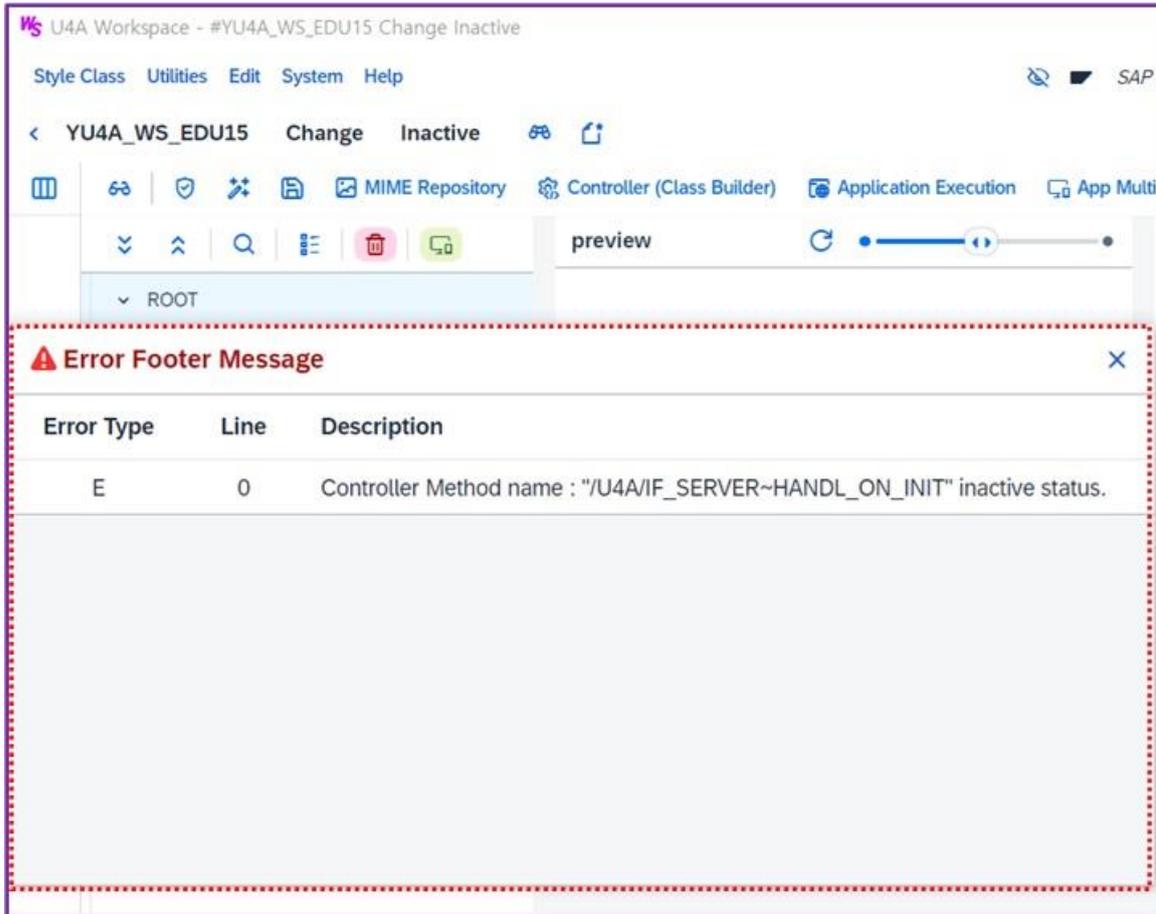
2. Activation을 통해 application의 오류점검과 정상 작동 확인 시 Application Execution을 작동 시킬 수 있다.

1-1) 정상 작동일 시

하단에 Activity success라는 문구가 출력되고, Application Execution 작동 시 Web 화면이 나타난다.



2-2) 오류 발생 시



구문 점검 오류 "팝업"이 나타나면서 오류의 원인을 확인 할 수 있고(Description 내역 Click 시 오류 발생 지점으로 이동), Application Execution도 작동되지 않는다.

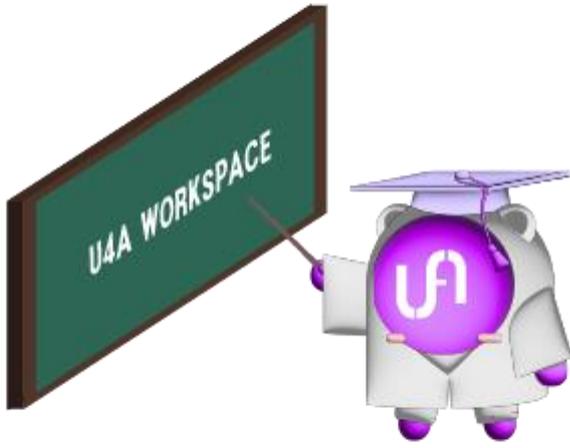


해당 상태로 Application Execution을 작동 시키면 가장 하단에 아래와 같은 메시지가 나타난다.



Controller(Class builder)에서도 Activation 할 수 있는데 여기서 진행하는 것은 ABAP의 Logic의 이상을 확인 하는 것이고, application에서는 ABAP Logic과 application의 이상을 확인하기 때문에 ABAP Logic을 구성하는 경우 Controller의 모든 기능이 먼저 Activation해야한다.

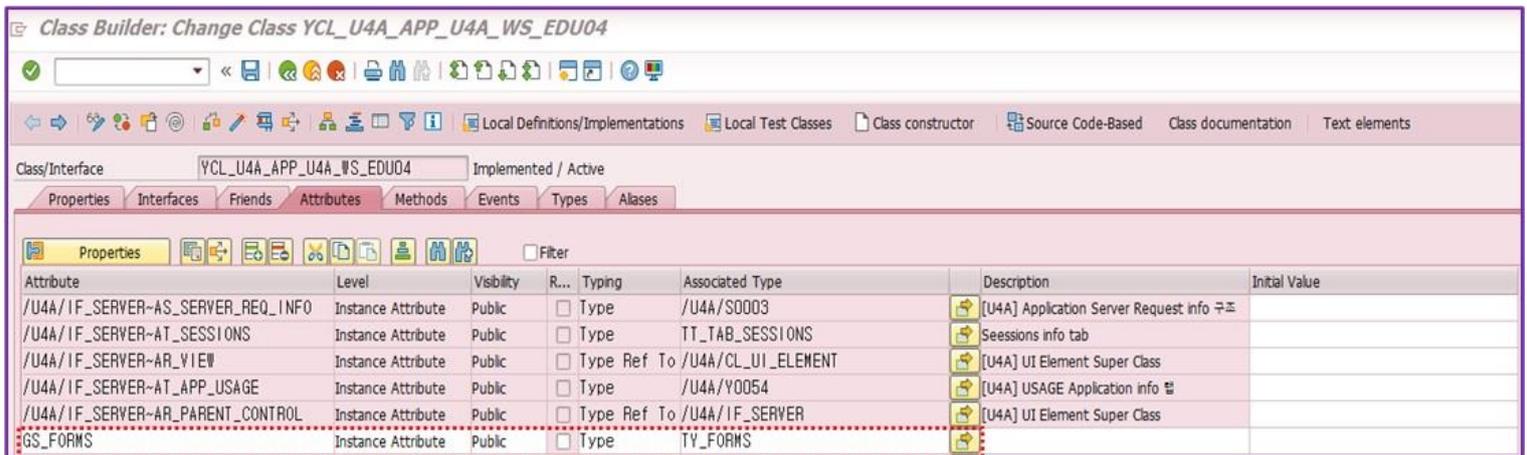
5.16 UI Template Wizard



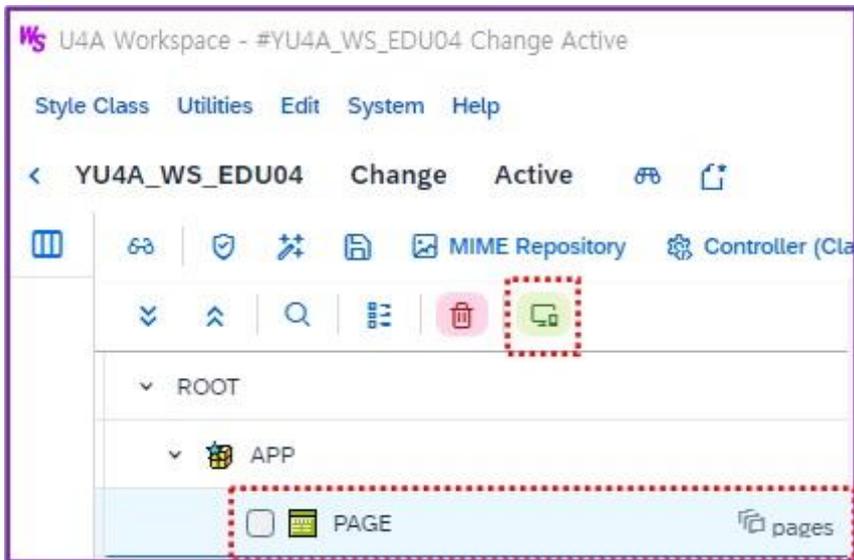
5.16.1 Forms UI Create

▣ 목적 : 자주 사용하는 Form UI를 Sample로 만들어 쉽게 생성하기 위한 목적.

1. Form의 Field 값을 Binding 시킬 Public Structure를 생성.

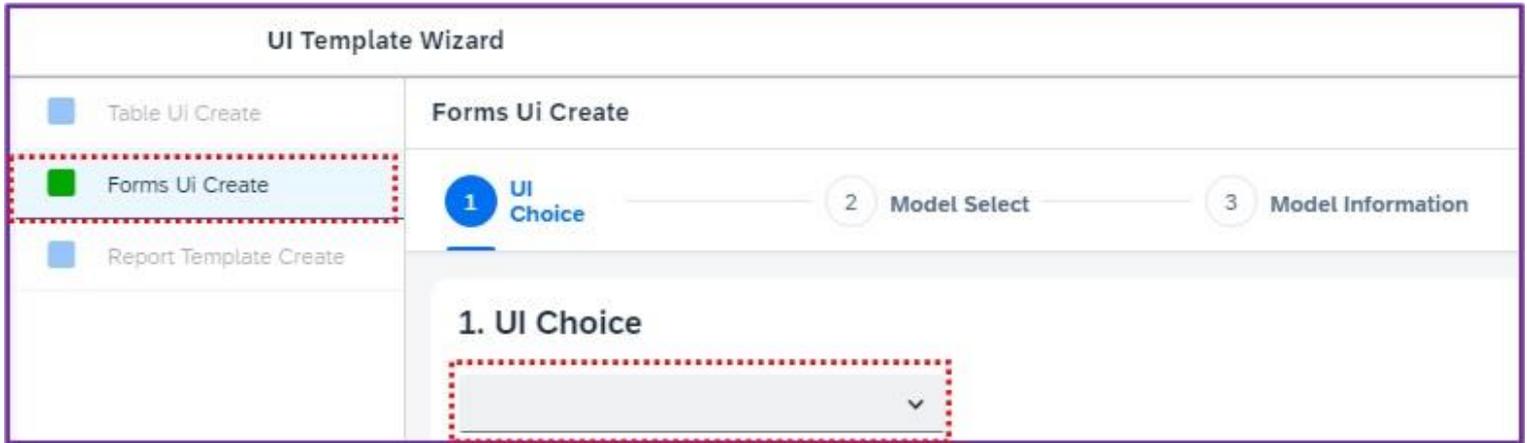


2. Form UI를 삽입 할 위치를 선택한 후 UI Template Wizard Button Click.

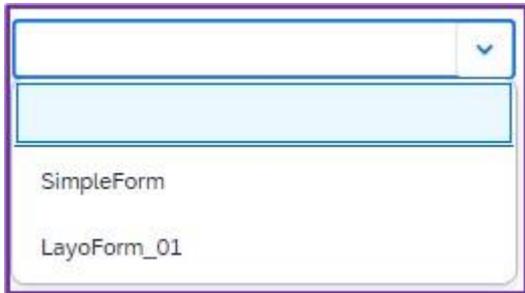


3. UI Template Wizard 화면에서 Form UI Create Tab을 선택. (녹색 Box는 Form Data를 Binding 시킬 Model이 존재함을 의미.)

※ 첫번째 단계인 UI Choice 박스 선택.

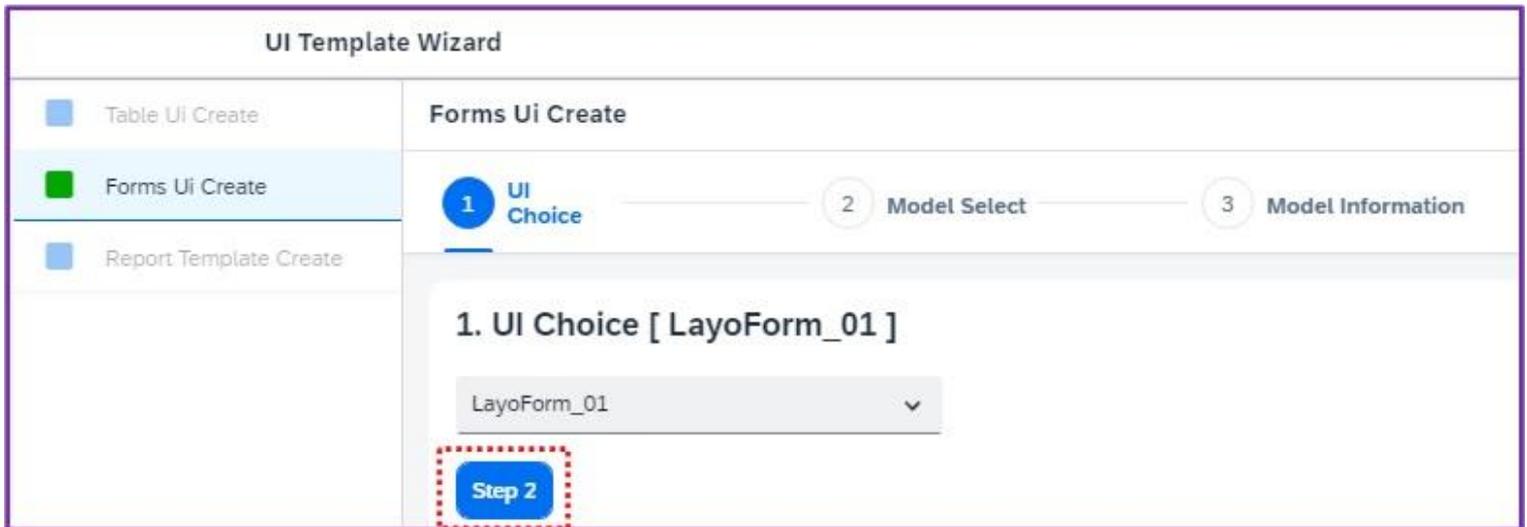


4. 생성하려는 Form UI 선택 (1. LayoForm 2. SimpleForm)

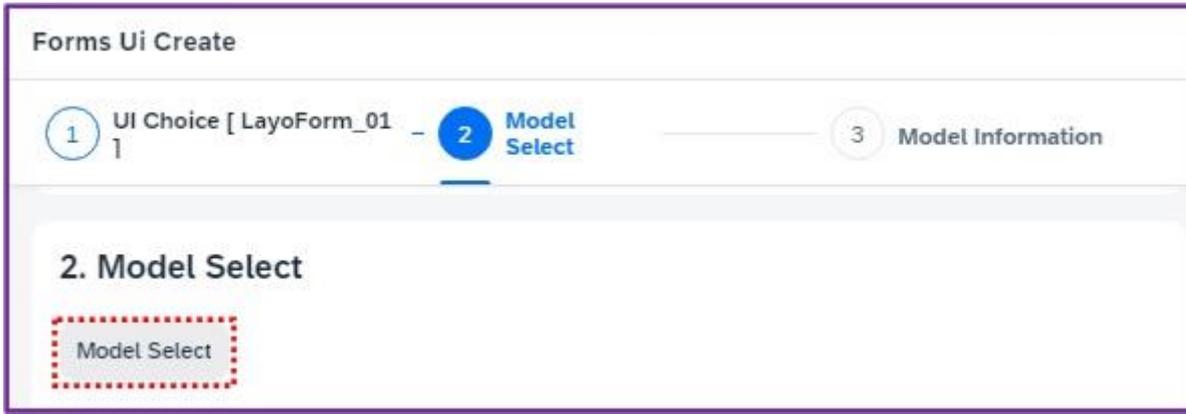


② Model Name : From UI에 Binding 시키는 Model 명 (Select Model Button Click)

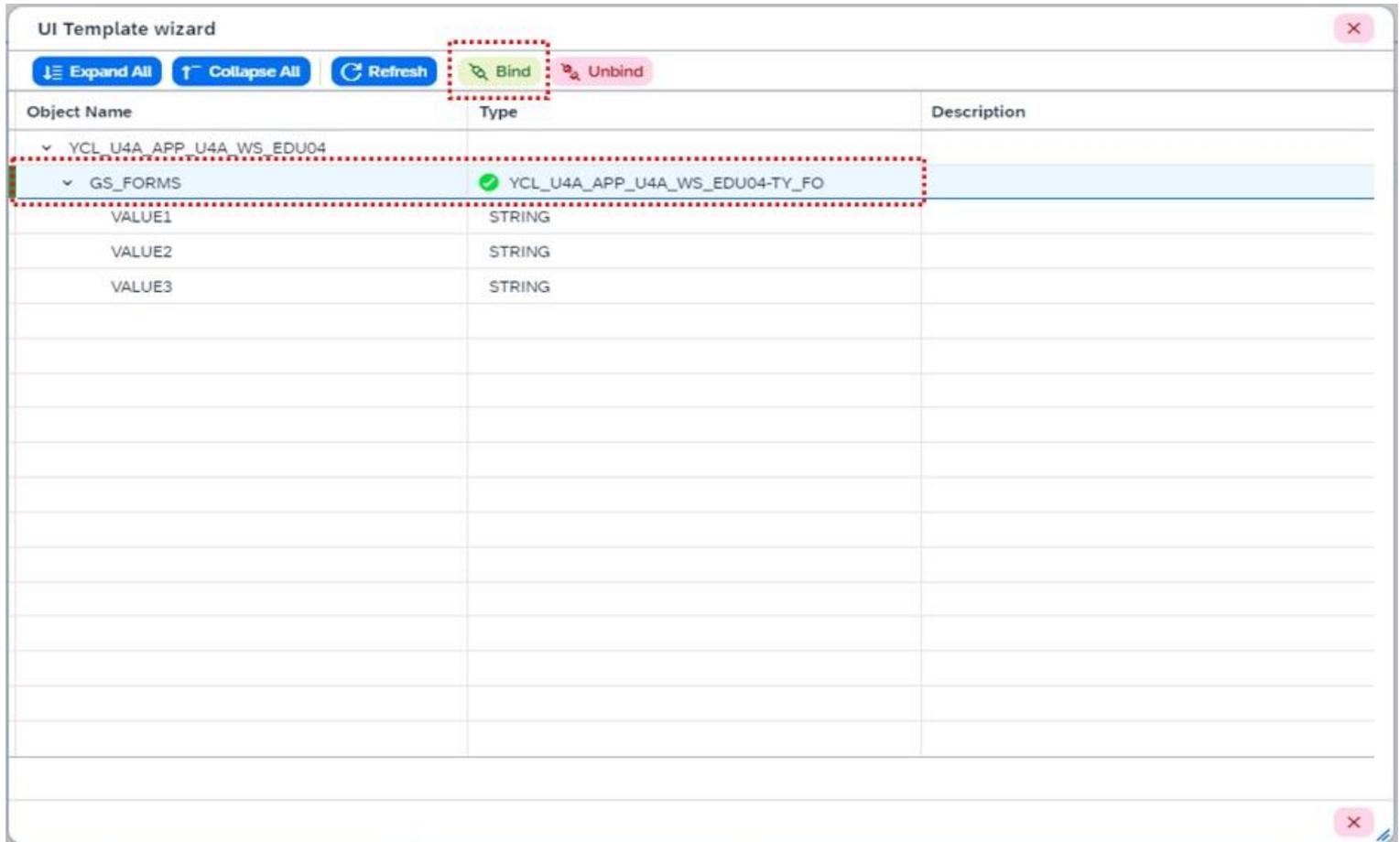
5. Form UI 선택 후 다음 Step으로 진행.



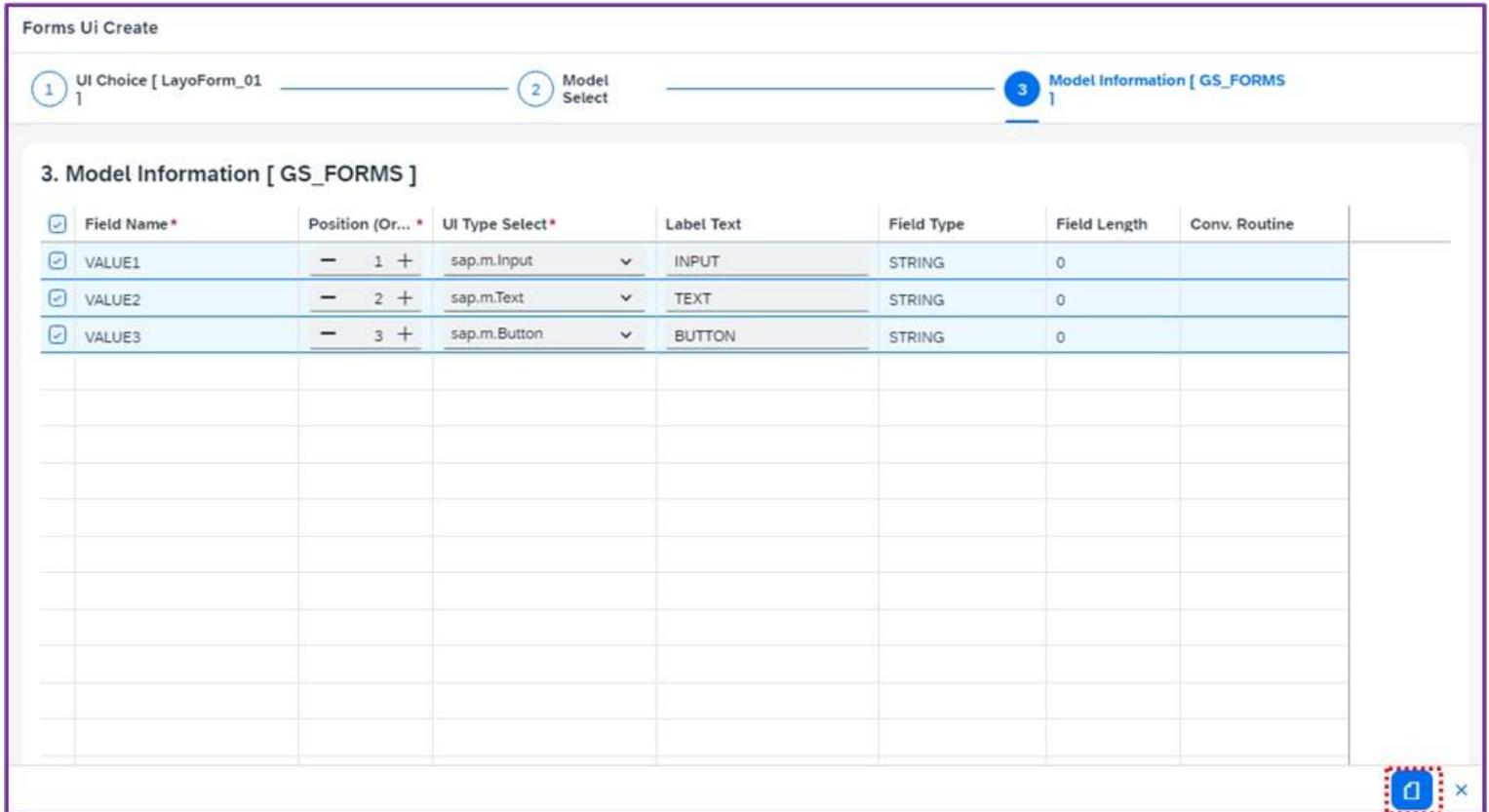
6. Form UI에 Binding 시키는 Model 명 (Model Select Button Click)



7. Binding할 Model Click 후 "Bind" Button Click.



8. Binding 되어있는 Structure의 Field 정보 및 Option 설정.



① √ : 선택하면 UI에 포함 되고 선택하지 않으면 UI에 포함되지 않음.

② Position : Form 안에서의 UI의 위치 설정.

② Field Name : Binding 되어있는 Structure Field 명.

③ UI type select : Structure Field UI Type.

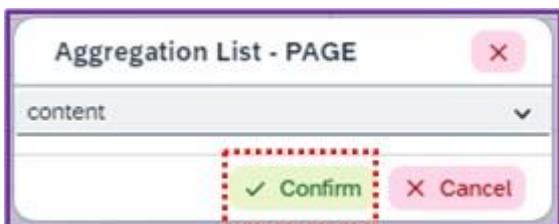
④ Label Text : Field Description.

⑤ Field type : Field Value Type.

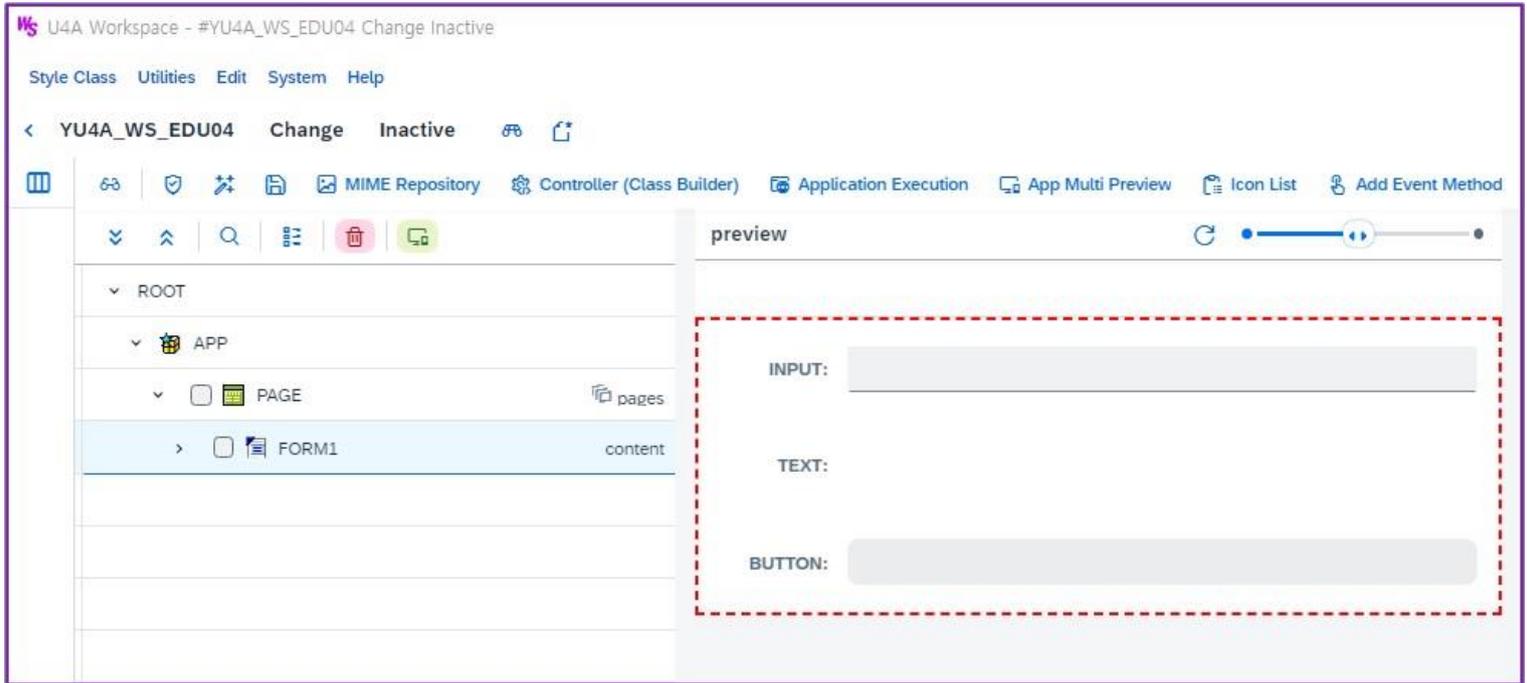
⑥ field Length : Field Value 길이.

⑦ Conversion Routine : Field에 적용 된 Conversion Routine.

9. UI가 어떤 Aggregation으로 생성 될 지 선택(선택지가 1개면 Popup은 생성 되지 않음)



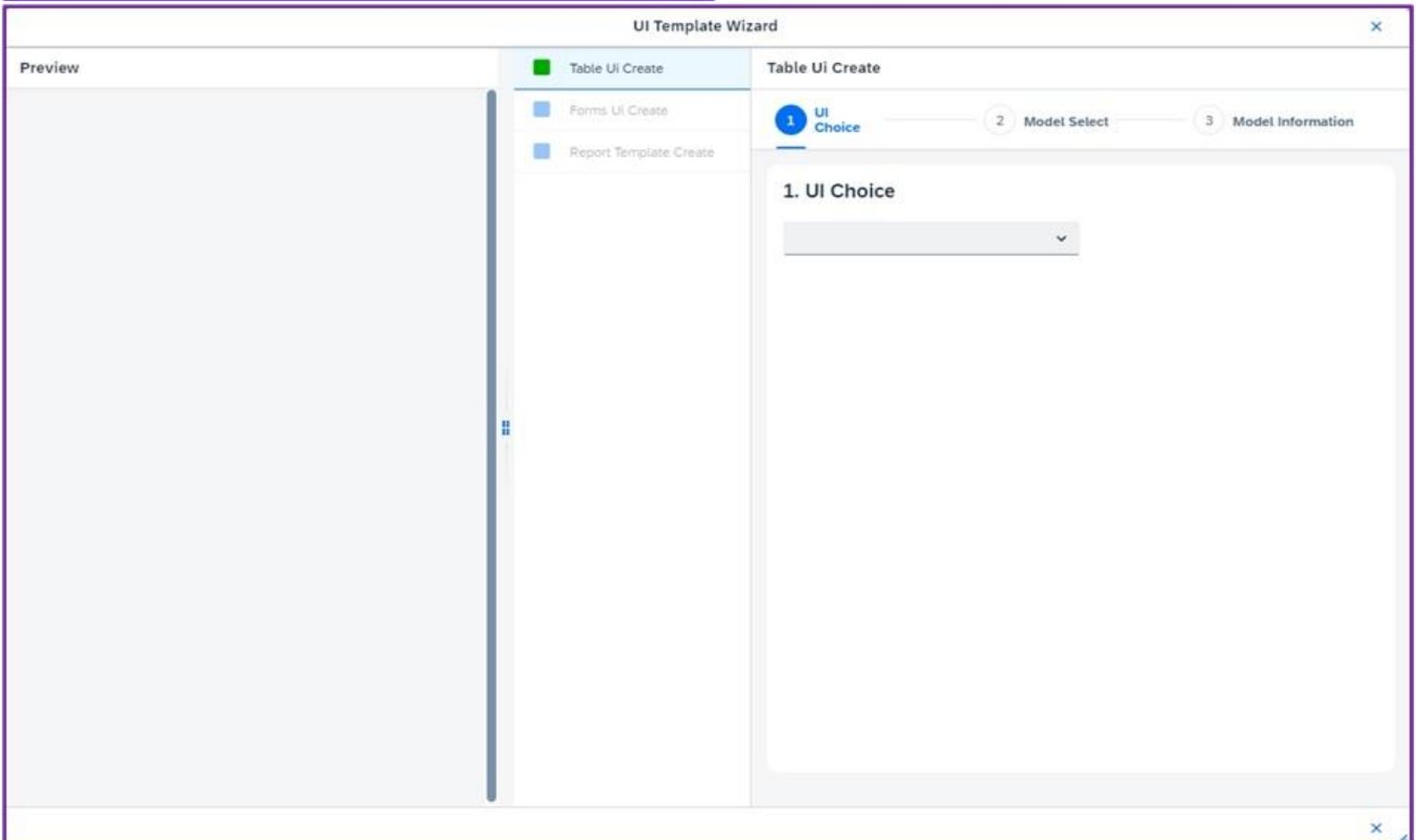
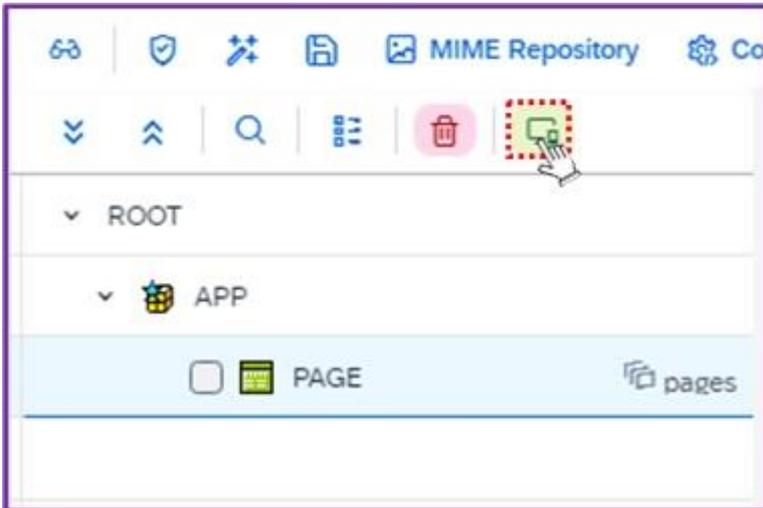
10. 해당 Application에 Form UI가 생성된 것을 확인.



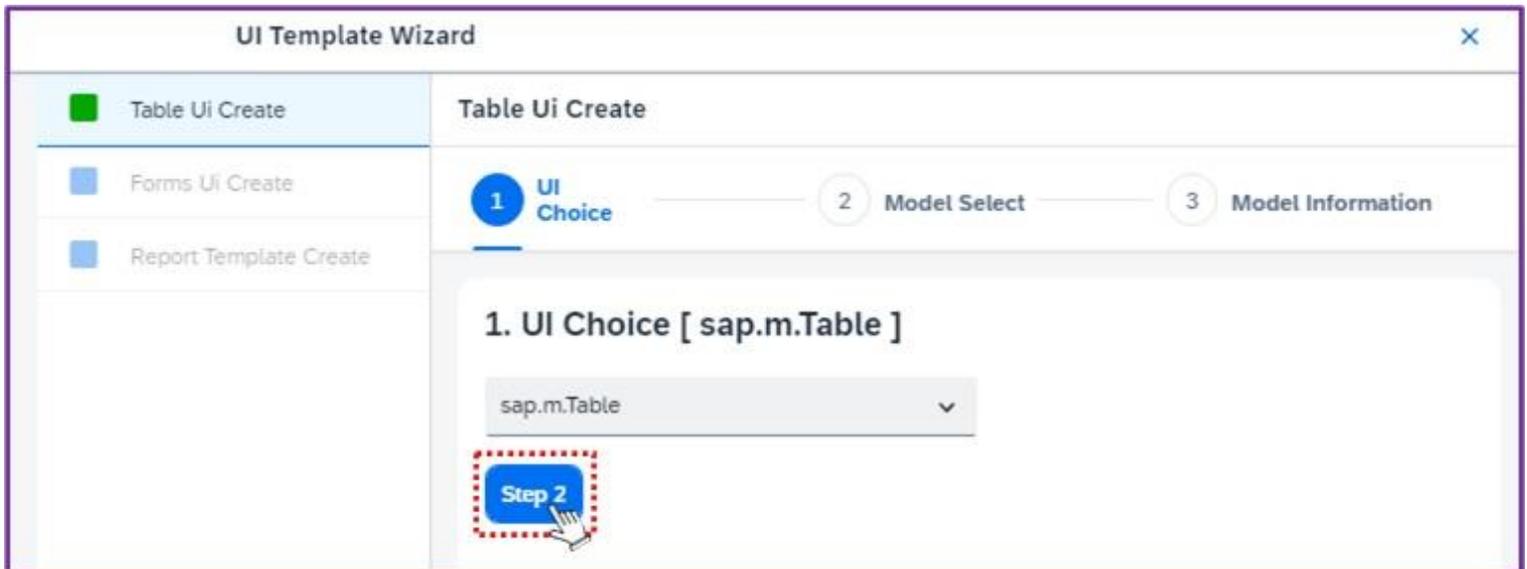
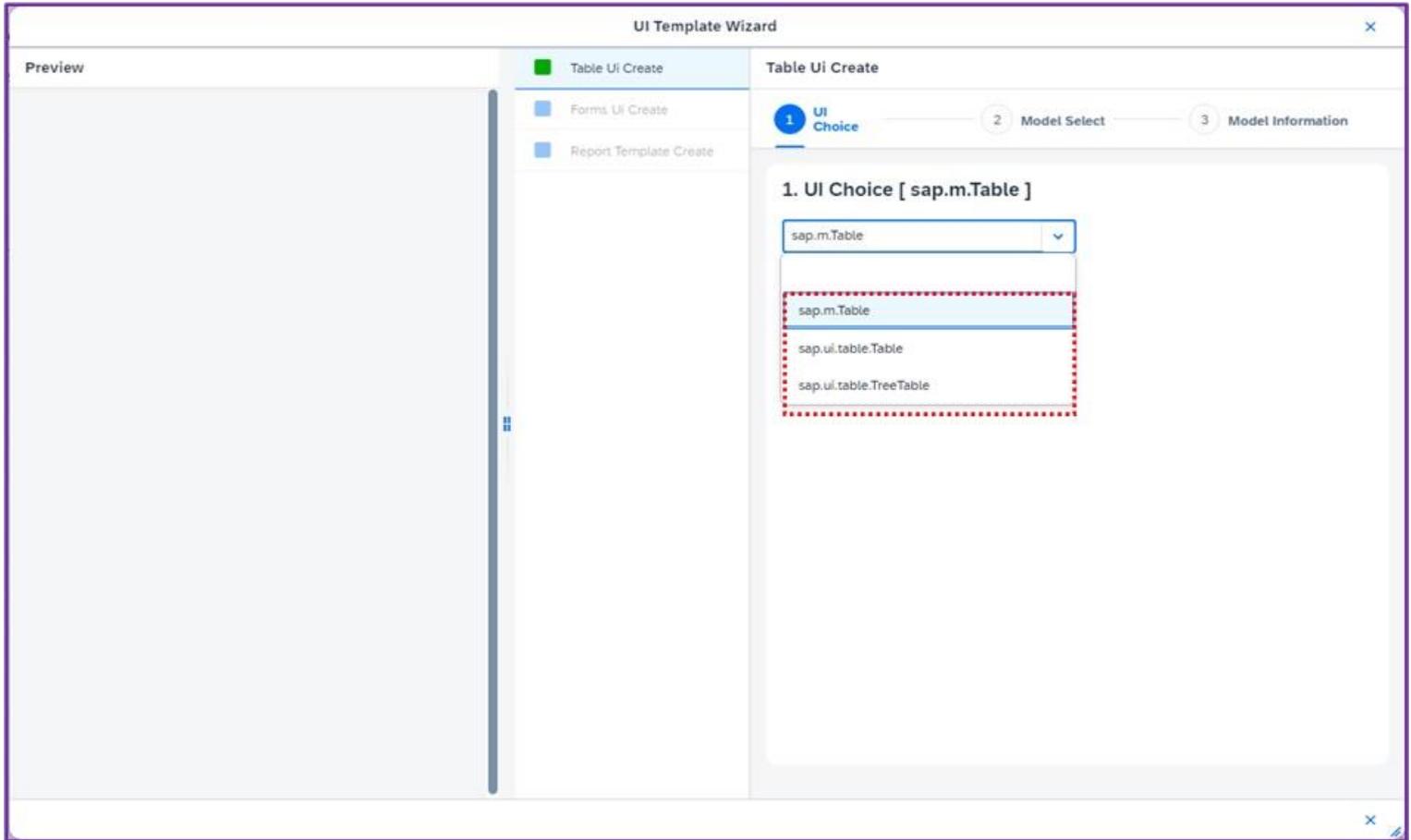
5.16.2 Table UI Create

■ 목적 : TABLE 형태의 UI Object를 편하게 생성 할 수 있는 기능이다.

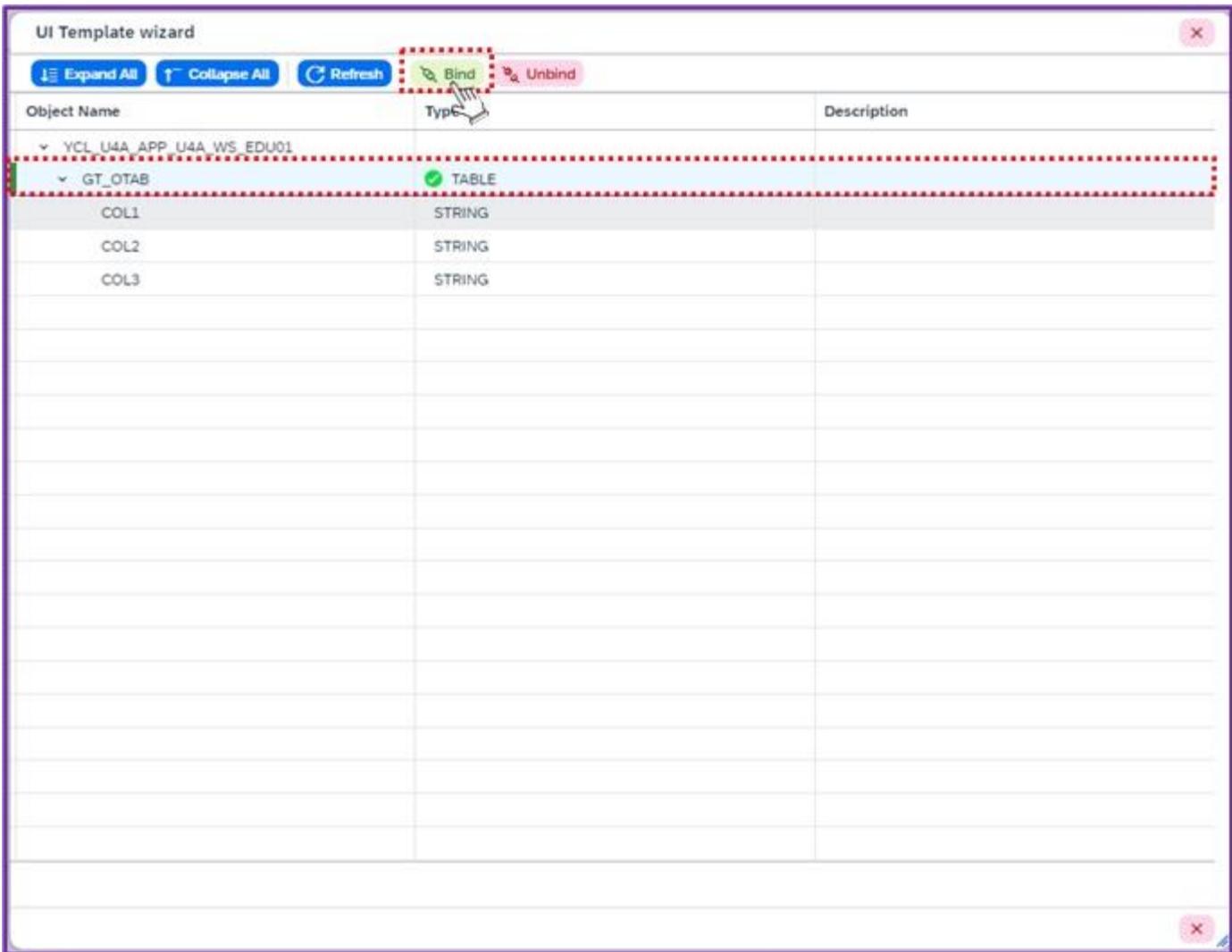
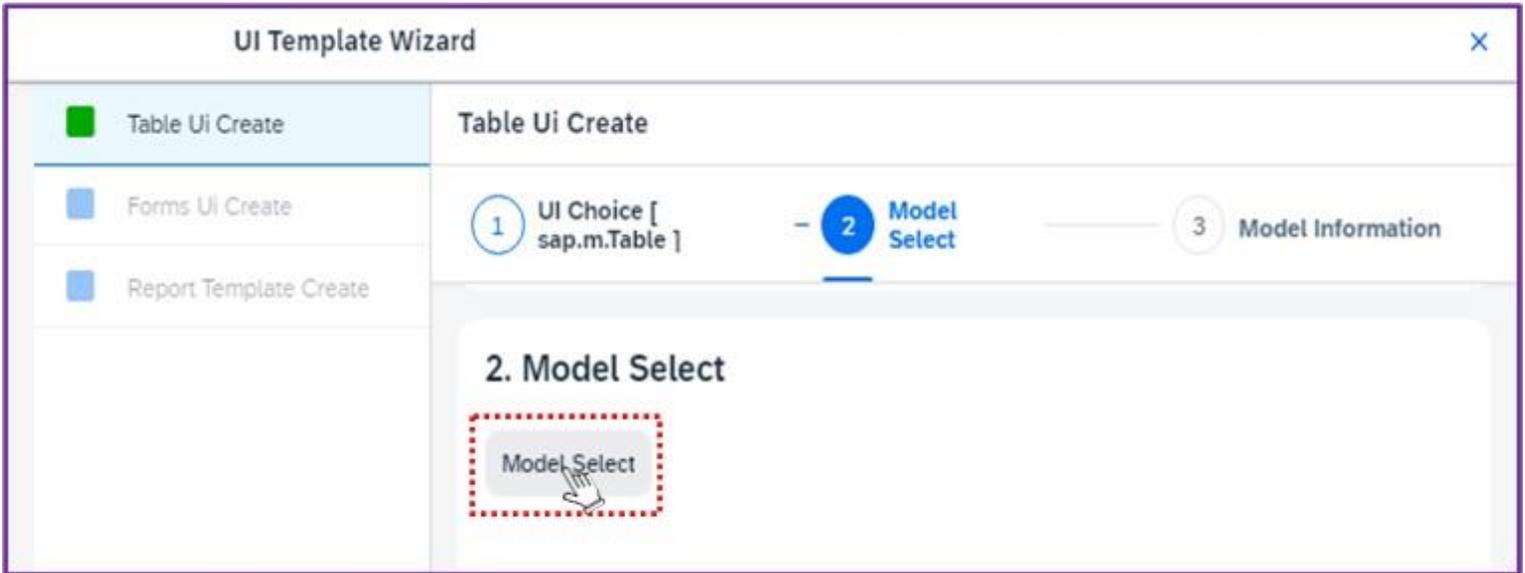
1. U4A는 application내에서 Table의 기본 형태에 대한 틀을 제공한다.



2. 원하는 Table의 UI를 선택하고 "Step 2" 버튼 클릭(m.table, ui.table, Treetable 제공)



3. "Model Select" 버튼을 클릭하여 Table의 Model을 binding 한다.



U4A IDE HELP

The screenshot displays the U4A IDE interface. On the left, a project tree shows a hierarchy: ROOT > APP > PAGE > TABLE1 > COLUMNLISTITEM1 > TEXT1. The 'TEXT1' object is selected and highlighted with a red dashed border. The central preview window shows a blank area. On the right, the 'Properties' panel for 'TEXT1' is visible, with the 'text' property highlighted by a red dashed border. The 'text' property is set to 'GT_OTAB-COL1'. Other visible properties include 'textDirection' (Inherit), 'wrapping' (true), 'textAlign' (Begin), 'width', 'maxLines', and 'busy' (false).

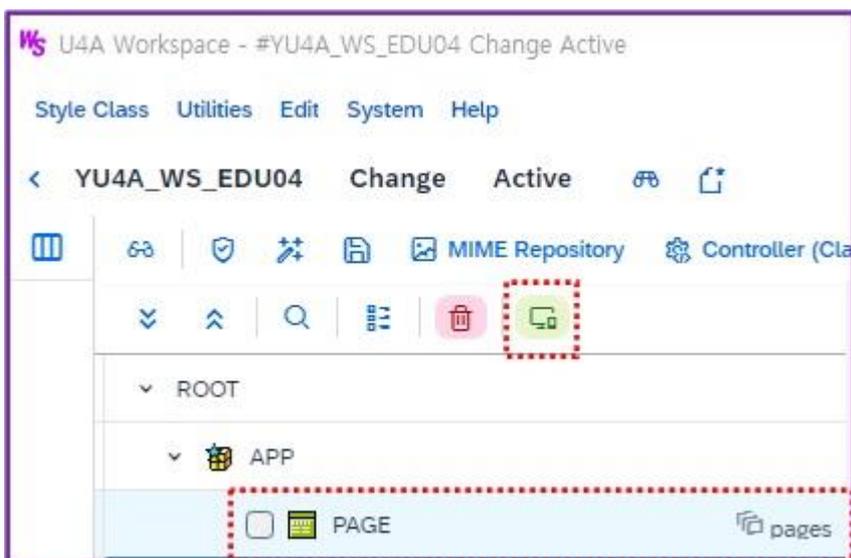
5.16.3 Report Template Create

▣ 목적 : 자주 사용하는 Form UI, TABLE UI 를 Sample로 만들어 쉽게 생성하기 위한 목적.

1. "Form"과 "Table"의 Field 값을 Binding 시킬 Public Structure 및 Table을 생성.

Attribute	Level	Visibility	R...	Typing	Associated Type	Description
/U4A/IF_SERVER~AS_SERVER_REQ_INFO	Instance Attribute	Public	<input type="checkbox"/>	Type	/U4A/S0003	[U4A] Application Server Request info 구조
/U4A/IF_SERVER~AT_SESSIONS	Instance Attribute	Public	<input type="checkbox"/>	Type	TT_TAB_SESSIONS	Seessions info tab
/U4A/IF_SERVER~AR_VIEW	Instance Attribute	Public	<input type="checkbox"/>	Type Ref To	/U4A/CL_UI_ELEMENT	[U4A] UI Element Super Class
/U4A/IF_SERVER~AT_APP_USAGE	Instance Attribute	Public	<input type="checkbox"/>	Type	/U4A/Y0054	[U4A] USAGE Application info 탭
/U4A/IF_SERVER~AR_PARENT_CONTROL	Instance Attribute	Public	<input type="checkbox"/>	Type Ref To	/U4A/IF_SERVER	[U4A] UI Element Super Class
GS_STRU	Instance Attribute	Public	<input type="checkbox"/>			
GT_OTAB	Instance Attribute	Public	<input type="checkbox"/>			
			<input type="checkbox"/>	Type		

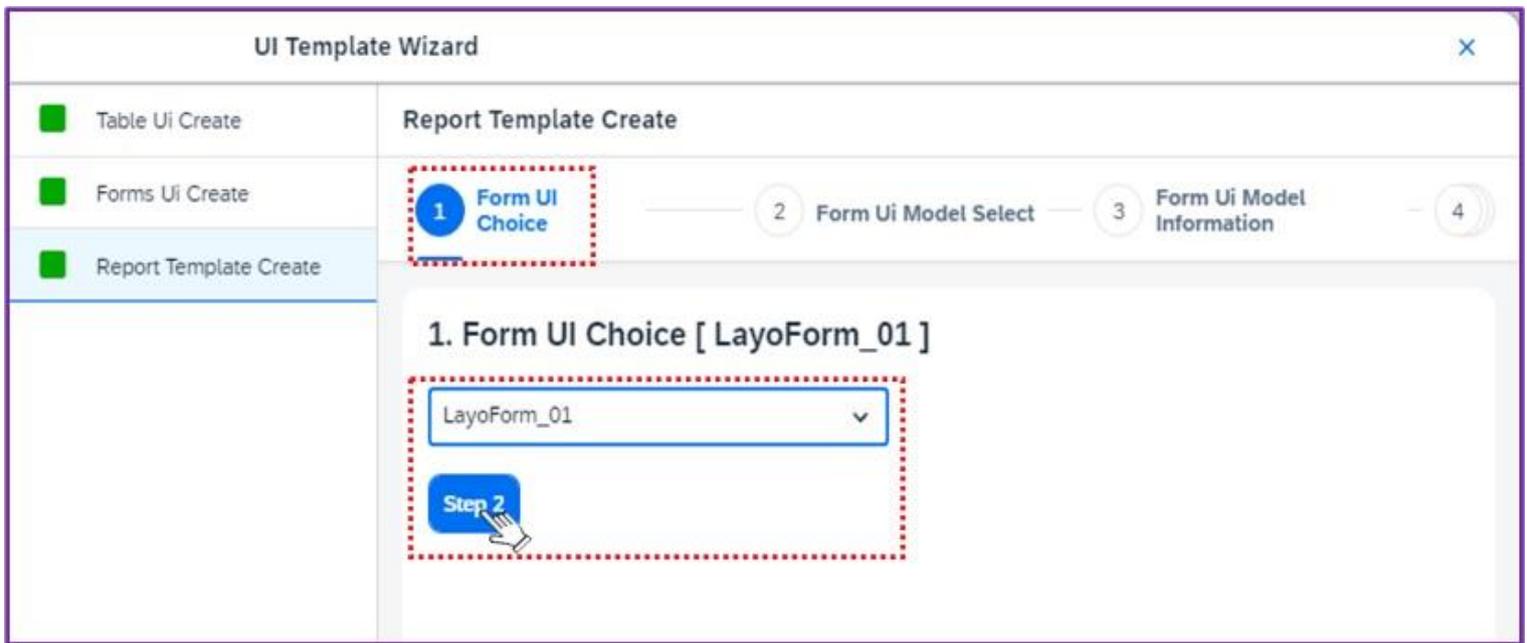
2. "Form"과 "Table" UI를 삽입 할 위치를 선택한 후 UI Template Wizard Button Click.



3. UI Template Wizard 화면에서 "Report Template Create" Tab을 선택. (녹색 Box는 Data를 Binding 시킬 Model이 존재함을 의미.)

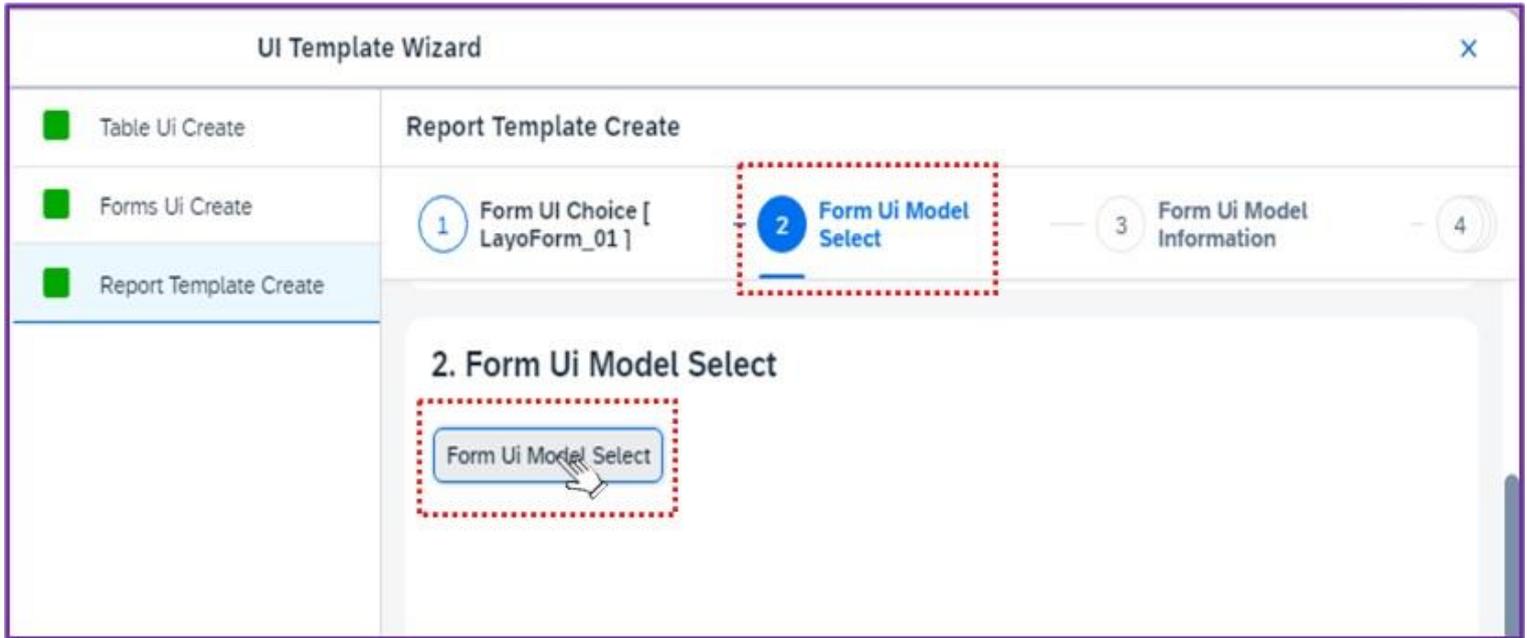
① Form UI Choice (본단원에서는 LayoForm_01로 진행)

※ 생성가능한 Form 종류 : SimpleForm, LayoForm_01

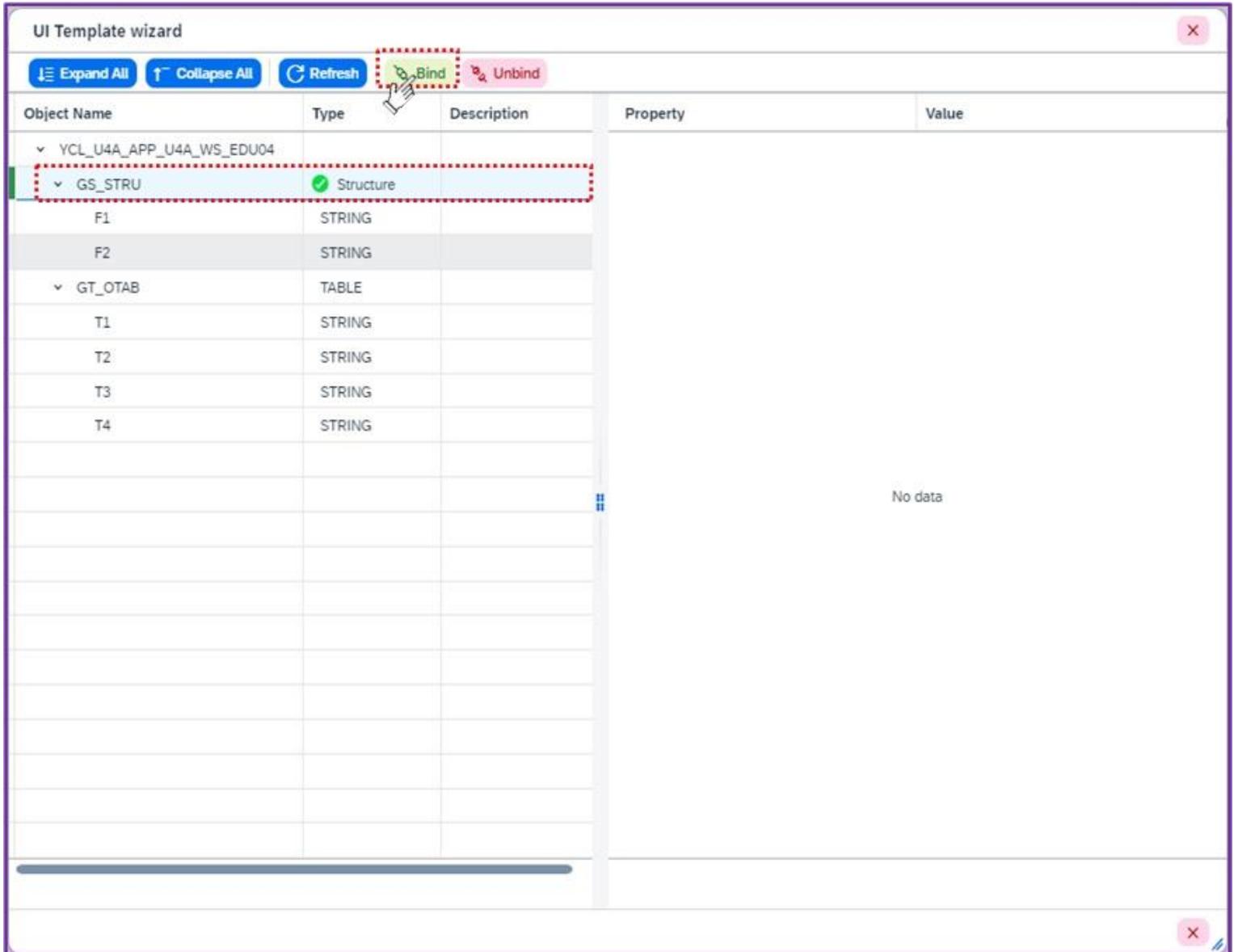


② Form UI Model Select

※ Form UI에 Binding 시키는 Model 명 (Model Select Button Click)

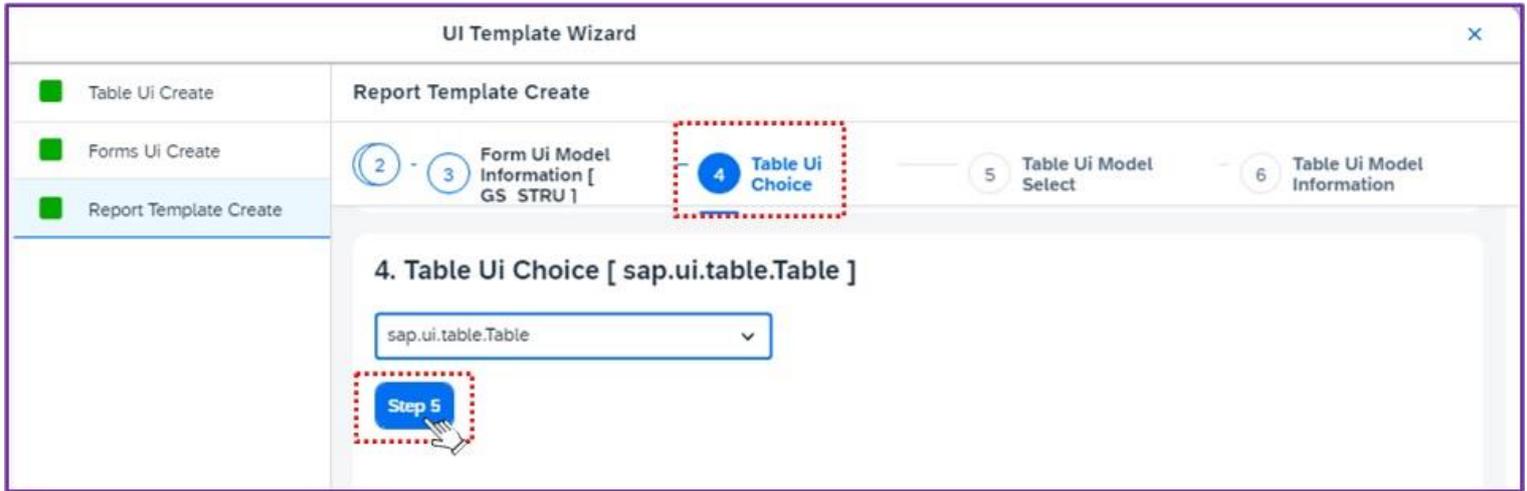


※ Binding할 Model Click 후 "Bind" Button Click.



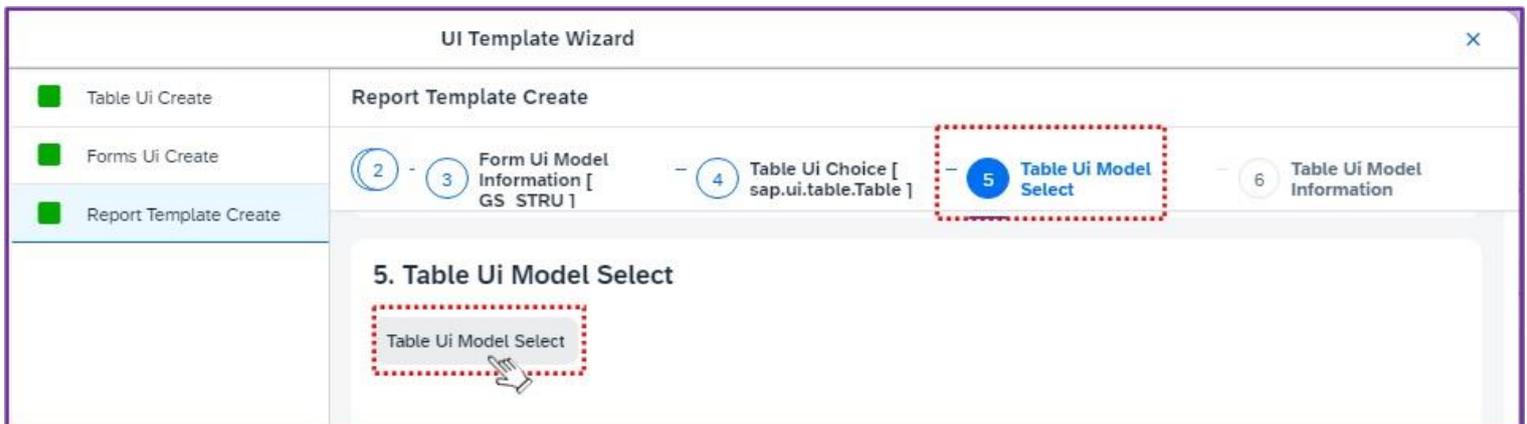
③ Form Ui Model Information (Binding 되어있는 Structure의 Field 정보 및 Option 설정).

※ Ui type, Label 등을 설정하고 Step 4 버튼 클릭.

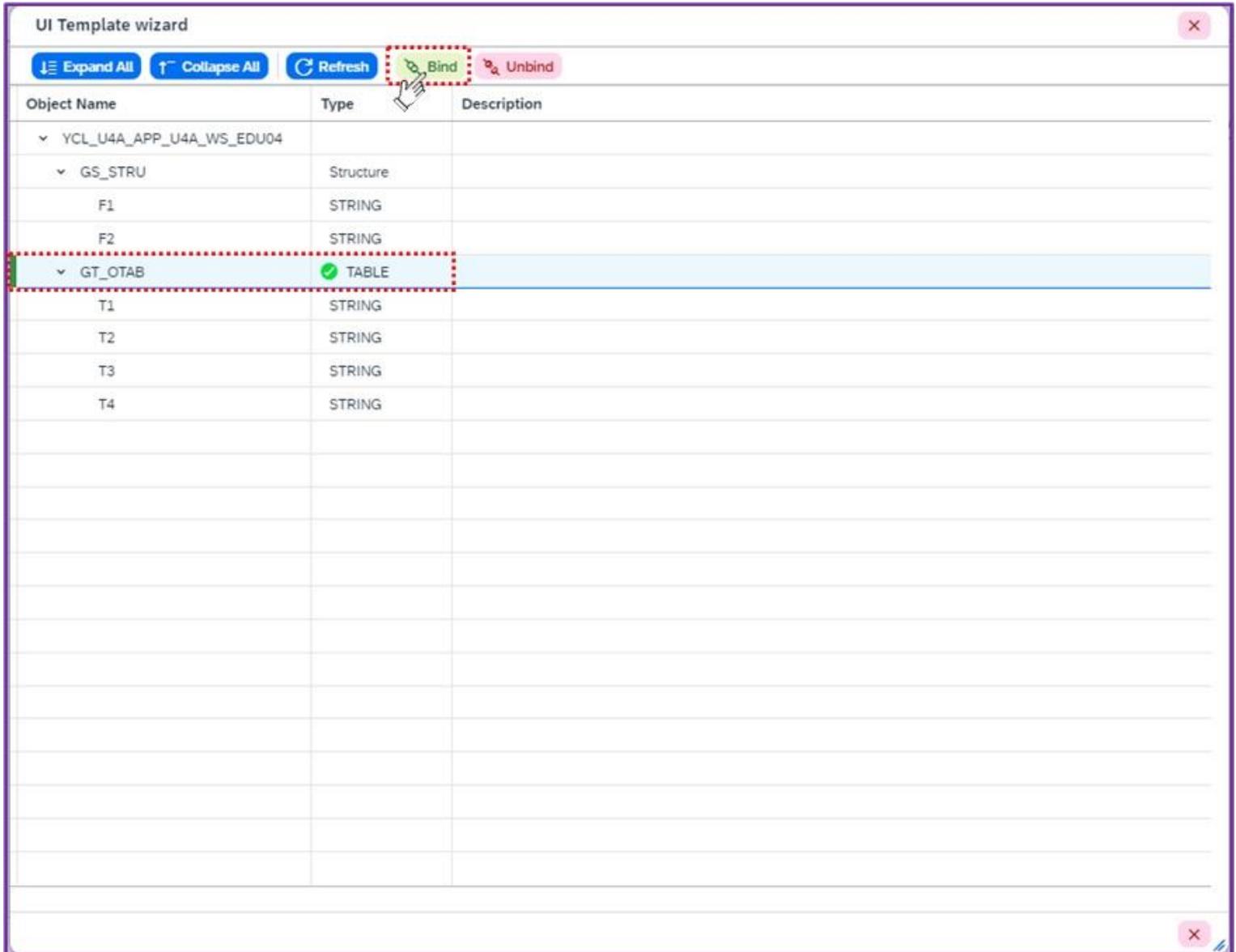


⑤ Table Ui Model Select

※ Table UI에 Binding 시키는 Model 명 (Model Select Button Click)



※ Binding할 Model Click 후 "Bind" Button Click.



⑥ Table Ui Model Information (Binding 되어있는 Table의 Field 정보 및 Option 설정).

※ Ui type, Label 등을 설정하고 생성 아이콘 클릭.

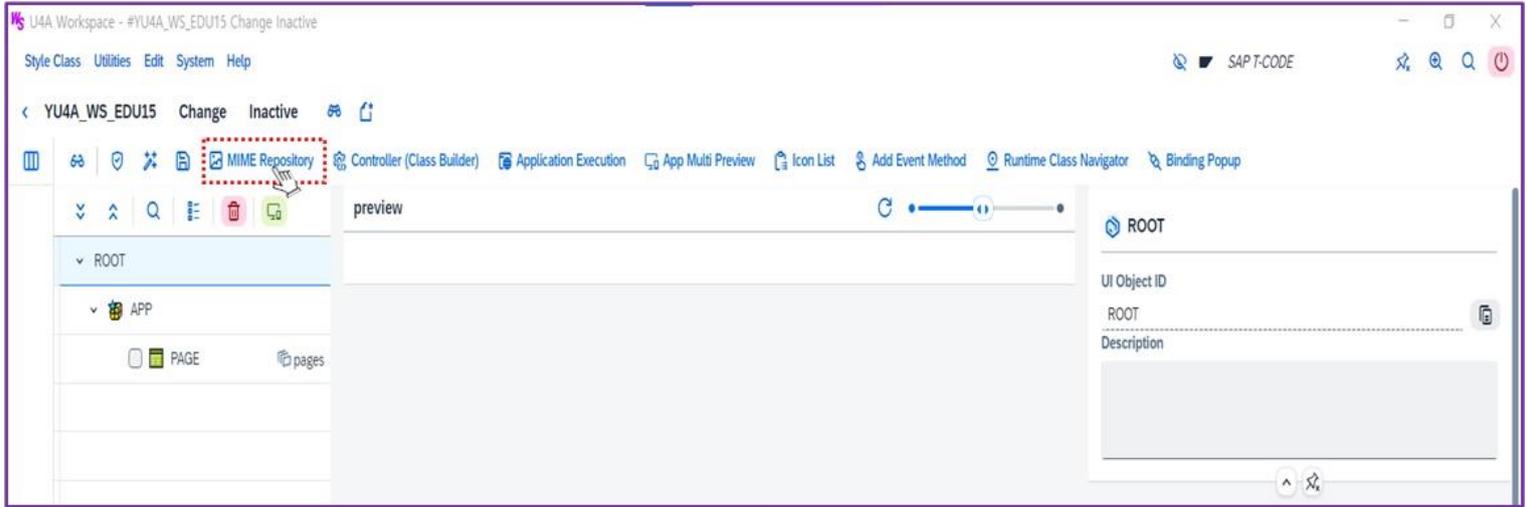
The screenshot displays the U4A IDE interface. On the left is a project tree with the following structure:

- ROOT
 - APP
 - PAGE (pages)
 - PANEL1 (content)
 - OVERFLOWTOOLBAR1 (headerToolbar)
 - FORM1 (content)
 - PANEL2 (content)
 - TABLE1 (content)

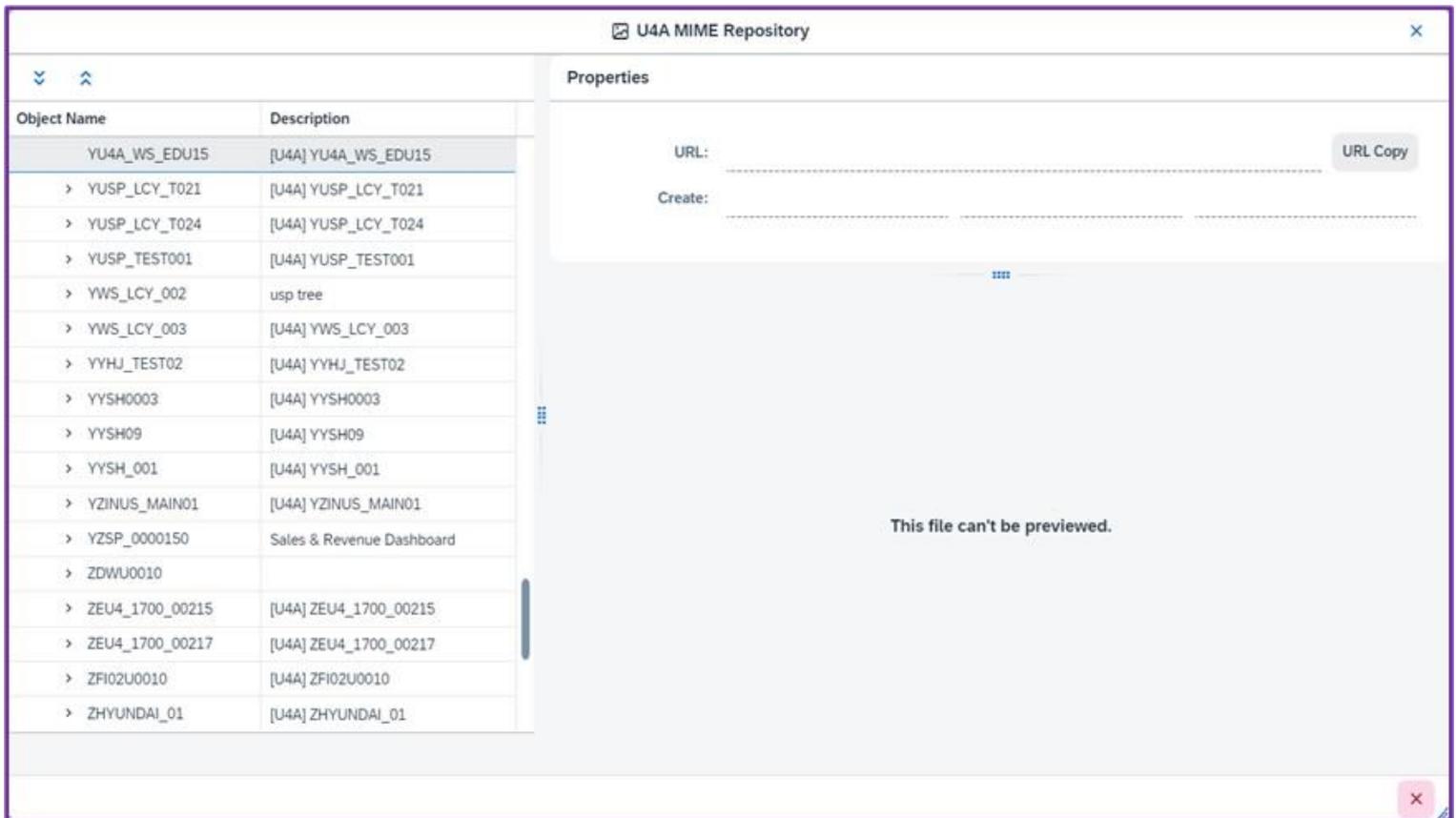
On the right is a preview window titled "preview" with a search interface. The "Search Condition" section includes an "INPUT:" field and a "TEXT:" field. Below it, the "Search Result" section shows a table with four columns: COL1, COL2, COL3, and COL4. The table is currently empty, displaying "No data".

COL1	COL2	COL3	COL4
No data			

5.17 MIME Repository 기능



2. MIME Repository를 실행하면 Popover창이 나타나는데 이때 자기가 접속한 application 이외에는 폴더를 생성, 삭제 할 수 없다.



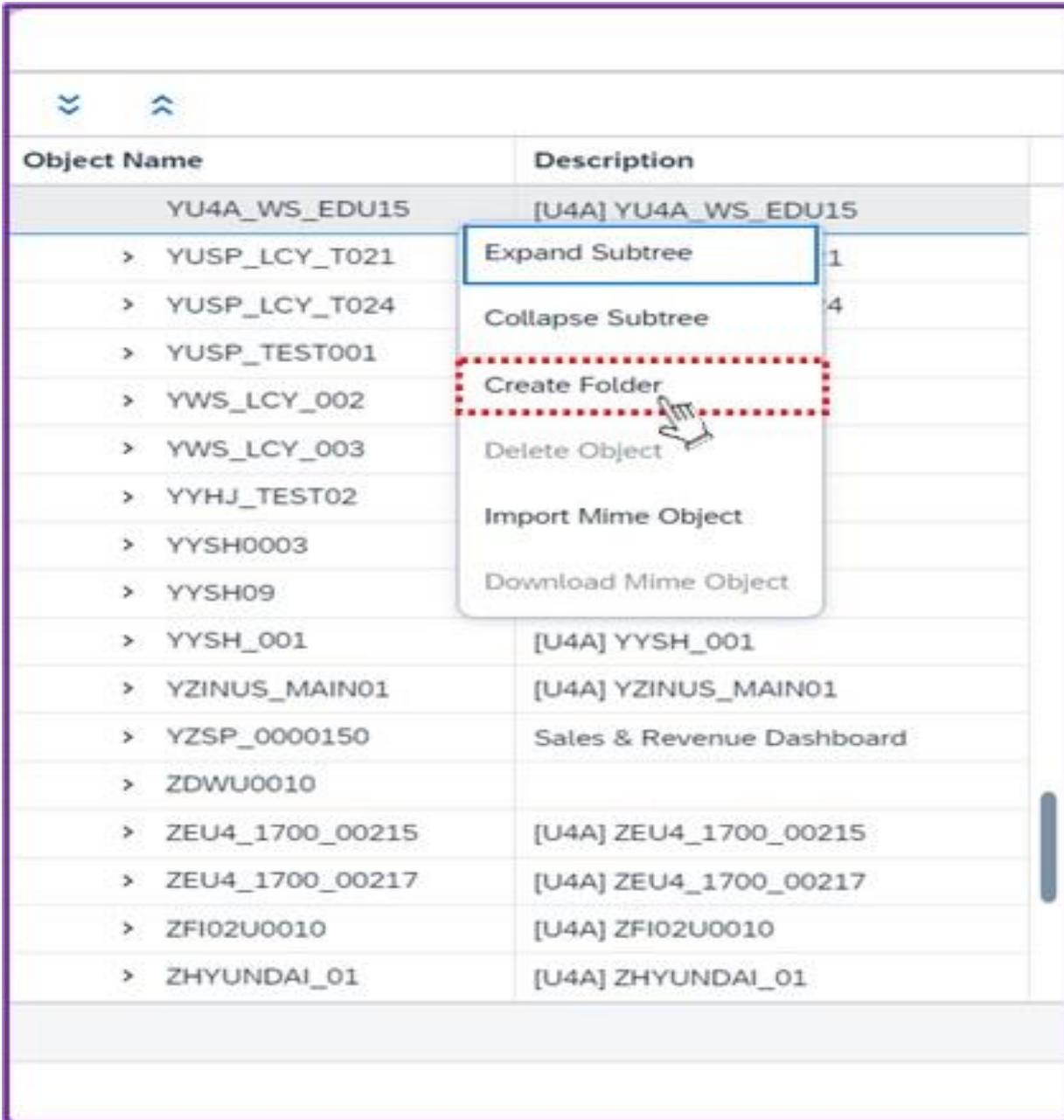
Object Name	Description
YU4A_WS_EDU15	[U4A] YU4A_WS_EDU15
> YUSP_LCY_T021	[U4A] YUSP_LCY_T021
> YUSP_LCY_T024	[U4A] YUSP_LCY_T024
> YUSP_TEST001	[U4A] YUSP_TEST001
> YWS_LCY_002	[U4A] YWS_LCY_002
> YWS_LCY_003	[U4A] YWS_LCY_003
> YYHJ_TEST02	[U4A] YYHJ_TEST02
> YYSH0003	[U4A] YYSH0003
> YYSH09	[U4A] YYSH09
> YYSH_001	[U4A] YYSH_001
> YZINUS_MAIN01	[U4A] YZINUS_MAIN01
> YZSP_0000150	Sales & Revenue Dashboard
> ZDWU0010	
> ZEU4_1700_00215	[U4A] ZEU4_1700_00215
> ZEU4_1700_00217	[U4A] ZEU4_1700_00217
> ZFI02U0010	[U4A] ZFI02U0010
> ZHYUNDAI_01	[U4A] ZHYUNDAI_01

- Expand Subtree
- Collapse Subtree
- Create Folder
- Delete Object
- Import Mime Object
- Download Mime Object

Object Name	Description
YU4A_WS_EDU15	[U4A] YU4A_WS_EDU15
> YUSP_LCY_T021	[U4A] YUSP_LCY_T021
> YUSP_LCY_T024	[U4A] YUSP_LCY_T024
> YUSP_TEST001	[U4A] YUSP_TEST001
> YWS_LCY_002	[U4A] YWS_LCY_002
> YWS_LCY_003	[U4A] YWS_LCY_003
> YYHJ_TEST02	[U4A] YYHJ_TEST02
> YYSH0003	[U4A] YYSH0003
> YYSH09	[U4A] YYSH09
> YYSH_001	[U4A] YYSH_001
> YZINUS_MAIN01	[U4A] YZINUS_MAIN01
> YZSP_0000150	Sales & Revenue Dashboard
> ZDWU0010	
> ZEU4_1700_00215	[U4A] ZEU4_1700_00215
> ZEU4_1700_00217	[U4A] ZEU4_1700_00217
> ZFI02U0010	[U4A] ZFI02U0010
> ZHYUNDAI_01	[U4A] ZHYUNDAI_01

- Expand Subtree
- Collapse Subtree
- Create Folder
- Delete Object
- Import Mime Object
- Download Mime Object

3. 자신이 접속한 application에서 Create Folder를 Click하면 Folder를 생성 할 수 있다.



4-1. Folder "Description"을 입력하고 체크 버튼을 선택한다. 선택후에는 자신의 application 하위 항목에 folder가 추가된다.

[U4A] MIME Folder Create

Folder Name: *

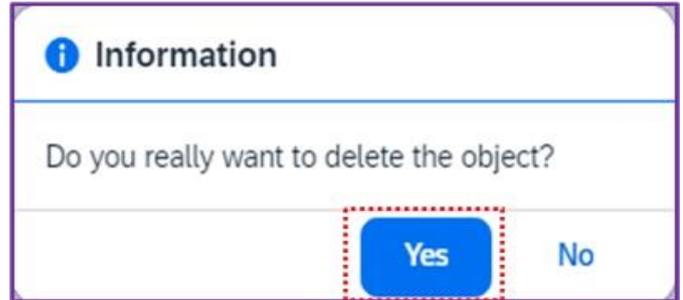
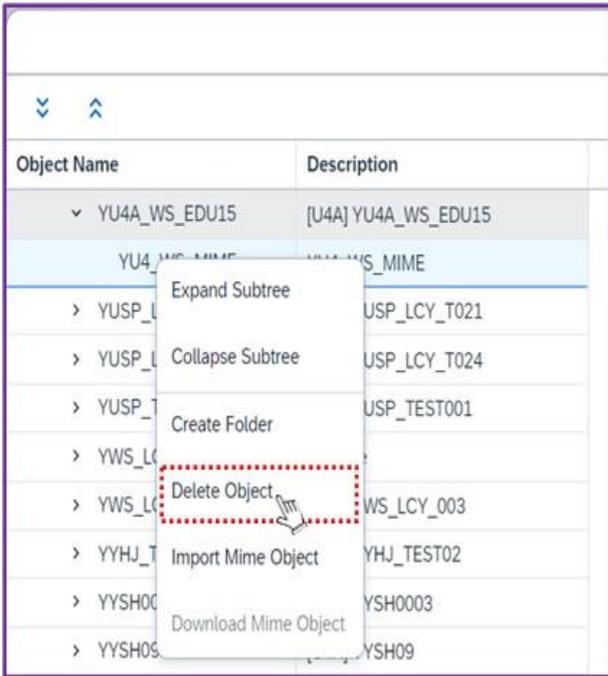
YU4_WS_MIME

Description:

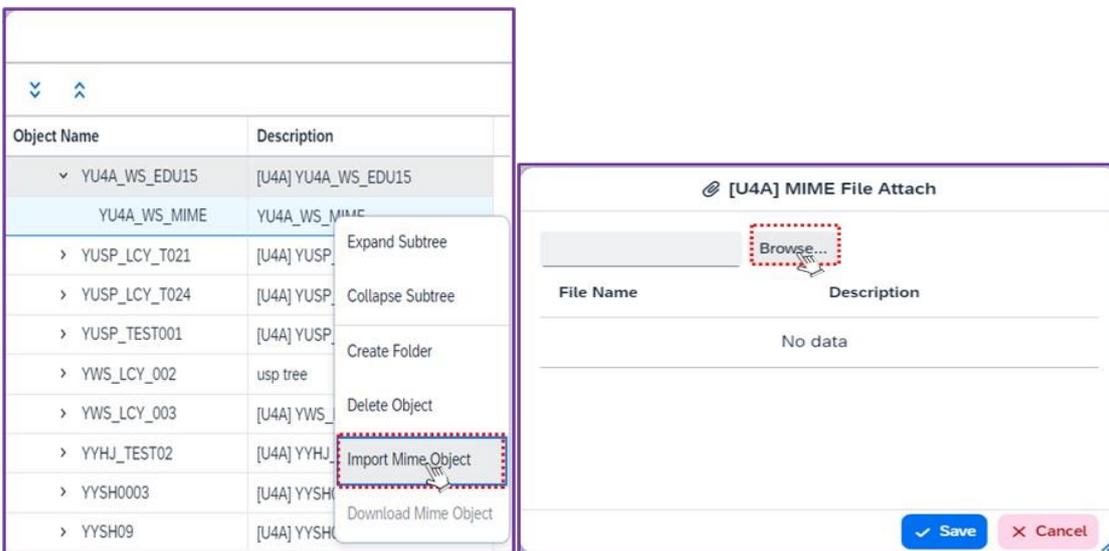
[U4A WS] MIME

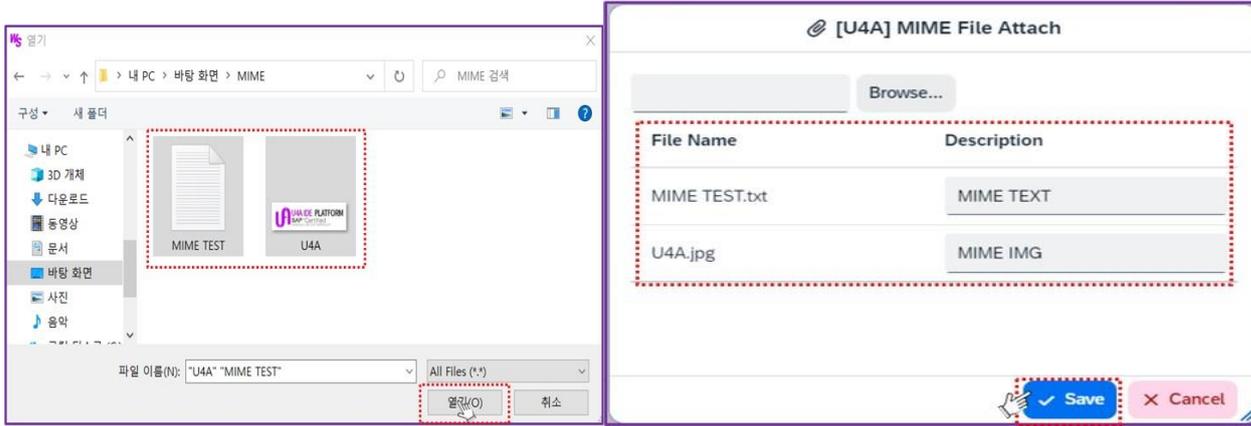
Object Name	Description
YU4A_WS_EDU15	[U4A] YU4A_WS_EDU15
YU4_WS_MIME	YU4_WS_MIME
YUSP_LCY_T021	[U4A] YUSP_LCY_T021
YUSP_LCY_T024	[U4A] YUSP_LCY_T024
YUSP_TEST001	[U4A] YUSP_TEST001
YWS_LCY_002	usp tree
YWS_LCY_003	[U4A] YWS_LCY_003
YYHJ_TEST02	[U4A] YYHJ_TEST02
YYSH0003	[U4A] YYSH0003
YYSH09	[U4A] YYSH09
YYSH_001	[U4A] YYSH_001
YZINUS_MAIN01	[U4A] YZINUS_MAIN01
YZSP_0000150	Sales & Revenue Dashboard
ZDWU0010	
ZEU4_1700_00215	[U4A] ZEU4_1700_00215
ZEU4_1700_00217	[U4A] ZEU4_1700_00217
ZFI02U0010	[U4A] ZFI02U0010

4-2. Delete Object를 통해 생성된 Folder, Object를 삭제 할 수 있다.



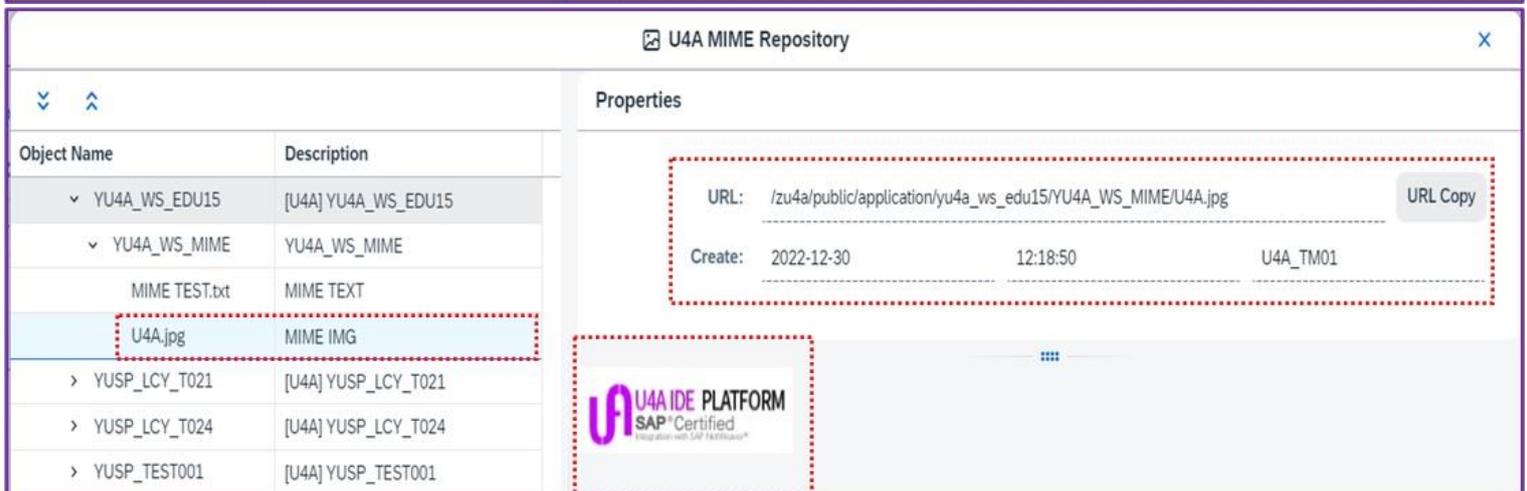
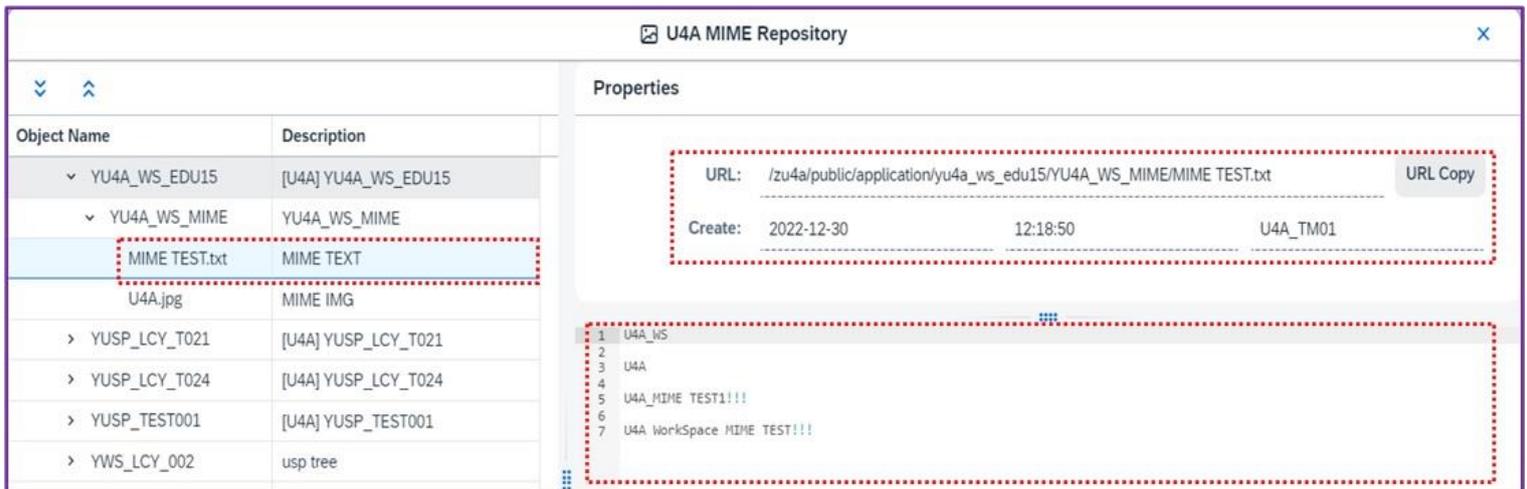
5. Import MIME Objects를 통해 application Folder안에 File을 담을 수 있다.



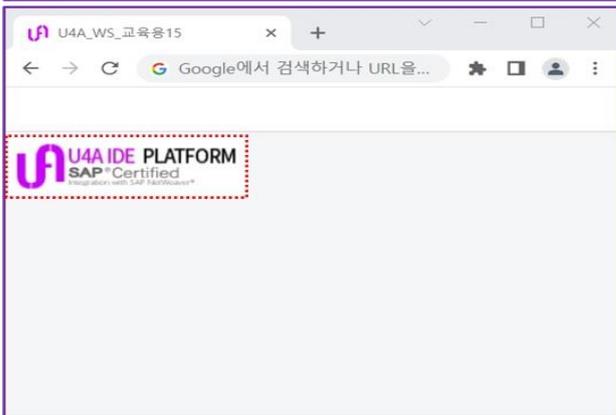
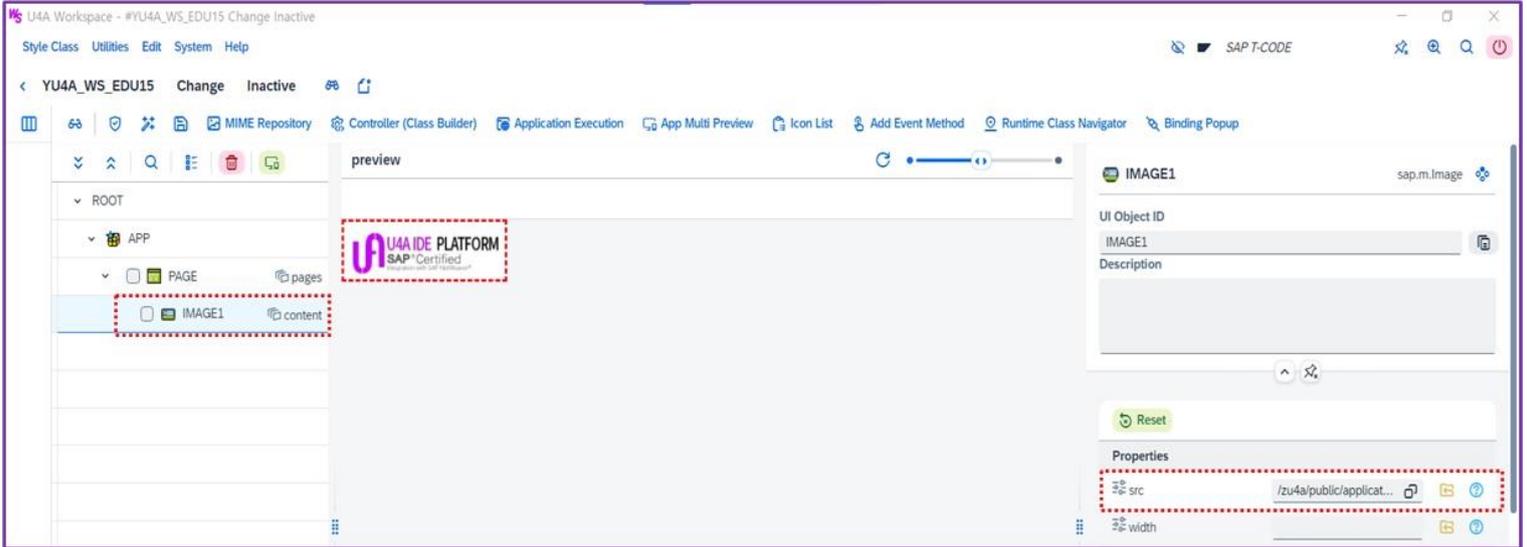


Object Name	Description
YU4A_WS_EDU15	[U4A] YU4A_WS_EDU15
YU4A_WS_MIME	YU4A_WS_MIME
MIME TEST.txt	MIME TEXT
U4A.jpg	MIME IMG

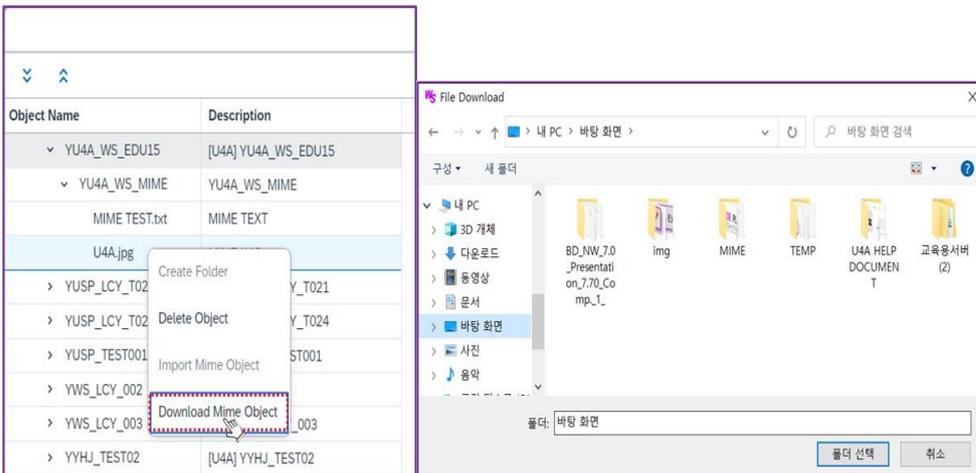
6. 담은 File을 Click했을때 Text 형식이나 Image 형식이면 오른쪽에 예시화면으로 나타난다.



7. "URL Copy" 버튼 클릭 후 알맞은 UI Object에 적용시키면 Web 화면에서도 똑같이 구현 되는 것을 확인 할 수 있다.



8. "Download MIME Object"를 통해 File을 다운로드 할 수 있다.



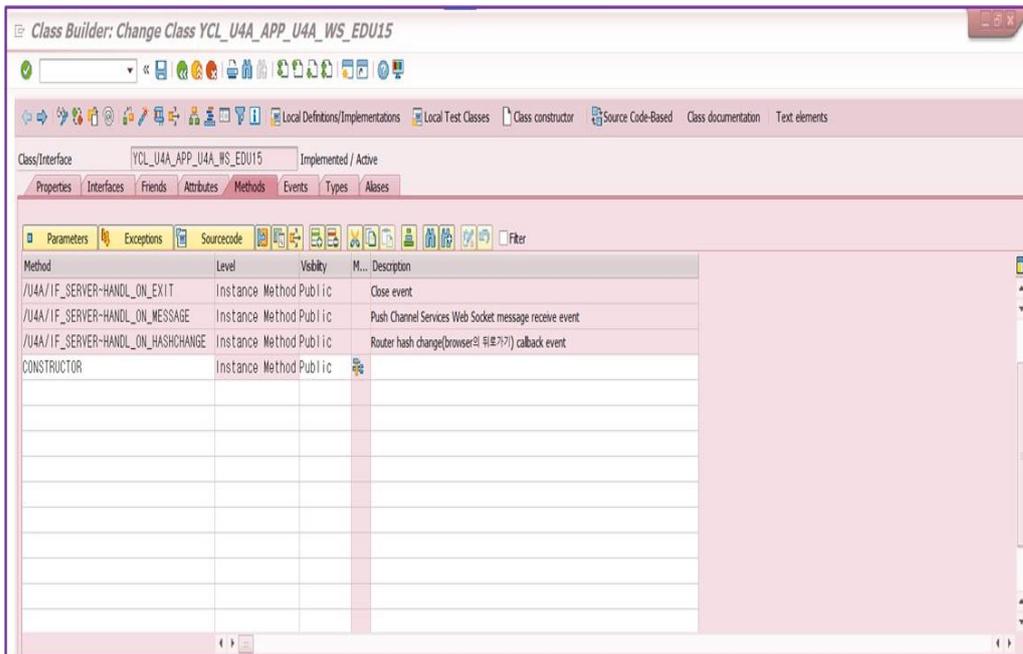
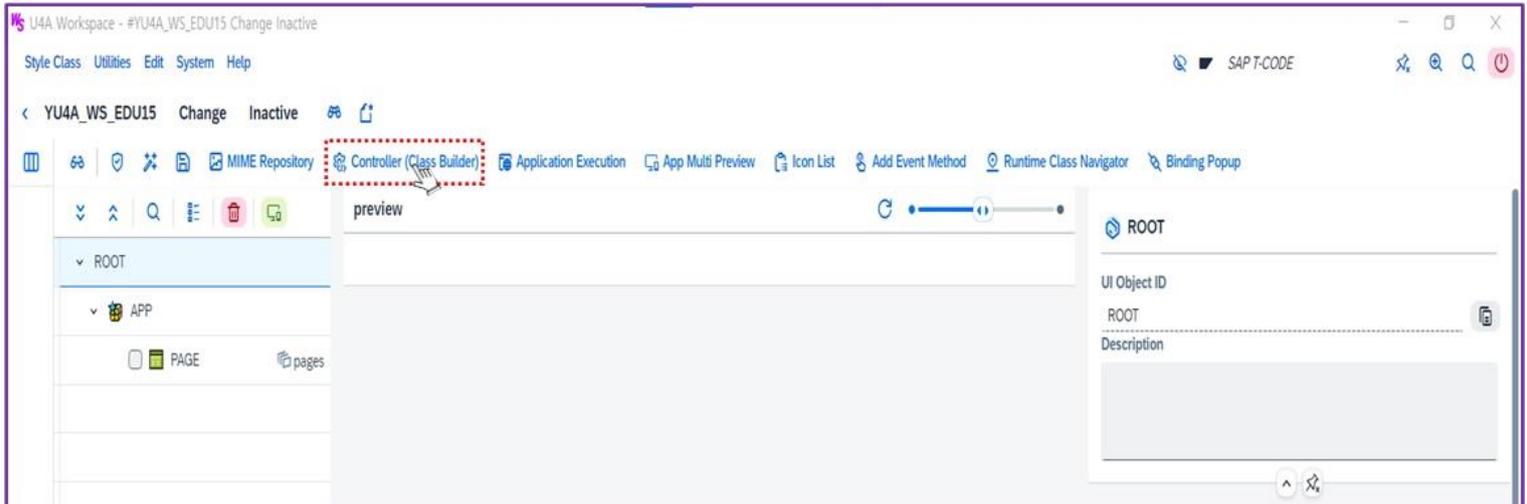


Text나 Image를 Upload 할 때 이름이 한글로 되어있으면 File을 불러올 수 없기 때문에 반드시 영어 또는 숫자로 변환해야 한다.

5.18 Controller(Class builder) 기능

■ 목적 : A pplication에서 사용할 ABAP Business Logic을 구현 하거나, binding 시 필요한 Table을 만드는 곳은 모두 Controller에서 관리한다

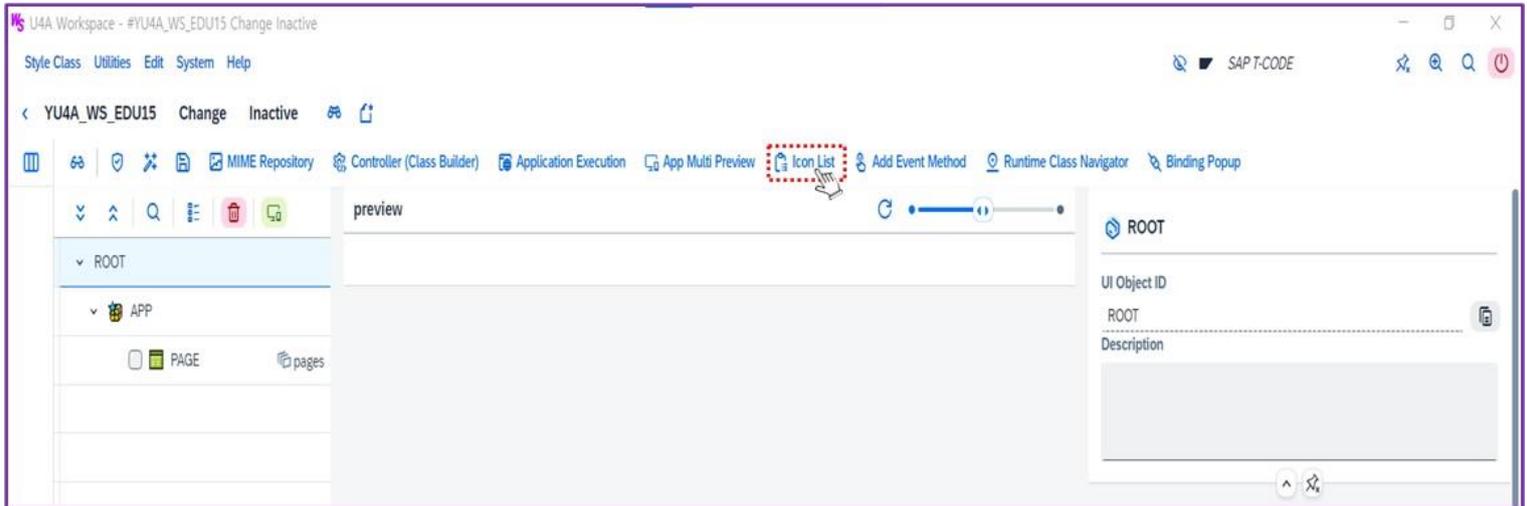
1. 하단의 붉은 네모칸을 Click하면 Controller 내부로 화면이 전환된다.



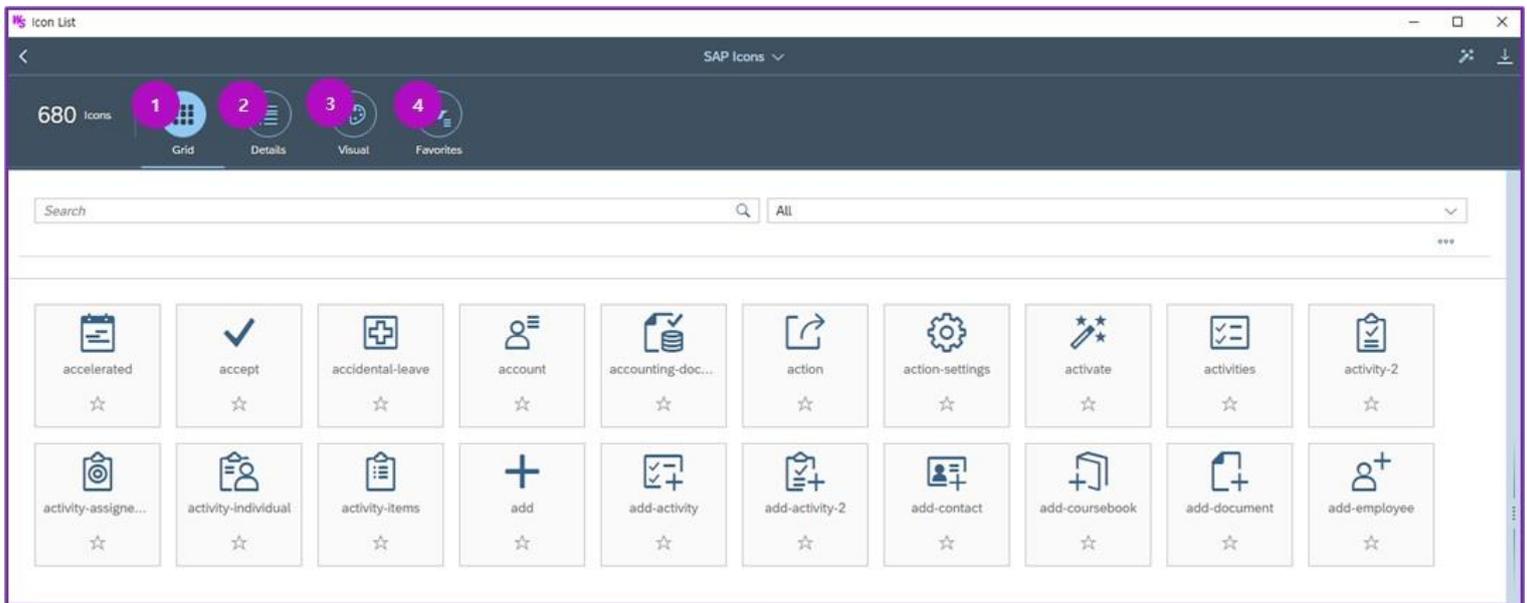
5.19 Icon list 기능

■ 목적 : 특정 행동이나 설명을 간단한 Image로 보여주고 싶을 때 Icon list를 이용한다.

1. 하단의 붉은색 네모칸에 있는 Icon list를 클릭한다.

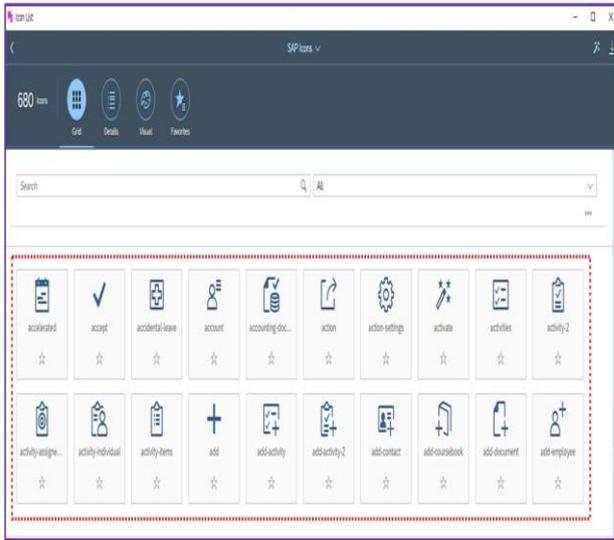


2. 하단 이미지의 붉은색 네모칸에서 Grid, Details, visual, Favorites 기능중 원하는 기능을 사용한다.



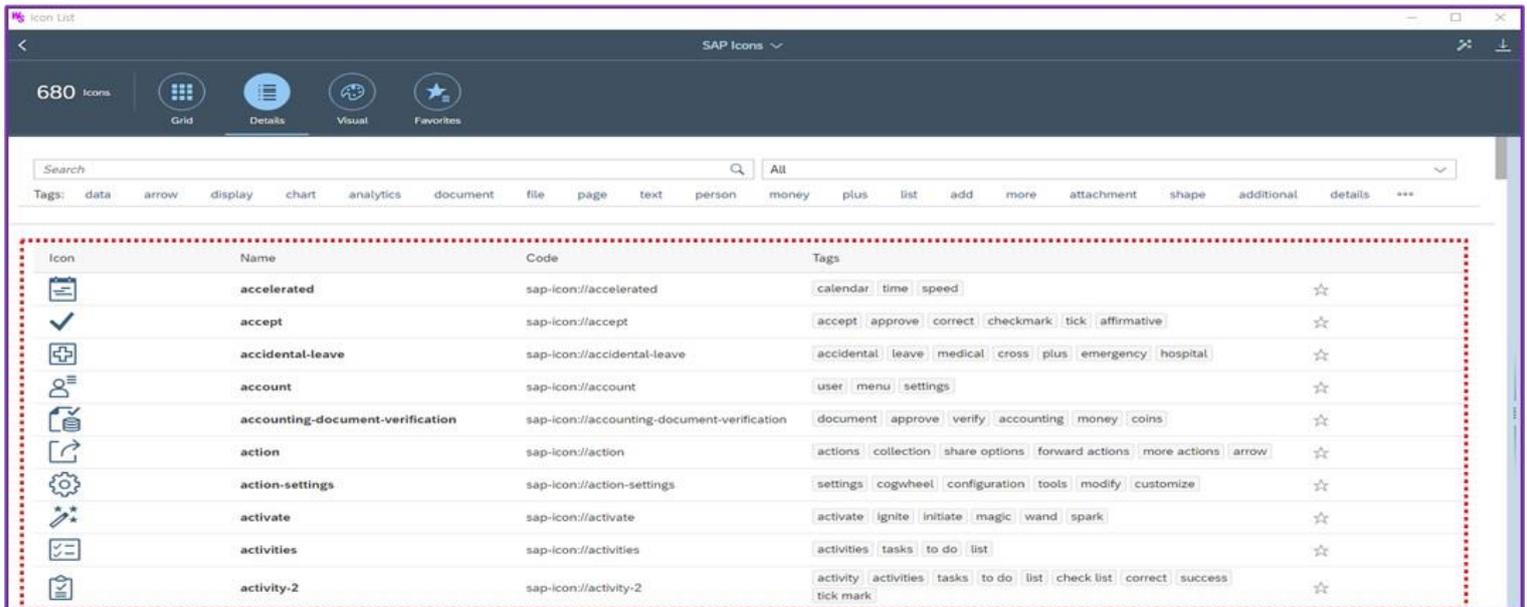
1) "Grid"

Icon 모양과 이름, Favorites가 가능한 간소한 형태로 보여준다.



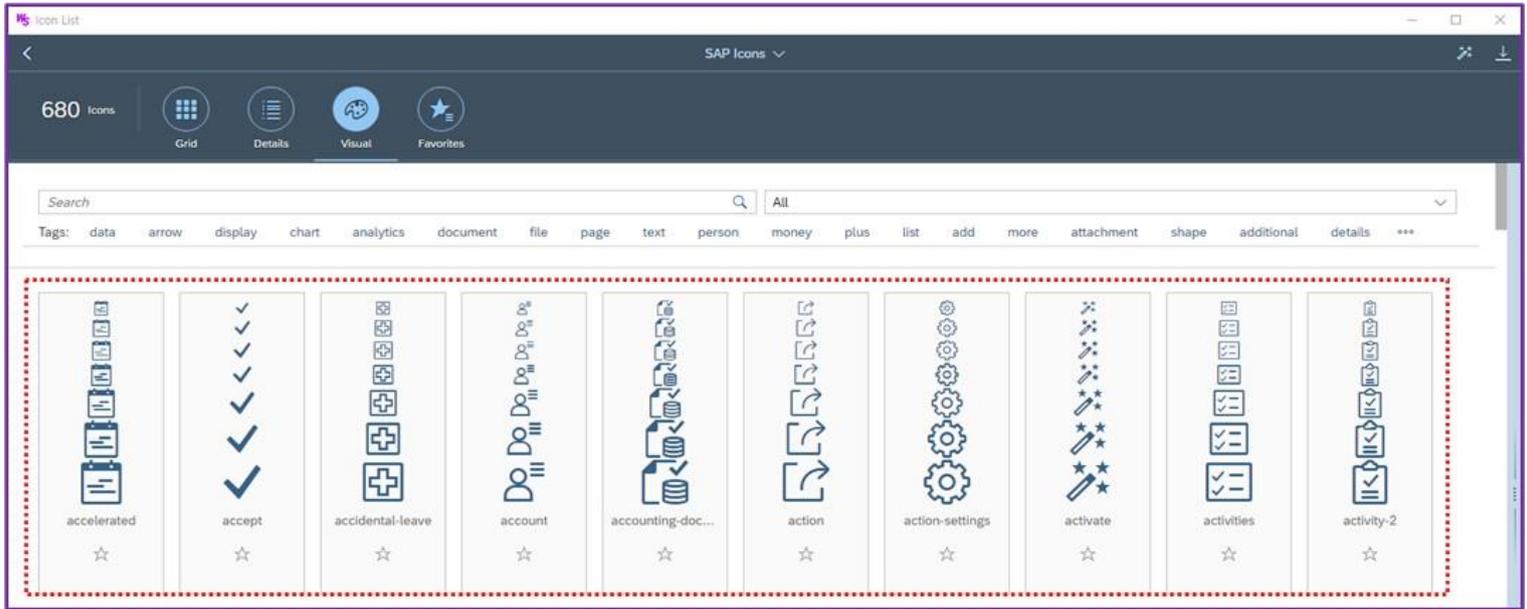
2) "Details"

Icon 모양, 이름, Code, tag, Favorites가 모두 나타나는 형태로 보여준다.



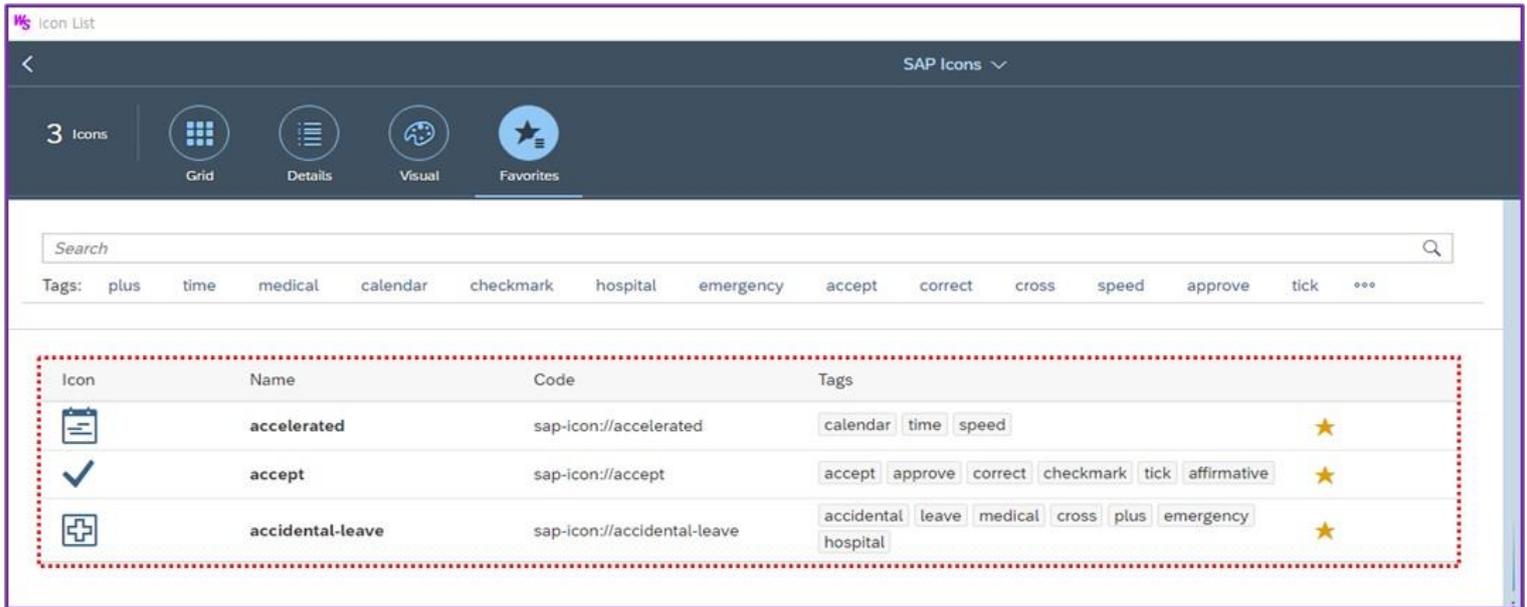
3) "Visual"

한 Icon을 다양한 크기로 보여준다.



4) "Favorites"

자신이 설정한 Favorites만 나타나는 항목이다.



3. 사용할 icon을 선택한 후 Copy의 첫번째 항목을 복사한다.

(우측에 파란색 Button을 눌렀을 시 clipboard에 복사되었다고 나오지만 현재 작동이 원만하지 않아 직접 Drag 해서 복사하는 것을 권장한다.)

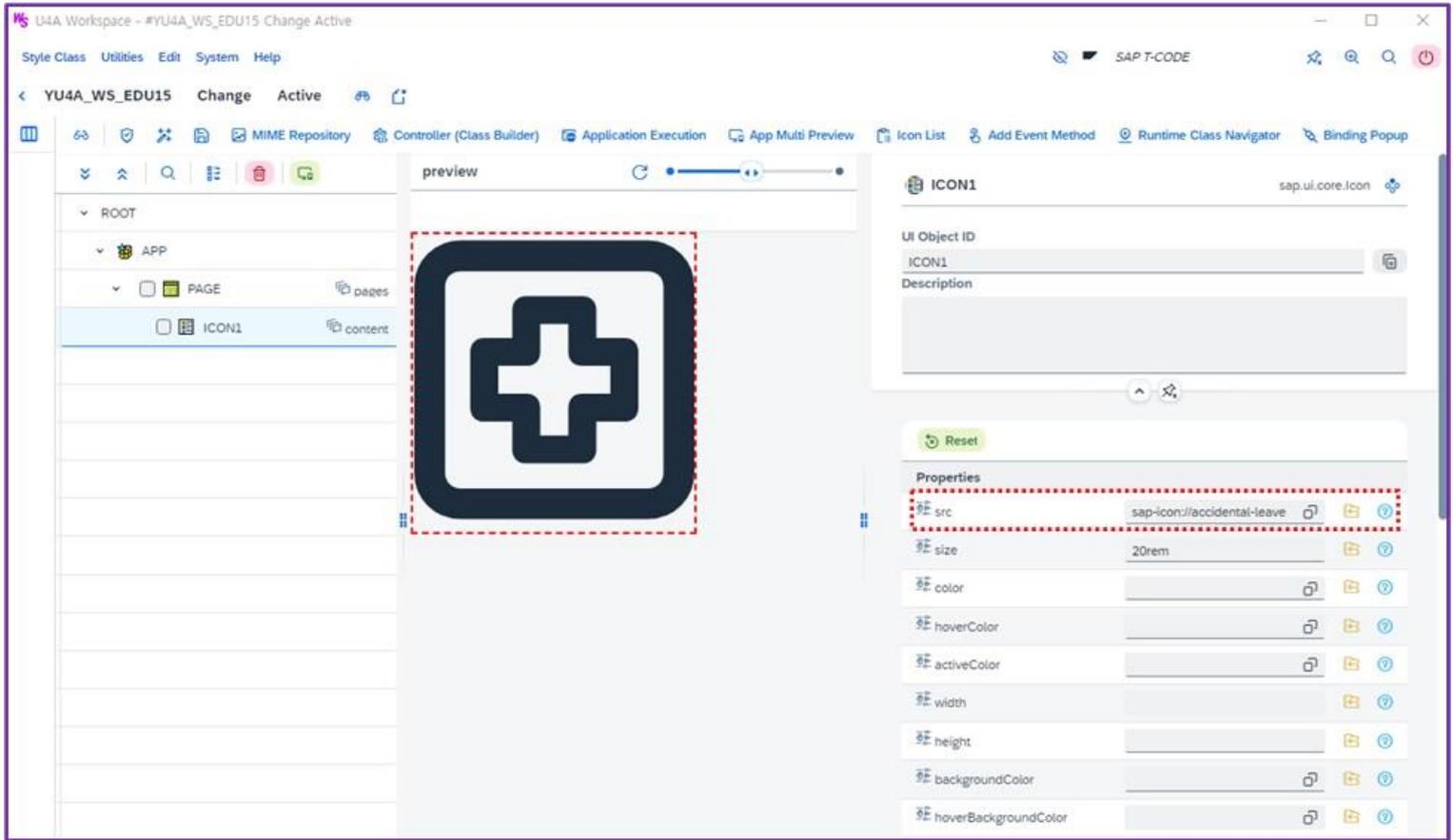
※ **Preview** : 선택한 ICON 모양 미리보기

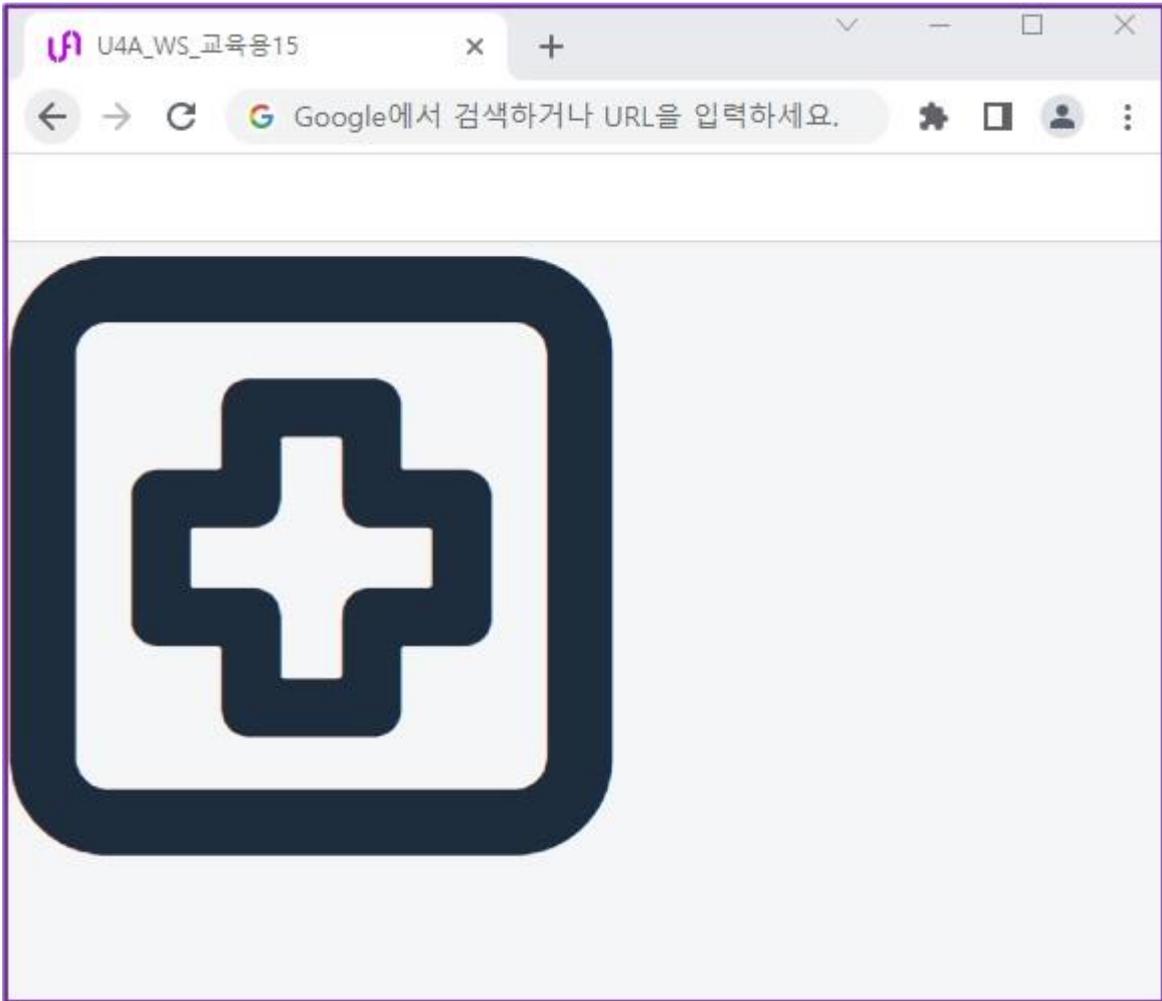
※ **Typical Use Cases** : 선택한 ICON이 사용된 예시



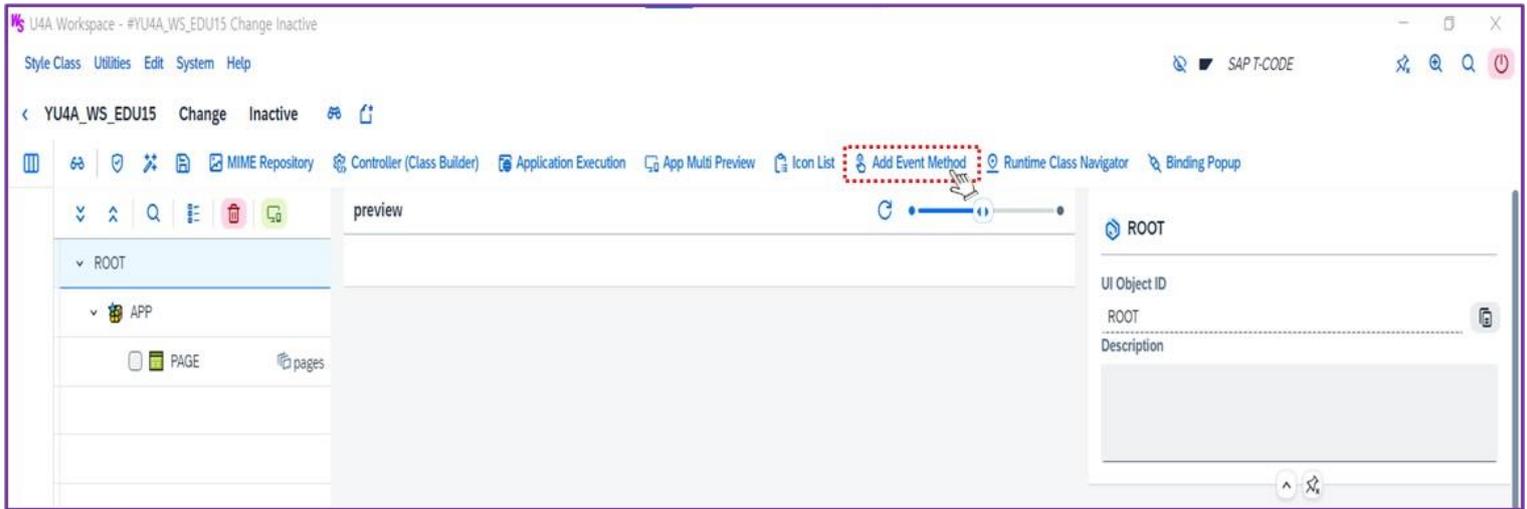
4. 복사한 주소를 UI Object에 담아주면 정상적으로 불러오는 것을 확인 할 수 있다.

U4A IDE HELP

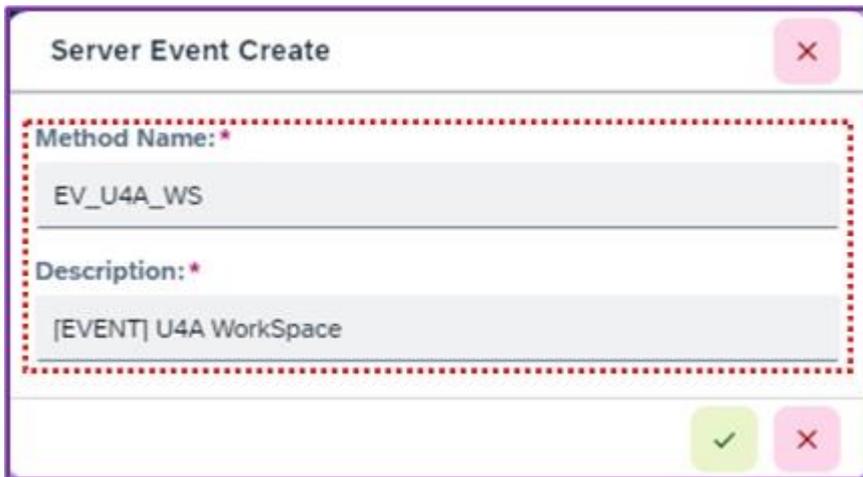




5.20 Add Event Method 기능



2. Add Event Method를 클릭하면 Event 생성 시 필요한 Name과 Description을 입력하는 창이 나오고 두 가지 항목을 모두 적는다.



3. Controller(Class builder)를 Click하면 Methods탭에 Events가 생성된 것을 확인 할 수 있다.

U4A IDE HELP

Class Builder: Change Class YCL_U4A_APP_U4A_WS_EDU15

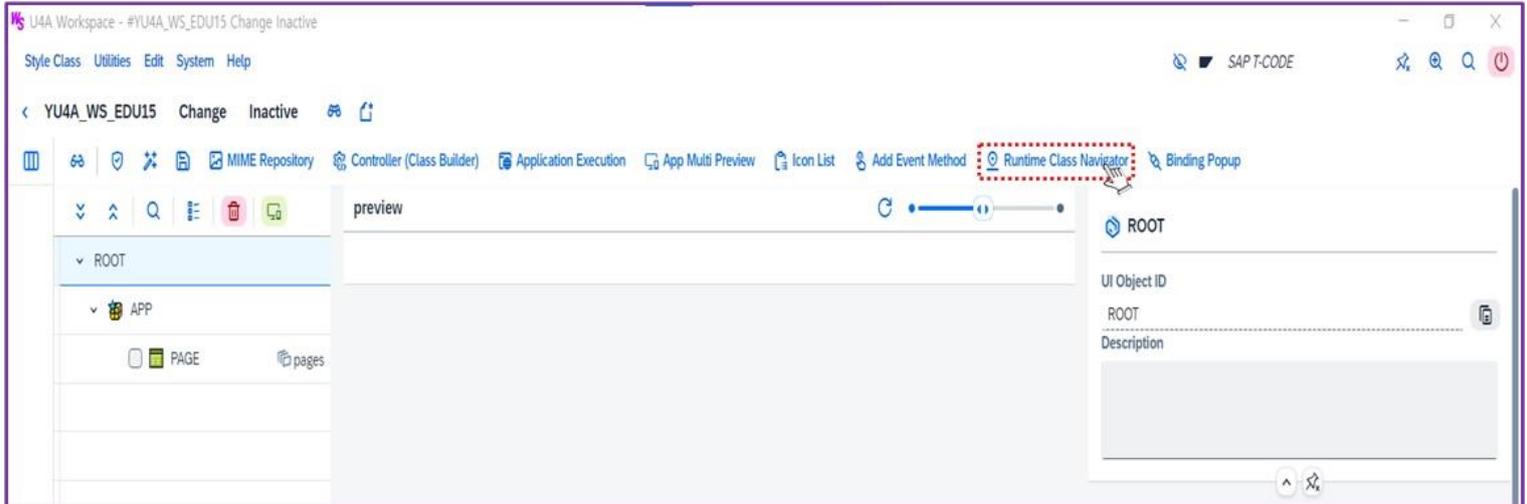
Class/Interface: **YCL_U4A_APP_U4A_WS_EDU15** Implemented / Active

Properties Interfaces Friends Attributes **Methods** Events Types Alases

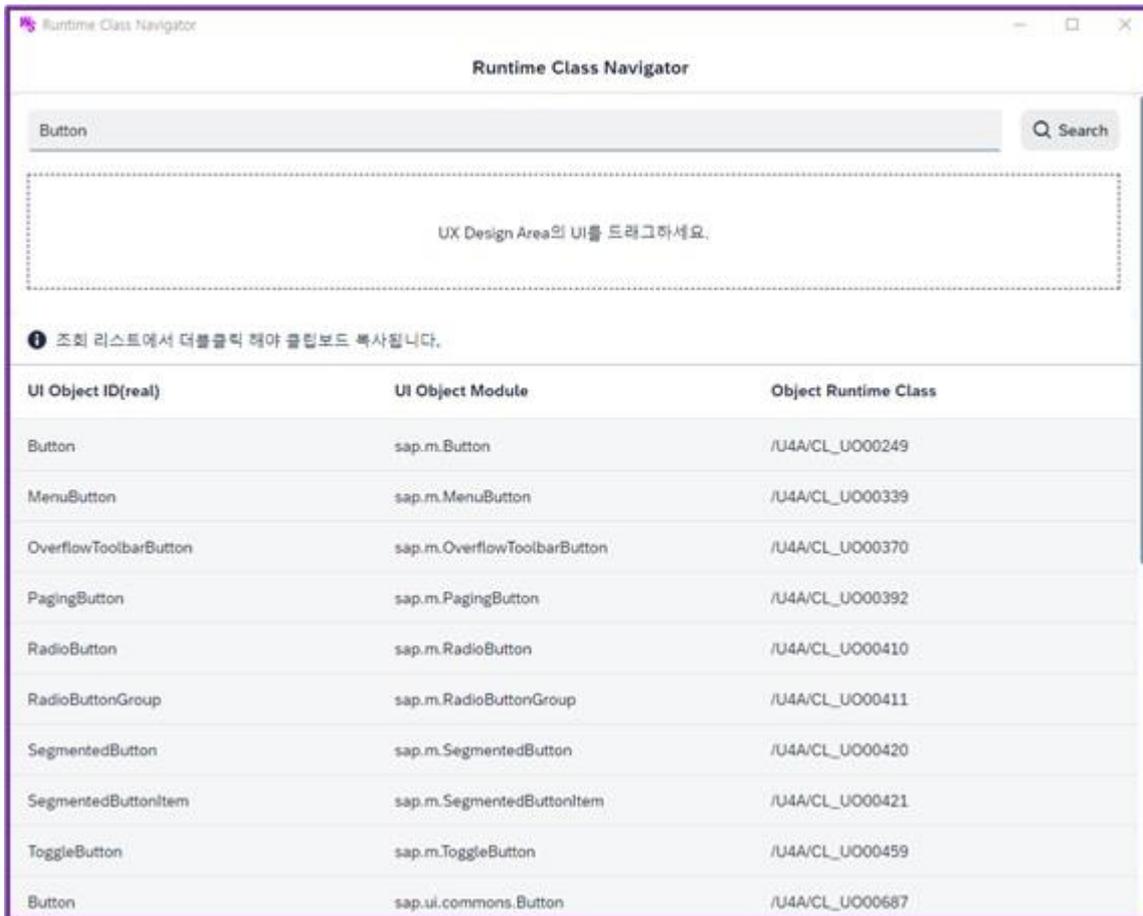
Parameters Exceptions Sourcecode

Method	Level	Visibility	M...	Description
/U4A/IF_SERVER~HANDL_ON_INIT	Instance Method	Public		Initialization event
/U4A/IF_SERVER~HANDL_ON_REQUEST	Instance Method	Public		Asynchronous event
/U4A/IF_SERVER~HANDL_ON_RESPONSE	Instance Method	Public		Not used
/U4A/IF_SERVER~HANDL_ON_EXIT	Instance Method	Public		Close event
/U4A/IF_SERVER~HANDL_ON_MESSAGE	Instance Method	Public		Push Channel Services Web Socket message receive event
/U4A/IF_SERVER~HANDL_ON_HASHCHANGE	Instance Method	Public		Router hash change(browser의 뒤로가기) callback event
CONSTRUCTOR	Instance Method	Public		
EV_U4A_WS	Instance Method	Public		[EVENT] U4A WorkSpace

5.21 RunTime Class Navigator 기능



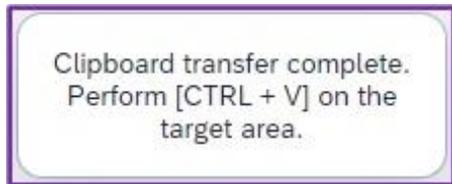
2.1 Navigator가 나타나면 자신이 변경하려는 UI Object의 ID를 검색하여 찾는다.



2.2 Navigator가 나타나면 자신이 변경하려는 UI Object를 Drag하여 Navigator Drag & Drop 영역에 놓는다.



3. 나타난 UI Object를 Double Click하면 Clipboard에 복사 되었다는 문구가 나타난 후 Navigator로 돌아온다.



4. 자신이 적용시키려는 위치에 붙여넣기를 시행하고 CALL METHOD를 통해 수행하려는 작업을 입력한다.

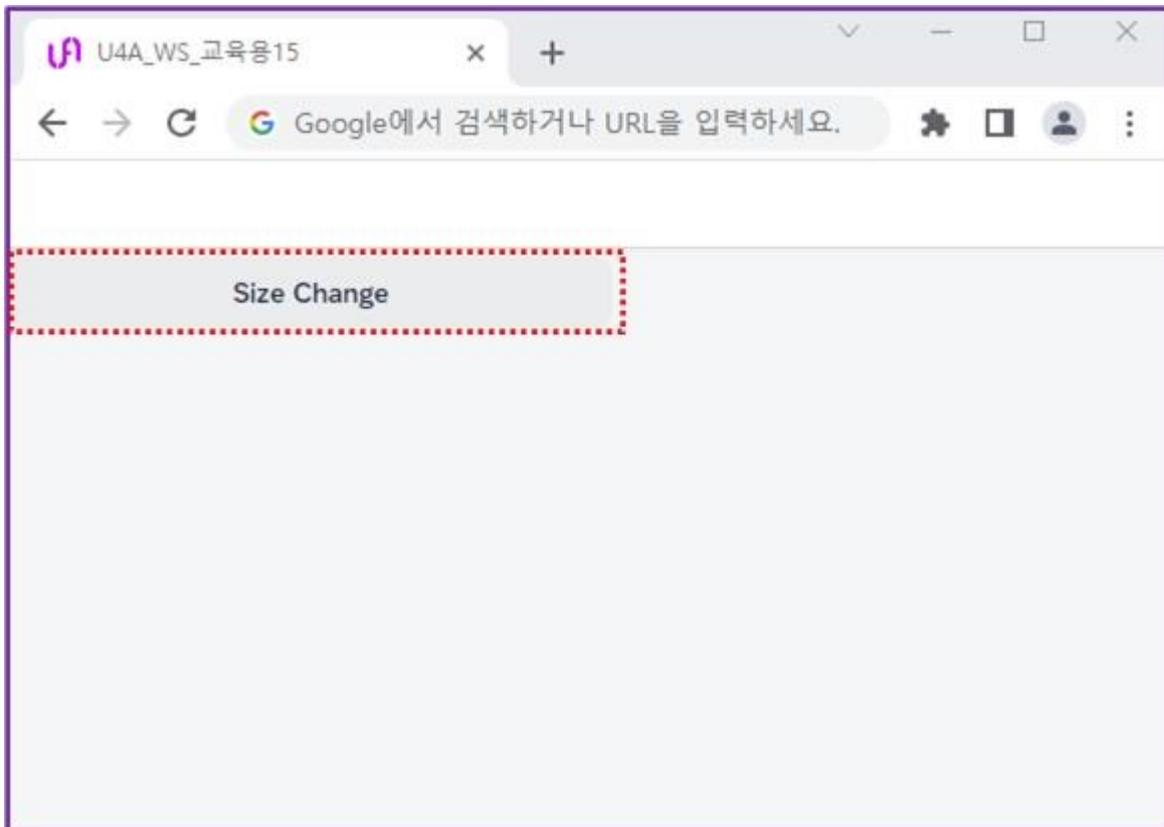
※ 붙여넣기를 하면 첫번째 사진과 같은 ABAP Logic만 뜬다

```
DATA LO_BUTTON TYPE REF TO /U4A/CL_U000249.
LO_BUTTON ?= ME->/U4A/IF_SERVER~AR_VIEW->GET_UI_INSTANCE( I_ID = '' ).
```

```

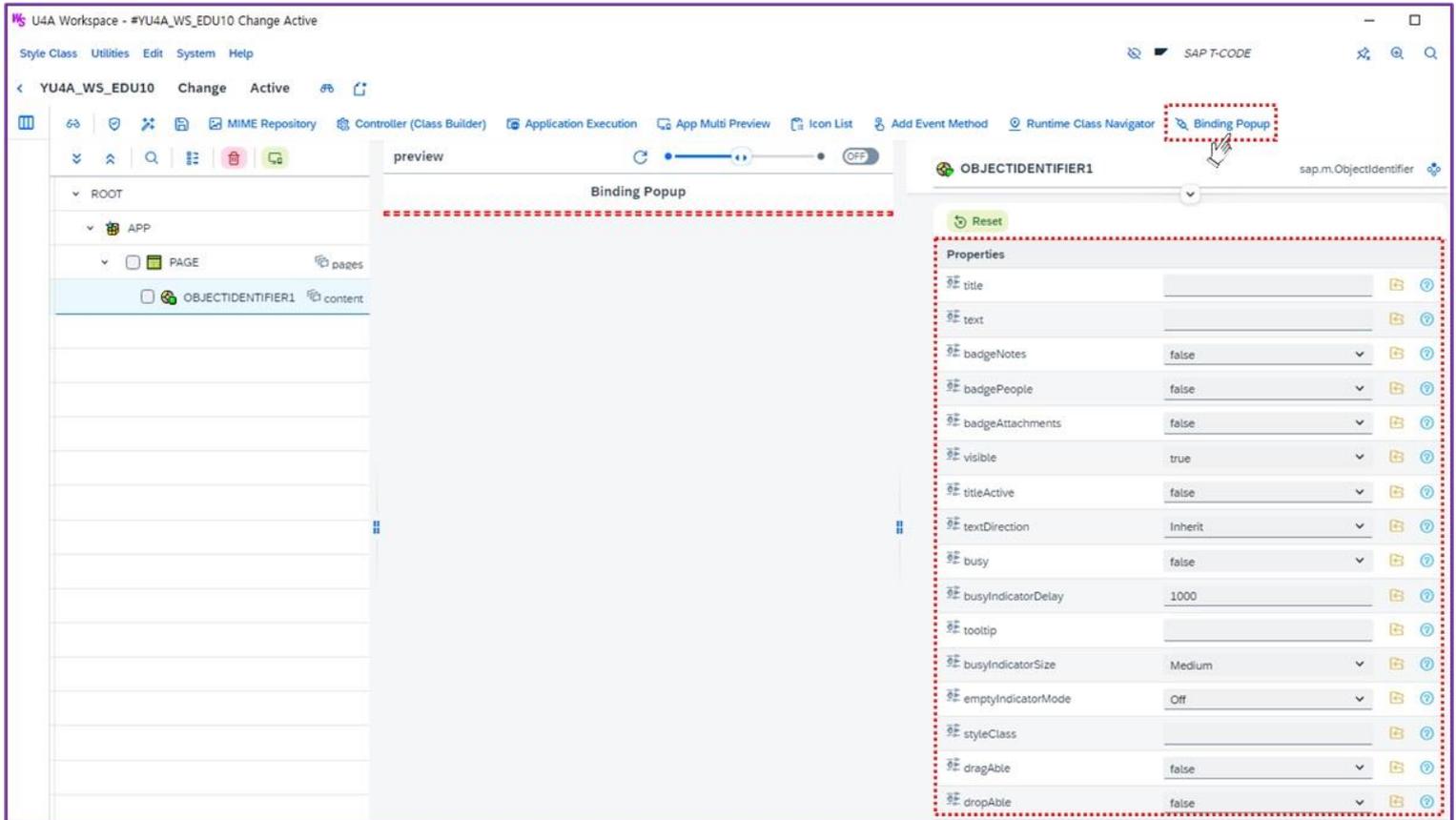
Method /U4A/IF_SERVER~HANDL_ON_INIT active
1  METHOD /U4A/IF_SERVER~HANDL_ON_INIT.
2
3  DATA LO_BUTTON TYPE REF TO /U4A/CL_U000249.
4  LO_BUTTON ?= ME->/U4A/IF_SERVER~AR_VIEW->GET_UI_INSTANCE( I_ID = 'BUTTON1' ).
5
6  CHECK LO_BUTTON IS BOUND.
7
8  CALL METHOD LO_BUTTON->SETWIDTH
9  EXPORTING
10     WIDTH = '300px'           " width
11 *   EXCEPTIONS
12 *     BIND_UI_NOT_HANDLE = 1   " Binding UI does not handling.
13 *     OTHERS              = 2
14
15 IF SY-SUBRC <> 0.
16 *   MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
17 *     WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.
18 ENDIF.
19
20 ENDMETHOD.
    
```

5.Web에 UI Object가 변경 된 것을 확인한다.

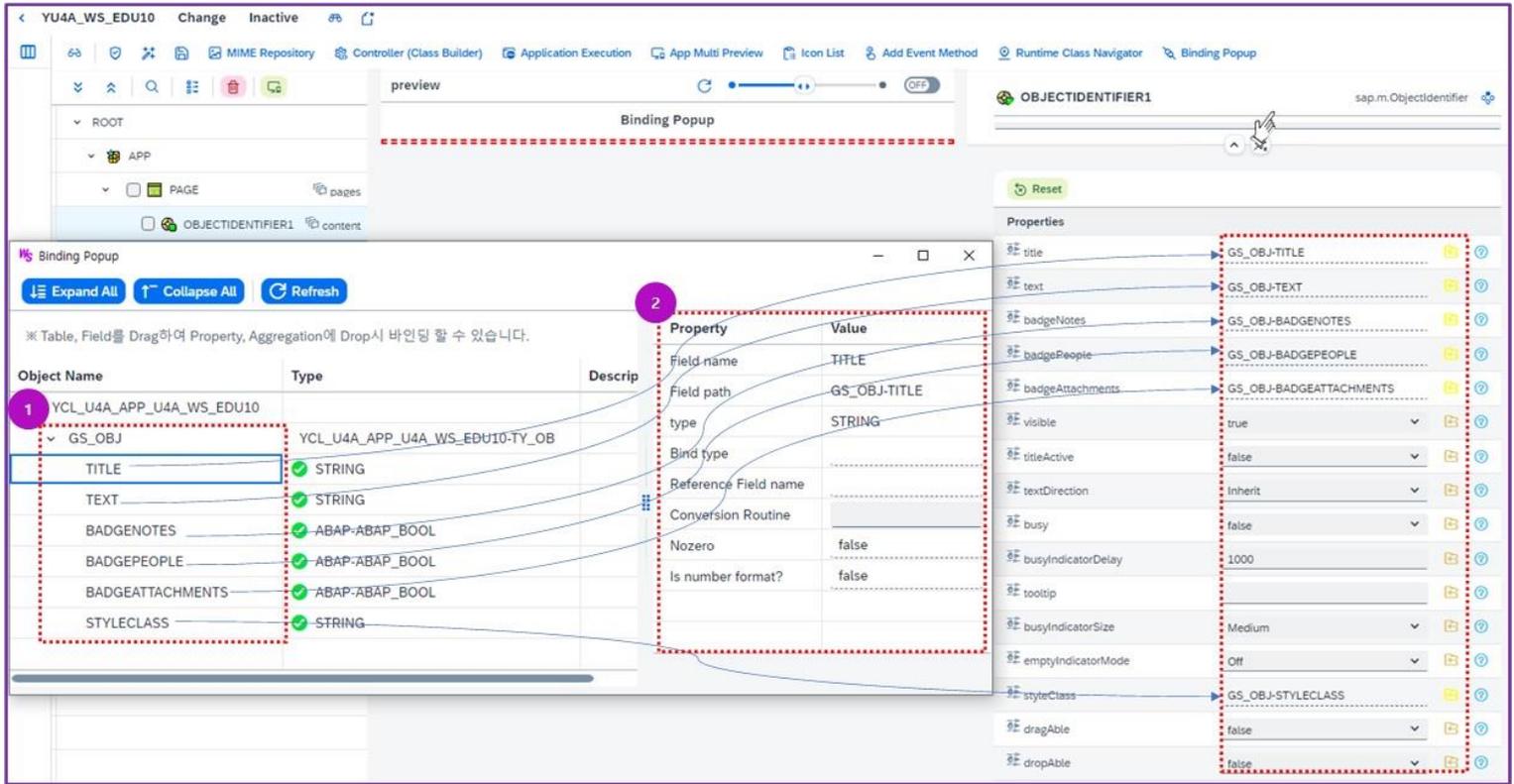


5.22 Binding Popup 기능

※ 본 단원에서는 "OBJECTIDENTIFIER" UI의 "Properties"영역에만 Binding 한다.



2. Binding Pop-Up에 대한 설명.



(1) Binding 가능한 Table, Field를 나타내며 Binding 하고자 하는 "Properties"에 Drag & Drop하여 Binding 한다.

(2) Help Document -> U4A IDE Workbench 사용 방법 -> Binding Options 영역을 참고한다.

3. Application을 실행하여 Binding이 정확하게 되었는지 확인.

```

Method /U4A/IF_SERVER~HANDL_ON_INIT active
1 METHOD /U4A/IF_SERVER~HANDL_ON_INIT.
2
3 "Binding Data 설정.
4 GS_OBJ-TITLE = 'BINDING POPUP TEST'.
5 GS_OBJ-TEXT = 'U4A WorkSpace'.
6 GS_OBJ-BADGENOTES = ABAP_TRUE.
7 GS_OBJ-BADGEPEOPLE = ABAP_TRUE.
8 GS_OBJ-BADGEATTACHMENTS = ABAP_TRUE.
9 GS_OBJ-STYLECLASS = 'sapUiLargeMargin'.
10
11 ENDMETHOD.
    
```

Binding Popup

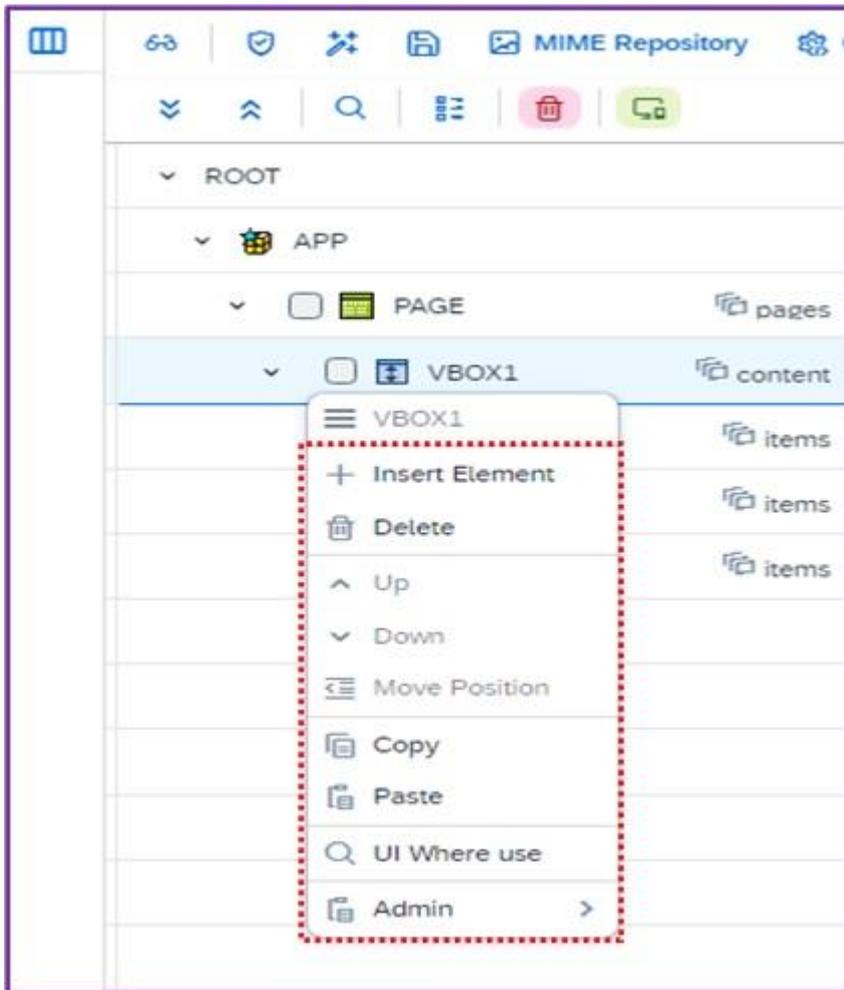
BINDING POPUP TEST   

U4A WorkSpace

5.23 UI Obj의 생성, 삭제, 위치 변경

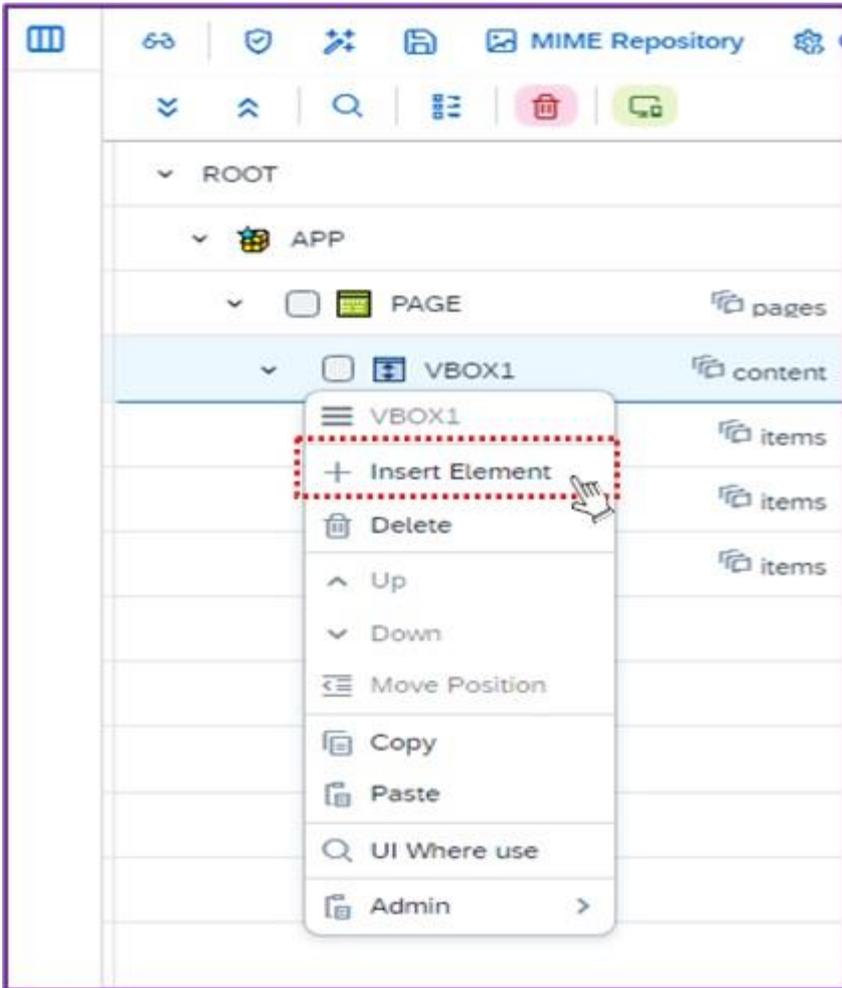
■ 목적 : UI Object를 오른쪽 마우스로 Click시 나타나며 하위 UI Object 생성, 삭제, 위치 변경, 사용처, 복사 및 붙여넣기 등을 수행 할 때 사용한다.

1. 자신이 지정한 Object를 오른쪽 마우스 Click하면 여러 기능이 있는 항목이 나온다.



1) Insert Elements

UI Object의 Aggregations에 원하는 하위 UI Obj를 생성하는 것이다.



Option Property	Means
Aggregations	UI Object 생성 시 상위 UI Object의 어느 Embed Aggregation에 해당시킬 것인지 지정하는 곳
Generated cnt	동일한 Aggregation에 다수의 UI Object 생성 시 input 필드에 적힌 개수 만큼 생성 (특정 Aggregations은 항목이 1개밖에 생성 될 수 없기때문에 나타나지 않을 수 있다.)
UI List	생성 가능한 UI Object를 나타냄

UI Object Select - VBox ✕

Aggregation Name: items
 Generated Cnt: 1 Max : 100
 UI Object: Text

Img	UI Object	UI Object(Fullname)	UI Key	UI Info
<input checked="" type="checkbox"/>	ComboBoxTextField	sap/m/ComboBoxTextField	U000258	?
	ExpandableText	sap/m/ExpandableText	U002453	?
	FormattedText	sap/m/FormattedText	U000289	?
	Text	sap/m/Text	U000452	?
	TextArea	sap/m/TextArea	U000453	?
	InvisibleText	sap/ui/core/InvisibleText	U000883	?
	RichTextEditor	sap/ui/richtexteditor/RichTextEditor	U001786	?

☰
68
🛡️
🔧
💾
📁 MIME Repository
🏠 Controller (Class Builder)
🏃 Application Execution
🖥️ App Multi Preview

▼ ROOT
 ▼ 📁 APP
 ▼ 📁 PAGE 📄 pages
 ▼ 📁 VBox1 📄 content
 📄 TEXT1 📄 items
 📄 BUTTON1 📄 items
 📄 ICON1 📄 items
 📄 TEXTAREA1 📄 items

preview

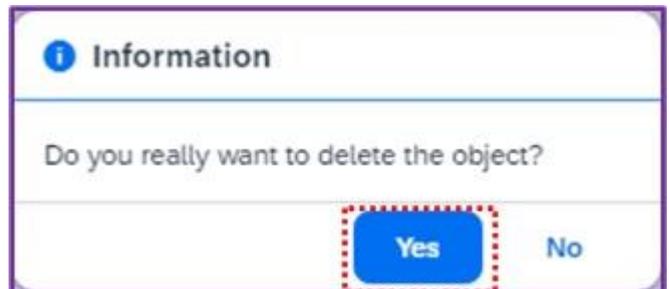
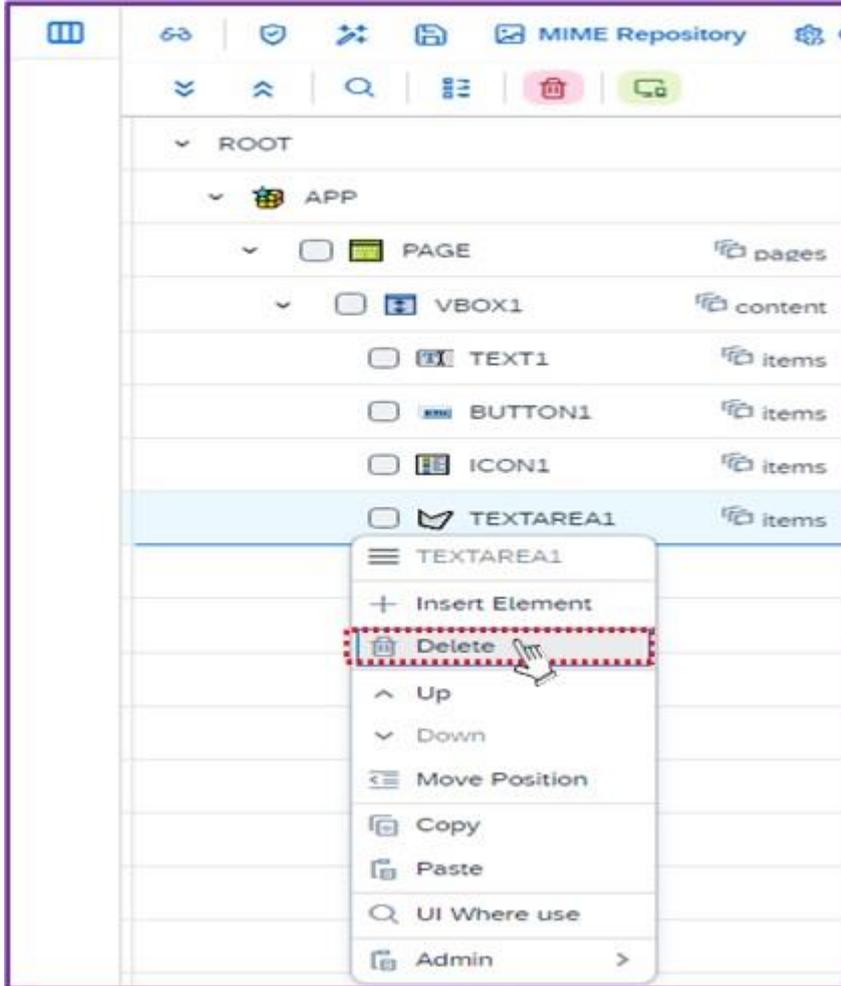
U4A WorkSpace

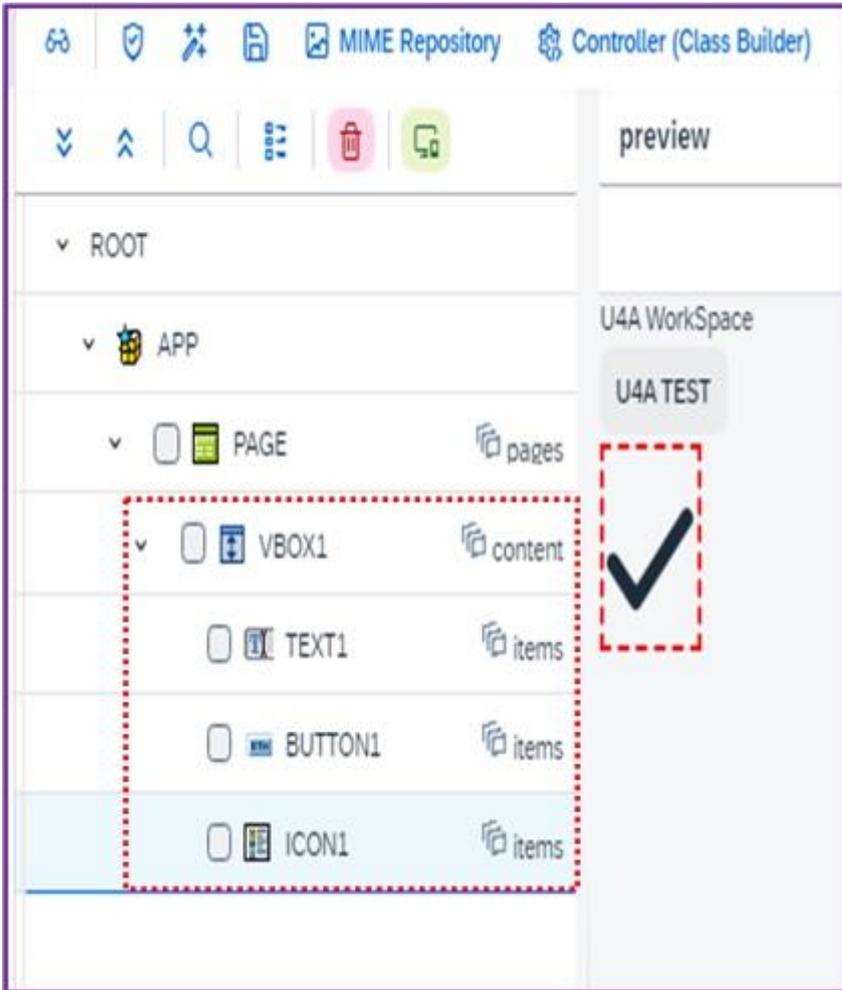
U4A TEST

✔

2) Delete

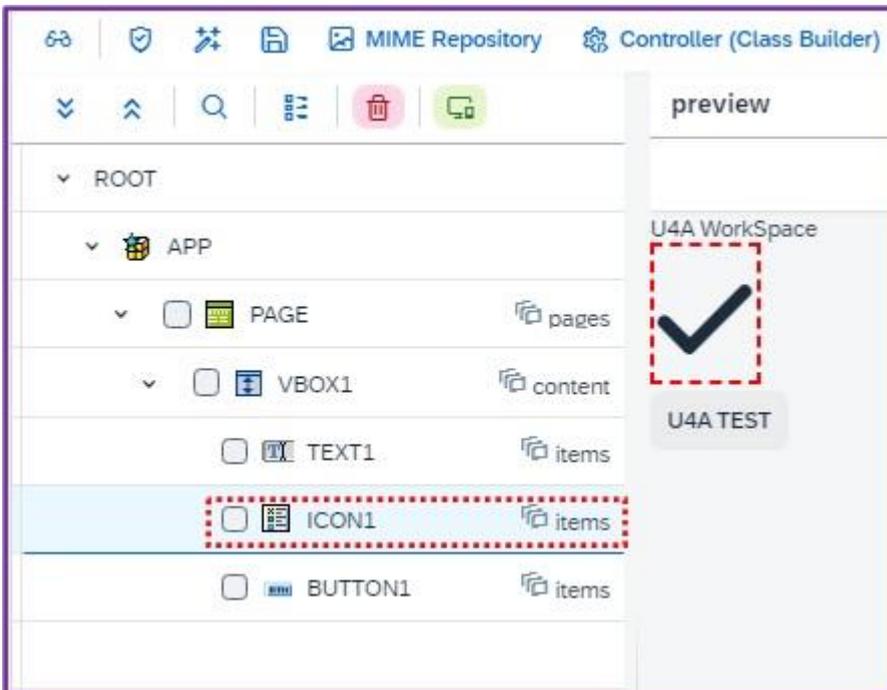
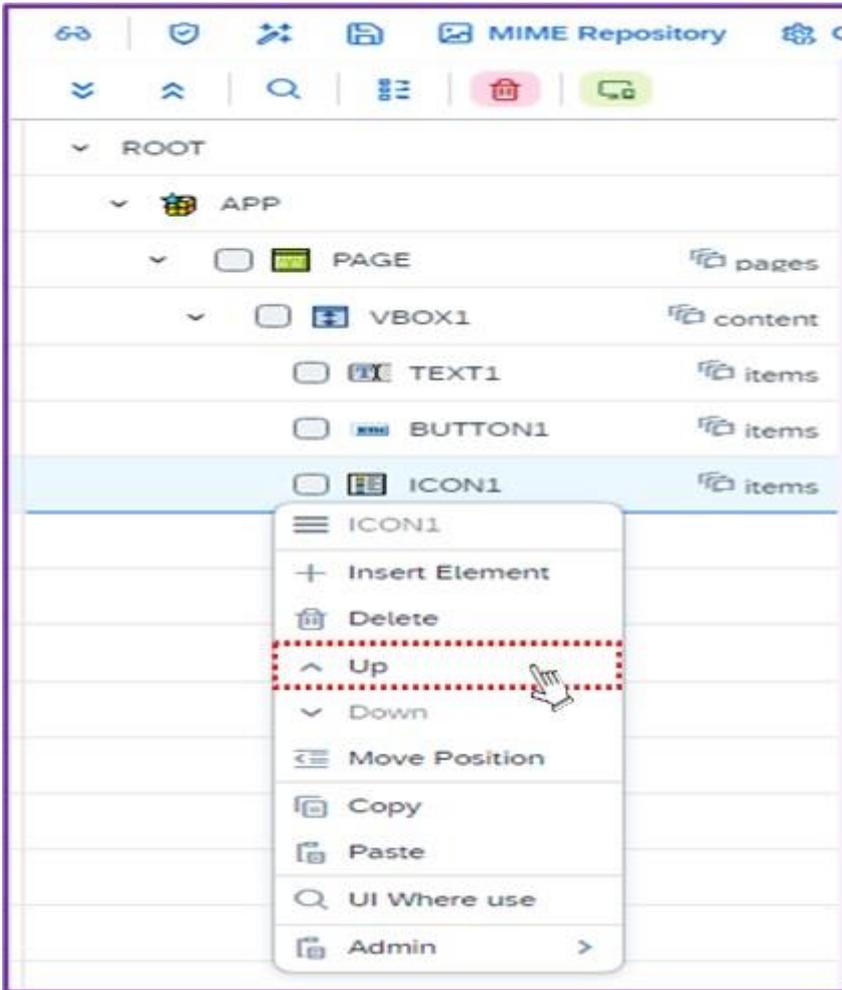
특정 UI Object를 삭제할 때 사용한다.



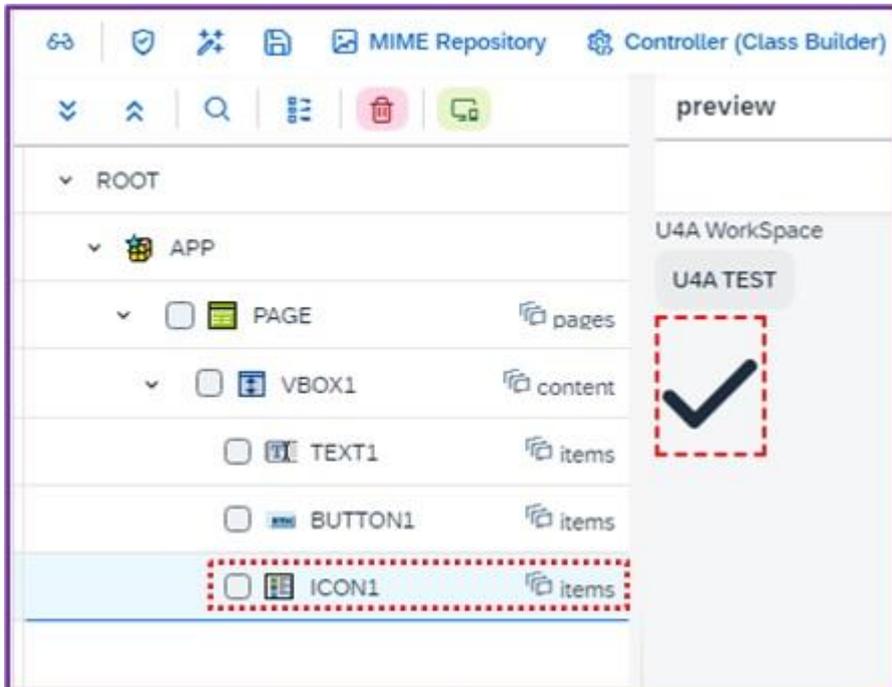
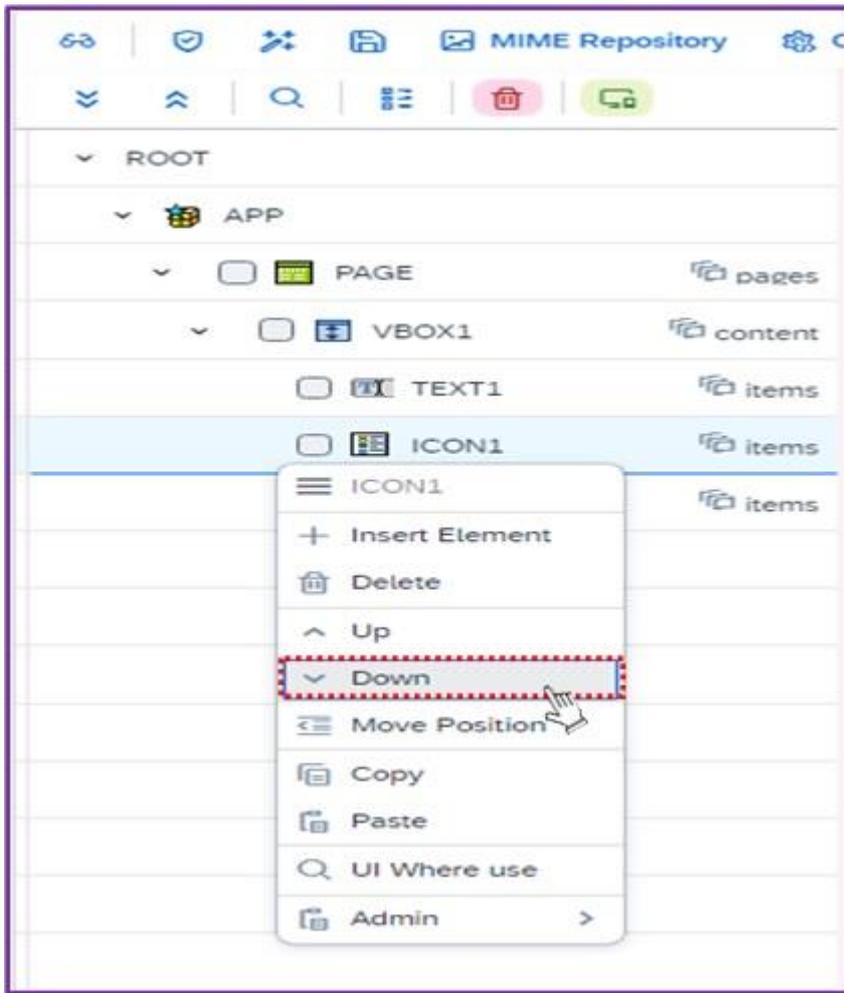


3) Up/Down/Move Position

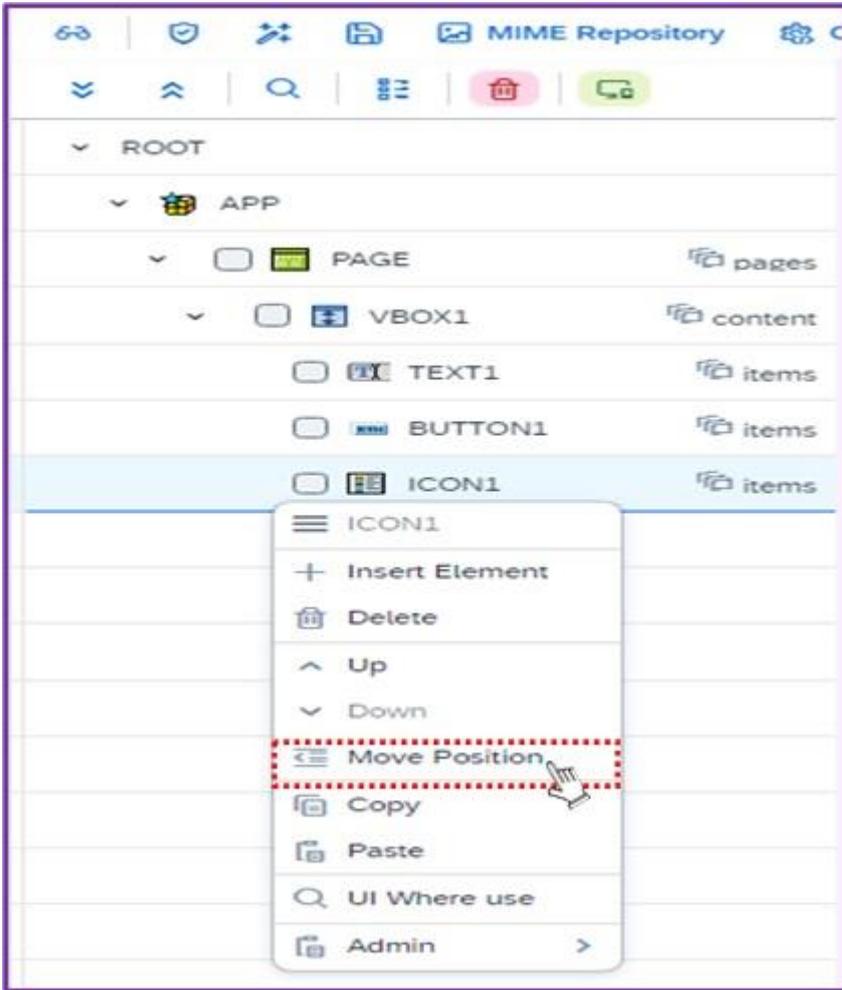
Up : UI Object의 위치를 1칸 올릴때 사용.

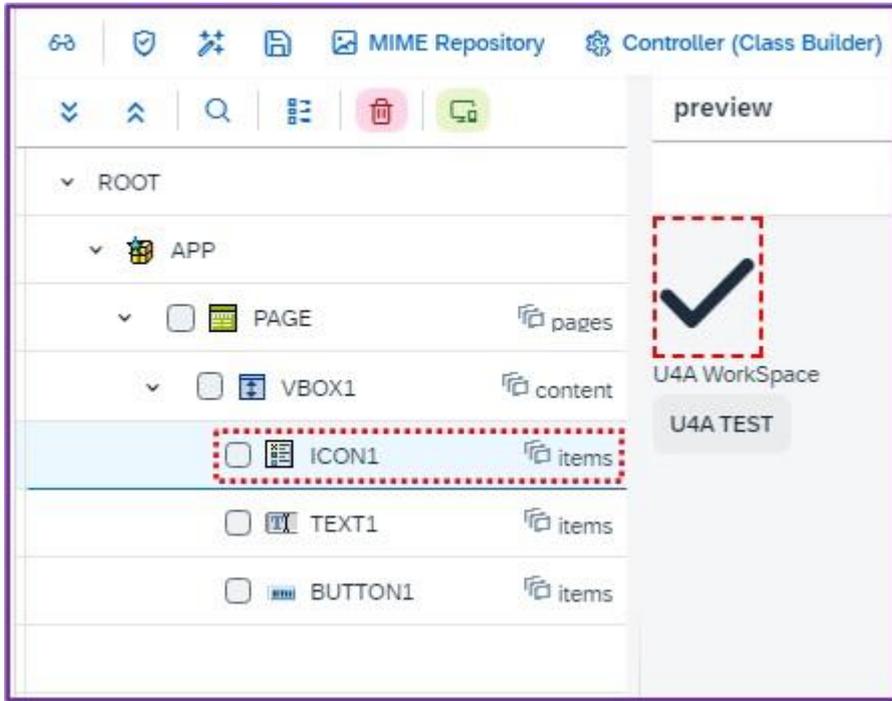


Down : UI Object의 위치를 1칸 내릴때 사용.



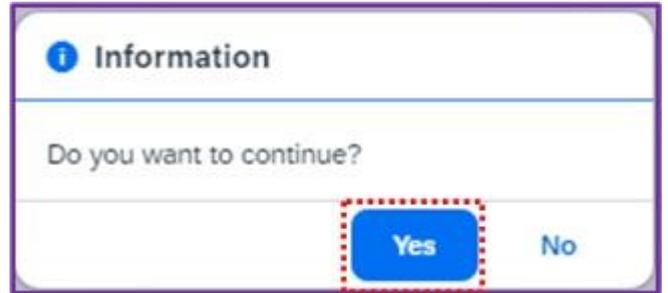
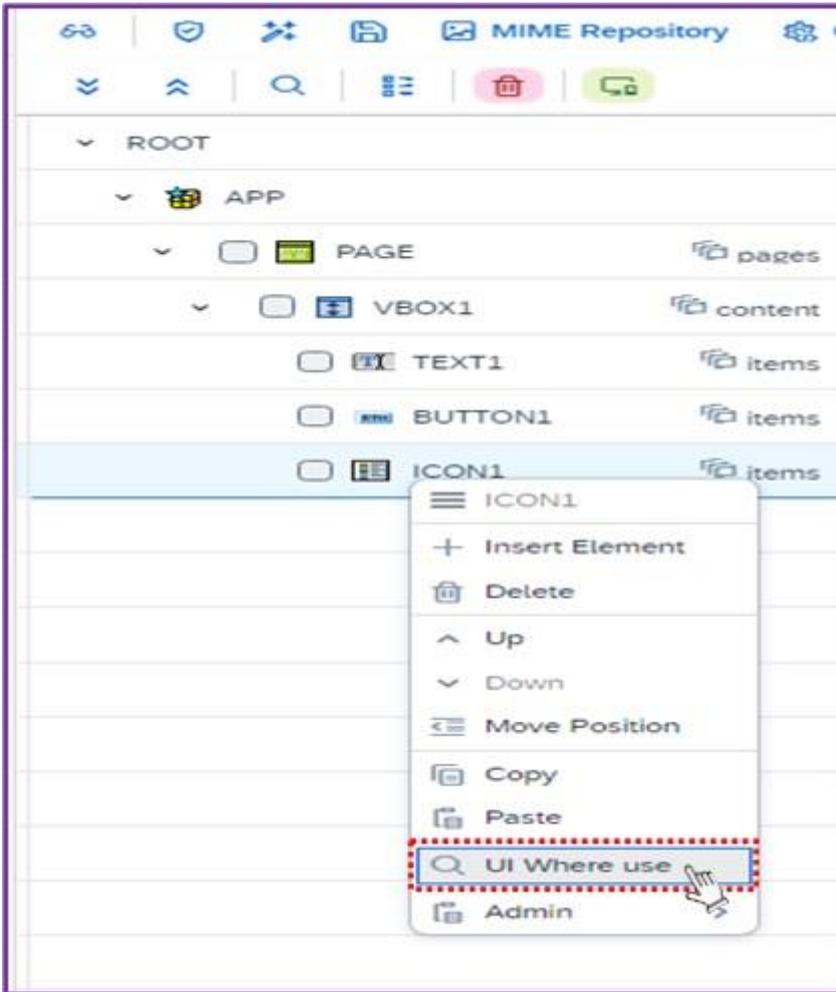
Move Position : UI Object의 위치를 원하는 곳에 배치할 때 사용.(Drag & Drop를 통해서 위치를 변경할 수 있다.)





4) UI Where use

현재 UI Object의 Class를 기준으로 어디 Application에서 사용되었는지 보여줄 때 사용.

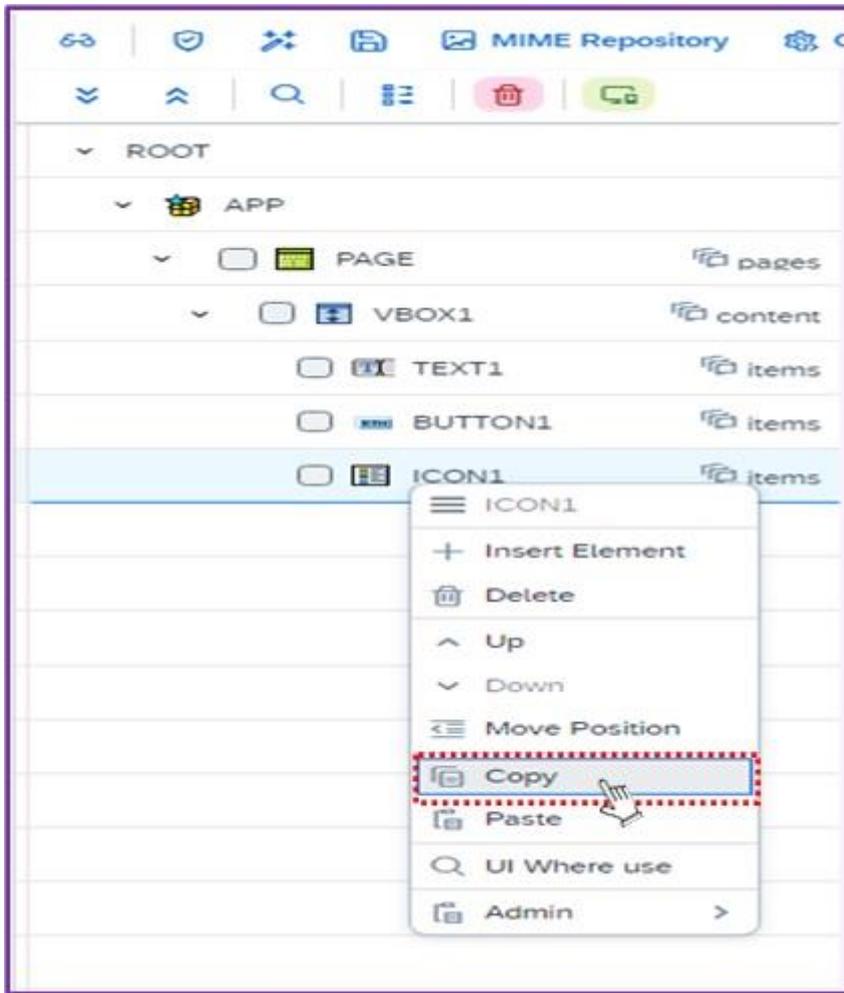


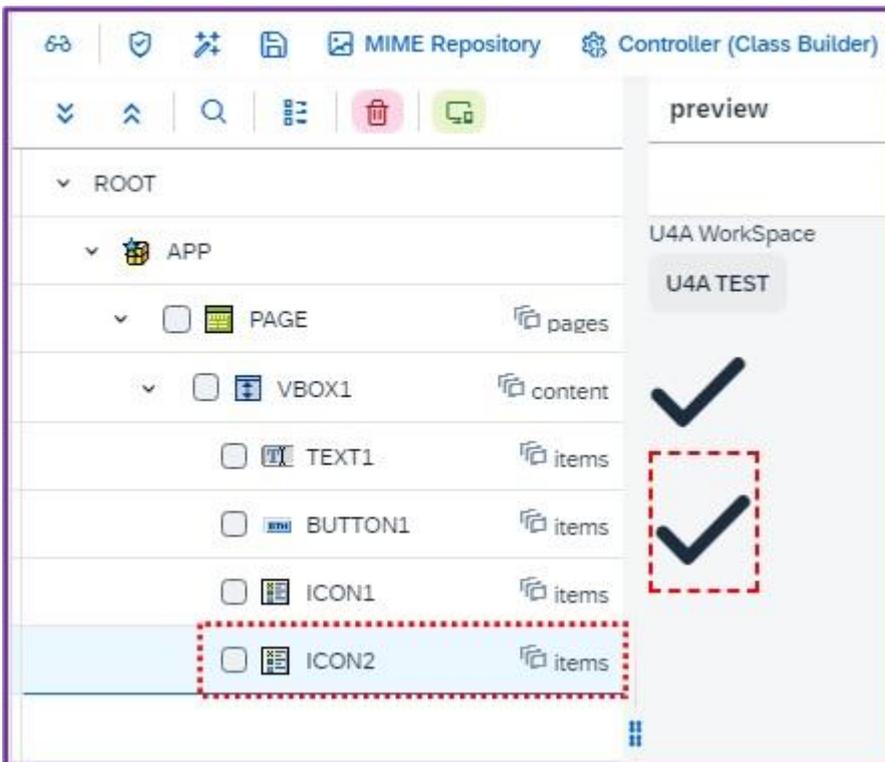
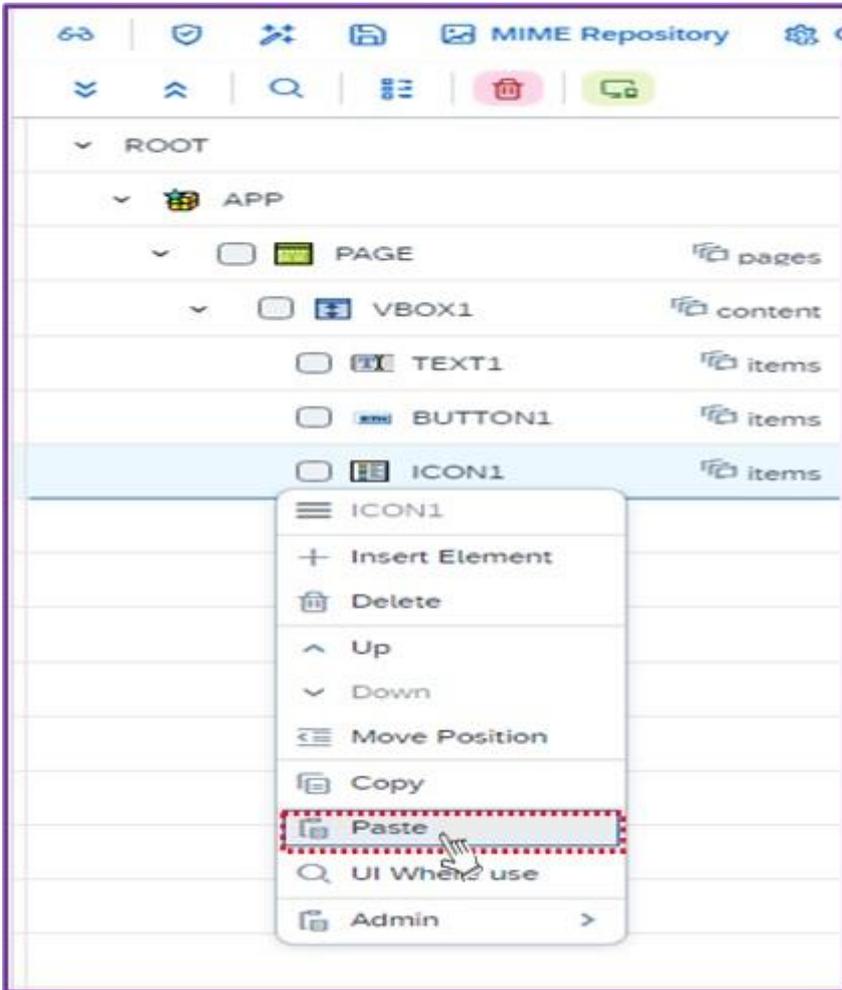
Web Application ID	Web Application Name	Assigned Class Object ID
YHSE_DAUM	DAUM	YCL_U4A_APP_HSE_DAUM
YHSE_PS3	PS 과제 1(모든것 무시 선택트)	YCL_U4A_APP_HSE_PS3
YJIY_TEST113	요청별 단계 3번째	YCL_U4A_APP_JIY_TEST113
YJNH_TEST013	유튜브 모바일 화면 그리기 연습	YCL_U4A_APP_JNH_TEST013
YJNH_TEST014	버스 정보 앱 화면 연습	YCL_U4A_APP_JNH_TEST014
YJNH_TEST015	NAVER PC 화면 그리기 연습	YCL_U4A_APP_JNH_TEST015
YJNH_TEST016	NAVER PC 화면 그리기 연습(개선)	YCL_U4A_APP_JNH_TEST016
YKHNP_ERP_PTAL	test	YCL_U4A_APP_KHNP_ERP_PTAL
YLHR_MAIN_01	YLHR_MAIN_01	YCL_U4A_APP_LHR_MAIN_01
YMAIN_SP01	[신세계] Main Home	YCL_U4A_APP_MAIN_SP01
YMD1I0020	initial screen	YCL_U4A_APP_MD1I0020
YMIGO_SP01	MIGO 샘플	YCL_U4A_APP_MIGO_SP01
YMIGO_SP02	MIGO 샘플	YCL_U4A_APP_MIGO_SP02
YMIGO_SP03	MIGO 샘플	YCL_U4A_APP_MIGO_SP03
YMIGO_SP04	MIGO 샘플	YCL_U4A_APP_MIGO_SP04

5) Copy / Paste

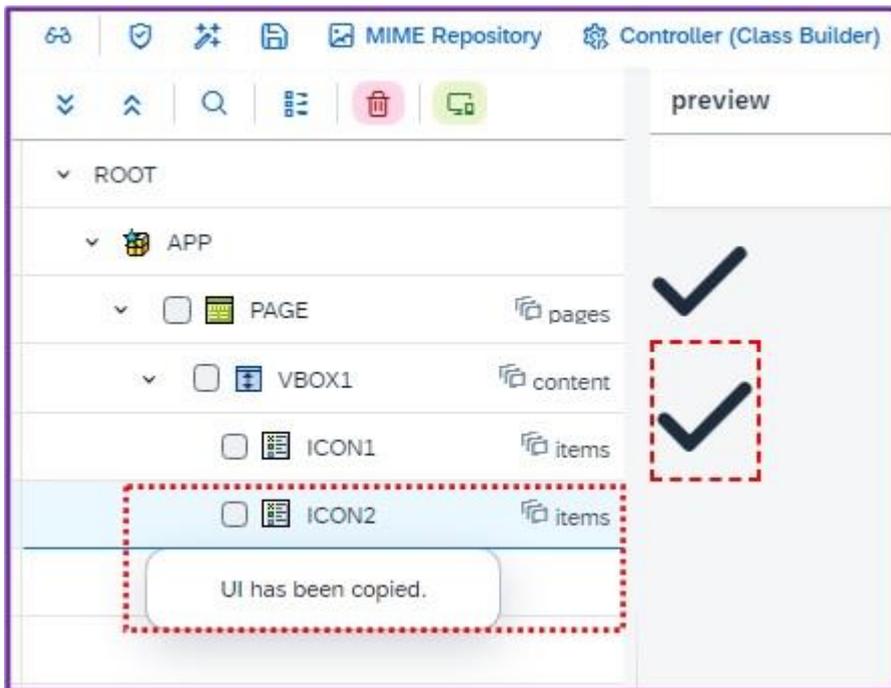
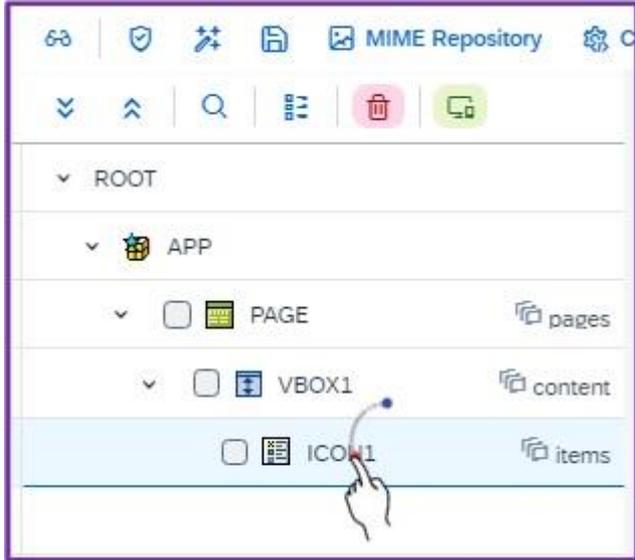
UI Object를 복사하고 붙여넣기 할 때 사용된다. 붙여넣기 시에는 UI Obj의 하위항목 및 Binding 또한 복사되니 주의해서 사용하도록 한다.

1) Copy, Paste 버튼 클릭을 통한 복사,붙여넣기





2) UI Drag & Drop을 통한 복사,붙여넣기

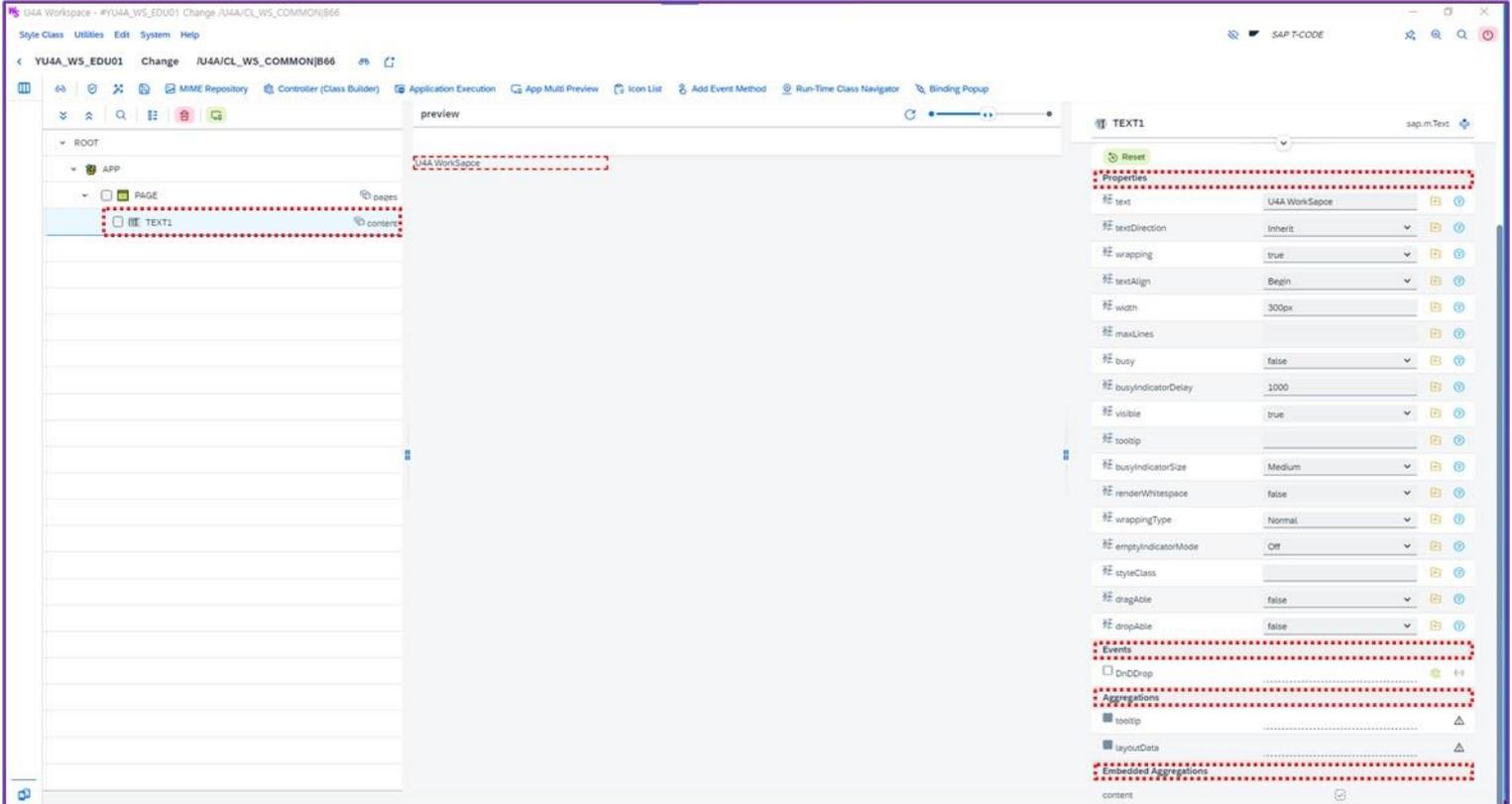


i Copy후 다른 위치에 paste를 진행 할 시 UI Object의 id를 다시 채번해서 만들기 때문에 복사 후 UI Object id와 관련된 Logic을 확인 하도록 한다.

5.24 UI Obj의 Attributes 구성 요소

■ **목적** : UI Object 형태, Event, Binding과 같은 요소들을 변경 및 생성이 가능하다.

1. UI Object를 생성 후 UI Object를 구성하고 있는 요소가 있다.



Option Property	Means
Properties	주로 UI Object의 디자인, value값, 위치와 같은 모양이나 형태를 결정하는데 사용된다.
Events	주로 UI Object의 Action이 발생했을 때 설정한 서버 event를 호출 시 사용한다.
Aggregations	UI를 추가할 수 있는 영역을 의미한다.
Embedded Aggregations	현재 UI Object가 부모의 어느 영역의 위치하는지 나타낸다.

2. UI Object의 값을 변경 후 초기값으로 되돌리고 싶으면 Attributes 상단의 "Reset" 버튼을 Click한다.

The screenshot shows the U4A IDE interface. On the left is a preview window titled 'preview' containing a text object labeled 'U4A WorkSpace'. On the right is the 'TEXT1' properties panel, which includes a 'Reset' button and a list of properties. The 'Reset' button and the 'width' property are highlighted with red dashed boxes.

Property	Value
text	U4A WorkSpace
textDirection	Inherit
wrapping	true
textAlign	Begin
width	300px
maxLines	
busy	false
busyIndicatorDelay	1000
visible	true
tooltip	
busyIndicatorSize	Medium
renderWhitespace	false
wrappingType	Normal
emptyIndicatorMode	Off
styleClass	
dragAble	false
dropAble	false

The screenshot shows an information dialog box with the following text:

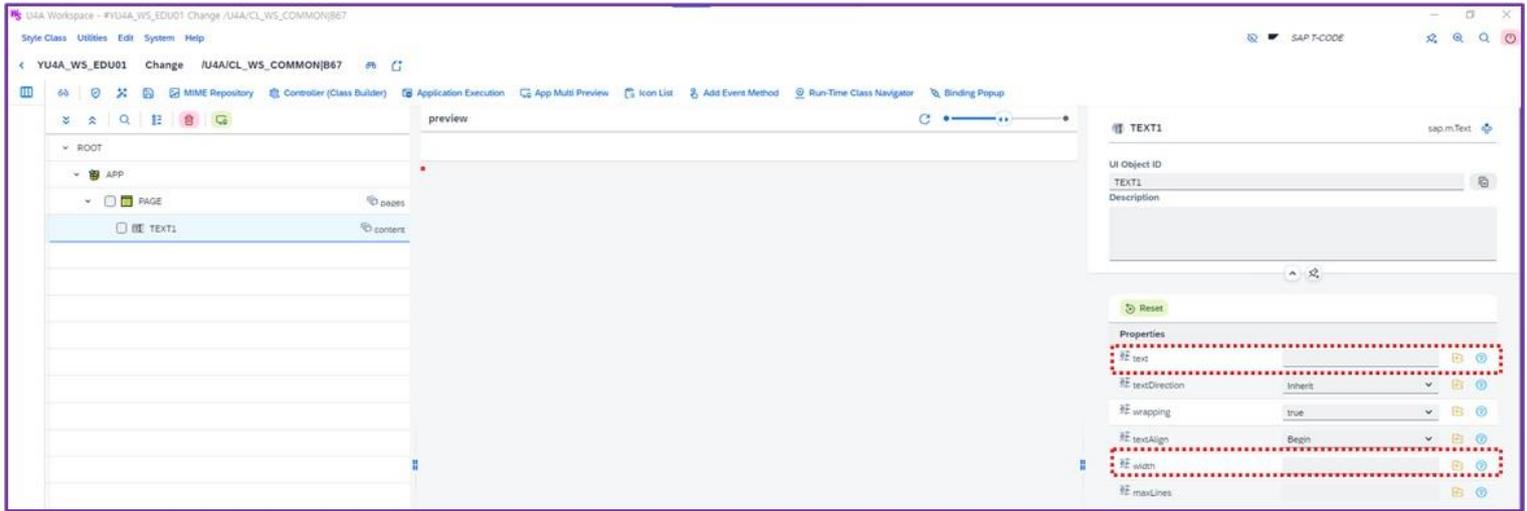
Information

Resets all properties to their default values.
The registered value will be lost. Do you want to proceed?

Yes No

The 'Yes' button is highlighted with a red dashed box.

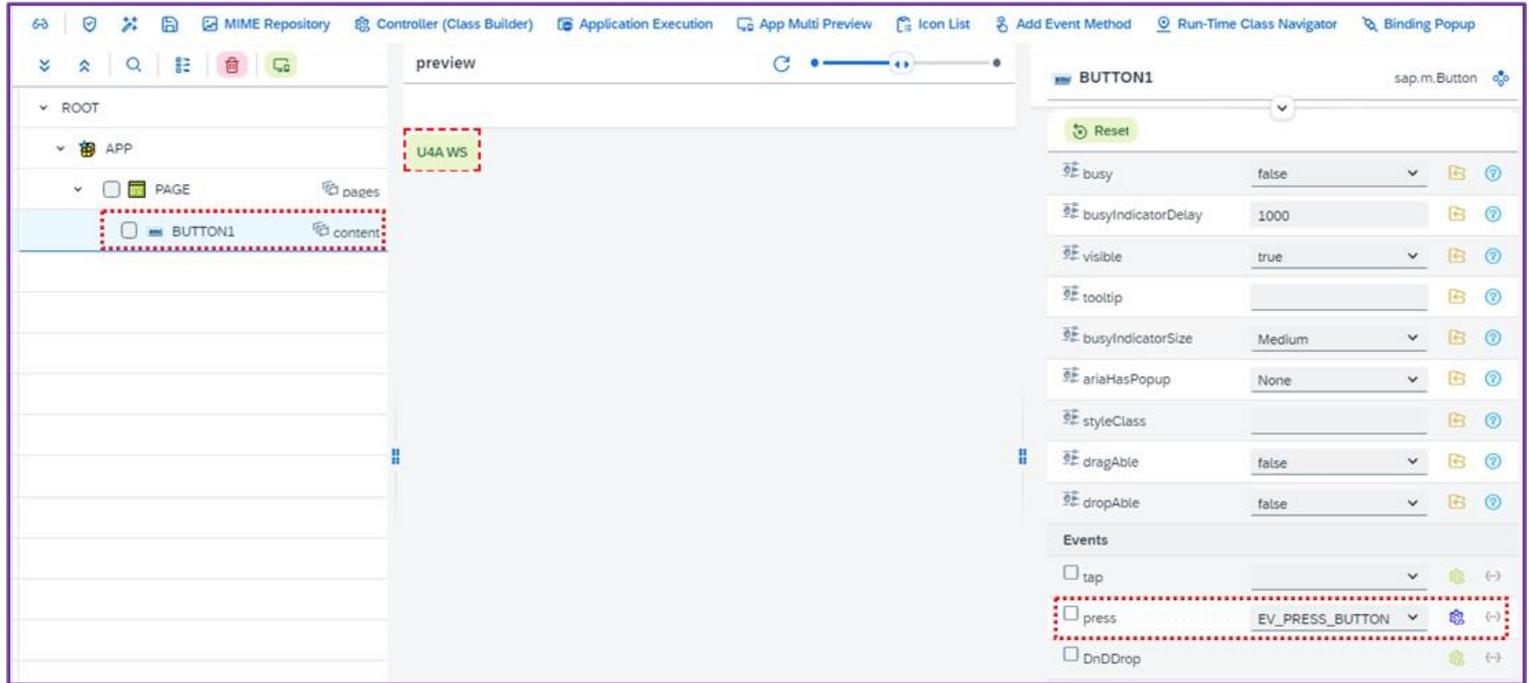
U4A IDE HELP



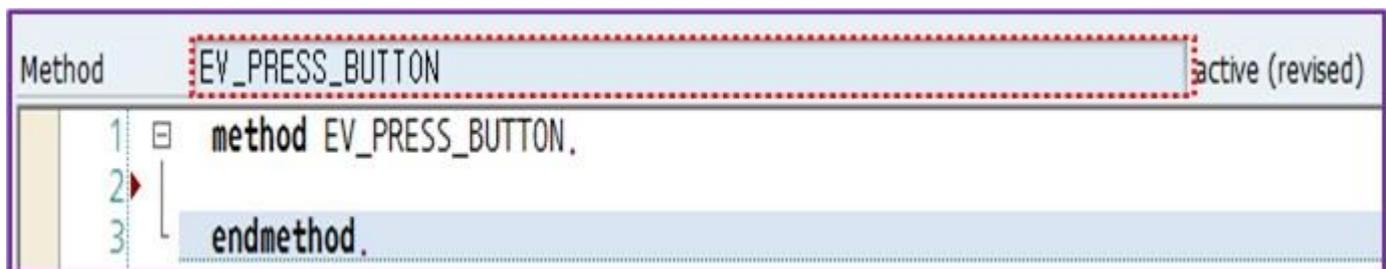
5.25 Event method 기초

■ 목적 : UI Object 에서 Event 가 발생 했을때 Web 에서 어떤 Data 를 전달 받는지 확인 하는 방법을 설명한다.

1. UI Object의 Events를 생성한다.



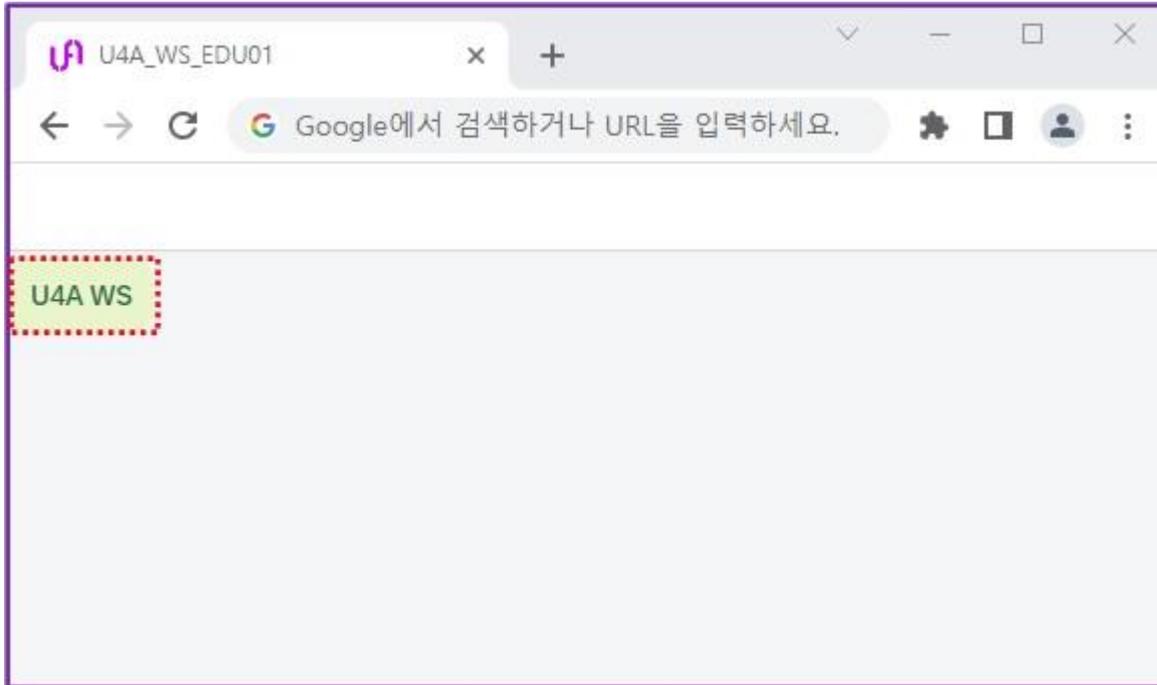
2. 생성된 Methode의 위치로 이동한다.



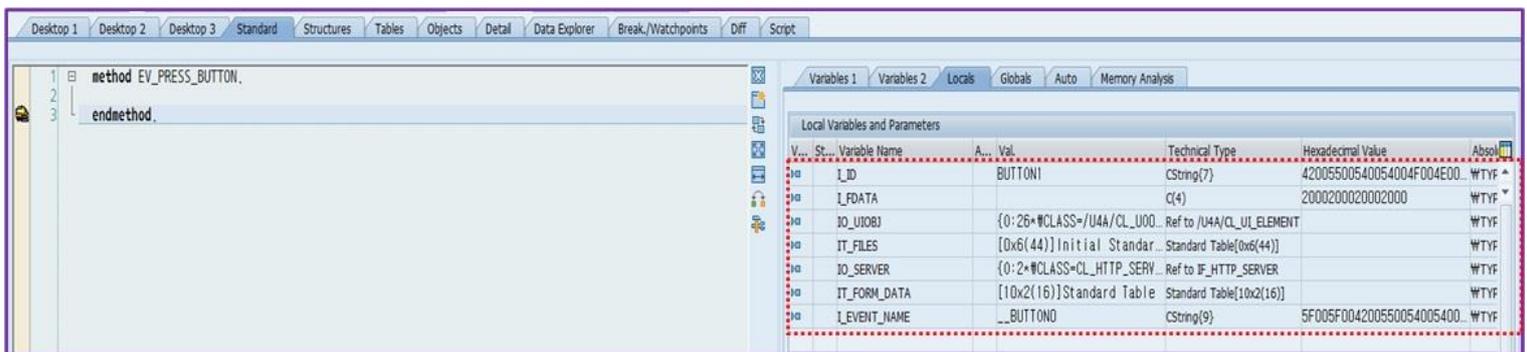
3. Methode에 Debug를 걸고 Web에서 동작시킨다(Debug는 반드시 Set/Delete External Breakpoint 를 걸도록 한다)

```

Method EV_PRESS_BUTTON active
1 method EV_PRESS_BUTTON,
2
3 endmethod.
    
```



4. Debug 화면에서 Locals 영역으로 들어가면 Web에서 어떤 Data가 전달되는지 확인 할 수 있다.



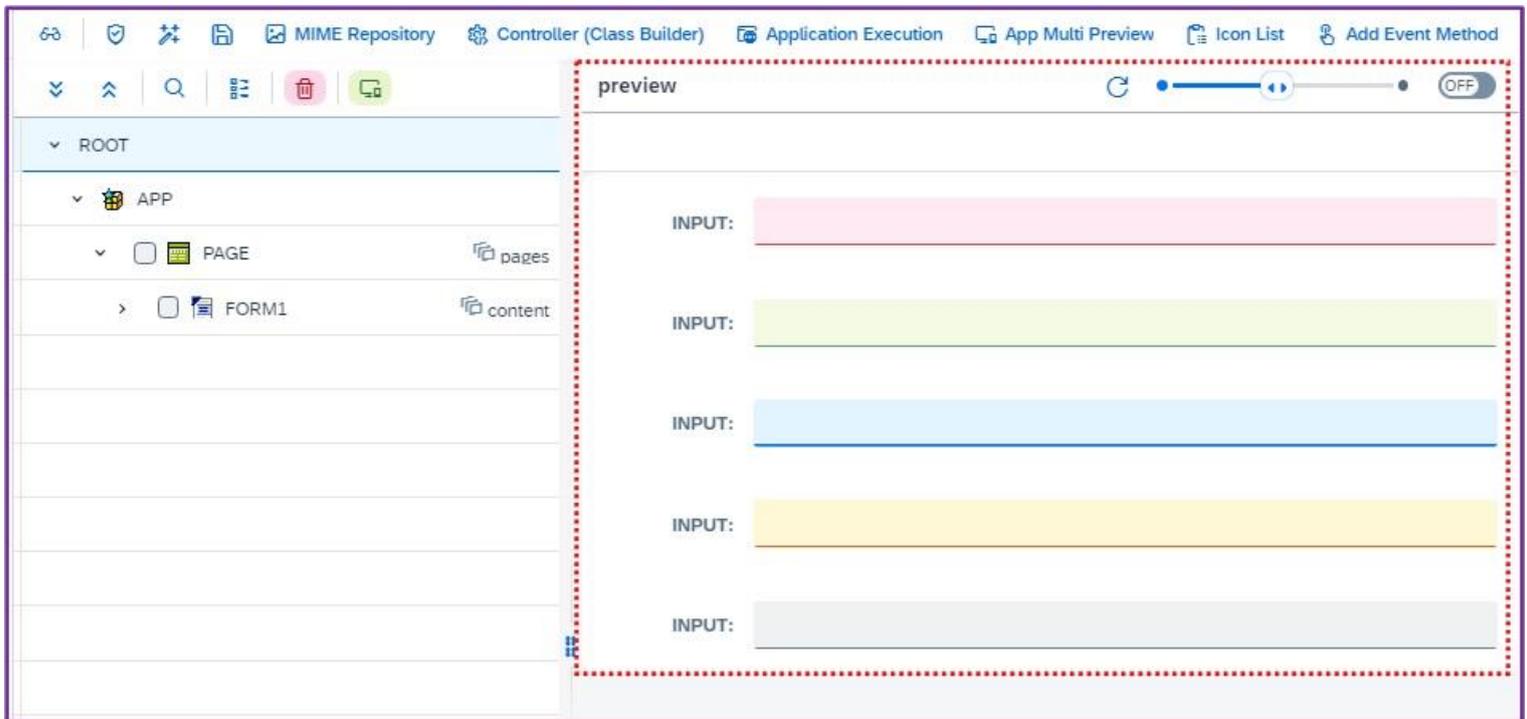
Option Property	Means
_I_ID	Events 작동시 UI Object의 ID 정보를 나타냄
_I_FDATA	F4 Help call back Event return Parameter 용 으로 사용된다

IO_UIOBJ	선택한 UI Object의 class instance
IT_FILES	File Upload UI 전용 Parameter
IT_FORM_DATA	UI Event의 client Parameter에 사용된 정보
I_EVENT_NAME	Events 작동시 UI Object의 Index 정보를 나타냄

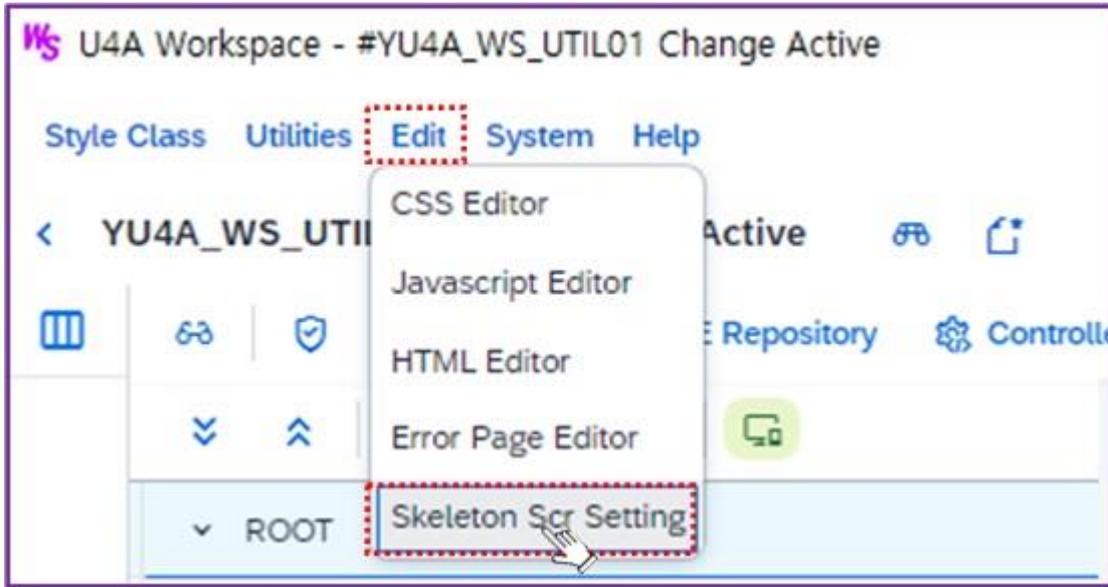
5.26 Skeleton Screen

■ 목적 : 네트환경이 좋지 않은 경우, Web 화면에서 실행 시 UI화면 나타나기전 미리보기용 화면 설정 ("Preview" 화면)

1. Web화면에 보여질 UI 구성.



2. Preview 화면 "Skeleton Screen" 세팅.



3. "Skeleton Screen"을 구성하고 확인버튼 클릭.



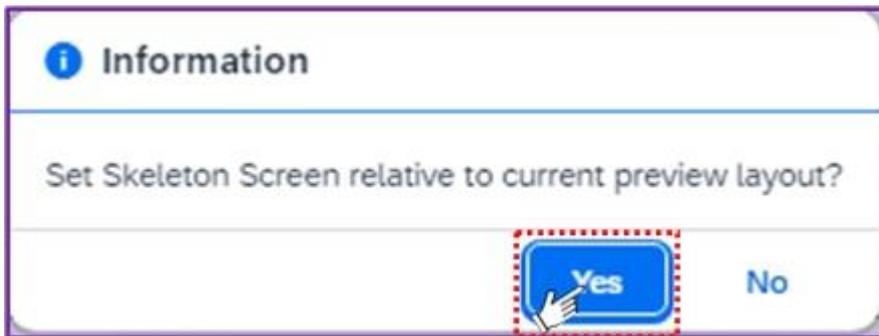
1) **Waiting mode effect** : 미리보기용 화면(= Skeleton Screen)이 출력된 뒤 Application 화면이 호출 되기전까지 미리보기 화면이라는 것을 나타내기 위한 효과 < /FONT>

2) **Use of glass** : Skeleton Screen의 blurry(= 흐릿한) 효과 사용여부 체크

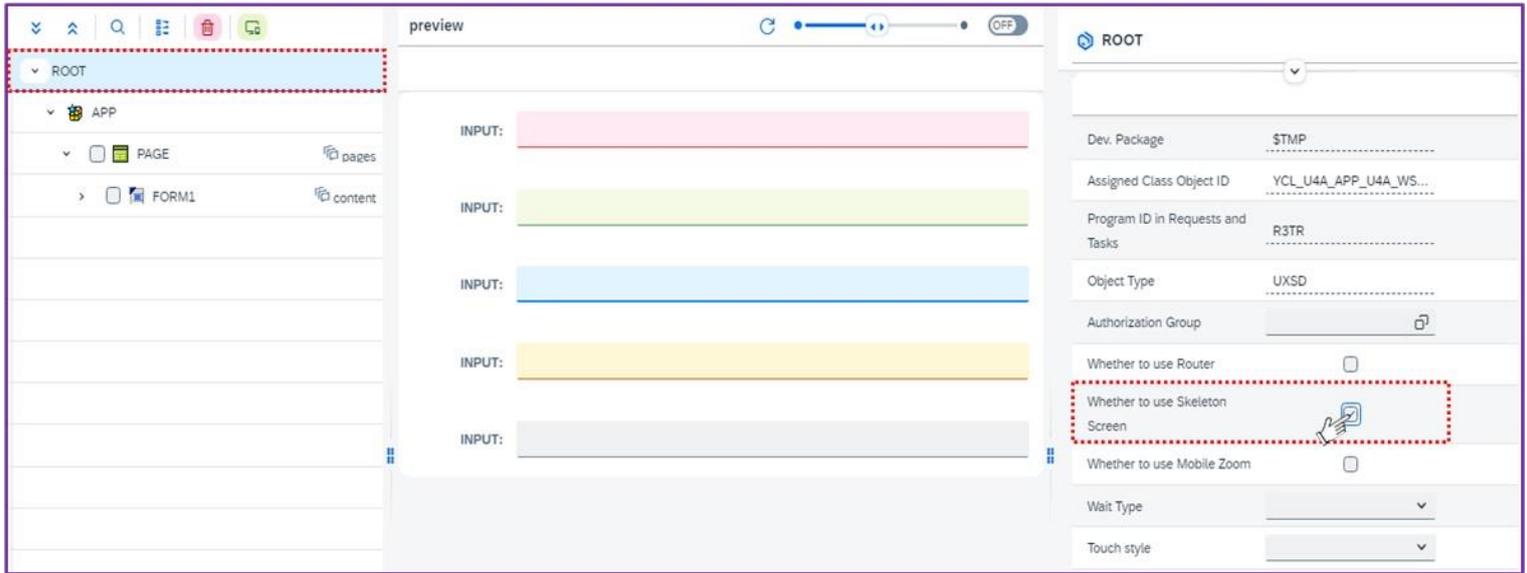
3) **Glass concentration** : Skeleton Screen의 blurry 농도 설정(0 ->100으로 갈수록 더욱 흐릿해진다, "Use of glass"가 미체크 되어있다면 적용되지 않음)

4. "Skeleton Screen"을 구성확인 Pop-Up "Yes' 클릭.

※ 현재 Application의 "Preview" 영역의 Layout을 Skeleton Screen으로 구성한다

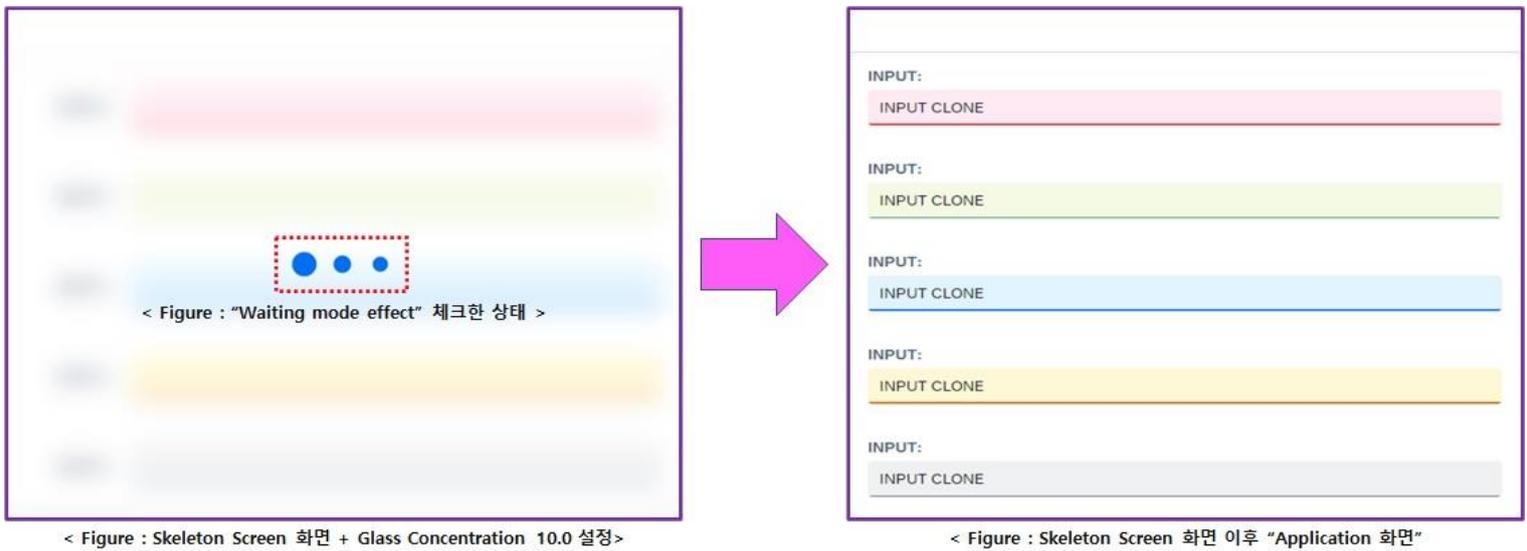


5. 위에서 구성한 Skeleton Screen을 사용하려면 아래와 같이 설정해야한다.

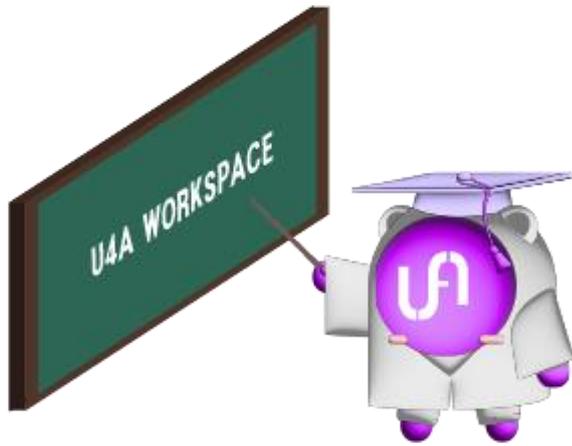


6. Application 실행하여 Skeleton Screen이 적용되었는지 확인.

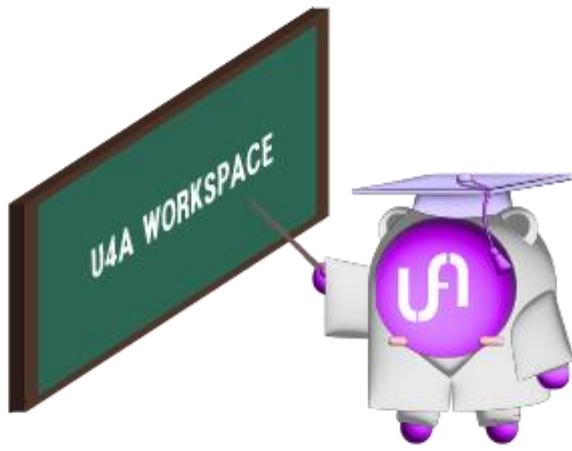
※ Application을 실행하면 Skeleton Screen 출력 후 해당 Application 화면 출력



6. APP 개발 하기 Part 2



6.1 ABAP Class /U4A/CL_UTILITIES



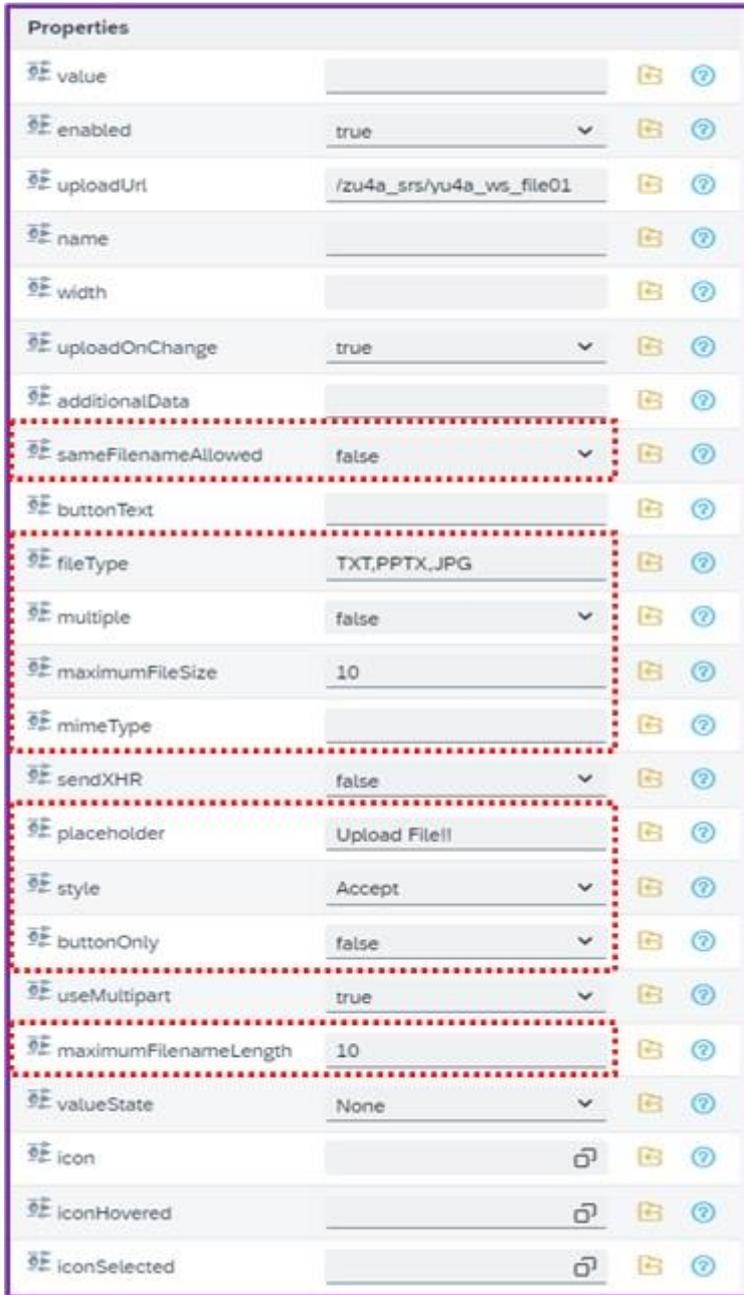
6.1.1 Methodes FILE_UPLOAD

■ 목적 : Application 외부에서 File을 Upload 하기 위한 목적.

1. File Uploader UI를 추가한 후 Properties에서 uploadOnChange 값을 ture로 설정, "Events uploadComplete" Event 추가.

The screenshot displays the U4A IDE interface. On the left, a tree view shows the project structure under 'ROOT' > 'APP' > 'PAGE' > 'FORM1'. The 'FILEUPLOADER1' component is selected and highlighted with a red dashed box. On the right, the 'Properties' and 'Events' panels are visible. In the 'Properties' panel, the 'uploadOnChange' property is set to 'true' and is also highlighted with a red dashed box. In the 'Events' panel, the 'uploadComplete' event is set to 'EV_FILE_UPLOAD' and is also highlighted with a red dashed box.

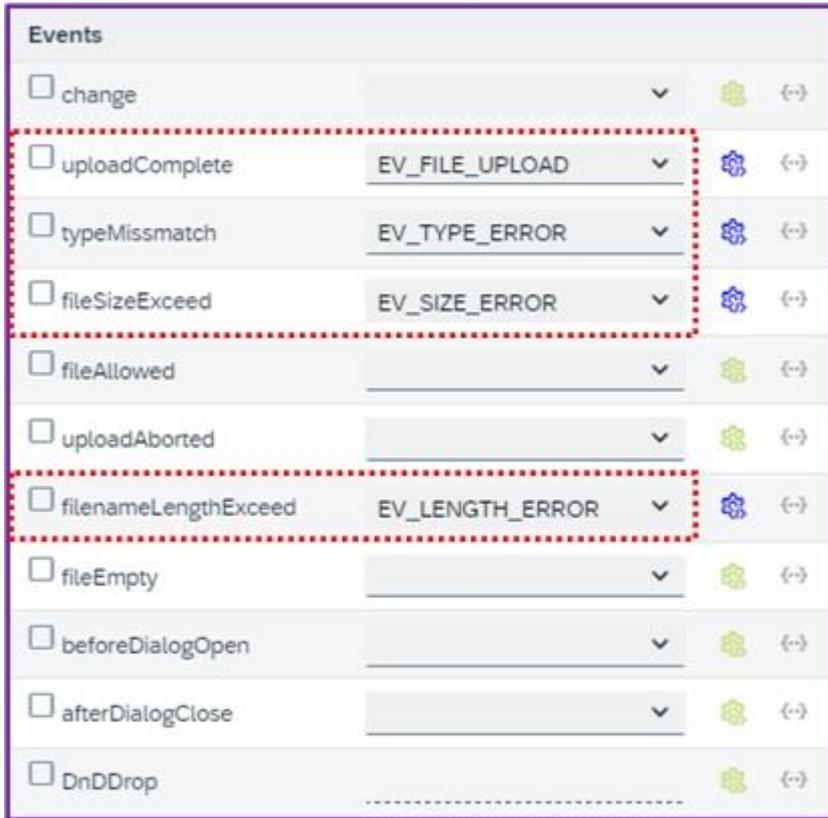
2. File Uploader Property 설명.



NAME	MEAN
sameFilenameAllowed	<p>동일한 File 명을 여러 번 Upload 할 수 있는 속성값이다.</p> <p>해당 값을 false로 설정한 경우 동일한 File을 Upload 할 시 Event method를 호출하지 않는다.</p> <p>true로 설정한 경우 동일 File Upload 할 때마다 Event method를 호출하게 된다.</p>
file Type	<p>입력된 확장자 이외의 File이 Upload 될 때 uploadComplete event method를 호출하지 않는다.</p>

	<p>값이 입력되지 않은 경우 확장자에 상관없이 Upload 할 수 있다.</p> <p>여러개의 확장자를 설정할 경우 콤마(,)를 통해 구분 지을 수 있으며, 콤마 사이의 공백은 허용하지 않는다.</p>
multiple	<p>해당 속성값을 true로 설정하게 될 경우 여러 개의 File을 한번에 Upload 할 수 있다.</p>
maximumFileSize	<p>Upload될 File의 최대 File 크기를 정할 수 있다.</p> <p>단위는 MB, 여러 개의 File이 Upload 가능한 경우 Upload한 File의 총합크기가 설정된 file크기는 초과할 수 없다. (각각의 File 중 한 개라도 초과할 경우)</p>
mimeType	<p>File 선택창에서는 설정된 File만 Filtering 할 수 있다.</p> <p>File 선택창에서 설정된 File만 Filtering 기능만 제공하는 것이므로 FileType에 Upload 대상 File 확장자를 설정하지 않은 경우 Upload 되지 않는다.</p> <p>EX) fileType = txt, mimeType = image/jpg 인 경우 jpg file Upload 불가.</p>
placeholder	<p>File 입력 input Field 영역에 배경 text를 설정할 수 있다.</p>
style	<p>Button의 Style을 변경할 수 있다.</p>
buttonOnly	<p>FileUploader UI의 Button만 표현할 수 있다.</p>
maximumFilenameLength	<p>Upload File의 제목 길이 제한을 설정할 수 있다.</p>

3. File Uploader Events 설명



NAME	MEAN
uploadComplete	File Upload에 성공 하였을 때 uploadComplete method를 호출하게 된다.
typeMismatch	입력된 확장자 이외의 File이 Upload 될 때 typeMismatch event method를 추가하였다면 해당 method를 호출하게 된다
fileSizeExceed	Upload한 File이 설정한 File Size를 초과할 경우 fileSizeExceed event method를 호출하게 된다.
filenameLengthExceed	Upload한 File이 설정한 File 제목의 길이가 초과할 경우 filenameLengthExceed event method를 호출하게 된다.

4. Application을 실행한 후 FileUploader Click.

Title: U4A WorkSpace File Upload

Writer:

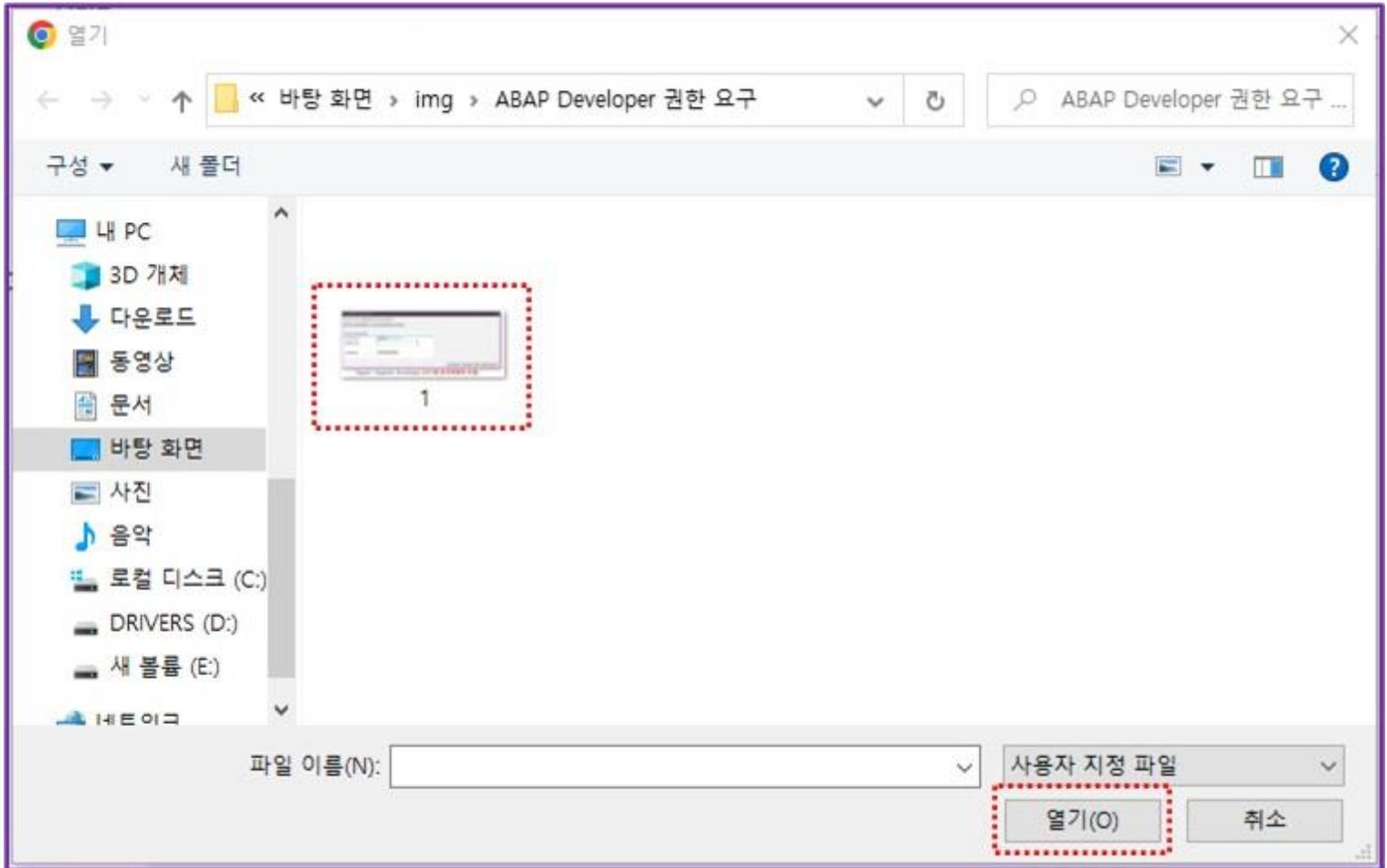
Date:

Time:

Contents: U4A WorkSpace File Upload

Contents: Upload File!! 

5. Upload 할 File 선택 후 열기 Button Click.



6. Upload Complete Method 과정에서 IT_FILES에 선택한 File Data가 삽입 된 것을 확인.

Method **EV_FILE_UPLOAD** active

```

1  METHOD EV_FILE_UPLOAD.
2
3  "Upload File Check.
4  CHECK IT_FILES IS NOT INITIAL.
5
6  "Move Upload File.
7  MOVE-CORRESPONDING IT_FILES TO GT_FILE.
8  CHECK SY-SUBRC EQ 0.
9
10 ENDMETHOD.
    
```

Variables 1 | Variables 2 | **Locals** | Globals | Auto | Memory Analysis

Local Variables and Parameters

V...	St...	Variable Name	A...	Val.	Technical Type
▶		I_ID		FILEUPLOADER1	CString{13}
▶		I_FDATA			C(4)
▶		IO_UIOBJ		{0: 44*#CLASS=/U4A/CL_U001180}	Ref to /U4A/CL_UI_ELEMENT
▶		IT_FILES		[1x6(44)] Standard Table	Standard Table[1x6(44)]
▶		IO_SERVER		{0: 2*#CLASS=CL_HTTP_SERVER_NET}	Ref to IF_HTTP_SERVER
▶		IT_FORM_DATA		[8x2(16)] Standard Table	Standard Table[8x2(16)]
▶		I_EVENT_NAME		__UPLOADER0	CString{11}

7. IT_FILES Fileds 정보 (TYPE : /U4A/Y0006).

Tables / Table Contents

Table: **IT_FILES**

Attributes: Standard [1x6(44)]

Insert Column: Columns ...

Row	FNAME [CString]	FSIZE [i4]	MIME [CString]	EXTEN [CString]	PATH [CString]	CONTENT [XString]
1	i.jpg	11975	image/jpeg	jpg		FFD8FFE001D4A46494600101000010010000FFD80084000707070807080909080C0C080C0C11100E0E10111A12141214121A27181D18181D182732A222022A233E

NAME	MEAN
FNAME	FILE 이름
FSIZE	FILE Size
MIME	FILE Type
EXTEN	FILE 확장자
CONTENT	FILE 내용 정보

8. 선택한 File이 Upload 된 것을 확인 .

Title: *U4A WorkSpace File Upload*

Writer: _____

Date: _____

Time: _____

Contents: *U4A WorkSpace File Upload*

Contents:



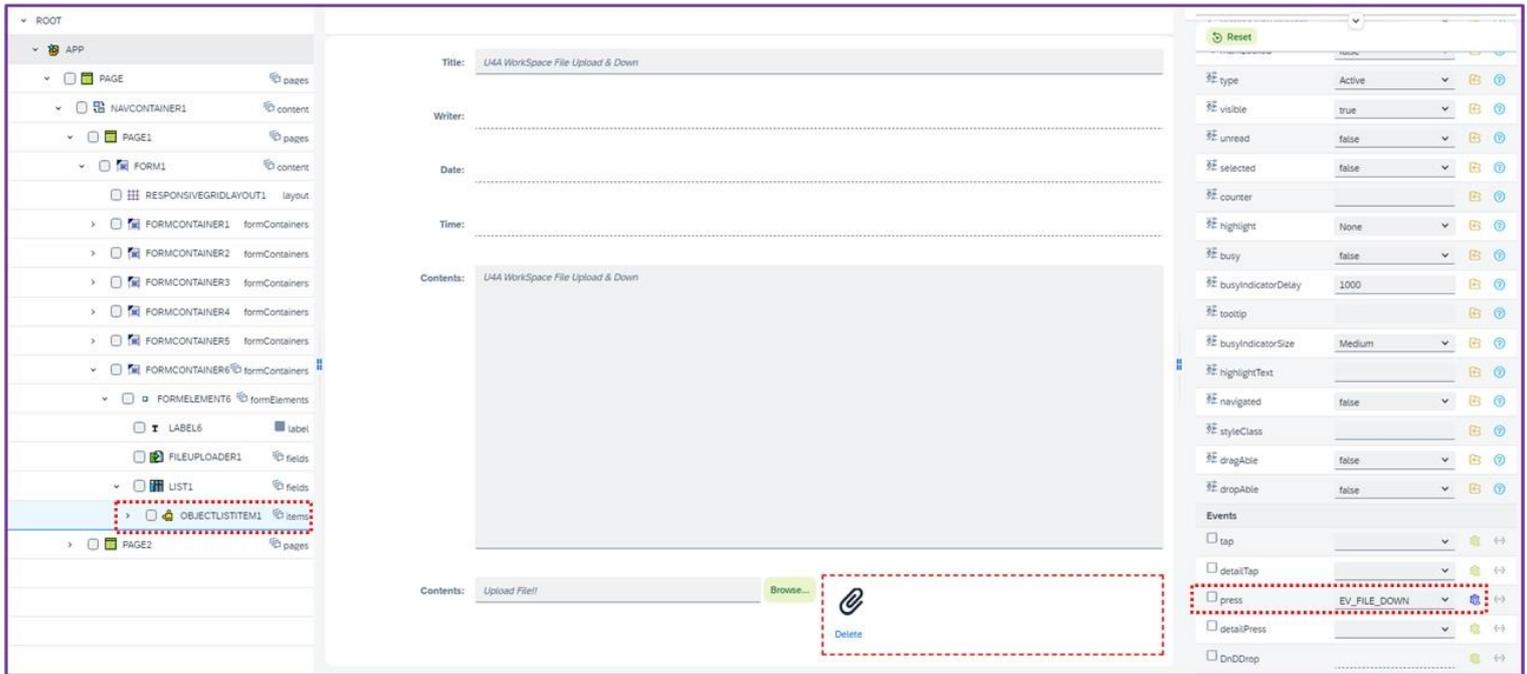
File Upload에 관한 더 상세한 정보를 얻으려면

https://www.u4ainfo.com/bbs/board.php?bo_table=u4aide_data&wr_id=8&sfl=wr_subject&stx=upload&sop=and 참조

6.1.2 Methodes FILE_DOWN

■ 목적 : Application 내에 Upload 되어 있는 File Download 및 URL 주소를 통한 추가 작업을 하기 위해서 사용 .

1. File Download Event를 특정 UI의 Events에 추가 (본 단원에서는 Upload 한 File Click 시 동작).



2. "FILE_DOWN" Method (File Download & File URL로 Image 불러오기)

[첨부 ABAP SOURCE (EV_FILE_DOWN)]

METHOD EV_FILE_DOWN.

```
DATA: LV_INDEX TYPE I.
DATA: LS_FILE TYPE TY_FILE.
DATA: LV_URL TYPE STRING.
```

" SELECTED FILE INDEX GET.

CALL METHOD /U4A/CL_UTILITIES=>GET_SELECTED_INDEX

EXPORTING

I_EVENT_NAME = I_EVENT_NAME

IMPORTING

E_INDEX = LV_INDEX.

"READ FILE INFORMATION."

READ TABLE GT_FILE INTO LS_FILE INDEX LV_INDEX.

CHECK SY-SUBRC EQ 0.

" SELECTED FILE DOWNLOAD ."

CALL METHOD /U4A/CL_UTILITIES=>FILE_DOWN

EXPORTING

IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW

I_FILENAME = LS_FILE-FNAME *" FILE NAME"*

I_MIME = LS_FILE-MIME *" FILE MIME"*

CONTENTS = LS_FILE-CONTENT *" FILE CONTENTS"*

IMPORTING

E_URL = LV_URL. *" FILE URL"*

"MOVE VALUE : LV_URL TO GS_IMG-SRC."

GS_IMG-SRC = LV_URL.

"PAGE NAVIGATION (IMG PAGE) ."

CALL METHOD /U4A/CL_UTILITIES=>PAGE_NAVIGATION

EXPORTING

IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW

I_APPID = 'NAVCONTAINER1'

I_PAGEID = 'PAGE2'.

ENDMETHOD.

NAME	MEAN
I_FILENAME	DOWNLOAD FILE 이름
I_MIME	DOWNLOAD FILE MIME Type (Optional Parameter)
CONTENTS	DOWNLOAD FILE 내용 정보

E_URL	DOWNLOAD FILE URL 정보
-------	----------------------

3. Upload 되어 있는 File Click.

Title: *U4A WorkSpace File Upload & Down*

Writer: _____

Date: _____

Time: _____

Contents: *U4A WorkSpace File Upload & Down*

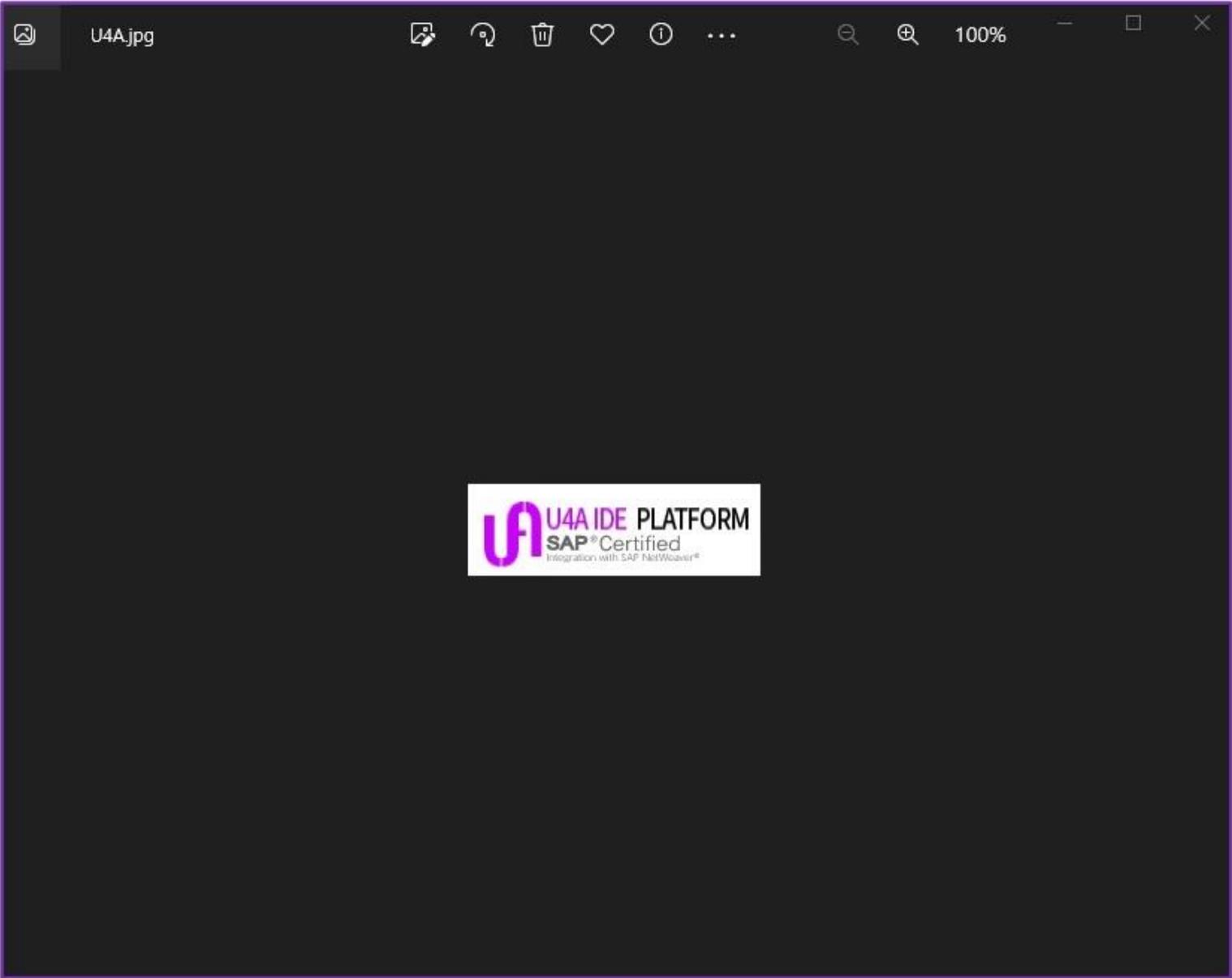
Contents:

 U4A.jpg 6065
[Delete](#)

4. 선택한 File Download 및 해당 URL로 Image 불러오기 확인.



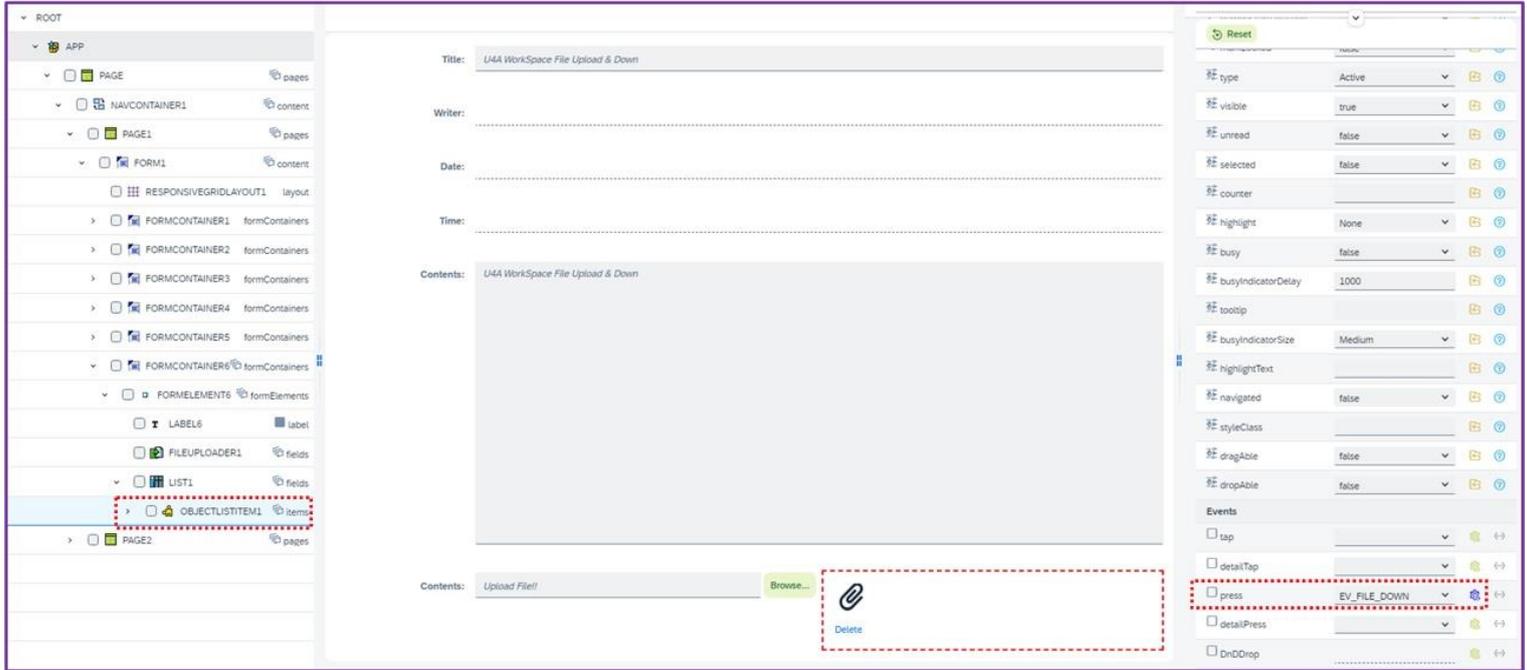
5. Download File 실행하여 정상적으로 Download 되었는지 확인.



6.1.3 Methodes FILE_DOWN2

■ 목적 : Application 내에 Upload 되어 있는 File Download 하기 위해서 사용.

1. File Download Event를 특정 UI의 Events에 추가 (본 단원에서는 Upload 한 File Click 시 동작).



2. "FILE_DOWN2" Method.

[첨부 ABAP SOURCE (EV_FILE_DOWN)]

METHOD EV_FILE_DOWN.

```
DATA: LV_INDEX TYPE I.
DATA: LS_FILE TYPE TY_FILE.
DATA: LV_URL TYPE STRING.
```

" SELECTED FILE INDEX GET.

CALL METHOD /U4A/CL_UTILITIES=>GET_SELECTED_INDEX

EXPORTING

I_EVENT_NAME = I_EVENT_NAME

IMPORTING

E_INDEX = LV_INDEX.

"READ FILE INFORMATION."

READ TABLE GT_FILE INTO LS_FILE INDEX LV_INDEX.

CHECK SY-SUBRC EQ 0.

" SELECTED FILE DOWNLOAD ."

CALL METHOD /U4A/CL_UTILITIES=>FILE_DOWN2

EXPORTING

IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW
 I_FILENAME = LS_FILE-FNAME " FILE NAME
 I_MIME = LS_FILE-MIME " FILE MIME
 CONTENTS = LS_FILE-CONTENT " FILE CONTENTS

EXCEPTIONS

FILE_OR_EXTENTION_REQUIERD = 1
 MIME_TYPE_NOT_FOUND = 2
 OTHERS = 3.

IF SY-SUBRC <> 0.

MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
 WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.

ENDIF.

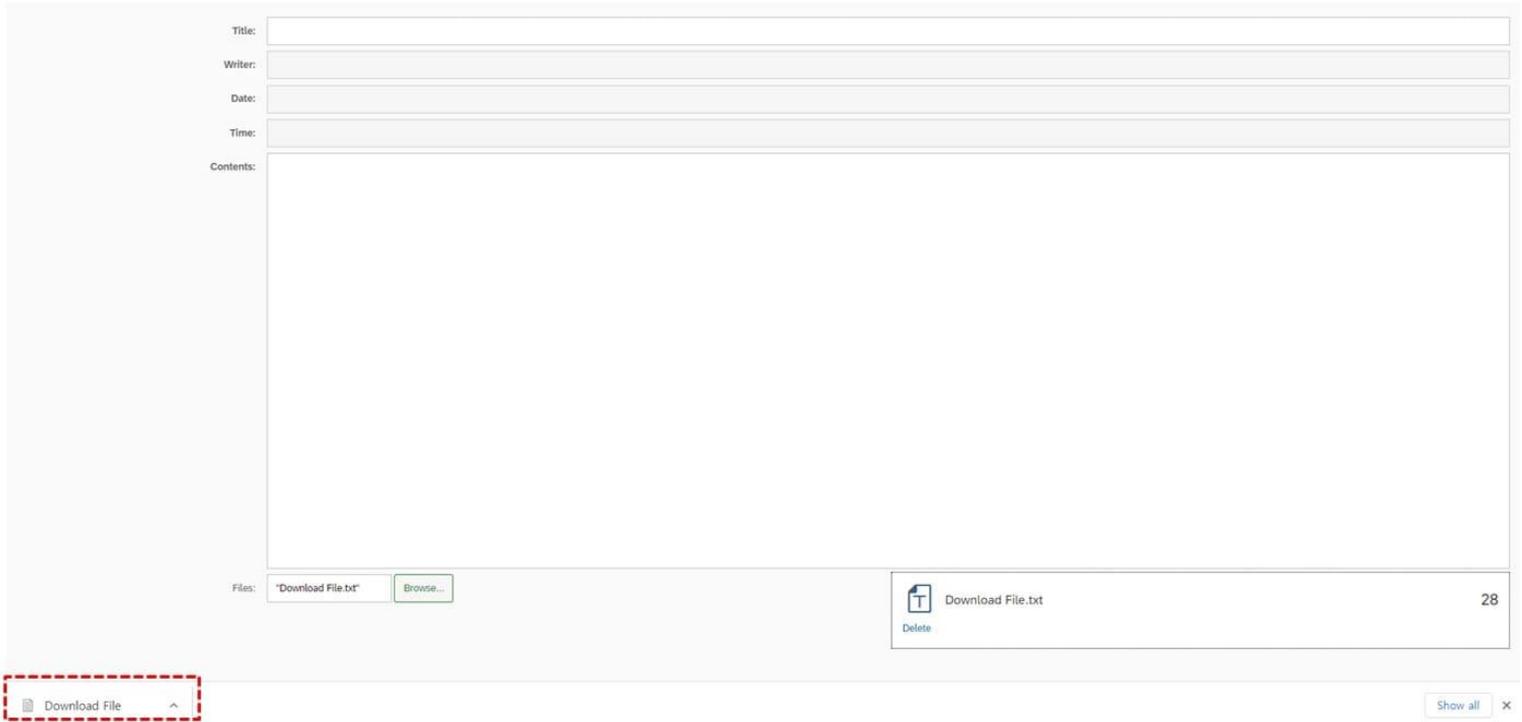
ENDMETHOD.

PARAMETERS	MEAN
I_FILENAME	DOWNLOAD FILE 이름
I_MIME	DOWNLOAD FILE MIME Type (Optional Parameter)
CONTENTS	DOWNLOAD FILE 내용 정보

3. Upload 되어 있는 File Click.

The screenshot displays the U4A IDE interface. At the top, there is a 'Title:' field containing 'U4A WorkSpace File Upload & Down'. Below it are 'Writer:', 'Date:', and 'Time:' fields, each followed by a dashed line. A large 'Contents:' area contains the text 'U4A WorkSpace File Upload & Down'. At the bottom, there is a 'Contents:' field with 'Download File.txt' and a 'Browse...' button. To the right, a file list entry for 'Download File.txt' is shown with a paperclip icon, a 'Delete' link, and the number '38'. This file list entry is enclosed in a red dashed box.

4. 선택한 File Download.



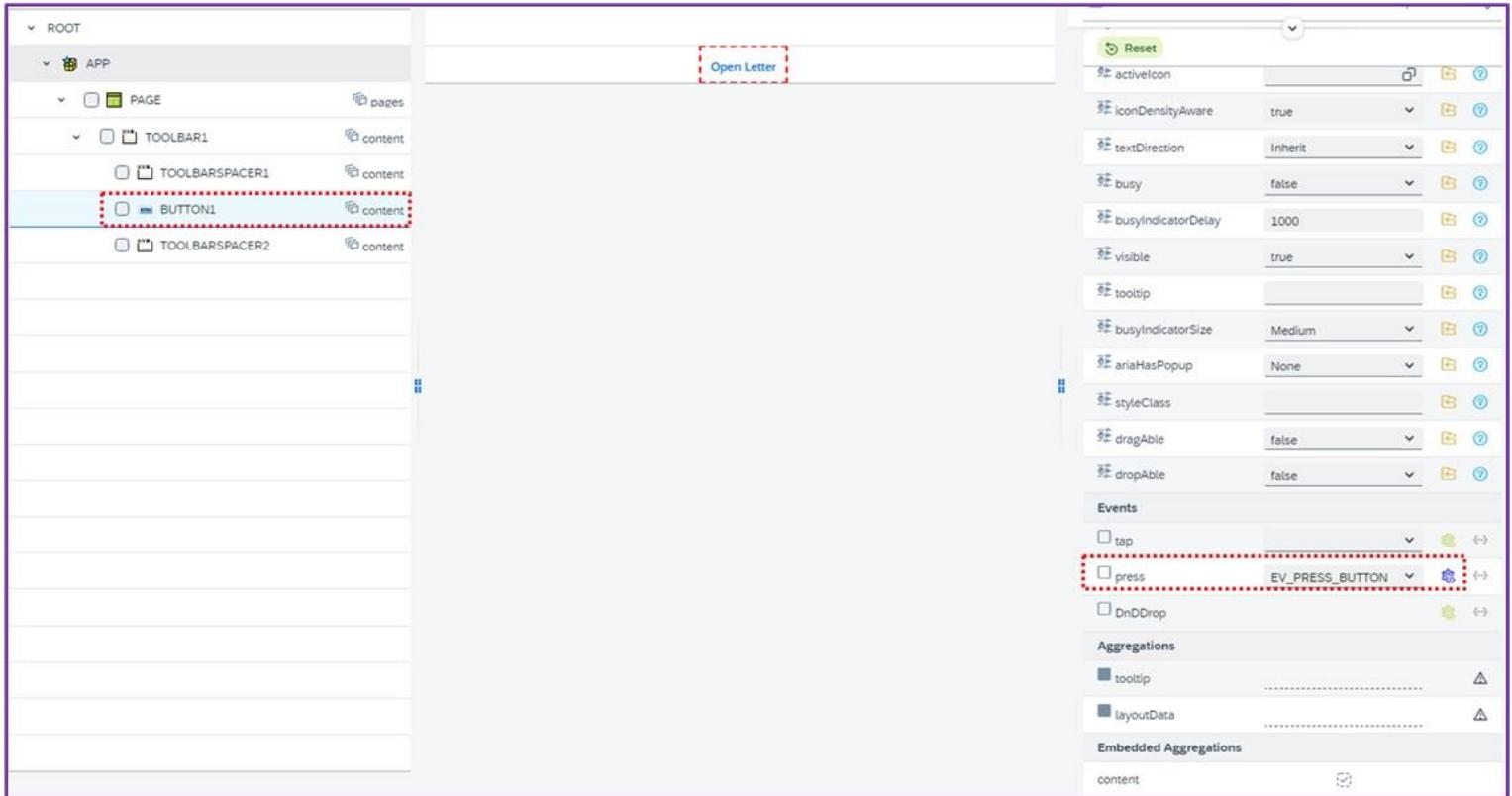
5. Download File 실행하여 정상적으로 Download 되었는지 확인.



6.1.4 Methodes M_LETTERMSG

■ 목적 : 편지 Image를 생성하고 이를 통한 Call Back Event를 사용하기 위한 용도 .

1. "M_LETTERMSG" Method를 실행할 UI 구성 (본 단원에서는 Button Click하여 Method 실행).



2. "M_LETTERMSG" Method.

[첨부 ABAP SOURCE (EV_PRESS_BUTTON)]

```
METHOD EV_PRESS_BUTTON.
```

```
CALL METHOD /U4A/CL_UTILITIES=>M_LETTERMSG
```

```
EXPORTING
```

```
IO_VIEW      = ME->/U4A/IF_SERVER~AR_VIEW
```

```
I_VPOS       = /U4A/CL_UTILITIES=>CS_M_MSG_TOS_VPOS-
```

```
CENTER      " Vertical Position
```

```
I_HPOS       = /U4A/CL_UTILITIES=>CS_M_MSG_TOS_HPOS-CENTER " Horizontal Position
```

```

I_CALLBACK_EVENT = 'EV_MSG_OPEN'
I_DURATION      = 250
I_EXPAND       = ''
I_SCALE        = 3.
    
```

간 (1000: 1초)
 침, ' ' : 닫힘)

" 닫기 후 콜백 이벤트명
 " fadeIn & fadeOut 시
 " 펼치기 여부 (X: 펼
 " Message Size (default : 1)

ENDMETHOD.

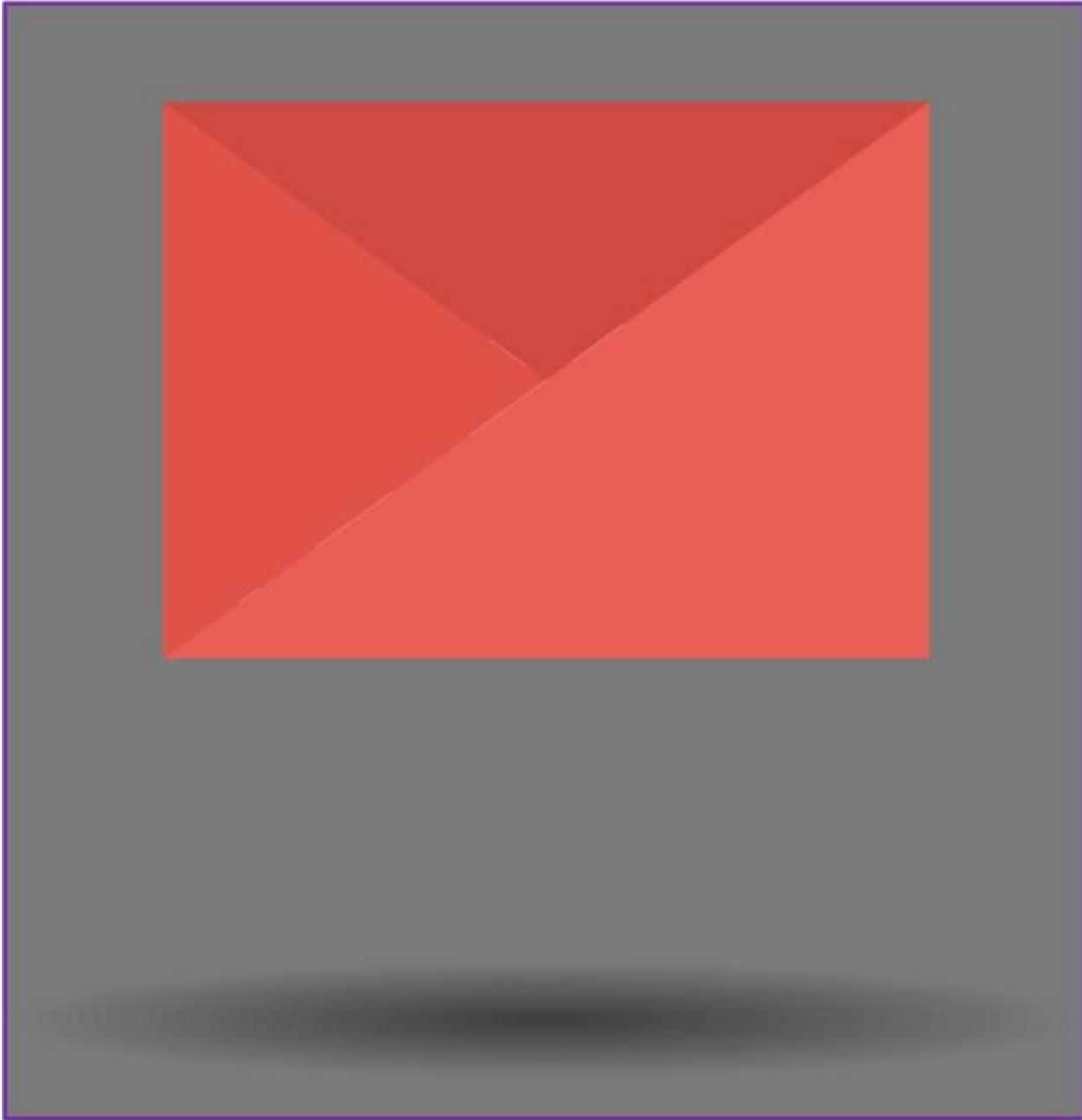
NAME	MEAN	
I_VPOS	편지 Image 세로 위치	
	CS_M_MSG_TOS_VPOS-BOTTOM	하단
	CS_M_MSG_TOS_VPOS-CENTER	중단
	CS_M_MSG_TOS_VPOS-TOP	상단
I_HPOS	편지 Image 가로 위치	
	CS_M_MSG_TOS_HPOS-BOTTOM	하단
	CS_M_MSG_TOS_HPOS-CENTER	중단
	CS_M_MSG_TOS_HPOS-TOP	상단
I_CALLBACK_EVENT	편지 Image 사라진 후 Event	
I_DURATION	편지 Image Click 시 사라질 때 까지 시간	
I_EXPAND	편지 Image가 펼쳐진 형태인지 아닌지 여부	
I_SCALE	편지 Image 크기	

3. Application 실행한 후 Button Click하여 Method 실행.



4. "M_LetterMsg" Method 실행 화면 (좌: I_EXPAND = 'X' , 우: I_EXPAND = ' ').





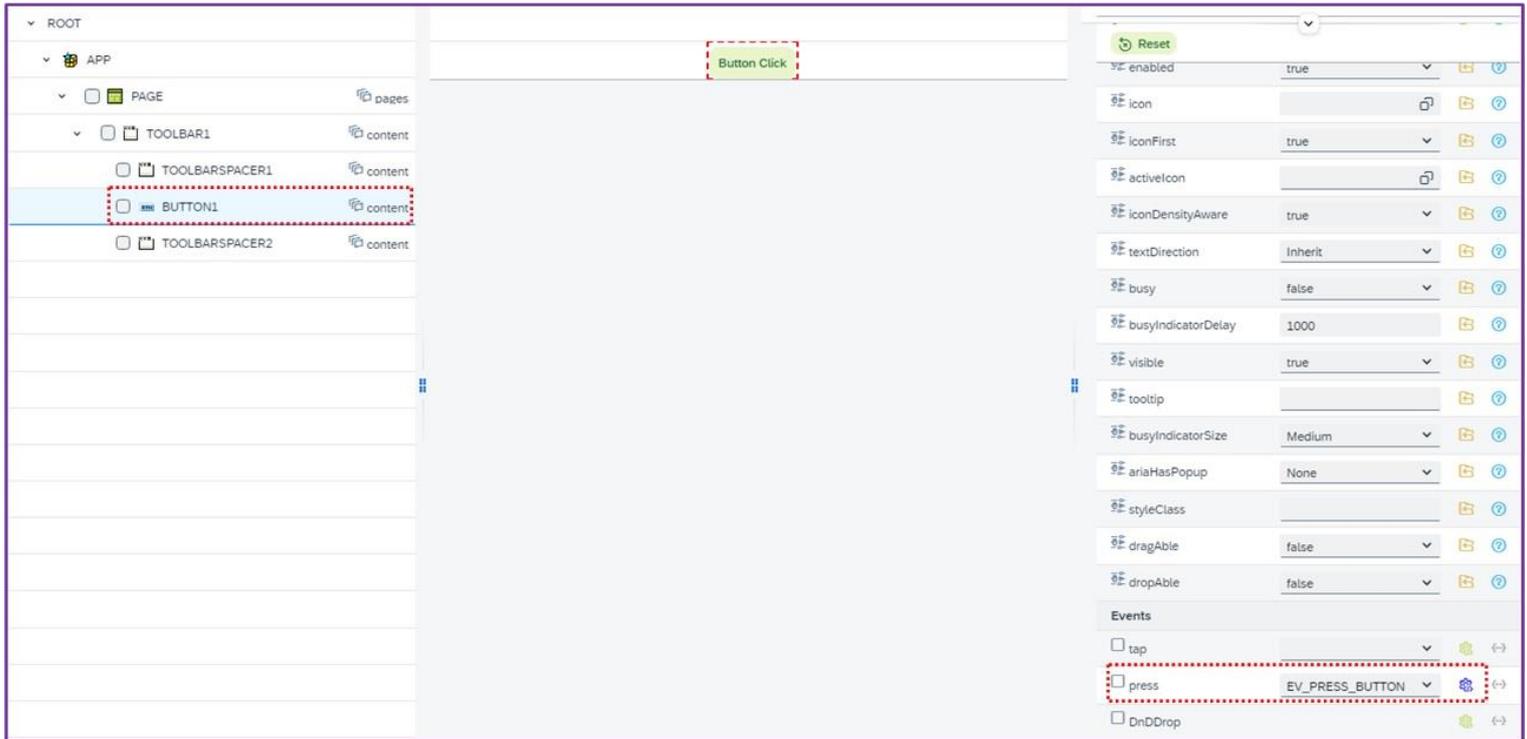
5. 편지 Image Click 하면 CallBack Method 실행 .



6.1.5 Methodes SET_FREESTYLE_SCRIPT

■ 목적 : Method 내에서 Java Script 명령문을 적용하기 위해서 사용 .

1. Java Script가 동작 할 UI 및 Method 설정 (본 단원에서는 Button Click하여 Script 문 실행) .



2. Java Script Method.

[첨부 ABAP SOURCE (EV_PRESS_BUTTON)]

```
METHOD EV_PRESS_BUTTON.
```

```
DATA: LV_SCRIPT TYPE STRING.
```

```
DATA: LT_SCRIPT TYPE STRING_TABLE.
```

```
*-----  
" Java Script Set.
```

```
LV_SCRIPT = 'alert("Hello U4A Workspace!!!."); {}'.
```

```
APPEND LV_SCRIPT TO LT_SCRIPT.
```

```
" Java Script Execution.
```

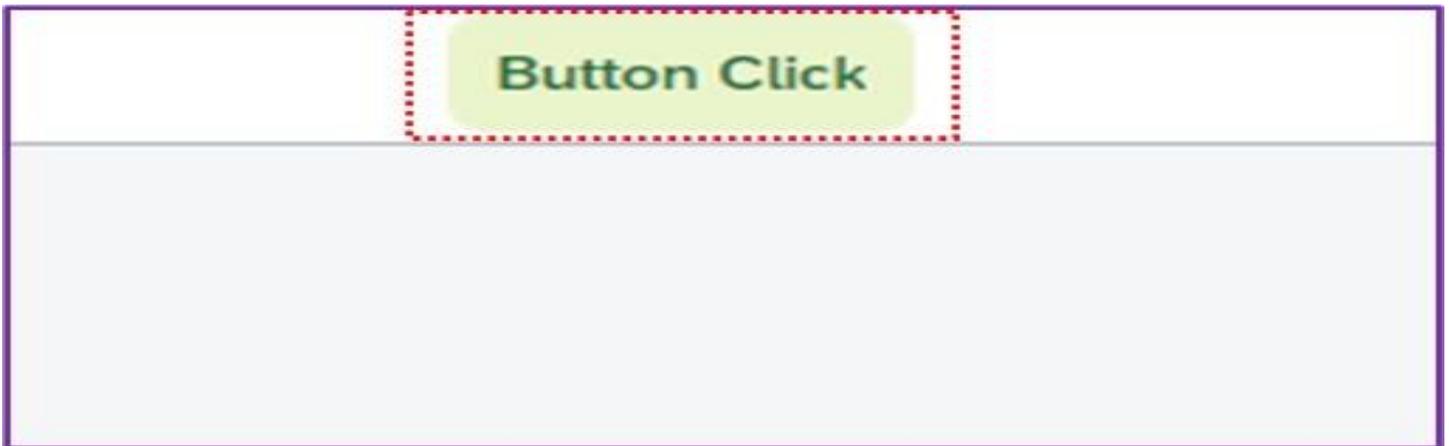
```
CALL METHOD /U4A/CL_UTILITIES=>SET_FREESTYLE_SCRIPT  
EXPORTING
```

```
IO_VIEW    = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class  
IT_SCRIPT  = LT_SCRIPT.                " Table of Strings
```

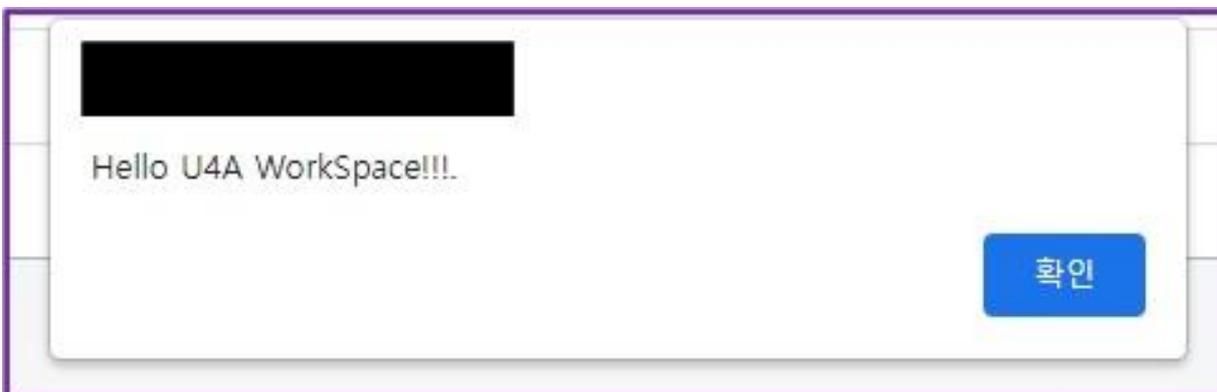
```
ENDMETHOD.
```

String_table 형태로 Java Script를 받아서 "SET_FREESTYLE_SCRIPT" Method 실행.

3. Application을 실행한 후 Button Click.



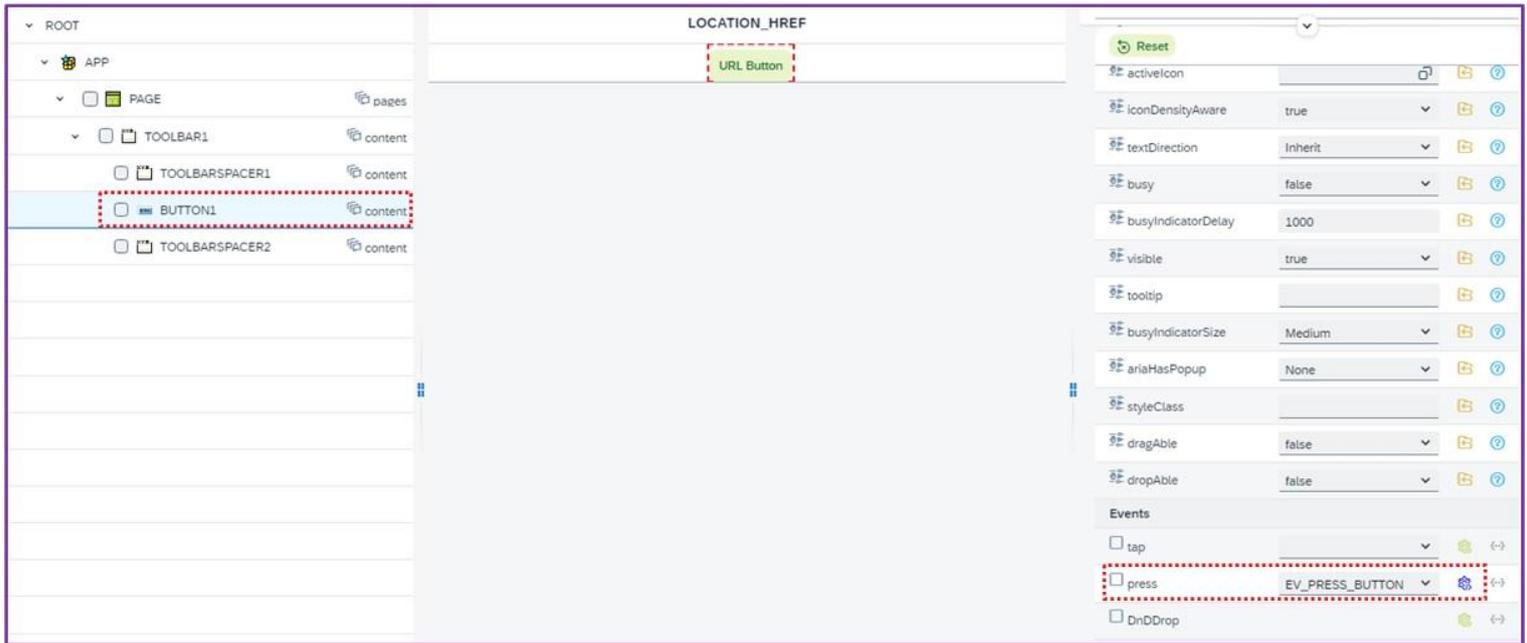
4. "Press_Button" Method에 설정해 둔 Java Script가 실행 된 것을 확인.



6.1.6 Methodes LOCATION_HREF

■ 목적 : URL 주소를 이용하여 Navigate 하기 위한 용도.

1. "LOCATION_HREF" Method를 동작할 Event를 특정 UI의 Events에 추가 (본 단원에서는 BUTTON Click하여 동작).



2. " LOCATION_HREF" Method (Application의 URL을 통해 이동).

[첨부 ABAP SOURCE (EV_MOVE_URL)]

METHOD EV_MOVE_URL.

"Movement Through URL"

CALL METHOD /U4A/CL_UTILITIES=>LOCATION_HREF

EXPORTING

IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class

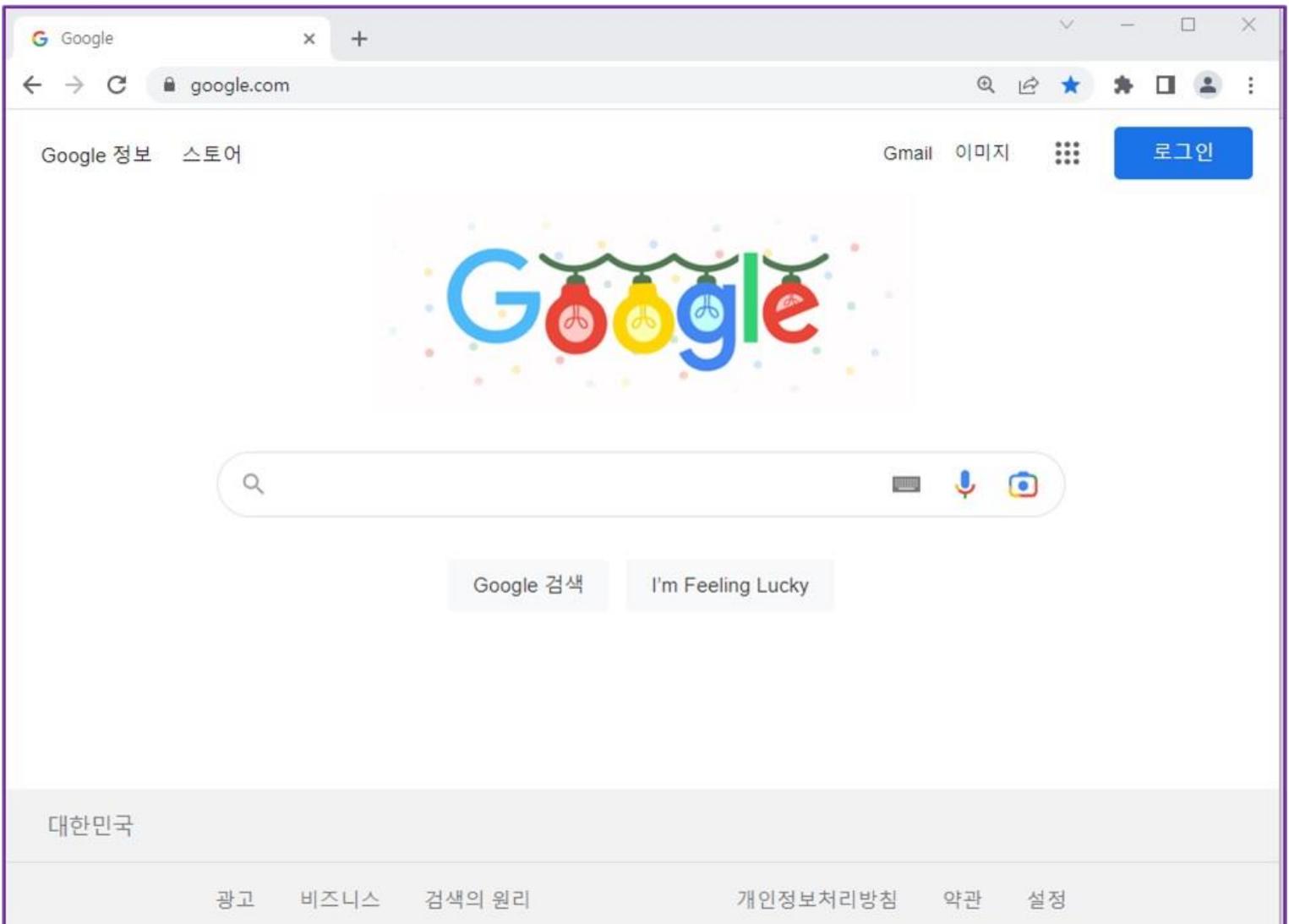
URL = 'https://www.google.com/'. "이동 할 URL .

ENDMETHOD.

3. Application 실행한 후 "Location_Href" Method 동작을 위한 Button Click.



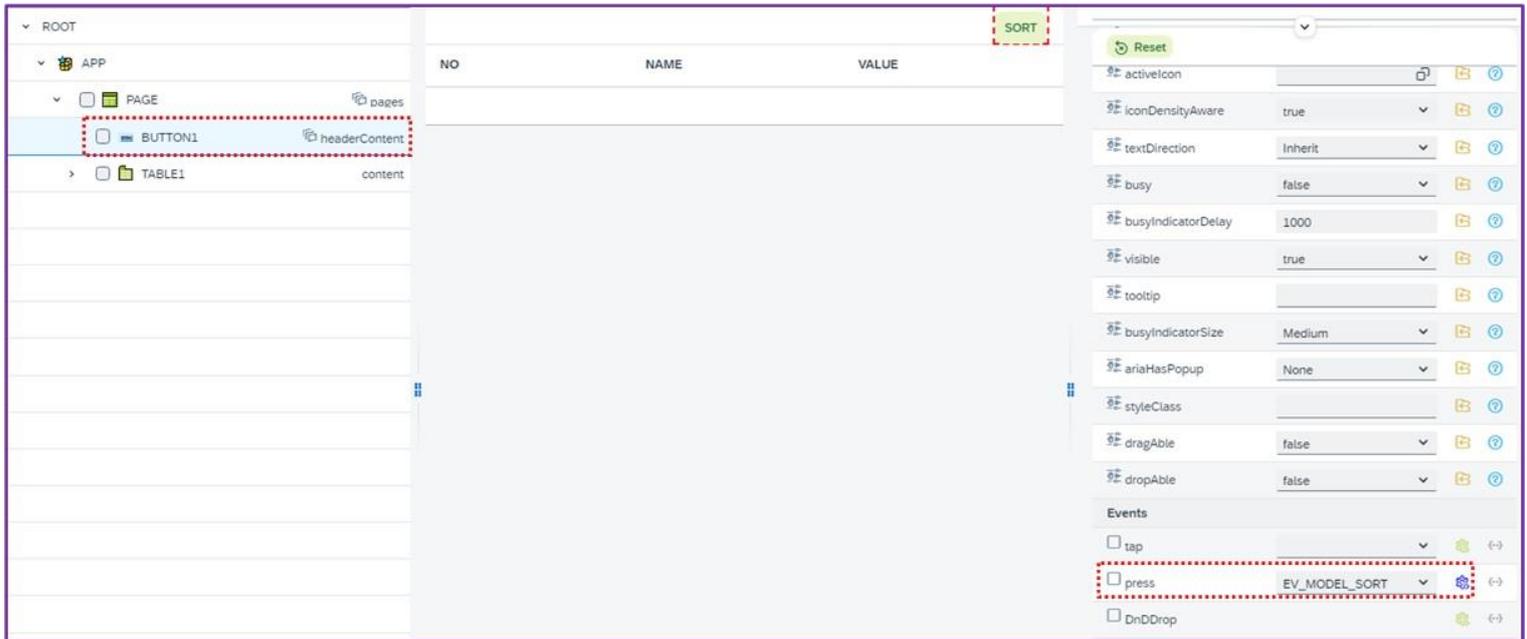
4. PARAMETER로 입력한 URL의 사이트로 이동한 것을 확인.



6.1.7 Methodes SET_MODEL_SORTER

■ 목적 : Sort 기능이 없는 UI에 Binding되어 있는 Data를 정렬하고 그룹화 하기 위해서 사용.

1. " Set_Model_sorter"를 동작할 Event를 특정 UI의 Events에 추가 (본 단원에서는 BUTTON Click 하여 동작).



2. " SET_MODEL_SORTER" Method (Sort 기능이 없는 UI도 정렬이 가능해짐).

[첨부 ABAP SOURCE (EV_MODEL_SORT)]

```
METHOD EV_MODEL_SORT.
```

```
DATA: LS_SORTER TYPE /U4A/S0083.
```

```
DATA: LT_SORTER TYPE /U4A/Y0083.
```

```
" APPEND LT_SORTER DATA.
```

```
LS_SORTER-FIELD_NAME = 'VALUE'.
```

```
LS_SORTER-IS_DESCENDING = 'X'.
```

```
LS_SORTER-ISNUM = 'X'.
```

```
APPEND LS_SORTER TO LT_SORTER.
CLEAR LS_SORTER.
```

```
"MODEL_SORTER
CALL METHOD /U4A/CL_UTILITIES=>SET_MODEL_SORTER
EXPORTING
    IO_VIEW      = ME->/U4A/IF_SERVER~AR_VIEW
    I_UIID       = 'TABLE1'           " UI ID
    I_AGGR_NAME  = 'items'          " Aggregation Name
    IT_SORTER    = LT_SORTER        " [U4A] Sorter Info Table type
    I_GROUP_FIELD = 'VALUE'         " Group Field Name

ENDMETHOD.
```

PARAMETERS	MEAN	
I_UIID	DATA가 BINDING 되어 있는 UI의 INSTANCE ID 이름	
I_AGGR_NAME	UI의 DATA를 BINDING 시킨 AGGREGATION 이름 ( 반드시 소문자로 작성해야 함)	
IT_SORTER	SORT 기준이 되는 TABLE	
	FIELD	MEAN
	FIELD_NAME	정렬 기준이 되는 FIELD 명
	IS_DESCENDING	정렬 방식 (X: 내림차순 ' ': 오름차순)
	ISNUM	숫자 타입인지 여부 (X: 숫자 ' ': 문자)
I_GROUP_FIELD	그룹화 기준이 되는 FIELD 이름	

3. Application 실행한 후 "SET_MODEL_SORTER Method 동작을 위한 Button Click.

			SORT
NO	NAME	VALUE	
1000001	COLUMN1	11111	
1000002	COLUMN2	22222	
1000003	COLUMN3	33333	
1000004	COLUMN4	44444	
1000005	COLUMN5	44444	

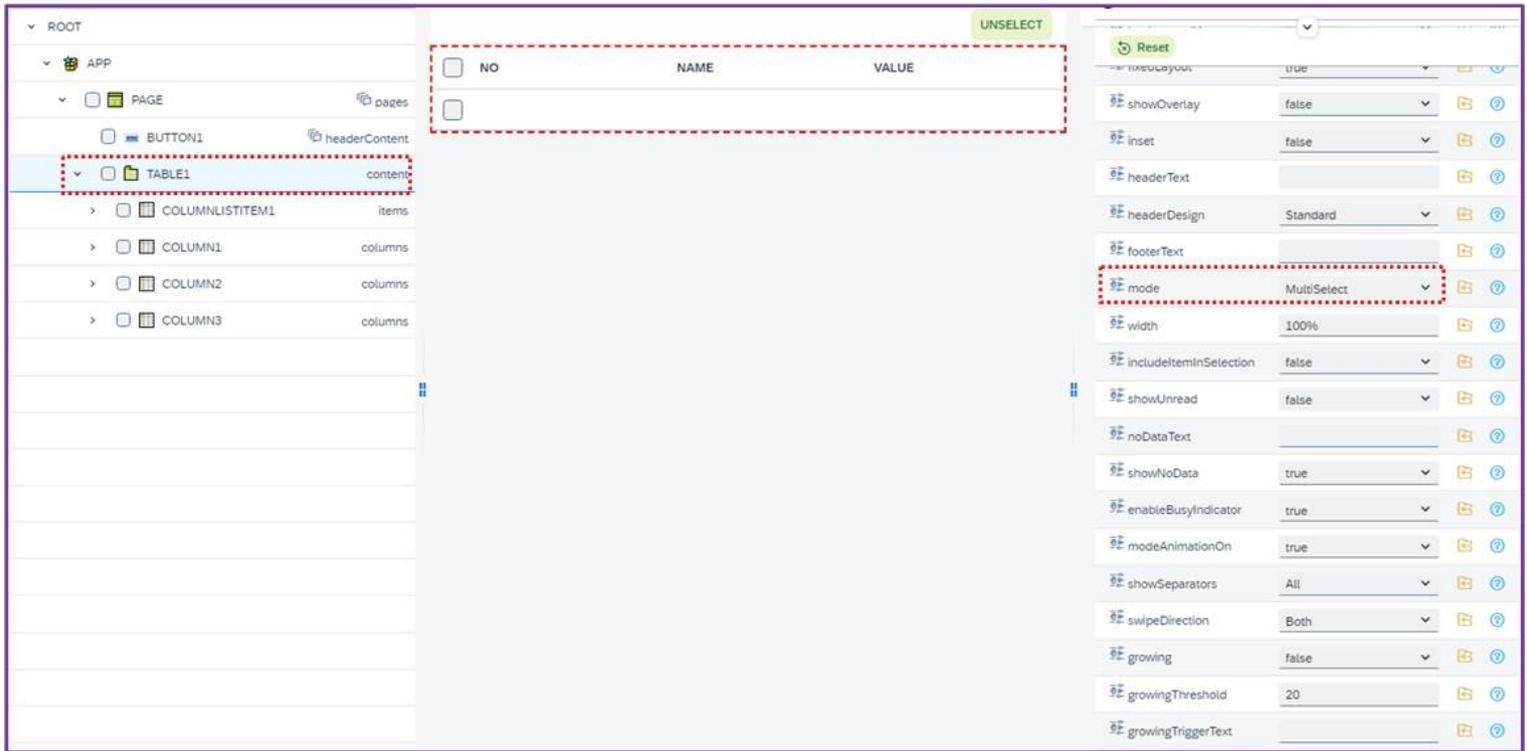
4. Table 결과가 Value 값이 Descending 정렬 되고 같은 Value 값은 그룹화 된 것을 확인.

			SORT
NO	NAME	VALUE	
44444			
1000004	COLUMN4	44444	
1000005	COLUMN5	44444	
33333			
1000003	COLUMN3	33333	
22222			
1000002	COLUMN2	22222	
11111			
1000001	COLUMN1	11111	

6.1.8 Methodes SET_M_TABLE_UNSELECT

■ 목적 : Select 상태의 Checkbox를 모두 UnSelect하기 위해 사용.

1. M.Table의 Mode를 "MultiSelect"로 변경한 후 "SET_M_TABLE_UNSELECT" Method 동작할 Event를 특정 UI의 Events에 추가 (본 단원에서는 BUTTON Click 시 동작).



2. " SET_M_TABLE_UNSELECT" Method

[첨부 ABAP SOURCE (EV_M_TABLE_UNSELECT)]

```
METHOD EV_M_TABLE_UNSELECT.
```

```
"SET_M_TABLE_UNSELECT
```

```
CALL METHOD /U4A/CL_UTILITIES=>SET_M_TABLE_UNSELECT
```

```
EXPORTING
```

```
IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW
```

```
I_UIID = 'TABLE1'.
```

```
" DATA BINDING TABLE.
```

ENDMETHOD.

3. Application 실행 후 Row 여러 건 Select 후 Button Click.

<input type="checkbox"/>	NO	NAME	VALUE
<input type="checkbox"/>	1000001	COLUMN1	11111
<input checked="" type="checkbox"/>	1000002	COLUMN2	22222
<input checked="" type="checkbox"/>	1000003	COLUMN3	33333
<input checked="" type="checkbox"/>	1000004	COLUMN4	44444
<input type="checkbox"/>	1000005	COLUMN5	44444

UNSELECT

4. Check 되어 있던 Row가 모두 UnSelected 되는 것을 확인.

<input type="checkbox"/>	NO	NAME	VALUE
<input type="checkbox"/>	1000001	COLUMN1	11111
<input type="checkbox"/>	1000002	COLUMN2	22222
<input type="checkbox"/>	1000003	COLUMN3	33333
<input type="checkbox"/>	1000004	COLUMN4	44444
<input type="checkbox"/>	1000005	COLUMN5	44444

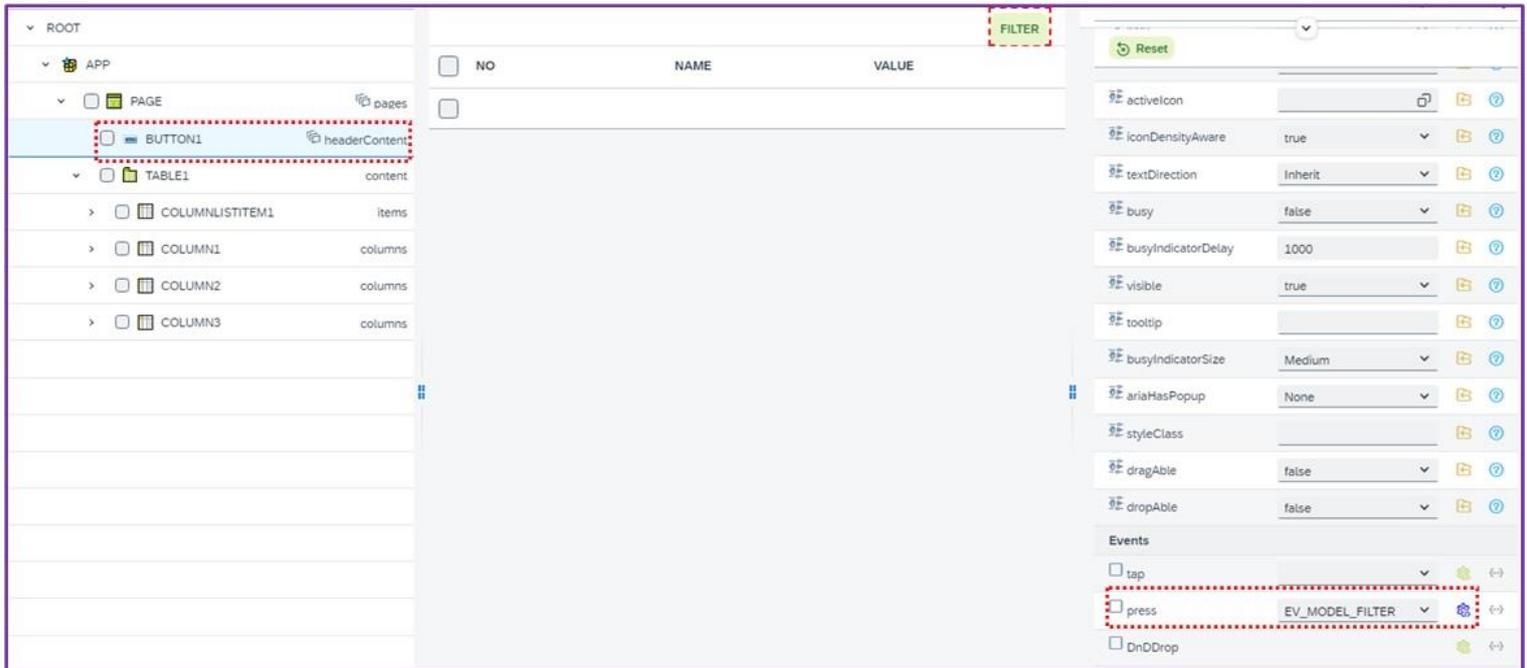
UNSELECT

 Table Type은 반드시 M.Table을 사용 (UI.Table은 허용 하지 않음)

6.1.9 Methodes SET_MODEL_FILTER

▣ 목적 : Filter 기능이 없는 UI에 Binding되어 있는 Data를 Filtering 하기 위해서 사용.

1. "SET_MODEL_FILTER" 동작할 Event를 특정 UI의 Events에 추가 (본 단원에서는 BUTTON Click하여 동작).



2. " SET_MODEL_FILTER" Method (Binding 되어 있는 Data Filtering 가능).

[첨부 ABAP SOURCE (EV_MODEL_FILTER)]

```
METHOD EV_MODEL_FILTER.
```

```
DATA: LS_FILTER TYPE /U4A/S0081.
```

```
DATA: LT_FILTER TYPE TABLE OF /U4A/S0081.
```

```
" APPEND LT_FILTER DATA.
```

```
LS_FILTER-FIELD_NAME = 'VALUE'.
```

```
LS_FILTER-FILTER_OPTION = 'BT'.
```

```
LS_FILTER-LOW           = 500.
LS_FILTER-HIGH         = 1000.
LS_FILTER-ISNUM        = 'X'.
LS_FILTER-CASESENSITIVE = 'X'.
```

```
APPEND LS_FILTER TO LT_FILTER.
CLEAR  LS_FILTER.
```

```
" MODEL_FILTER.
CALL METHOD /U4A/CL_UTILITIES=>SET_MODEL_FILTER
EXPORTING
    IO_VIEW      = ME->/U4A/IF_SERVER~AR_VIEW
    I_UIID       = 'TABLE1'      " 필터 대상 UI명 .
    I_AGGR_NAME = 'items'       " 필터 대상 하위 Aggregation 명 .
    IT_FILTER    = LT_FILTER    " 다중 필터링 정보(필드별 필터링) .
    I_OPERATOR   = 'X'.         " Filter Operator (' ' = OR, 'X' = AND) .

ENDMETHOD.
```

PARAMETERS	MEAN														
I_UIID	DATA가 BINDING 되어 있는 UI의 INSTANCE ID 이름														
I_AGGR_NAME	UI의 DATA를 BINDING 시킨 AGGREGATION 이름 (반드시 소문자로 작성해야 함)														
IT_FILTER	FILTER 기준이 되는 TABLE														
	<table border="1"> <thead> <tr> <th>FIELD</th> <th>MEAN</th> </tr> </thead> <tbody> <tr> <td>FIELD_NAME</td> <td>FILTER 기준이 되는 FIELD 이름</td> </tr> <tr> <td>FILTER_OPTION</td> <td>FILTER OPTION EX) EQ, NE, GE,GT, LE, LT</td> </tr> <tr> <td>LOW</td> <td>FIELD VALUE 최소 값</td> </tr> <tr> <td>HIGH</td> <td>FIELD VALUE 최대 값</td> </tr> <tr> <td>ISNUM</td> <td>숫자 Type인지 여부 (X: 숫자 ' ' : 문자)</td> </tr> <tr> <td>CASESENSITIVE</td> <td>대소문자 구분 (X: 대소문자 구분 함 ' ' : 대소문자 구분 못함)</td> </tr> </tbody> </table>	FIELD	MEAN	FIELD_NAME	FILTER 기준이 되는 FIELD 이름	FILTER_OPTION	FILTER OPTION EX) EQ, NE, GE,GT, LE, LT	LOW	FIELD VALUE 최소 값	HIGH	FIELD VALUE 최대 값	ISNUM	숫자 Type인지 여부 (X: 숫자 ' ' : 문자)	CASESENSITIVE	대소문자 구분 (X: 대소문자 구분 함 ' ' : 대소문자 구분 못함)
	FIELD	MEAN													
	FIELD_NAME	FILTER 기준이 되는 FIELD 이름													
	FILTER_OPTION	FILTER OPTION EX) EQ, NE, GE,GT, LE, LT													
	LOW	FIELD VALUE 최소 값													
	HIGH	FIELD VALUE 최대 값													
	ISNUM	숫자 Type인지 여부 (X: 숫자 ' ' : 문자)													
CASESENSITIVE	대소문자 구분 (X: 대소문자 구분 함 ' ' : 대소문자 구분 못함)														

I_OPERATOR	Filter Operator (X: AND ' ': OR)
------------	-----------------------------------

3. Application 실행한 후 " Set_Model_Filter" Method 동작을 위한 Button Click.

<input type="checkbox"/>	NO	NAME	VALUE
<input type="checkbox"/>	1000001	COLUMN1	120
<input type="checkbox"/>	1000002	COLUMN2	6547
<input type="checkbox"/>	1000003	COLUMN3	364
<input type="checkbox"/>	1000004	COLUMN4	20
<input type="checkbox"/>	1000005	COLUMN5	999

FILTER

4. Table 결과가 Filter 조건에 따라 Value 값이 500 ~ 1000 사이의 값에 해당하는 999인 Row 만이 Filtering 된 것을 확인.

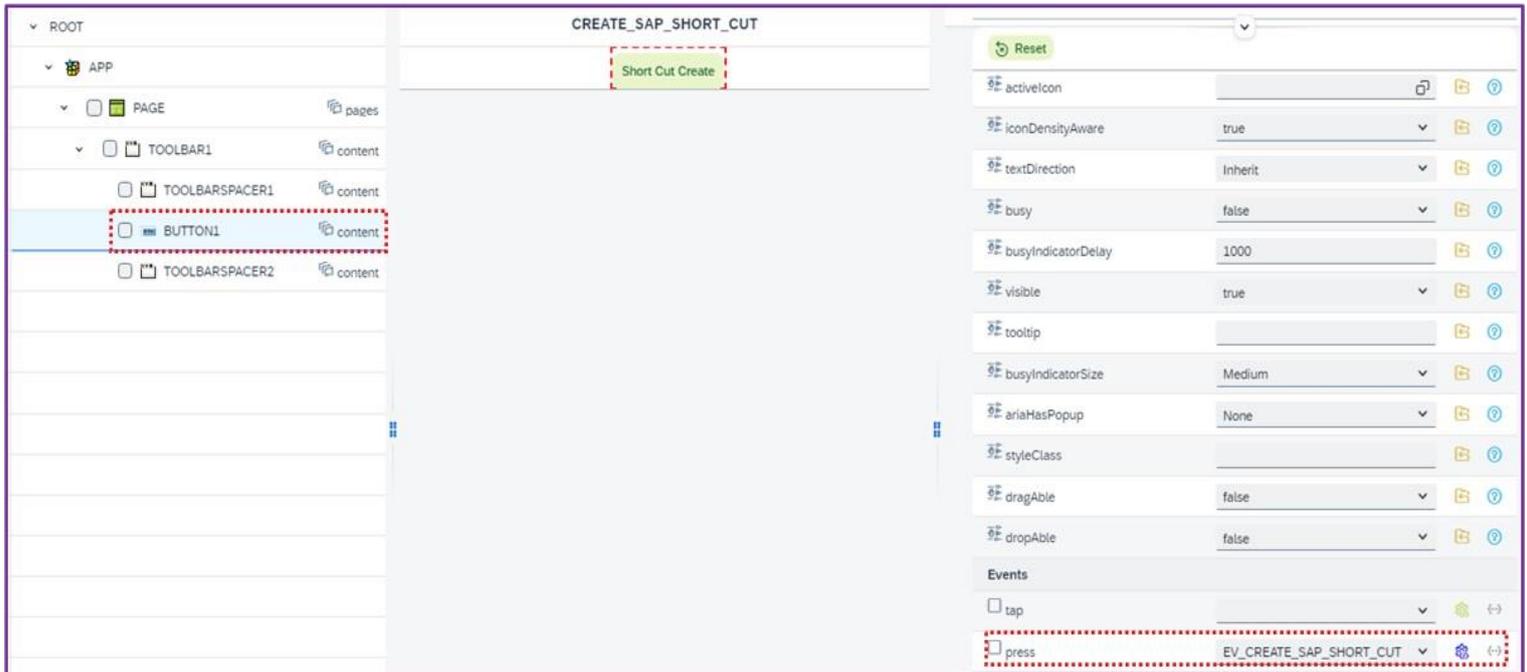
<input type="checkbox"/>	NO	NAME	VALUE
<input type="checkbox"/>	1000005	COLUMN5	999

FILTER

6.1.10 Methodes CREATE _SAP_SHORT_CUT

■ 목적 : Method를 통해서 T_Code에 대한 Short_Cut 생성을 하기 위해서 사용.

1. "CREATE_SAP_SHORT_CUT" Method 동작할 Event를 특정 UI의 Events에 추가 (본 단원에서는 BUTTON Click 하여 동작).



2. "CREATE_SAP_SHORT_CUT" Method.

[첨부 ABAP SOURCE (EV_CREATE_SAP_SHORT_CUT)]

METHOD EV_CREATE_SAP_SHORT_CUT.

```

DATA: LV_RTCOD TYPE SYSUBRC.
DATA: LV_RTMSG TYPE STRING.
DATA: LV_FNAME TYPE STRING.
DATA: LV_RESULT TYPE XSTRING.
DATA: LV_URL TYPE STRING.
    
```

" CREATE_SAP_SHORT_CUT .

CALL METHOD /U4A/CL_UTILITIES=>CREATE_SAP_SHORT_CUT

EXPORTING

IV_SYSEM = SY-SYSID	" System SID
IV_DESCR = 'u4e [210.179.213.249]'	" System Description
IV_MANDT = SY-MANDT	" Client
IV_USER = SY-UNAME	" SAP User ID
IV_LANGU = SY-LANGU	" Logon Language
IV_TITLE = 'Object Navigator'	" Command Title
IV_COMMD = 'SE80'	" Command(T-Code)
IV_WINDW = /U4A/CL_UTILITIES=>CO_SHORTCUT_WIN_MAX	" Window Size
IV_REUSE = /U4A/CL_UTILITIES=>CO_YES_1	" Short Cut Reuse (Yes:1, No:0)

IMPORTING

EV_RTCOD = LV_RTCOD	" Return Code(0: Ok, 4: Error)
EV_RTMSG = LV_RTMSG	" Return Message
EV_FNAME = LV_FNAME	" Short Cut File Name
EV_RESULT = LV_RESULT.	" Short Cut Commands

" SHORT_CUT_FILE_DOWNLOAD .

CALL METHOD /U4A/CL_UTILITIES=>FILE_DOWN

EXPORTING

IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW
I_FILENAME = LV_FNAME
CONTENTS = LV_RESULT

IMPORTING

E_URL = LV_URL.

ENDMETHOD.

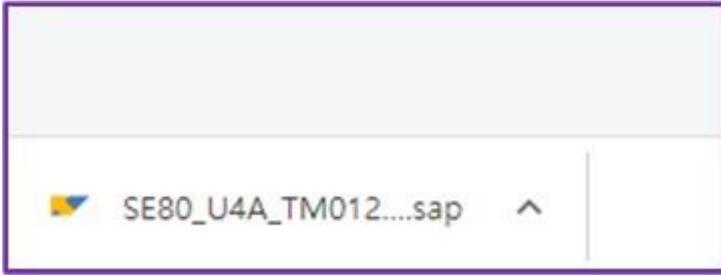
PARAMETERS	MEAN
IV_SYSEM	System ID
IV_DESCR	System 내역 (Optional)
IV_MANDT	Client 번호
IV_USER	접속 User ID

IV_LANGU	Login 언어	
IV_TITLE	권한 체크 화면 Title 이름 (Optional)	
IV_COMMD	T-CODE 이름	
IV_WINDW	WINDOW 창 크기	
	CO_SHORTCUT_WIN_MAX	'Maximized'
	CO_SHORTCUT_WIN_NOR	'Normal window'
IV_REUSE	재사용 여부	
	CO_YES_1	1 (TRUE)
	CO_NO_0	0 (FALSE)
EV_RTCOD	성공 및 Error 여부 Check	
EV_RTMSG	성공 및 Error 여부 Message	
EV_FNAME	File 이름	
EV_RESULT	File 내용 정보 (T-CODE)	

3. Application 실행한 후 " CREATE_SAP_SHORT_CUT" Method 동작을 위한 Button Click.



4. Short_Cut File을 Download.



5. Download 된 Short_Cut File을 Click하여 실행.



6. File 실행 하기 위한 권한 Check, User Name과 Password 입력 후 Log On Button Click.

(화면 상단에 Method에서 설정한 IV_TITLE 값으로 Title 설정 된 것을 확인).



7. T-COED SE80인 Object Navigator 화면이 생성.



6.1.11 Methodes UI_PRINT

▣ 목적 : 화면 전체를 Print 하지 않고 특정 UI 만 선택하여 Print 하기 위한 용도.

1. "UI_PRINT" Method 동작할 Event를 특정 UI의 Events에 추가 (본 단원에서는 BUTTON Click 하여 동작).



2. "UI_PRINT" Method (화면에서 원하는 Ui만 선택해서 Print 가능).

[첨부 ABAP SOURCE (EV_UI_PRINT)]

METHOD EV_UI_PRINT.

" UI_PRINT .

CALL METHOD /U4A/CL_UTILITIES=>UI_PRINT

EXPORTING

IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW

I_UIID = 'TABLE1'

" UI Object ID 예:APP,PAGE.. 워크벤치 디자인 영역

예 UI ID 명

```
I_TITLE = 'TABLE1 PRINT'. " Print Title
```

```
ENDMETHOD.
```

3. Application 실행한 후 " UI_PRINT" Method 동작을 위한 Button Click.

<input type="checkbox"/>	NO	NAME	VALUE
<input type="checkbox"/>	1000001	COLUMN1	120
<input type="checkbox"/>	1000002	COLUMN2	6547
<input type="checkbox"/>	1000003	COLUMN3	364
<input type="checkbox"/>	1000004	COLUMN4	20
<input type="checkbox"/>	1000005	COLUMN5	999



4. Method에서 설정한 I_TITLE 명으로 Print Title이 설정 되며 화면 전체가 아닌 Table 만 Print 대상으로 설정 된 것을 확인 .

U4A IDE HELP

22. 12. 22. 오후 3:10

TABLE1 PRINT

<input type="checkbox"/> NO	NAME	VALUE
<input type="checkbox"/> 1000001	COLUMN1	120
<input type="checkbox"/> 1000002	COLUMN2	6547
<input type="checkbox"/> 1000003	COLUMN3	364
<input type="checkbox"/> 1000004	COLUMN4	20
<input type="checkbox"/> 1000005	COLUMN5	999

인쇄 용지 1장

대상 Samsung C480W

페이지 전체

매수 1

레이아웃 세로 방향

컬러 컬러

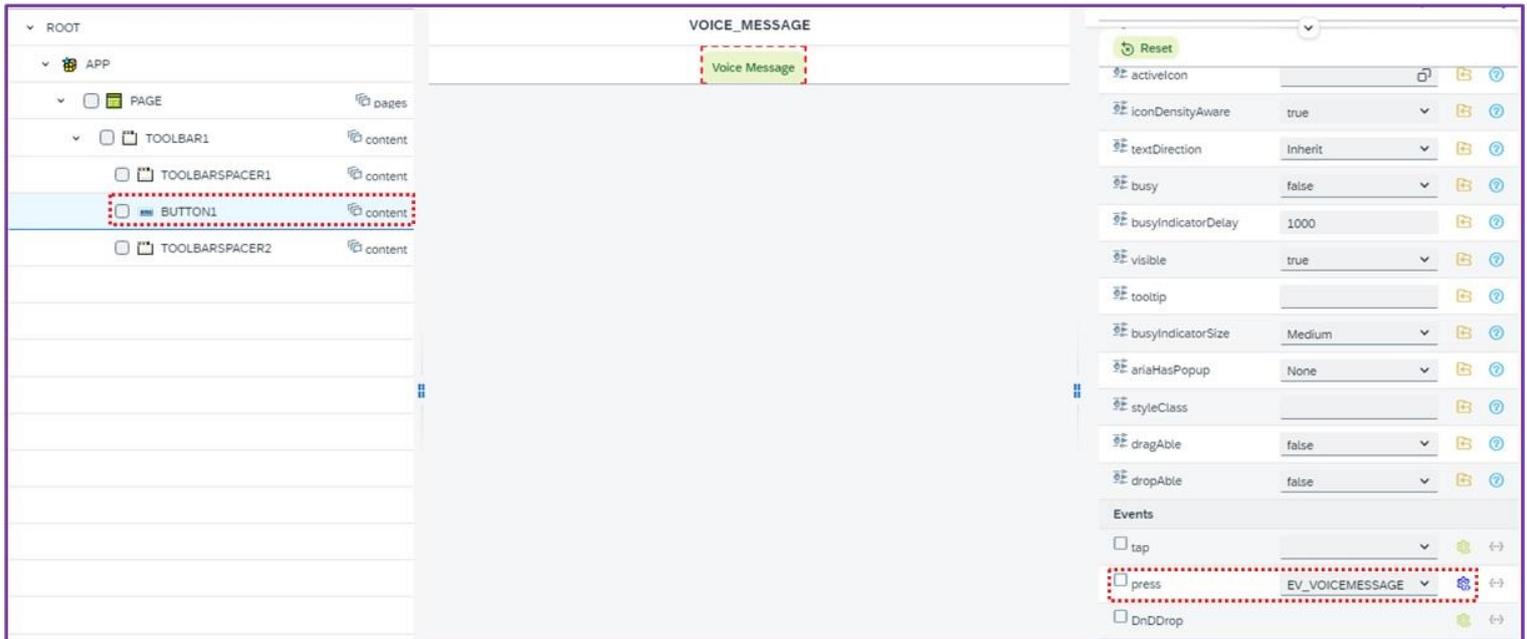
설정 더보기

인쇄 취소

6.1.12 Methodes VOICE_MESSAGE

■ 목적 : 음성 Message를 출력하기 위해서 사용.

1. "Voice_Message" Method 동작할 Event를 특정 UI의 Events에 추가 (본 단원에서는 BUTTON Click 하여 동작).



2. "Voice_Message" Method (메시지 내용 음성으로 출력).

[첨부 ABAP SOURCE (EV_VOICEMESSAGE)]

```
METHOD EV_VOICEMESSAGE.
```

```
DATA LV_MSG TYPE STRING.
```

```
"Write Message
```

```
LV_MSG = 'Hello U4A Workspace'.
```

" VOICE_MESSAGE 출력.

CALL METHOD /U4A/CL_UTILITIES=>VOICE_MESSAGE

EXPORTING

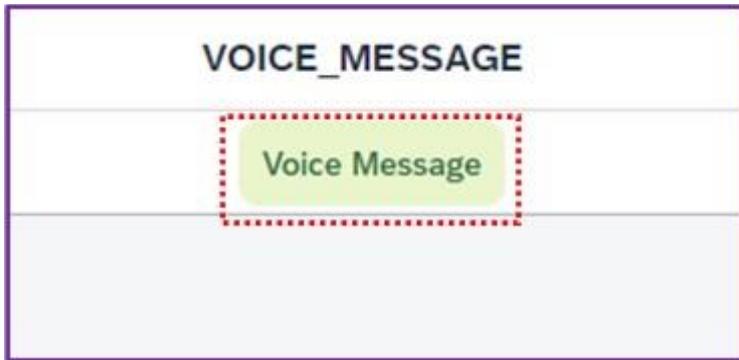
IO_VIEW = /U4A/IF_SERVER~AR_VIEW

I_MESSAGE = LV_MSG.

" 메시지 내용

ENDMETHOD.

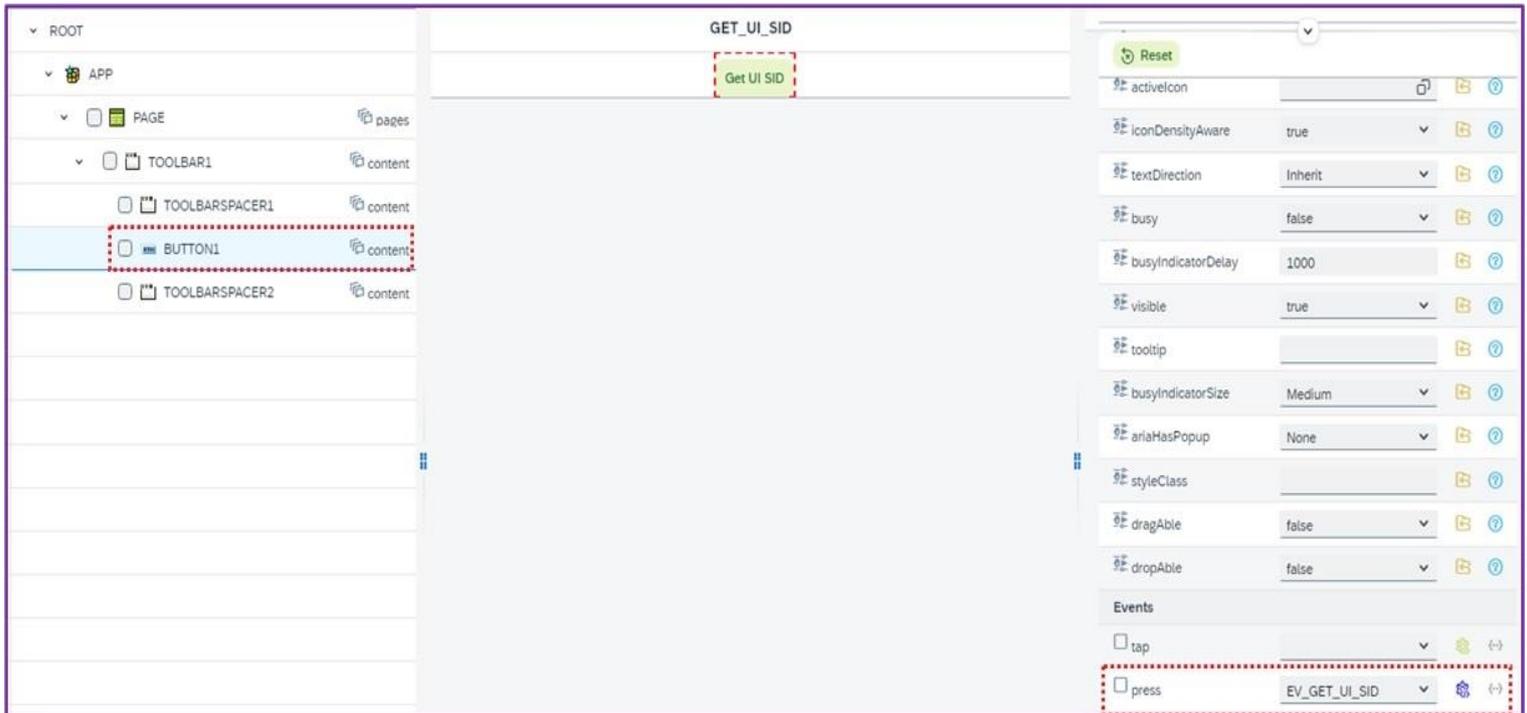
3. Application 실행한 후 '음성 메시지 출력' Button Click 하여 음성이 들리는지 확인.



6.1.13 Methodes GET_UI_SID

▣ 목적 : 특정 UI의 System ID를 가져오기 위해서 사용.

1. "GET_UI_SID " Method 동작할 Event를 특정 UI의 Events에 추가 (본 단원에서는 BUTTON Click 하여 동작).



2. " GET_UI_SID" Method.

[첨부 ABAP SOURCE (EV_GET_UI_SID)]

```
METHOD EV_GET_UI_SID.
```

```
DATA: LV_SID TYPE STRING.
```

```
" Selected UI의 SID 가져오기.
```

```
CALL METHOD /U4A/CL_UTILITIES=>GET_UI_SID
```

EXPORTING

IT_FORM_DATA = IT_FORM_DATA " HTTP Framework (iHTTP) Table Name/Value Pairs

IMPORTING

E_SID = LV_SID. " RETURN SID

" 가져온 SID 메시지로 출력 .

CALL METHOD /U4A/CL_UTILITIES=>M_MESSAGE_TOAST

EXPORTING

IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW

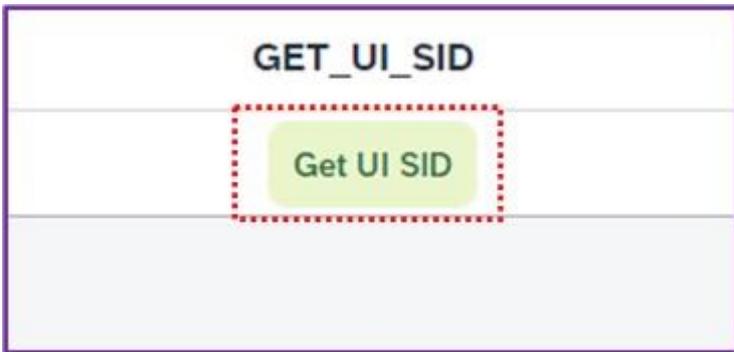
I_MSGTX = LV_SID " Structure of message variable

I_VPOS = /U4A/CL_UTILITIES=>CS_M_MSG_TOS_VPOS-CENTER " Vertical position

I_HPOS = /U4A/CL_UTILITIES=>CS_M_MSG_TOS_HPOS-CENTER. " Horizontal position

ENDMETHOD.

3. Application 실행한 후 UI SID 가져오기 Button Click .



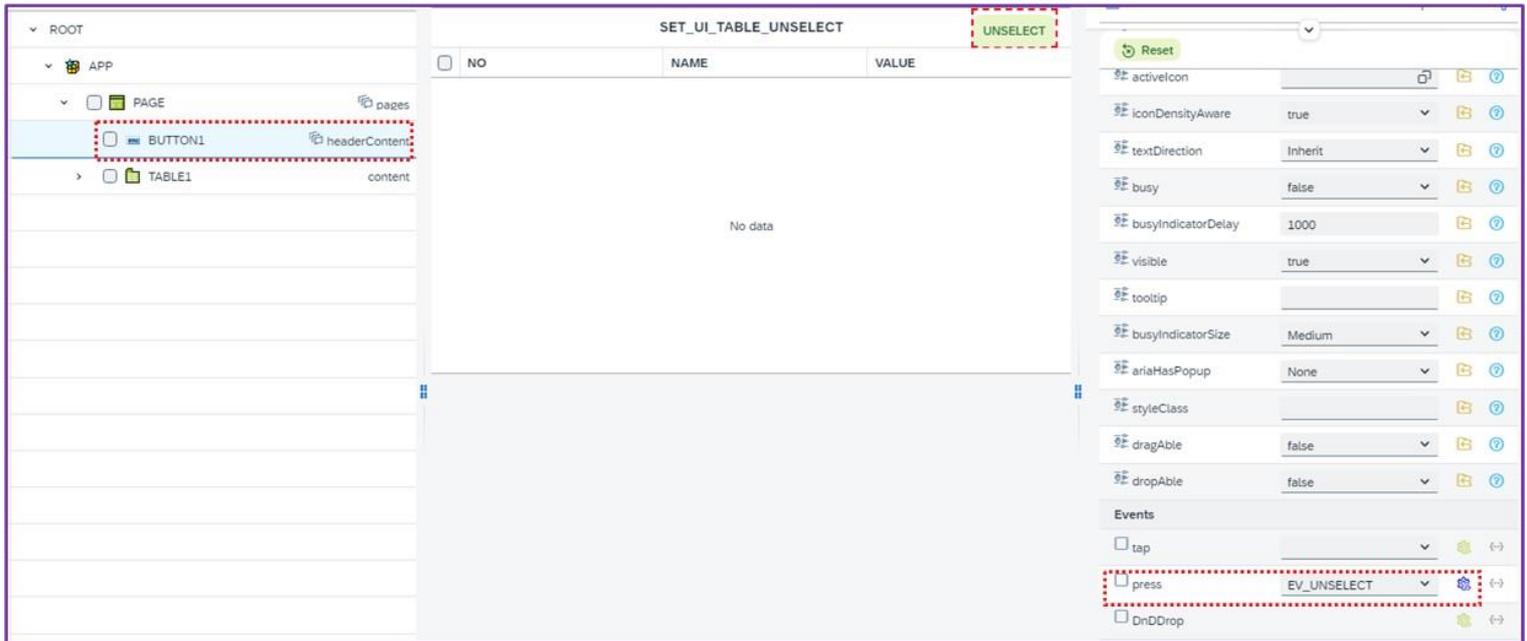
4. Click한 Button의 SID 출력 .



6.1.14 Methodes SET_UI_TABLE_UNSELECT

■ 목적 : Ui.Table에서 Select 상태의 Checkbox를 모두 UnSelect하기 위해 사용.

1. "SET_UI_TABLE_UNSELECT" Method 동작할 Event를 특정 UI의 Events에 추가 (본 단원에서는 BUTTON Click 하여 동작).



2. SET_UI_TABLE_UNSELECT Method

[첨부 ABAP SOURCE (EV_UNSELECT)]

METHOD EV_UNSELECT.

" SET_UI_TABLE_UNSELECT .

CALL METHOD /U4A/CL_UTILITIES=>SET_UI_TABLE_UNSELECT

EXPORTING

IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW

I_UIID = 'TABLE1' .

" BINDING DATA TABLE.

ENDMETHOD.

3. Application 실행 후 Row 여러 건 Select 후 Button Click.

SET_UI_TABLE_UNSELECT				UNSELECT
<input type="checkbox"/>	NO	NAME	VALUE	
<input type="checkbox"/>	1	COLUMN1	100	
<input checked="" type="checkbox"/>	2	COLUMN2	200	
<input type="checkbox"/>	3	COLUMN3	300	
<input checked="" type="checkbox"/>	4	COLUMN4	400	
<input type="checkbox"/>	5	COLUMN5	500	
<input checked="" type="checkbox"/>	6	COLUMN6	600	
<input checked="" type="checkbox"/>	7	COLUMN7	700	
<input checked="" type="checkbox"/>	8	COLUMN8	800	
<input type="checkbox"/>	9	COLUMN9	900	
<input type="checkbox"/>	10	COLUMN10	1000	

4. Check 되어 있던 Row가 모두 UnSelected 되는 것을 확인.

SET_UI_TABLE_UNSELECT				UNSELECT
<input type="checkbox"/>	NO	NAME	VALUE	
<input type="checkbox"/>	1	COLUMN1	100	
<input type="checkbox"/>	2	COLUMN2	200	
<input type="checkbox"/>	3	COLUMN3	300	
<input type="checkbox"/>	4	COLUMN4	400	
<input type="checkbox"/>	5	COLUMN5	500	
<input type="checkbox"/>	6	COLUMN6	600	
<input type="checkbox"/>	7	COLUMN7	700	
<input type="checkbox"/>	8	COLUMN8	800	
<input type="checkbox"/>	9	COLUMN9	900	
<input type="checkbox"/>	10	COLUMN10	1000	

 Table Type은 반드시 UI.Table을 사용 (M.Table은 허용 하지 않음).

6.1.15

Methodes

SET_UI_TABLE_FIRST_VISIBLE_ROW

■ 목적 : Ui.Table 조회 시 Table 최상단에 보일 Row를 지정 하기 위해서 사용.

1. "SET_UI_TABLE_FIRST_VISIBLE_ROW" Method 동작할 Event를 특정 UI의 Events에 추가 (본 단원에서는 BUTTON Click 하여 동작).



2. " SET_UI_TABLE_FIRST_VISIBLE_ROW" Method (가장 먼저 보일 Table Row 설정 가능).

[첨부 ABAP SOURCE (EV_OPENDIALOG)]

METHOD EV_OPENDIALOG.

```

" DIALOG OPEN .
CALL METHOD /U4A/CL_UTILITIES=>DIALOG_OPEN
EXPORTING
  IO_VIEW      = ME->/U4A/IF_SERVER~AR_VIEW
  I_DIALOG_ID = 'DIALOG1'.
    
```

" SET : FIRST VISIBLE TABLE ROW.

CALL METHOD /U4A/CL_UTILITIES=>SET_UI_TABLE_FIRST_VISIBLE_ROW

EXPORTING

IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW

I_TABNM = 'TABLE1'

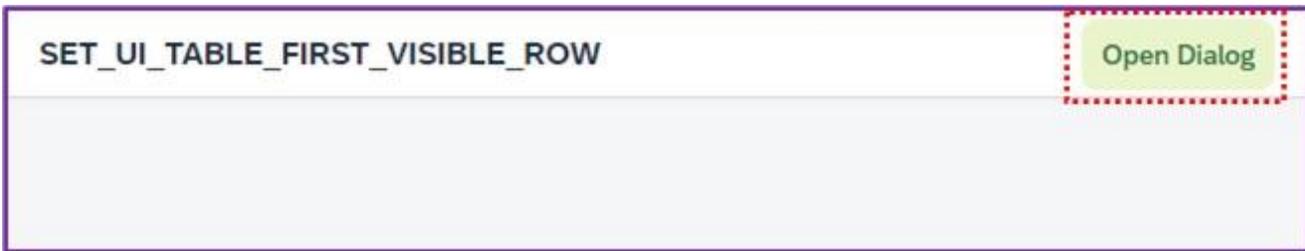
" table UI name

I_ROW_INDEX = 22.

" table Row Index (Start Index 0!)

ENDMETHOD.

3. Application 실행한 후 " Open Dialog" Button Click.



4. Dialog Open 시 Table 가장 위에 'I_ROW_INDEX' 에 해당하는 Row가 보이는 것을 확인 (Index 는 0부터 시작하기 때문에 Index 22에 해당하는 Row의 NO는 23).

<input type="checkbox"/>	NO	NAME	VALUE
<input type="checkbox"/>	23	COLUMN23	2300
<input type="checkbox"/>	24	COLUMN24	2400
<input type="checkbox"/>	25	COLUMN25	2500
<input type="checkbox"/>	26	COLUMN26	2600
<input type="checkbox"/>	27	COLUMN27	2700
<input type="checkbox"/>	28	COLUMN28	2800
<input type="checkbox"/>	29	COLUMN29	2900
<input type="checkbox"/>	30	COLUMN30	3000
<input type="checkbox"/>	31	COLUMN31	3100
<input type="checkbox"/>	32	COLUMN32	3200

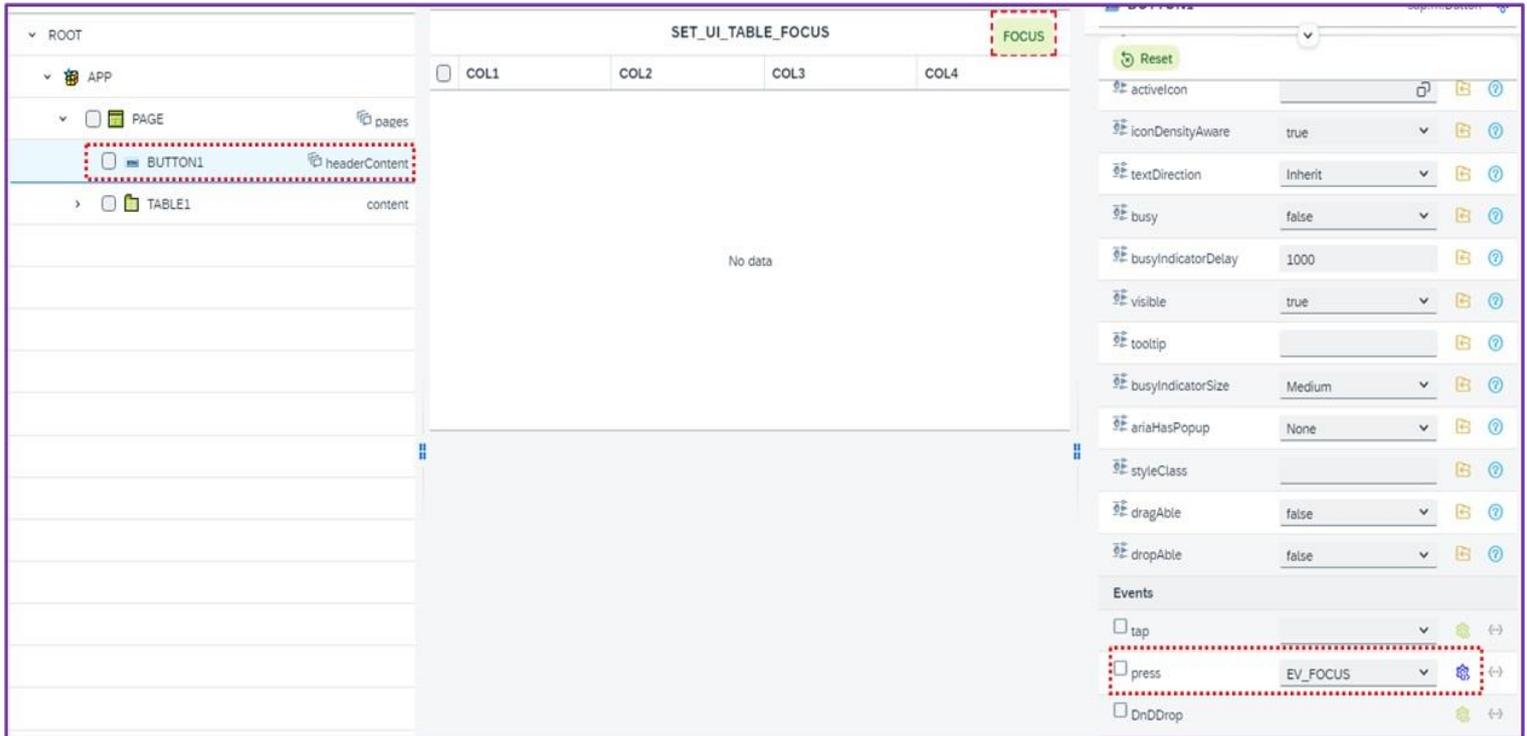
The table above shows a list of rows with columns for selection (checkbox), NO, NAME, and VALUE. The row with NO 23 is highlighted with a red dashed border, indicating it is the first row visible in the dialog.

 Table Type은 반드시 UI.Table을 사용 (M.Table은 허용 하지 않음).

6.1.16 Methodes SET_UI_TABLE_FOCUS

■ 목적 : Ui_Table 에서 특정 Data 를 선택 하기 위해 Focus 를 주기 위한 용도로 사용 .

1. "SET_UI_TABLE_FOCUS" Method 동작할 Event를 특정 UI의 Events에 추가 (본 단원에서는 BUTTON Click 하여 동작).



2. "SET_UI_TABLE_FOCUS" Method .

[첨부 ABAP SOURCE (EV_FOCUS)]

METHOD EV_FOCUS.

```
" UI_TABLE_FOCUS .
CALL METHOD /U4A/CL_UTILITIES=>SET_UI_TABLE_FOCUS
EXPORTING
  IO_VIEW          = ME->/U4A/IF_SERVER~AR_VIEW
```

I_TABNM = 'TABLE1'

* I_FOCUS_UINM = " Do not Use!(Deprecated)

I_BIND_FIELD_NAME = 'F02' " Filed 이름

I_INDEX = 10 " Row Index

I_WAIT = 100. " setTimeout 시간

ENDMETHOD.

PARAMETERS	MEAN
I_TABNM	FOCUS 하려는 대상이 BINDING 되어있는 UI TABLE 이름
I_FOCUS_UINM	폐기 처리
I_BIND_FIELD_NAME	FOCUS 하려는 대상의 FILED 이름
I_INDEX	FOCUS 하려는 대상의 ROW INDEX
I_WAIT	지정한 시간 뒤에 FOCUS 동작 (단위 : MS)

3. Application 실행한 후 FOCUS Button Click.

SET_UI_TABLE_FOCUS				FOCUS
<input type="checkbox"/>	COL1	COL2	COL3	COL4
<input type="checkbox"/>	COL1_1	COL2_1	COL3_1	COL4_1
<input type="checkbox"/>	COL1_2	COL2_2	COL3_2	COL4_2
<input type="checkbox"/>	COL1_3	COL2_3	COL3_3	COL4_3
<input type="checkbox"/>	COL1_4	COL2_4	COL3_4	COL4_4
<input type="checkbox"/>	COL1_5	COL2_5	COL3_5	COL4_5
<input type="checkbox"/>	COL1_6	COL2_6	COL3_6	COL4_6
<input type="checkbox"/>	COL1_7	COL2_7	COL3_7	COL4_7
<input type="checkbox"/>	COL1_8	COL2_8	COL3_8	COL4_8
<input type="checkbox"/>	COL1_9	COL2_9	COL3_9	COL4_9
<input type="checkbox"/>	COL1_10	COL2_10	COL3_10	COL4_10

4. Method에 설정한 대로 F02 Field의 10번째 Index에 해당하는 Field가 Focus 된 것을 확인 .

SET_UI_TABLE_FOCUS				FOCUS
<input type="checkbox"/>	COL1	COL2	COL3	COL4
<input type="checkbox"/>	COL1_10	COL2_10	COL3_10	COL4_10
<input type="checkbox"/>	COL1_11	COL2_11	COL3_11	COL4_11
<input type="checkbox"/>	COL1_12	COL2_12	COL3_12	COL4_12
<input type="checkbox"/>	COL1_13	COL2_13	COL3_13	COL4_13
<input type="checkbox"/>	COL1_14	COL2_14	COL3_14	COL4_14
<input type="checkbox"/>	COL1_15	COL2_15	COL3_15	COL4_15
<input type="checkbox"/>	COL1_16	COL2_16	COL3_16	COL4_16
<input type="checkbox"/>	COL1_17	COL2_17	COL3_17	COL4_17
<input type="checkbox"/>	COL1_18	COL2_18	COL3_18	COL4_18
<input type="checkbox"/>	COL1_19	COL2_19	COL3_19	COL4_19

 Table Type은 반드시 UI.Table을 사용 (M.Table은 허용 하지 않음).

6.1.17 Methodes RESET_UITABLE_SORT

■ 목적 : Ui.Table에 적용되어 있는 모든 Sort Option을 해제 하기 위해서 사용.

1. Table을 정렬할 수 있도록 Column의 "Property"에서 "sorted"에 "True", "sortProperty"에 Column 의 "필드명"을 설정.

"RESET_UITABLE_SORT" Method 동작할 Event를 특정 UI의 Events에 추가 (본 단원에서는 BUTTON Click 하여 동작).

The screenshot shows the SAP IDE interface. On the left, the 'Properties' panel for 'COLUMN1' is visible, with the 'sorted' property set to 'true' and 'sortProperty' set to 'F01'. The 'preview' view shows a table with columns COL1 through COL7. A 'RESET' button is located in the top right corner of the table. The 'COLUMN1' property panel is highlighted with a red dashed box, and the 'sorted' and 'sortProperty' properties are also highlighted with a red dashed box.

2. "RESET_UITABLE_SORT" Method .

[첨부 ABAP SOURCE (EV_RESET_SORT)]

METHOD EV_RESET_SORT.

"UI Table Sort Reset

CALL METHOD /U4A/CL_UTILITIES=>RESET_UTABLE_SORT

EXPORTING

IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class

I_TABNM = 'TABLE1'. " UI TABLE ID

ENDMETHOD.

3. Application 실행 후 Data를 아래와 같이 "Descending" Sort 한 후 RESET Button Click.

COL1	IF	COL2	COL3	COL4	COL5	COL6	COL7
10		10	10	10	10	10	10
9		9	9	9	9	9	9
8		8	8	8	8	8	8
7		7	7	7	7	7	7
6		6	6	6	6	6	6
5		5	5	5	5	5	5
4		4	4	4	4	4	4
3		3	3	3	3	3	3
2		2	2	2	2	2	2
1		1	1	1	1	1	1

RESET

4. Defalt Sort Order로 설정했던 Ascending Sort로 Reset 된 것을 확인.

COL1	COL2	COL3	COL4	COL5	COL6	COL7
1	1	1	1	1	1	1
2	2	2	2	2	2	2
3	3	3	3	3	3	3
4	4	4	4	4	4	4
5	5	5	5	5	5	5
6	6	6	6	6	6	6
7	7	7	7	7	7	7
8	8	8	8	8	8	8
9	9	9	9	9	9	9
10	10	10	10	10	10	10

RESET

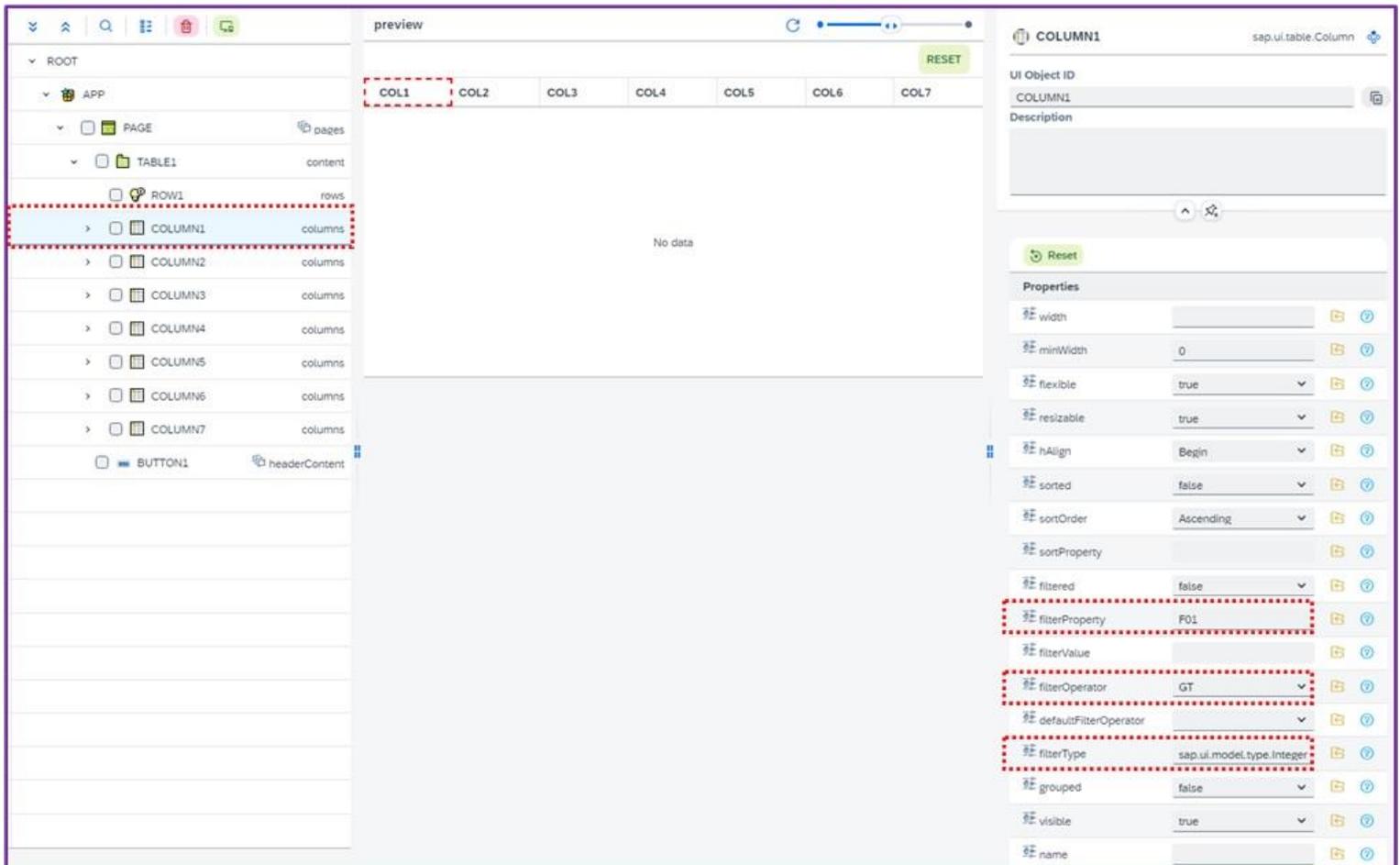
6.1.18 Methodes RESET_UITABLE_FILTER

■ 목적 : Ui.Table에 적용되어 있는 모든 Filter Option을 해제 하기 위해서 사용.

1. Table 을 Filtering 할 수 있도록 "Column"의 "Property"에서 "filterProperty"에 해당 Column의 "필드명"을 설정, "filterOperator"에 원하는 Option 선택.

"RESET_UITABLE_FILTER " Method 동작할 Event를 특정 UI의 Events에 추가 (본 단원에서는 BUTTON Click하여 동작).

※ 숫자타입의 필드를 Filter 하는 경우 Properties = > "filterType"에 "sap.ui.model.type.Integer" 를 입력해주어야 한다



2. " RESET_UITABLE_FILTER" Method .

[첨부 ABAP SOURCE (EV_RESET_FILTER)]

METHOD EV_RESET_FILTER.

"UI Table Filter Reset

CALL METHOD /U4A/CL_UTILITIES=>RESET_UTABLE_FILTER

EXPORTING

IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class

I_TABNM = 'TABLE1'. " UI TABLE ID

ENDMETHOD.

3. Application 실행 후 필터를 설정할 Column을 선택하여 Data를 원하는 값으로 Filtering 후 "RESET" Button 클릭 (본 단원에서는 Greater 5로 설정).

COL1	▼ COL2	COL3	COL4	COL5	COL6	COL7
6	6	6	6	6	6	6
7	7	7	7	7	7	7
8	8	8	8	8	8	8
9	9	9	9	9	9	9
10	10	10	10	10	10	10

RESET

4. Filtering이 해제 되어 모든 값이 출력된 것을 확인.

U4A IDE HELP

							RESET
COL1	COL2	COL3	COL4	COL5	COL6	COL7	
1	1	1	1	1	1	1	
2	2	2	2	2	2	2	
3	3	3	3	3	3	3	
4	4	4	4	4	4	4	
5	5	5	5	5	5	5	
6	6	6	6	6	6	6	
7	7	7	7	7	7	7	
8	8	8	8	8	8	8	
9	9	9	9	9	9	9	
10	10	10	10	10	10	10	

6.1.19

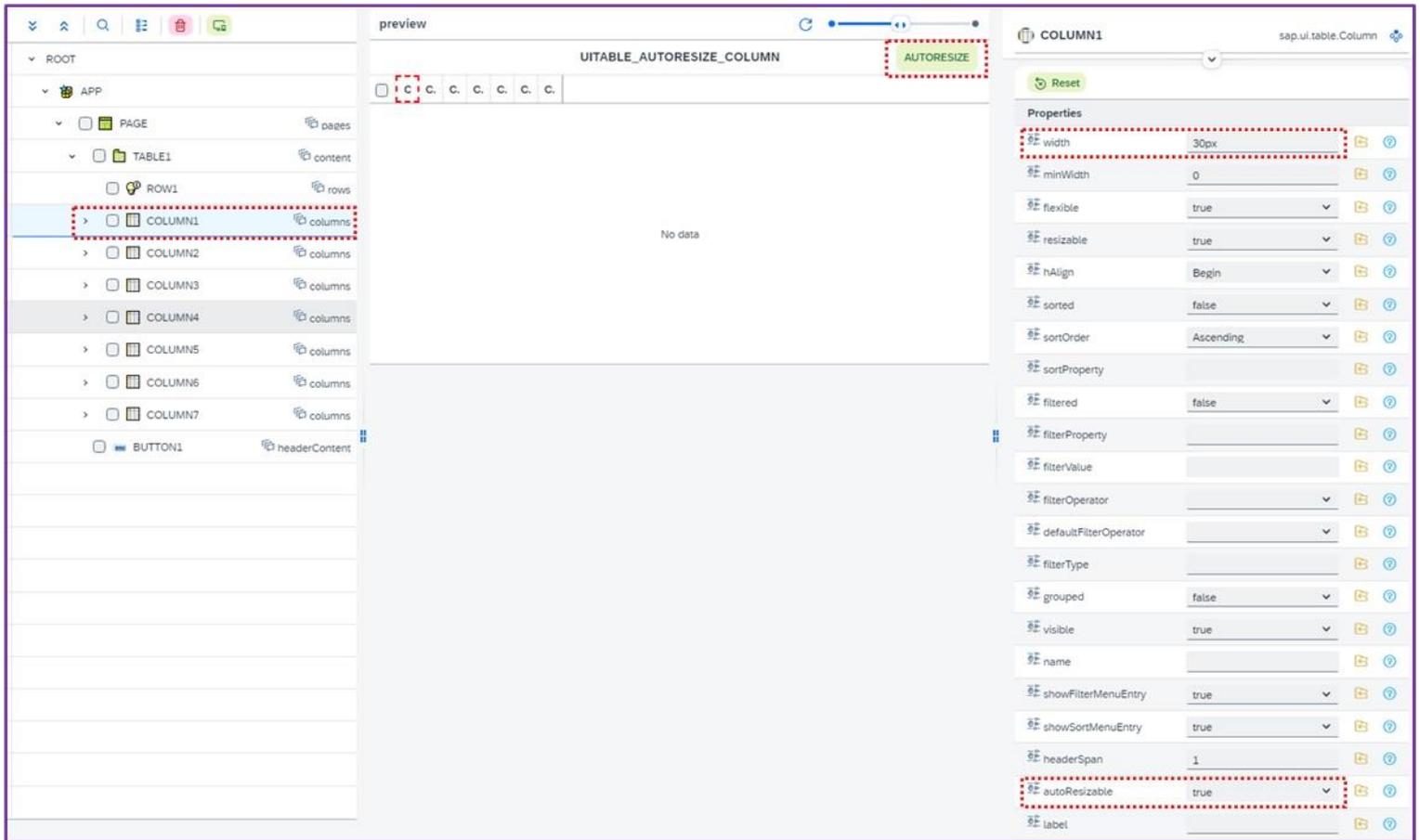
Methodes

UITABLE_AUTORESIZES_COLUMN

■ 목적 : Ui.Table에 Column Width를 해당 Column의 Data Length에 맞게 조정하기 위한 용도.

1. "UITABLE_AUTORESIZES_COLUMN" Method 동작할 Event를 특정 UI의 Events에 추가 (본 단원에서서는 Button Click 하여 동작).

변경하려는 "Column"의 "Property"에서 "autoResizable"을 반드시 "TRUE"로 설정해야 Method가 적용



2. "UITABLE_AUTORESIZES_COLUMN" Method (Text 길이에 맞게 Column 길이 재설정).

[첨부 ABAP SOURCE (EV_AUTORESIZES_COLUMN)]

METHOD EV_AUTORESIZE_COLUMN.

" UI TABLE COLUMN 길이 TEXT LENGTH에 맞게 재설정 하기 .

CALL METHOD /U4A/CL_UTILITIES=>UITABLE_AUTORESIZE_COLUMN

EXPORTING

IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW

I_TABNM = 'TABLE1'. " UI TABLE 이름

* I_TIMER = " Do not Use!(Deprecated)

ENDMETHOD.

3. Application 실행한 후 AUTORESIZE Button Click.



4. Column의 길이가 Text 길이에 맞게 자동으로 조정되는 것을 확인.

U4A IDE HELP

UITABLE_AUTORESIZABLE_COLUMN								AUTORESIZABLE
<input type="checkbox"/>	COL1	COL2	COL3	COL4	COL5	COL6	COL7	
<input type="checkbox"/>	TEST_DATA_1							
<input type="checkbox"/>	TEST_DATA_2							
<input type="checkbox"/>	TEST_DATA_3							
<input type="checkbox"/>	TEST_DATA_4							
<input type="checkbox"/>	TEST_DATA_5							
<input type="checkbox"/>	TEST_DATA_6							
<input type="checkbox"/>	TEST_DATA_7							
<input type="checkbox"/>	TEST_DATA_8							
<input type="checkbox"/>	TEST_DATA_9							
<input type="checkbox"/>	TEST_DATA_10							

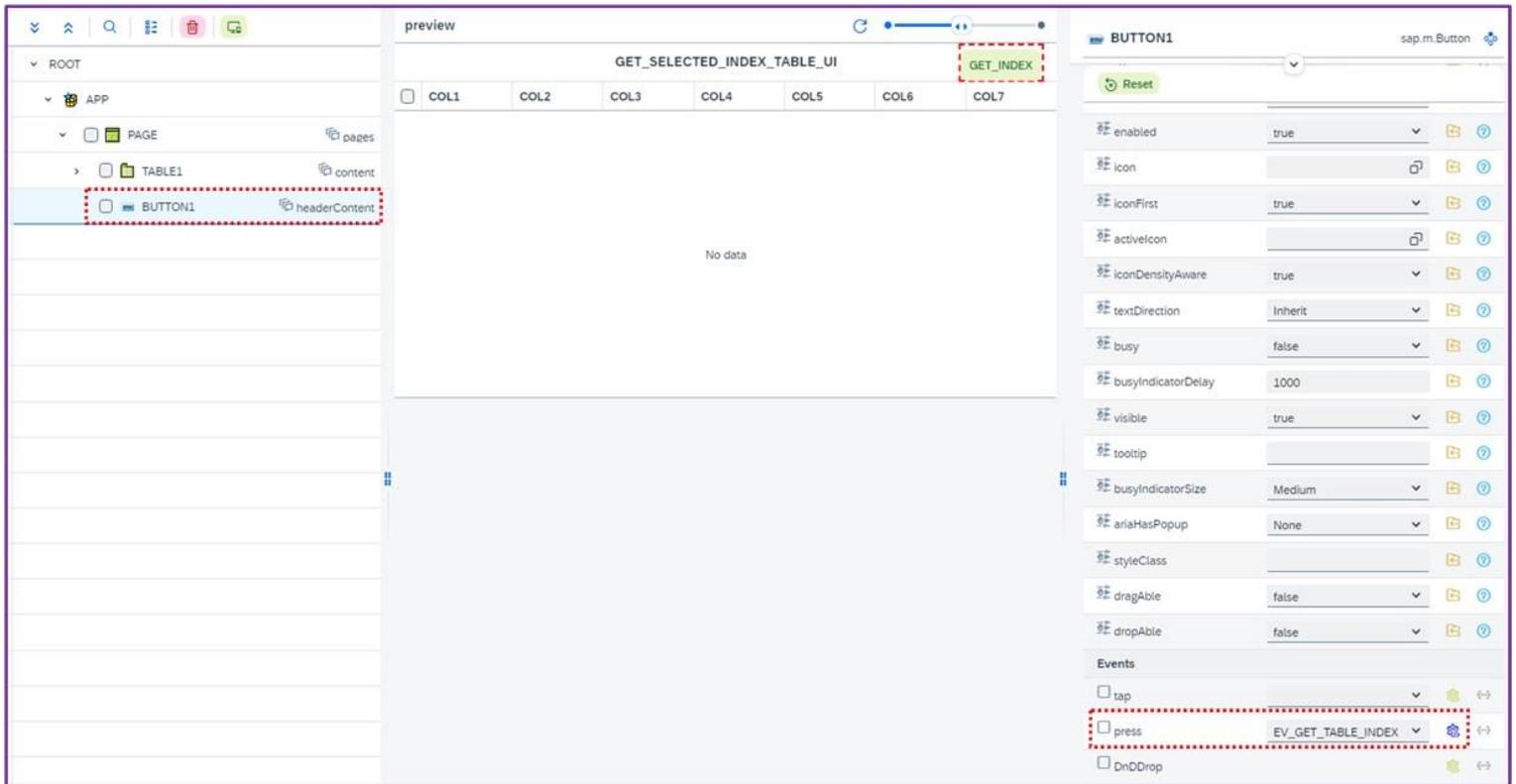
6.1.20

Methodes

GET_SELECTED_INDEX_TABLE_UI

■ 목적 : Table 내에서 복수 Row를 선택하여 다수의 Index 값을 가져오기 위해서 사용.

1. 복수의 Row를 선택할 수 있는 Table UI를 추가하고 "GET_SELECTED_INDEX_TABLE_UI" Method를 동작할 Event를 UI에 추가 (본 단원에서는 BUTTON 클릭 시 동작).



2. "GET_SELECTED_INDEX_TABLE_UI" (Table에서 복수 Row 선택건에 대한 Index 가져옴, 복수 INDEX이므로 INDEX 값을 INT4_TABLE TYPE으로 받음).

[첨부 ABAP SOURCE (EV_GET_TABLE_INDEX)]

METHOD EV_GET_TABLE_INDEX.

DATA: LT_INDEX TYPE INT4_TABLE.

* -----

" UI TABLE 선택한 복수 INDEX 가져오기.

CALL METHOD /U4A/CL_UTILITIES=>GET_SELECTED_INDEX_TABLE_UI

EXPORTING

IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW

I_TABNM = 'TABLE1' " TABLE UI NAME

IMPORTING

ET_INDEX = LT_INDEX. " 선택 라인 INDEX 정보

ENDMETHOD.

3. Application 실행 후 복수 Row 선택 후 GET_INDEX Button Click (본 단원에서는 홀수 Row 5건 선택) .

GET_SELECTED_INDEX_TABLE_UI							GET_INDEX
<input type="checkbox"/>	COL1	COL2	COL3	COL4	COL5	COL6	COL7
<input checked="" type="checkbox"/>	1	1	1	1	1	1	1
<input type="checkbox"/>	2	2	2	2	2	2	2
<input checked="" type="checkbox"/>	3	3	3	3	3	3	3
<input type="checkbox"/>	4	4	4	4	4	4	4
<input checked="" type="checkbox"/>	5	5	5	5	5	5	5
<input type="checkbox"/>	6	6	6	6	6	6	6
<input checked="" type="checkbox"/>	7	7	7	7	7	7	7
<input type="checkbox"/>	8	8	8	8	8	8	8
<input checked="" type="checkbox"/>	9	9	9	9	9	9	9
<input type="checkbox"/>	10	10	10	10	10	10	10

4. UI Table에서 선택한 5건의 Row Index 정보가 INT4_TABLE 구조로 담겨 있는 것을 확인.

Tables Table Contents

Table 

Attributes  

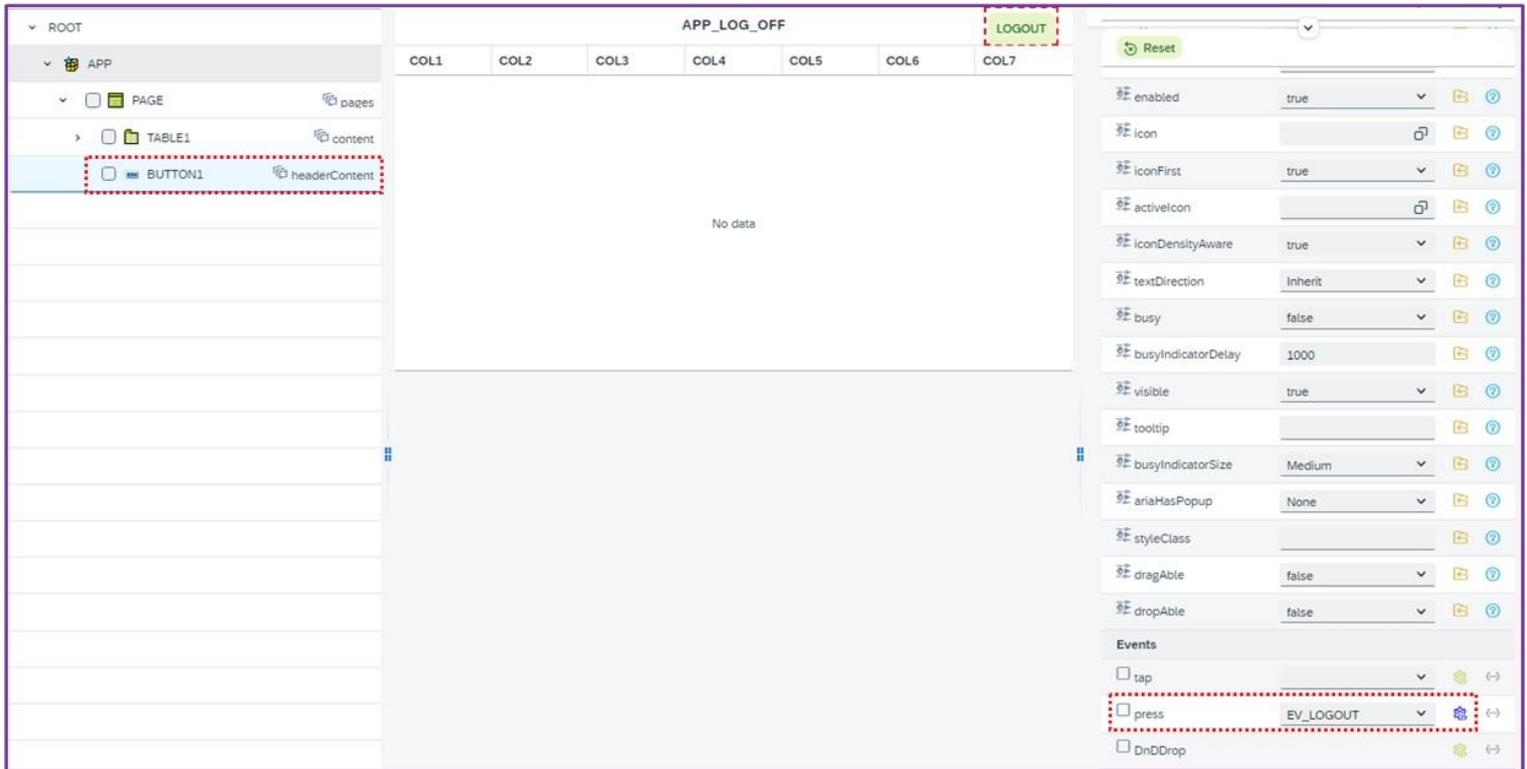
Insert Column  Columns ... 

Row	TABLE_LINE [I(4)]
1	1
2	3
3	5
4	7
5	9

6.1.21 Methodes APP_LOG_OFF

■ 목적 : Application을 실행하여 Login 되어 있을 때 Logout 하기 위한 용도.

1. "APP_LOG_OFF" Method를 동작할 Event를 UI에 추가 (본 단원에서는 BUTTON Click 하여 동작).



2. "APP_LOG_OFF" Method.

[첨부 ABAP SOURCE (EV_LOGOUT)]

```
METHOD EV_LOGOUT.
```

```
"APPLICATION LOG OFF.
```

```
CALL METHOD /U4A/CL_UTILITIES=>APP_LOG_OFF
EXPORTING
```

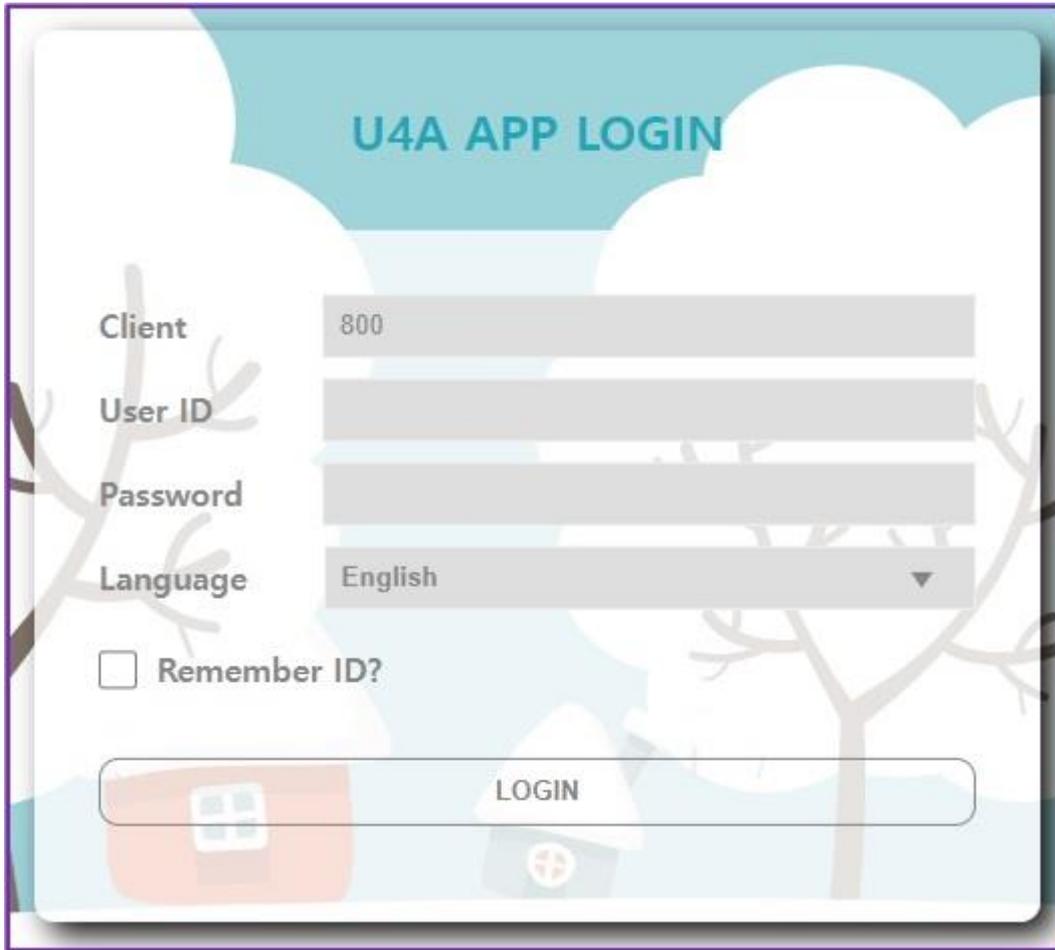
IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW
 I_REDIRECT_URL = 'https://www.google.com/' " 로그아웃 후 이동할 페이지 URL
 I_WINCLOSE = 'X'.

ENDMETHOD.

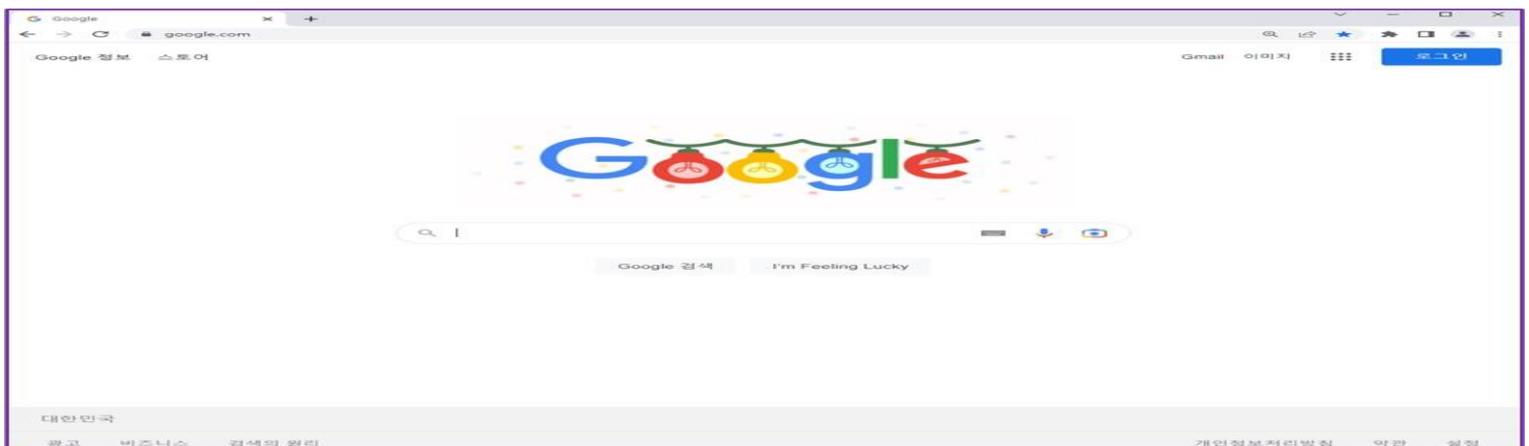
3. APPLICATION 실행 후 LOGOUT Button Click하여 Application LogOut.

APP_LOG_OFF							LOGOUT
COL1	COL2	COL3	COL4	COL5	COL6	COL7	
1	1	1	1	1	1	1	
2	2	2	2	2	2	2	
3	3	3	3	3	3	3	
4	4	4	4	4	4	4	
5	5	5	5	5	5	5	
6	6	6	6	6	6	6	
7	7	7	7	7	7	7	
8	8	8	8	8	8	8	
9	9	9	9	9	9	9	
10	10	10	10	10	10	10	

4. I_REDIRECT_URL = ' ', I_WINCLOSE = ' ' 인 경우 LOGOUT (처음 Login 화면으로 이동함).

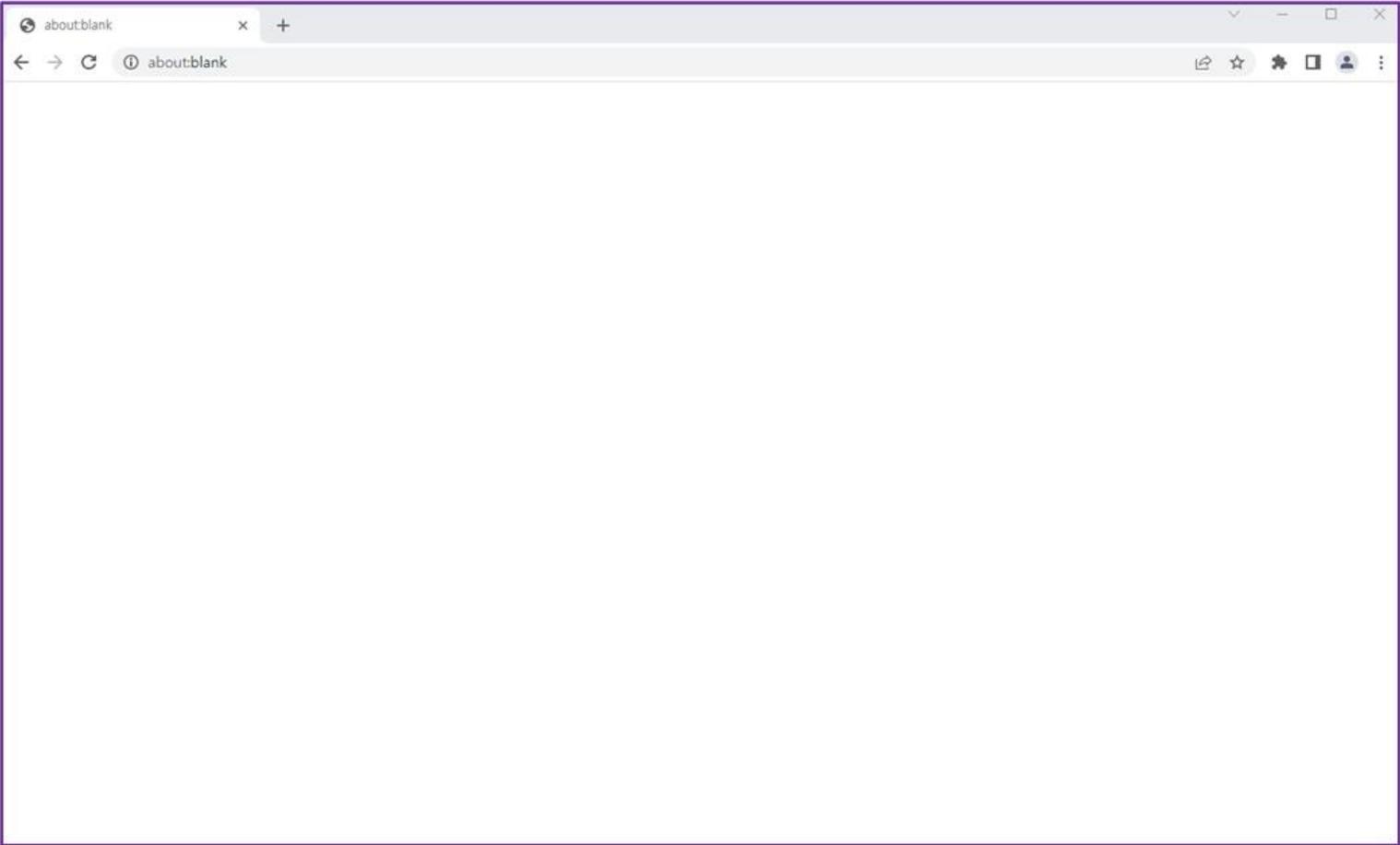


5. I_REDIRECT_URL = ' URL 주소' 인 경우 LOGOUT (입력한 URL 주소에 해당하는 사이트로 이동, 본 단원에서는 GOOGLE URL 사용).



6. I_WINCLOSE = 'X' 인 경우 LOGOUT (WINDOW 종료 화면으로 이동).

U4A IDE HELP



6.1.22 Methodes RESET_USAGE_APP

▣ 목적 : "Application Rendering" Method가 동작하기 전에 App_Container에 연결되어 있는 Rendering 정보를 Reset 하기 위해 사용 .

1. Application Rendering을 할 때 Reset이 필요하다면 " Application Rendering" Method 이전에 호출 .

2. "RESET_USAGE_APP" Method.

[첨부 ABAP SOURCE (EV_RENDERING_USAGE_APP)]

```
METHOD EV_RENDERING_USAGE_APP.
```

```
" Application Rendering 하기 전 AppContainer Reset .
```

```
CALL METHOD /U4A/CL_UTILITIES=>RESET_USAGE_APP
```

```
EXPORTING
```

```
IO_VIEW      = ME->/U4A/IF_SERVER~AR_VIEW      " [U4A] UI Element Super Class
```

```
IV_APP_NM    = 'APPCONTAIN1'.                " APP_CONTAINER_ID
```

```
CASE I_ID.
```

```
  WHEN 'BUTTON1'.
```

```
" Application Rendering .
```

```
CALL METHOD /U4A/CL_UTILITIES=>RENDERING_USAGE_APP
```

```
EXPORTING
```

```
IO_VIEW      = ME->/U4A/IF_SERVER~AR_VIEW
```

```
I_APP_CONTAINER_ID = 'APPCONTAIN1'
```

```
I_APPID      = 'YUTIL_TEST004'                " Application ID
```

```
* I_RESET    =                                " Do not Use!(Deprecated)
```

```
I_WAIT      = 1000.
```

```
WHEN 'BUTTON2' .
```

```
" Application Rendering .
```

```
CALL METHOD /U4A/CL_UTILITIES=>RENDERING_USAGE_APP
```

```
EXPORTING
```

```
IO_VIEW          = ME->/U4A/IF_SERVER~AR_VIEW
```

```
I_APP_CONTAINER_ID = 'APPCONTAIN1'
```

```
I_APPID          = 'YUTIL_TEST005'
```

```
" Application ID
```

```
* I_RESET          = " Do not Use!(Deprecated)
```

```
I_WAIT          = 1000.
```

```
ENDCASE.
```

```
ENDMETHOD.
```



반드시 RESET_USAGE_APP Method의 IV_APP_NM과 RENDERING_USAGE_APP Method의 I_APP_CONTAINER_ID가 동일해야 RESET이 정상적으로 동작.

6.1.23 Methodes MESSAGE_STRIP

■ 목적 : 다수의 Message를 한번에 보여주기 위해서 호출하기 위해서 사용.

1. "MESSAGE_STRIP" Method를 동작할 Event를 특정 UI의 Events에 추가 (본 단원에서는 BUTTON Click 하여 동작).



2. "MESSAGE_STRIP" Method.

[첨부 ABAP SOURCE (EV_MESSAGE_STRIP)]

METHOD EV_MESSAGE_STRIP.

DATA: LS_MSG_INFO TYPE /U4A/CL_UTILITIES=>GTY_MSGSTRIP.

DATA: LT_MSG_INFO TYPE /U4A/CL_UTILITIES=>GTT_MSGSTRIP.

* -----

```
" MSG_INFO 데이터 생성 .
```

```
LS_MSG_INFO-TYPE = 'E'.
LS_MSG_INFO-TEXT = 'Error Message'.
```

```
APPEND LS_MSG_INFO TO LT_MSG_INFO.
CLEAR LS_MSG_INFO.
```

```
LS_MSG_INFO-TYPE = 'W'.
LS_MSG_INFO-TEXT = 'Warning Message'.
```

```
APPEND LS_MSG_INFO TO LT_MSG_INFO.
CLEAR LS_MSG_INFO.
```

```
LS_MSG_INFO-TYPE = 'I'.
LS_MSG_INFO-TEXT = 'Information Message'.
```

```
APPEND LS_MSG_INFO TO LT_MSG_INFO.
CLEAR LS_MSG_INFO.
```

```
LS_MSG_INFO-TYPE = 'S'.
LS_MSG_INFO-TEXT = 'Success Message'.
```

```
APPEND LS_MSG_INFO TO LT_MSG_INFO.
CLEAR LS_MSG_INFO.
```

```
" MESSAGE_STRIP
```

```
CALL METHOD /U4A/CL_UTILITIES=>MESSAGE_STRIP
EXPORTING
```

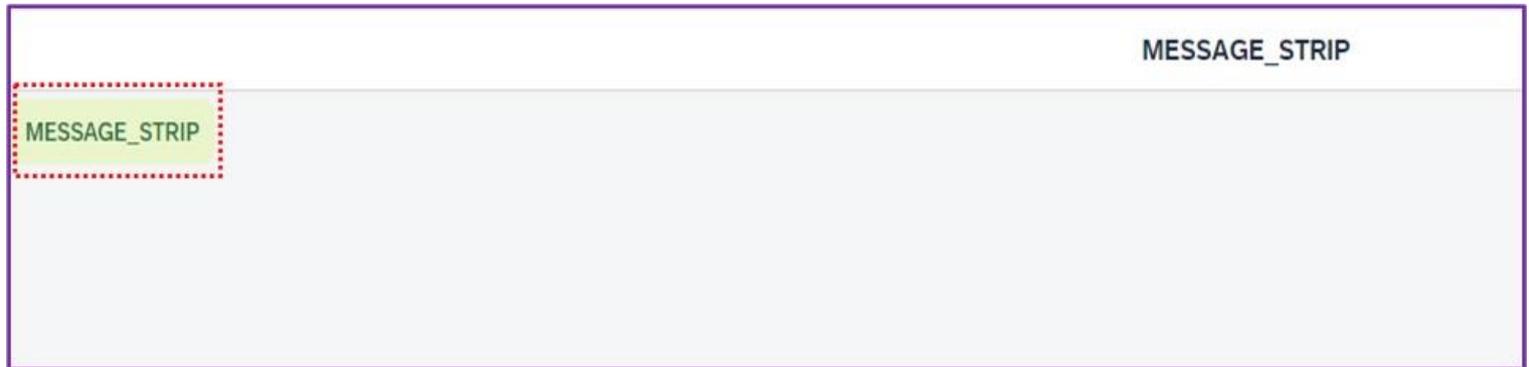
```
IO_VIEW      = ME->/U4A/IF_SERVER~AR_VIEW
I_TITLE      = 'Title'           " 타이틀
I_WIDTH      = '500PX'          " 넓이 SIZE
I_HEIGHT     = '500PX'          " 높이 SIZE
IT_MSG_INFO  = LT_MSG_INFO.     " 처리 메세지 정보탭
```

```
ENDMETHOD.
```

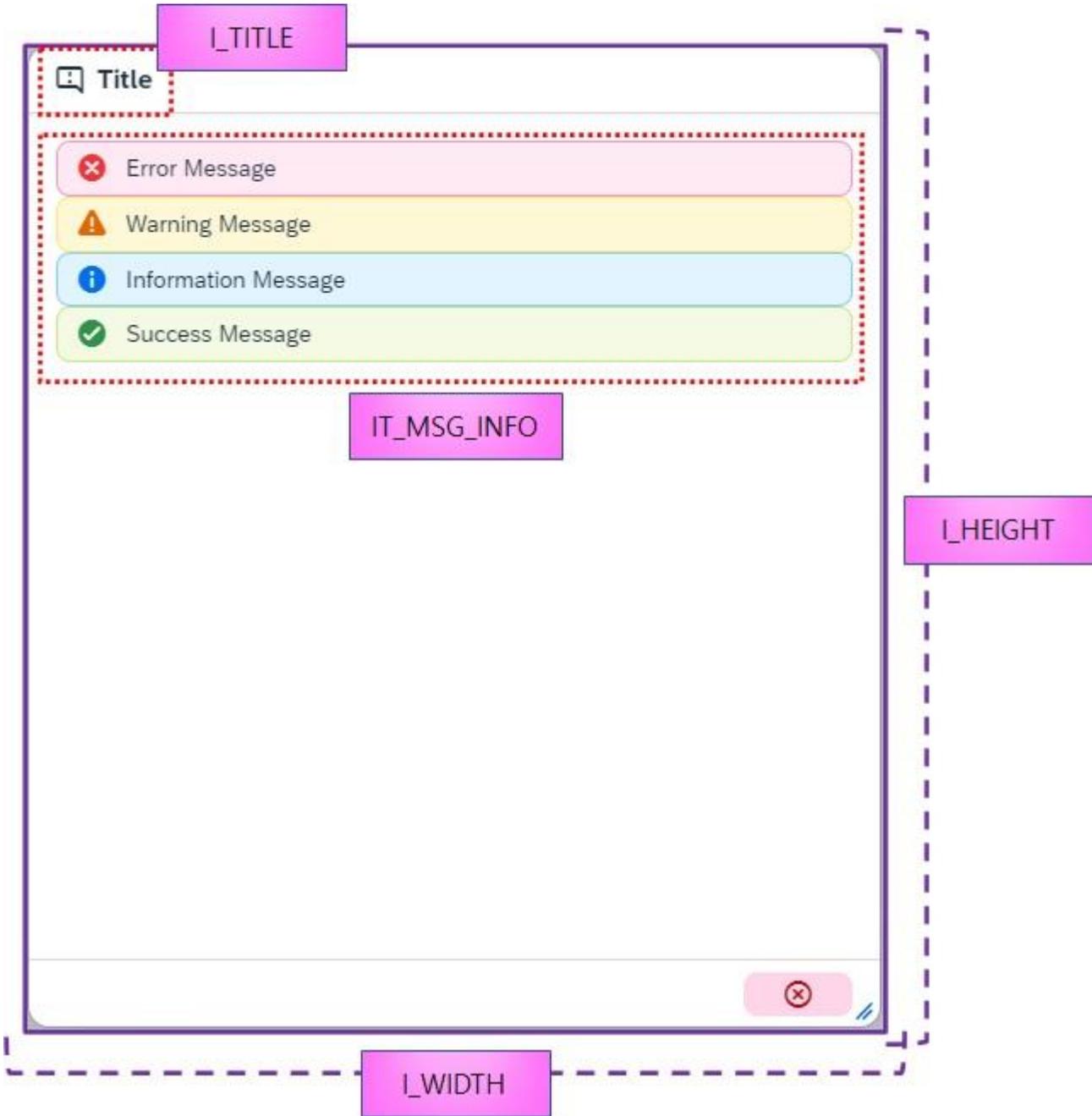
PARAMETERS	MEAN
------------	------

I_TITLE	Message_Strip Dialog 제목	
I_WIDTH	Message_Strip Dialog 가로 넓이	
I_HEIGHT	Message_Strip Dialog 세로 높이	
IT_MSG_INFO	Message_Strip Dialog에 보여 줄 Message 정보 Table	
	TYPE	E : Error, W : Warning, I : Information, S : Success
	TEXT	메시지 텍스트 내용

3. Application 실행 한 후 MESSAGE_STRIP_TEST Button Click.



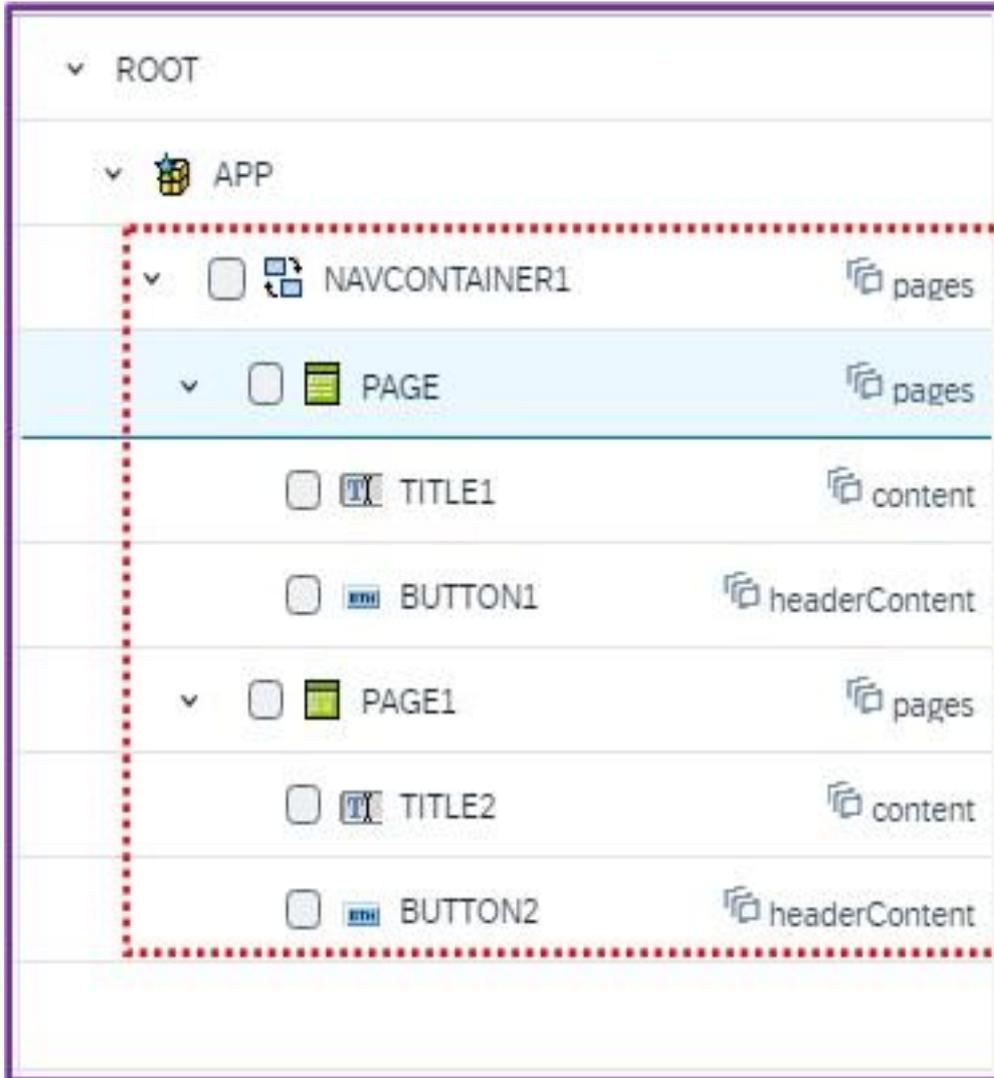
4. Message_Strip Dialog 화면이 호출되는 것을 확인.



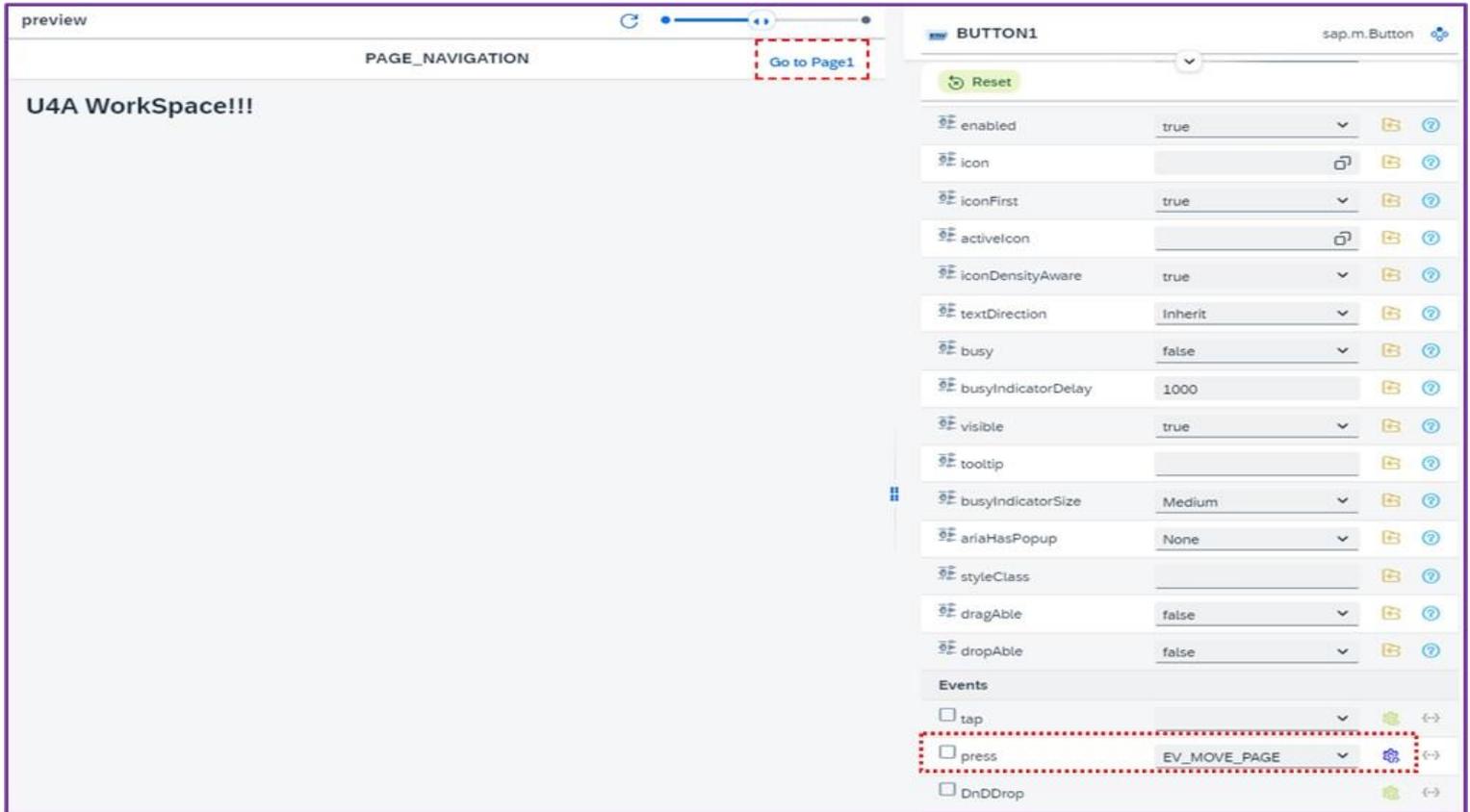
6.1.24 Methodes PAGE NAVIGATION

■ 목적 : UTILITIES 중 "PAGE_NAVIGATION"은 PAGE끼리 이동할 때 사용하는 Method이다.

1. Page와 Navcontainer UI Object를 생성한다. 이때, PAGE, PAGE1은 하나의 UI Object 하위 항목에 존재해야 한다.



2. PAGE 이동을 위한 Events를 만든다.



3. PAGE_NAVIGATION을 입력하고 PAGE가 포함 되어있는 NAVIGATION Object ID 와 이동하려는 PAGE를 입력한다.

[첨부ABAP SOURCE (EV_MOVE_PAGE)]

```
METHOD EV_MOVE_PAGE.
```

```
DATA : LV_PAGE_ID TYPE STRING.
```

```
*-----*
```

```
CASE I_ID.
```

```
WHEN 'BUTTON1'. "PAGE에 있는 버튼
```

```
"PAGE1로 이동
```

```
LV_PAGE_ID = 'PAGE1'.
```

WHEN 'BUTTON2'. "PAGE1에 있는 버튼

"PAGE로 이동

LV_PAGE_ID = 'PAGE'.

ENDCASE.

"PAGE MOVE.

CALL METHOD /U4A/CL_UTILITIES=>PAGE_NAVIGATION

EXPORTING

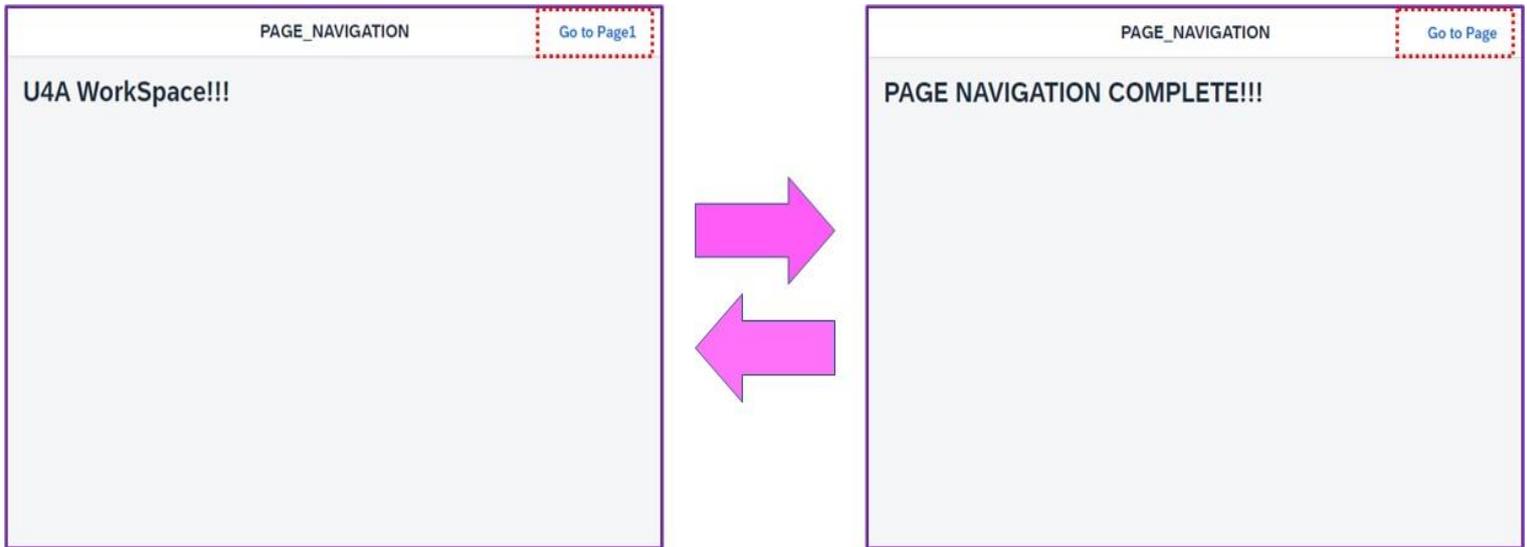
IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class

I_APPID = 'NAVCONTAINER1'

I_PAGEID = LV_PAGE_ID.

ENDMETHOD.

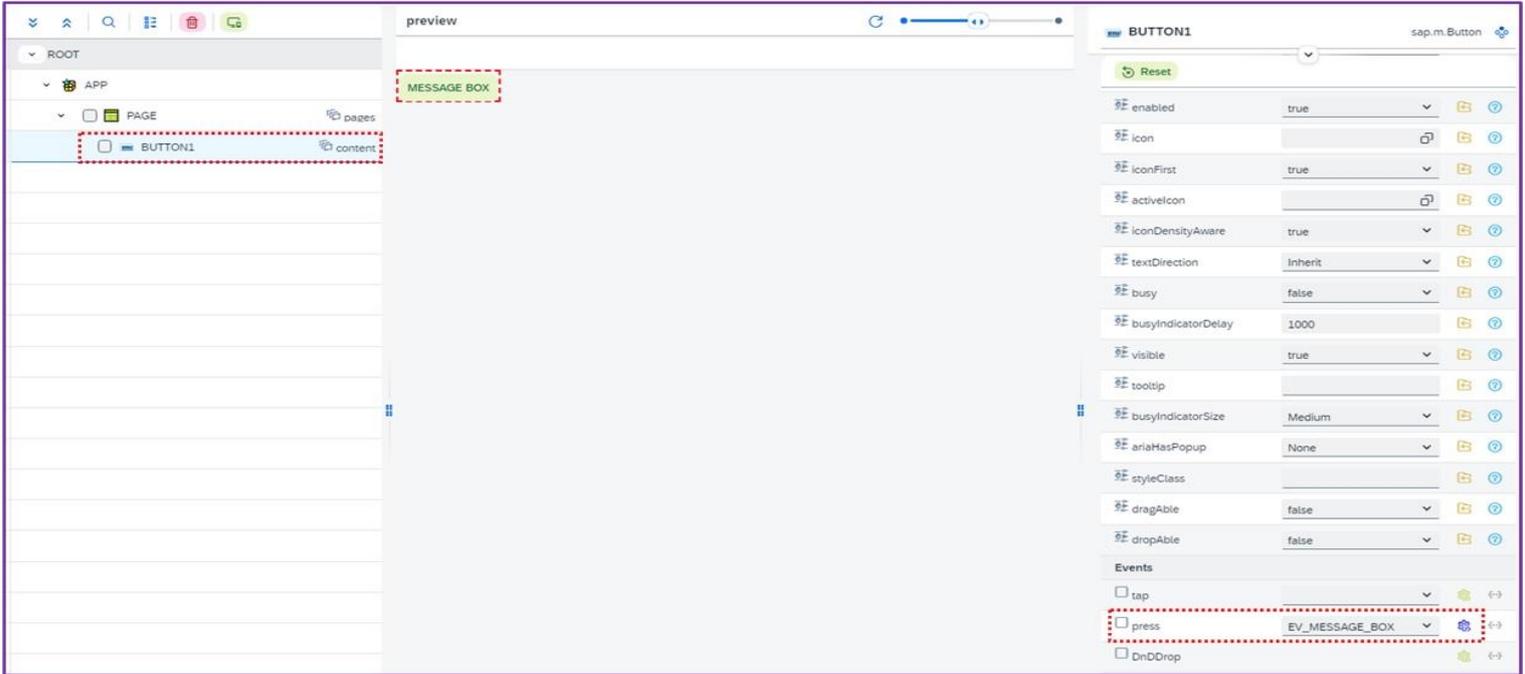
4. Web에서 Event를 발동 시키면 PAGE가 바뀌는 것을 확인 할 수 있다.



6.1.25 Methodes M_MESSAGEBOX

■ 목적 : UTILITIES 중 "M_MESSAGEBOX"는 Web에서 POPOVER형태로 Message를 호출할 수 있다.

1. Application에서 UI Object와 Event를 생성한다.



2. Messagebox 호출을 위한 Event를 만든다.

[첨부ABAP SOURCE (EV_MESSAGE_BOX)]

```
METHOD EV_MESSAGE_BOX.
```

```
DATA : LV_MSG TYPE STRING,  
       LT_MSG TYPE STRING_TABLE.
```

"View Detail 내역 및 ACTION BUTTON 내역 입력"

```
LV_MSG = 'ONE'.  
APPEND LV_MSG TO LT_MSG.  
CLEAR LV_MSG.
```

```
LV_MSG = 'TWO'.
APPEND LV_MSG TO LT_MSG.
CLEAR LV_MSG.
```

"MESSAGE BOX 호출

```
CALL METHOD /U4A/CL_UTILITIES=>M_MESSAGEBOX
EXPORTING
```

```
IO_VIEW = ME-
```

```
>/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class
```

```
* IT_MSGTX = " Table of Strings
```

```
* IS_MSG = " Structure of message variables
```

```
I_MSGTX = 'U4A WorkSpace!!' " MESSAGE TEXT
```

```
I_POPUP_TYPE = /U4A/CL_UTILITIES=>CS_M_MSG_BOX_TP-
```

```
ALERT " /U4A/CL_UTILITIES=>CS_M_MSG_BOX_TP
```

```
I_TITLE = 'MESSAGE BOX'
```

```
I_ICON = 'sap-icon://appear-offline'
```

```
I_CALLBACK_EVENT = 'EV_ANOTHER_MESSAGE'
```

```
I_ACTION = LT_MSG " 처리 액션 버튼 정의
```

```
I_DETAILS = LT_MSG " Long text (상세)
```

```
I_TEXTDIRECTION = 'RTL'
```

```
I_INITIALFOCUS = 'TWO'. " Initial button focus is set by attribute
```

```
ENDMETHOD.
```

EXPORTING	Means
IO_VIEW	UI Element Super Class 를 입력한다
IT_MSGTX	MESSAGE가 TABLE 구조이면 해당 PARAMETER에 입력한다
IS_MSG	ABAP에 등록되어 있는 MESSAGE를 사용할 수 있다.
I_MSGTX	MESSAGE가 FIELD 구조이면 해당 PARAMETER에 입력한다.
I_POPUP_TYPE	MESSAGE BOX의 TYPE을 결정하는 부분으로 ALERT, CONFIRM, ERROR, INFORMATION, SHOW, SUCCESS, WARNING가 있다.
I_TITEL	MESSAGE BOX의 제목을 입력한다.

I_ICON	제목 옆 ICON을 추가할때 사용한다.
I_CALLBACK_EVENT	MESSAGE BOX를 종료 후 작동하는 Event
I_ACTION	처리하는 ACTION BUTTON 정의
I_DETAILS	View Detail 내역 입력
I_TEXTDIRECTION	일반문자와 특수문자의 방향을 결정한다 (Inherit, LTR, RTL)
I_INITIALFOCUS	Message의 초기 focus 위치를 변경

Method	Level	Visibility	M...	Description
/U4A/IF_SERVER~HANDL_ON_INIT	Instance Method	Public		Initialization event
/U4A/IF_SERVER~HANDL_ON_REQUEST	Instance Method	Public		Asynchronous event
/U4A/IF_SERVER~HANDL_ON_RESPONSE	Instance Method	Public		Not used
/U4A/IF_SERVER~HANDL_ON_EXIT	Instance Method	Public		Close event
/U4A/IF_SERVER~HANDL_ON_MESSAGE	Instance Method	Public		Push Channel Services Web Socket message receive event
/U4A/IF_SERVER~HANDL_ON_HASHCHANGE	Instance Method	Public		Router hash change(browser의 뒤로가기) callback event
CONSTRUCTOR	Instance Method	Public		
EV_ANOTHER_MESSAGE	Instance Method	Public		[EVENT] SECOND MESSAGE
EV_MESSAGE_BOX	Instance Method	Public		[EVENT] MESSAGE BOX

[첨부ABAP SOURCE (EV_ANOTHER_MESSAGE)]

METHOD EV_ANOTHER_MESSAGE .

DATA : LS_MES TYPE SYMSG .

"ABAP에 등록되어 있는 MESSAGE의 형태에 맞춰 데이터를 입력

LS_MES-MSGID = '/U4A/MSG_WS' .

LS_MES-MSGNO = 024 .

LS_MES-MSGTY = 'E' .

LS_MES-MSGV1 = 'U4A' .

"MESSAGE BOX 호출

CALL METHOD /U4A/CL_UTILITIES=>M_MESSAGEBOX

EXPORTING

IO_VIEW = ME-

>/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class

* IT_MSGTX = " Table of Strings

```
IS_MSG      = LS_MES
```

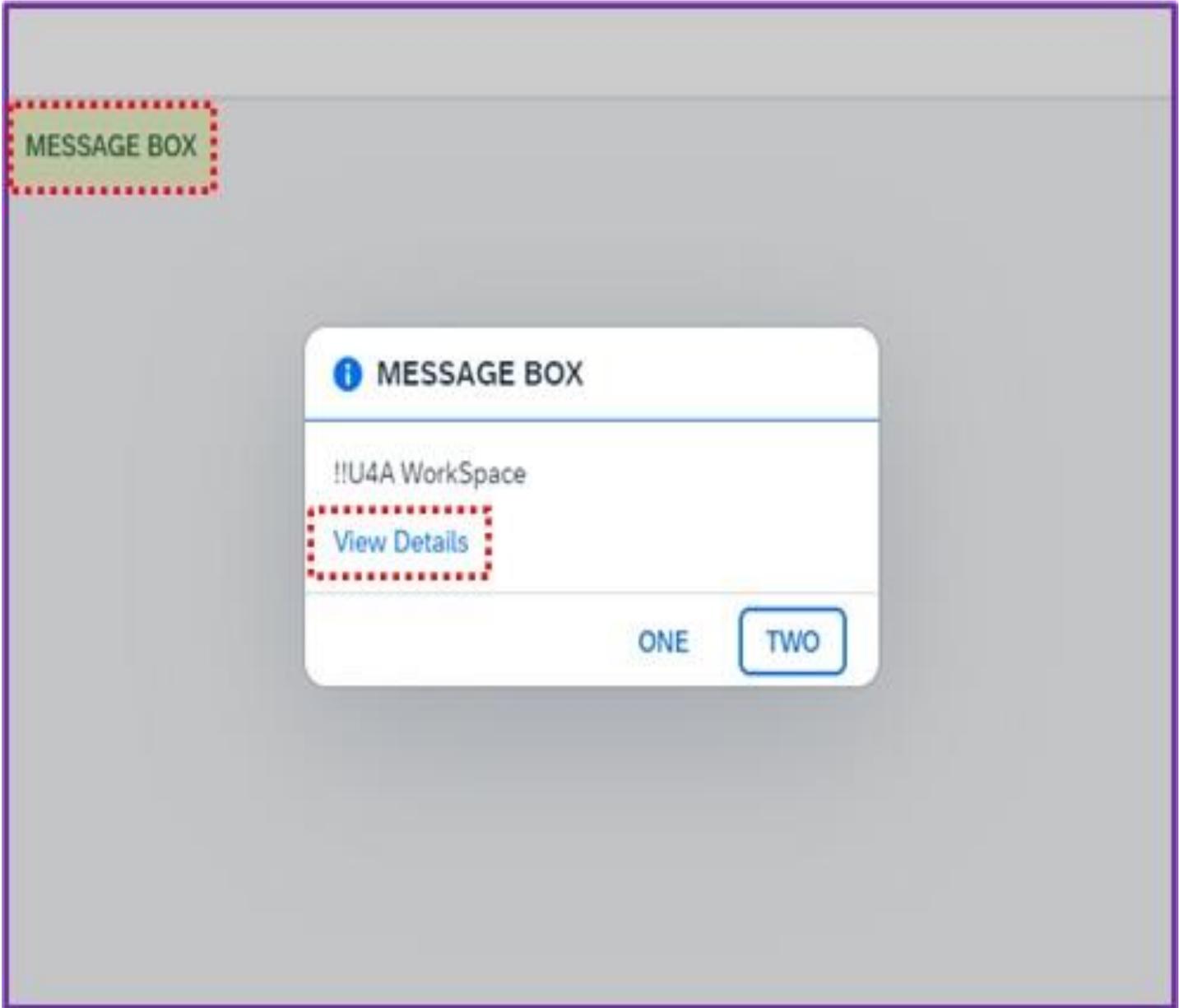
```
" Structure of message varia
```

```
bles
```

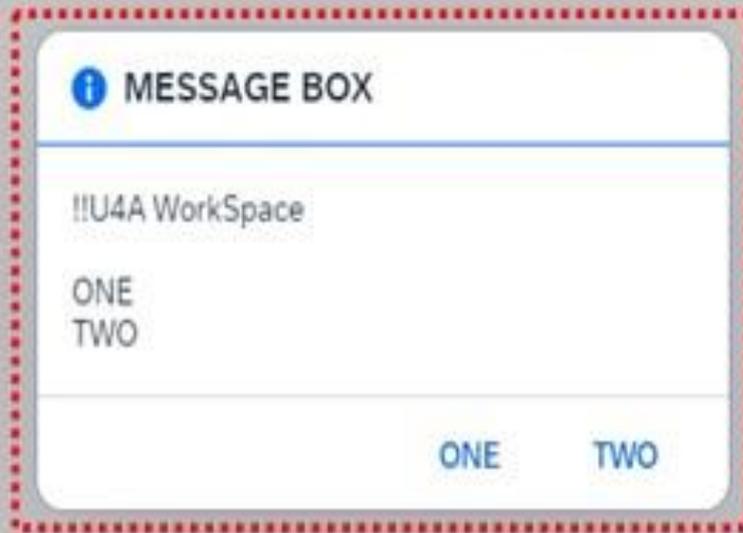
```
* I_MSGTX      = " MESSAGE TEXT
I_POPUP_TYPE = /U4A/CL_UTILITIES=>CS_M_MSG_BOX_TP-
CONFIRM " /U4A/CL_UTILITIES=>CS_M_MSG_BOX_TP
I_TITLE      = 'SECOND MESSAGE BOX'
* I_ICON      =
* I_CALLBACK_EVENT =
* I_ACTION    = " 처리 액션 버튼 정의
* I_DETAILS   = " Long text (상세)
* I_TEXTDIRECTION =
* I_INITIALFOCUS = " Initial button focus is set by attribute
```

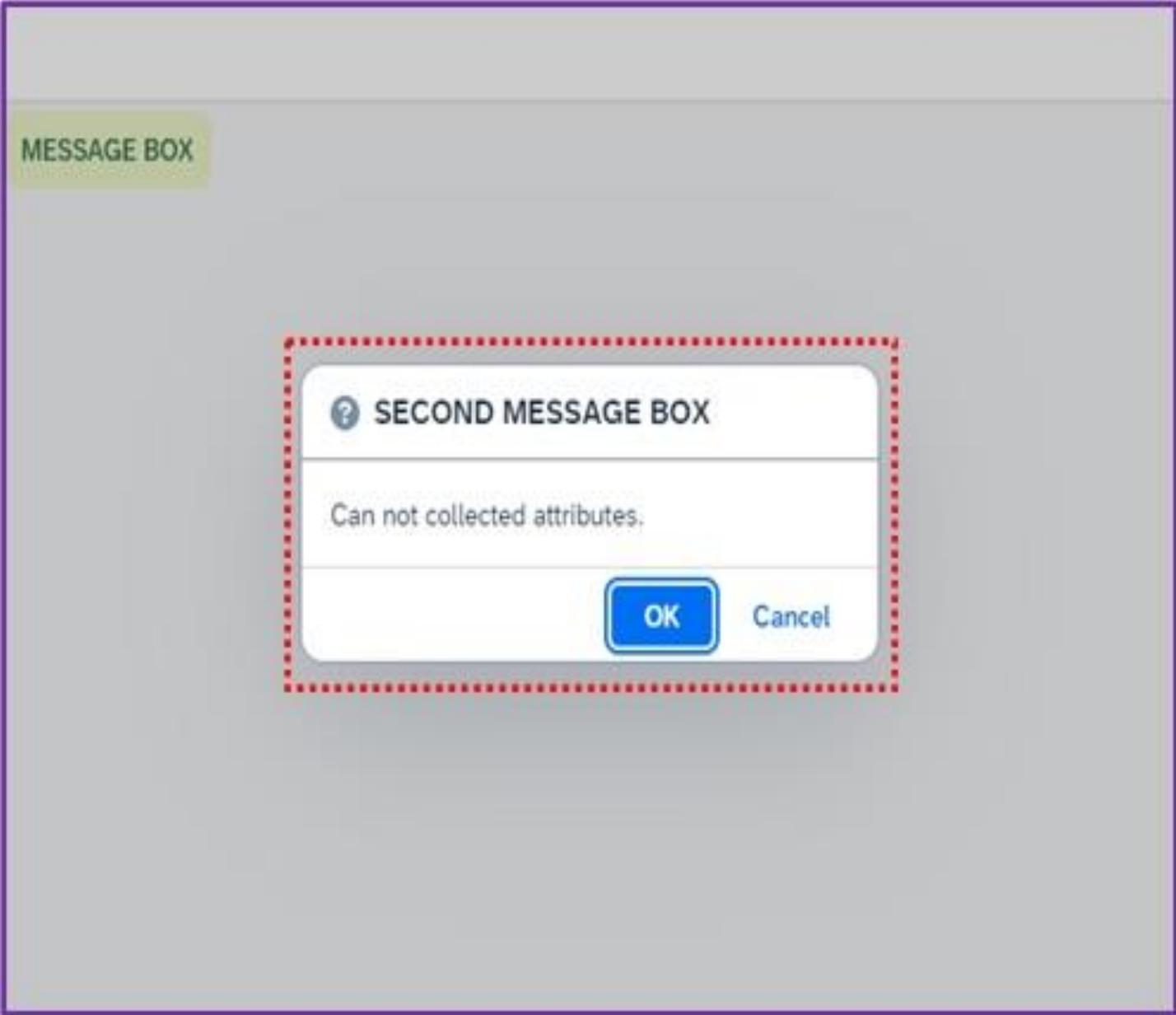
```
ENDMETHOD.
```

3. Web에서 BUTTON을 누르면 MESSAGE BOX가 나타난다.



MESSAGE BOX

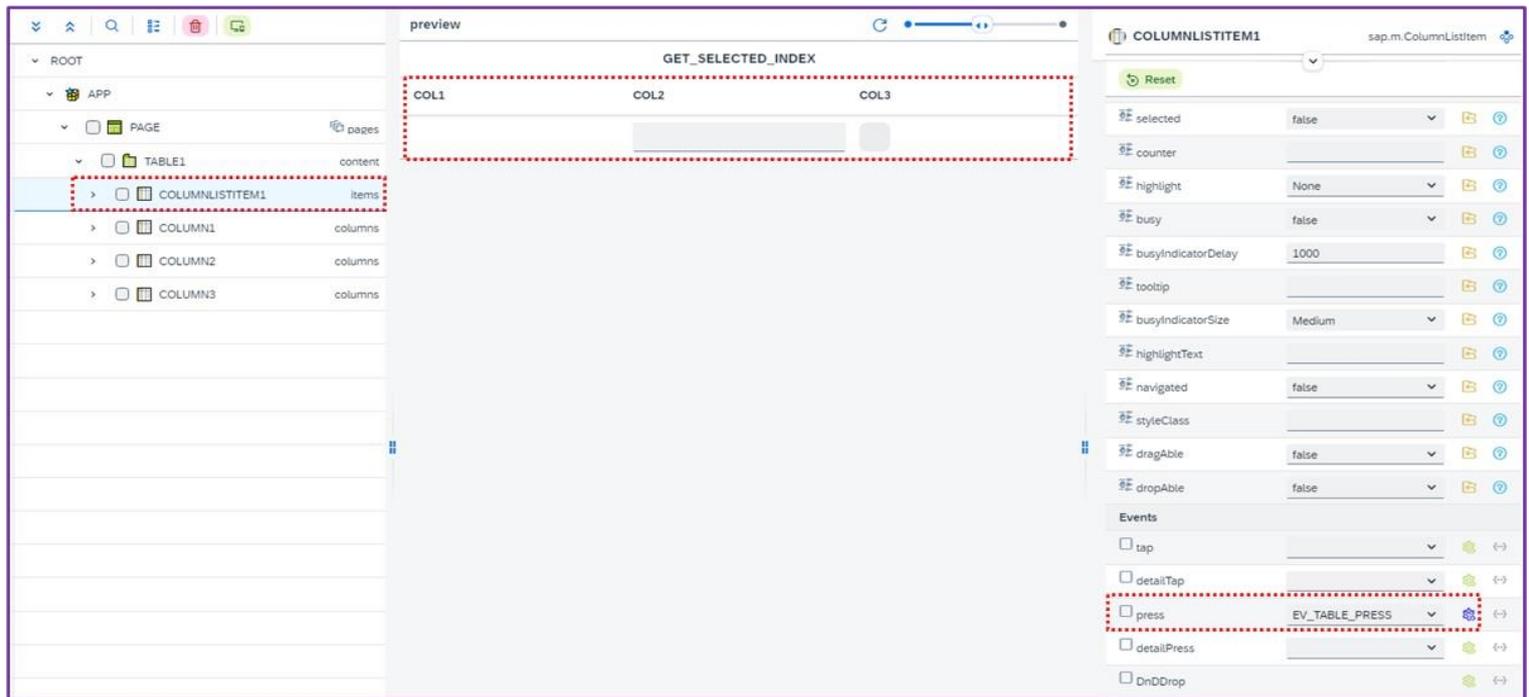




6.1.26 Methodes GET_SELECTED_INDEX

■ 목적 : UTILITIES 중 "GET_SELECTED_INDEX"는 Events 작동시 UI Object의 Index 정보를 가져올 수 있다.

1. Application에서 UI Object와 Event를 생성한다.



2. "GET_SELECTED_INDEX" Method 호출 후 I_EVENT_NAME에서 INDEX값만 가져온다.

[첨부ABAP SOURCE (EV_TABLE_PRESS)]

```
METHOD EV_TABLE_PRESS.
```

```
DATA : LV_INDEX TYPE STRING,  
       LV_MSG   TYPE STRING.
```

"TABLE LINE 선택 시 INDEX 값 가져오기"

```
CALL METHOD /U4A/CL_UTILITIES=>GET_SELECTED_INDEX
EXPORTING
  I_EVENT_NAME = I_EVENT_NAME
IMPORTING
  E_INDEX      = LV_INDEX.
```

"Message 내역

```
CONCATENATE 'You have selected the' `` LV_INDEX `` 'Row of the TABLE' INTO LV_MSG.
```

"MESSAGE BOX 호출

```
CALL METHOD /U4A/CL_UTILITIES=>M_MESSAGEBOX
EXPORTING
```

```
  IO_VIEW      = ME-
```

```
>/U4A/IF_SERVER~AR_VIEW
```

" [U4A] UI Element Super Class

```
* IT_MSGTX      = " Table of Strings
```

```
* IS_MSG        = " Structure of message variables
```

```
  I_MSGTX       = LV_MSG
```

" MESSAGE TEXT

```
  I_POPUP_TYPE = /U4A/CL_UTILITIES=>CS_M_MSG_BOX_TP-
```

```
ALERT " /U4A/CL_UTILITIES=>CS_M_MSG_BOX_TP
```

```
* I_TITLE       =
```

```
* I_ICON        =
```

```
* I_CALLBACK_EVENT =
```

```
* I_ACTION      = " 처리 액션 버튼 정의
```

```
* I_DETAILS     = " Long text (상세)
```

```
* I_TEXTDIRECTION =
```

```
* I_INITIALFOCUS = " Initial button focus is set by attribute
```

```
ENDMETHOD.
```

3. Web에서 TABLE의 LINE을 눌러 Event를 작동시키면 Index값을 읽어온다.

GET_SELECTED_INDEX		
COL1	COL2	COL3
1	U4A WorkSpace	U4A
2		
3		

Alert

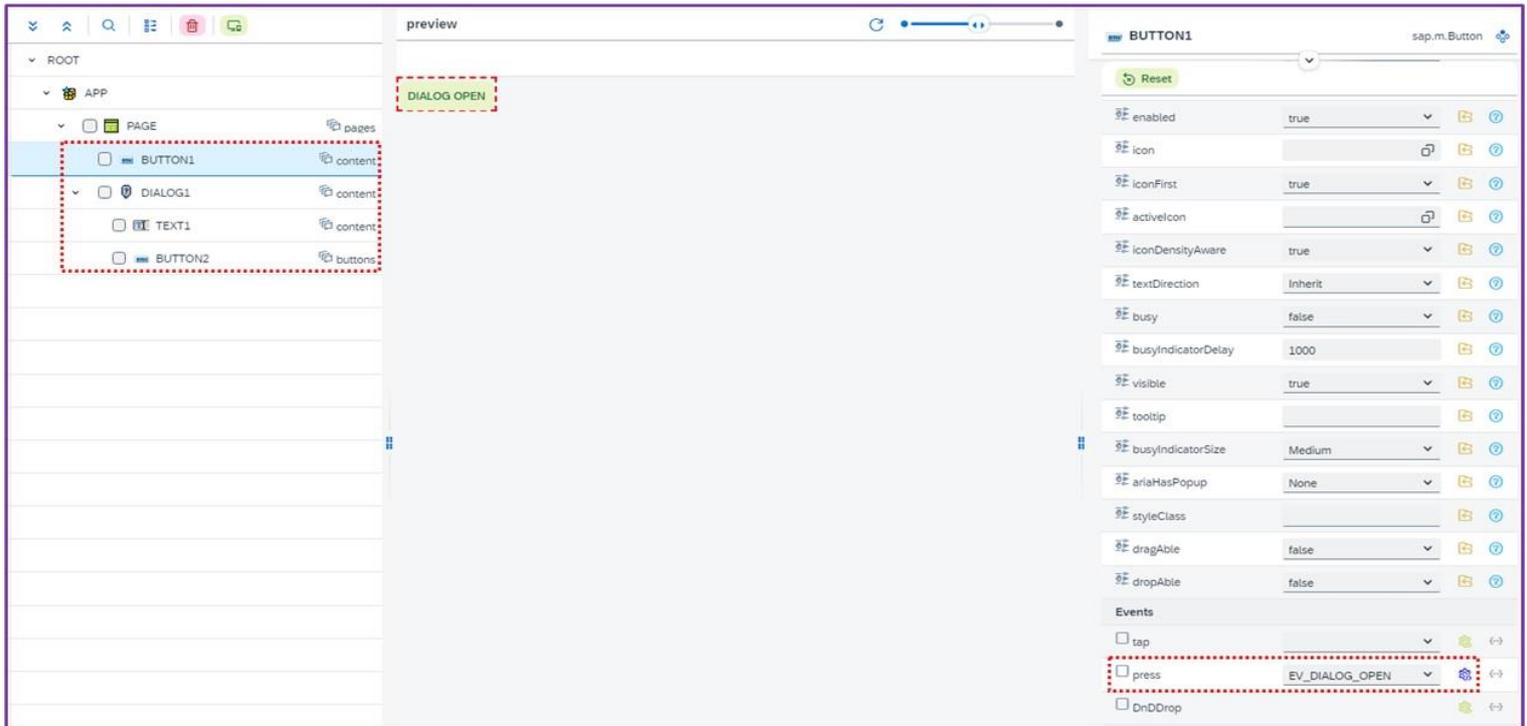
You have selected the 2'Row of the TABLE

OK

6.1.27 Methodes DIALOG OPEN

■ 목적 : UTILITIES 중 "DIALOG_OPEN"은 DIALOG UI를 현재 PAGE내에서 작은 창 형태로 호출할 때 사용한다.

1. Application에서 UI Object와 Event를 생성한다.



2. Event 생성 후 "DIALOG_OPEN" Method를 구성한다.

[첨부ABAP SOURCE (EV_DIALOG_OPEN)]

```
METHOD EV_DIALOG_OPEN.
```

```
"DIALOG OPEN EVENT.
```

```
CALL METHOD /U4A/CL_UTILITIES=>DIALOG_OPEN
```

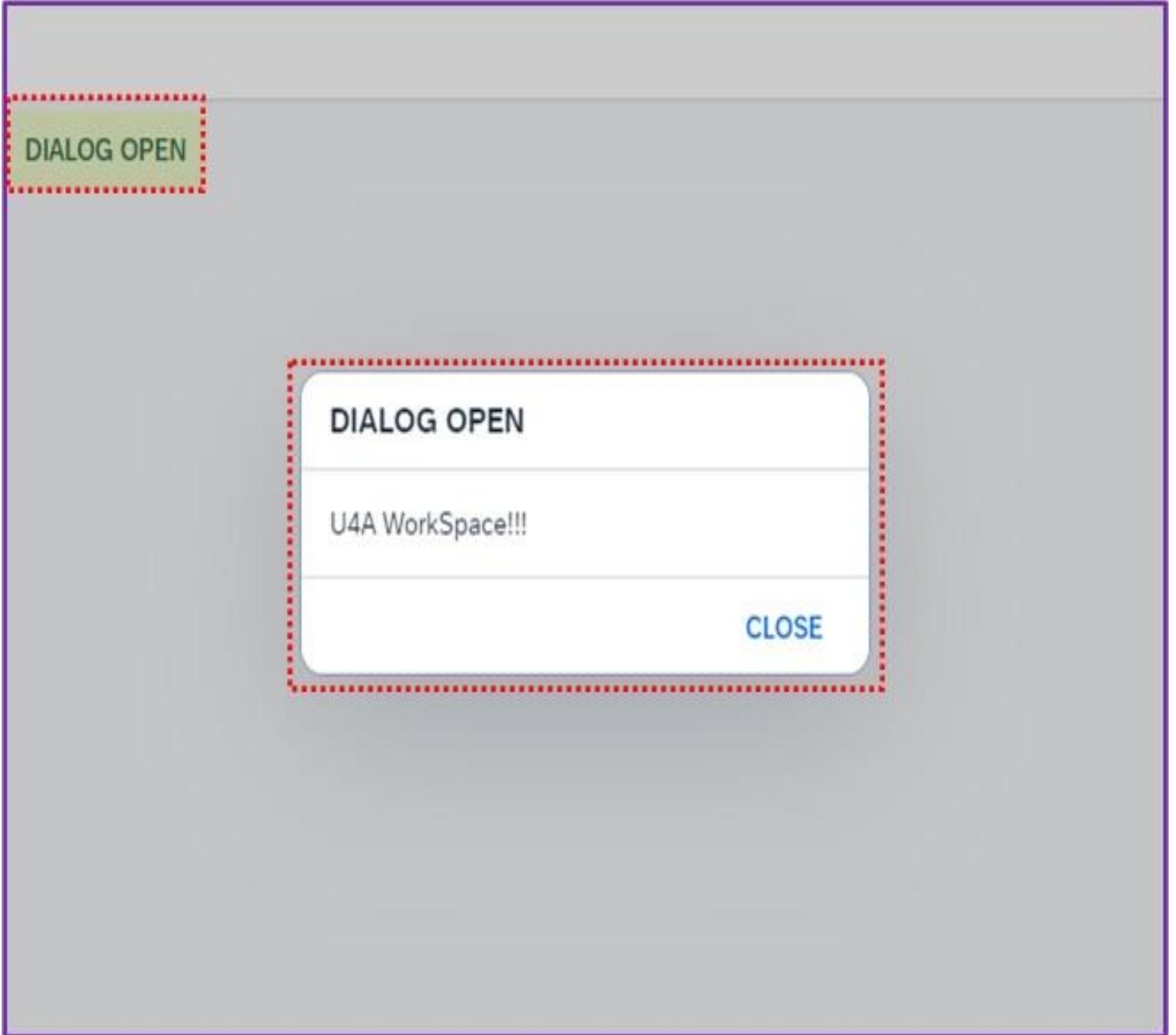
```
EXPORTING
```

```
IO_VIEW      = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class
```

```
I_DIALOG_ID = 'DIALOG1'. " DIALOG UI ID
```

ENDMETHOD .

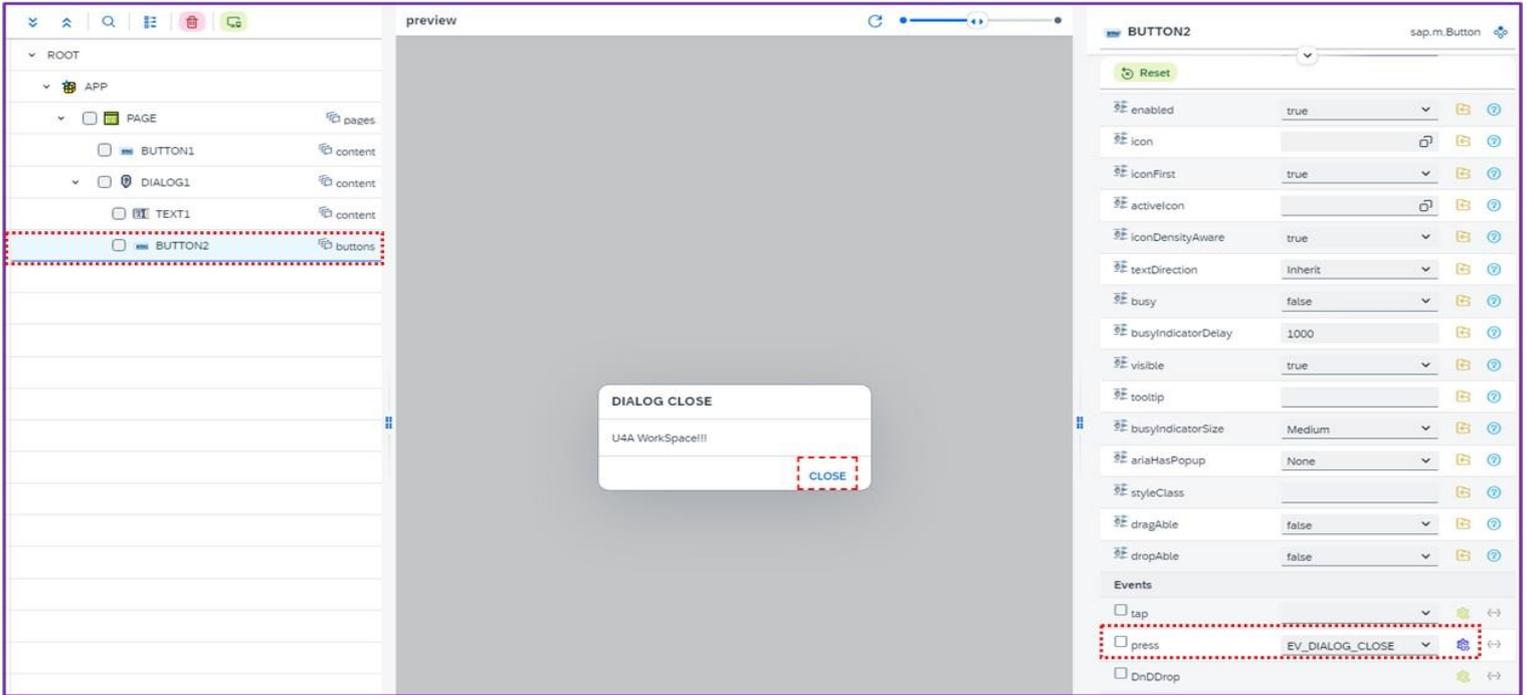
3. Web에서 DIALOG_OPEN Event를 작동 시키면 DIALOG가 호출되는 것을 확인 할 수 있다.



6.1.28 Methodes DIALOG CLOSE

■ 목적 : UTILITIES 중 "DIALOG_CLOSE"는 DIALOG창을 닫을 때 사용한다.

1. DIALOG UI에서 UI Object와 Event를 생성한다.



2. Event 생성 후 "DIALOG_CLOSE" Method를 구성한다.

[첨부ABAP SOURCE (EV_DIALOG_CLOSE)]

METHOD EV_DIALOG_CLOSE.

"DIALOG_CLOSE EVENT.

CALL METHOD /U4A/CL_UTILITIES=>DIALOG_CLOSE

EXPORTING

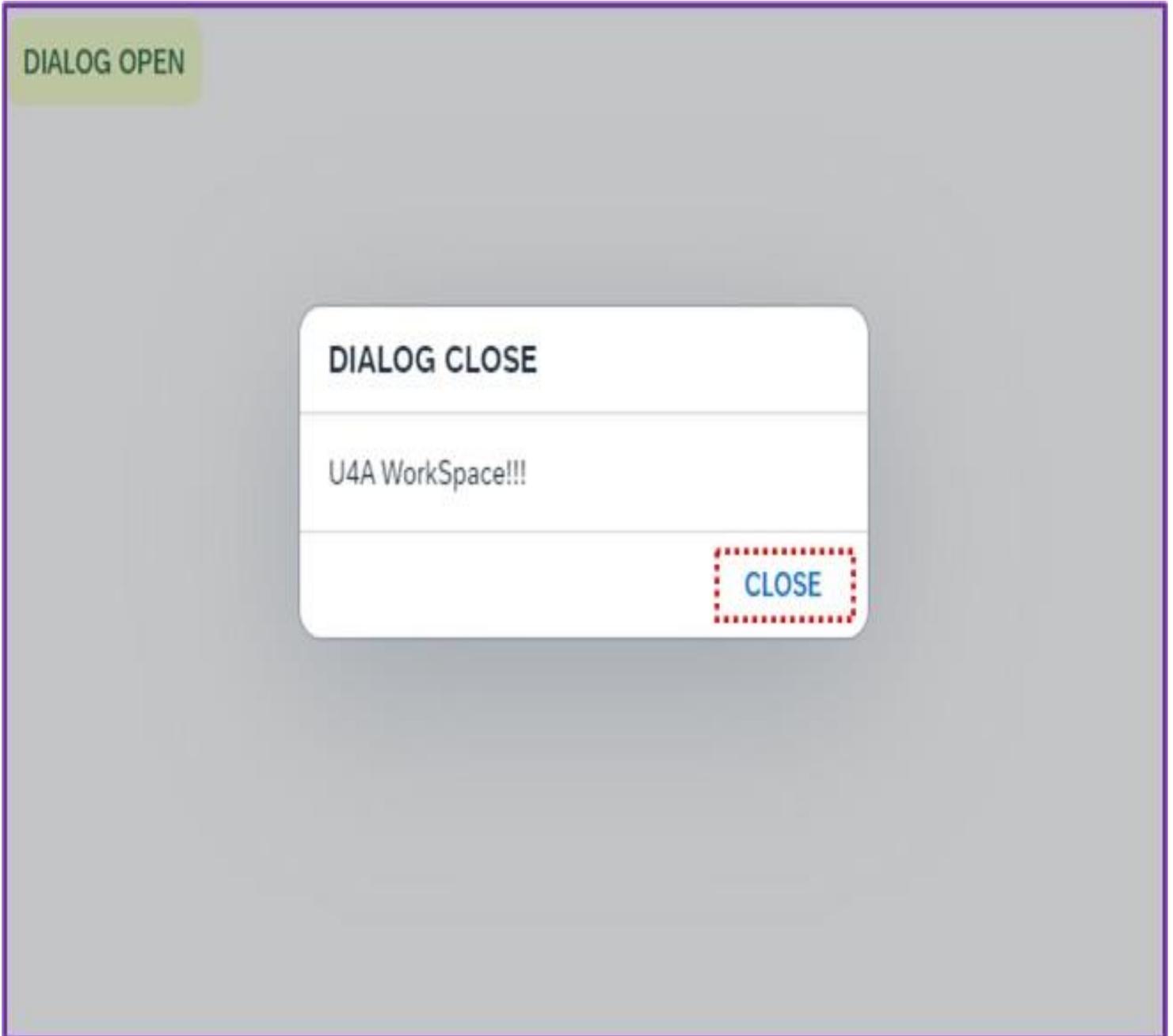
IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class

I_DIALOG_ID = 'DIALOG1'. " DIALOG UI ID

* I_PARENT_REFRESH = " 부모 APPLICATION존재시 갱신여부('X' = 갱신)!!Do not Use!(Deprecated)!

ENDMETHOD.

3. Web에서 DIALOG_CLOSE Event를 작동 시키면 DIALOG창이 종료되는것을 확인 할 수 있다.



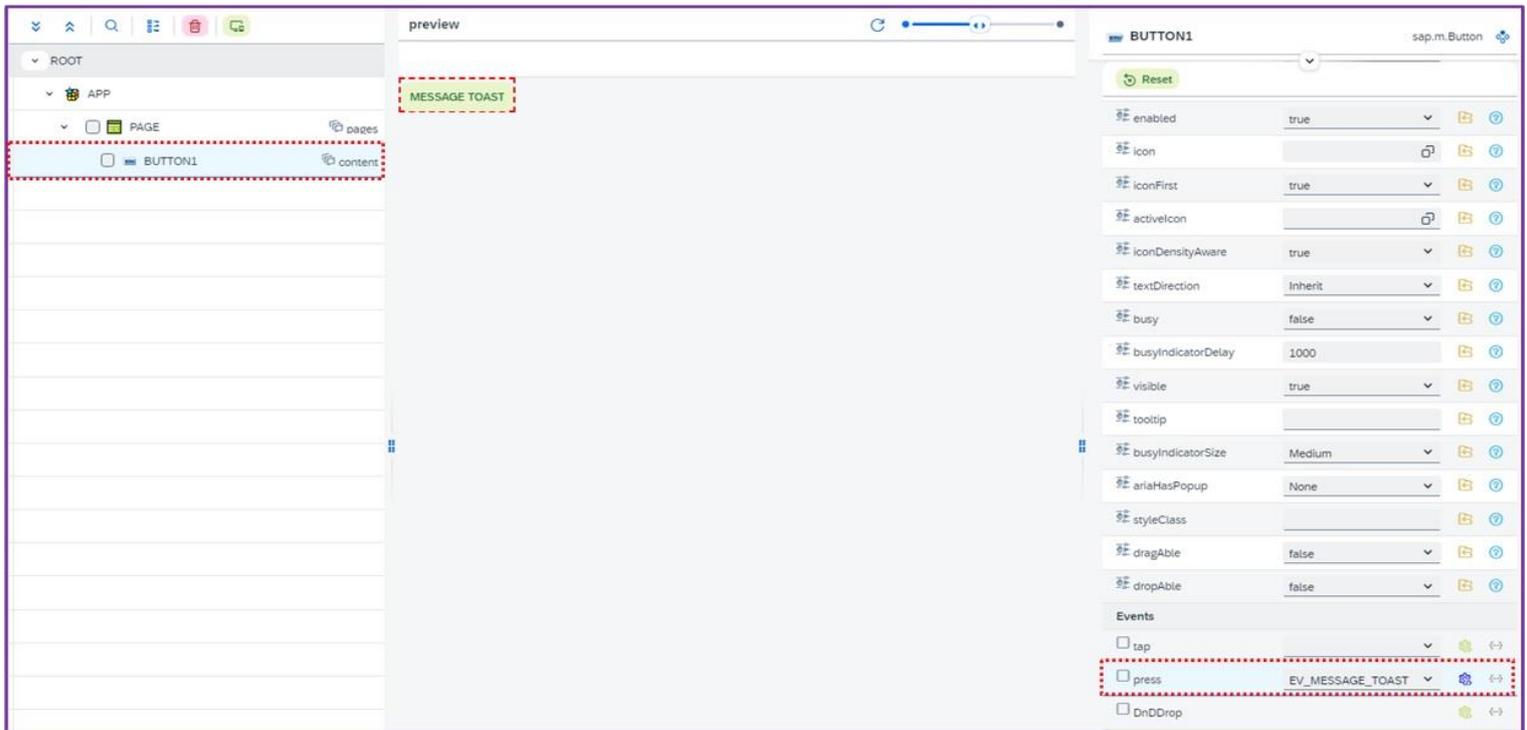
DIALOG OPEN

 DIALOG_CLOSE중 I_PARENT_REFRESH는 현재 폐기 상태로 사용하지 않는다.

6.1.29 Methodes M_MESSAGE_TOAST

■ 목적 : UTILITIES 중 "M_MESSAGE_TOAST"는 Web에서 MESSAGE 출력만 가능한 POPOVER를 호출한다.

1. application에서 UI Object와 Event를 생성한다.



2. Message 호출을 위한 Event를 만든다.

[첨부ABAP SOURCE (EV_MESSAGE_TOAST)]

```
METHOD EV_MESSAGE_TOAST.
```

```
DATA LV_MSG TYPE STRING.
```

```
"MESSAGE Description
```

```
LV_MSG = 'U4A MESSAGE TOAST TEST!!'.
```

"MESSAGE TOAST EVENT.

CALL METHOD /U4A/CL_UTILITIES=>M_MESSAGE_TOAST

EXPORTING

IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW

" [U4A] UI Element Super Class

* IS_MSG = " Structure of message variables

I_MSGTX = LV_MSG

" MESSAGE TEXT

* IT_MSGTX = " Table of Strings

I_VPOS = /U4A/CL_UTILITIES=>CS_M_MSG_TOS_VPOS-CENTER

" Vertical position

I_HPOS = /U4A/CL_UTILITIES=>CS_M_MSG_TOS_HPOS-CENTER

" Horizontal position

* I_VOFF = '0' " Vertical offset

* I_HOFF = '0' " Horizontal offset

* I_DURATION = 10000

ENDMETHOD.

EXPORTING	Means
IO_VIEW	UI Element Super Class 를 입력한다
IS_MSG	ABAP에 등록되어 있는 MESSAGE를 사용할 수 있다.
I_MSGTX	MESSAGE가 FIELD 구조이면 해당 PARAMETER에 입력한다.
IT_MSGTX	MESSAGE가 TABLE 구조이면 해당 PARAMETER에 입력한다.
I_VPOS	MESSAGETOAST의 Vertical position을 BOTTOM, CENTER, TOP로 조정 할 수 있다.
I_HPOS	MESSAGETOAST의 Horizontal position을 BEGIN, CENTER, END, LEFT, RIGHT로 조정 할 수 있다.
I_VOFF	MESSAGETOAST의 Vertical position을 숫자값으로 조정한다.
I_HOFF	MESSAGETOAST의 Horizontal position을 숫자값으로 조정한다.
I_DURATION	MESSAGETOAST의 호출 후 유지 시간 조정(1000 = 1초)

3. Web에서 BUTTON을 누르면 "MESSAGE TOAST"가 나타난다 .

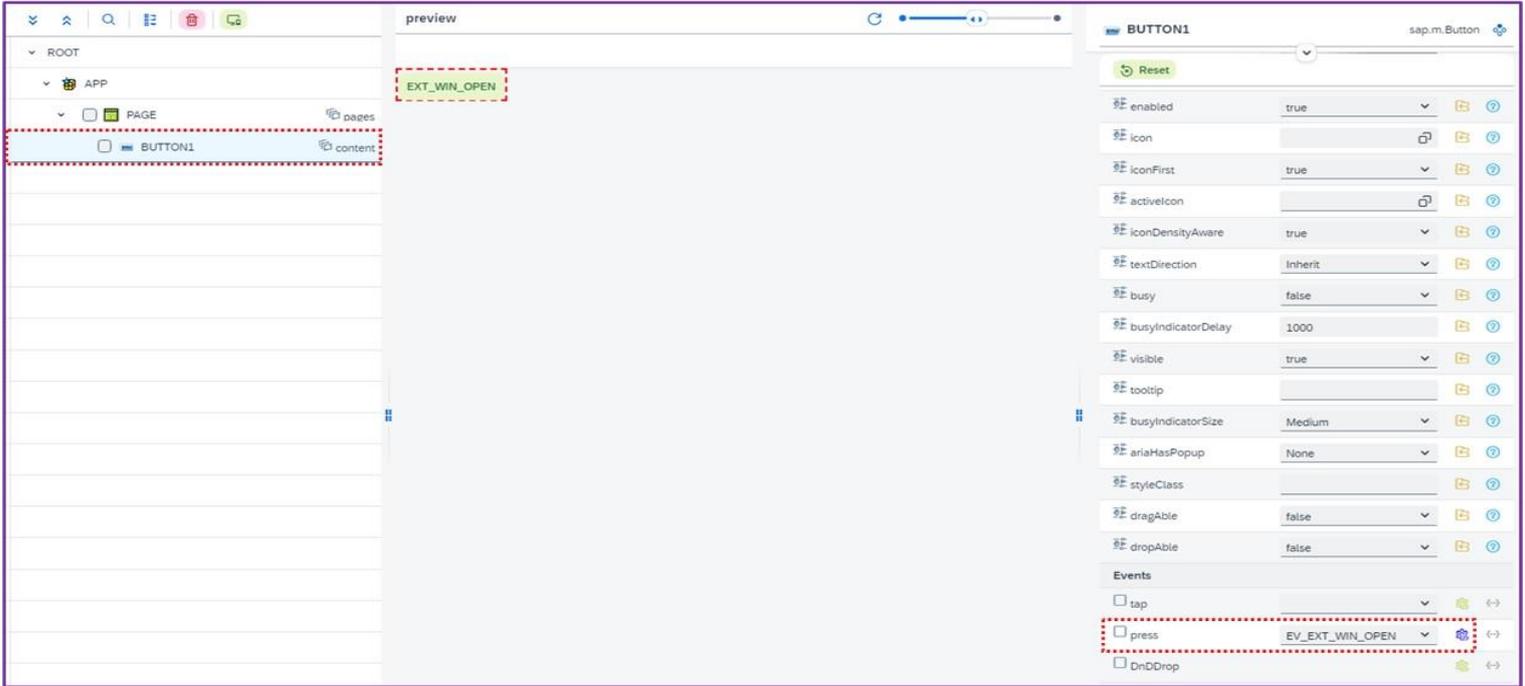
MESSAGE TOAST

U4A MESSAGE TOAST TEST!!

6.1.30 Methodes EXT_WIN_OPEN

■ 목적 : UTILITIES 중 "EXT_WIN_OPEN"는 Event작동 시 자신이 설정한 URL 주소를 호출한다.

1. Application에서 UI Object와 Event를 생성한다.



2. Event 생성 후 "EXT_WIN_OPEN" Method를 구성한다.(이때 IT_PARA는 URL주소를 접속 할 때 Post 방식으로 Parameter에 값을 전달하기 위해 사용한다)

[첨부ABAP SOURCE (EV_EXT_WIN_OPEN)]

METHOD EV_EXT_WIN_OPEN.

DATA: LT_PARA TYPE TIHTTPNVP,
LS_PARA TYPE IHTTPNVP.

"URL주소를 접속 할 때 Parameter에 전달 할 값 설정.

```
LS_PARA-NAME = 'bo_table'.
LS_PARA-VALUE = 'technology'.
APPEND LS_PARA TO LT_PARA.
CLEAR LS_PARA.
```

"URL 주소 호출

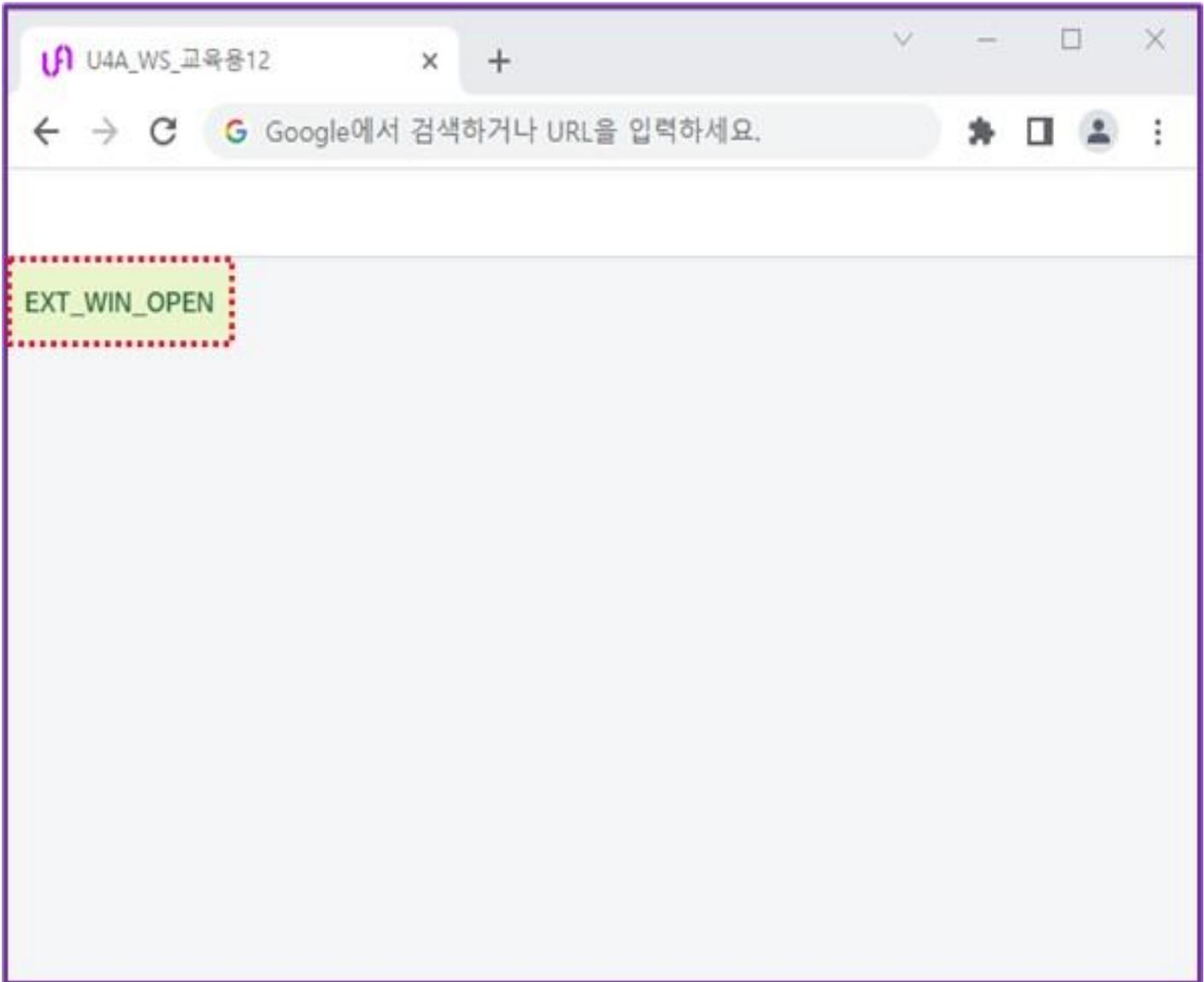
```
CALL METHOD /U4A/CL_UTILITIES=>EXT_WIN_OPEN
EXPORTING
    IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW           " [U4A] UI Element Super Class
    URL      = 'https://www.u4ainfo.com/bbs/board.php' " Target URL
    IT_PARA  = LT_PARA                             " HTTP Framework (iHTTP) Table Name/
```

Value Pairs

- * I_WIDTH = " width(chrome일경우 width, height를 같이 설정해야 동작함)
- * I_HEIGHT = " height(chrome일경우 width, height를 같이 설정해야 동작함)
- * I_LEFT = " open left position
- * I_TOP = " open top position
- * I_RESIZABLE = " resizable(IE only!) 'X' = yes
- * I_SCROLLBARS = " scrollbars(IE only!) 'X' = yes
- * I_FULL = " channelmode(IE only!) 'X' = yes
- * I_CLOSE_CHILD = " 'X' = 메인 window 종료시 open된 하위window 같이 종료됨
- * I_BROWSER_TYPE = " 호출 브라우저 종류 (!!일렉트론 앱 가능함!!)CR=크롬, IE=IE, EDGE=엣지

ENDMETHOD.

3. Web에서 "EXT_WIN_OPEN" Event를 작동 시키면 자신이 지정한 URL 주소로 새 창이 나타난다.



U4A IDE HELP

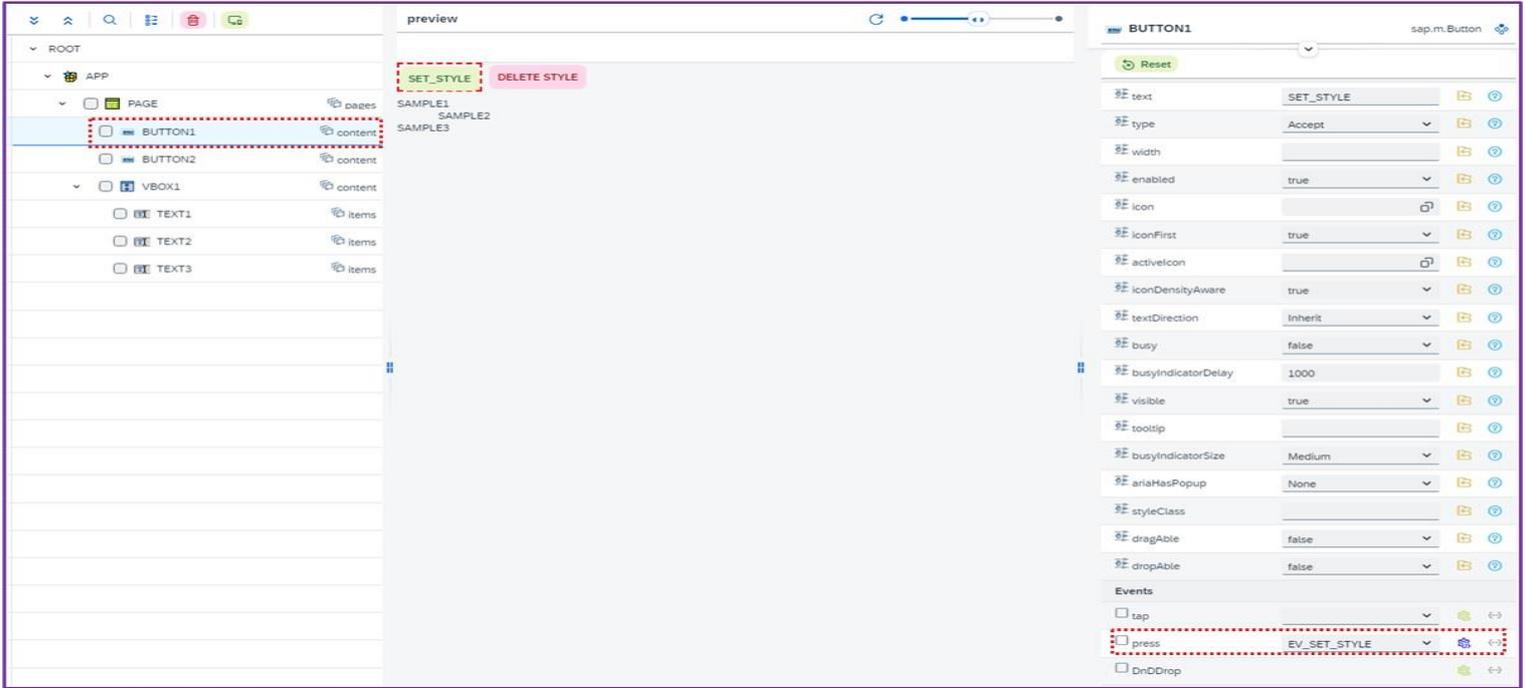
The screenshot shows a web browser window with the URL u4ainfo.com/ebbi/board.php. The page header includes the U4A IDE PLATFORM logo and navigation links: U4A IDE 정보, 라이선스 공지, 관련기술, 파트너, and 문의하기. Below the header is a blue banner with the text 'U15 Application Development Solution for SAP Developers' and 'U15 For ABAP IDE'. The main content area is titled '관련기술' (Related Technology) and contains a table of related links.

번호	제목	글쓴이	날짜
공지	관련 기술 정보 사이트 공유 U4A IDE Platform 관련 기술 정보 사이트를 공유해 드립니다.	관리자 [수정]	2019-03-06
7	TensorFlow An end-to-end open source machine learning platform	관리자 [수정]	2019-05-08
6	SAP Help Portal Welcome to the SAP Help Portal Find SAP product documentation, Learning Journeys, and more ...	관리자 [수정]	2019-03-06
5	SAP Community Home Welcome to the SAP Community!!	관리자 [수정]	2019-03-06
4	SAP Support Portal Home Welcome to the SAP Support Portal, Search for SAP Notes, SAP Knowledge Base Articles, SAP Community conte...	관리자 [수정]	2019-03-06
3	w3schools.com The World's Largest WEB Developer Site!!	관리자 [수정]	2019-03-06

6.1.31 Methodes SET_STYLE

▣ 목적 : UTILITIES 중 "SET_STYLE"은 UI Object의 Style을 적용시키거나 삭제 할 수 있다.

1. application에서 UI Object와 Event를 생성하고, UI Object에 styleClass를 적용시킨다.



2. Event 생성 후 "SET_STYLE"을 구성한다.

[첨부ABAP SOURCE (EV_SET_STYLE)]

METHOD EV_SET_STYLE.

"SET STYLE EVENT.

CALL METHOD /U4A/CL_UTILITIES=>SET_STYLE

EXPORTING

IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class

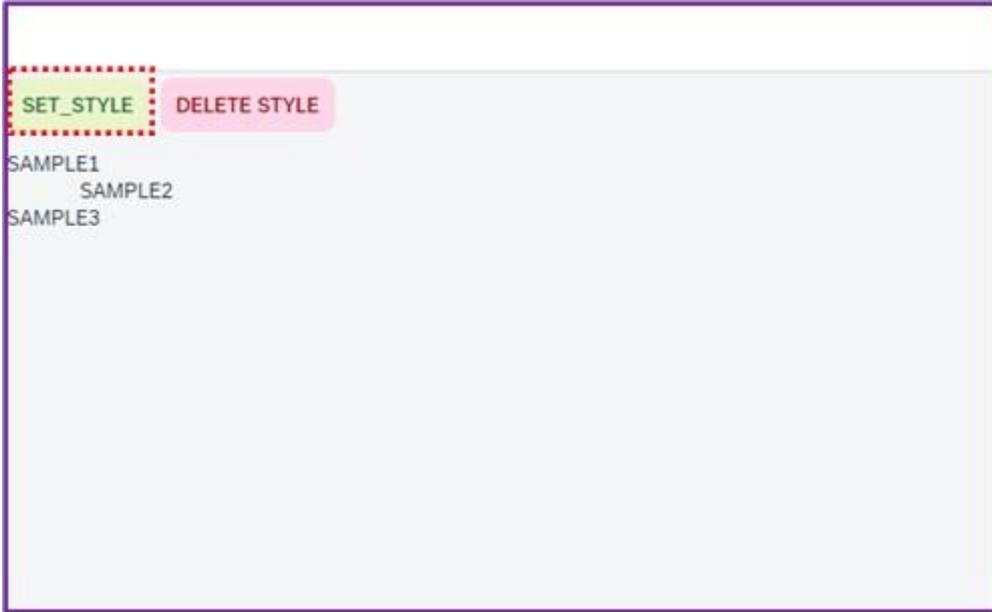
ID = 'TEXT2' " UI 오브젝트 ID

* ISDEL = " 스타일 클래스명 삭제 여부(X)

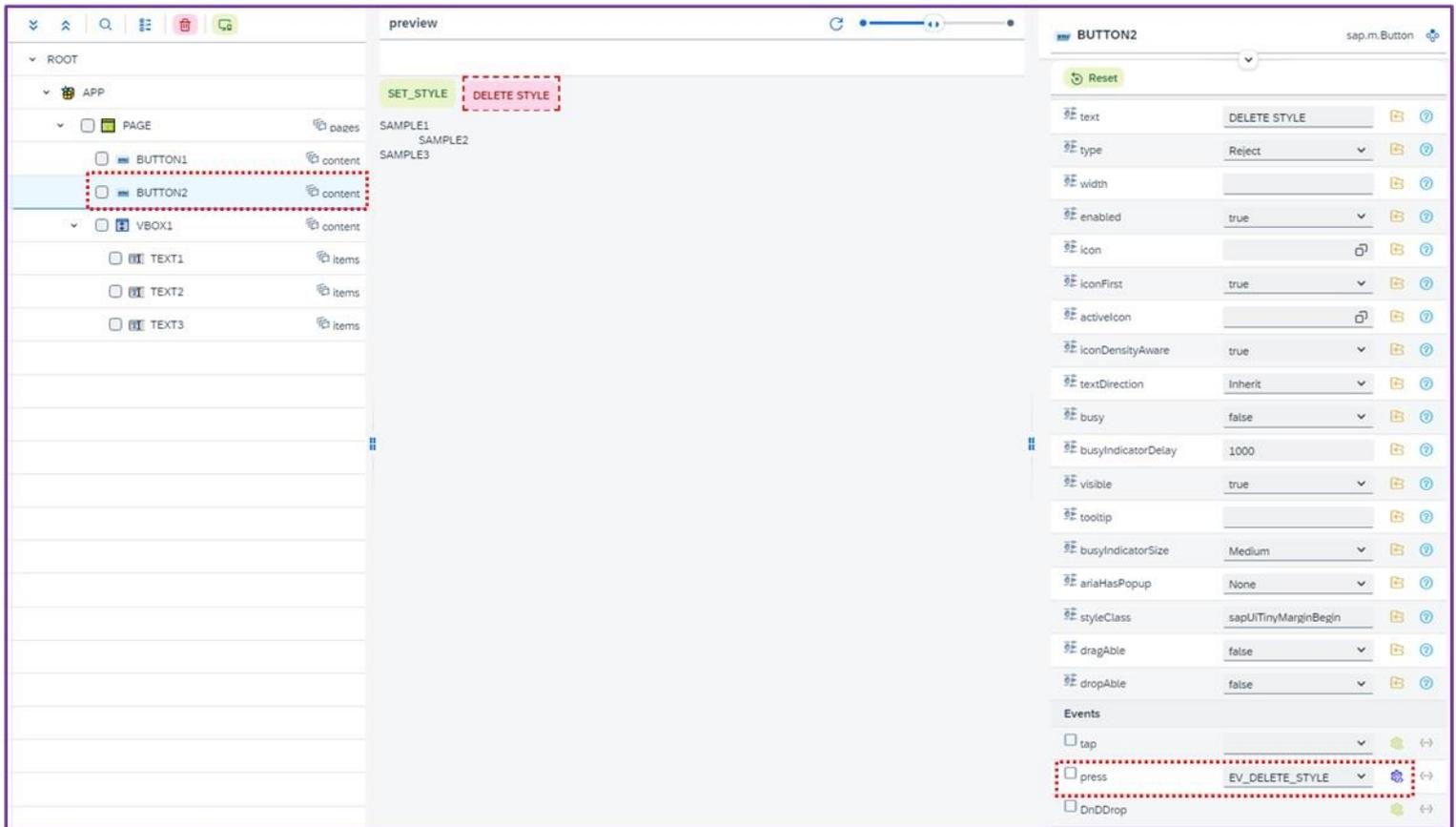
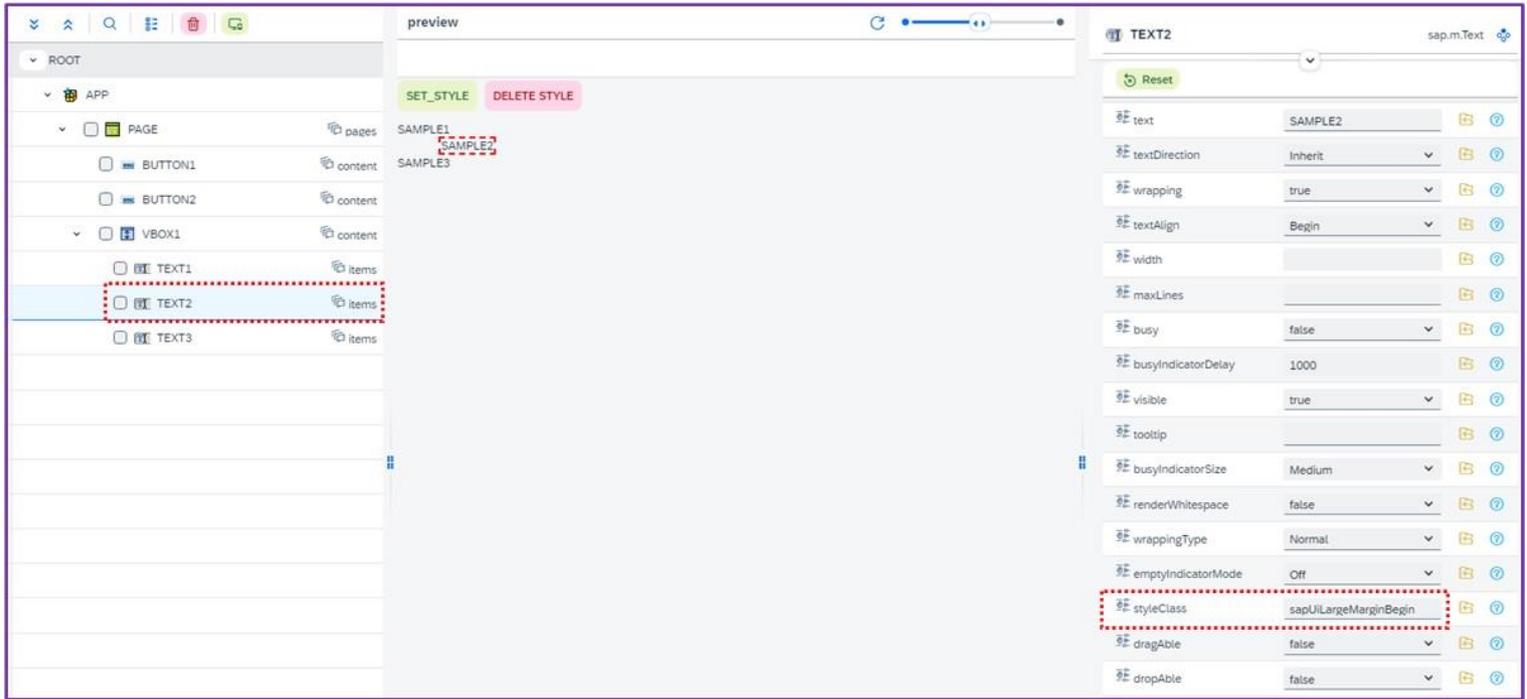
SYTLECLASS = 'sapUiLargeMarginTopBottom'. " 스타일 클래스명

ENDMETHOD.

3. Web에서 SET_STYLE Event를 작동 시키면 UI Object의 Style이 바뀌는 것을 확인 할 수 있다.(기존에 적용한 Style은 계속 남아있다.)



4. SET_STYLE는 적용된 UI Object의 Style도 삭제 할 수 있다.



5. Event 생성 후 SET_STYLE을 구성한다. 이때, ISDEL의 값을 'X'로 바꾸고 삭제하려는 Styleclass를 입력한다.

[첨부ABAP SOURCE (EV_DELETE_STYLE)]

```
METHOD EV_DELETE_STYLE.
```

```
"DELETE STYLE EVENT.
```

```
CALL METHOD /U4A/CL_UTILITIES=>SET_STYLE
```

```
EXPORTING
```

```
IO_VIEW      = ME->/U4A/IF_SERVER~AR_VIEW
```

```
ID           = 'TEXT2'
```

```
ISDEL       = 'X'
```

```
SYTLECLASS  = 'sapUiLargeMarginBegin'.
```

```
" [U4A] UI Element Super Class
```

```
" UI 오브젝트 ID
```

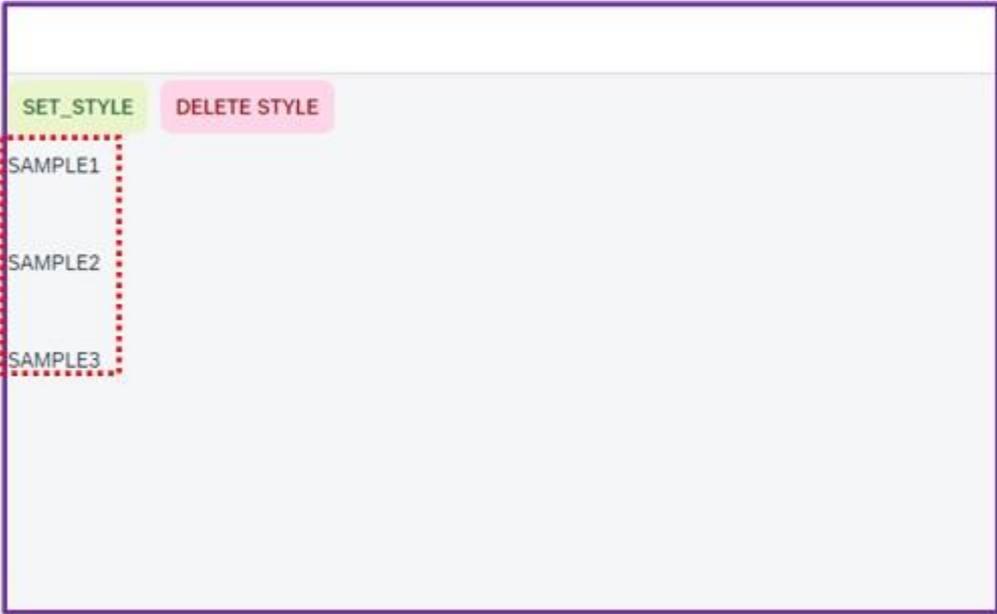
```
" 스타일 클래스명 삭제 여부(X)
```

```
" 스타일 클래스명
```

```
ENDMETHOD.
```

6. Web에서 SET_STYLE Event를 작동 시키면 UI Obj의 Style이 삭제된 것을 확인 할 수 있다.

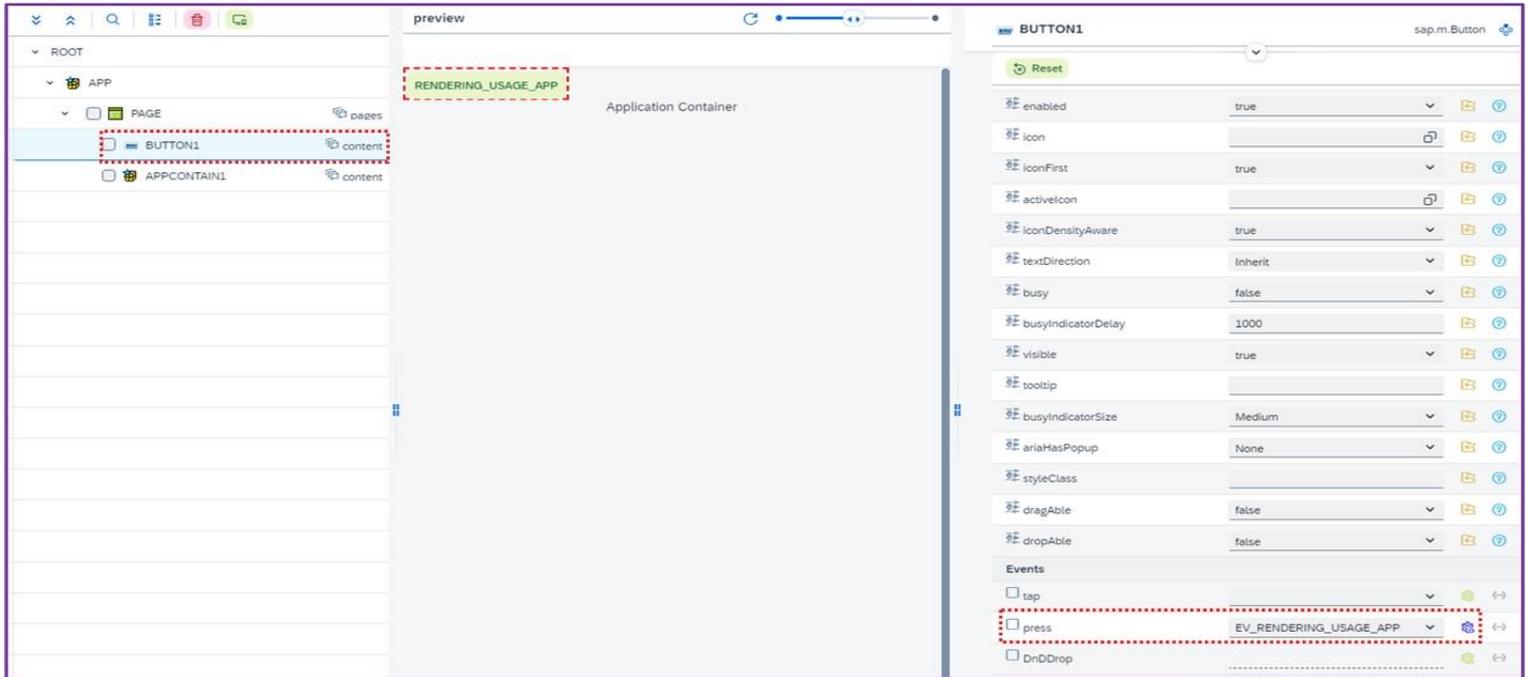




6.1.32 Methodes RENDERING_USAGE_APP

▣ 목적 : UTILITIES 중 "RENDERING_USAGE_APP"는 Application내에서 다른 Application을 호출 할 때 사용한다.

1. Application에서 UI Object와 Event를 생성한다.



2. Event 생성 후 "RENDERING_USAGE_APP"를 구성한다.

[첨부ABAP SOURCE (EV_RENDERING_USAGE_APP)]

METHOD EV_RENDERING_USAGE_APP.

"Another Application Call"

CALL METHOD /U4A/CL_UTILITIES=>RENDERING_USAGE_APP

EXPORTING

IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class

I_APP_CONTAINER_ID = 'APPCONTAIN1'

I_APPID = 'YU4A_WS_EDU11' " Application ID

* I_RESET = " Do not Use!(Deprecated)

* I_WAIT =

```
* RECEIVING
* EO_CONTROLLER =
```

```
ENDMETHOD.
```

PARAMETERS	Means
IO_VIEW	UI Element Super Class 를 입력한다
I_APP_CONTAINER_ID	호출한 Application이 담기는 UI Obj의 ID
I_APPID	호출할 Application의 ID
I_WAIT	Application을 호출하는데 걸리는 시간
EO_CONTROLLER	호출한 Application의 Instance 정보

3.Web에서 Event 작동 시 다른 Application이 호출 되는 것을 확인 할 수 있다.

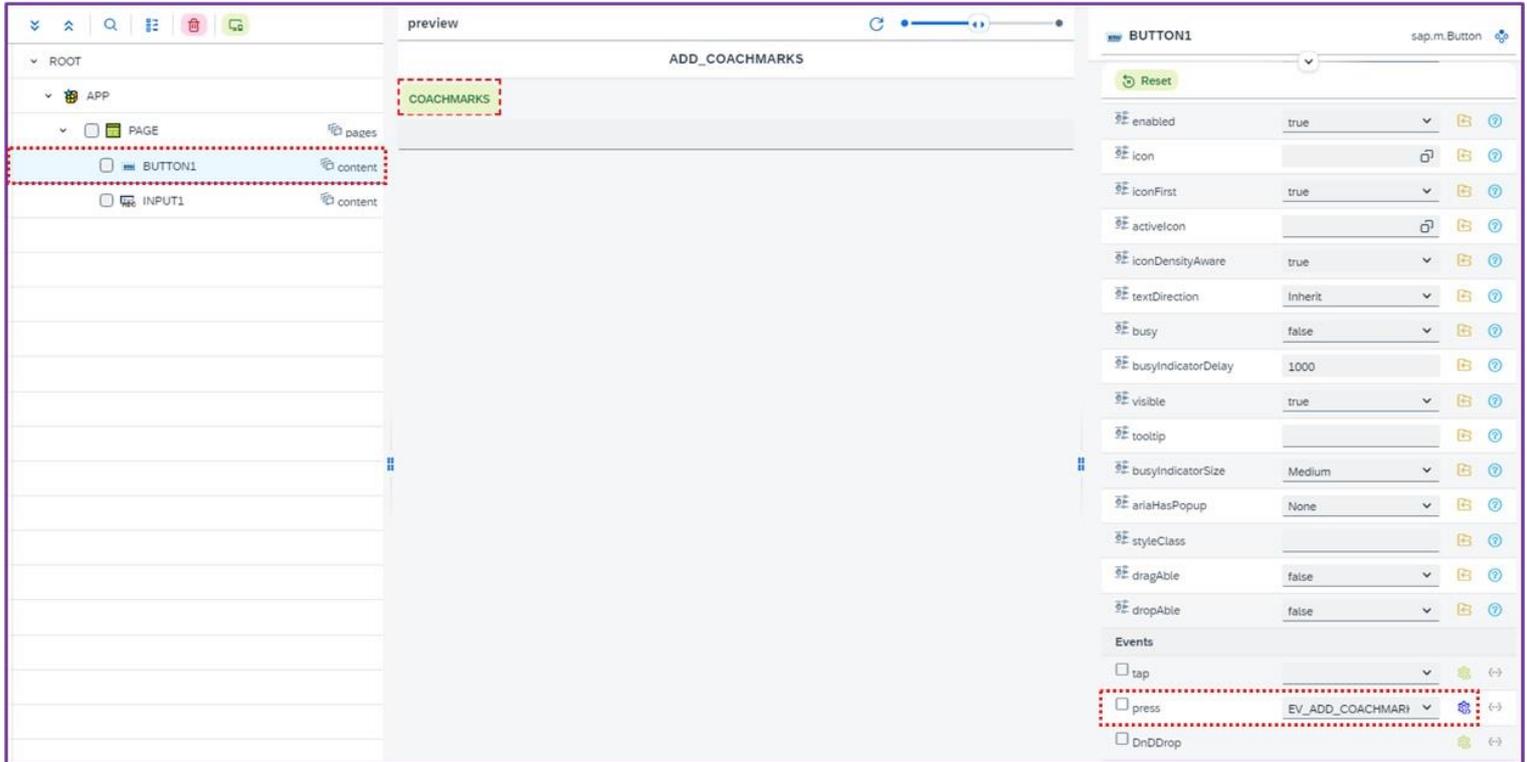
RENDERING_USAGE_APP		
GET_SELECTED_INDEX		
COL1	COL2	COL3
1	U4A WorkSpace	U4A
2	U4A WorkSpace	U4A
3	U4A WorkSpace	U4A

1. 호출할 Application의 DOCUMENT->APP의 Properties->ChgAppToNavCont 항목을 false에서 true로 변경해서 사용 하도록 한다.(변경 안할 시 UI가 잘리거나 안보이는 현상이 나타 날 수 있음)
2. 현재 RENDERING_USAGE_APP의 I_RESET은 폐기된 상태로 사용하지 않는다.

6.1.33 Methodes ADD_COACHMARKS

▣ 목적 : UTILITIES 중 "ADD_COACHMARKS"는 UI Object의 설명을 POPOVER 형태로 하나씩 알려 줄 때 사용한다.

1. Application에서 UI Object와 Event를 생성한다.



2. Event 생성 후 "ADD_COACHMARKS"을 구성한다.

[첨부ABAP SOURCE (EV_ADD_COACHMARKS)]

METHOD EV_ADD_COACHMARKS.

DATA : LT_MARKS TYPE /U4A/CL_UTILITIES=>GTT_CMARK_INFO,
LS_MARK LIKE LINE OF LT_MARKS.

"COACHMARKS의 내역 구성

```
LS_MARK-UID = 'BUTTON1'.
LS_MARK-TEXT = 'U4A WorkSpace!!!'.
APPEND LS_MARK TO LT_MARKS.
CLEAR LS_MARK.
```

```
LS_MARK-UID = 'INPUT1'.
LS_MARK-TEXT = 'U4A!!!'.
APPEND LS_MARK TO LT_MARKS.
CLEAR LS_MARK.
```

"ADD_COACHMARKS조회

```
CALL METHOD /U4A/CL_UTILITIES=>ADD_COACHMARKS
EXPORTING
  IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW
  I_WAIT  = 2000
  IT_MARKS = LT_MARKS.
```

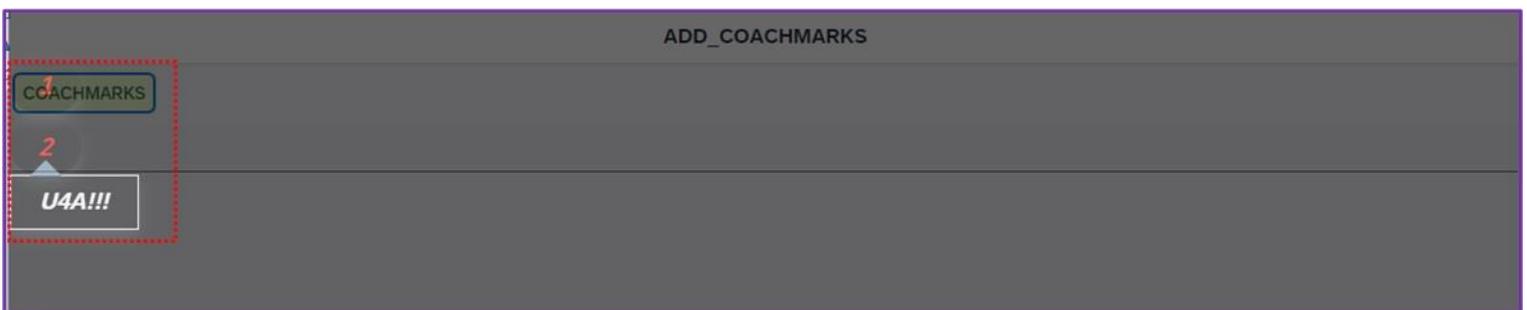
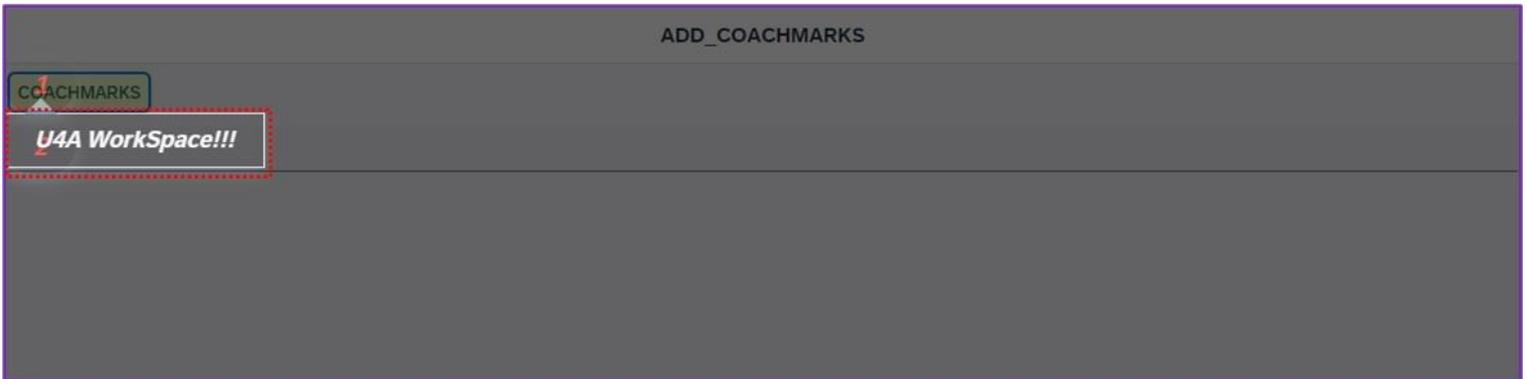
" [U4A] UI Element Super Class

" 랜더링 시간 설정 (예 : 1000 <- 1초)

" 기능 설명 처리 UI ID 및 설명 정보탭

ENDMETHOD.

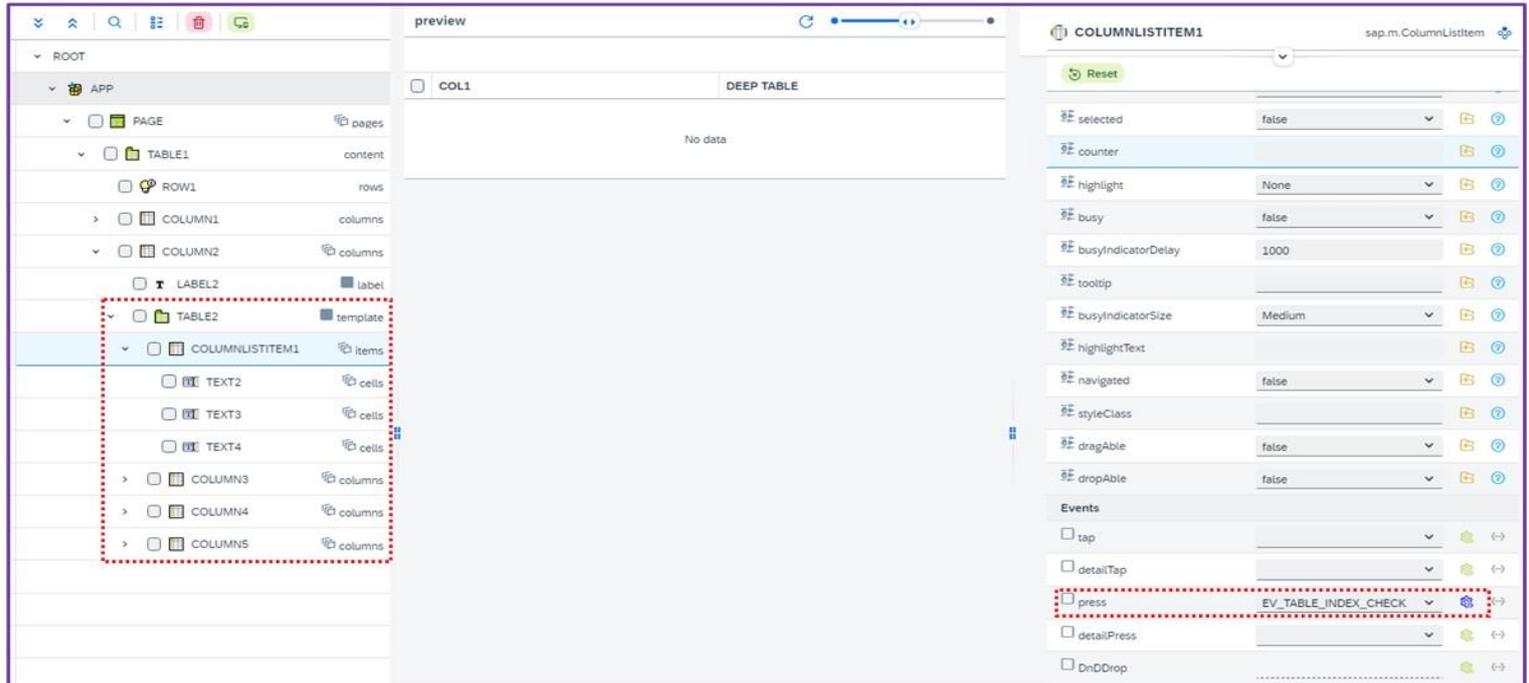
3. Web에서 ADD_COACHMARKS Event 작동시 UI Object 위쪽으로 하나씩 POPOVER가 생성된다.



6.1.34 Methodes GET_SELECTED_INDEX DEP

■ 목적 : UTILITIES 중 "GET_SELECTED_INDEX_DEP"는 TABLE DATA 중 TABLE구조의 INDEX값을 구할때 사용된다.

1. Application에서 UI Object와 Event를 생성한다.



2. Event 생성 후 "GET_SELECTED_INDEX_DEP"을 구성한다.

[첨부ABAP SOURCE (EV_TABLE_INDEX_CHECK)]

METHOD EV_TABLE_INDEX_CHECK.

```
DATA : LS_DEEP TYPE /U4A/S_TS0002,
       LS_DRATH TYPE /U4A/S0066,
       LT_DRATH TYPE /U4A/Y0066.
```

```
DATA : LV_MSG      TYPE STRING,
       LV_IDX      TYPE C LENGTH 1,
       LV_DEEPIX   TYPE C LENGTH 1.
```

"TABLE안의 TABLE의 INDEX값 추출

```
CALL METHOD /U4A/CL_UTILITIES=>GET_SELECTED_INDEX_DEP
EXPORTING
  IO_CONTROLLER = ME
  I_EVENT_NAME  = I_EVENT_NAME
IMPORTING
  ES_SEL_LINE   = LS_DEEP      " 선택 라인 INDEX의 STRUCTURE
  ET_DPATH      = LT_DRATH.    " [U4A] Dep Structure binding update path
```

"첫번째 TABLE IDX.

```
READ TABLE LT_DRATH INTO LS_DRATH WITH KEY TABNM = 'GT_OTAB'.
CHECK SY-SUBRC EQ 0.
```

```
LV_IDX = LS_DRATH-LINE_IDX.
```

```
CLEAR LS_DRATH.
```

"DEEP TABLE IDX.

```
READ TABLE LT_DRATH INTO LS_DRATH WITH KEY TABNM = 'DEEP'.
CHECK SY-SUBRC EQ 0.
```

```
LV_DEEPIX = LS_DRATH-LINE_IDX.
```

"MESSAGE 내역

```
CONCATENATE `Table Index Value is ` LV_IDX`,` CL_ABAP_CHAR_UTILITIES=>NEWLINE
            ` Deep Table Index Value is ` LV_DEEPIX `.` CL_ABAP_CHAR_UTILITIES=>NEWLINE
            `The corresponding index holds the values of ` LS_DEEP-FLD01 `, ` LS_DEEP-
FLD02 `, ` LS_DEEP-FLD03 `.`
            INTO LV_MSG.
```

"MESSAGE 호출

```
CALL METHOD /U4A/CL_UTILITIES=>M_MESSAGE_TOAST
EXPORTING
  IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class
*      IS_MSG = " Structure of message variables
```

U4A IDE HELP

```

I_MSGTX = LV_MSG           " MESSAGE TEXT
*   IT_MSGTX =              " Table of Strings
*   I_VPOS  = CS_M_MSG_TOS_VPOS-CENTER " Vertical position
*   I_HPOS  = CS_M_MSG_TOS_HPOS-CENTER " Horizontal position
*   I_VOFF  = '0'           " Vertical offset
*   I_HOFF  = '0'           " Horizontal offset
*   I_DURATION = 10000
.
ENDMETHOD.
    
```

3. Web에서 Event 작동시 TABLE의 INDEX 정보와 어떤 DATA가 담겼는지 확인 할 수 있다.

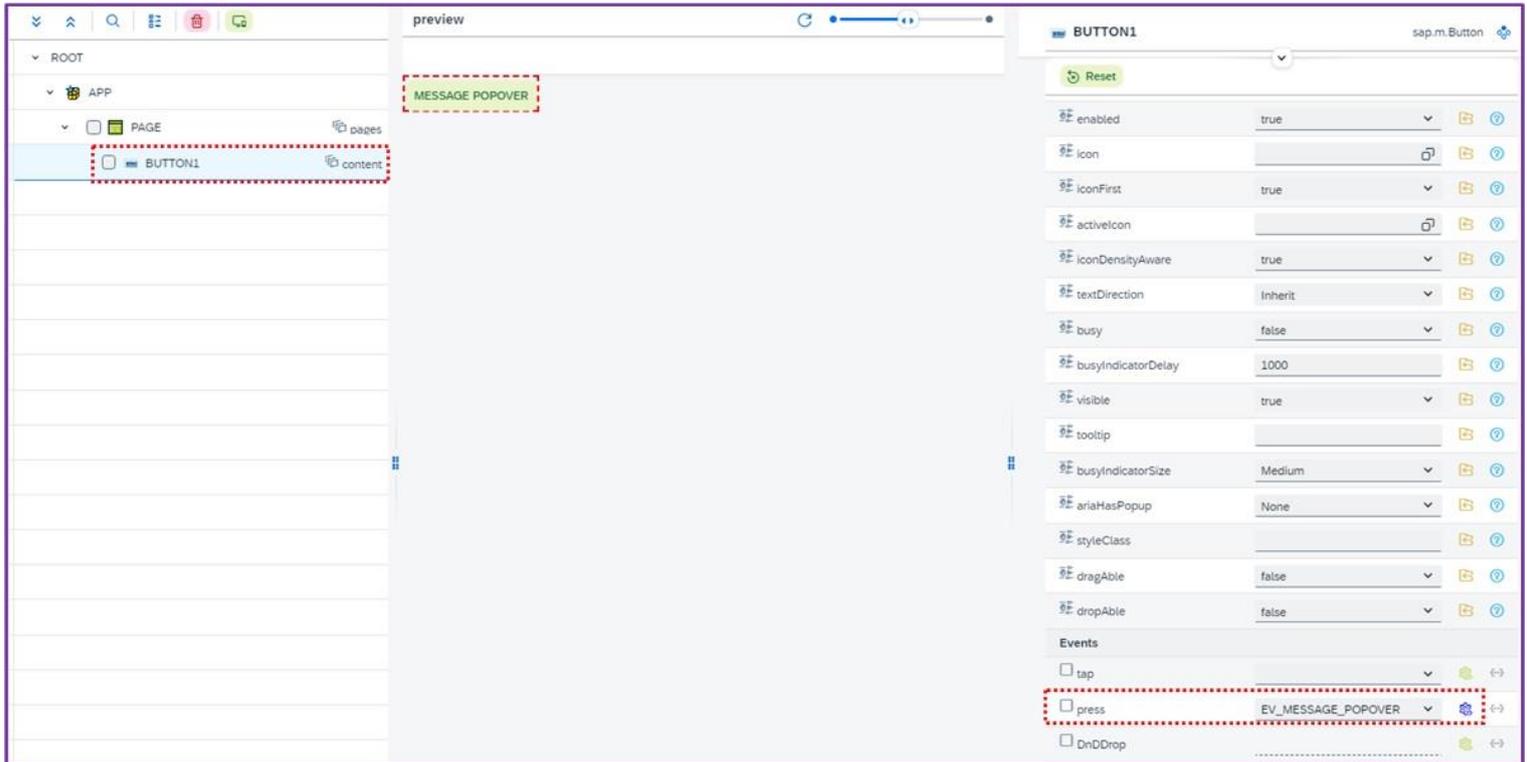
<input type="checkbox"/> COL1	DEEP TABLE		
	DEEP1	DEEP2	DEEP3
<input type="checkbox"/> U4A	U4A_1	U4A_2	U4A_3
		U4A_5	U4A_6
<input type="checkbox"/> U4A_WS		DEEP2	DEEP3
		U4A_WS2	U4A_WS3
	U4A_WS4	U4A_WS5	U4A_WS6
	DEEP1	DEEP2	DEEP3
<input type="checkbox"/> U4A_TEST	U4A_TEST1	U4A_TEST2	U4A_TEST3
	U4A_TEST4	U4A_TEST5	U4A_TEST6

Table Index Value is 2,
Deep Table Index Value is 2.
The corresponding index holds
the values of U4A_WS4,
U4A_WS5, U4A_WS6.

6.1.35 Methodes MESSAGE_POPOVER

■ 목적 : UTILITIES 중 "MESSAGE_POPOVER"은 UI Object 근처에 POPOVER 창을 호출할 때 사용한다.

1. Application에서 UI Object와 Event를 생성한다.



2. Event 생성 후 "MESSAGE_POPOVER"를 구성한다.

[첨부ABAP SOURCE (EV_MESSAGE_POPOVER)]

METHOD EV_MESSAGE_POPOVER.

DATA : LT_MSG TYPE /U4A/CL_UTILITIES=>GTT_MSGPOPOVER,
LS_MSG LIKE LINE OF LT_MSG.

"MESSAGE_POPOVER 내역"

```

LS_MSG-TYPE      = 'I'.           "Message 타입 E,W,I,S
LS_MSG-COUNTER   = 1.             "ROW counter
LS_MSG-M_TITLE   = 'U4A'.         "Message title
LS_MSG-M_SUBTITLE = 'Workspace'. "Message subtitle
    
```

```

APPEND INITIAL LINE TO LS_MSG-M_LONGMSG.
APPEND 1 TO LS_MSG-M_LONGMSG.
APPEND 2 TO LS_MSG-M_LONGMSG.
APPEND 3 TO LS_MSG-M_LONGMSG. "Message 내용
APPEND LS_MSG TO LT_MSG.
    
```

"MESSAGE_POPOVER 호출

```
CALL METHOD /U4A/CL_UTILITIES=>MESSAGE_POPOVER
```

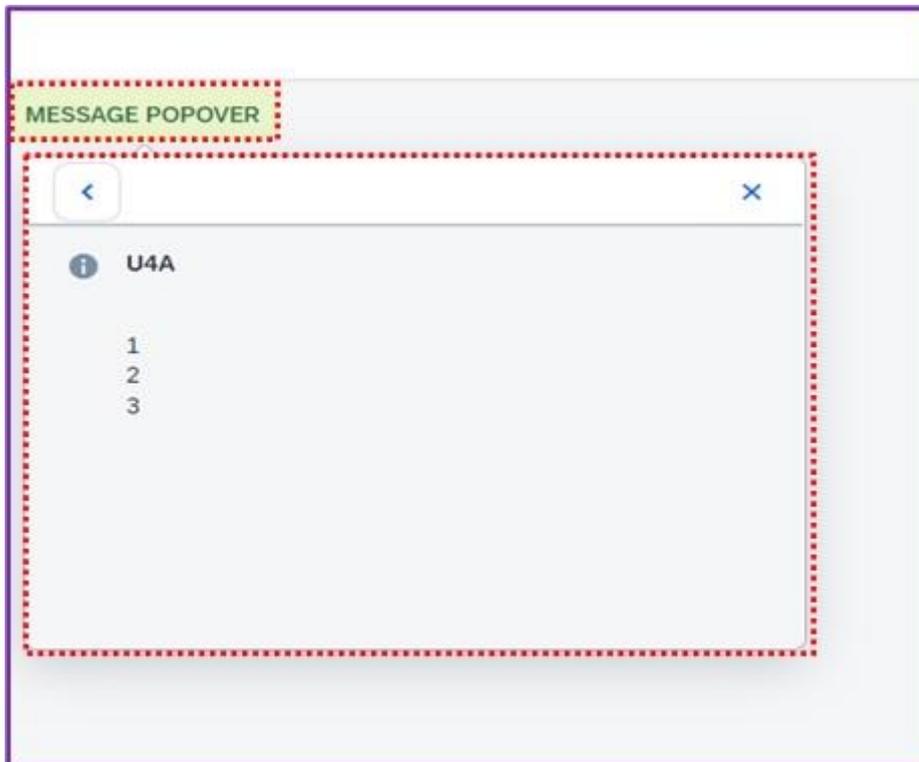
```
EXPORTING
```

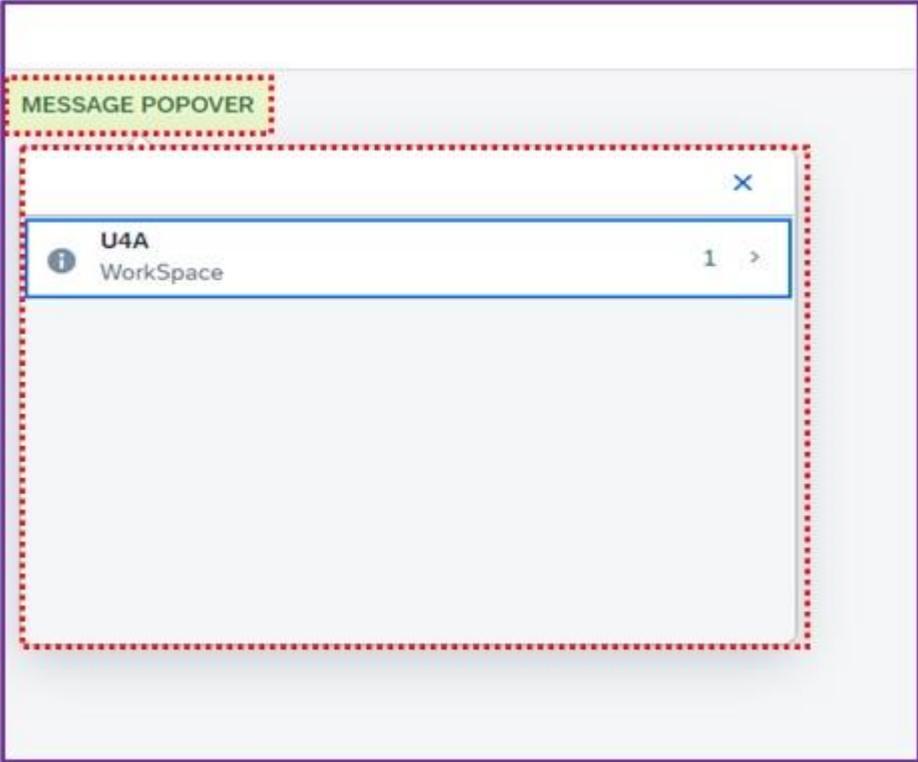
```

IO_VIEW      = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class
I_OPENBY_ID  = 'BUTTON1'                  " 메시지 팝업 위치 UI 인스턴스 ID
I_CLOSE_TIME = 10                          " 메시지 팝업 자동 종료 : 1 <-- 1초
IT_MSG_INFO  = LT_MSG.                     " 메시지 처리 컨피그
    
```

```
ENDMETHOD.
```

3. Web에서 Event 작동시 POPOVER가 지정한 UI Object근처에 호출된 것을 확인 할 수 있다.

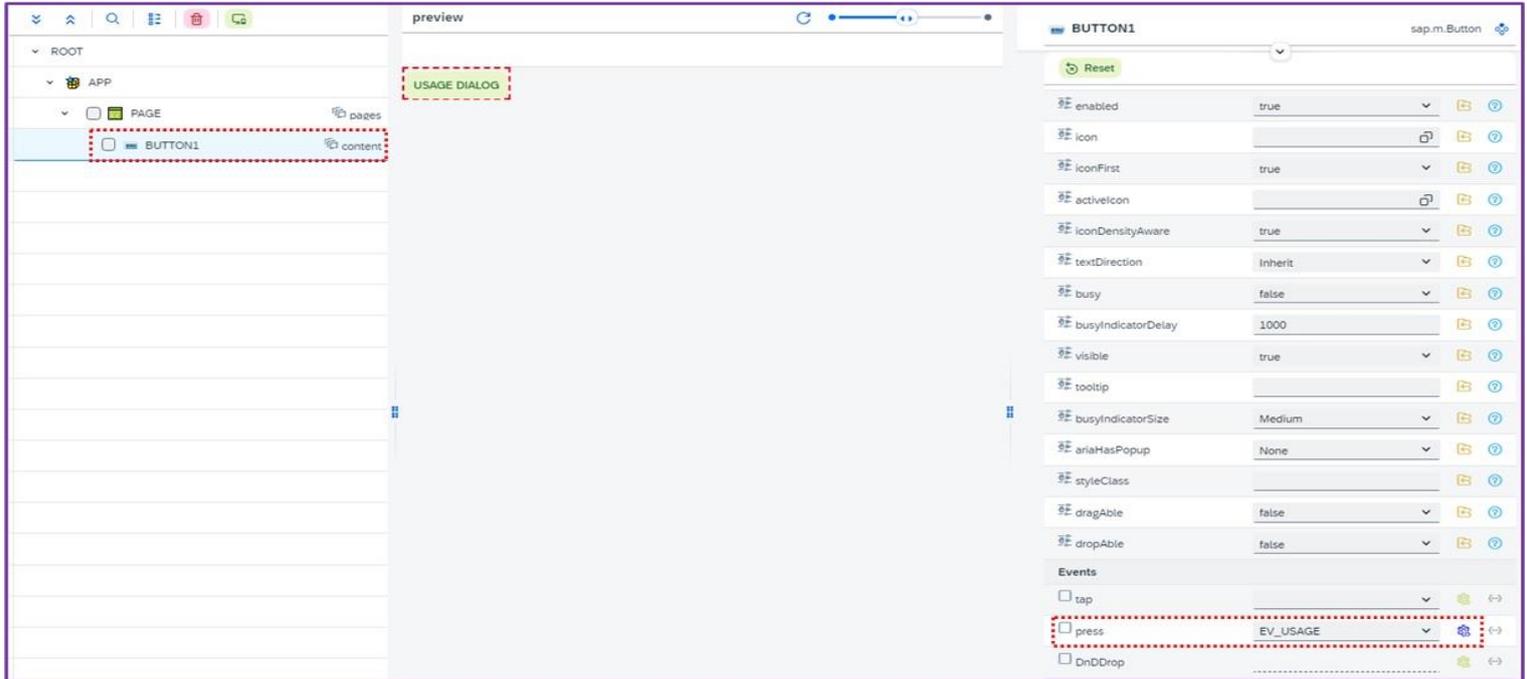




6.1.36 Methodes USAGE_DIALOG

■ 목적 : " DIALOG USAGE"는 Application 내에서 다른 Application 을 DIALOG 형태로 호출 할 수 있는 Methodes 이다.

1. Applicaiton에서 UI Object와 Event를 생성한다.



2. 하단의 그림과 같은 Call methodes로 불러온 후 DIALOG설정, 불러올 Application명을 입력한다.(EO_CONTROLLER를 통해 생성된 Application의 Controller 정보를 얻을 수 있다.)

[첨부 ABAP SOURCE (EV_USAGE)]

```
METHOD EV_USAGE.
```

```
DATA : LS_CONF TYPE /U4A/S0084,  
       LS_BTN  TYPE /U4A/S0085.
```

"DIALOG 구성

```

LS_CONF-ICON           = 'sap-icon://activity-individual'. "DIALOG HEADER에 ICON
LS_CONF-TITLE          = 'U4A_TEST'. "DIALOG HEADER의 TITLE 명
LS_CONF-CLOSE_EVT      = 'EV_USAGE_CLOSE'. "DIALOG 닫기 EVENT
LS_CONF-RESIZABLE      = 'X'. "DIALOG의 크기 조절 여부
LS_CONF-DRAGGABLE      = 'X'. "DIALOG의 이동 가능 여부
LS_CONF-WIDTH          = '60%'. "DIALOG의 가로 길이
LS_CONF-HEIGHT         = '70%'. "DIALOG의 세로 길이
LS_CONF-STYLECLASS     = 'sapUiSizeCompact'. "DIALOG에 적용할 CSS Class
LS_CONF-HIDE_CLOSE_BTN = ''. "닫기 버튼 비활성화 여부
LS_CONF-HIDE_HEAD      = ''. "HEADER TOOLBAR 비활성
LS_CONF-SHOW_MAXIMIZE_BTN = ''. "최대화 버튼 활성화(X = 활성화)
LS_CONF-ESC_CLOSE      = ''.
    
```

"DIALOG 버튼 추가

```

LS_BTN-TEXT           = 'DIALOG CLOSE'. "DIALOG BUTTON TEXT
LS_BTN-ICON           = 'sap-icon://decline'. "DIALOG BUTTON ICON
LS_BTN-WIDTH          = '250px'. "DIALOG BUTTON 크기 설정
LS_BTN-TYPE           = '03'. "DIALOG BUTTON 스타일 결정
LS_BTN-EVENT          = ''. "DIALOG BUTTON 클릭시 호출 EVENT 설정
APPEND LS_BTN TO LS_CONF-BTN_CONF.
CLEAR LS_BTN.
    
```

"Application을 DIALOG 형태로 불러오기

```

CALL METHOD /U4A/CL_APP_USAGE_DIALOG=>USAGE_APP_OPEN_DIALOG
EXPORTING
    IO_VIEW           = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class
    IS_CONFIG         = LS_CONF " [U4A] Applicaton Usage Dialog Config Info
    I_APPID           = 'YU4A_WS_EDU14' " Application ID
* RECEIVING
* EQ_CONTROLLER      = " Application Controller Instance
.
    
```

ENDMETHOD.

※ 만약 USAGE의 Instance 정보를 가져오고 싶으면 GET_APP_CONTROLLER를 이용한다 .

[첨부ABAP SOURCE (EV_USAGE)]

DATA : LO_APP TYPE REF TO YCL_U4A_APP_U4A_WS_EDU14.

"Application에서 USAGE의 Instance 정보를 가져올 때 사용한다.

```
LO_APP ?= /U4A/CL_APP_USAGE_DIALOG=>GET_APP_CONTROLLER(
    IO_VIEW      = ME->/U4A/IF_SERVER~AR_VIEW
    I_APPID      = 'YU4A_WS_EDU14'
).
```

3. DIALOG의 닫기 Button 선택 시 Server Event를 호출하려면 LS_CONF-CLOSE_EVT에 DIALOG 종료 EVENT를 추가한다.

```
7  "DIALOG 구성
8  LS_CONF-ICON      = 'sap-icon://activity-individual'. "DIALOG HEADER에 ICON
9  LS_CONF-TITLE     = 'U4A_TEST' "DIALOG HEADER의 TITLE 명
10 LS_CONF-CLOSE_EVT = 'EV_USAGE_CLOSE'. "DIALOG 닫기 클릭 시 작동 EVENT
11 LS_CONF-RESIZABLE = 'X'. "DIALOG의 크기 조절 여부
12 LS_CONF-DRAGGABLE = 'X'. "DIALOG의 이동 가능 여부
13 LS_CONF-WIDTH     = '60%'. "DIALOG의 가로 길이
14 LS_CONF-HEIGHT    = '70%'. "DIALOG의 세로 길이
15 LS_CONF-STYLECLASS = 'sapUiSizeCompact'. "DIALOG에 적용할 CSS Class
16 LS_CONF-HIDE_CLOSE_BTN = ''. "닫기 버튼 비활성화 여부
17 LS_CONF-HIDE_HEAD = ''. "HEADER TOOLBAR 영역 비활성 여부
18 LS_CONF-SHOW_MAXIMIZE_BTN = ''. "최대화 버튼 활성화여부 ('X' = 활성화)
19 LS_CONF-ESC_CLOSE = ''. "Esc로 팝업 종료 여부 ('X' = Esc로 닫힘)
```

Method	Level	Visibility	M...	Description
/U4A/IF_SERVER~HANDL_ON_INIT	Instance Method	Public		Initialization event
/U4A/IF_SERVER~HANDL_ON_REQUEST	Instance Method	Public		Asynchronous event
/U4A/IF_SERVER~HANDL_ON_RESPONSE	Instance Method	Public		Not used
/U4A/IF_SERVER~HANDL_ON_EXIT	Instance Method	Public		Close event
/U4A/IF_SERVER~HANDL_ON_MESSAGE	Instance Method	Public		Push Channel Services Web Socket message receive event
/U4A/IF_SERVER~HANDL_ON_HASHCHANGE	Instance Method	Public		Router hash change(browser의 뒤로가기) callback event
CONSTRUCTOR	Instance Method	Public		
EV_USAGE	Instance Method	Public		[EVENT] USAGE DIALOG
EV_USAGE_CLOSE	Instance Method	Public		[EVENT] USAGE DIALOG CLOSE

4. 현재 호출된 DIALOG를 종료할 시 아래 Method를 이용한다.

[첨부ABAP SOURCE (EV_USAGE_CLOSE)]

METHOD EV_USAGE_CLOSE.

"USAGE DIALOG CLOSE.

CALL METHOD /U4A/CL_APP_USAGE_DIALOG=>DIALOG_CLOSE

EXPORTING

IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class

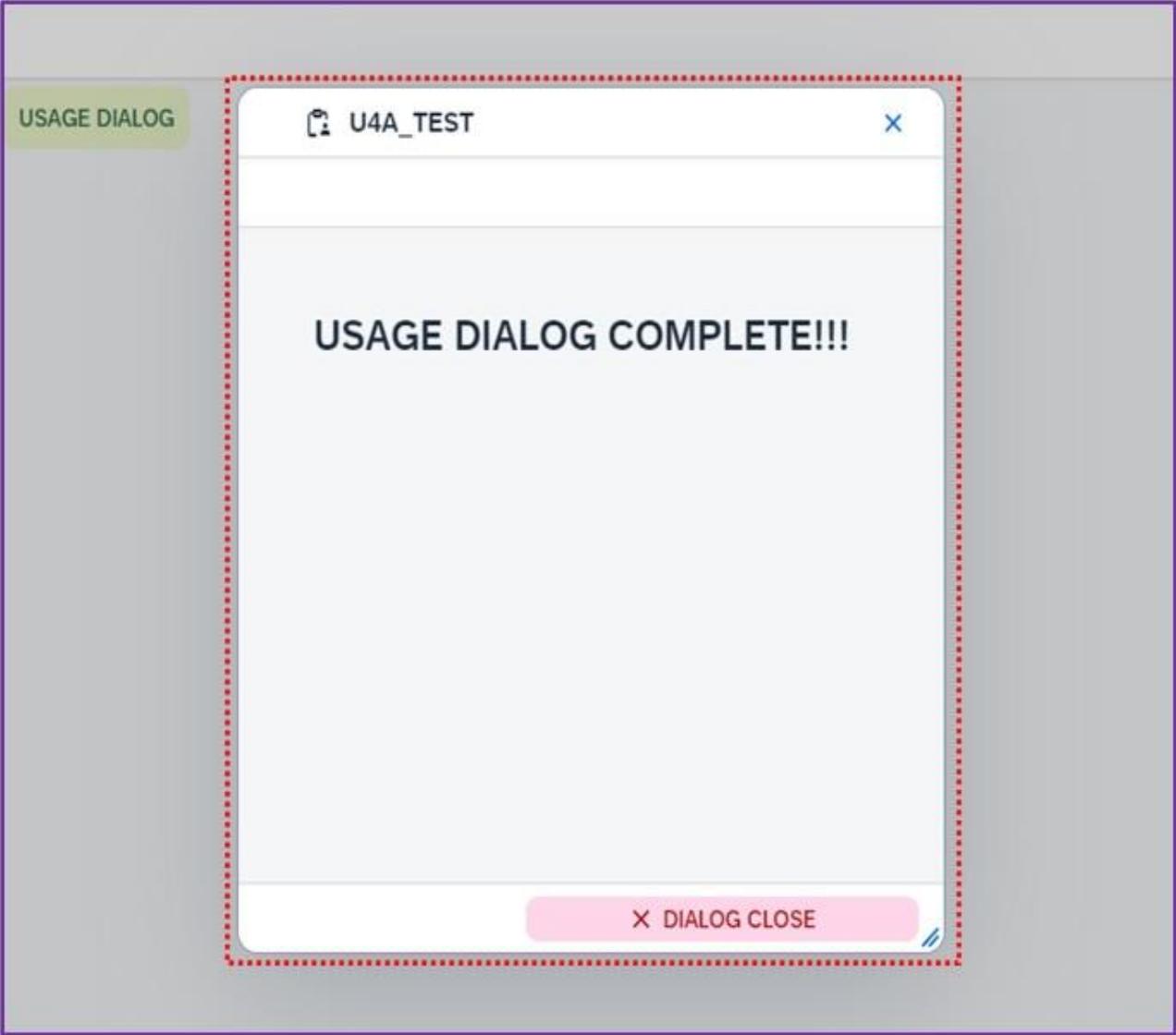
I_APPID = 'YU4A_WS_EDU14' " Application ID

I_APP_RESET = 'X' . " Usage Application Insage 초기화 여부

('X' = 초기화)

ENDMETHOD.

5. Web에서 DIALOG Event를 실행 시키면 DIALOG창에 다른 Application이 호출된다.



6.1.37 Methodes RUN_ASYNC_ON_REQUEST

※ " RUN_ASYNC_ON_REQUEST"는 Application의 비동기 실행 Methodes이다.

▣ 목적 : 동기실행으로 모든 데이터를 수집하면 동기실행이 끝날때까지 다른 행동을 할 수 없다. 이때 동기 + 비동기를 같이 실행하면 이와 같은 상황을 해결 가능함.

1. Application에서 UI Object와 Event를 생성한다.

The screenshot displays the U4A IDE interface during the execution of the 'RUN_ASYNC_ON_REQUEST' method. The interface is divided into several sections:

- Project Tree (Left):** Shows a hierarchical view of the application structure, including 'APP', 'PAGE', 'TOOLPAGE1', 'TOOLHEADER1', 'SIDENAVIGATION1', 'FORM1', 'HBOX1', 'PAGE1', 'VBOX1', 'TITLE1', and 'TITLE2'.
- Preview Window (Center):** Displays the 'RUN_ASYNC_ON_REQUEST' method execution. It includes a 'Toggle Collapse/Expand' button and a list of pages: 'PAGE1' and 'PAGE2'.
- Data Collection Form (Right):** Titled '동기 상태 데이터 수집' (Synchronous State Data Collection), it contains four input fields labeled 'INPUT1', 'INPUT2', 'INPUT3', and 'INPUT4'.
- Chart (Right):** Titled '차트' (Chart), it shows a bar chart with three samples: 'sample01', 'sample02', and 'sample03'. The y-axis ranges from 5 to 35. The chart includes two data series: 'graph1' (blue bars) and 'graph2' (purple line). The data points are approximately:

Sample	graph1 (Bar)	graph2 (Line)
sample01	10	30
sample02	20	20
sample03	30	10

2. Form UI, Chart UI를 모두 "동기실행"으로 데이터를 구성 하고 Application 실행.

[첨부ABAP SOURCE (/U4A/IF_SERVER~HANDL_ON_INIT)]

```
METHOD /U4A/IF_SERVER~HANDL_ON_INIT.
```

```
*--동기로 데이터 수집.
```

```
"Form UI.
```

```
GS_FORM-F1 = 'U4A'.
```

```
GS_FORM-F2 = 'U4A WorkSpace'.
```

```
GS_FORM-F3 = '중요 항목'.
```

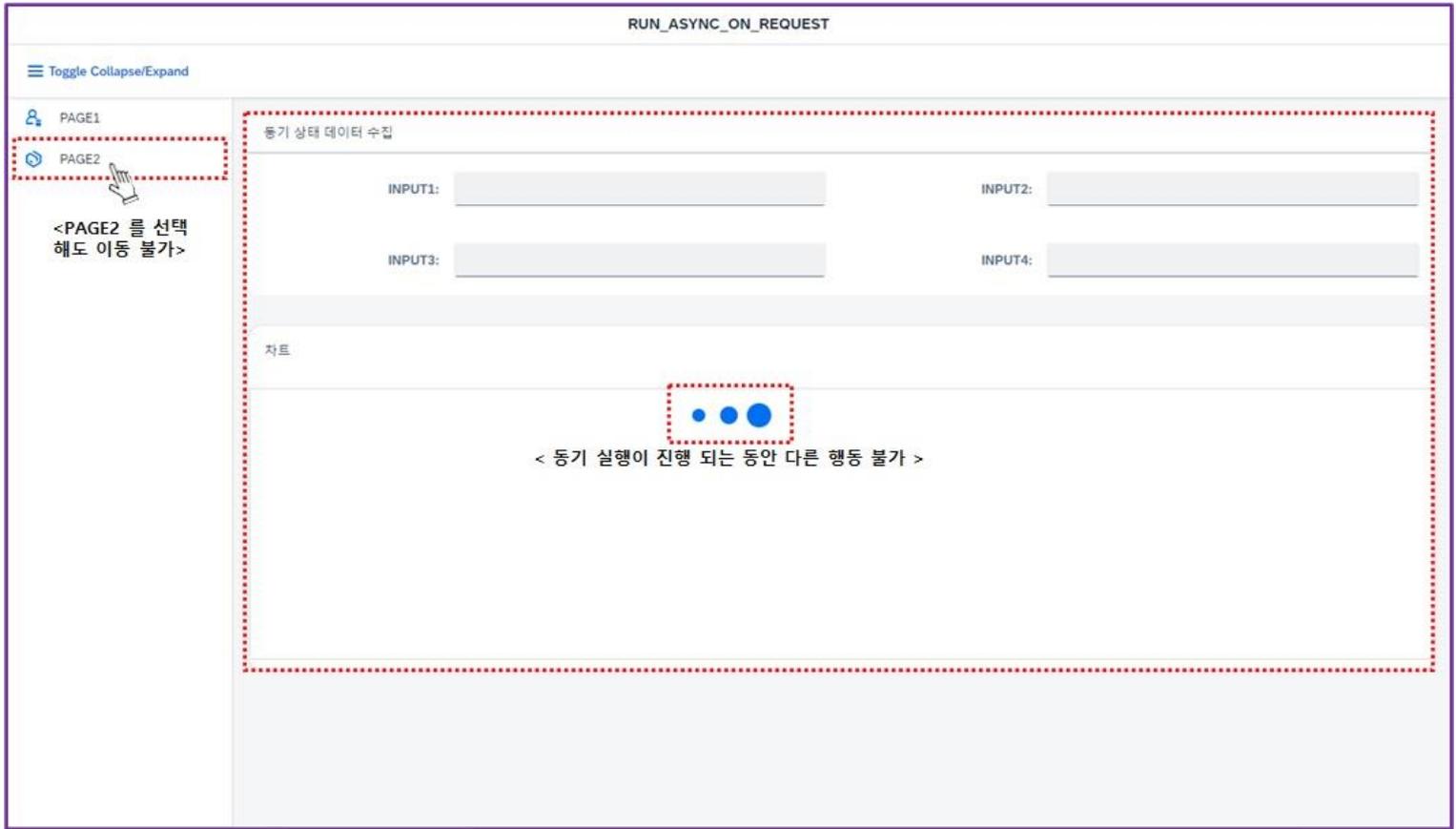
```
GS_FORM-F4 = '동기상태 데이터 수집'.
```

```
"차트 구성.
```

```
CALL METHOD SET_CHART.
```

```
ENDMETHOD.
```

① 동기 실행이 끝나기전에는 다른 행동을 불가 .(PAGE2 선택 불가능)



② 동기 실행이 완전히 진행된 후 다른 행동 가능 (PAGE2 선택 가능)

U4A IDE HELP

RUN_ASYNC_ON_REQUEST

Toggle Collapse/Expand

PAGE1

PAGE2

<PAGE2 를 선택하면 페이지 이동>

동기 상태 데이터 수집

INPUT1: U4A

INPUT2: U4A WorkSpace

INPUT3: 중요 항목

INPUT4: 동기상태 데이터 수집

차트



RUN_ASYNC_ON_REQUEST

비동기 진행 중 페이지 이동 가능
동기 진행 중 페이지 이동 불가능

3. Form UI => "동기실행", 으로 CHART UI => "비동기 실행" 데이터를 구성 하고 Application 실행.

[첨부ABAP SOURCE (/U4A/IF_SERVER~HANDL_ON_INIT)]

```
METHOD /U4A/IF_SERVER~HANDL_ON_INIT.
```

*--동기 실행.

```
GS_FORM-F1 = 'U4A'.
```

GS_FORM-F2 = 'U4A Workspace'.
 GS_FORM-F3 = '중요 항목'.
 GS_FORM-F4 = '동기상태 데이터 수집'.

*--비동기 실행.

CALL METHOD /U4A/CL_UTILITIES=>RUN_ASYNC_ON_REQUEST

EXPORTING

IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class

* I_PRCCD = " Callback Event에 전달받을 process code

* IT_PARAM = " HTTP Framework (iHTTP) Table Name/Value Pairs

* I_IGNORE_ERR = " 비동기에서 발생한 덤프 오류 무시 여부('X' = 덤프오류 무시함)

* I_SYNC_DYN_UI = " 현재 동적생성한 UI를 비동기 시점에 적용 여부 ('X' = 비동기 시점에 적용함)

I_FINISH_SEVT = 'EV_FINISH_ASYNC'. " 비동기 처리 완료 후 sync session에 호출

할 Callback Server Event 명

* IT_FINISH_CEVT = " 비동기 처리 완료 후 수행할 Client Event script

* I_USE_WAIT = 'X' " 비동기 처리 완료 후 Server Event 호출시 WAIT 사용 여부('X' = 사용함)

"메세지 호출

CALL METHOD /U4A/CL_UTILITIES=>M_MESSAGE_TOAST

EXPORTING

IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class

I_MSGTX = '비동기 상태 차트 그리는중입니다'. " MESSAGE TEXT\

ENDMETHOD.

[첨부ABAP SOURCE (/U4A/IF_SERVER~HANDL_ON_REQUEST)]

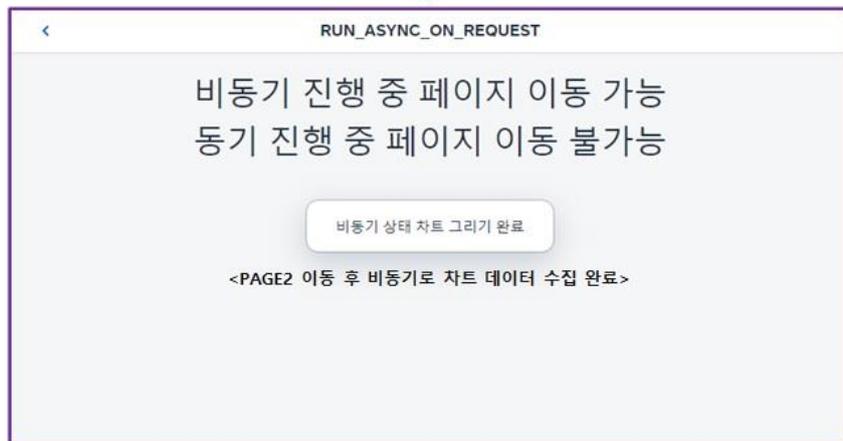
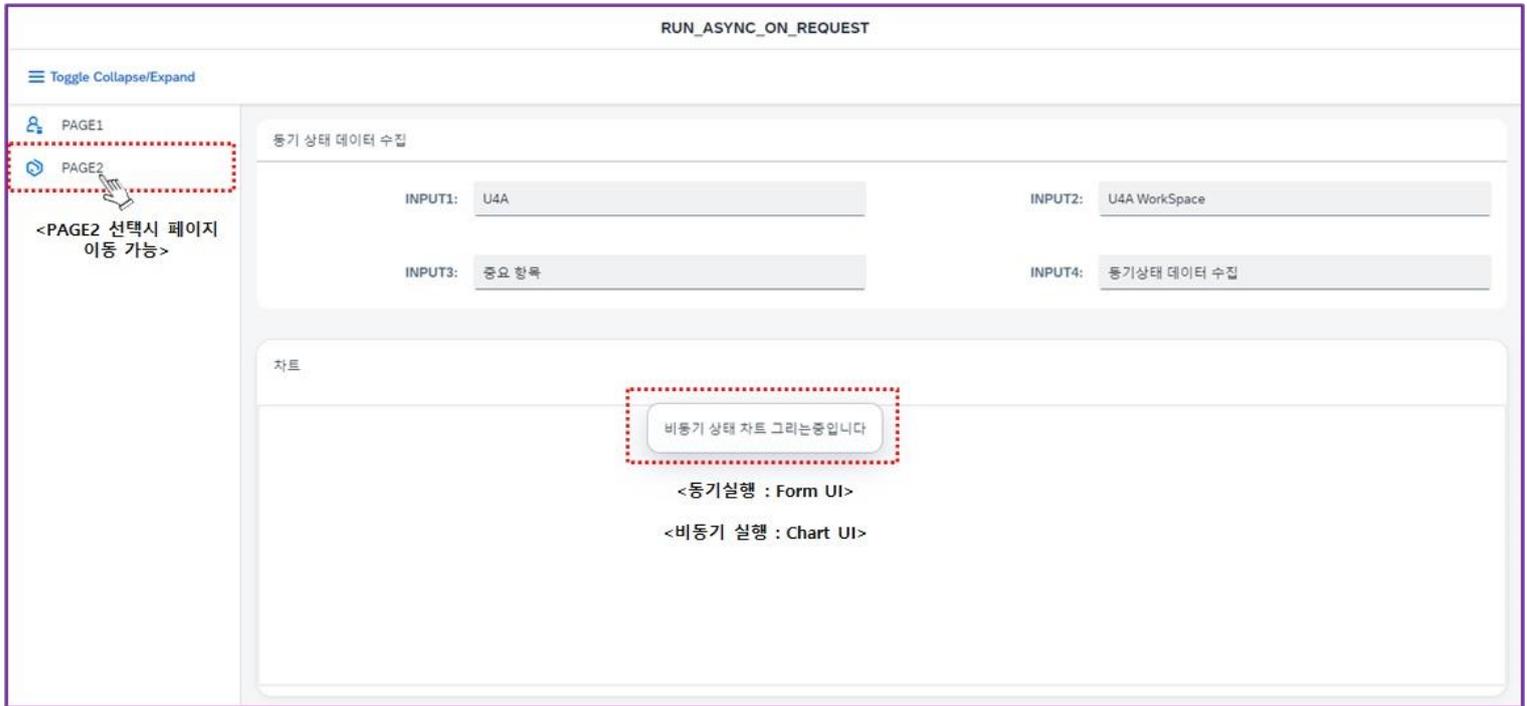
METHOD /U4A/IF_SERVER~HANDL_ON_REQUEST.

"차트 구성

CALL METHOD SET_CHART.

ENDMETHOD.

- ① 동기실행 완료 + 비동기 실행 "미완료" 상태여도 다른 행동이 가능하다 (PAGE2 선택 가능)



- ② 다시 차트 화면으로 이동 했을때 비동기 실행으로 구성된 차트 데이터가 구성된 것을 확인.

U4A IDE HELP

RUN_ASYNC_ON_REQUEST

☰ Toggle Collapse/Expand

PAGE1

PAGE2

통기 상태 데이터 수집

INPUT1: U4A

INPUT3: 중요 항목

INPUT2: U4A WorkSpace

INPUT4: 통기상태 데이터 수집

차트

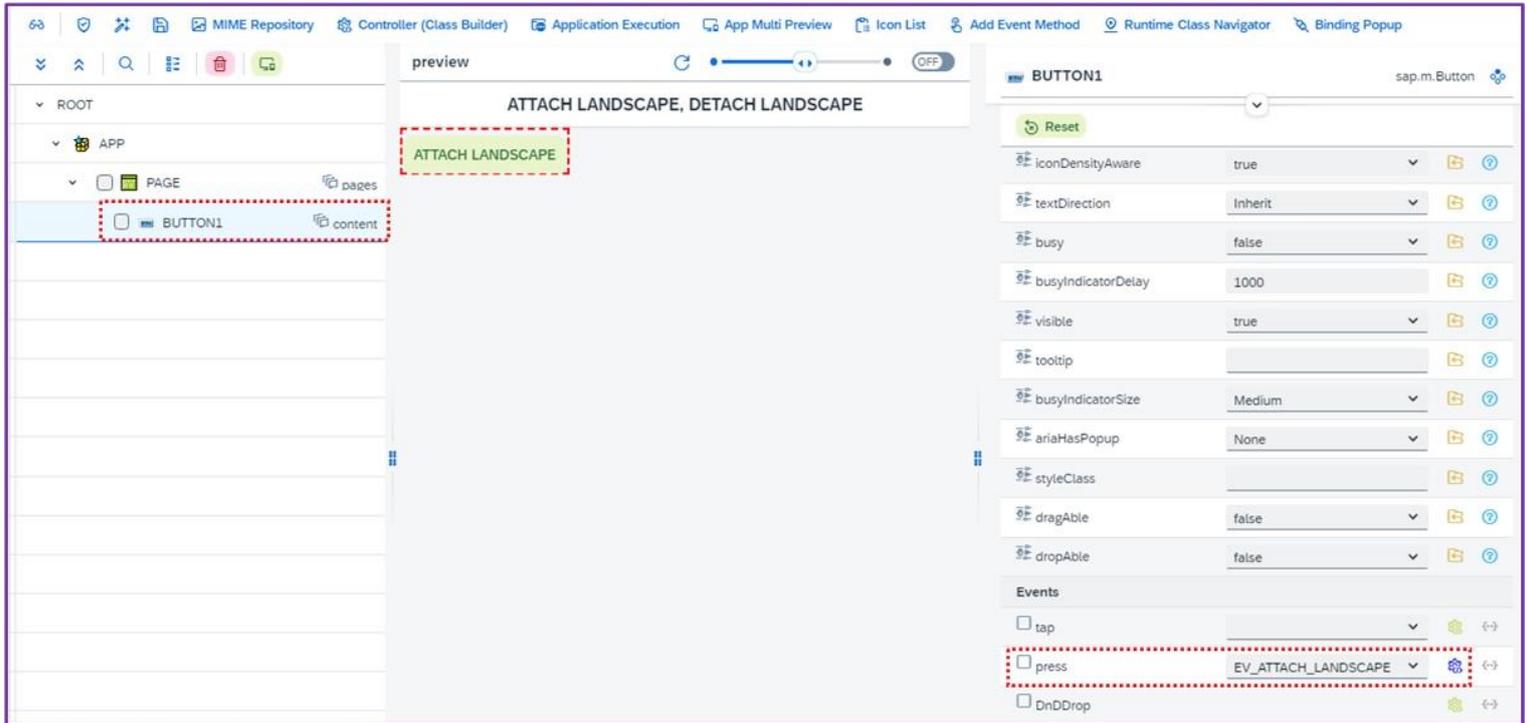
날짜	Series 1 (Pink)	Series 2 (Orange)
JAN 12	3,000,000	3,000,000
JAN 13	6,000,000	6,000,000
JAN 14	9,000,000	9,000,000
JAN 15	12,000,000	12,000,000
JAN 16	15,000,000	15,000,000
JAN 17	3,000,000	3,000,000
JAN 18	6,000,000	6,000,000
JAN 19	9,000,000	9,000,000
JAN 20	12,000,000	12,000,000
JAN 21	15,000,000	15,000,000

547

6.1.38 Methodes ATTACH_LANDSCAPE

■ 목적 : Application 화면 회전시 이벤트 처리 (PC, Mobile, Tablet 환경에서 모두 사용 가능)

1. Application에서 화면 회전 이벤트를 설정 하기 위한 UI Object 및 Event 생성 (Button 사용)



2. Application 실행시 화면 회전 이벤트 설정 (Methodes : ATTACH_LANDSCAPE)

[첨부ABAP SOURCE (/U4A/IF_SERVER~HANDL_ON_INIT)]

```
METHOD /U4A/IF_SERVER~HANDL_ON_INIT.
```

"화면 회전시 이벤트 설정"

```
CALL METHOD /U4A/CL_UTILITIES=>ATTACH_LANDSCAPE
EXPORTING
```

```

IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class
* IT_CEVT = " Client Event Script
I_SEVT = 'EV_ATTACH_LANDSCAPE'. " Server Event name
* I_WAIT_OFF =. " 'X' = wait off

```

```
ENDMETHOD.
```

3. 화면 회전 이벤트 내역 설정(본 단원에서는 가로,세로 회전 시 메시지 호출)

[첨부ABAP SOURCE (EV_ATTACH_LANDSCAPE)]

```
METHOD EV_ATTACH_LANDSCAPE.
```

```
DATA LV_MSG TYPE STRING.
```

```
DATA LS_FORM TYPE IHTTNPV.
```

```
*-----*
```

```
READ TABLE IT_FORM_DATA INTO LS_FORM WITH KEY NAME = 'landscape'.
```

```
CHECK SY-SUBRC EQ 0.
```

"화면 회전에 따른 메시지 내역 설정

```
CASE LS_FORM-VALUE.
```

```
WHEN ABAP_TRUE. "세로 -> 가로 회전
```

```
LV_MSG = '가로 모드입니다'.
```

```
WHEN ABAP_FALSE. "가로 -> 세로 회전
```

```
LV_MSG = '세로 모드입니다'.
```

```
ENDCASE.
```

"메시지 호출

```
CALL METHOD /U4A/CL_UTILITIES=>M_MESSAGE_TOAST
```

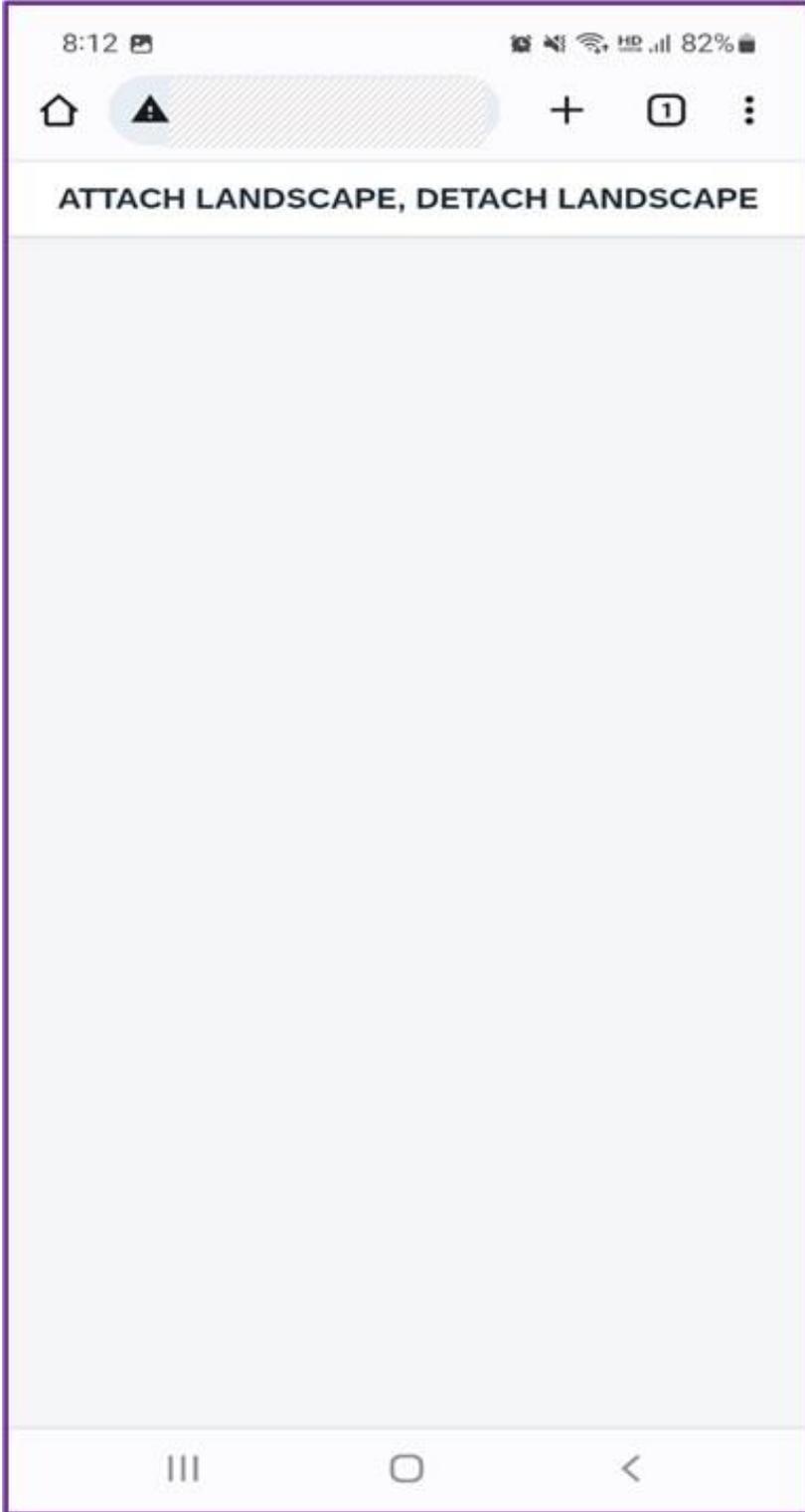
```
EXPORTING
```

```
IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class
```

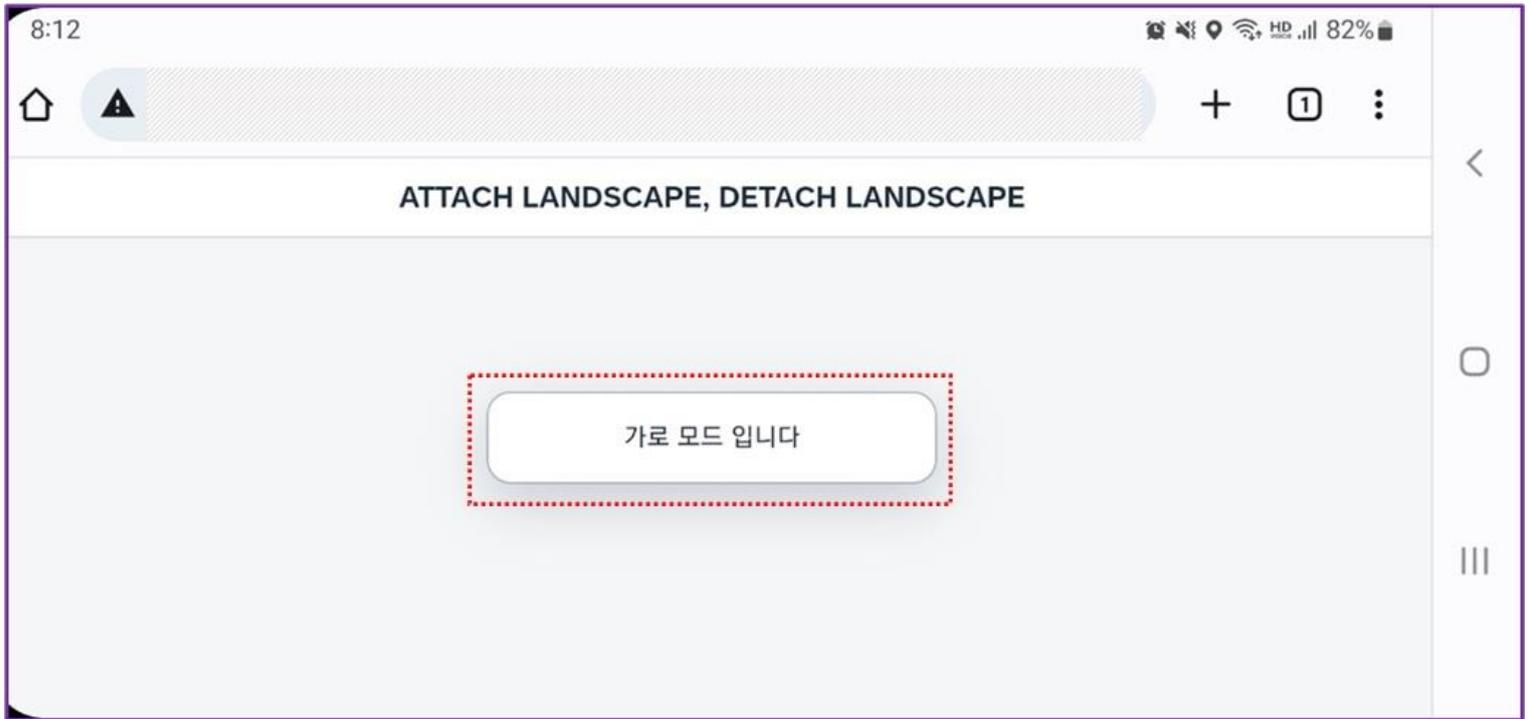
```
I_MSGTX = LV_MSG. " MESSAGE TEXT
```

```
ENDMETHOD.
```

4. Application 실행(본 단원에서는 Mobile 기기 + 세로모드 시작)



5. Mobile 기기 세로-> 가로 회전에 따른 메시지 호출



6. Mobile 기기 가로-> 세로 회전에 따른 메시지 호출

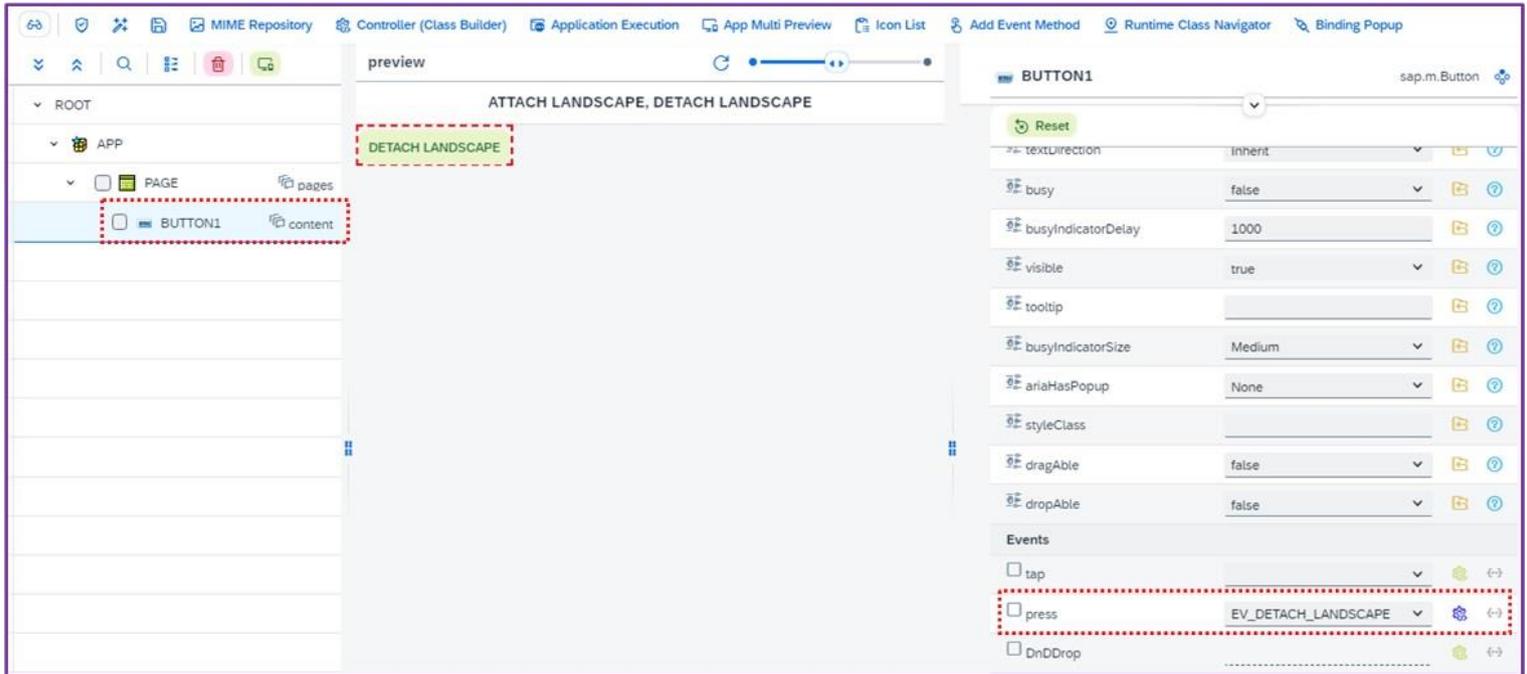


1. "Methodes : ATTACH_LANDSCAPE_JQUERY" 는 위의 ATTACH_LANDSCAPE 가 동작하지 않는 환경에서 사용
2. "Methodes : ATTACH_LANDSCAPE_JQUERY"의 사용 방법은 위의 방법과 동일하다.

6.1.39 Methodes DETACH_LANDSCAPE

■ 목적 : Application 화면 회전 이벤트 처리(Methodes ATTACH_LANDSCAPE)를 해제 하기 위해서 사용한다.

1. Application에서 화면 회전 이벤트를 해제하기 위한 UI Object 및 Event 생성 (Button 사용)



2. 화면 회전 이벤트 해제를 위한 내역 설정.

[첨부ABAP SOURCE (EV_DETACH_LANDSCAPE)]

```
METHOD EV_DETACH_LANDSCAPE .
```

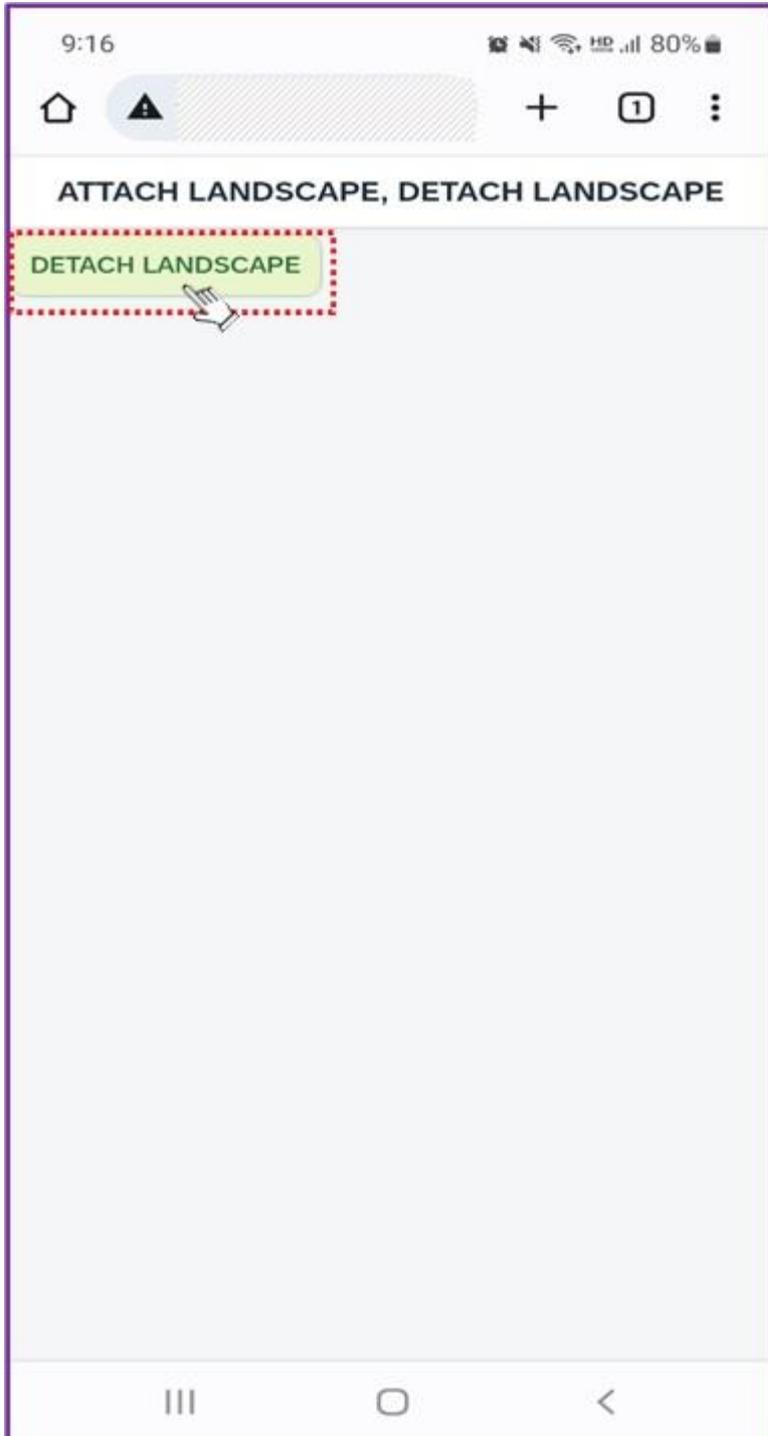
```
CALL METHOD /U4A/CL_UTILITIES=>DETACH_LANDSCAPE
```

```
EXPORTING
```

```
IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW. " [U4A] UI Element Super Class
```

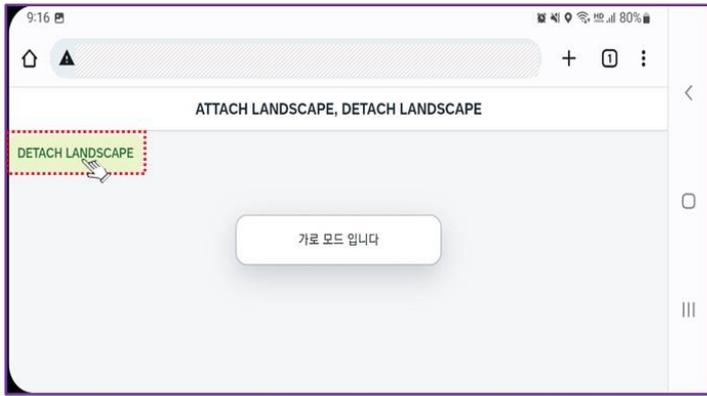
```
ENDMETHOD.
```

3. Application 실행(본 단원에서는 Mobile 기기 + 세로모드 시작)하여 DETACH LANDSCAPE 버튼 클릭하여 ATTACH_LANDSCAPE 설정 해제

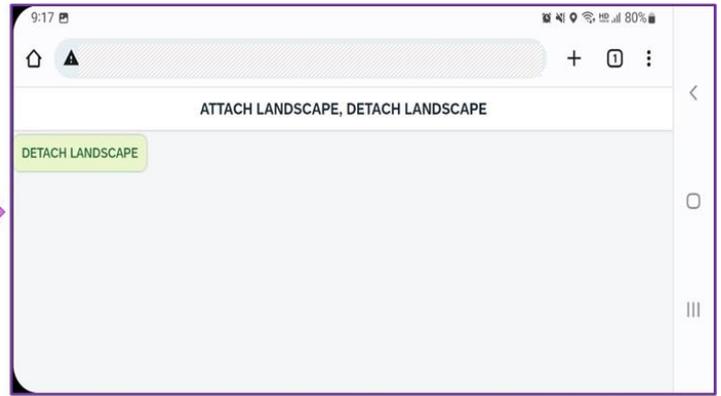
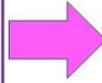


4. Mobile 기기 세로-> 가로 회전시 ATTACH_LANDSCAPE가 설정된 상태에서 "DETACH_LANDSCAPE" 수행 여부에 따른

차이



< Figure : ATTACH_LANDSCAPE "적용", DETACH_LANDSCAPE "미적용" 상태
세로 -> 가로 전환 >

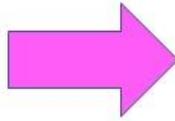


< Figure : DETACH_LANDSCAPE "적용" 하여 ATTACH_LANDSCAPE "해제" 상태
가로 -> 세로 전환 >

5.Mobile 기기 가로 -> 세로시 ATTACH_LANDSCAPE가 설정된 상태에서 "DETACH_LANDSCAPE" 수행 여부에 따른 차이



< Figure : ATTACH_LANDSCAPE 적용, DETACH_LANDSCAPE 미적용 상태
가로 -> 세로 전환 >



< Figure : DETACH_LANDSCAPE 적용 하여 ATTACH_LANDSCAPE 해제 상태
가로 -> 세로 전환 >

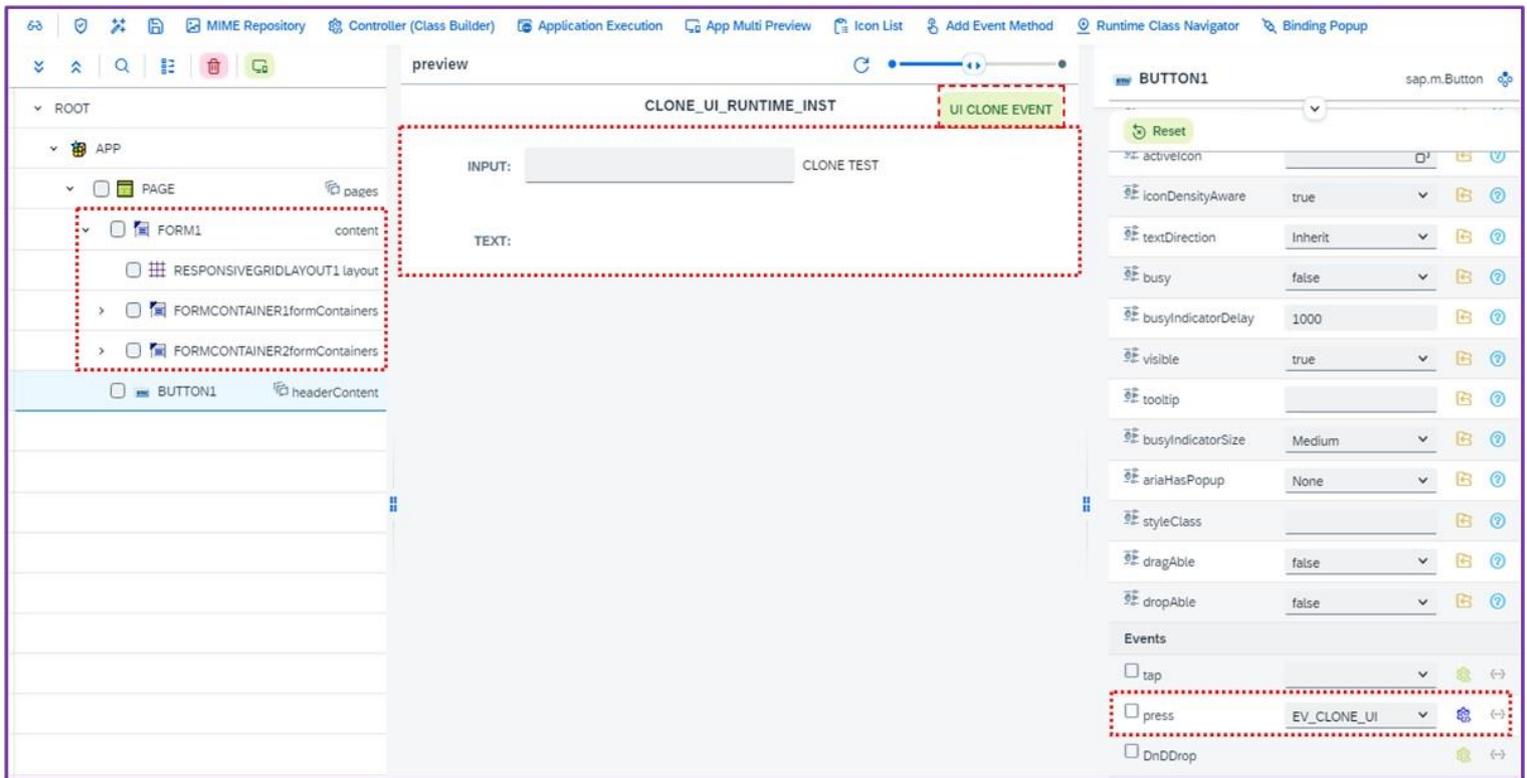
1. "Methodes : DETACH_LANDSCAPE_JQUERY" 는 위의 DE TACH_LANDSCAPE 가 동작하지 않는 환경에서 사용
2. "Methodes : DETACH_LANDSCAPE_JQUERY"의 사용 방법은 위의 방법과 동일하다.

6.1.40 Methodes CLONE_UI_RUNTIME_INST

■ 목적 : 동일한 UI INSTANCE 복사 생성 하기 위해 사용.

1. Application에서 복사 생성할 UI Object를 생성하고 이를 수행할 Event 생성 (Button 사용)

※ 본 단원에서 복사할 대상 UI : "Form" 설정



2. "Methodes : CLONE_UI_RUNTIME_INST" 를 구성한다.

[첨부ABAP SOURCE (EV_CLONE_UI)]

METHOD EV_CLONE_UI.

DATA : LO_FORM TYPE REF TO /U4A/CL_U001001,
 LO_FORM1 TYPE REF TO /U4A/CL_U001001.

```
DATA : LO_PAGE TYPE REF TO /U4A/CL_U000389.
```

```
"FORM UI INSTANCE.
```

```
LO_FORM ?= ME->/U4A/IF_SERVER~AR_VIEW->GET_UI_INSTANCE( I_ID = 'FORM1' ).
```

```
CHECK LO_FORM IS BOUND.
```

```
"복사 생성될 FORM UI INSTANCE.
```

```
LO_FORM1 ?= /U4A/CL_UTILITIES=>CLONE_UI_RUNTIME_INST(
```

```
EXPORTING
```

```
IO_VIEW      = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class
```

```
IO_SOURCE    = LO_FORM " 복사 대상 UI
```

```
IV_TARGET_ID = 'CPFORM1' ). " ID(복사 된 UI의 NEW ID 예 PAGE1 -
```

```
> CPPAGE1)
```

```
CHECK LO_FORM1 IS BOUND.
```

```
"PAGE UI INSTANCE.
```

```
LO_PAGE ?= ME->/U4A/IF_SERVER~AR_VIEW->GET_UI_INSTANCE( I_ID = 'PAGE' ). " UI Object ID
```

```
CHECK LO_PAGE IS BOUND.
```

```
" 복사 생성된 Form UI 삽입.
```

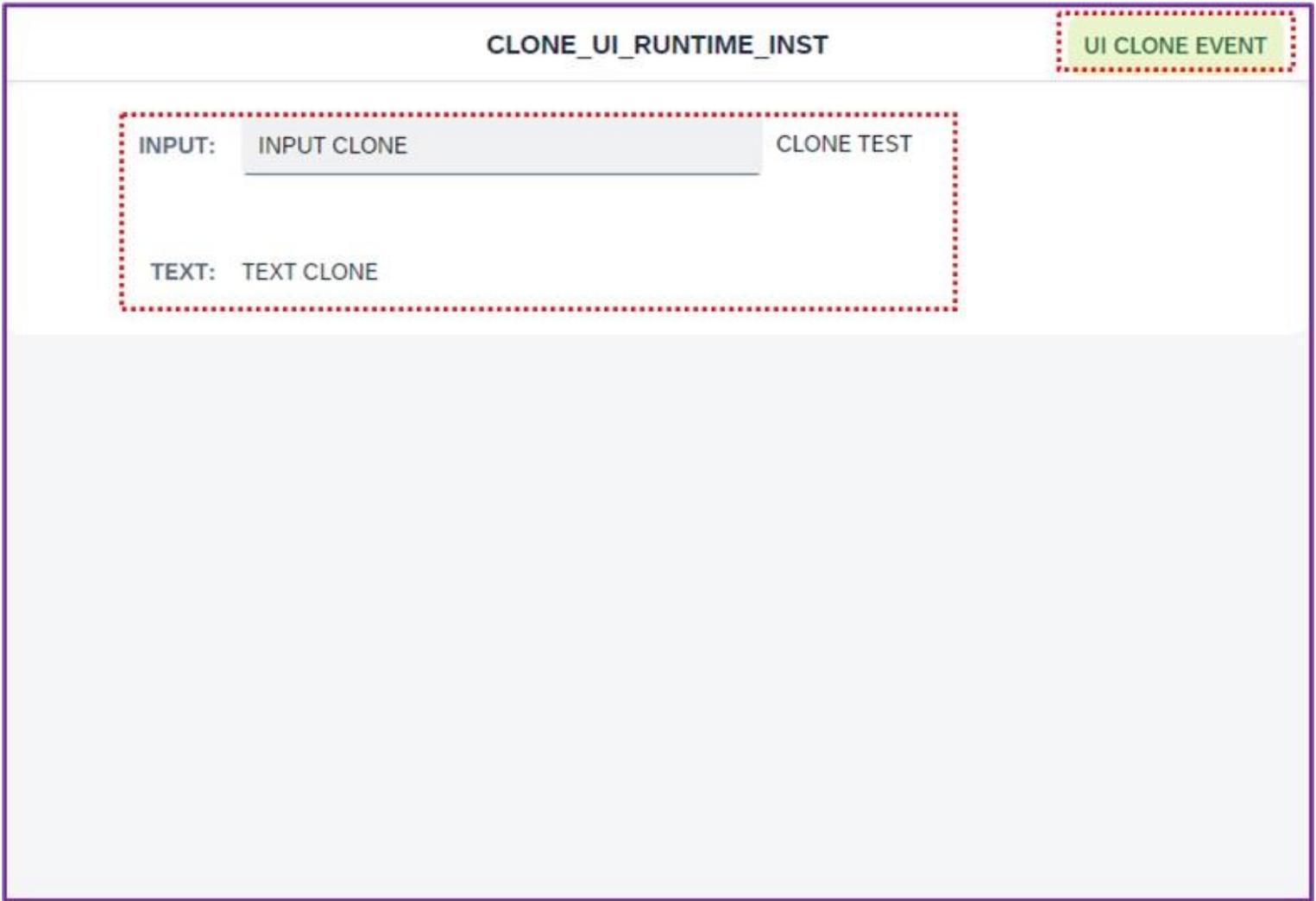
```
CALL METHOD LO_PAGE->ADDCONTENT
```

```
EXPORTING
```

```
CONTENT = LO_FORM1. " content
```

```
ENDMETHOD.
```

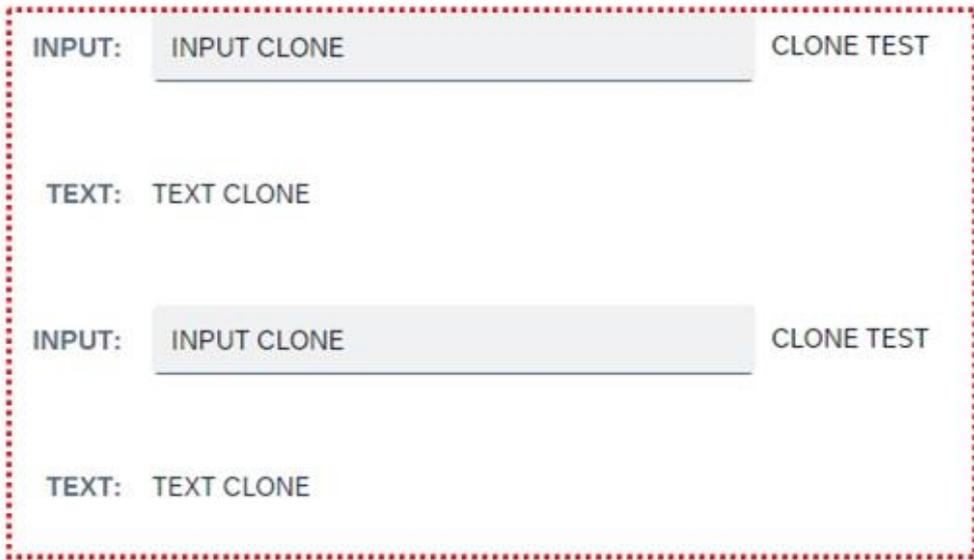
3. Application을 실행하여 상단의 "UI CLONE EVENT" 버튼 클릭



4. "Methodes : CLONE_UI_RUNTIME_INST" 실행결과 기존과 동일한 FORM UI가 복사 생성 된것을 확인.

CLONE_UI_RUNTIME_INST

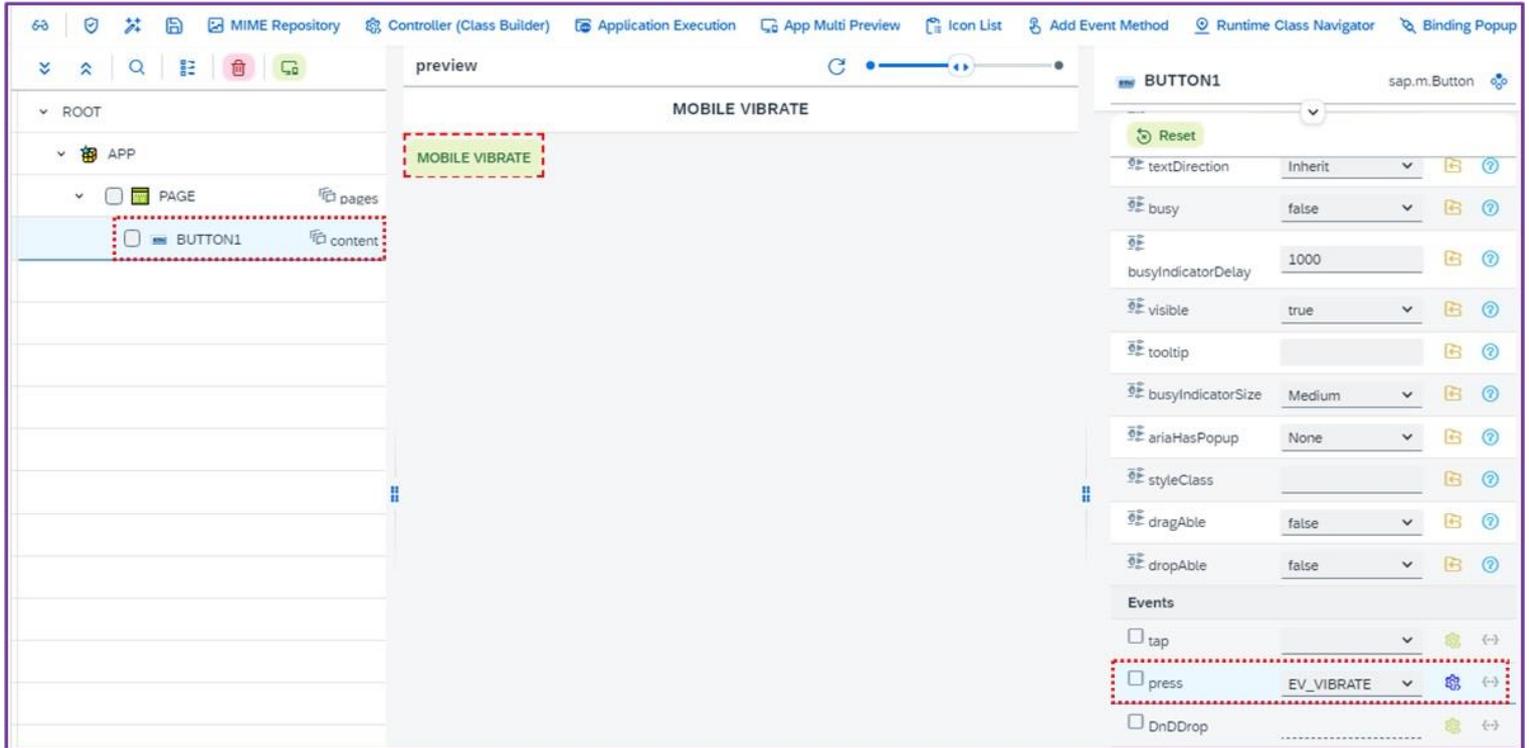
UI CLONE EVENT



6.1.41 Methodes MOBILE_VIBRATE

목적 : 모바일 기기에 진동을 주기 위해 사용. (Android 전용, ios 는 사용불가)

1. Application에서 진동 기능을 수행시킬 UI Object 및 Event 생성 (Button 사용)



2. "Methodes : MOBILE_VIBRATE" 를 구성한다.

[첨부ABAP SOURCE (EV_VIBRATE)]

```
METHOD EV_VIBRATE.
```

```
DATA LV_SIGNAL TYPE INT4.
```

```
DATA LT_SIGNAL TYPE INT4_TABLE.
```

"진동 주기 설정."

```
LV_SIGNAL = 1000.
APPEND LV_SIGNAL TO LT_SIGNAL.
CLEAR LV_SIGNAL.
```

```
LV_SIGNAL = 2000.
APPEND LV_SIGNAL TO LT_SIGNAL.
CLEAR LV_SIGNAL.
```

```
LV_SIGNAL = 3000.
APPEND LV_SIGNAL TO LT_SIGNAL.
CLEAR LV_SIGNAL.
```

```
LV_SIGNAL = 4000.
APPEND LV_SIGNAL TO LT_SIGNAL.
CLEAR LV_SIGNAL.
```

"모바일 기기 진동 설정."

```
CALL METHOD /U4A/CL_UTILITIES=>MOBILE_VIBRATE
EXPORTING
```

```
IO_VIEW    = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class
```

```
IT_SIGNAL  = LT_SIGNAL " 진동 주기(1초 = 1000) 값이 없는경우 진동 종
```

료.(진동, 대기, 진동, 대기 순으로 진행됨)

```
* I_SHOW_MESSAGE = " 진동 기능 사용 불가능 device일경우 메시지 호
출 여부('X' = 사용불가 메시지 호출)
```

```
ENDMETHOD.
```

3. 모바일 기기(Android)를 통해 Web에서 Event 작동시 1초진동 -> 2초 휴식 -> 3초 진동-> 4초 휴식 후 종료 되는것을 확인할 수 있다.

MOBILE VIBRATE

MOBILE VIBRATE



1. Methodes : MOBILE_VIBRATE 는 (Android 전용, ios는 사용불가)



2. 진동 주기 (1초 = 1000)

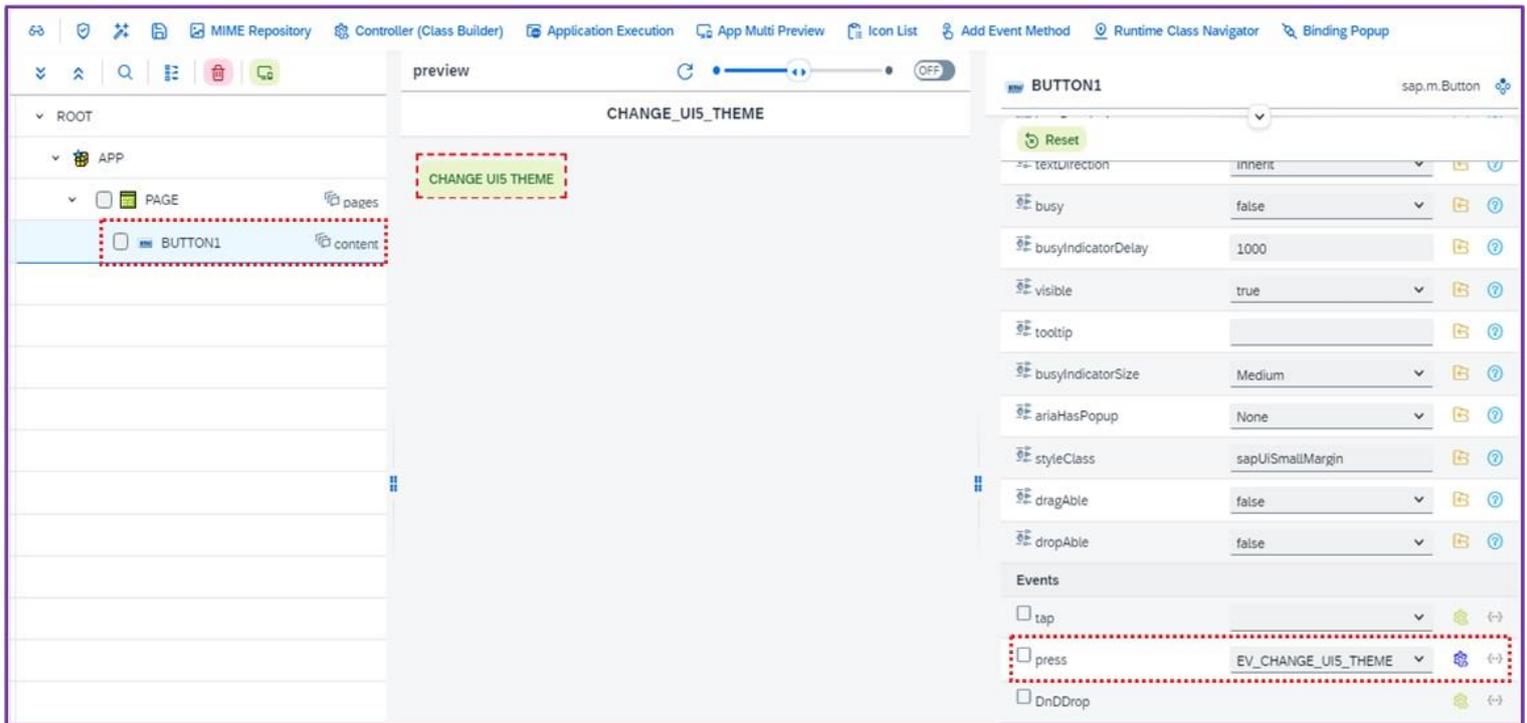


3. 진동 주기 : IT_SIGNAL의 ROW 값이 1000,2000,1000,2000 구성 => 1초 진동 -> 2초 휴식 -> 1초 진동 -> 2초 휴식으로 진행.

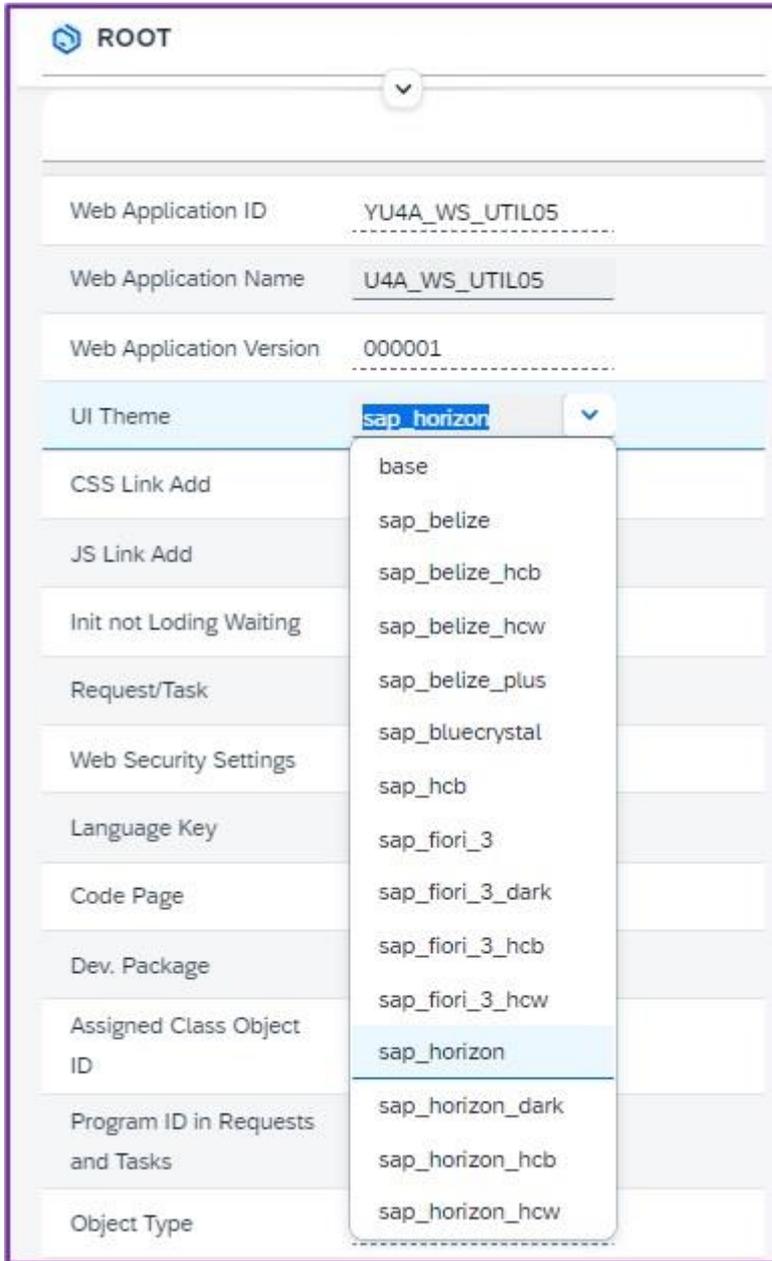
6.1.42 Methodes CHANGE_UI5_THEME

■ 목적 : 해당 Application의 UI5 Theme를 변경하기 위함

1. Application에서 UI5 Theme를 변경을 수행시킬 UI Object 및 Event 생성 (Button 사용)



2. 위의 Application의 현재 UI5 Theme 및 설정가능한 UI5 Theme 목록 (ROOT = > Properties = > UI Theme 확인).



3. " Methodes : CHANGE_UI5_THEME" 를 구성한다.

[첨부ABAP SOURCE (EV_CHANGE_UI5_THEME)]

METHOD EV_CHANGE_UI5_THEME.

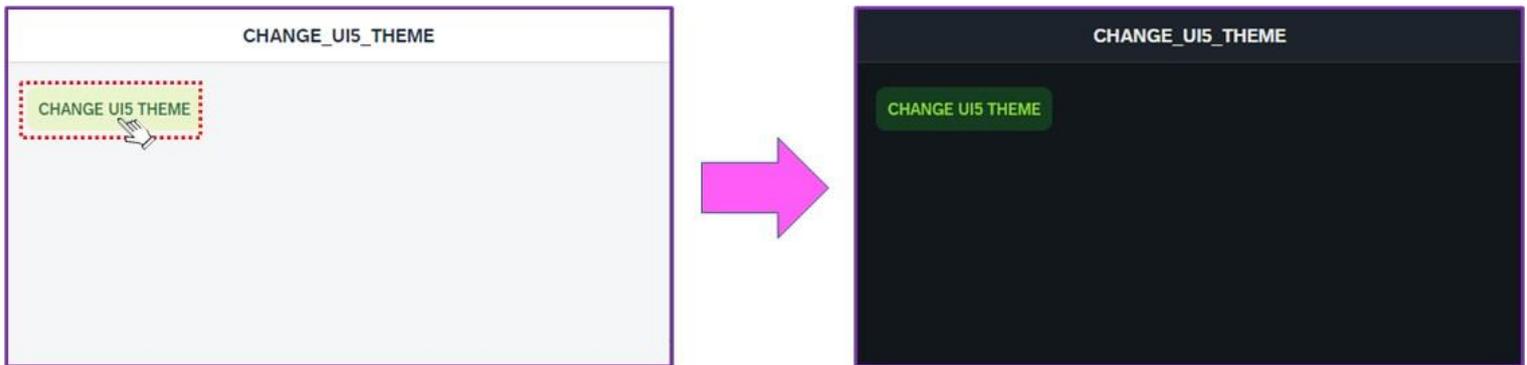
"현재 UI Theme : sap_horizon / 변경 UI Theme : sap_horizon_dark"

```
CALL METHOD /U4A/CL_UTILITIES=>CHANGE_UI5_THEME
EXPORTING
  IO_VIEW      = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class
  I_THEME_NAME = 'sap_horizon_dark'.      " UI5 Theme Name

ENDMETHOD.
```

4. Web에서 Event 작동시 UI Theme가 지정한 Theme로 변경되는것을 확인.

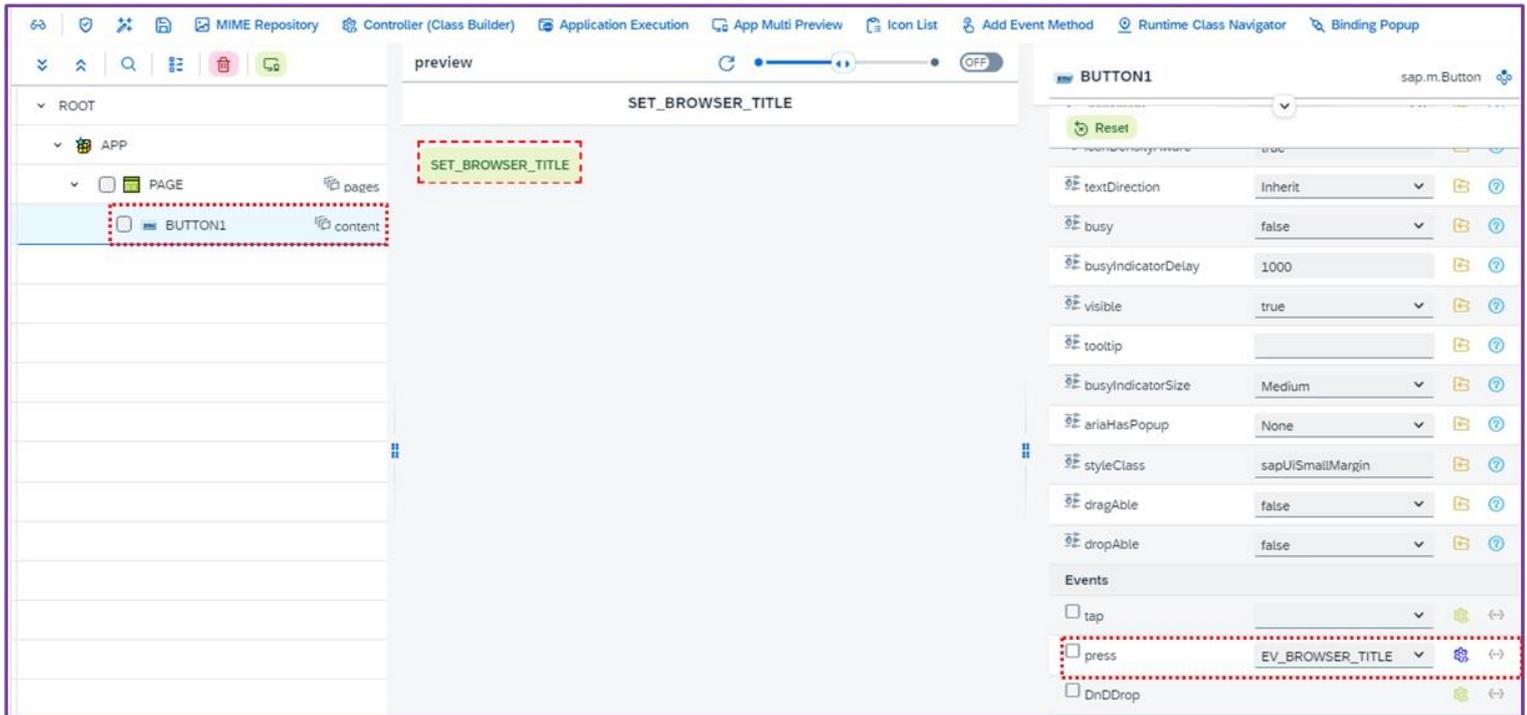
※ sap_horizon = > sap_horizon_dark



6.1.43 Methodes SET_BROWSER_TITLE

■ 목적 : Application 실행 시 설정된 Browser Title을 변경하기 위함

1. Application에서 Browser Title의 변경을 수행시킬 UI Object 및 Event 생성 (Button 사용)



2. "Methodes : SET_BROWSER_TITLE" 를 구성한다.

[첨부ABAP SOURCE (EV_BROWSER_TITLE)]

```
METHOD EV_BROWSER_TITLE.
```

```
CALL METHOD /U4A/CL_UTILITIES=>SET_BROWSER_TITLE
EXPORTING
```

```
IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW
```

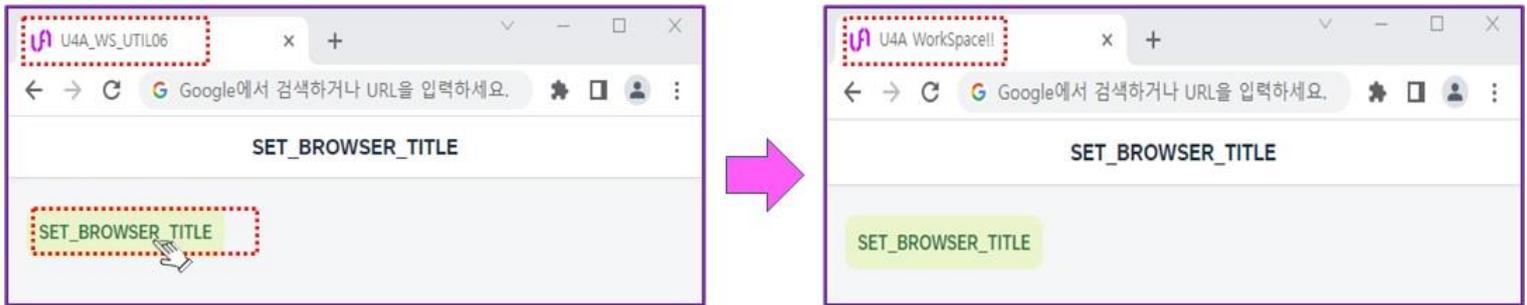
" [U4A] UI Element Super Class

```
I_TITLE = 'U4A WorkSpace!!'.
```

" Title

```
ENDMETHOD.
```

3. Application 실행하여 Web에서 Event 작동시 Browser Title이 지정한 Title로 변경되는지 확인

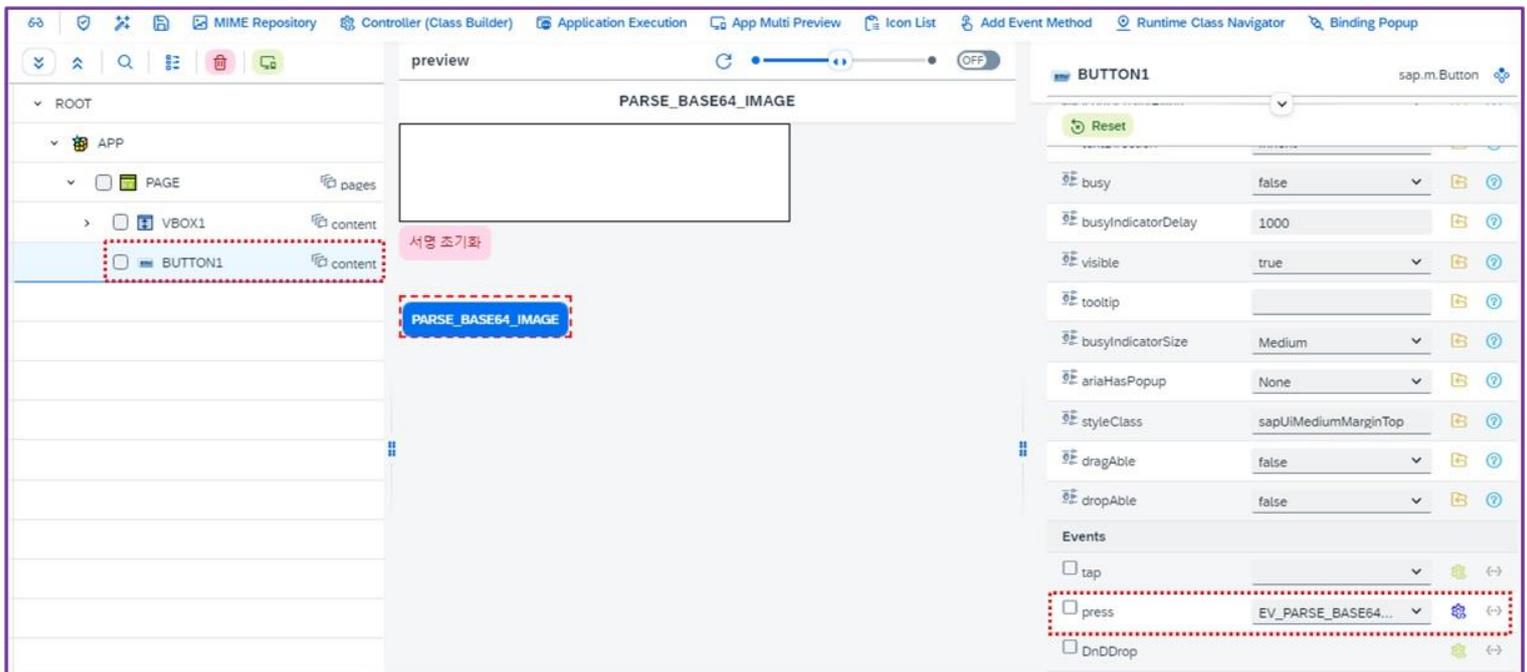


6.1.44 Methodes PARSE_BASE64_IMAGE

■ 목적 : BASE64 형식의 이미지 정보를 XSTRING 형태로 변환하기 위함

1. Application에서 "BASE64 형식"의 이미지를 -> "XSTRING 형식"으로 변환하기 위해 SIGNATURE, BUTTON UI 생성.

※ Signature UI : 서명 이미지가 "BASE64" 형식임.



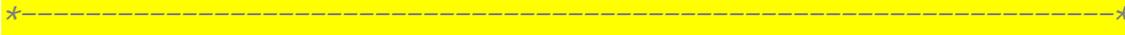
2. "Methodes : SET_BROWSER_TITLE" 를 구성한다.

[첨부ABAP SOURCE (EV_PARSE_BASE64_IMAGE)]

METHOD EV_PARSE_BASE64_IMAGE.

DATA : LV_MIME TYPE STRING,

```
LV_FILE TYPE STRING,
LV_XSTR TYPE XSTRING.
```



CHECK GS_SIGN-URL IS NOT INITIAL.

"BASE64 -> XSTRING 변환

CALL METHOD /U4A/CL_UTILITIES=>PARSE_BASE64_IMAGE

EXPORTING

I_FILE = GS_SIGN-URL " BASE64 형식 IMAGE

IMPORTING

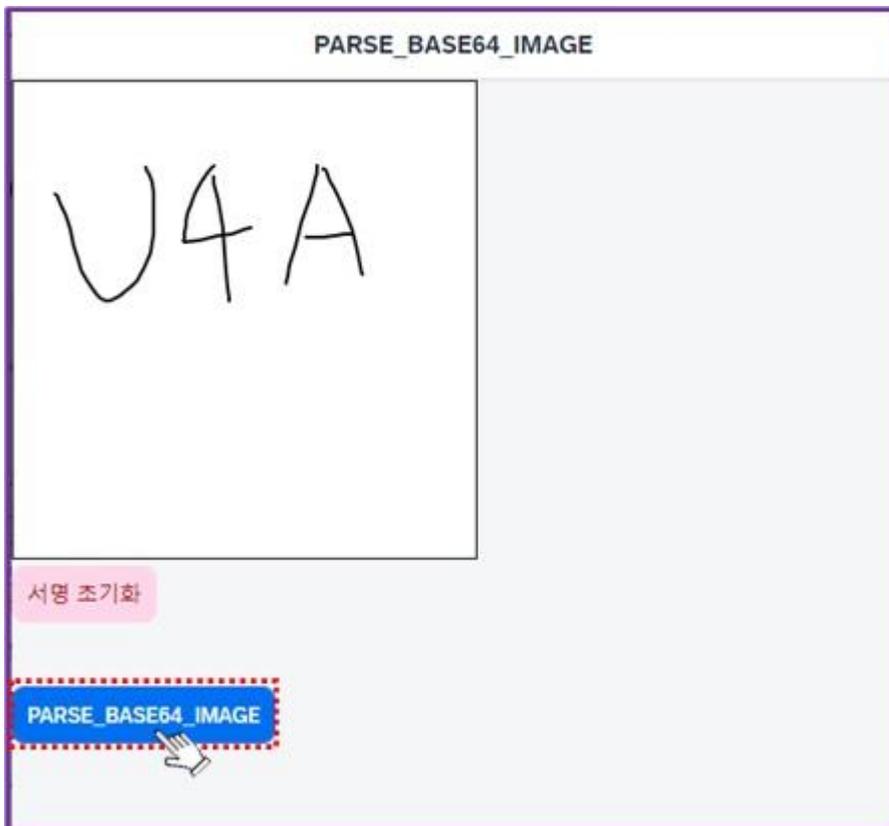
E_MIME = LV_MIME " MIME 정보(image/png)

E_FILE = LV_FILE " BASE64 형식 IMAGE(data:image/png;base64,부분 제거)

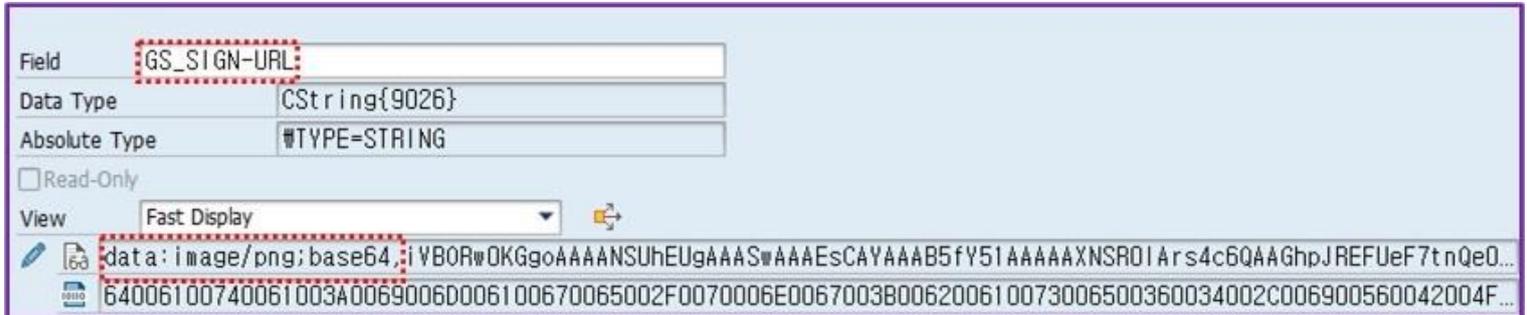
E_XSTR = LV_XSTR. " BASE64 -> xstring

ENDMETHOD.

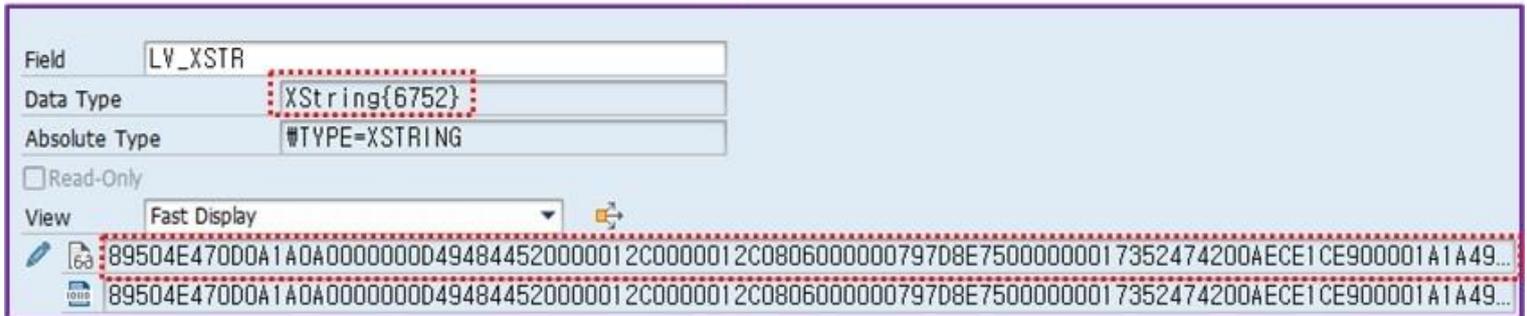
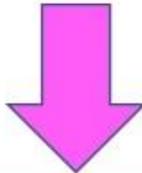
3. Application 실행 후 Signature UI에 서명을 한뒤 "PARSE_BASE64_IMAGE" Button 클릭



4. "BASE64" 형식의 이미지 정보를 "XSTRING" 형태로 변환 확인.



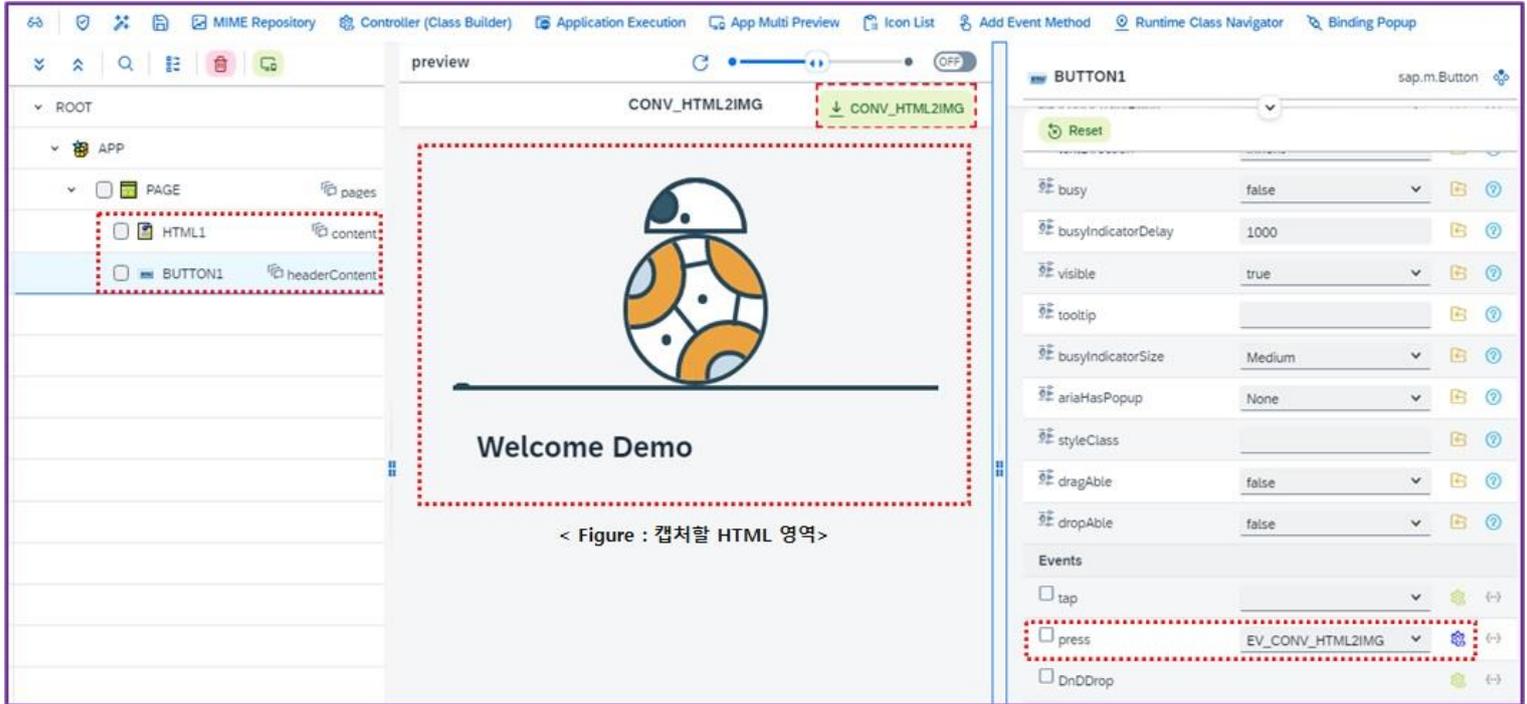
< Figure : 초기 Signature Ui 서명 이미지 정보(BASE64)>



< Figure : BASE64 -> XSTRING 변환 결과 >

6.1.45 Methodes CONV_HTML2IMG

1. Application에서 HTML 영역의 캡처를 수행하기위한 UI Object 및 Event 생성 (Button 사용)



2. " Methodes : CONV_HTML2IMG" 를 구성한다.

[첨부ABAP SOURCE (EV_CONV_HTML2IMG)]

METHOD EV_CONV_HTML2IMG.

"HTML 영역을 이미지로 캡처하는 기능"

CALL METHOD /U4A/CL_UTILITIES=>CONV_HTML2IMG
EXPORTING

IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW
I_UIID = 'HTML1'
I_MIME_TYPE = 'JPG'
I_FILE_NAME = 'U4A HTML'

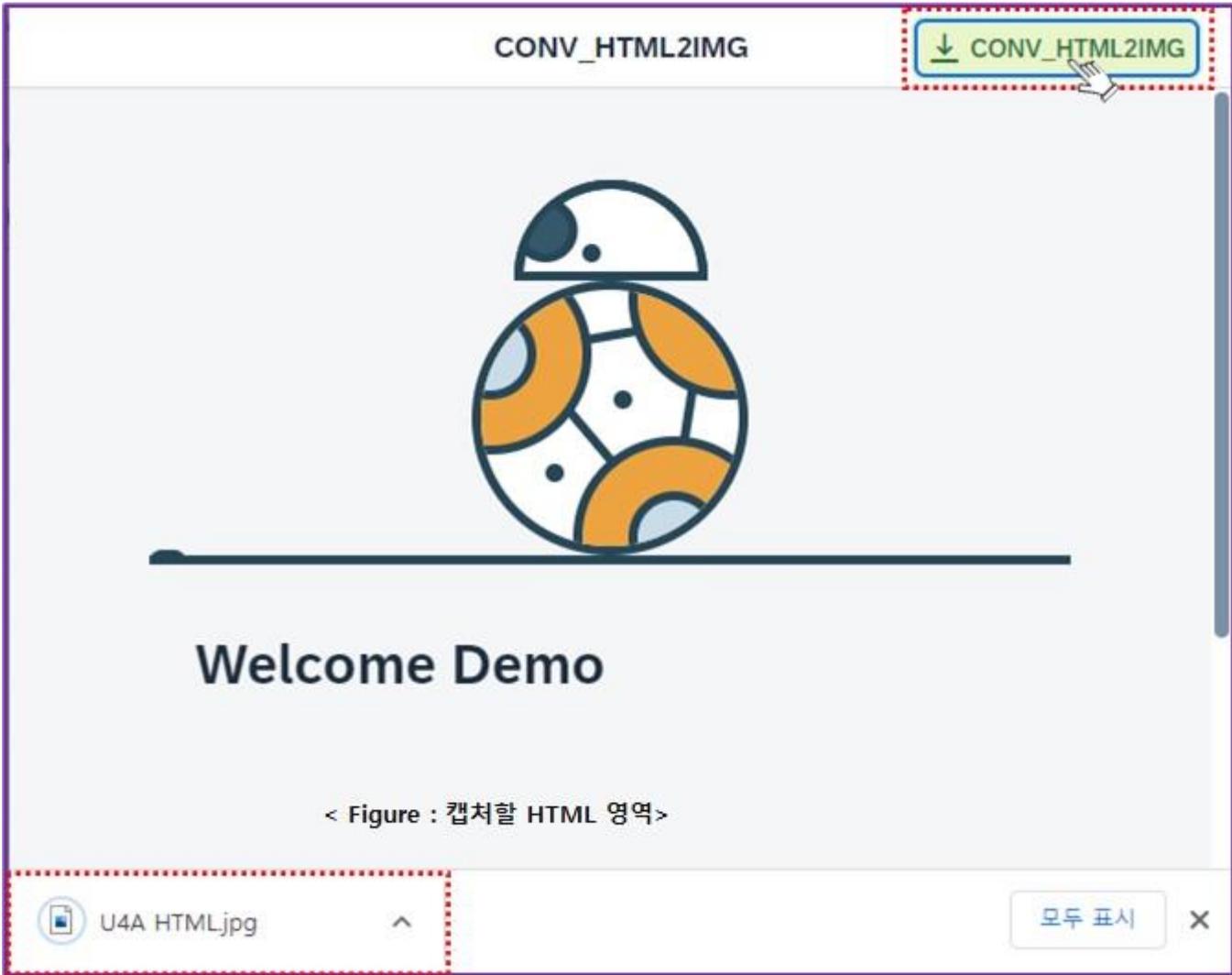
" [U4A] UI Element Super Class
" UI Object 명
" MIME TYPE
" 이미지 파일 명

```
* EXCEPTIONS
*      MIMETYPE_NOT_ALLOWED = 1           " 허용되지 않는 MIMETYPE
*      OTHERS                = 2

      .
      IF SY-SUBRC <> 0.
* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO
* WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.
      ENDIF.

      ENDMETHOD.
```

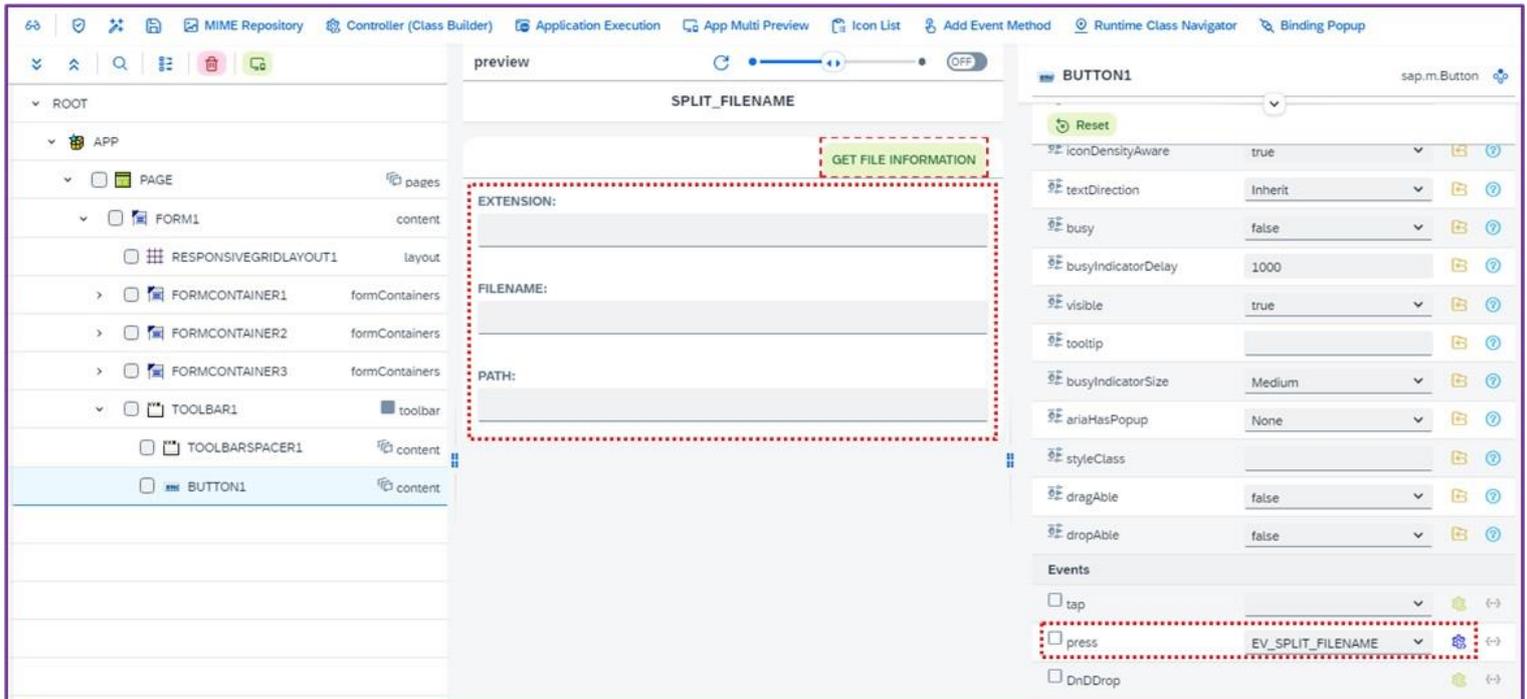
3. Application 실행 후 " CONV_HTML2IMG " Button을 클릭하면 HTML영역을 이미지로 캡처하여 다운로드 한다.



6.1.46 Methodes SPLIT_FILENAME

■ 목적 : PC에 있는 File의 Path, File Name, Extension 정보를 알려고 할때 사용함.

1. Application에서 UI Object와 Event를 생성한다.



2. " Methodes : SPLIT_FILENAME" 를 구성한다.

[첨부ABAP SOURCE (EV_SPLIT_FILENAME)]

```
METHOD EV_SPLIT_FILENAME.
```

"PC File의 Extensint, File Name, Path 분리"

```
CALL METHOD /U4A/CL_UTILITIES=>SPLIT_FILENAME
```

```
EXPORTING
```

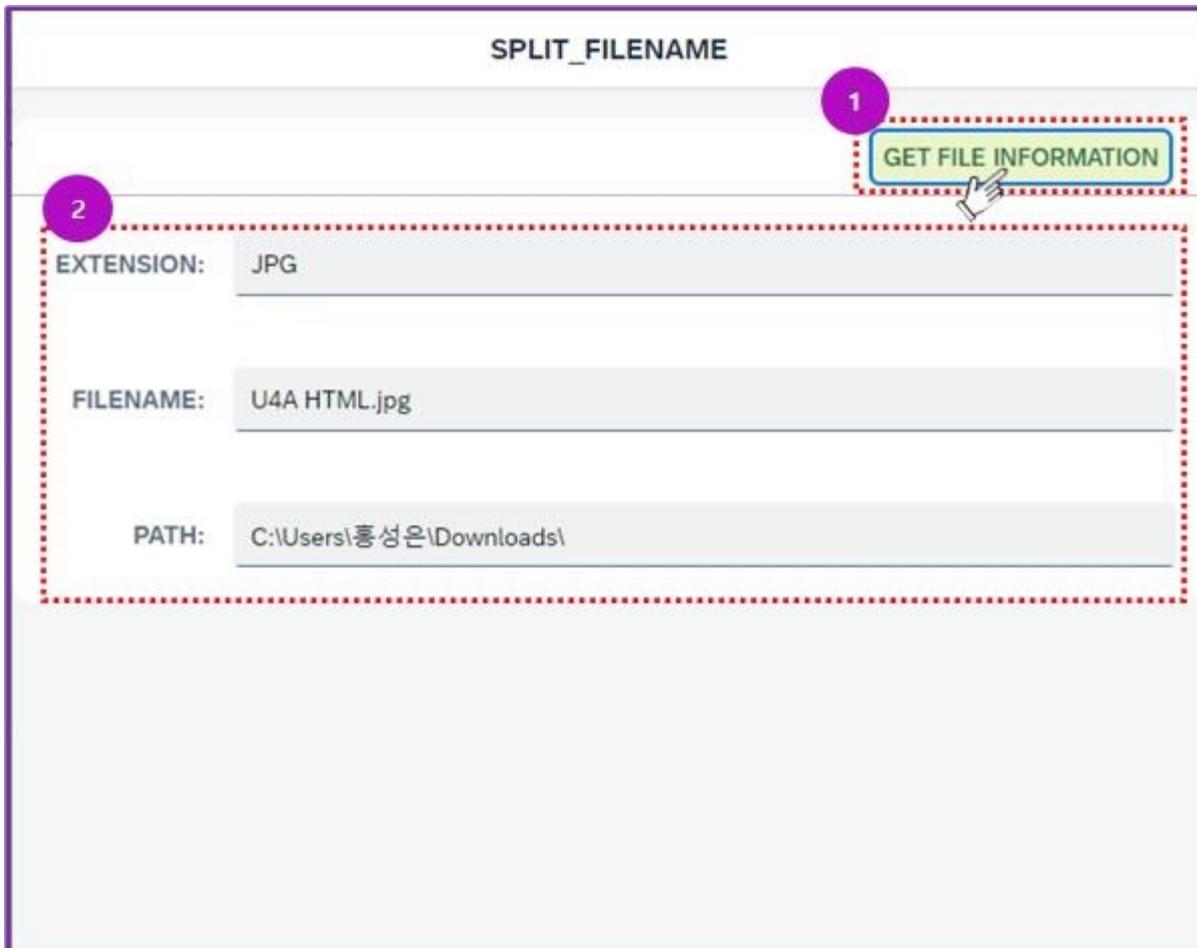
```
  I_FILENAME      = 'C:\Users\홍성은\Downloads\U4A HTML.jpg'
```

IMPORTING

```
* E_DRIVE =
  E_EXTENSION = GS_FILE-EXTENSION
* E_NAME =
  E_NAME_WITH_EXT = GS_FILE-NAME_WITH_EXT
  E_PATH = GS_FILE-PATH.
```

ENDMETHOD.

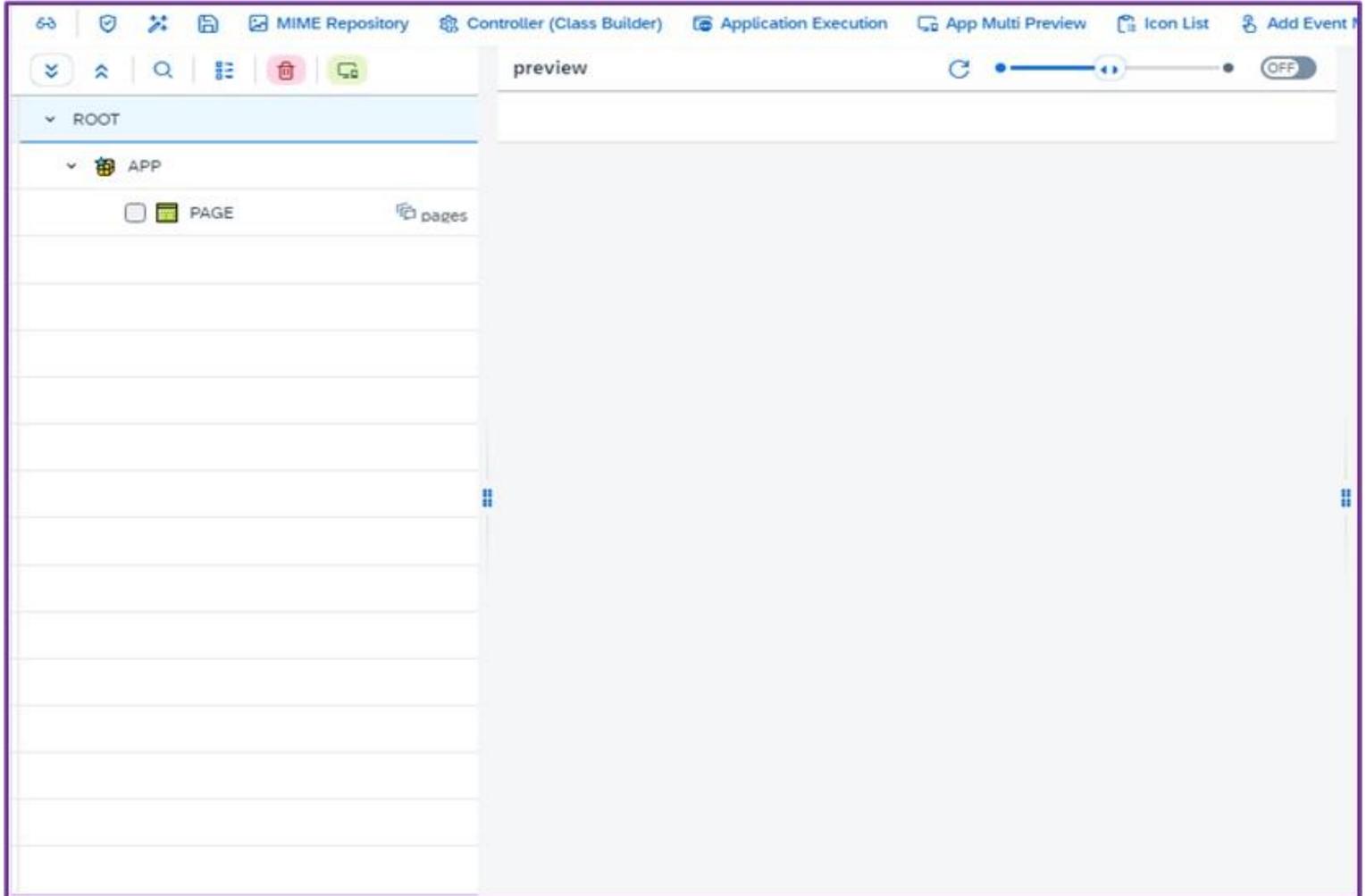
3. Application 실행 후 "GET FILE INFORMATION" Button을 클릭하여 EXTENSION, FILENAME, PATH 정보를 확인한다.



6.1.47 Methodes JS_FILES_LOAD

■ 목적 : Java Script File을 불러와서 Application에 적용하기 위함.

1. JS File 불러오기전 실행 화면



JS_FILES_LOAD

Hello World

2. " Methodes : JS_FILES_LOAD" 를 구성한다.

[첨부ABAP SOURCE (/U4A/IF_SERVER~HANDL_ON_INIT)]

```
METHOD /U4A/IF_SERVER~HANDL_ON_INIT.
```

```
DATA : LT_SRC TYPE /U4A/Y0075,  
       LS_SRC TYPE /U4A/S0075.
```

```
*-----*
```

```
LS_SRC-SRC      = '/zu4a/public/application/zu4a_ws_util14/JS FILE.js'. "JS URL  
LS_SRC-ISSYNC  = 'X'. "JS FILE LOAD 동기 : X 비동기 : SPACE
```

```
APPEND LS_SRC TO LT_SRC .
CLEAR LS_SRC .
```

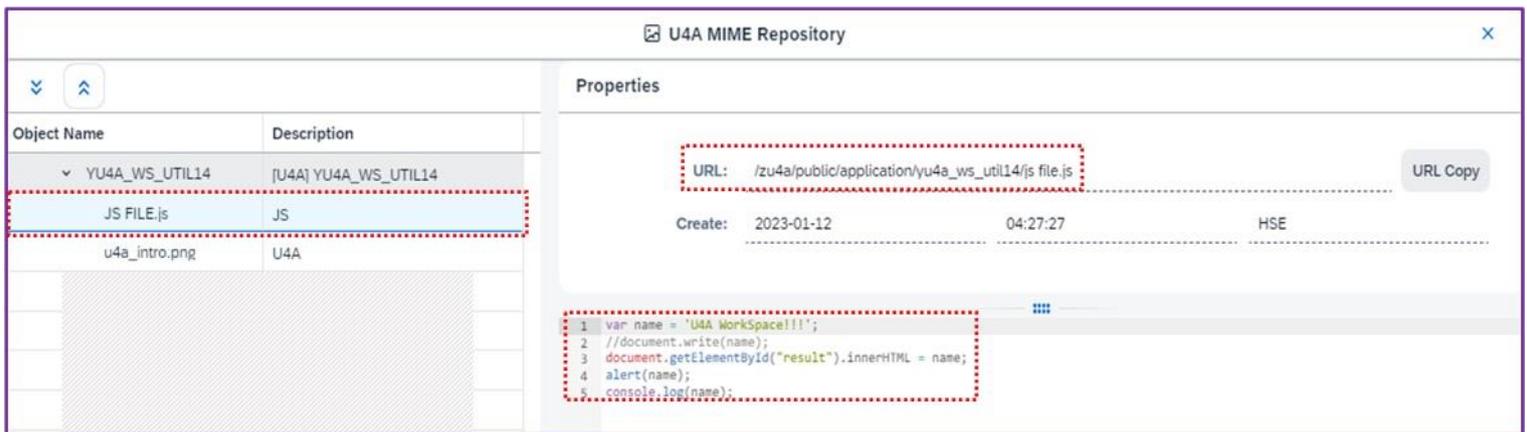
"JS File 불러오기."

```
CALL METHOD /U4A/CL_UTILITIES=>JS_FILES_LOAD
EXPORTING
  IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW
  IT_SRC = LT_SRC .
```

" [U4A] UI Element Super Class
" JS files 수집 탭

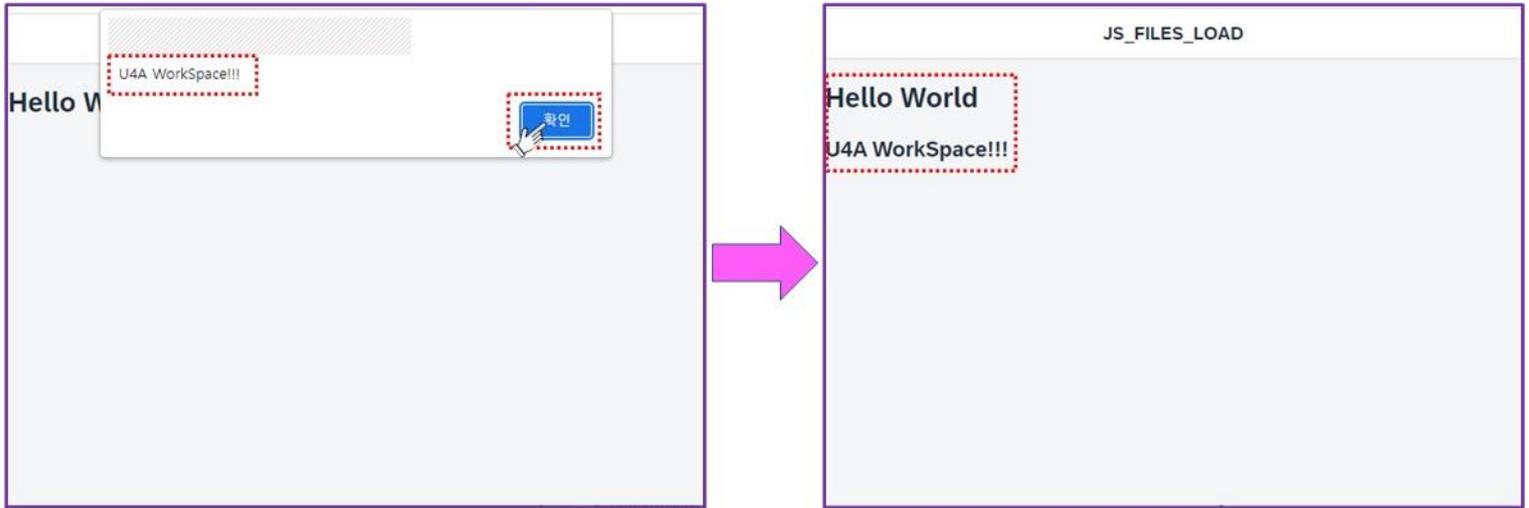
```
ENDMETHOD .
```

※. 본 단원에서는 Mime Repository에 등록된 JS File을 불러온다.



3. Application 실행 후 JS File을 불러온 결과값 확인.

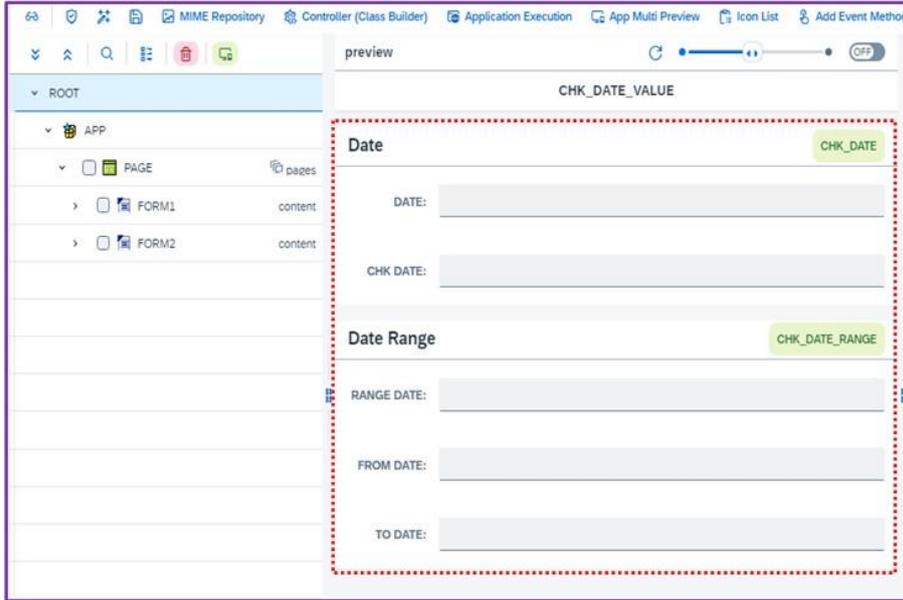
U4A IDE HELP



6.1.48 Methodes CHK_DATE_VALUE

■ 목적 : 날짜 형식으로 입력값을 정확히 입력했는지 정합성을 체크하기 위하여 사용.

1. Application에서 UI Object와 Event를 생성한다.



UI Where to Use the Event		
Event ID	Event Target Properties	UI OBJ ID
EV_CHK_DATE_VALUE	press	BUTTON2
Event Text: CHK_DATE_VALUE		
EV_CHK_DATE_RANGE_VALUE	press	BUTTON3
Event Text: [EVENT] CHK RANGE DATE		

2. 단일날짜 입력값 정합성 체크.

① " Methodes : CHK_DATE_VALUE" 를 구성한다.

[첨부ABAP SOURCE (EV_CHK_DATE_VALUE)]

METHOD EV_CHK_DATE_VALUE.

DATA : LV_ERROR TYPE ABAP_BOOL,
LV_MSG TYPE STRING.

```
*-----*
```

"단일 날짜 OR 범위 날짜 입력값 점검"

```
CALL METHOD /U4A/CL_UTILITIES=>CHK_DATE_VALUE
```

```
EXPORTING
```

```
    IV_DATE      = GS_COND-DATE " 단일 날짜 입력값(YYYYMMDD) => '/', '.', '-', '_' 4개의 특수기호만 포함 되어있어도 체크 가능
```

```
*    IV_DATE_RANGE = " 범위 날짜 입력값(YYYYMMDD - YYYYMMDD)
```

```
IMPORTING
```

```
    EV_ERROR     = LV_ERROR      " 'X' = 오류
```

```
    EV_MESSAGE   = LV_MSG        " 오류시 오류 메시지
```

```
    EV_FROM      = GS_COND-CHK_DATE " FROM 날짜(IV_DATE의 경우 EV_FROM으로 RETURN)
```

```
*    EV_TO       = " TO 날짜(IV_DATE_RANGE 의 TO에 해당하는 날짜 RETURN)
```

"날짜 정합성 체크 오류가 없는 경우"

```
IF LV_ERROR EQ SPACE.
```

```
    LV_MSG = '날짜 정합성 점검 완료하였습니다'.
```

```
ENDIF.
```

"메세지 호출."

```
CALL METHOD /U4A/CL_UTILITIES=>M_MESSAGE_TOAST
```

```
EXPORTING
```

```
    IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class
```

```
    I_MSGTX = LV_MSG " MESSAGE TEXT
```

```
ENDMETHOD.
```

② Application을 실행하여 "Date" 필드에 날짜값을 입력한뒤 "CHK_DATE" Button Click

CHK_DATE_VALUE

Date CHK_DATE

DATE: 2023-01-12 ('/', ' ', '-', '_' 4가지 특수문자 입력 가능)

CHK DATE: 2023/01/12

Date Range CHK_DATE_RANGE

날짜 정합성 점검 완료하였습니다

RANGE DATE:

FROM DATE:

TO DATE:

3. 범위날짜 입력값 정합성 체크.

- ① " Methodes : CHK_DATE_VALUE" 를 구성한다.

[첨부ABAP SOURCE (EV_CHK_DATE_RANGE_VALUE)]

```
METHOD EV_CHK_DATE_RANGE_VALUE.
```

```
DATA : LV_ERROR TYPE ABAP_BOOL,
       LV_MSG   TYPE STRING.
```

```
*-----*
```

```
"단일 날짜 OR 범위 날짜 입력값 점검
```

```
CALL METHOD /U4A/CL_UTILITIES=>CHK_DATE_VALUE
```

```
EXPORTING
```

```
IV_DATE_RANGE = GS_COND-DATE_RANGE " 범위 날짜 입력값(YYYYMMDD - YYYYMMDD)
```

```
IMPORTING
```

```
EV_ERROR      = LV_ERROR      " 'X' = 오류
```

```
EV_MESSAGE    = LV_MSG        " 오류시 오류 메시지
```

```
EV_FROM       = GS_COND-FROM   " FROM 날짜(IV_DATE의 경우 EV_FROM으로 RETURN)
```

```
EV_TO         = GS_COND-TO     " TO 날짜(IV_DATE_RANGE 의 TO에 해당하는 날짜 RETURN)
```

```
"날짜 정합성 체크 오류가 없는 경우
```

```
IF LV_ERROR EQ SPACE.
```

```
LV_MSG = '날짜 정합성 점검 완료하였습니다'.
```

```
ENDIF.
```

```
"메세지 호출.
```

```
CALL METHOD /U4A/CL_UTILITIES=>M_MESSAGE_TOAST
```

```
EXPORTING
```

```
IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class
```

```
I_MSGTX = LV_MSG.                  " MESSAGE TEXT
```

```
ENDMETHOD.
```

② Application을 실행하여 "RANGE DATE" 필드에 범위 날짜값을 입력한뒤 "CHK_DATE_RANGE" Button Click

CHK_DATE_VALUE

Date

CHK_DATE

DATE:

CHK DATE:

Date Range

날짜 정합성 점검 완료하였습니다

CHK_DATE_RANGE

RANGE DATE:

20230112-20230131

FROM DATE:

2023/01/12

TO DATE:

2023/01/31

6.1.49 Methodes GET_MIME_REPOSITORY_API

■ 목적 : MIME Repository에 등록된 파일의 Binary 정보를 얻기위함.

1. MIME PATH 정보를 찾는다. (본 단원에서는 해당 Application에 Import 된 MIME OBJECT 사용)

The screenshot displays the U4A IDE interface. On the left, a table lists MIME objects:

Object Name	Description
YU4A_WS_UTILO4	[U4A] YU4A_WS_UTILO4
u4a_intro.png	U4A

The 'u4a_intro.png' row is highlighted with a red dashed box. To the right, the details for this object are shown:

< URL = PATH >
 URL: /zu4a/public/application/yu4a_ws_util04/u4a_intro.png
 Create: 2023-01-13 02:39:37 U4A_TM01
 URL Copy

Below the details, a large image is displayed with the text 'WS U4A IDE PLATFORM' in purple and blue, and 'version 3.3.0' below it. At the bottom, there are logos for 'UI5 For ABAP', 'SAP Certified Integration with SAP NetWeaver', and 'SAP Certified Integration with SAP S/4HANA Cloud'.

2. " Methodes : GET_MIME_REPOSITORY_API" 를 구성한다.

[첨부ABAP SOURCE (/U4A/IF_SERVER~HANDL_ON_INIT)]

METHOD /U4A/IF_SERVER~HANDL_ON_INIT.

```
DATA LV_CONTENT TYPE XSTRING.
```

"API for MIME Repository"

```
CALL METHOD /U4A/CL_UTILITIES=>GET_MIME_REPOSITORY_API
```

```
EXPORTING
```

```
I_PATH = 'zu4a/public/application/zu4a_ws_util04/zu4a_intro.png'
```

" Mime Path 예) zu

4a/test/.. sap/bc/..

```
IMPORTING
```

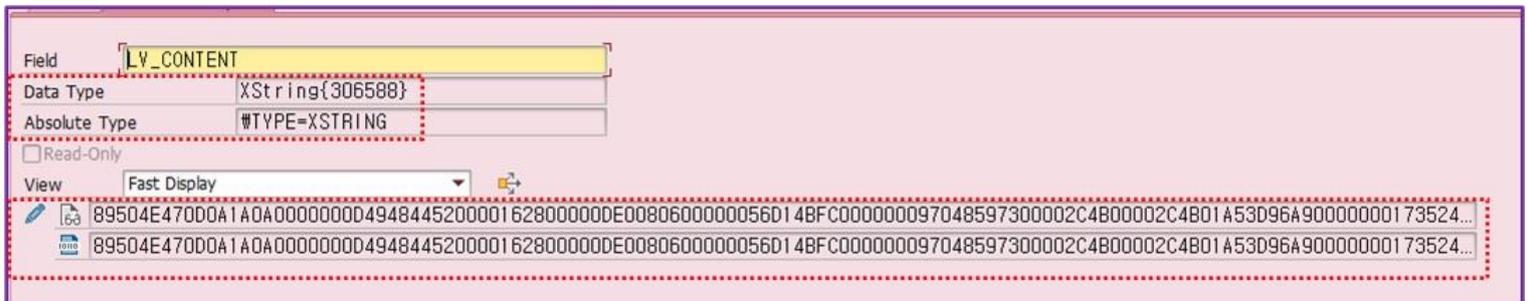
```
E_CONTENT = LV_CONTENT.
```

" 물리적인 파

일 XSTRING

```
ENDMETHOD.
```

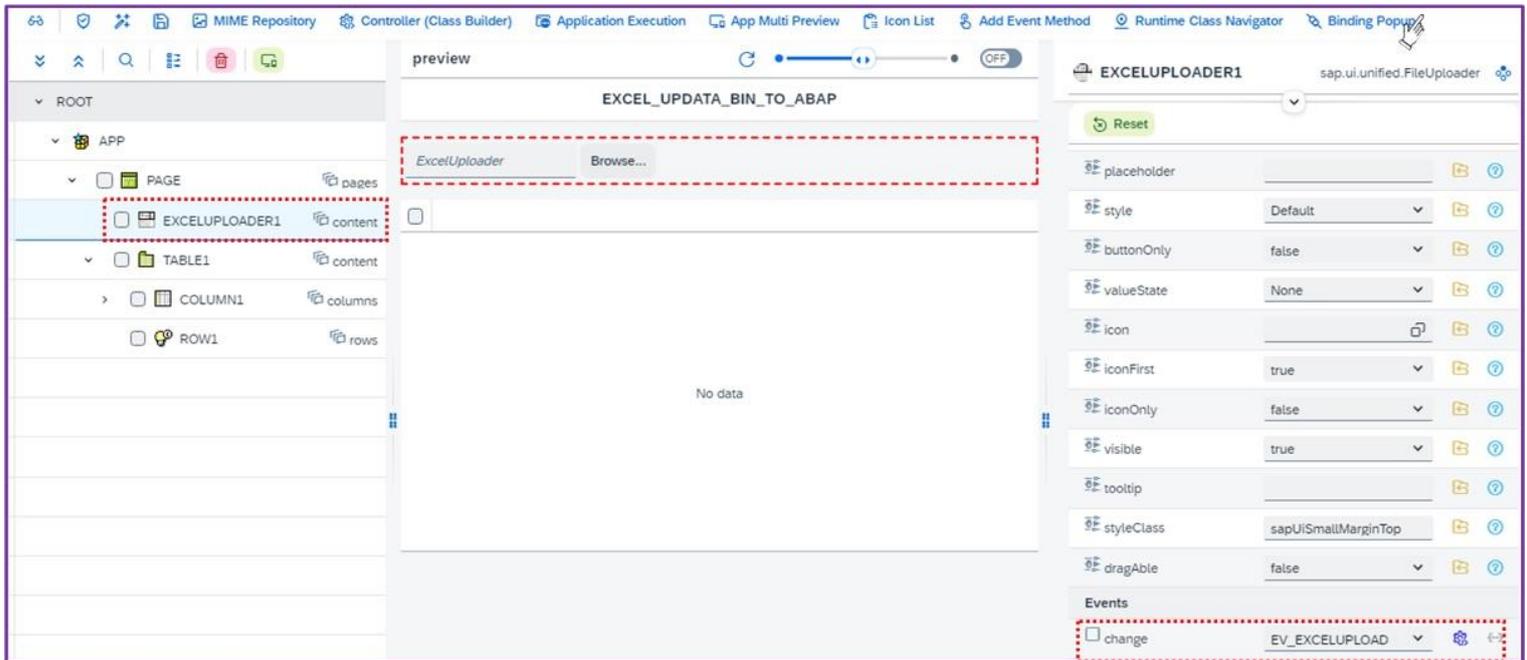
3. " Methodes : GET_MIME_REPOSITORY_API" 를 구성한다.



6.1.50 Methodes EXCEL_UPDATA_BIN_TO_ABAP

■ 목적 : Excel Uploader UI를 통해 업로드한 Excel Data를 itab으로 Parse 처리.

1. Application에서 UI Object와 Event를 생성한다. (EXCELUPLOADER UI만 가능, FILEUPLOADER, UPLOADCOLLECTION 불가)



2. " Methodes : EXCEL_UPDATA_BIN_TO_ABAP" 를 구성한다.

※ EXCEL을 UPLOAD 하는 시점에 "Methodes : EXCEL_UPDATA_BIN_TO_ABAP" 수행하고 이 정보를 통하여 TABLE을 생성한다.

[첨부ABAP SOURCE (EV_EXCELUPLOAD)]

METHOD EV_EXCELUPLOAD.

```

TYPES: BEGIN OF TY_OTAB,
        COL1 TYPE STRING,
        COL2 TYPE STRING,
        COL3 TYPE STRING,
        COL4 TYPE STRING,
        COL5 TYPE STRING,
        COL6 TYPE STRING,
        COL7 TYPE STRING,
        COL8 TYPE STRING,
        COL9 TYPE STRING,
        COL10 TYPE STRING,
END OF TY_OTAB.

```

```

DATA: LS_FILE TYPE /U4A/S0006,
      LT_OTAB TYPE TABLE OF TY_OTAB.

```

```

READ TABLE IT_FILES INTO LS_FILE INDEX 1.
CHECK SY-SUBRC EQ 0.

```

"Excel UPLOAD Data TO ABAP Content"

```

CALL METHOD /U4A/CL_UTILITIES=>EXCEL_UPDATA_BIN_TO_ABAP
EXPORTING
  I_CONTENT = LS_FILE-CONTENT
IMPORTING
  ET_OTAB   = LT_OTAB.

```

"EXCEL 정보를 이용하여 TABLE 생성."

```

CALL METHOD SET_TABLE
EXPORTING
  IT_OTAB = LT_OTAB.

```

```

ENDMETHOD.

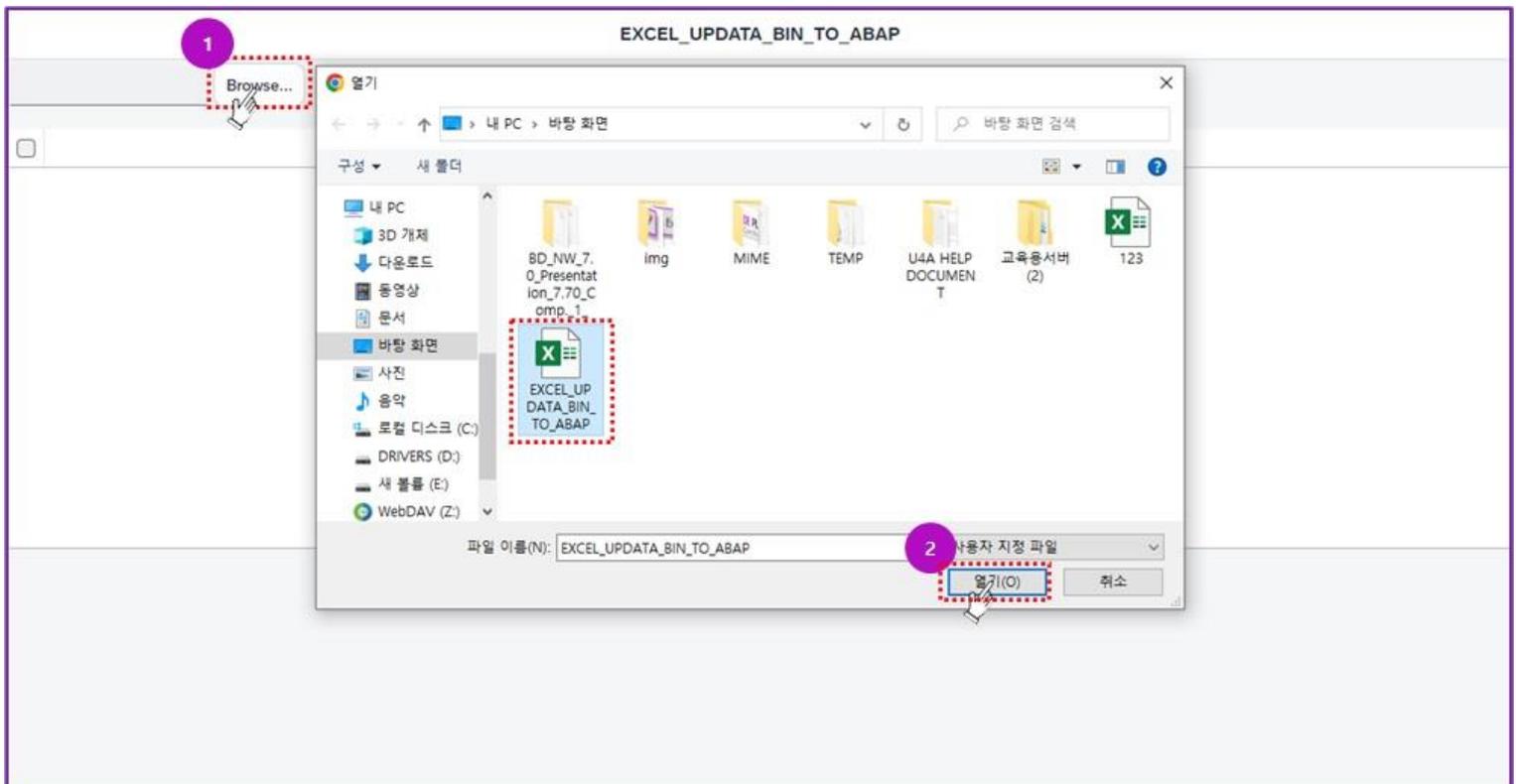
```

※ "Methodes : EXCEL_UPDATA_BIN_TO_ABAP" 수행한 결과값.

Table: LT_OTAB
 Attributes: Standard [101x10(80)]

Row	COL1 [CString]	COL2 [CString]	COL3 [CString]	COL4 [CString]	COL5 [CString]	COL6 [CString]	COL7 [CString]	COL8 [CString]	COL9 [CString]	COL10 [CString]
1	A	B	C	D	E	F	G	H	I	J
2	TEST A001	TEST B001	TEST C001	TEST D001	TEST E001	TEST F001	TEST G001	TEST H001	TEST I001	TEST J001
3	TEST A002	TEST B002	TEST C002	TEST D002	TEST E002	TEST F002	TEST G002	TEST H002	TEST I002	TEST J002
4	TEST A003	TEST B003	TEST C003	TEST D003	TEST E003	TEST F003	TEST G003	TEST H003	TEST I003	TEST J003
5	TEST A004	TEST B004	TEST C004	TEST D004	TEST E004	TEST F004	TEST G004	TEST H004	TEST I004	TEST J004
6	TEST A005	TEST B005	TEST C005	TEST D005	TEST E005	TEST F005	TEST G005	TEST H005	TEST I005	TEST J005
7	TEST A006	TEST B006	TEST C006	TEST D006	TEST E006	TEST F006	TEST G006	TEST H006	TEST I006	TEST J006
8	TEST A007	TEST B007	TEST C007	TEST D007	TEST E007	TEST F007	TEST G007	TEST H007	TEST I007	TEST J007
9	TEST A008	TEST B008	TEST C008	TEST D008	TEST E008	TEST F008	TEST G008	TEST H008	TEST I008	TEST J008
10	TEST A009	TEST B009	TEST C009	TEST D009	TEST E009	TEST F009	TEST G009	TEST H009	TEST I009	TEST J009
11	TEST A010	TEST B010	TEST C010	TEST D010	TEST E010	TEST F010	TEST G010	TEST H010	TEST I010	TEST J010
12	TEST A011	TEST B011	TEST C011	TEST D011	TEST E011	TEST F011	TEST G011	TEST H011	TEST I011	TEST J011
13	TEST A012	TEST B012	TEST C012	TEST D012	TEST E012	TEST F012	TEST G012	TEST H012	TEST I012	TEST J012
14	TEST A013	TEST B013	TEST C013	TEST D013	TEST E013	TEST F013	TEST G013	TEST H013	TEST I013	TEST J013
15	TEST A014	TEST B014	TEST C014	TEST D014	TEST E014	TEST F014	TEST G014	TEST H014	TEST I014	TEST J014
16	TEST A015	TEST B015	TEST C015	TEST D015	TEST E015	TEST F015	TEST G015	TEST H015	TEST I015	TEST J015
17	TEST A016	TEST B016	TEST C016	TEST D016	TEST E016	TEST F016	TEST G016	TEST H016	TEST I016	TEST J016
18	TEST A017	TEST B017	TEST C017	TEST D017	TEST E017	TEST F017	TEST G017	TEST H017	TEST I017	TEST J017
19	TEST A018	TEST B018	TEST C018	TEST D018	TEST E018	TEST F018	TEST G018	TEST H018	TEST I018	TEST J018
20	TEST A019	TEST B019	TEST C019	TEST D019	TEST E019	TEST F019	TEST G019	TEST H019	TEST I019	TEST J019
21	TEST A020	TEST B020	TEST C020	TEST D020	TEST E020	TEST F020	TEST G020	TEST H020	TEST I020	TEST J020
22	TEST A021	TEST B021	TEST C021	TEST D021	TEST E021	TEST F021	TEST G021	TEST H021	TEST I021	TEST J021
23	TEST A022	TEST B022	TEST C022	TEST D022	TEST E022	TEST F022	TEST G022	TEST H022	TEST I022	TEST J022
24	TEST A023	TEST B023	TEST C023	TEST D023	TEST E023	TEST F023	TEST G023	TEST H023	TEST I023	TEST J023
25	TEST A024	TEST B024	TEST C024	TEST D024	TEST E024	TEST F024	TEST G024	TEST H024	TEST I024	TEST J024

3. Application 실행 후 "EXCELUPLOADER" UI에 "EXCEL 파일을 업로드 한다.



4. TABLE UI에 EXCEL 정보가 매핑되었는지 확인.

EXCEL_UPDATA_BIN_TO_ABAP										
EXCEL_UPDATA_BIN_... Browse...										
A	B	C	D	E	F	G	H	I	J	
<input type="checkbox"/>	TEST A001	TEST B001	TEST C001	TEST D001	TEST E001	TEST F001	TEST G001	TEST H001	TEST I001	TEST J001
<input type="checkbox"/>	TEST A002	TEST B002	TEST C002	TEST D002	TEST E002	TEST F002	TEST G002	TEST H002	TEST I002	TEST J002
<input type="checkbox"/>	TEST A003	TEST B003	TEST C003	TEST D003	TEST E003	TEST F003	TEST G003	TEST H003	TEST I003	TEST J003
<input type="checkbox"/>	TEST A004	TEST B004	TEST C004	TEST D004	TEST E004	TEST F004	TEST G004	TEST H004	TEST I004	TEST J004
<input type="checkbox"/>	TEST A005	TEST B005	TEST C005	TEST D005	TEST E005	TEST F005	TEST G005	TEST H005	TEST I005	TEST J005
<input type="checkbox"/>	TEST A006	TEST B006	TEST C006	TEST D006	TEST E006	TEST F006	TEST G006	TEST H006	TEST I006	TEST J006
<input type="checkbox"/>	TEST A007	TEST B007	TEST C007	TEST D007	TEST E007	TEST F007	TEST G007	TEST H007	TEST I007	TEST J007
<input type="checkbox"/>	TEST A008	TEST B008	TEST C008	TEST D008	TEST E008	TEST F008	TEST G008	TEST H008	TEST I008	TEST J008
<input type="checkbox"/>	TEST A009	TEST B009	TEST C009	TEST D009	TEST E009	TEST F009	TEST G009	TEST H009	TEST I009	TEST J009
<input type="checkbox"/>	TEST A010	TEST B010	TEST C010	TEST D010	TEST E010	TEST F010	TEST G010	TEST H010	TEST I010	TEST J010

Figure : < "EXCEL_UPDATA_BIN_TO_ABAP"을 수행하여 TABLE 생성 >

	A	B	C	D	E	F	G	H	I	J
1	A	B	C	D	E	F	G	H	I	J
2	TEST A001	TEST B001	TEST C001	TEST D001	TEST E001	TEST F001	TEST G001	TEST H001	TEST I001	TEST J001
3	TEST A002	TEST B002	TEST C002	TEST D002	TEST E002	TEST F002	TEST G002	TEST H002	TEST I002	TEST J002
4	TEST A003	TEST B003	TEST C003	TEST D003	TEST E003	TEST F003	TEST G003	TEST H003	TEST I003	TEST J003
5	TEST A004	TEST B004	TEST C004	TEST D004	TEST E004	TEST F004	TEST G004	TEST H004	TEST I004	TEST J004
6	TEST A005	TEST B005	TEST C005	TEST D005	TEST E005	TEST F005	TEST G005	TEST H005	TEST I005	TEST J005
7	TEST A006	TEST B006	TEST C006	TEST D006	TEST E006	TEST F006	TEST G006	TEST H006	TEST I006	TEST J006
8	TEST A007	TEST B007	TEST C007	TEST D007	TEST E007	TEST F007	TEST G007	TEST H007	TEST I007	TEST J007
9	TEST A008	TEST B008	TEST C008	TEST D008	TEST E008	TEST F008	TEST G008	TEST H008	TEST I008	TEST J008
10	TEST A009	TEST B009	TEST C009	TEST D009	TEST E009	TEST F009	TEST G009	TEST H009	TEST I009	TEST J009
11	TEST A010	TEST B010	TEST C010	TEST D010	TEST E010	TEST F010	TEST G010	TEST H010	TEST I010	TEST J010

Figure : < 업로드한 Excel 양식 >

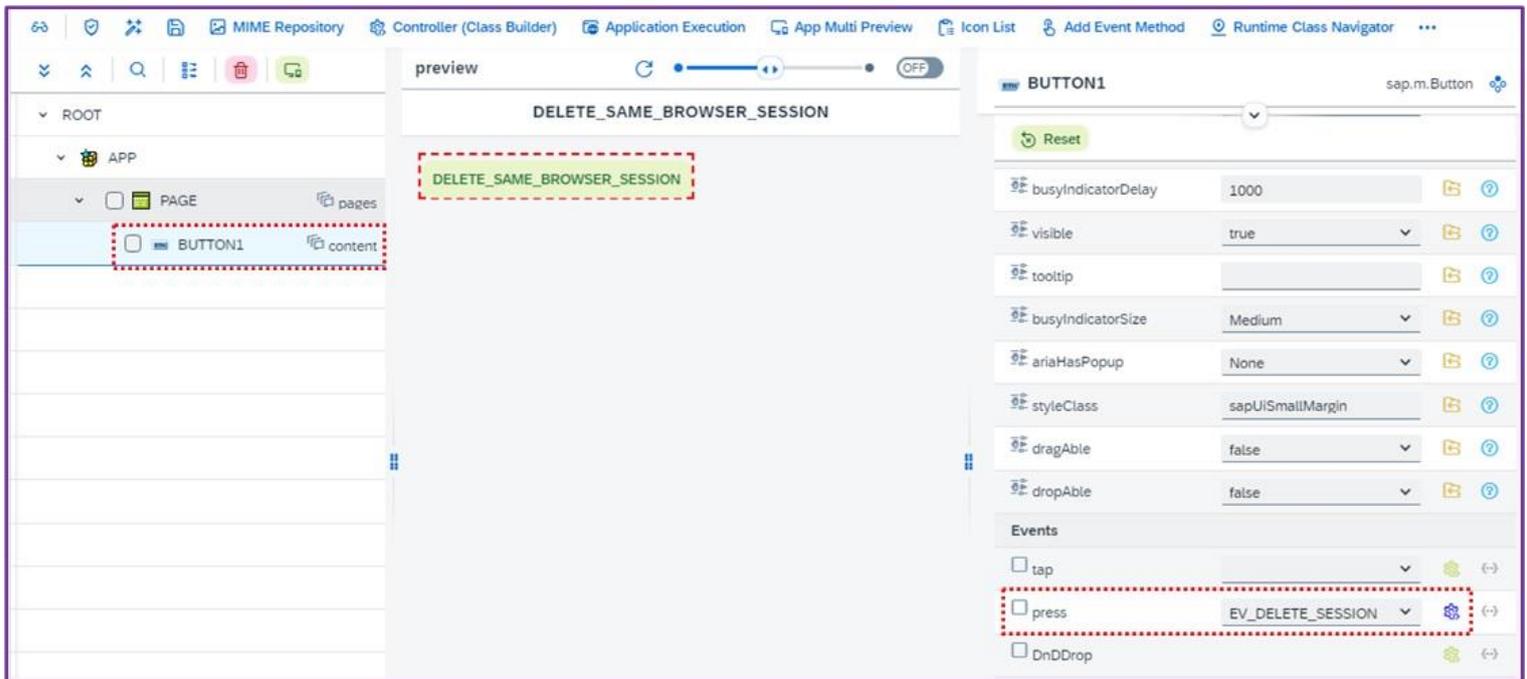
6.1.51

Methodes

DELETE_SAME_BROWSER_SESSION

■ 목적 : 같은 Browser 안에서 Event를 수행하는 Application을 제외한 "다른 Application 세션 삭제".

1. Application에서 UI Object와 Event를 생성한다. (Button 사용)



2. " Methodes : DELETE_SAME_BROWSER_SESSION" 를 구성한다.

[첨부ABAP SOURCE (EV_DELETE_SESSION)]

METHOD EV_DELETE_SESSION.

"같은 브라우저내에 EVENT를 수행하는 Application을 제외한 다른 Application 세션삭제.

CALL METHOD /U4A/CL_UTILITIES=>DELETE_SAME_BROWSER_SESSION

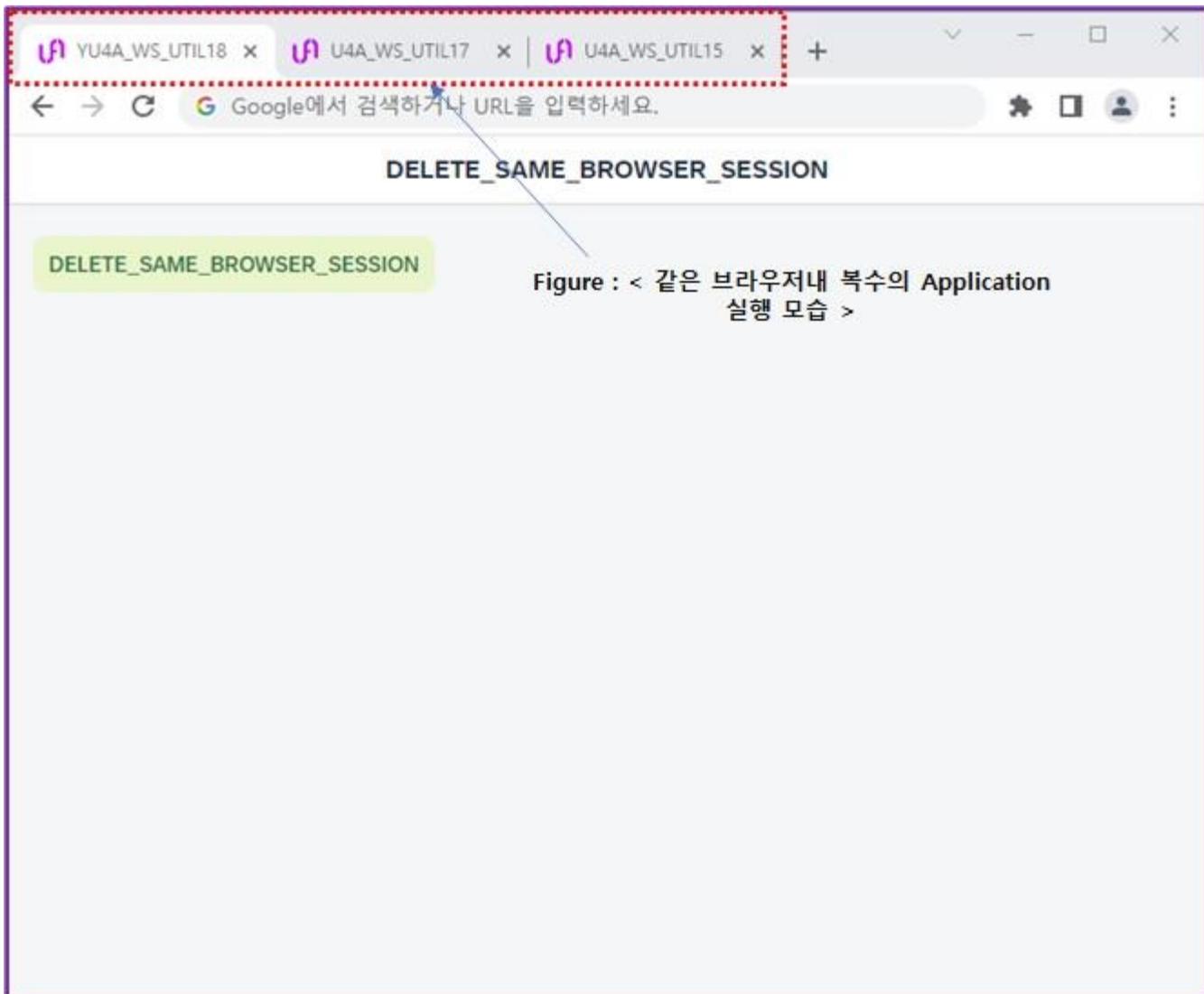
EXPORTING

IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class

I_APPID = 'ALL'. " 대상 APPLICATION(자신 APPLICATION은 세션종료 불가)

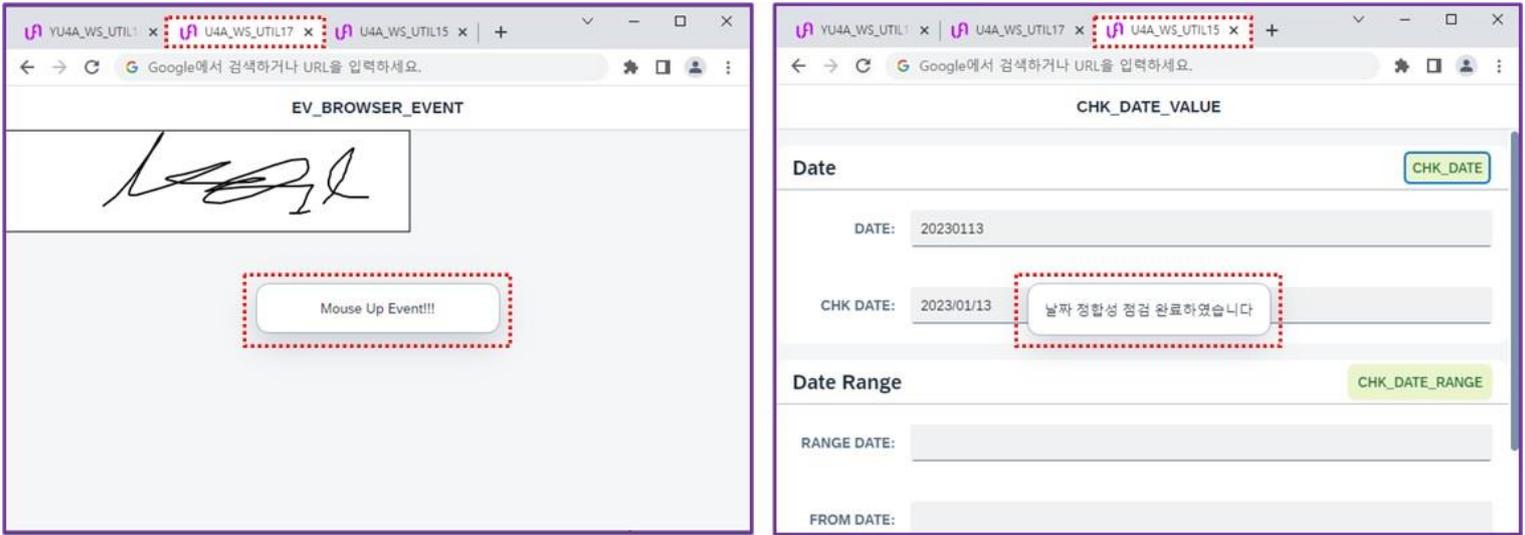
ENDMETHOD.

3. 같은 브라우저 안에서 해당 Application을 포함한 복수의 Application을 실행한다(본 단원에서는 총 3개 실행).



① "DELETE_SAME_BROWSER_SESSION" Button 클릭 안 한 경우

=> 다른 Application의 세션이 유지되어 있으므로 Application 사용가능.



② "DELETE_SAME_BROWSER_SESSION" Button 클릭 한 경우

=> 다른 Application의 세션이 삭제되어 Application 사용 불가.

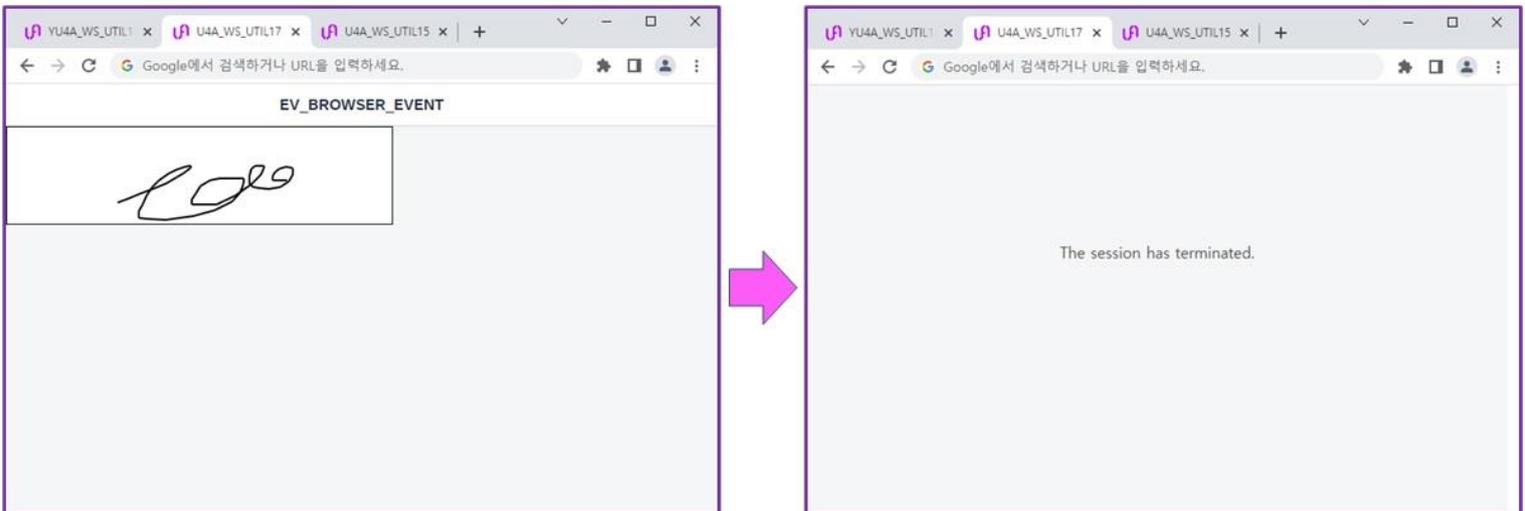


Figure : < Mouseup Event 요청 했지만 세션을 삭제 시켰기 때문에 Event 수행 불가

U4A IDE HELP

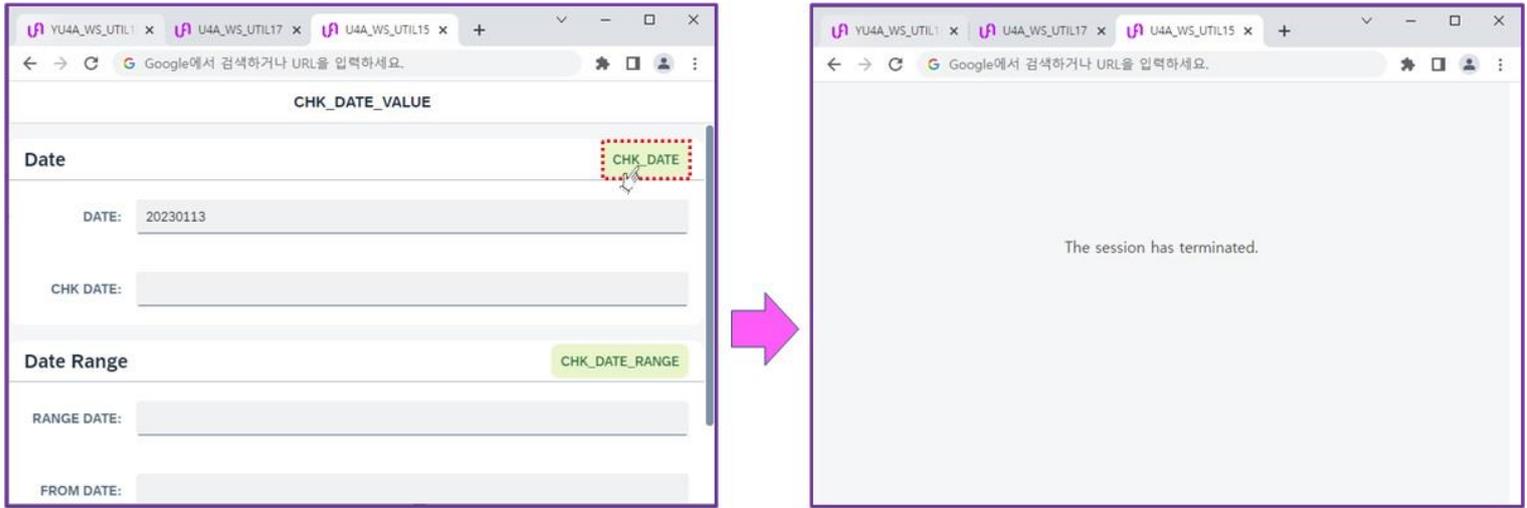
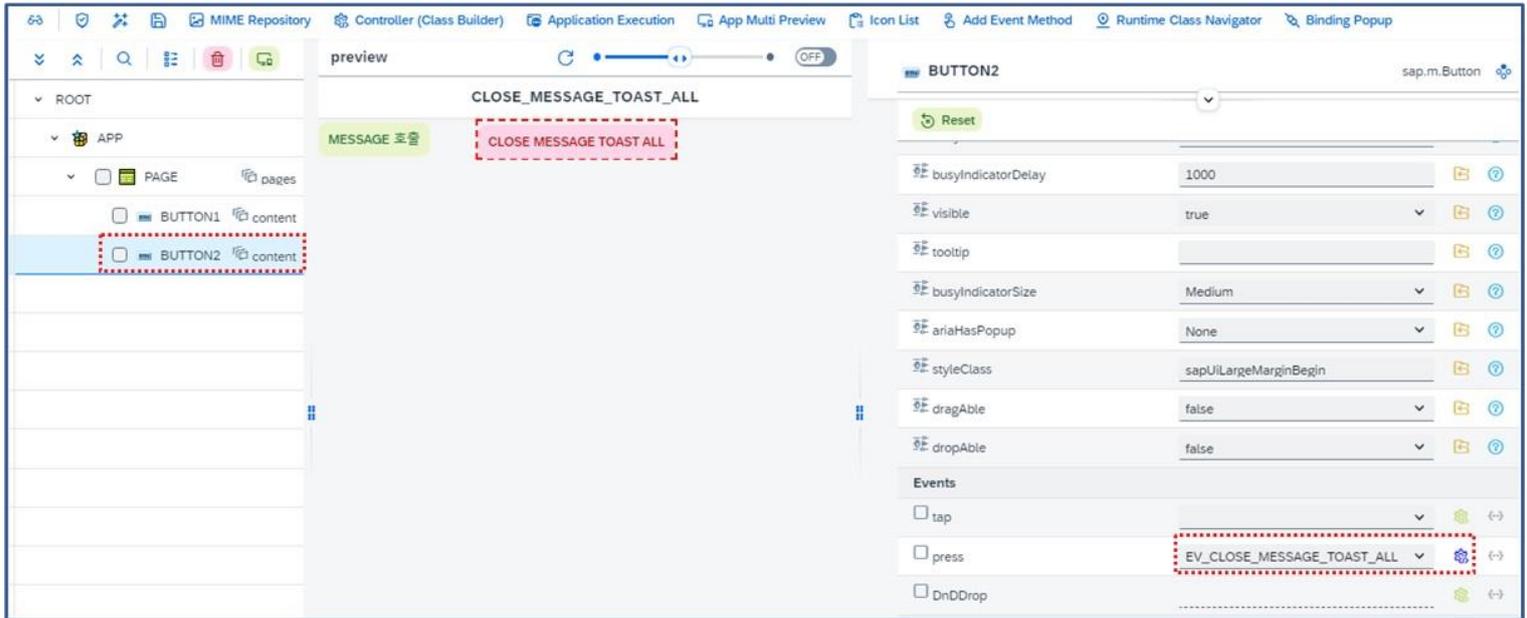


Figure : < 날짜 정합성 체크 Event 요청 했지만 세션을 삭제 시켰기 때문에 Event 수행 불가 >

6.1.52 Methodes CLOSE_MESSAGE_TOAST_ALL

■ 목적 : Application에서 호출된 "모든 MESSAGE TOAST"를 강제 종료 시키기 위함.

1. Application에서 UI Object와 Event를 생성한다. (Button 사용)



2. " Methodes : CLOSE_MESSAGE_TOAST_ALL" 를 구성한다.

[첨부ABAP SOURCE (EV_CLOSE_MESSAGE_TOAST_ALL)]

METHOD EV_CLOSE_MESSAGE_TOAST_ALL .

"M_MESSAGE_TOAST 로 호출한 팝업 강제 종료 처리 메소드.

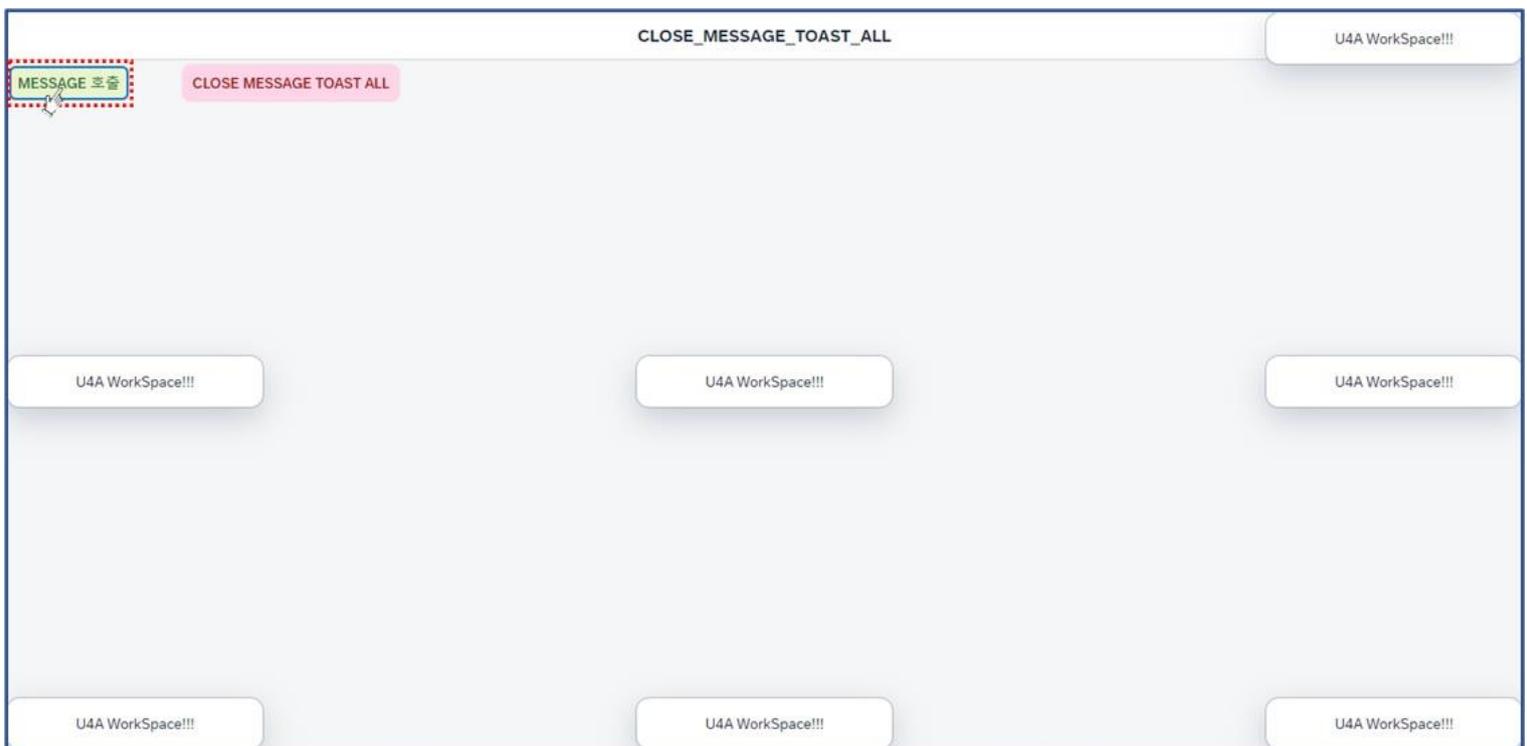
CALL METHOD /U4A/CL_UTILITIES=>CLOSE_MESSAGE_TOAST_ALL

EXPORTING

IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW. " [U4A] UI Element Super Class

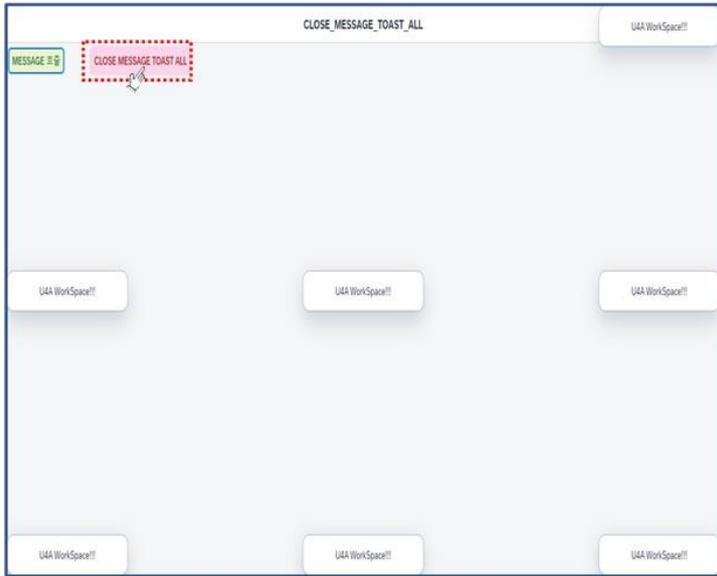
ENDMETHOD.

3. Application 실행 후 "MESSAGE 호출" Button Click 하여 복수의 MESSAGE TOAST 호출.



4. 복수의 MESSAGE TOAST 호출된 상태에서 "CLOSE MESSAGE TOAST ALL" Button Click하면 모든 MESSAGE TOAST 강제 종료.

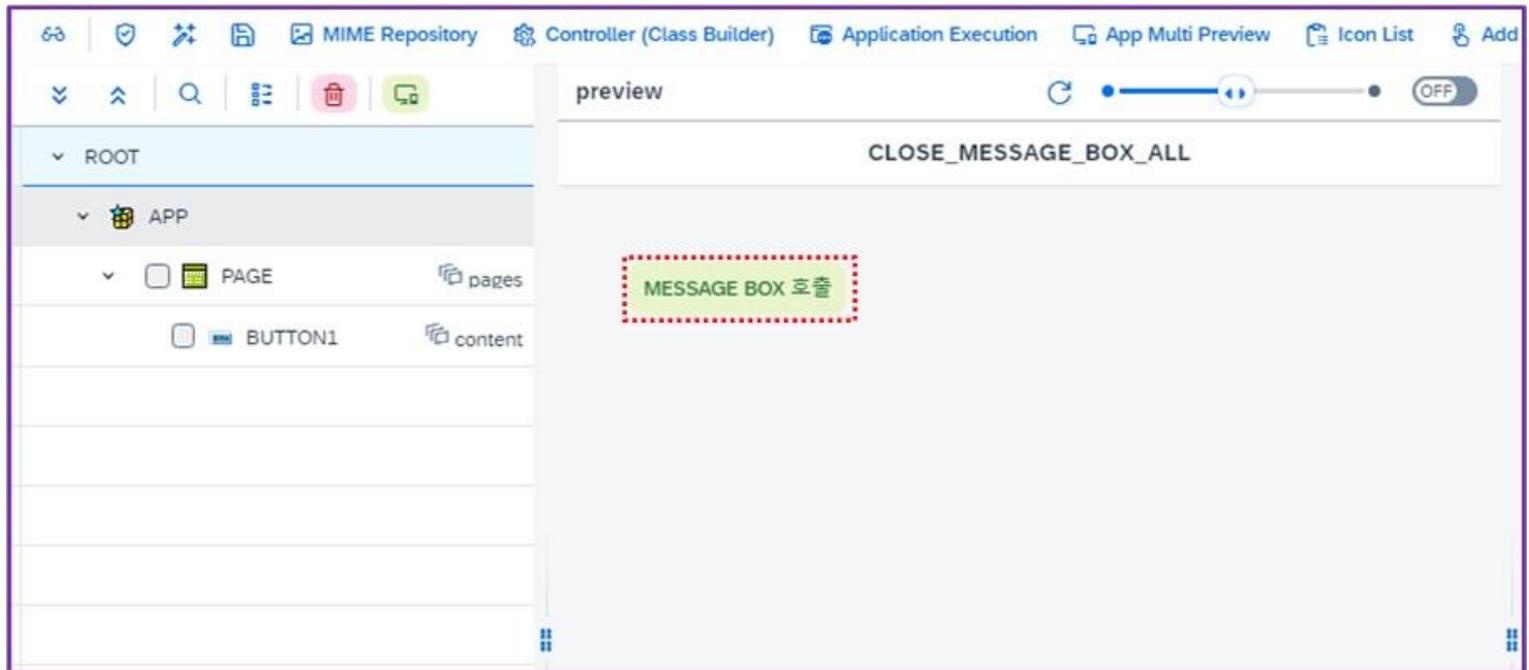
U4A IDE HELP



6.1.53 Methodes CLOSE_MESSAGE_BOX_ALL

▣ 목적 : Application에서 호출된 "모든 MESSAGE BOX"를 강제 종료 시키기 위함.

1. Application에서 UI Object와 Event를 생성한다. (Button 사용)



2. 메세지 박스의 복수 호출 및 메세지 박스의 콜백 이벤트를 설정한다.

① 메세지 박스를 복수 호출 하는 "EV_MESSAGE" 구성

[첨부ABAP SOURCE (EV_MESSAGE)]

```
METHOD EV_MESSAGE.
```

```
"메세지 박스 복수 호출( 100개 ).
```

```
DO 100 TIMES.
```

```
CALL METHOD /U4A/CL_UTILITIES=>M_MESSAGEBOX
```

```
EXPORTING
```

```
IO_VIEW = ME-
```

```
>/U4A/IF_SERVER~AR_VIEW
```

```
" [U4A] UI Element Super Class
```

```
I_MSGTX = '호출된 메세지 박스 모두 종료 하시겠습니까?' " MESSAGE TEXT
```

```
I_POPUP_TYPE = /U4A/CL_UTILITIES=>CS_M_MSG_BOX_TP-
```

```
CONFIRM " ZCL_U4A_UTILITIES=>CS_M_MSG_BOX_TP
```

```
I_TITLE = 'U4A'
```

```
* I_ICON =
```

```
I_CALLBACK_EVENT = 'EV_CLOSE_MESSAGE_BOX_ALL'
```

```
* I_ACTION = " 처리 액션 버튼 정의
```

```
* I_DETAILS = " Long text (상세)
```

```
* I_TEXTDIRECTION =
```

```
* I_INITIALFOCUS = " Initial button focus i  
s set by attribute
```

```
ENDDO.
```

```
ENDMETHOD.
```

② 메세지 박스 호출 뒤 "OK 또는 CANCEL" 선택에 따른 콜백 이벤트
"EV_CLOSE_MESSAGE_BOX_ALL" 구성

[첨부ABAP SOURCE (EV_CLOSE_MESSAGE_BOX_ALL)]

```
METHOD EV_CLOSE_MESSAGE_BOX_ALL.
```

```
"OK 누른경우( 메세지 박스 모두 종료 ) / CANCEL 누르면 한개씩 종료.
```

```
CHECK I_EVENT_NAME EQ 'OK'.
```

"M_MESSAGEBOX로 호출한 팝업 강제 종료 처리 메소드.

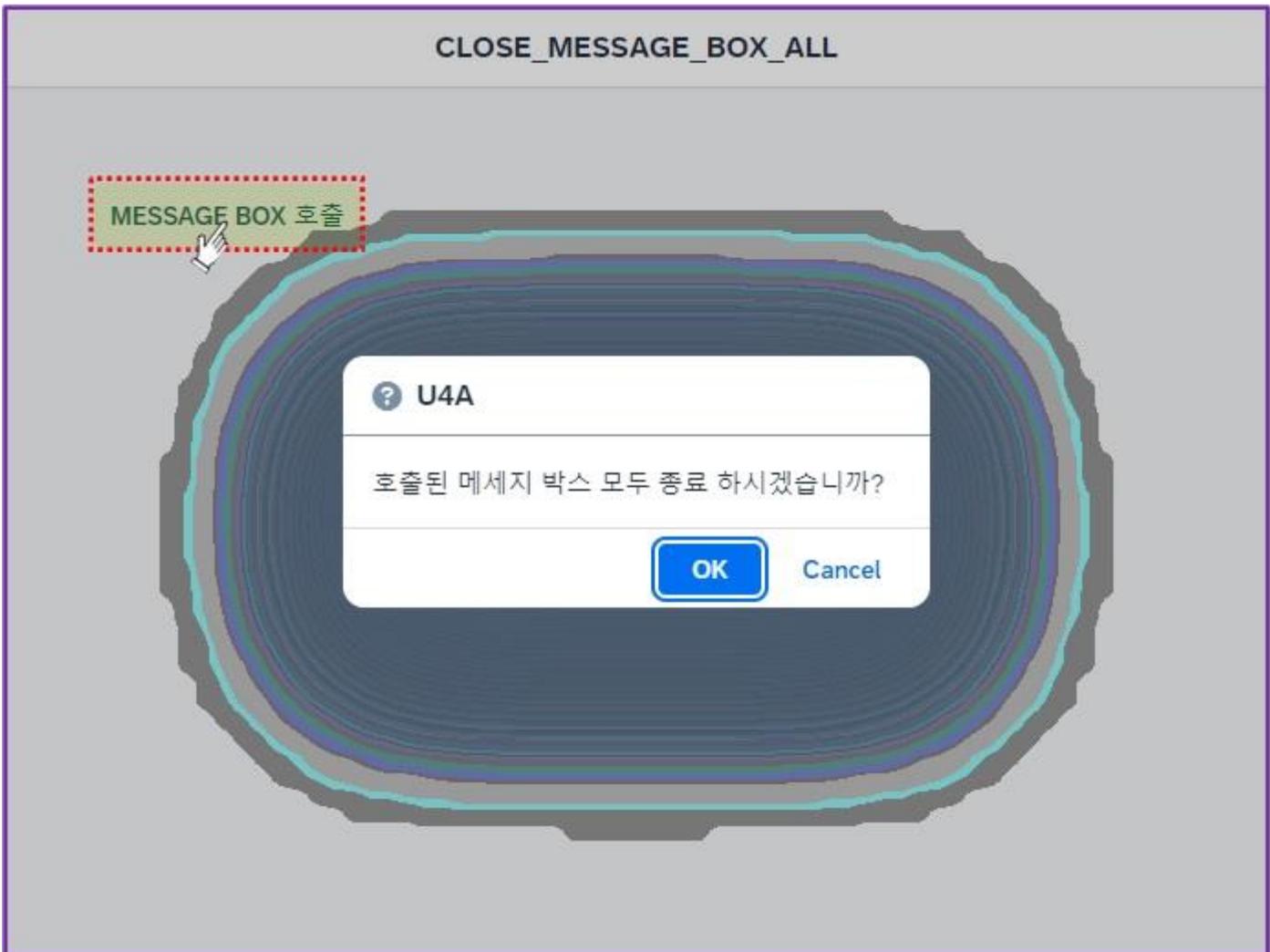
```
CALL METHOD /U4A/CL_UTILITIES=>CLOSE_MESSAGE_BOX_ALL
```

```
EXPORTING
```

```
IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW. " [U4A] UI Element Super Class
```

```
ENDMETHOD.
```

3. Application 실행 후 "MESSAGE BOX 호출" Button Click 하여 복수의 MESSAGE BOX 호출.



4. "OK" Button Click => 100개의 메세지 박스 모두 종료 / "Cancel" Button Click => 1개의 메세지 박스만 종료

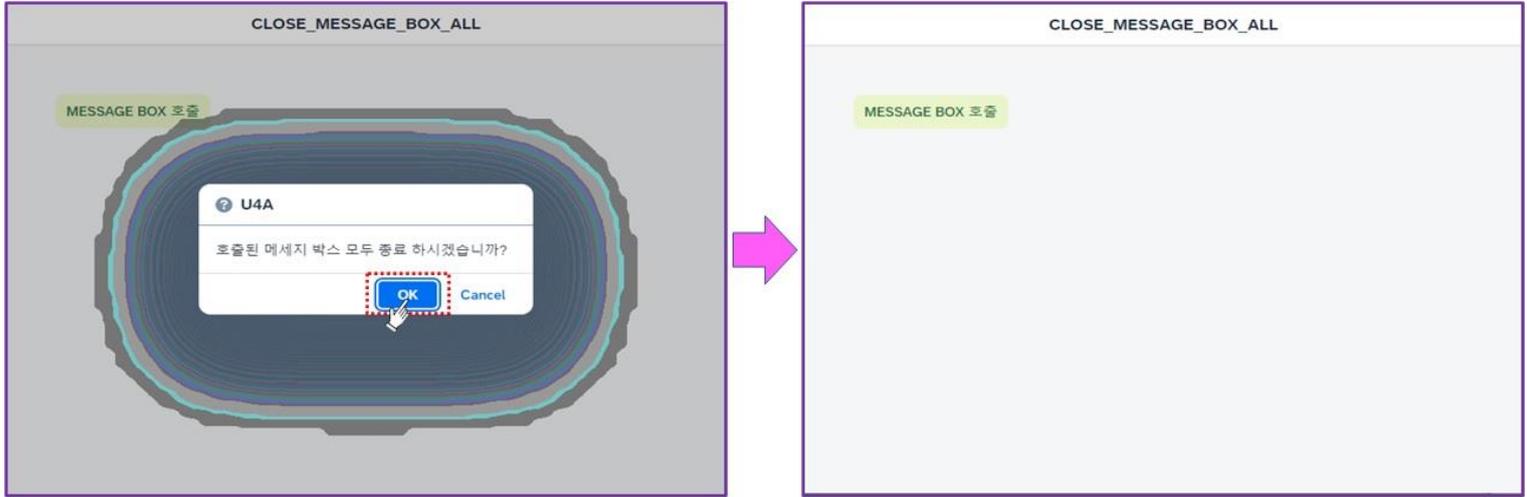
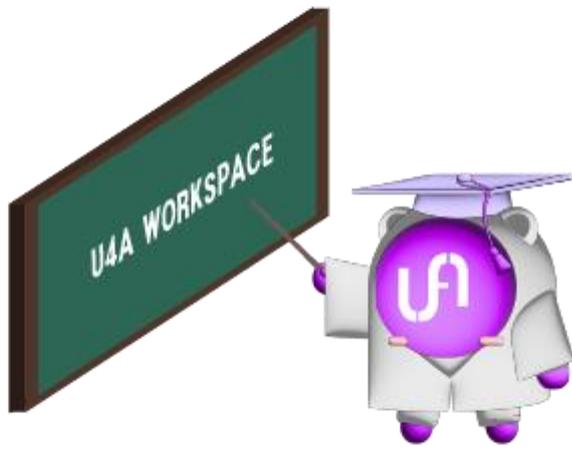


Figure : < "OK" 버튼 클릭하여 100개의 MESSAGE BOX를 모두 종료시킨 상황 >

6.2 ABAP Class /U4A/CL_F4_HNADL



6.2.1 Methodes F4_HNADL_SH

▣ 목적 : DDLC에서 만들어진 F4_HELP의 가능 Entries를 별도의 변경 없이 그대로 사용하기 위한 목적.

1. SEARCH HELP를 설정하기 위한 UI 구성.

※ 본 단원에서는 INPUT FIELD를 이용한 SEARCH HELP 예시 소개.

The screenshot displays the SAP IDE interface for configuring an input field. On the left, the object browser shows a hierarchy: ROOT > APP > PAGE > TABLE1 > COLUMNLISTITEM1 > INPUT1. The center pane shows a preview of the 'INPUT' field, which is highlighted with a red dashed box. The right pane shows the configuration for the 'INPUT' field, with several properties and events highlighted by red dashed boxes:

- Field Properties:**
 - showValueHelp: true
 - showSuggestion: false
 - valueHelpOnly: false
 - filterSuggests: true
 - maxSuggestionWidth: (empty)
 - startSuggestion: 1
 - showTableSuggestionValue: true
 - description: (empty)
 - fieldWidth: 50%
 - valueLiveUpdate: false
 - selectedKey: (empty)
 - textFormatMode: Value
 - textFormatter: (empty)
 - suggestionRowValidator: (empty)
 - enableSuggestionsHighlight: true
 - value: GT_MAT-MATNR
 - width: (empty)
- Events:**
 - liveChange: (empty)
 - valueHelpRequest: EV_F4_SH_OPEN
 - suggest: (empty)
 - suggestionItemSelected: (empty)
 - submit: (empty)
 - change: (empty)
 - DnDDrop: (empty)

2. SEARCH HELP LAYOUT 구성 및 Field Option 설정.

[첨부 ABAP SOURCE (EV_F4_SH_OPEN)]

METHOD EV_F4_SH_OPEN.

```
DATA: LV_INDEX TYPE I.
DATA: LS_LAYO TYPE /U4A/CL_F4_HNADL=>GTY_F4_CONFIG.
DATA: LS_INFO TYPE /U4A/S9997.
```

"Search Help layout

```
LS_LAYO-MAXHITS      = 450.           " 조건 MAX HIT
LS_LAYO-AUTO_SEARCH = ABAP_FALSE .   " 초기 팝업 OPEN 자동 조회
LS_LAYO-WIDTH       = '1000px'.     " F4 팝업 넓이
LS_LAYO-HEIGHT     = '800px'.     " F4 팝업 높이
LS_LAYO-SELMULT    = ABAP_FALSE.    " 선택 멀티 여부 (예:X)
LS_LAYO-STYLECLASS = ''.           " CSS Class
LS_LAYO-DRAGGABLE  = ABAP_TRUE.     " 팝업 Draggable
```

" 필드 정보 탭(LS_LAYO-FIELD_INFO)

```
LS_INFO-SHLPNAME    = 'MATOM'.       " Search Help 명
LS_INFO-SHLPFIELD   = 'MATNR'.       " Search Help 필드명
LS_INFO-INITVALUE   = '1234'.        " 필드 초기값
LS_INFO-PLACEHOLDER = 'Material Number'. " Place Holder 값
LS_INFO-WIDTH       = '300px'.       " 필드 넓이
LS_INFO-DISPLAY     = ABAP_FALSE.    " 필드 EDIT 가능 여부
```

```
APPEND LS_INFO TO LS_LAYO-FILED_INFO.
```

▣ GTY_F4_CONFIG TYPE을 통해 SEARCH HELP DIALOG LAYOUT 설정 가능.

FIELD	MEAN
MAXHITS	검색 결과 조회 시 최대 건수
AUTO_SEARCH	SEARCH HELP DIALOG OPEN 시에 자동으로 검색 조회
WIDTH	SEARCH HELP DIALOG 가로 넓이 설정
HEIGHT	SEARCH HELP DIALOG 세로 높이 설정
SELMULT	검색 결과 복수 선택 가능 여부
STYLECLASS	SEARCH HELP DIALOG CSS Style Class 설정

DRAGGABLE	SEARC HELP DIALOG DRAG 가능 여부
FILED_INFO	조회 조건 Field Option 설정

▣ **"/U4A/S9997" TYPE을 통해 검색 조건 FIELD Option 설정 가능.**

FIELD	MEAN
SHLPNAME	SEARCH HELP 명
SHLPFIELD	Option 변경 할 Field 명
INITVALUE	SEARCH HELP DIALOG OPEN 시 미리 입력 되어 있을 FIELD 초기 값
PLACEHOLDER	PLACEHOLDER 값
WIDTH	FIELD 가로 길이
DISPLAY	FIELD EDIT 가능 여부

3. SEARCH HELP DIALOG SCREEN.

The screenshot shows a search interface with the following elements highlighted:

- WIDTH**: A label pointing to the top search bar area.
- HEIGHT**: A label pointing to the right side of the search bar area.
- SEARCH**: A label pointing to the search button.
- QUERY CONDITION FIELD**: A label pointing to the input fields for Description, Language, and Material.
- MAXHITS**: A label pointing to the 'Maximum No. of Hits' field, which is set to 450.
- ROWS**: A label pointing to the 'Search Result : 7' indicator.
- RESULT**: A label pointing to the table of search results.

Descr.	Language	Material
TEST SEARCH	EN	1234TEST
TEST	EN	AB-1234
TEST ABC-123	EN	ABC-1234
AFS T-SHIRT - PROTOTYPE	EN	AFS-T1234
NEW BATTERY (AG)	EN	H123453
INTEL CORE I5-760 2.80 8MB/1333 QC CPU	EN	HT-S2011234
TEST ABC-123	EN	SMO-1234

4. 선택한 ROW의 값을 이용한 CALL BACK METHOD 설정.

- CALL BACK METHOD는 ADD EVENT METHOD Button을 통해 생성해야 함.

[첨부 ABAP SOURCE (EV_F4_SH_OPEN)]

```
" GET INDEX
```

```
CALL METHOD /U4A/CL_UTILITIES=>GET_SELECTED_INDEX
```

```
EXPORTING
```

```
I_EVENT_NAME = I_EVENT_NAME
```

```
IMPORTING
```

```
E_INDEX = LV_INDEX.
```

```
" Search Help Open
```

```
CALL METHOD /U4A/CL_F4_HNADL=>F4_SEARCH_HELP
```

```
EXPORTING
```

```
IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW
```

```
I_SHLPNAME = 'MATOM' " Search Help 명
```

```
I_ROW = LV_INDEX " 선택한 ROW
```

```
I_SEVENT_NAME = 'EV_F4_SH_SELECT' " Callback Method
```

```
IS_LAYO_CONF = LS_LAYO. " Search Help Layout 옵션
```

```
ENDMETHOD.
```

5. Call Back Method를 통해 Input Field에 선택한 MATNR 값 설정

(선택한 Row Index는 I_FDATA-ROWINDEX 에 존재함).

[첨부 ABAP SOURCE (EV_F4_SH_SELECT)]

```
METHOD EV_F4_SH_SELECT.
```

```
DATA: I_VALUE TYPE C LENGTH 18.
```

```
DATA: GV_FNAME TYPE STRING.
```

```
FIELD-SYMBOLS : <FS_FDATA> TYPE DATA.
```

```
FIELD-SYMBOLS : <FS_MAT> TYPE TY_MATNR.
```

```
" Search Help Select
```

```
CALL METHOD /U4A/CL_F4_HNADL=>F4_GET_SELECT01
  EXPORTING
    IS_F4DATA = I_FDATA
    I_FLDNM   = 'MATNR'
  IMPORTING
    E_VALUE   = I_VALUE.
```

```
" 선택한 ROW 값 읽어오기
```

```
GV_FNAME = 'I_FDATA-ROWINDEX'.
```

```
ASSIGN (GV_FNAME) TO <FS_FDATA>.
```

```
CHECK <FS_FDATA> IS ASSIGNED.
```

```
" 선택한 MATNR을 INPUT VALUE로 설정
```

```
READ TABLE GT_MAT ASSIGNING <FS_MAT> INDEX <FS_FDATA>.
```

```
CHECK SY-SUBRC EQ 0.
```

```
<FS_MAT>-MATNR = I_VALUE.
```

```
UNASSIGN <FS_FDATA>.
```

```
UNASSIGN <FS_MAT>.
```

```
ENDMETHOD.
```

6. Application 실행 후 원하는 Input Filed의 Search Help Click.

NO	INPUT
1	<input type="text"/> 
2	<input type="text"/> 
3	<input type="text"/> 
4	<input type="text"/> 

7. Search Help Dialog 화면에서 Search 후 원하는 조회 결과 선택.

Material Number/Material Description

Selection
Search

Description:

Language:

Material:

Maximum No. of Hits Search Result : 7

Descr.	Language	Material
TEST SEARCH	EN	1234TEST
TEST	EN	AB-1234
TEST ABC-123	EN	ABC-1234
AFS T-SHIRT - PROTOTYPE	EN	AFS-T1234
NEW BATTERY (AG)	EN	H123453
INTEL CORE I5-760 2.80 8MB/1333 QC CPU	EN	HT-52011234
TEST ABC-123	EN	SMO-1234

Close

8. Call Back Method 결과 (선택한 Row의 Input Field에 검색 조회 결과 시 선택한 Material Number 가 들어간 것을 확인)

U4A IDE HELP

NO	INPUT
1	<input type="text"/>
2	<input type="text"/>
3	ABC-1234 <input type="text"/>
4	<input type="text"/>

6.2.2 Methodes F4_HNADL_SHDYN

■ 목적 : 사용자가 정의한 F4_HELP의 가능 Entries를 사용하기 위한 목적.

1. SEARCH HELP를 설정하기 위한 UI 구성.

본 단원에서는 INPUT FIELD를 이용한 SEARCH HELP 예시 소개.

The screenshot displays the SAP IDE interface for configuring a search help. On the left, a project tree shows a table structure with columns NO, NAME, QUANTITY, and DATE. A search help field is highlighted in the NAME column. The configuration panel on the right shows various properties for the search help, including showValueHelp (true), showSuggestion (false), valueHelpOnly (false), filterSuggests (true), maxSuggestionWidth, startSuggestion (1), showTableSuggestionValue (true), description, fieldWidth (50%), valueLiveUpdate (false), selectedKey, textFormatMode (Value), textFormatter, suggestionRowValidator, enableSuggestionsHighlight (true), value (GT_OTAB-NAME), and width. The Events section shows the valueHelpRequest event configured with the event name EV_F4_SH_OPEN.

2. "Search Help" 조회를 위한 Table 구성.

[첨부 ABAP SOURCE (EV_F4_DYN_SH_OPEN)]

METHOD EV_F4_DYN_SH_OPEN.

DATA: LV_INDEX TYPE I.

```

DATA: LT_FINFO    TYPE /U4A/CL_F4_HNADL=>GTT_FLD_INFO.
DATA: LS_FINFO    TYPE /U4A/CL_F4_HNADL=>GTY_FLD_INFO.
DATA: LS_ITAB     TYPE TY_OTAB.
DATA: LT_ITAB     TYPE TABLE OF TY_OTAB.
DATA: LV_NAME     TYPE STRING.
DATA: LV_NUM      TYPE N LENGTH 2.
DATA: LV_QUANTITY TYPE I.

```

```
" Search Help Table 구성
```

```
DO 10 TIMES.
```

```
ADD 1 TO LV_NUM.
```

```
CONCATENATE 'OBJECT_' LV_NUM INTO LV_NAME.
```

```
ADD 100 TO LV_QUANTITY.
```

```
LS_ITAB-NO      = LV_NUM.
```

```
LS_ITAB-NAME    = LV_NAME.
```

```
LS_ITAB-QUANTITY = LV_QUANTITY.
```

```
LS_ITAB-DATE    = SY-DATUM.
```

```
APPEND LS_ITAB TO LT_ITAB.
```

```
CLEAR LS_ITAB.
```

```
CLEAR LV_NAME.
```

```
ENDDO.
```

3. SEARCH HELP의 검색조건 Field 추가 및 Option 설정.

[첨부 ABAP SOURCE (EV_F4_DYN_SH_OPEN)]

```
" 검색조건 필드 속성 추가
```

```
LS_FINFO-FNAME = 'NAME'.
```

```
" 필드명
```

```

LS_FINFO-LABEL      = 'Name'.
LS_FINFO-ISSH       = ABAP_TRUE.
LS_FINFO-WIDTH      = '80%'.
위 px,%em 예:100%, 50px...반드시 준수
LS_FINFO-CWIDTH    = '300PX'.
위 px,%em 예:100%, 50px...반드시 준수"
LS_FINFO-MWIDTH     = '40PX'.
위 px,%em 예:100%, 50px...반드시 준수"
LS_FINFO-NOZERO     = ABAP_FALSE.
LS_FINFO-PLACEHOLDER = 'Please enter your name.'.
    
```

```

" 라벨
" 조회조건 입력란으로 사용(X)
" 조회조건 입력필드 길이 *경고 : 단
" 컬럼 길이 *경고 : 단
" minWidth *경고 : 단
" Value Nozero
" 자리 표시자
    
```

```

APPEND LS_FINFO TO LT_FINFO.
CLEAR LS_FINFO.
    
```

```

LS_FINFO-FNAME      = 'QUANTITY'.
LS_FINFO-LABEL      = 'Quantity'.
LS_FINFO-ISSH       = ABAP_TRUE.
LS_FINFO-WIDTH      = '80%'.
위 px,%em 예:100%, 50px...반드시 준수
LS_FINFO-CWIDTH    = '100PX'.
위 px,%em 예:100%, 50px...반드시 준수"
LS_FINFO-MWIDTH     = '40PX'.
위 px,%em 예:100%, 50px...반드시 준수"
LS_FINFO-NOZERO     = ABAP_FALSE.
LS_FINFO-PLACEHOLDER = 'Please enter quantity.'.
    
```

```

" 필드명
" 라벨
" 조회조건 입력란으로 사용(X)
" 조회조건 입력필드 길이 *경고 : 단
" 컬럼 길이 *경고 : 단
" minWidth *경고 : 단
" Value Nozero
" 자리 표시자
    
```

```

APPEND LS_FINFO TO LT_FINFO.
CLEAR LS_FINFO.
    
```

▣ "GTY_F4_CONFIG" 구조의 TABLE에 검색 조건 FIELD 수 만큼 ROW 추가 .

NAME	MEAN
FNAME	검색 조건으로 사용할 Field 이름.
LABEL	조회 결과 테이블의 Column 이름
ISSH	해당 Field를 조회 조건으로 사용 할 지 여부
WIDTH	조회 조건 입력 Field의 가로 넓이

CWIDTH	조회조건 결과 Table의 column 길이
MWIDTH	Search Help Dialog의 최소 가로 넓이
NOZERO	조회 조건 Field 의 Initial 값으로 0 설정 여부
PLACEHOLDER	Placeholder 값 설정

4. 선택한 ROW의 값을 이용한 CALL BACK METHOD 설정 및 SEARCH HELP DIALOG LAYOUT 설정.

[첨부 ABAP SOURCE (EV_F4_DYN_SH_OPEN)]

" 선택한 ROW INDEX 가져오기

```
CALL METHOD /U4A/CL_UTILITIES=>GET_SELECTED_INDEX
EXPORTING
  I_EVENT_NAME = I_EVENT_NAME
IMPORTING
  E_INDEX      = LV_INDEX.
```

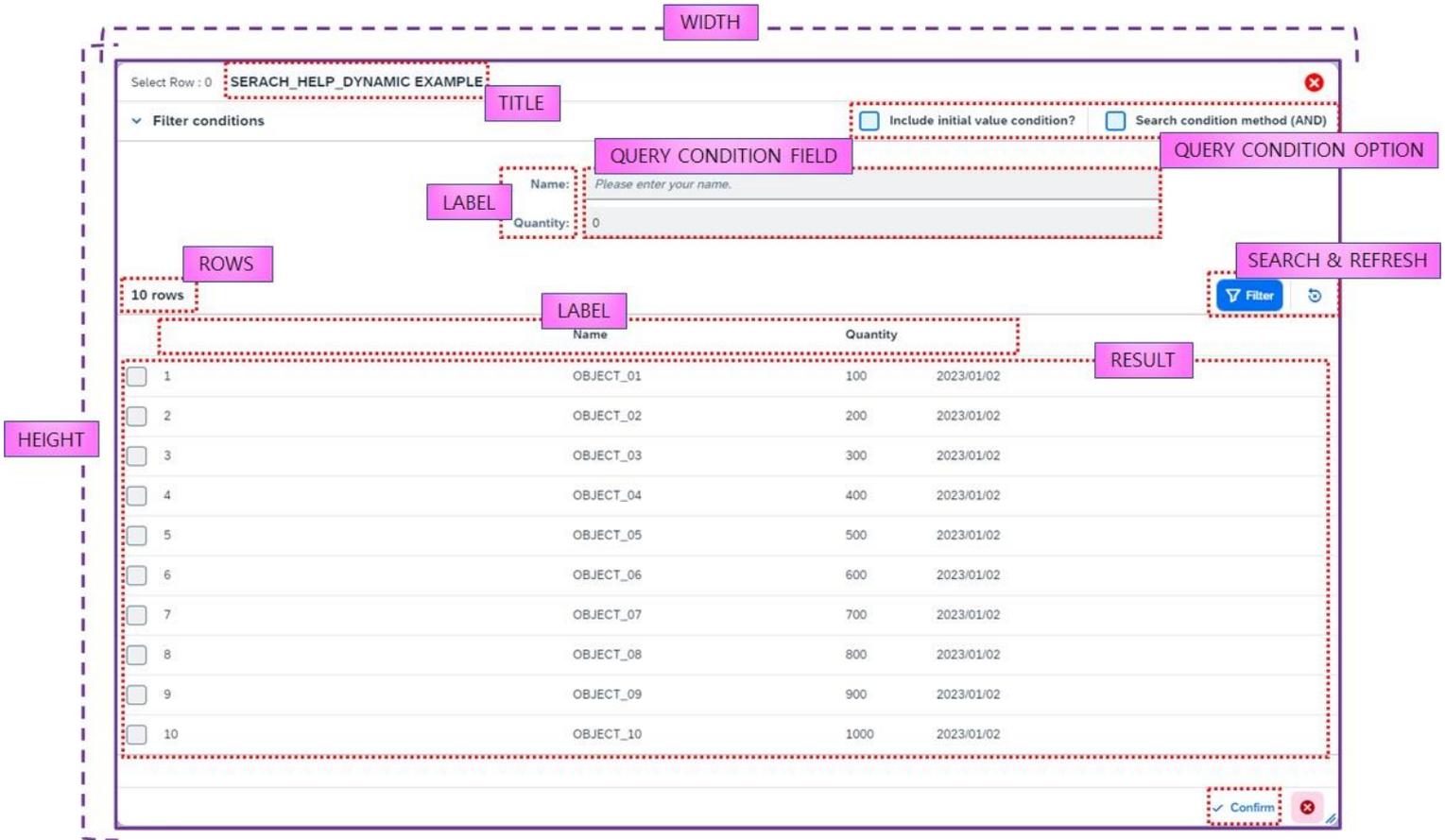
" SEARCH HELP DIALOG OPEN

```
CALL METHOD /U4A/CL_F4_HNADL=>F4_SEARCH_HELP_DYNAMIC
EXPORTING
  IO_VIEW           = ME->/U4A/IF_SERVER~AR_VIEW           " [U4A] UI Element Super Class
  I_CSS_STY         = ABAP_FALSE                          " CSS Style Name
  I_DRAGGABLE       = ABAP_TRUE                           " 팝업 드래그 여부(예 : X)
  I_RESIZABLE       = ABAP_TRUE                           " 팝업 리사이징 여부 (예 : X)
  I_HEIGHT          = '80%'                               " 팝업 높이 - CSS SIZE 단위 px, %
  I_WIDTH           = '70%'                               " 팝업 넓이 - CSS SIZE 단위 px, %
  I_TITLE           = 'SERACH_HELP_DYNAMIC EXAMPLE'      " 팝업 타이틀
  I_MULTSEL         = ABAP_TRUE                           " 리스트 멀티 선택 여부 (예 : X)
  I_EXPANDED        = ABAP_TRUE                           " 조회조건 영역 펼침여부(예 : X)
  I_ROW             = LV_INDEX                            " 호출처 INDEX
  I_SEVENT_NAME     = 'EV_F4_SH_SELECT'                  " CALL BACK 서버 이벤트 명
  IT_FLD_INFO       = LT_FINFO                           " 필드 속성 추가 탭
  IT_DATA           = LT_ITAB.                           " 출력 정보 인터널 테이블 값
```

ENDMETHOD.

NAME	MEAN
I_CSS_STY	Search Help Dialog CSS Style Class 설정
I_DRAGGABLE	Search Help Dialog Drag 가능 여부
I_RESIZABLE	Search Help Dialog Size 조절 가능 여부
I_HEIGHT	Search Help Dialog 세로 높이 설정
I_WIDTH	Search Help Dialog 가로 넓이 설정
I_TITLE	Search Help Dialog의 TITLE
I_MULTSEL	검색 결과 복수 선택 가능 여부
I_EXPANDED	Search Help Dialog 조회 조건 펼침 여부
I_ROW	화면의 DISPLAY TABLE에서 선택한 ROW INDEX
I_SEVENT_NAME	CALL BACK METHOD 명
IT_FLD_INFO	조회 조건 Field 정보 담고 있는 TABLE
IT_DATA	조회 결과 TABLE

5. SEARCH HELP DIALOG SCREEN.



▣ 조회 조건 옵션

① Include Initial Value Condition? : Check 하면 검색 조건에서 Initial 값과 정확히 일치하는 결과를 조회

EX) 이름 Field 값을 없는 상태로 조회 한다면 결과도 이름 Field의 값이 존재하지 않는 결과만 조회

② Search Condition Method(AND) : Check 하면 조회 조건 Field 값을 AND 조건으로 조회, Check 하지 않는다면 OR 조건으로 조회

6. Call Back Method를 통해 Input Field에 조회 결과 Table에서 선택한 ROW 값 입력 .

▣ CALL BACK METHOD는 ADD EVENT METHOD Button을 통해 생성해야 함.

[첨부 ABAP SOURCE (EV_F4_SH_SELECT)]

```
METHOD EV_F4_SH_SELECT.
```

```
DATA: I_VALUE TYPE STRING.
```

```
DATA: I_VALUE2 TYPE STRING.
```

```
DATA: GV_FNAME TYPE STRING.
```

```
FIELD-SYMBOLS: <FS_FDATA> TYPE DATA.
```

```
FIELD-SYMBOLS: <FS_OTAB> TYPE TY_OTAB.
```

```
" Search Help Select
```

```
CALL METHOD /U4A/CL_F4_HNADL=>F4_GET_SELECT01
```

```
EXPORTING
```

```
IS_F4DATA = I_FDATA
```

```
I_FLDNM = 'NAME'
```

```
IMPORTING
```

```
E_VALUE = I_VALUE.
```

```
CALL METHOD /U4A/CL_F4_HNADL=>F4_GET_SELECT01
```

```
EXPORTING
```

```
IS_F4DATA = I_FDATA
```

```
I_FLDNM = 'QUANTITY'
```

```
IMPORTING
```

```
E_VALUE = I_VALUE2.
```

```
" 선택한 INDEX 값 읽어오기
```

```
GV_FNAME = 'I_FDATA-ROWINDEX'.
```

```
ASSIGN (GV_FNAME) TO <FS_FDATA>.
```

```
CHECK <FS_FDATA> IS ASSIGNED.
```

```
" 선택한 NAME을 INPUT VALUE로 설정.
```

```
READ TABLE GT_OTAB ASSIGNING <FS_OTAB> INDEX <FS_FDATA>.
```

```
CHECK SY-SUBRC EQ 0.
```

```
<FS_OTAB>-NAME = I_VALUE.
```

```
<FS_OTAB>-QUANTITY = I_VALUE2.
```

```
ENDMETHOD.
```

7. Application 실행 후 원하는 Input Filed의 Search Help Click.

NO	NAME	QUANTITY	DATE
1		0	2022/12/25
2		0	2022/12/25
3		0	2022/12/25
4		0	2022/12/25
5		0	2022/12/25
6		0	2022/12/25
7		0	2022/12/25
8		0	2022/12/25
9		0	2022/12/25
10		0	2022/12/25

8. Search Help Dialog 화면에서 Filter 후 원하는 조회 결과 선택하고 Confirm Button Click.

Select Row : 1 SERACH_HELP_DYNAMIC EXAMPLE

Filter conditions Include initial value condition? Search condition method (AND)

Name:

Quantity:

10 rows Filter

	Name	Quantity	
<input type="checkbox"/>	1	OBJECT_01	100 2023/01/02
<input type="checkbox"/>	2	OBJECT_02	200 2023/01/02
<input type="checkbox"/>	3	OBJECT_03	300 2023/01/02
<input checked="" type="checkbox"/>	4	OBJECT_04	400 2023/01/02
<input type="checkbox"/>	5	OBJECT_05	500 2023/01/02
<input type="checkbox"/>	6	OBJECT_06	600 2023/01/02
<input type="checkbox"/>	7	OBJECT_07	700 2023/01/02
<input type="checkbox"/>	8	OBJECT_08	800 2023/01/02
<input type="checkbox"/>	9	OBJECT_09	900 2023/01/02
<input type="checkbox"/>	10	OBJECT_10	1000 2023/01/02

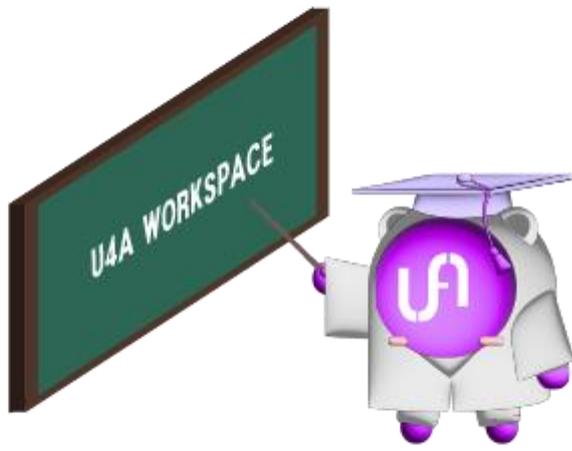
Confirm

9. Call Back Method 결과 (선택한 Row의 Input Field에 검색 조회 결과 시 선택한 Name, Quantity 값이 들어간 것을 확인)

U4A IDE HELP

NO	NAME	QUANTITY	DATE
1		0	2022/12/25
2		0	2022/12/25
3		0	2022/12/25
4	OBJECT_04	400	2022/12/25
5		0	2022/12/25
6		0	2022/12/25
7		0	2022/12/25
8		0	2022/12/25
9		0	2022/12/25
10		0	2022/12/25

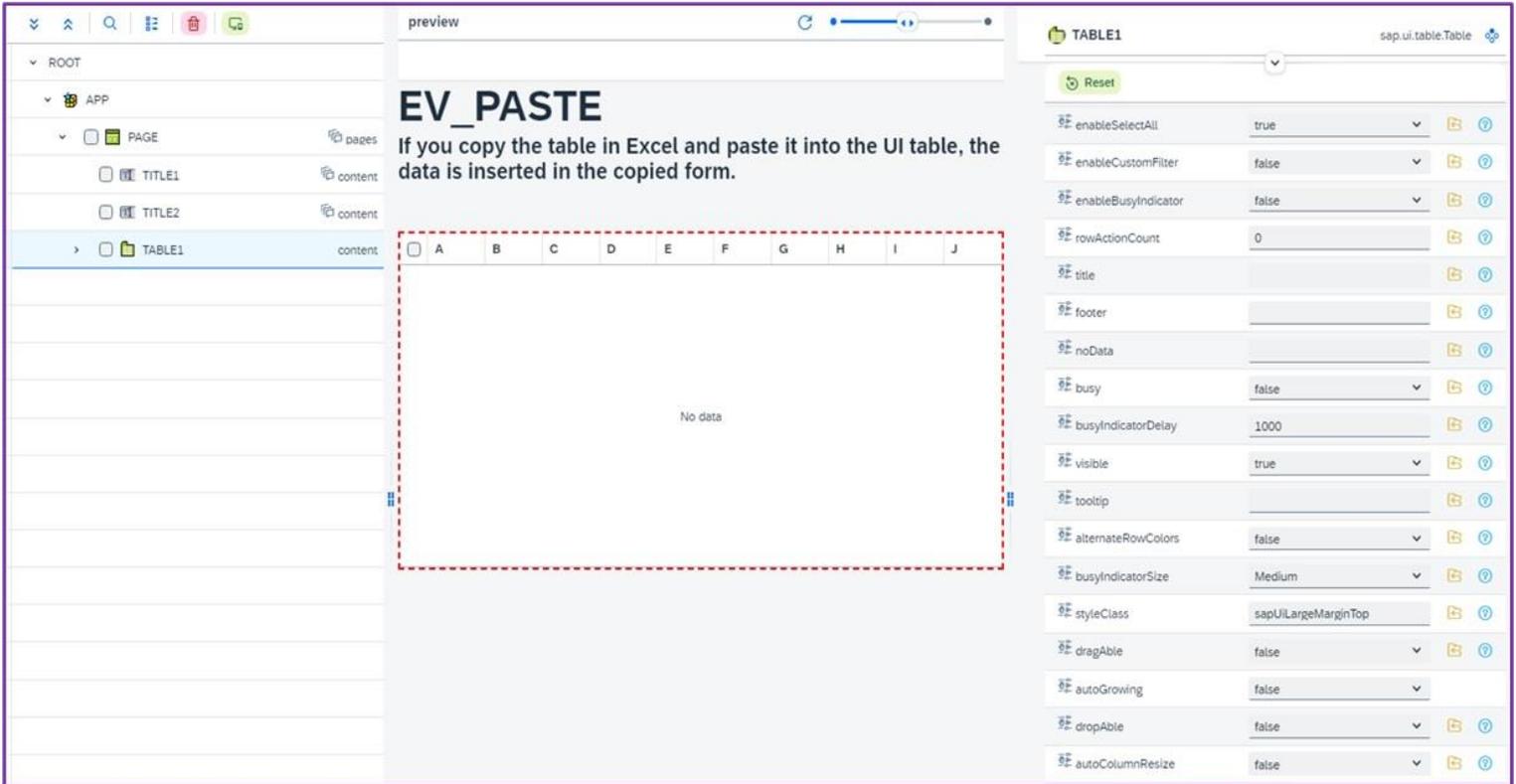
6.3 ABAP Class /U4A/CL_EXTUI_EVENTS



6.3.1 Methodes EXTUI PASTE

▣ 목적 : EXTUI_EVENTS 중 "EV_PASTE"는 외부에서 복사한 Data 를 UI Object 에 적용 시킬 수 있도록 한다.

1. 외부에서 복사할 Data와 Application에서 복사한 데이터를 적용시킬 UI Object를 생성한다.



TEMP20200702_051423.xlsx [제한된 보기] - Excel

	A	B	C	D	E	F	G	H	I	J	K	L
1	A	B	C	D	E	F	G	H	I	J		
2	TEST A001	TEST B001	TEST C001	TEST D001	TEST E001	TEST F001	TEST G001	TEST H001	TEST I001	TEST J001		
3	TEST A002	TEST B002	TEST C002	TEST D002	TEST E002	TEST F002	TEST G002	TEST H002	TEST I002	TEST J002		
4	TEST A003	TEST B003	TEST C003	TEST D003	TEST E003	TEST F003	TEST G003	TEST H003	TEST I003	TEST J003		
5	TEST A004	TEST B004	TEST C004	TEST D004	TEST E004	TEST F004	TEST G004	TEST H004	TEST I004	TEST J004		
6	TEST A005	TEST B005	TEST C005	TEST D005	TEST E005	TEST F005	TEST G005	TEST H005	TEST I005	TEST J005		
7	TEST A006	TEST B006	TEST C006	TEST D006	TEST E006	TEST F006	TEST G006	TEST H006	TEST I006	TEST J006		
8	TEST A007	TEST B007	TEST C007	TEST D007	TEST E007	TEST F007	TEST G007	TEST H007	TEST I007	TEST J007		
9	TEST A008	TEST B008	TEST C008	TEST D008	TEST E008	TEST F008	TEST G008	TEST H008	TEST I008	TEST J008		
10	TEST A009	TEST B009	TEST C009	TEST D009	TEST E009	TEST F009	TEST G009	TEST H009	TEST I009	TEST J009		
11	TEST A010	TEST B010	TEST C010	TEST D010	TEST E010	TEST F010	TEST G010	TEST H010	TEST I010	TEST J010		
12	TEST A011	TEST B011	TEST C011	TEST D011	TEST E011	TEST F011	TEST G011	TEST H011	TEST I011	TEST J011		
13	TEST A012	TEST B012	TEST C012	TEST D012	TEST E012	TEST F012	TEST G012	TEST H012	TEST I012	TEST J012		

2. ON_INIT에서 EXTUI_EVENTS중 "EV_PASTE"를 호출하고, 붙여넣기 대상이 되는 "UI Instance ID"와 "Server Event" Method를 입력한다.

[첨부 ABAP SOURCE (/U4A/IF_SERVER~HANDL_ON_INIT)]

METHOD /U4A/IF_SERVER~HANDL_ON_INIT.

"EXTUI_EVENTS 중 EV_PASTE는 복사한 TABLE을 UI TABLE에 붙여넣기 할 수 있는 Methode

CALL METHOD /U4A/CL_EXTUI_EVENTS=>EV_PASTE

EXPORTING

IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class

I_UID = 'TABLE1' " 대상 UI 인스턴스 ID (예:워크벤치에 디

자인영역에 ID정보)

I_SERVER_EVENT = 'EV_PASTE1'.

" 서버 이벤트 ID(예:워크벤치에서 생성

된 서버이벤트(컨트롤 메소드명)

ENDMETHOD.

Method	Level	Visibility
/U4A/IF_SERVER~HANDL_ON_INIT	Instance Method	Public
/U4A/IF_SERVER~HANDL_ON_REQUEST	Instance Method	Public
/U4A/IF_SERVER~HANDL_ON_RESPONSE	Instance Method	Public
/U4A/IF_SERVER~HANDL_ON_EXIT	Instance Method	Public
/U4A/IF_SERVER~HANDL_ON_MESSAGE	Instance Method	Public
/U4A/IF_SERVER~HANDL_ON_HASHCHANGE	Instance Method	Public
CONSTRUCTOR	Instance Method	Public
EV_PASTE1	Instance Method	Public

3. Table 구조에서 붙여넣기 한 Data 확인 후 열로 들어온 데이터를 행단위로, 행단위로 만든 데이터를 Colum 단위로 변경 후 UI TABLE에 담아준다.

[첨부 ABAP SOURCE (EV_PASTE1)]

```
METHOD EV_PASTE1.
```

```
DATA : LS_FORM TYPE IHTTPNVP,
        LT_TAB  TYPE STRING_TABLE,
        LS_TAB  TYPE STRING,
        LT_STRU TYPE STRING_TABLE,
        LS_STRU TYPE STRING,
        LS_OTAB TYPE TY_TAB.
```

```
FIELD-SYMBOLS <FS_FLD> TYPE ANY.
```

```
DEFINE _SET_MSG.
```

```
CALL METHOD /U4A/CL_UTILITIES=>M_MESSAGEBOX
```

```
EXPORTING
```

```
IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class
```

```
I_MSGTX = &1. " MESSAGE TEXT
```

```
END-OF-DEFINITION.
```

```
*-----*
```

*복사한 TABLE구조는 IT_FORM_DATA에 담기고 'CPASTED|TABLE1'기준으로 data를 조회한다.

```
READ TABLE IT_FORM_DATA INTO LS_FORM
```

```
WITH KEY NAME = 'CPASTED|TABLE1'.
```

*붙여넣기 data 확인

```
IF SY-SUBRC NE 0.
```

```
_SET_MSG '붙여넣기 data가 존재하지 않습니다.'
```

```
EXIT.
```

```
ENDIF.
```

*NEWLINE을 통해 LS_FORM의 하나의 열로 받은 data를 행단위로 나눈다.

```
SPLIT LS_FORM-VALUE AT CL_ABAP_CHAR_UTILITIES=>NEWLINE INTO TABLE LT_TAB.
```

*NEWLINE을 통해 정상적으로 TABLE이 생성되었는지 확인.

```
IF LT_TAB[] IS INITIAL.
```

```
  _SET_MSG '잘못된 형식의 붙여넣기 정보 입니다.'
```

```
  EXIT.
```

```
ENDIF.
```

*기존 UI TABLE있던 data 초기화

```
REFRESH GT_TAB.
```

*STRING TABLE의 data를 UI TABLE의 형태로 변경

```
LOOP AT LT_TAB INTO LS_TAB.
```

*HORIZONTAL_TAB을 통해 LS_TAB의 열 data를 행으로 변경

```
SPLIT LS_TAB AT CL_ABAP_CHAR_UTILITIES=>HORIZONTAL_TAB INTO TABLE LT_STRU.
```

*LT_STRU에 담긴 행단위 data를 COLUMN 단위로 분할

```
LOOP AT LT_STRU INTO LS_STRU.
```

```
  UNASSIGN <FS_FLD>.
```

```
  ASSIGN COMPONENT SY-TABIX OF STRUCTURE LS_OTAB TO <FS_FLD>.
```

```
  CHECK <FS_FLD> IS ASSIGNED.
```

```
  <FS_FLD> = LS_STRU.
```

```
ENDLOOP.
```

*UI TABLE에 COLUMN 단위 data를 전달.

```
APPEND LS_OTAB TO GT_TAB.
```

```
CLEAR LS_OTAB.
```

```
ENDLOOP.
```

ENDMETHOD.

4. Method를 구성 후 Excel 에서 복사한 Data를 UI TABLE에 붙여넣기를 하면 Table 형태로 담기게 된다.

EV_PASTE
If you copy the table in Excel and paste it into the UI table, the data is inserted in the copied form.

	A	B	C	D	E	F	G	H	I	J
1	A	B	C	D	E	F	G	H	I	J
2	TEST A001	TEST B001	TEST C001	TEST D001	TEST E001	TEST F001	TEST G001	TEST H001	TEST I001	TEST J001
3	TEST A002	TEST B002	TEST C002	TEST D002	TEST E002	TEST F002	TEST G002	TEST H002	TEST I002	TEST J002
4	TEST A003	TEST B003	TEST C003	TEST D003	TEST E003	TEST F003	TEST G003	TEST H003	TEST I003	TEST J003
5	TEST A004	TEST B004	TEST C004	TEST D004	TEST E004	TEST F004	TEST G004	TEST H004	TEST I004	TEST J004
6	TEST A005	TEST B005	TEST C005	TEST D005	TEST E005	TEST F005	TEST G005	TEST H005	TEST I005	TEST J005
7	TEST A006	TEST B006	TEST C006	TEST D006	TEST E006	TEST F006	TEST G006	TEST H006	TEST I006	TEST J006
8	TEST A007	TEST B007	TEST C007	TEST D007	TEST E007	TEST F007	TEST G007	TEST H007	TEST I007	TEST J007
9	TEST A008	TEST B008	TEST C008	TEST D008	TEST E008	TEST F008	TEST G008	TEST H008	TEST I008	TEST J008
10	TEST A009	TEST B009	TEST C009	TEST D009	TEST E009	TEST F009	TEST G009	TEST H009	TEST I009	TEST J009
11	TEST A010	TEST B010	TEST C010	TEST D010	TEST E010	TEST F010	TEST G010	TEST H010	TEST I010	TEST J010
12	TEST A011	TEST B011	TEST C011	TEST D011	TEST E011	TEST F011	TEST G011	TEST H011	TEST I011	TEST J011
13	TEST A012	TEST B012	TEST C012	TEST D012	TEST E012	TEST F012	TEST G012	TEST H012	TEST I012	TEST J012
14	TEST A013	TEST B013	TEST C013	TEST D013	TEST E013	TEST F013	TEST G013	TEST H013	TEST I013	TEST J013
15	TEST A014	TEST B014	TEST C014	TEST D014	TEST E014	TEST F014	TEST G014	TEST H014	TEST I014	TEST J014
16	TEST A015	TEST B015	TEST C015	TEST D015	TEST E015	TEST F015	TEST G015	TEST H015	TEST I015	TEST J015
17	TEST A016	TEST B016	TEST C016	TEST D016	TEST E016	TEST F016	TEST G016	TEST H016	TEST I016	TEST J016
18	TEST A017	TEST B017	TEST C017	TEST D017	TEST E017	TEST F017	TEST G017	TEST H017	TEST I017	TEST J017
19	TEST A018	TEST B018	TEST C018	TEST D018	TEST E018	TEST F018	TEST G018	TEST H018	TEST I018	TEST J018

No data

EV_PASTE
If you copy the table in Excel and paste it into the UI table, the data is inserted in the copied form.

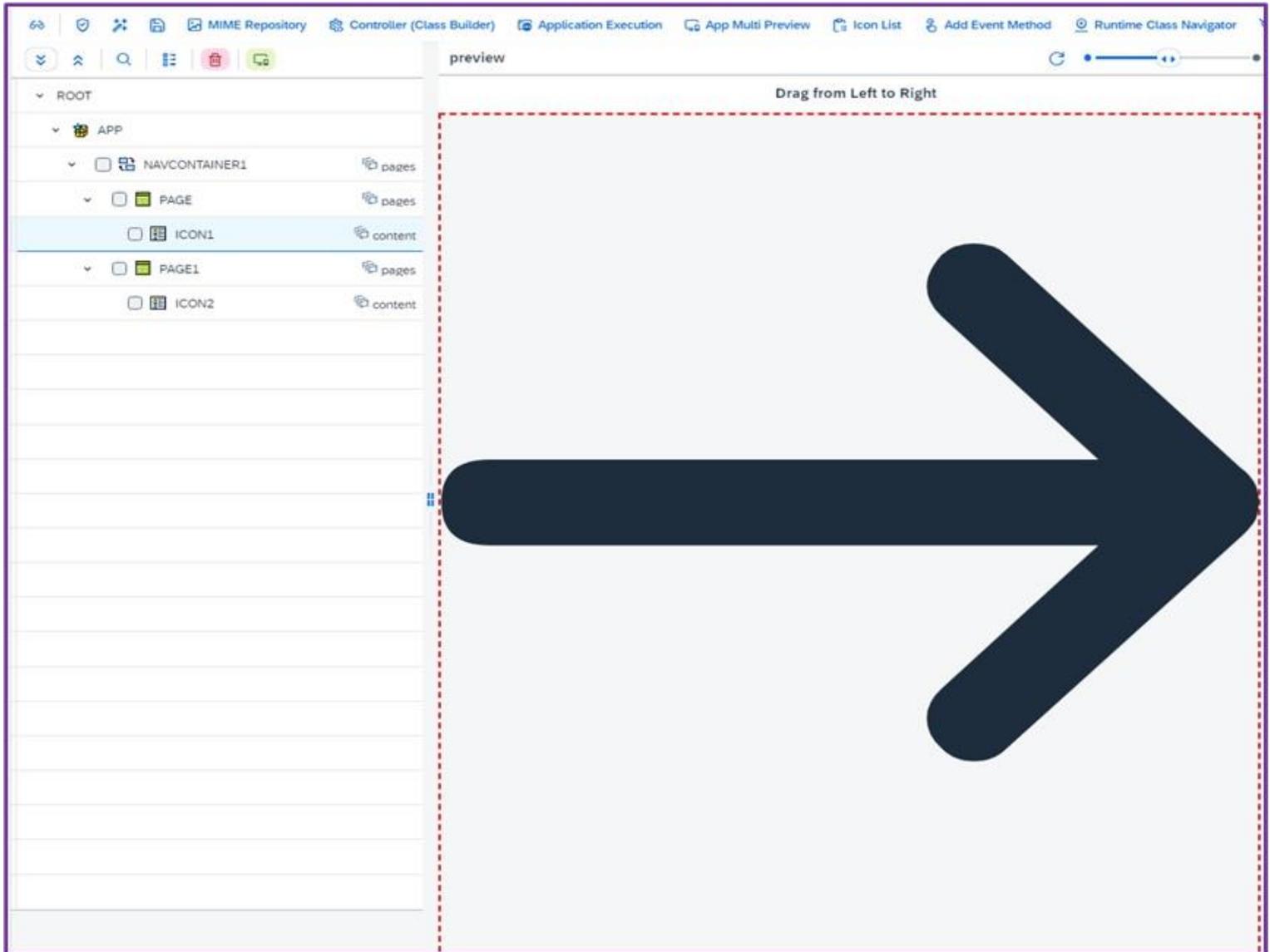
	A	B	C	D	E	F	G	H
	A	B	C	D	E	F		
	TEST A001	TEST B001	TEST C001	TEST D001	TEST E001	TEST F001		
	TEST A002	TEST B002	TEST C002	TEST D002	TEST E002	TEST F002		
	TEST A003	TEST B003	TEST C003	TEST D003	TEST E003	TEST F003		
	TEST A004	TEST B004	TEST C004	TEST D004	TEST E004	TEST F004		
	TEST A005	TEST B005	TEST C005	TEST D005	TEST E005	TEST F005		
	TEST A006	TEST B006	TEST C006	TEST D006	TEST E006	TEST F006		
	TEST A007	TEST B007	TEST C007	TEST D007	TEST E007	TEST F007		
	TEST A008	TEST B008	TEST C008	TEST D008	TEST E008	TEST F008		
	TEST A009	TEST B009	TEST C009	TEST D009	TEST E009	TEST F009		

1. 현재 해당 Method는 Chrome에서 사용 가능하다(explorer는 사용 불가능)
2. Event를 생성할 때 반드시 Add Event Method를 이용해 만들도록 한다.

6.3.2 Methodes EXTUI SWIPE

▣ 목적 : EXTUI_EVENTS 중 "EV_SWIPE"는 Web 화면에 마우스로 drag시 Event를 작동시킬 수 있다.

1. Application을 만들고 UI Object를 생성한다.



2. ON_INIT에서 EXTUI의 EV_SWIPE를 생성한다.

[첨부ABAP SOURCE (/U4A/IF_SERVER~HANDL_ON_INIT)]

METHOD /U4A/IF_SERVER~HANDL_ON_INIT.

"페이지 클릭 후 drag 했을 때 EVENT를 작동시키는 METHOD

CALL METHOD /U4A/CL_EXTUI_EVENTS=>EV_SWIPE

EXPORTING

IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW

" [U4A] UI Element Super Class

I_UID = 'PAGE'

" 대상 UI 인스턴스 ID (예:워크벤치에 디자인영역에 ID정보)

I_MODE = 'R'

" swipe:S,swipeleft:L,swiperight:R

I_SERVER_EVENT = 'EV_MOVE_PAGE'.

" 서버 이벤트 ID(예:워크벤치에서 생성된 서버이벤트(컨트롤 메소드명))

된 서버이벤트(컨트롤 메소드명)

* IT_SCRIPT_SOURCE = " javascript source탭

* I_USE_WAIT = 'X' " Waiting 사용여부('X' = 사용함, ' ' = 사용안함)

* EXCEPTIONS

* MODE_NULL = 1

* NOT_SUPPORT_ELEMENT = 2

* NOT_FOUND_UI = 3

* OTHERS = 4

"페이지 클릭 후 drag 했을 때 EVENT를 작동시키는 METHOD

CALL METHOD /U4A/CL_EXTUI_EVENTS=>EV_SWIPE

EXPORTING

IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW

" [U4A] UI Element Super Class

I_UID = 'PAGE1'

" 대상 UI 인스턴스 ID (예:워크벤치에 디자인영역에 ID정보)

인영역에 ID정보)

I_MODE = 'L'

" swipe:S,swipeleft:L,swiperight:R

I_SERVER_EVENT = 'EV_MOVE_PAGE'.

" 서버 이벤트 ID(예:워크벤치에서 생성된 서버이벤트(컨트롤 메소드명))

된 서버이벤트(컨트롤 메소드명)

* IT_SCRIPT_SOURCE = " javascript source탭

* I_USE_WAIT = 'X' " Waiting 사용여부('X' = 사용함, ' ' = 사용안함)

* EXCEPTIONS

* MODE_NULL = 1

* NOT_SUPPORT_ELEMENT = 2

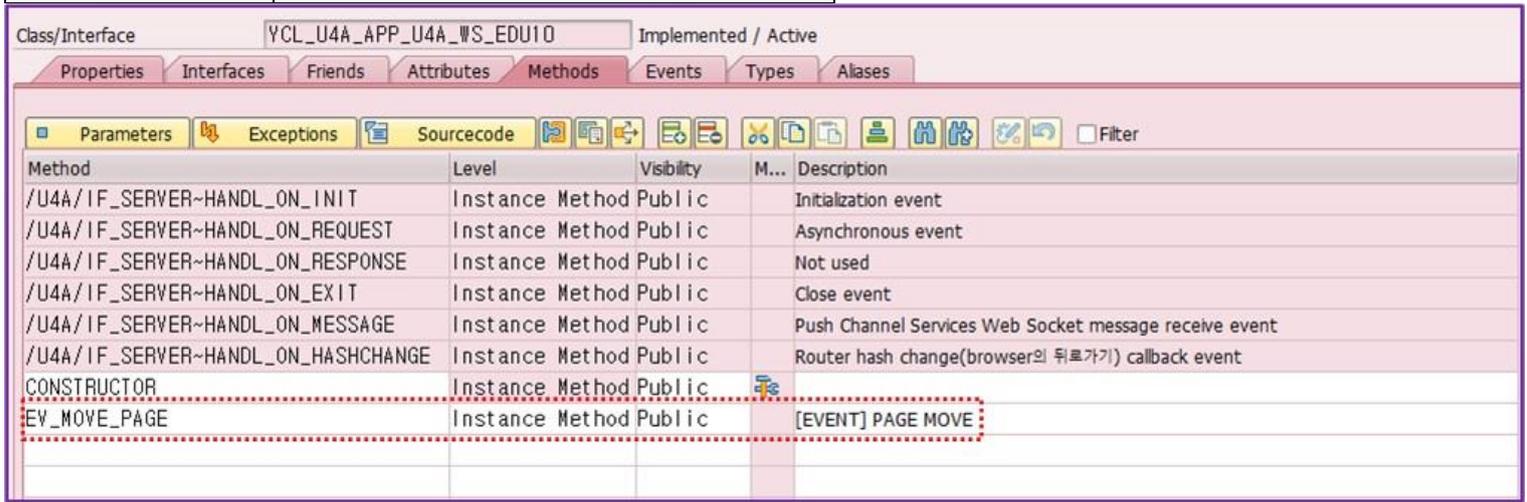
* NOT_FOUND_UI = 3

* OTHERS = 4

ENDMETHOD.

EXPORTING	Means
-----------	-------

I_UID	Method를 적용시키려는 UI Object의 ID
I_MODE	S : 왼쪽에서 오른쪽, 오른쪽에서 왼쪽 Drag 시 Event 작동 L : 오른쪽에서 왼쪽 Drag 시 Event 작동 R : 왼쪽에서 오른쪽 Drag 시 Event 작동
I_SERVER_EVENT	Drag시 발생하는 Event
IT_SCRIPT_SOURCE	Drag시 작동시키는 SCRIPT SOURCE



3. SWIPE시 작동하는 Method를 구성한다.

[첨부 ABAP SOURCE (EV_MOVE_PAGE)]

```
METHOD EV_MOVE_PAGE.
```

```
DATA : LV_PAGE_ID TYPE STRING.
```

-----*

```
CASE I_ID.
```

```
WHEN 'PAGE'.
```

```
"ID가 PAGE이면 PAGE1로 이동
```

```
LV_PAGE_ID = 'PAGE1'.
```

```
WHEN 'PAGE1'.
```

```
"ID가 PAGE1이면 PAGE로 이동
LV_PAGE_ID = 'PAGE'.
```

```
ENDCASE.
```

```
"PAGE MOVE.
```

```
CALL METHOD /U4A/CL_UTILITIES=>PAGE_NAVIGATION
```

```
EXPORTING
```

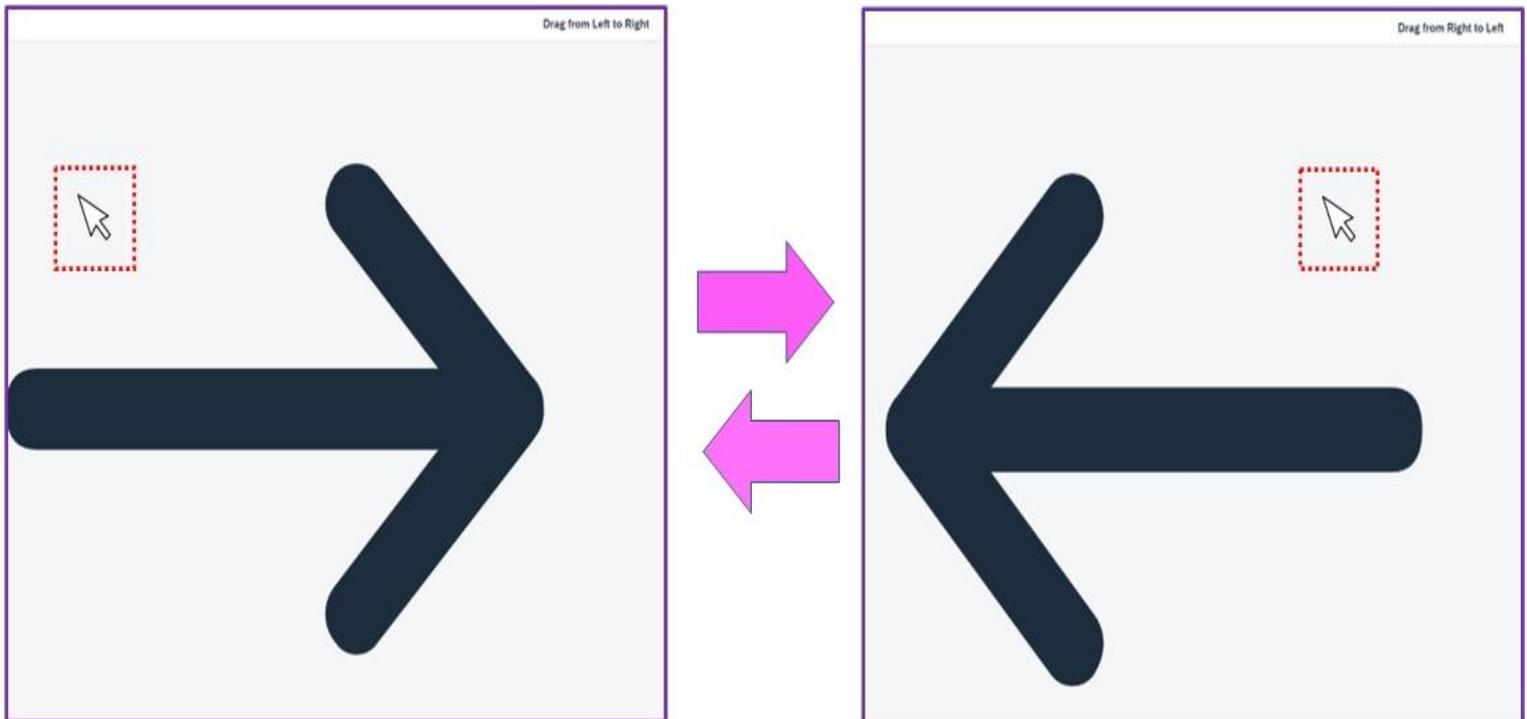
```
IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class
```

```
I_APPID = 'NAVCONTAINER1'
```

```
I_PAGEID = LV_PAGE_ID.
```

```
ENDMETHOD.
```

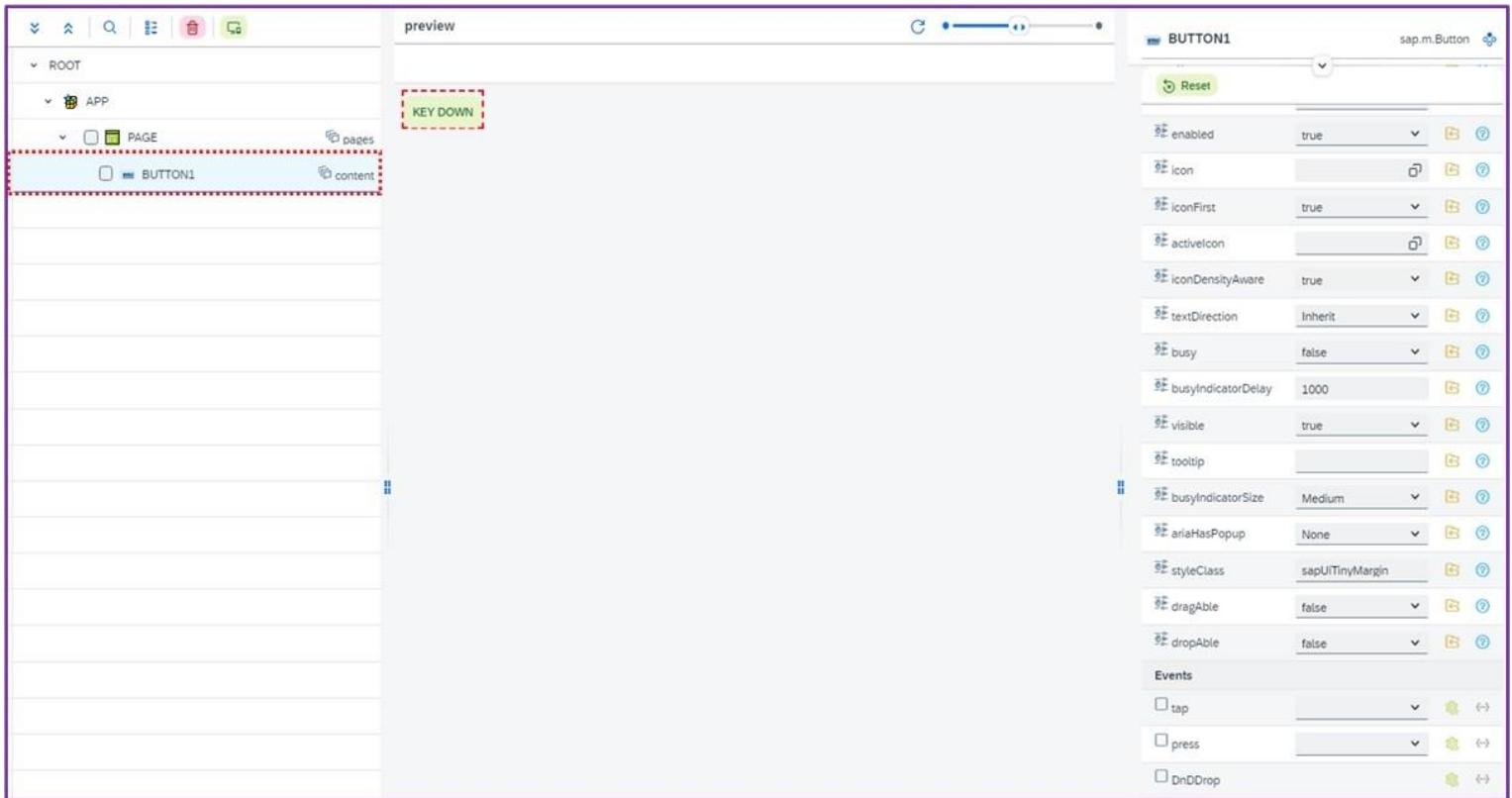
4. Web에서 자신이 설정한 SWIPE에 따른 Event가 작동하는지 확인한다.



6.3.3 Methodes EXTUI KEYDOWN

- 목적 : EXTUI_EVENTS 중 "EV_KEYDOWN"은 Keyboard 입력 Event가 없는 UI Object에 Keyboard 값을 입력하면 Server Event가 작동한다.

1. Application에서 UI Object를 생성한다.



2. ON_INIT에서 EXTUI의 KEYDOWN을 입력한다.

[첨부ABAP SOURCE (/U4A/IF_SERVER~HANDL_ON_INIT)]

METHOD /U4A/IF_SERVER~HANDL_ON_INIT.

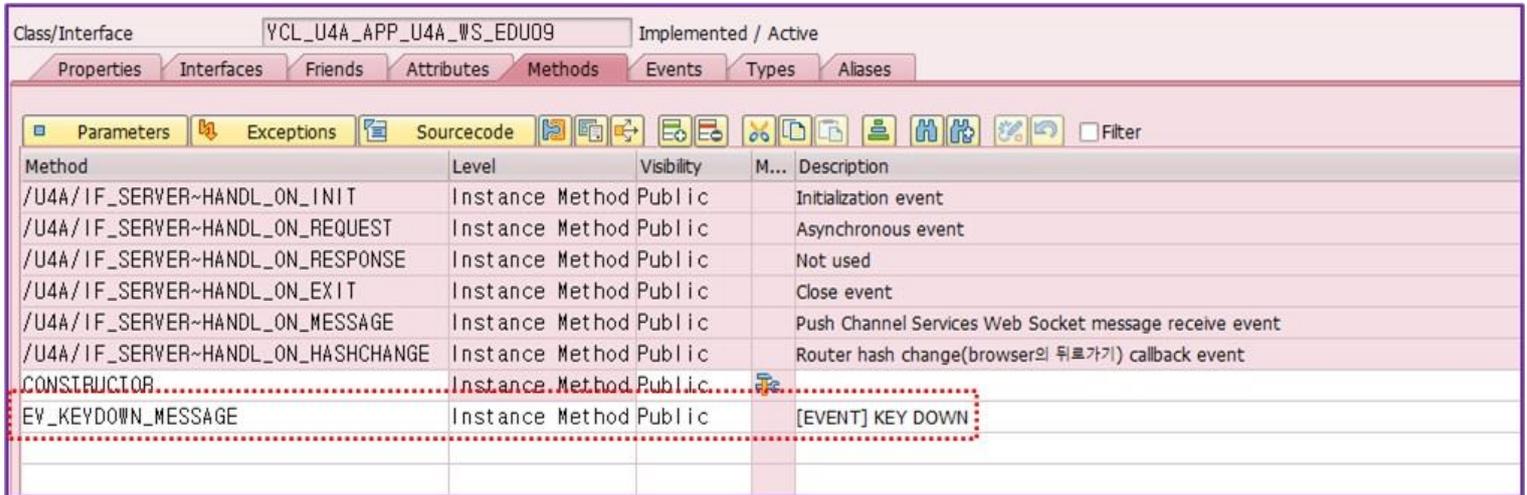
"INPUT1에 KEYBOARD값이 들어오면 SERVER EVENT 발생.

```
CALL METHOD /U4A/CL_EXTUI_EVENTS=>EV_KEYDOWN
EXPORTING
```

```

IO_VIEW          = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class
I_UID            = 'BUTTON1'                " UI Object ID
I_SERVER_EVENT  = 'EV_KEYDOWN_MESSAGE'.    " Server Event
* IT_SCRIPT_SOURCE =                        " Client Event
* I_USE_WAIT     = 'X'                      " Waiting 사용여부('X' = 사용함, ' ' = 사용안함)
* I_USE_PROPAGATION = 'X'                  " 이벤트 전파방지 사용여부('X' = 사용함)
* EXCEPTIONS
* NOT_SUPPORT_ELEMENT = 1                  " attachBrowserEvent not supported UI.
* NOT_FOUND_UI       = 2                  " UI Instance not found.
* OTHERS             = 3
    
```

ENDMETHOD.



3. KEYDOWN 입력 시 MESSAGE가 나타나는 Method를 만든다.

[첨부ABAP SOURCE (EV_KEYDOWN_MESSAGE)]

METHOD EV_KEYDOWN_MESSAGE.

DATA LV_MSG TYPE STRING.

"MESSAGE 내용

LV_MSG = 'Keydown is Worked!!!'.

"MESSAGE 창 띄우기

```
CALL METHOD /U4A/CL_UTILITIES=>M_MESSAGE_TOAST
```

```
EXPORTING
```

```
IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class
```

```
* IS_MSG = " Structure of message variables
```

```
I_MSGTX = LV_MSG. " MESSAGE TEXT
```

```
* IT_MSGTX = " Table of Strings
```

```
* I_VPOS = CS_M_MSG_TOS_VPOS-CENTER " Vertical position
```

```
* I_HPOS = CS_M_MSG_TOS_HPOS-CENTER " Horizontal position
```

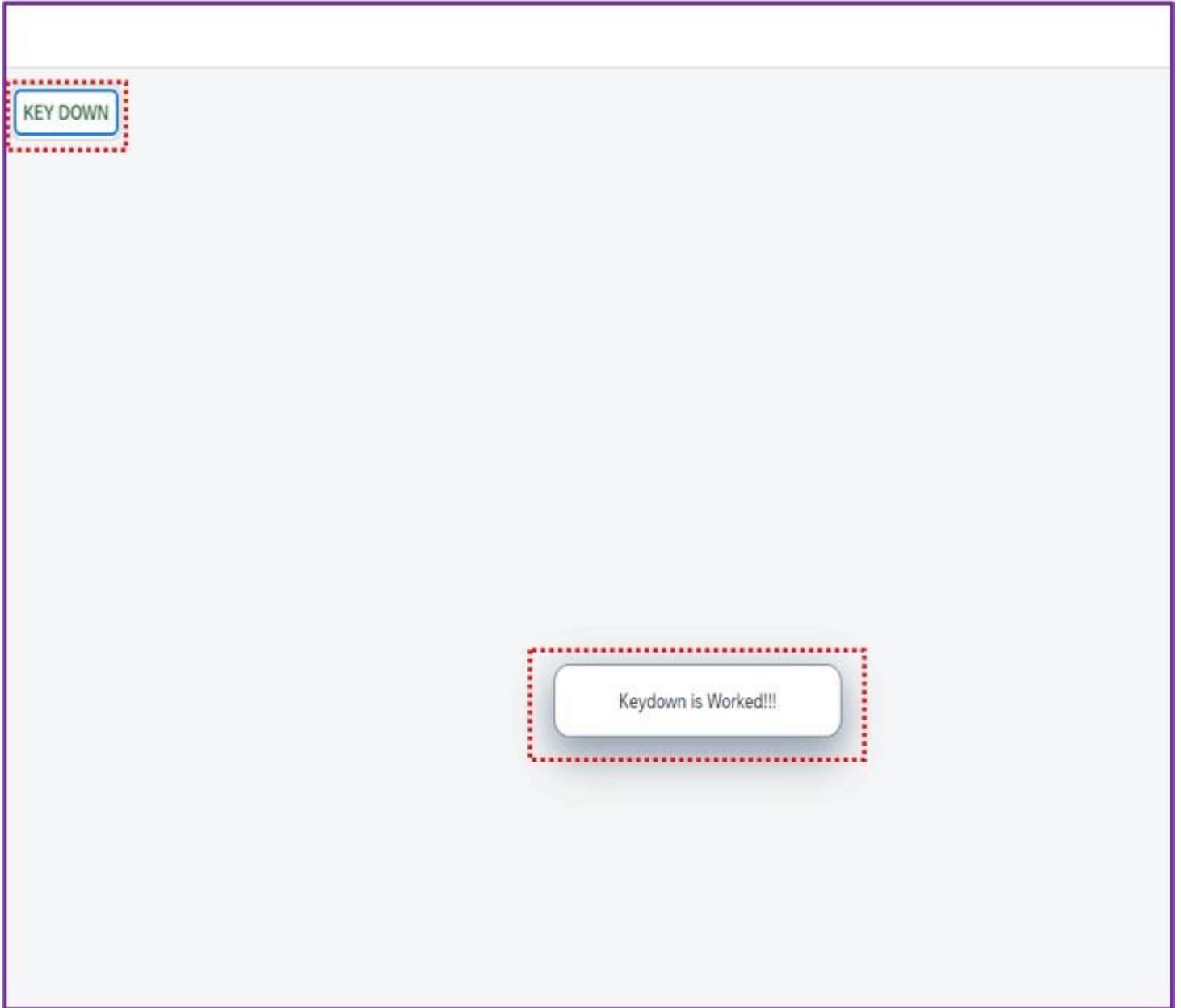
```
* I_VOFF = '0' " Vertical offset
```

```
* I_HOFF = '0' " Horizontal offset
```

```
* I_DURATION = 10000
```

```
ENDMETHOD.
```

4. BUTTON이 KEYDOWN을 통해 결과가 나타나는것을 확인 할 수 있다.

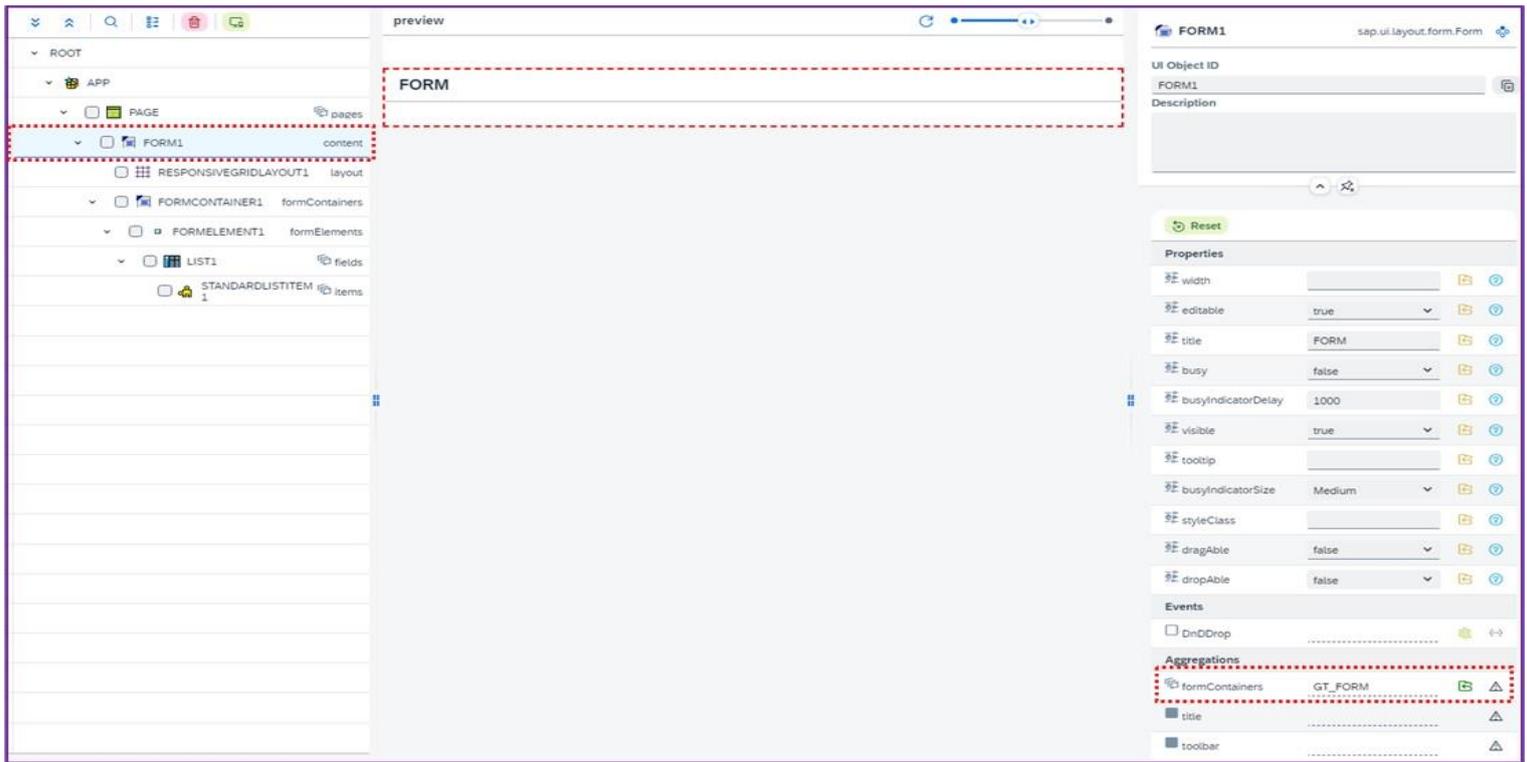


6.3.4 Methodes EXTUI CLICK LAYOUT

▣ 목적 : EXTUI_EVENTS 중 "EV_CLICK_UI_LAYOUT_FORM"은 FORM CLICK 시 Server Event 가 작동한다.

※ Form UI는 Click에 관련된 이벤트가 없기에 Extend로 이벤트를 추가하고자함.

1. Application에서 Form UI Object를 생성한다.



2. ON_INIT에서 EXTUI의 CLICK_UI_LAYOUT_FORM과 FORM에 Binding할 TABLE을 입력한다.

[첨부ABAP SOURCE (/U4A/IF_SERVER~HANDL_ON_INIT)]

METHOD /U4A/IF_SERVER~HANDL_ON_INIT.

DATA LS_FORM TYPE TY_FORM.

DATA LS_LIST TYPE TY_LIST.

```
*-----*
```

```
"FORM 바인딩 데이터 구성
```

```
LS_FORM-H1 = 'U4A WS'.
```

```
LS_LIST-F1 = 'ONE'.
```

```
LS_LIST-F2 = '1'.
```

```
APPEND LS_LIST TO LS_FORM-LIST.
```

```
CLEAR LS_LIST.
```

```
LS_LIST-F1 = 'TWO'.
```

```
LS_LIST-F2 = '2'.
```

```
APPEND LS_LIST TO LS_FORM-LIST.
```

```
CLEAR LS_LIST.
```

```
LS_LIST-F1 = 'THREE'.
```

```
LS_LIST-F2 = '3'.
```

```
APPEND LS_LIST TO LS_FORM-LIST.
```

```
CLEAR LS_LIST.
```

```
APPEND LS_FORM TO GT_FORM.
```

```
CLEAR LS_FORM.
```

```
LS_FORM-H1 = 'U4A WS2'.
```

```
APPEND LS_FORM TO GT_FORM.
```

```
CLEAR LS_FORM.
```

```
"FORM1 UI CLICK => SERVER EVENT 실행.
```

```
CALL METHOD /U4A/CL_EXTUI_EVENTS=>EV_CLICK_UI_LAYOUT_FORM
```

```
EXPORTING
```

```
IO_VIEW          = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class
```

```
I_UID            = 'FORM1' " UI ID
```

```
* I_ISBIND       = " !!폐기!! BIND 일 경우만 서버이벤트 수행 여부
```

```
(예: X, ALL:SPACE)
```

```
I_SERVER_EVENT  = 'EV_CLICK_LAYOUT'. " 서버이벤트명
```

```
* IT_SCRIPT_SOURCE = " Table of Strings
```

```
* I_USE_WAIT     = 'X' " Waiting 사용여부('X' = 사용함, ' ' = 사용안함)
```

```
* I_USE_PROPAGATION = 'X' " 이벤트 전파방지 사용여부('X' = 사용함)
```

```
* EXCEPTIONS
* NOT_SUPPORT_ELEMENT = 1
* OTHERS = 2
```

ENDMETHOD.

Method	Level	Visibility	M...	Description
/U4A/IF_SERVER~HANDL_ON_INIT	Instance Method	Public		Initialization event
/U4A/IF_SERVER~HANDL_ON_REQUEST	Instance Method	Public		Asynchronous event
/U4A/IF_SERVER~HANDL_ON_RESPONSE	Instance Method	Public		Not used
/U4A/IF_SERVER~HANDL_ON_EXIT	Instance Method	Public		Close event
/U4A/IF_SERVER~HANDL_ON_MESSAGE	Instance Method	Public		Push Channel Services Web Socket message receive event
/U4A/IF_SERVER~HANDL_ON_HASHCHANGE	Instance Method	Public		Router hash change(browser의 뒤로가기) callback event
CONSTRUCTOR	Instance Method	Public		
EV_CLICK_LAYOUT	Instance Method	Public		[EVENT] CLICK LAYOUT MESSAGE

3. LAYOUT CLICK 시 MESSAGE가 나타나는 Method를 만든다.

[첨부ABAP SOURCE (EV_CLICK_LAYOUT)]

```
METHOD EV_CLICK_LAYOUT.
```

```
DATA LV_MSG TYPE STRING.
```

```
*-----*
```

```
"MESSAGE 내역
```

```
LV_MSG = 'FORM1을 CLICK 하였습니다!!!'.
```

```
"MESSAGE 창 호출
```

```
CALL METHOD /U4A/CL_UTILITIES=>M_MESSAGE_TOAST
```

```
EXPORTING
```

```
IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class
```

```
* IS_MSG = " Structure of message variables
```

```
I_MSGTX = LV_MSG. " MESSAGE TEXT
```

```
* IT_MSGTX = " Table of Strings
```

```
* I_VPOS = CS_M_MSG_TOS_VPOS-CENTER " Vertical position
```

```
* I_HPOS = CS_M_MSG_TOS_HPOS-CENTER " Horizontal position
```

```
*      I_VOFF      = '0'          " Vertical offset
*      I_HOFF      = '0'          " Horizontal offset
*      I_DURATION  = 10000
```

ENDMETHOD.

4. FORM을 클릭하면 MESSAGE가 나타나는것을 확인 할 수 있다.

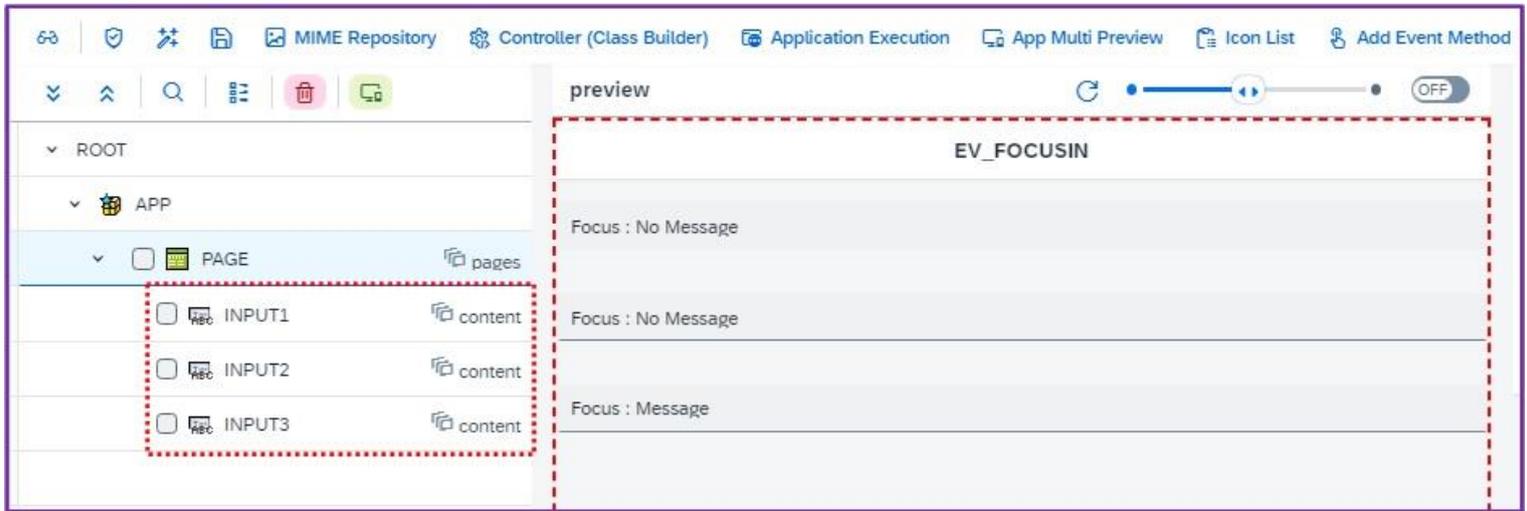
The screenshot shows a graphical user interface for a U4A IDE. At the top, there's a title bar labeled 'FORM'. Below it, there are two data tables. The first table is under the heading 'U4A WS' and has two columns: 'ONE' and 'TWO'. The first row contains the values '1' and '2'. The second table is under the heading 'U4A WS2' and has one column: 'THREE', with the value '3'. At the bottom center of the window, there is a message box with a red border and a white background, containing the text 'FORM1을 CLICK 하였습니다!!!'.

-  1. EV_CLICK_UI_LAYOUT_FORM중 EXPORTING의 I_ISBIND는 현재 폐기 상태로 사용하지 않는다.
-  2. FORM과 하위 항목은 반드시 Binding을 해야 SERVER_EVENT가 작동한다.

6.3.5 Methodes EXTUI FOCUSIN

▣ 목적 : EXTUI_EVENTS 중 "EV_FOCUSIN"은 특정 UI Focus 시 Event 를 설정할 수 있다.

1. Application에서 특정 UI Focus시 Event를 수행할 UI Object 생성(본 단원에서는 "INPUT3")



2. ON_INIT에서 EXTUI_EVENTS의 " EV_FOCUSIN"을 설정한다.

[첨부ABAP SOURCE (/U4A/IF_SERVER~HANDL_ON_INIT)]

```
METHOD /U4A/IF_SERVER~HANDL_ON_INIT .
```

```
"INPUT3 UI Focus시 이벤트 Attach.
```

```
CALL METHOD /U4A/CL_EXTUI_EVENTS=>EV_FOCUSIN
```

```
EXPORTING
```

```
IO_VIEW          = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class
```

```
I_UID            = 'INPUT3' " UI object ID
```

```
I_SERVER_EVENT  = 'EV_FOCUSIN' " Server Event
```

```
* IT_SCRIPT_SOURCE = " Client Event
```

```
* I_USE_WAIT       = " Waiting 사용여부('X' = 사용함, ' ' = 사용안함)
```

```
* I_USE_PROPAGATION = " 이벤트 전파방지 사용여부('X' = 사용함)
* EXCEPTIONS
* NOT_SUPPORT_ELEMENT = 1 " attachBrowserEvent not supported UI.
* NOT_FOUND_UI = 2 " UI Instance not found.
* OTHERS = 3
```

ENDMETHOD.

3. UI가 "FOCUS" 되었을시 MESSAGE를 호출하는 Method를 만든다.

Method	Level	Visibility	M...	Description
/U4A/IF_SERVER~HANDL_ON_INIT	Instance Method	Public		Initialization event
/U4A/IF_SERVER~HANDL_ON_REQUEST	Instance Method	Public		Asynchronous event
/U4A/IF_SERVER~HANDL_ON_RESPONSE	Instance Method	Public		Not used
/U4A/IF_SERVER~HANDL_ON_EXIT	Instance Method	Public		Close event
/U4A/IF_SERVER~HANDL_ON_MESSAGE	Instance Method	Public		Push Channel Services Web Socket message receive event
/U4A/IF_SERVER~HANDL_ON_HASHCHANGE	Instance Method	Public		Router hash change(browser의 뒤로가기) callback event
CONSTRUCTOR	Instance Method	Public		
EV_FOCUSIN	Instance Method	Public		[EVENT] FOCUS

[첨부ABAP SOURCE (EV_FOCUSIN)]

METHOD EV_FOCUSIN.

"메세지 호출

CALL METHOD /U4A/CL_UTILITIES=>M_MESSAGE_TOAST

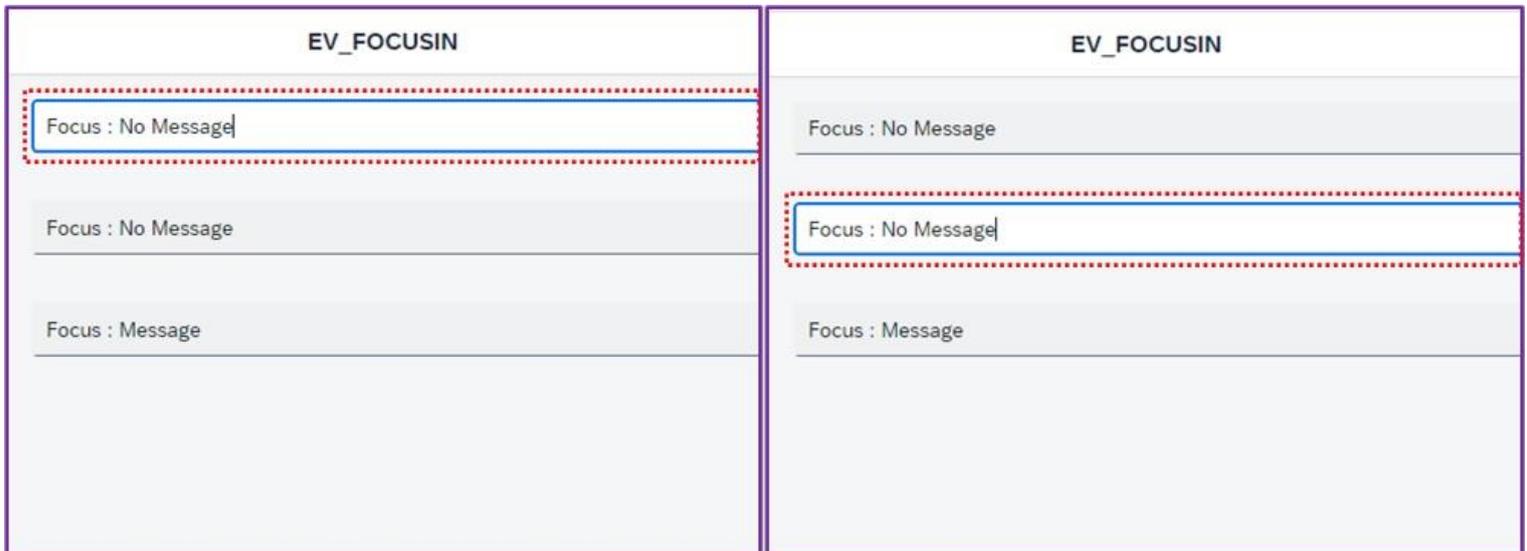
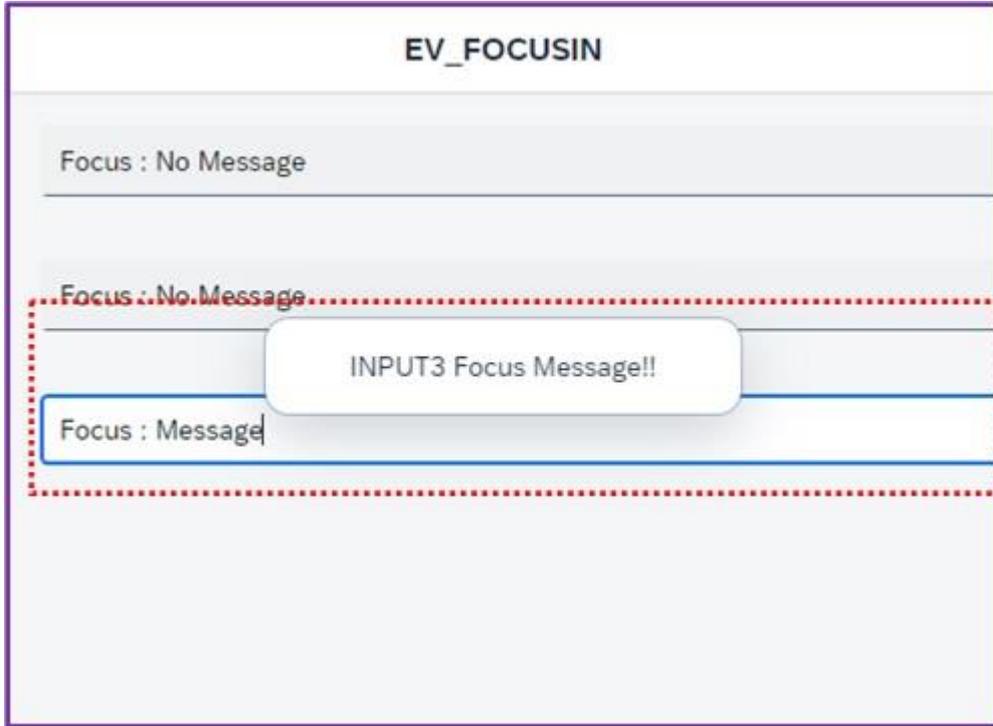
EXPORTING

IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class

I_MSGTX = 'INPUT3 Focus Message!!'. " MESSAGE TEXT

ENDMETHOD.

4. Application 실행 후 "Focus : Message" 영역 INPUT UI에 Focus시 Message 호출 (그외 INPUT UI Focus시 Message 호출 불가)

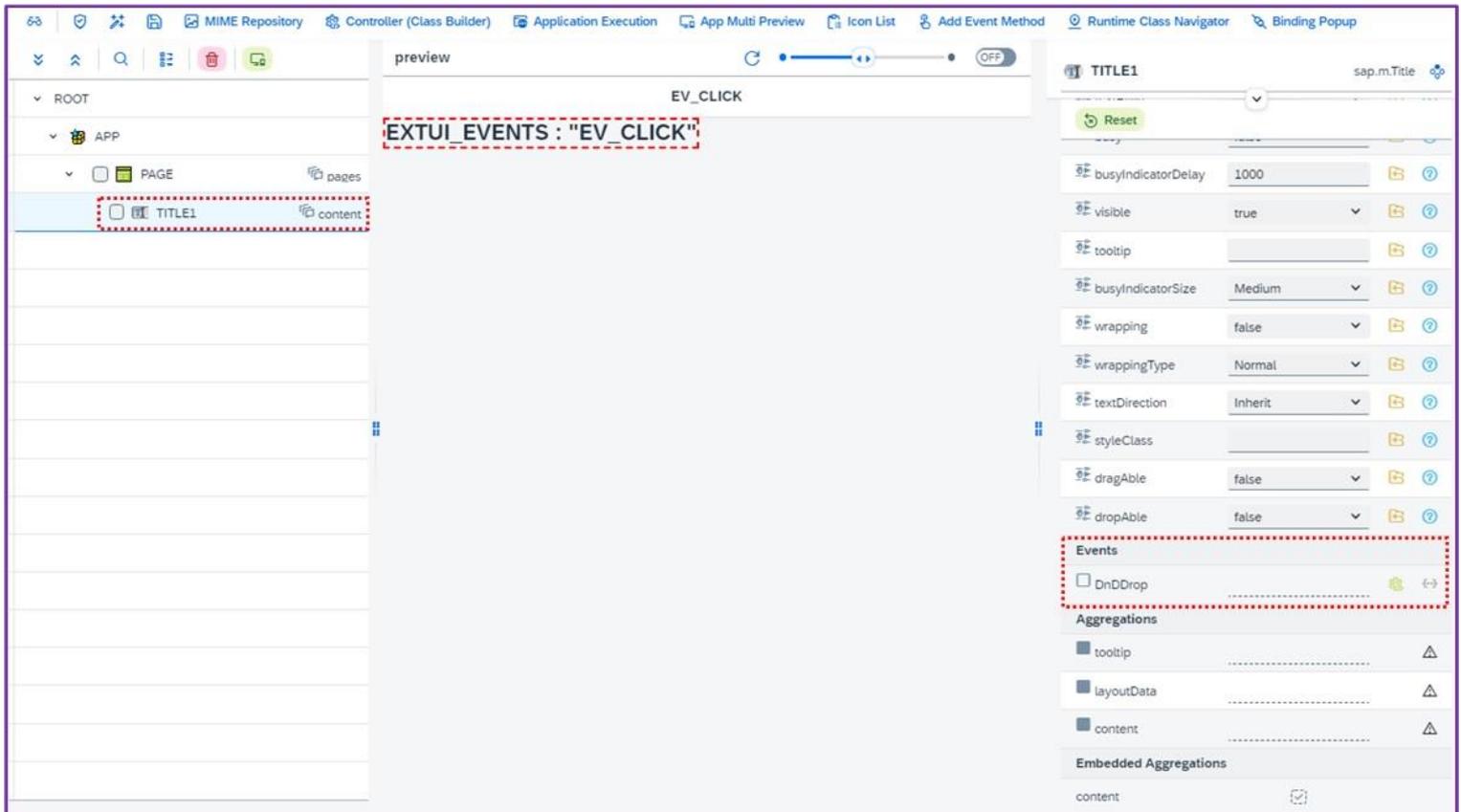


< Figure : FOCUSIN Event 미설정 INPUT UI => No Message >

6.3.6 Methodes EXTUI CLICK

▣ 목적 : EXTUI_EVENTS 중 "EV_CLICK"은 Press Event가 없는 UI Object를 Click 했을때 Server Event가 작동한다.

1. Press Event가 존재하지 않는 UI Object를 생성한다. (본 단원에서는 "TITLE" UI 사용)



2. ON_INIT에서 EXTUI_EVENTS의 " EV_CLICK"을 설정한다.

[첨부ABAP SOURCE (/U4A/IF_SERVER~HANDL_ON_INIT)]

METHOD /U4A/IF_SERVER~HANDL_ON_INIT.

"EXTUI_EVENTS 중 EV_CLICK는 UI를 CLICK 했을때 EVENT를 작동시킨다

CALL METHOD /U4A/CL_EXTUI_EVENTS=>EV_CLICK

EXPORTING

IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class

I_UID = 'TITLE1' " UI object ID

I_SERVER_EVENT = 'EV_CLICK' " Server Event

* IT_SCRIPT_SOURCE = " Client Event

* I_USE_WAIT = 'X' " Waiting 사용여부('X' = 사용함, ' ' = 사용하지 않음)

* I_USE_PROPAGATION = 'X' " 이벤트 전파방지 사용여부('X' = 사용함)

* EXCEPTIONS

* NOT_SUPPORT_ELEMENT = 1 " attachBrowserEvent not supported UI.

* NOT_FOUND_UI = 2 " UI Instance not found.

* OTHERS = 3

ENDMETHOD.

3. "Title UI"를 Click 했을때 Message를 호출하는 Method를 만든다.

Method	Level	Visibility	M...	Description
/U4A/IF_SERVER~HANDL_ON_INIT	Instance Method	Public		Initialization event
/U4A/IF_SERVER~HANDL_ON_REQUEST	Instance Method	Public		Asynchronous event
/U4A/IF_SERVER~HANDL_ON_RESPONSE	Instance Method	Public		Not used
/U4A/IF_SERVER~HANDL_ON_EXIT	Instance Method	Public		Close event
/U4A/IF_SERVER~HANDL_ON_MESSAGE	Instance Method	Public		Push Channel Services Web Socket message receive event
/U4A/IF_SERVER~HANDL_ON_HASHCHANGE	Instance Method	Public		Router hash change(browser의 뒤로가기) callback event
CONSTRUCTOR	Instance Method	Public		
EV_CLICK	Instance Method	Public		[EVENT] CLICK EVENT

[첨부ABAP SOURCE (EV_CLICK)]

```
METHOD EV_CLICK.
```

```
"메세지 호출.
```

```
CALL METHOD /U4A/CL_UTILITIES=>M_MESSAGE_TOAST
```

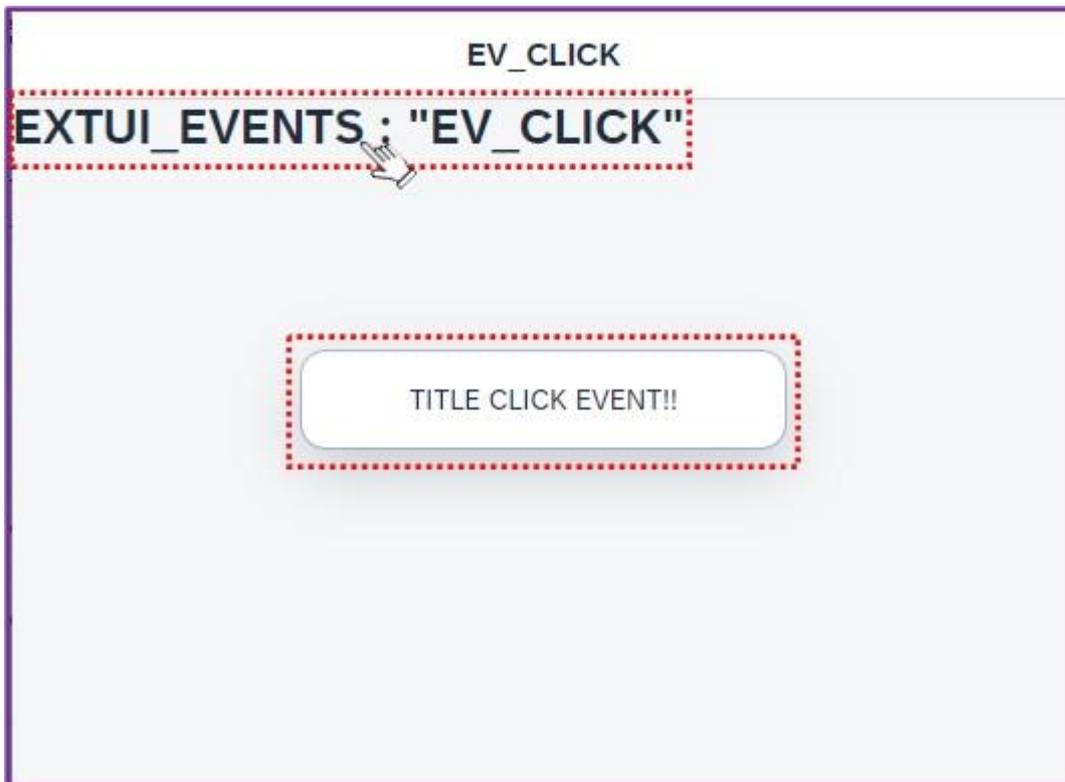
```
EXPORTING
```

```
IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class
```

```
I_MSGTX = 'TITLE CLICK EVENT!!'. " MESSAGE TEXT
```

```
ENDMETHOD.
```

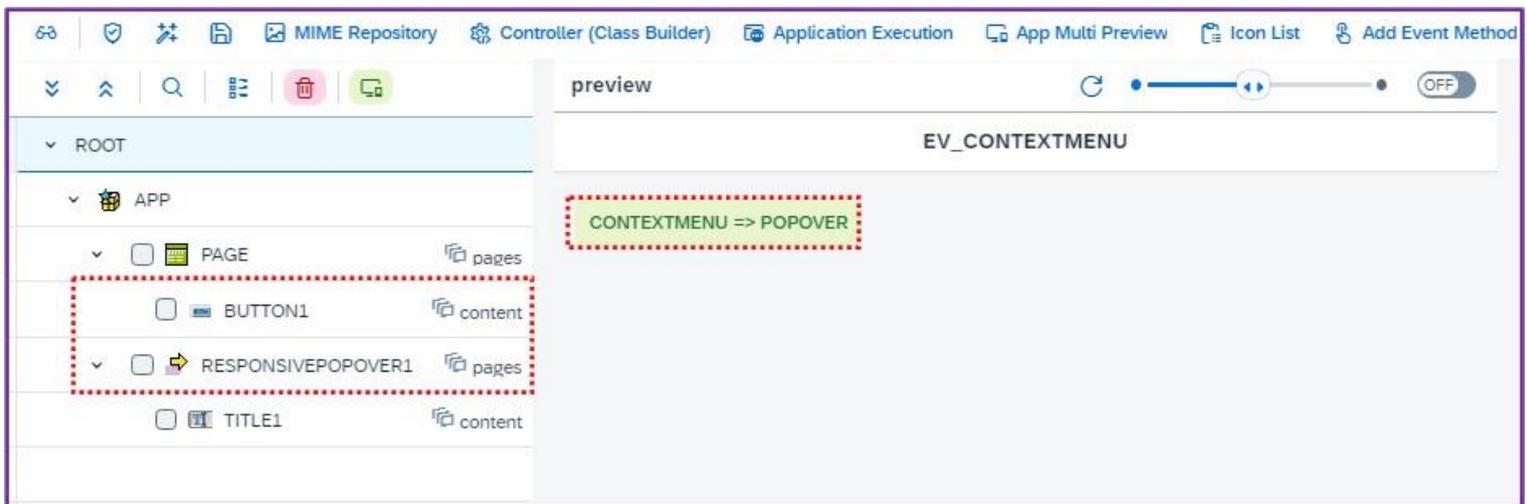
4. Application 실행 후 "TITLE" UI Click시 메세지 호출 여부 확인.



6.3.7 Methodes EXTUI CONTEXTMENU

■ 목적 : EXTUI_EVENTS 중 "EV_CONTEXTMENU"는 UI Object에 마우스 우클릭을 했을 경우 Server Event가 작동한다.

1. UI Object를 생성한다. (마우스 우클릭 대상 UI => "Button" / 마우스 우클릭시 호출 UI => "Responsive Popover")



2. ON_INIT에서 EXTUI_EVENTS의 "EV_CONTEXTMENU"을 설정한다.

[[첨부ABAP SOURCE \(/U4A/IF_SERVER~HANDL_ON_INIT\)](#)]

METHOD /U4A/IF_SERVER~HANDL_ON_INIT.

"EXTUI_EVENTS 중 EV_CONTEXTMENU UI에 마우스 우클릭 했을때 EVENT를 작동시킨다

CALL METHOD /U4A/CL_EXTUI_EVENTS=>EV_CONTEXTMENU

EXPORTING

IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class

I_UID = 'BUTTON1' " UI ID

I_SERVER_EVENT = 'EV_CONTEXTMENU' " Server Event Name(예 : EV_PRESS)

* I_USE_WAIT = 'X' " Waiting 사용여부('X' = 사용함, ' ' = 사용안함)

* IT_SCRIPT_SOURCE = " Client Event

* I_USE_PROPAGATION = 'X' " 이벤트 전파방지 사용여부('X' = 사용함)

* EXCEPTIONS

* NOT_SUPPORT_ELEMENT = 1 " attachBrowserEvent not supported UI.

* NOT_FOUND_UI = 2 " UI Instance not found.

* OTHERS = 3

ENDMETHOD.

3. "Button UI"를 마우스 우클릭 했을때 Responsive Popover를 호출하는 Method를 만든다.

Method	Level	Visibility	M...	Description
/U4A/IF_SERVER~HANDL_ON_INIT	Instance Method	Public		Initialization event
/U4A/IF_SERVER~HANDL_ON_REQUEST	Instance Method	Public		Asynchronous event
/U4A/IF_SERVER~HANDL_ON_RESPONSE	Instance Method	Public		Not used
/U4A/IF_SERVER~HANDL_ON_EXIT	Instance Method	Public		Close event
/U4A/IF_SERVER~HANDL_ON_MESSAGE	Instance Method	Public		Push Channel Services Web Socket message receive event
/U4A/IF_SERVER~HANDL_ON_HASHCHANGE	Instance Method	Public		Router hash change(browser의 뒤로가기) callback event
CONSTRUCTOR	Instance Method	Public		
EV_CONTEXTMENU	Instance Method	Public		마우스 우클릭시 CONTEXTMENU EVENT

[첨부ABAP SOURCE (EV_CONTEXTMENU)]

METHOD EV_CONTEXTMENU.

DATA LO_RESPONSIVEPOPOVER TYPE REF TO /U4A/CL_U000415.

LO_RESPONSIVEPOPOVER ?= ME->/U4A/IF_SERVER~AR_VIEW->GET_UI_INSTANCE(I_ID = 'RESPONSIVEPOPOVER1').

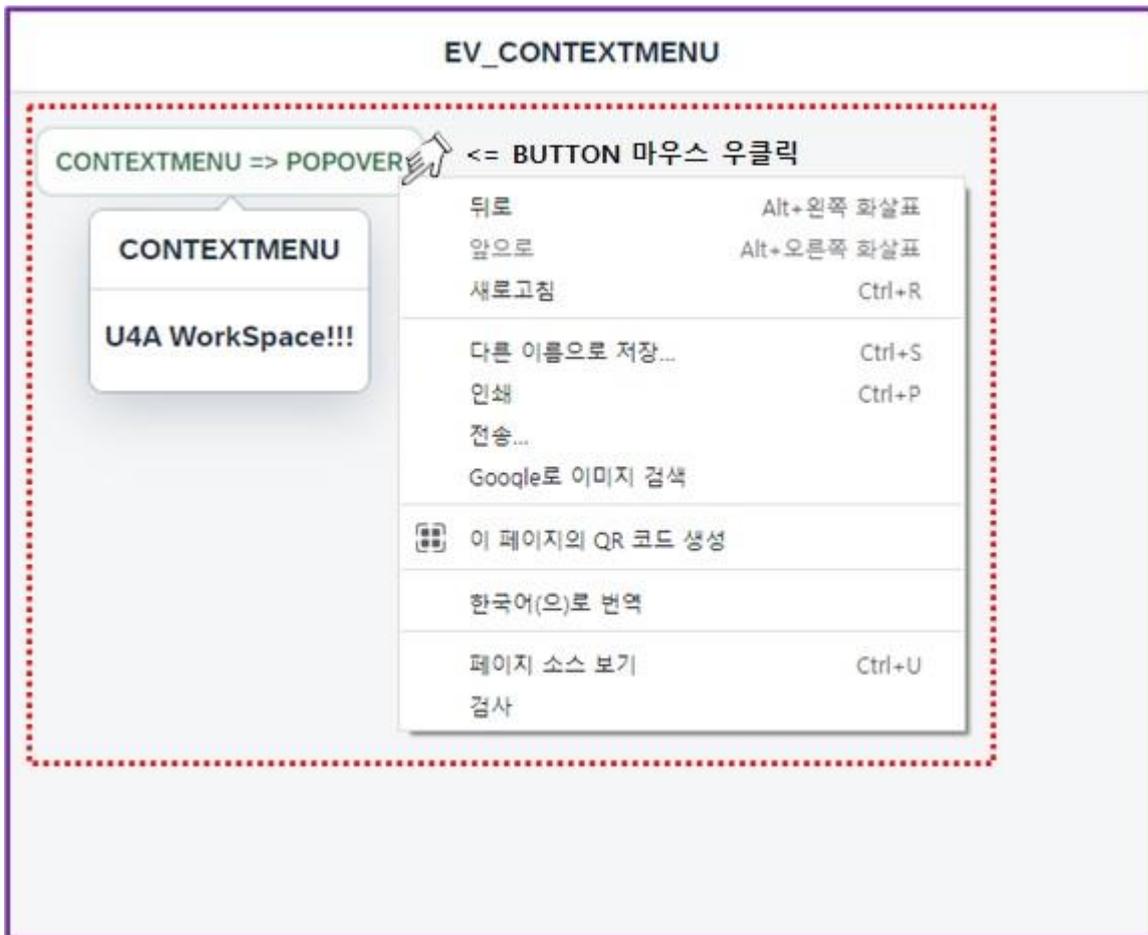
CHECK LO_RESPONSIVEPOPOVER IS BOUND.

"RESPONSIVEPOPOVER OPEN.

LO_RESPONSIVEPOPOVER->OPENBY(EXPORTING OPARENT = 'BUTTON1'). " oParent

ENDMETHOD.

4. Application 실행 후 "BUTTON" UI 마우스 우클릭시 POPOVER 호출 확인.



6.3.8 Methodes EXTUI BROWSER_EVENT

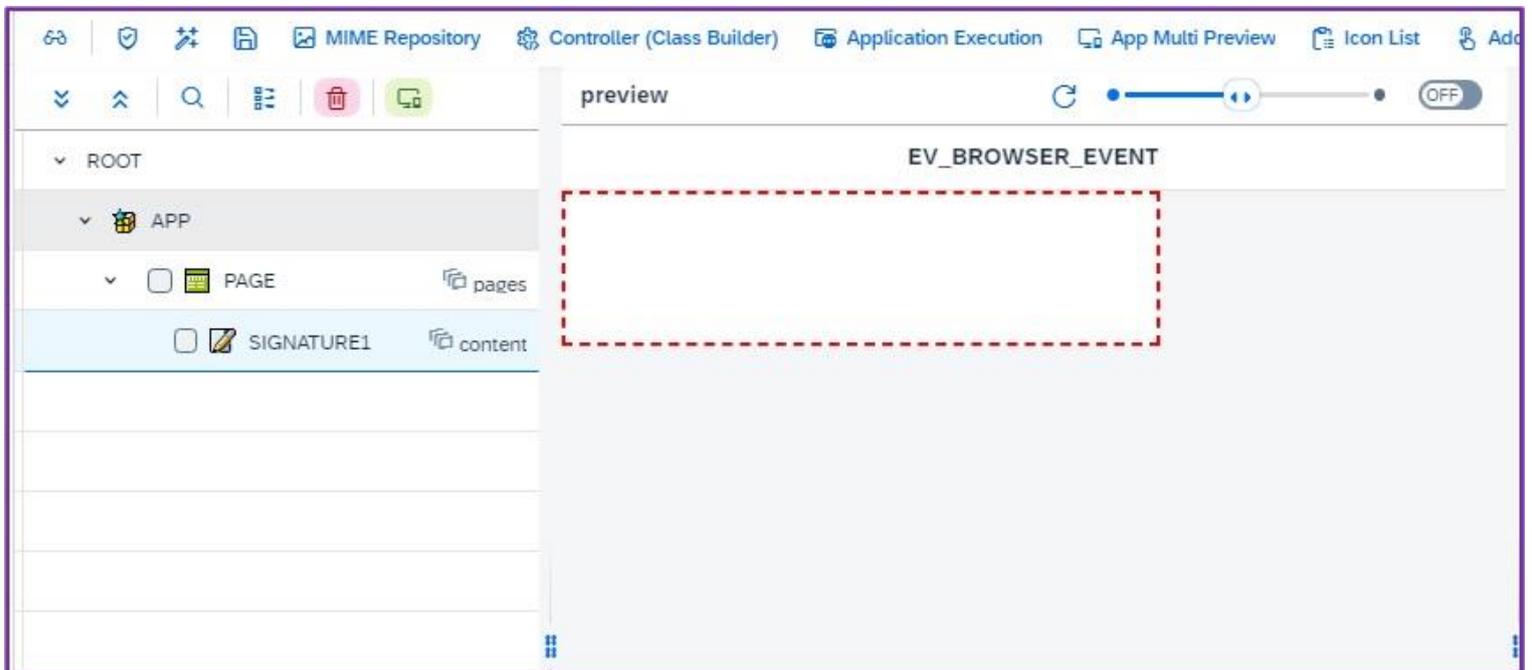
▣ 목적 : EXTUI_EVENTS 중 "EV_BROWSER_EVENT"는 EXTUI_EVENTS, UI가 가지고 있는 EVENT 외 다른 Browser Event가 필요할때 사용한다.

※ 주의사항

- ① 가급적이면 UI에서 제공하는 EVENT를 사용하도록 한다. (Press, Change 등)
- ② 정확하게 사용하지 않고 무분별한 서바이벤트 발생시 프로그램의 퍼포먼스에 영향을 미칠 수 있다.

1. UI Object를 생성한다. (본 단원에서는 SIGNATURE UI 생성)

※ Browser Event 중 "Mouseup" EVENT 사용 => MouseUp (마우스 왼쪽 버튼을 "떨때" 발생하는 이벤트)



2. ON_INIT에서 EXTUI_EVENTS의 " EV_BROWSER_EVENT"을 설정한다.

※ Browser Event 참고 사이트 : <https://developer.mozilla.org/en-US/docs/Web/Events>

[첨부ABAP SOURCE (/U4A/IF_SERVER~HANDL_ON_INIT)]

METHOD /U4A/IF_SERVER~HANDL_ON_INIT.

"EXTUI_EVENTS 중 제공하는 Event외 다른 Browser Event가 필요할때 사용.

CALL METHOD /U4A/CL_EXTUI_EVENTS=>EV_BROWSER_EVENT

EXPORTING

IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class

I_EVENT_NAME = 'mouseup' " events : <https://developer.mozilla.org/en-US/docs/Web/Events>

I_UID = 'SIGNATURE1' " UI object ID

I_SERVER_EVENT = 'EV_BROWSER_EVENT' " Server Event

* IT_SCRIPT_SOURCE = " Client Event

* I_USE_WAIT = " Waiting 사용여부('X' = 사용함, ' ' = 사용안함)

* I_USE_PROPAGATION = " 이벤트 전파방지 사용여부('X' = 사용함)

* EXCEPTIONS

* NOT_SUPPORT_ELEMENT = 1 " attachBrowserEvent not supported UI.

* NOT_FOUND_UI = 2 " UI Instance not found.

* OTHERS = 3

ENDMETHOD.

3. "SIGNATURE UI"에 서명 후 마우스를 "떨때" 메시지를 호출하는 Method를 만든다.

Method	Level	Visibility	M...	Description
/U4A/IF_SERVER~HANDL_ON_INIT	Instance Method	Public		Initialization event
/U4A/IF_SERVER~HANDL_ON_REQUEST	Instance Method	Public		Asynchronous event
/U4A/IF_SERVER~HANDL_ON_RESPONSE	Instance Method	Public		Not used
/U4A/IF_SERVER~HANDL_ON_EXIT	Instance Method	Public		Close event
/U4A/IF_SERVER~HANDL_ON_MESSAGE	Instance Method	Public		Push Channel Services Web Socket message receive event
/U4A/IF_SERVER~HANDL_ON_HASHCHANGE	Instance Method	Public		Router hash change(browser의 뒤로가기) callback event
CONSTRUCTOR	Instance Method	Public		
EV_BROWSER_EVENT	Instance Method	Public		[EVENT] Browser Event

[첨부ABAP SOURCE (EV_BROWSER_EVENT)]

```
METHOD EV_BROWSER_EVENT.
```

"메세지 호출

```
CALL METHOD /U4A/CL_UTILITIES=>M_MESSAGE_TOAST
```

```
EXPORTING
```

```
IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW " [U4A] UI Element Super Class
```

```
I_MSGTX = 'Mouse Up Event!!!'. " MESSAGE TEXT
```

```
ENDMETHOD.
```

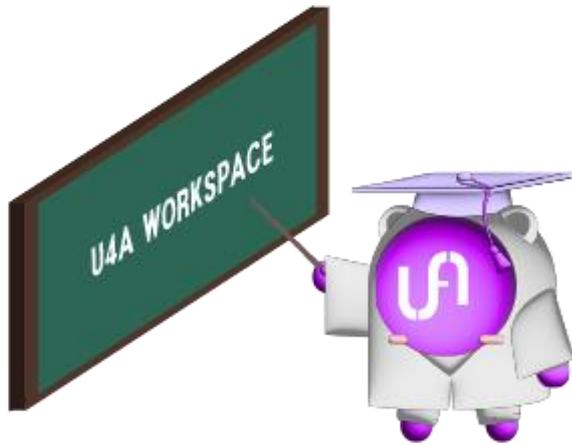
4. Application 실행 후 "SIGNATURE UI" 서명 후 마우스를 떨때 "Mouseup" Event 호출 확인.

EV_BROWSER_EVENT

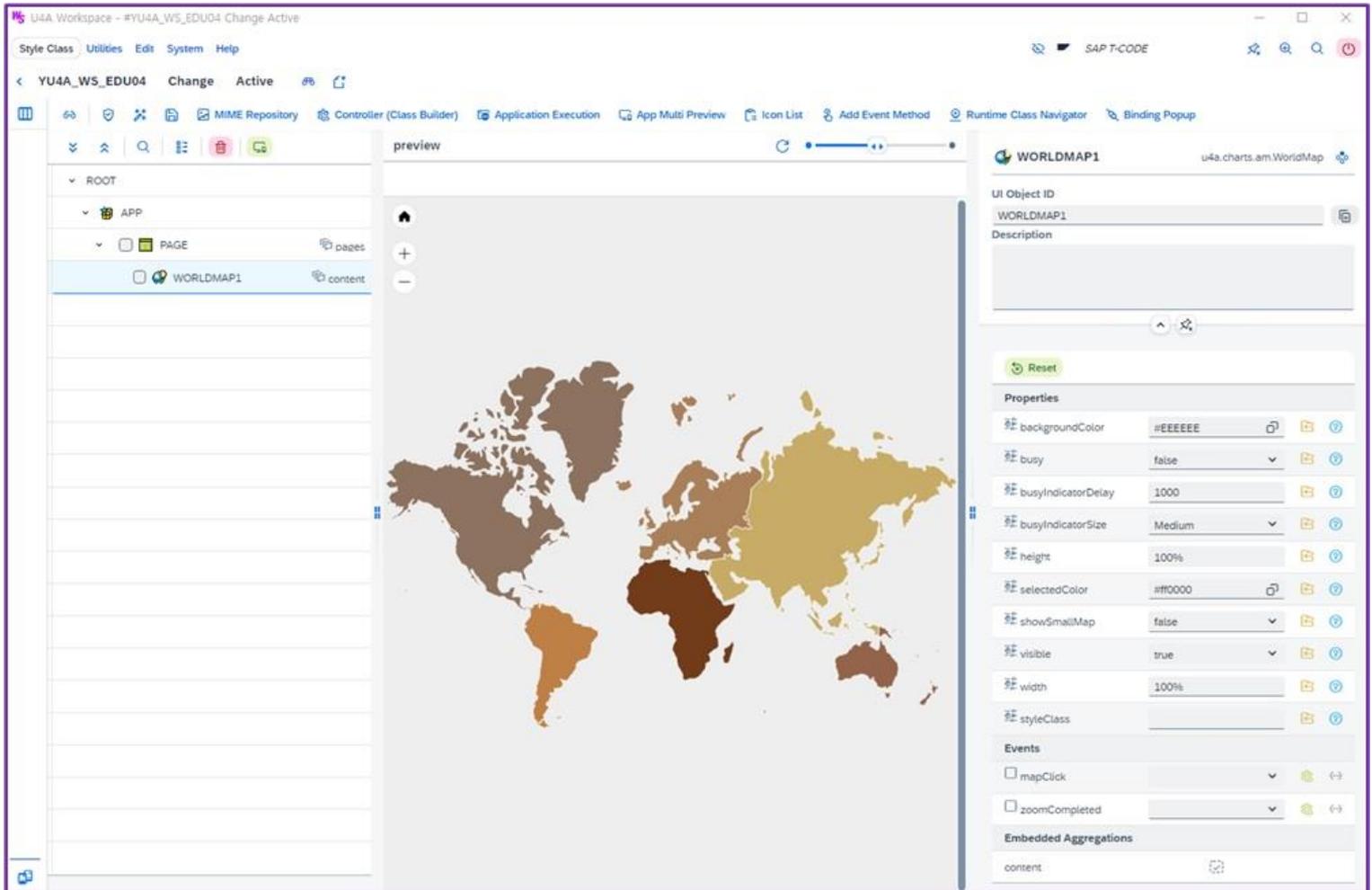
A white rectangular box containing the handwritten signature 'lew' in black ink.

Mouse Up Event!!!

7. APP 개발 하기 Part 3



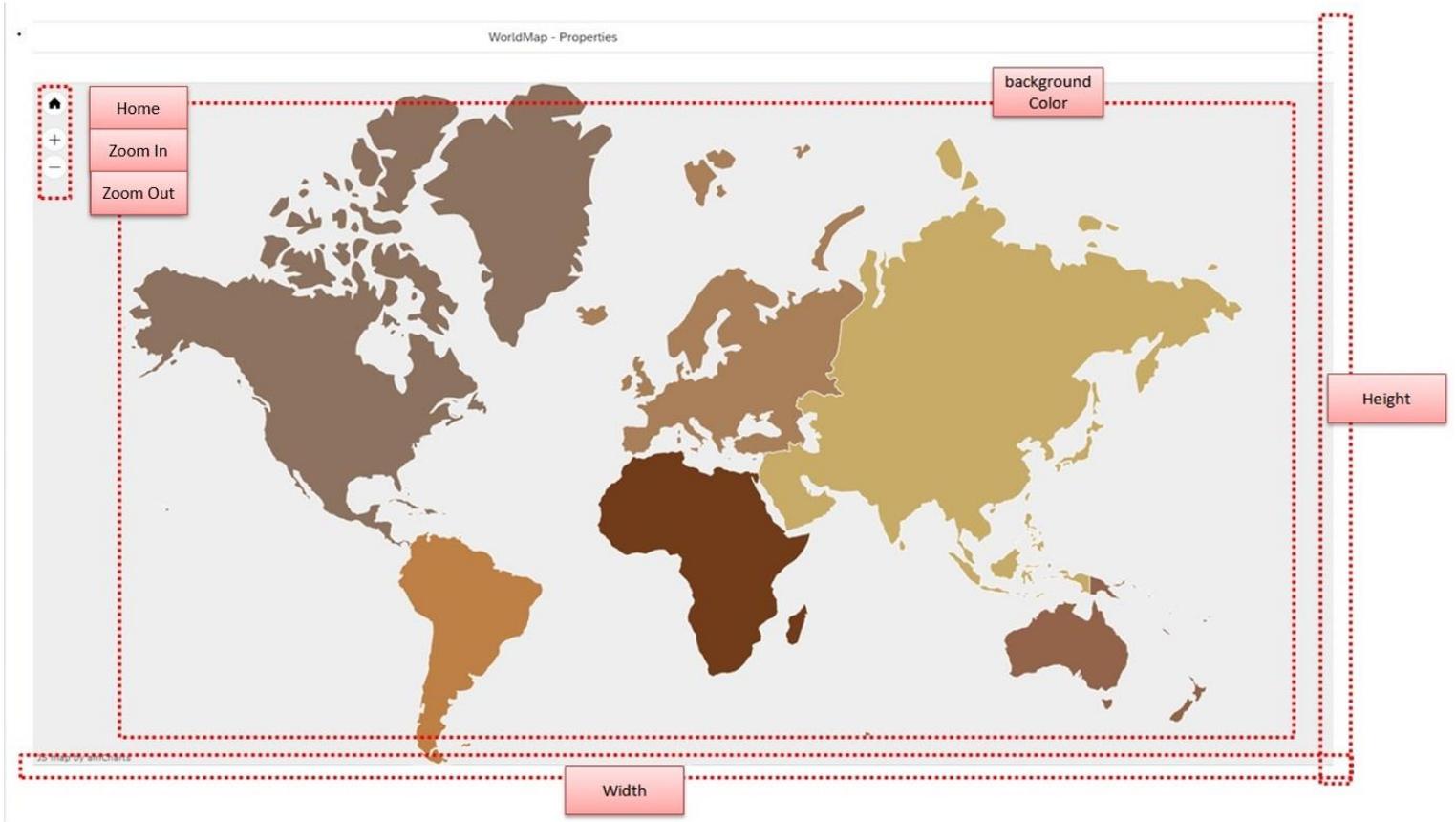
7.1 WorldMap



PROPERTIES	MEAN
backgroundColor	WorldMap UI Color
busy	Loading 상태 전환 여부
busyindicatorDelay	지정한 시간 뒤에 Loading 상태로 전환 (단위 : ms).
busyindicatorSize	Loading Indicator의 크기
height	WorldMap 세로 높이
selectedColor	지역 선택 했을 때 표시 되는 Color
showSmallMap	우측 상단에 축소된 형태의 UnitedStateMap 생성

visible	WorldMap을 보여줄 지 여부
width	WorldMap 가로 넓이
styleClass	WorldMap 적용 할 CSS

2. WorldMap 실행 화면.



3. WorldMap Properties - Busy.



- WorldMap Busy 상태, busyindicatorDelay에서 지정한 시간 후에 Indicator 생성 .

4. WorldMap Properties - selectedColor.



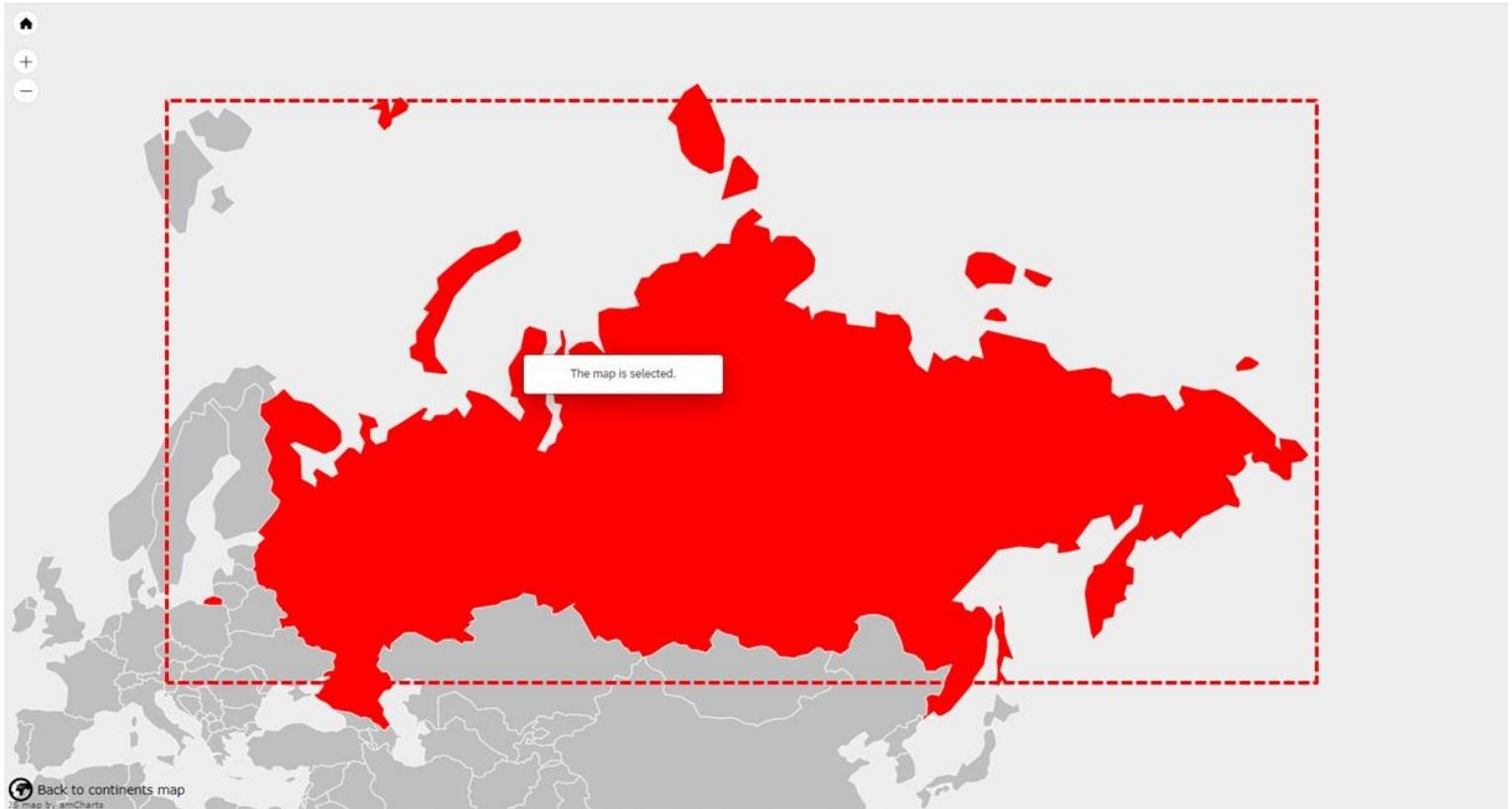
☐ 마우스로 Click한 지역은 위 Image와 같이 Color가 부여되어 표시되며, selected Color는 이 때 부여되는 Color를 설정.

5. WorldMap Properties - showSmallMap.



▣ 우측 상단에 SmallMap이 생성된 것을 확인, SmallMap은 접기, 펼치기가 가능하며 SmallMap에서 지역을 Click하면 WorldMap 화면이 해당 지역으로 이동.

6. WorldMap Events - mapClick.



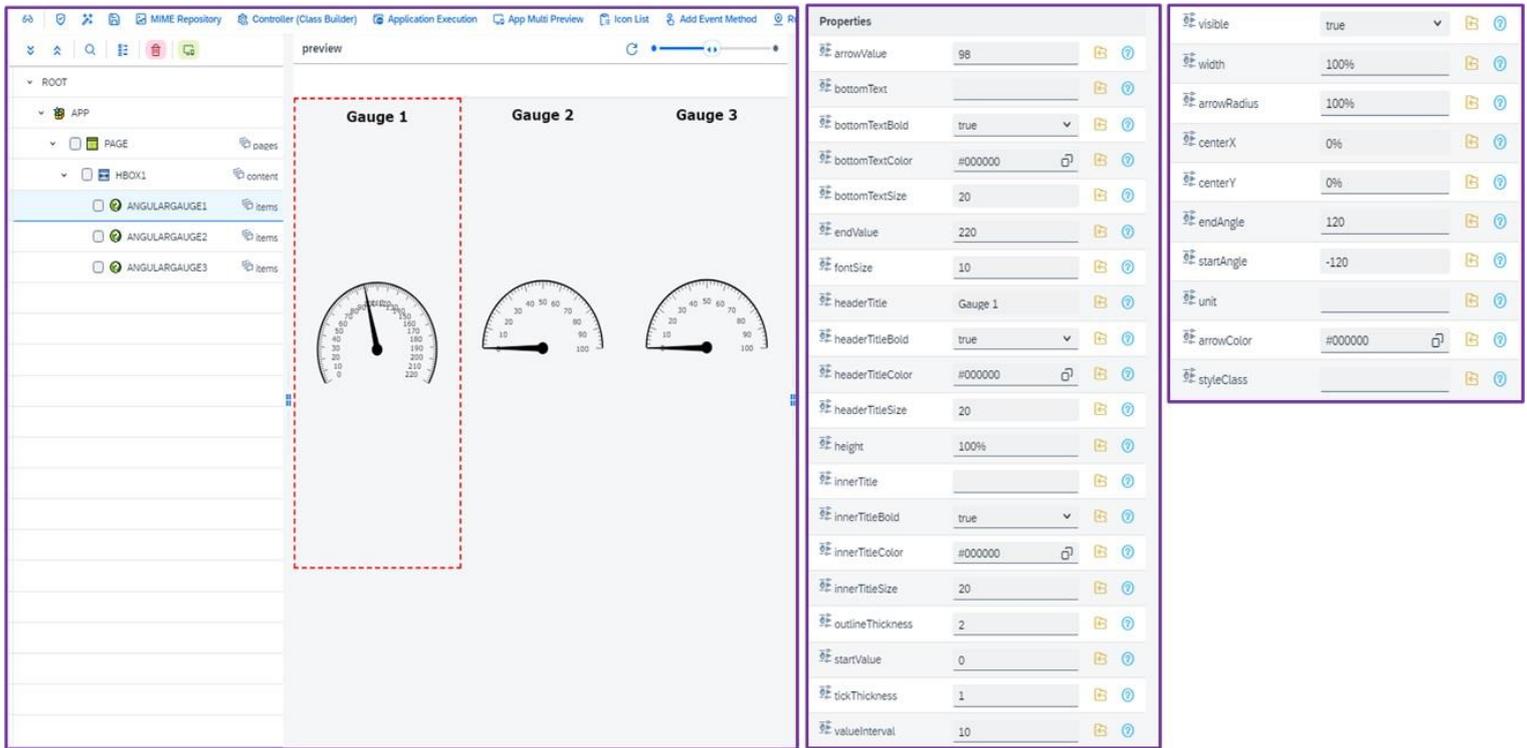
▣ WorldMap을 Click 했을 때 발생하는 Event를 정의.

7. WorldMap Events - zoomCompleted.



▣ WorldMap을 확대하거나 축소했을 때 발생하는 Event를 정의.

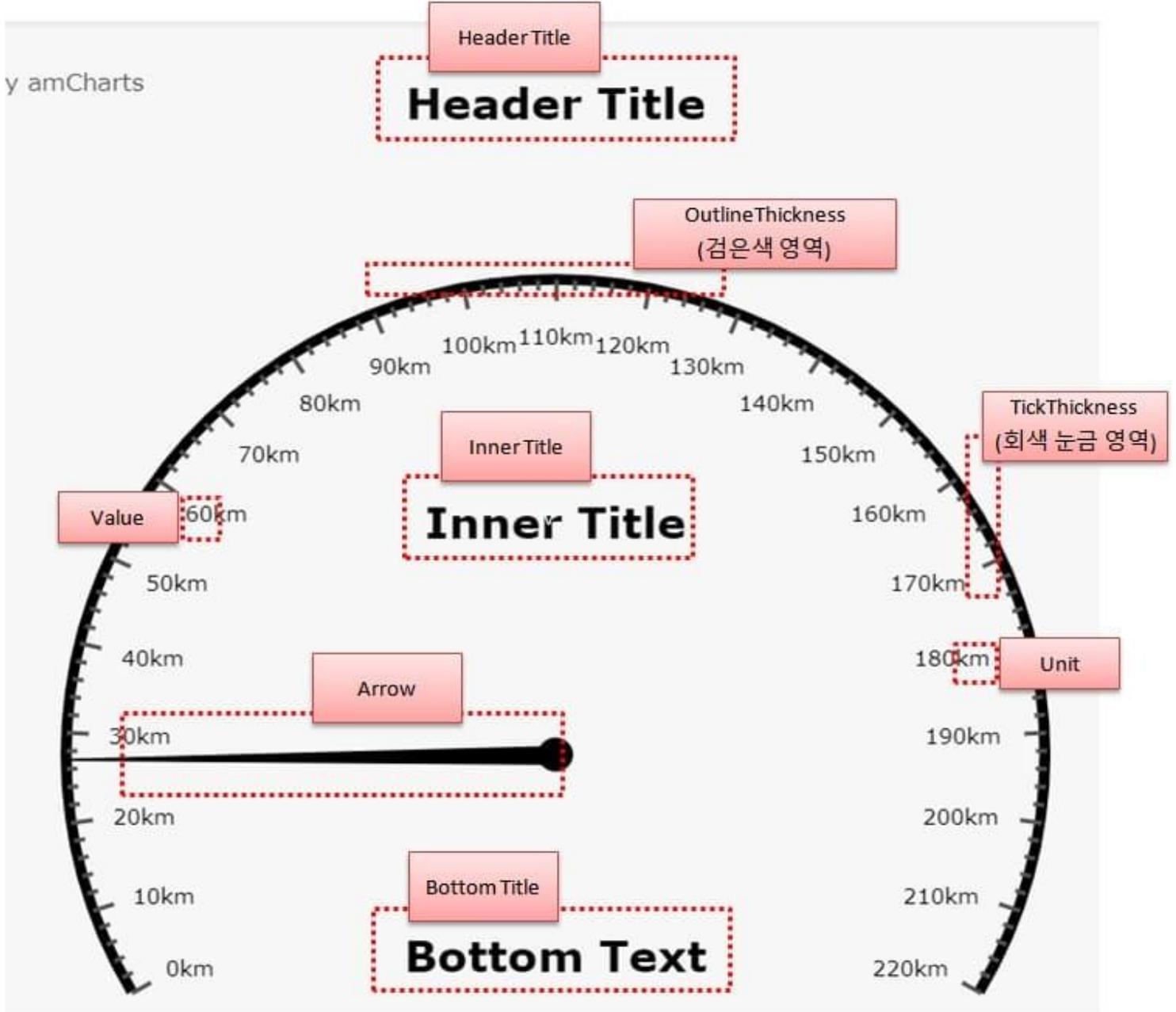
7.2 Angulargauge



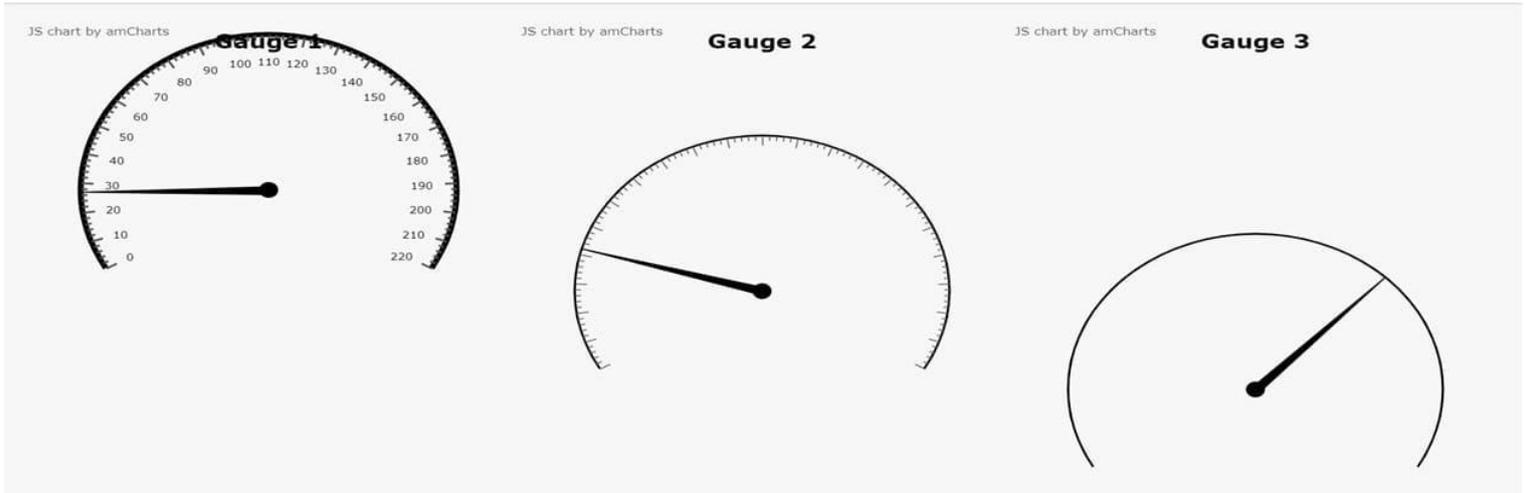
PROPERTIES	MEAN
arrowValue	계기판 화살표 값
bottom Text	계기판 중앙 아래 Text 입력
bottom TextBold	입력한 bottom Text 굵기 변경
bottom TextColor	입력한 bottom Text Color 변경
bottom TextSize	입력한 bottom Text Size 변경
endValue	계기판 속도 종료값
fontSize	계기판 속도 값 size
header Title	계기판 상단에 Title 입력
headerTitleBold	입력한 header Title 굵기 변경
headerTitleColor	입력한 header Title Color 변경

headerTitleSize	입력한 header Title Size 변경
height	계기판 세로 높이 조절
innerTitle	계기판 내부에 Title 입력
innerTitleBold	입력한 innerTitle 굵기 변경
innerTitleColor	입력한 innerTitle Color 변경
innerTitleSize	입력한 innerTitle Size 변경
outlineThickness	계기판 라인 두께 변경
startValue	계기판 속도 시작값
tickThickness	계기판 속도 눈금 굵기
valueInterval	계기판 속도 간격 설정
visible	보이게 할지 여부 설정
width	계기판 가로 넓이 조절
arrowRadius	계기판 화살표 길이
centerX	계기판 가로 위치 (Ex: 100px, -80px, 80%, -100%)
centerY	계기판 세로 위치 (Ex: 100px, -80px, 80%, -100%)
endAngle	계기판 오른쪽 각도
startAngle	계기판 왼쪽 각도
unit	계기판 속도 값 옆에 UNIT 값 설정
styleClass	계기판 CSS 설정

2. Angulargauge Property 위치.

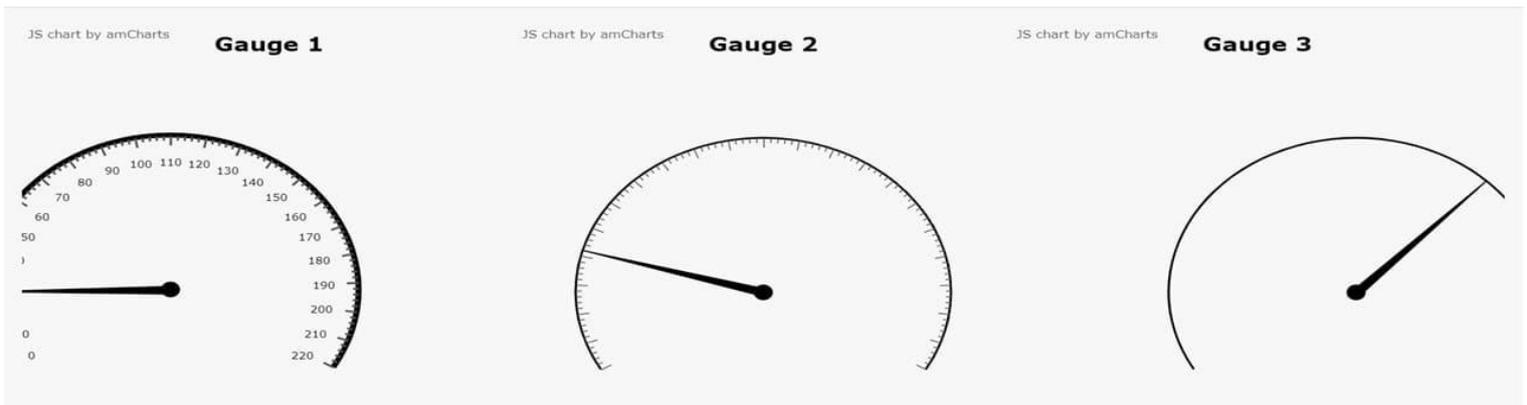


3. Angulargauge Property - centerY 조절 화면.



■ -값이 증가하면 ui가 위로 올라가고 +값이 증가하면.ui가 아래로 내려감.

4. Angulargauge Property - centerX 조절 화면.

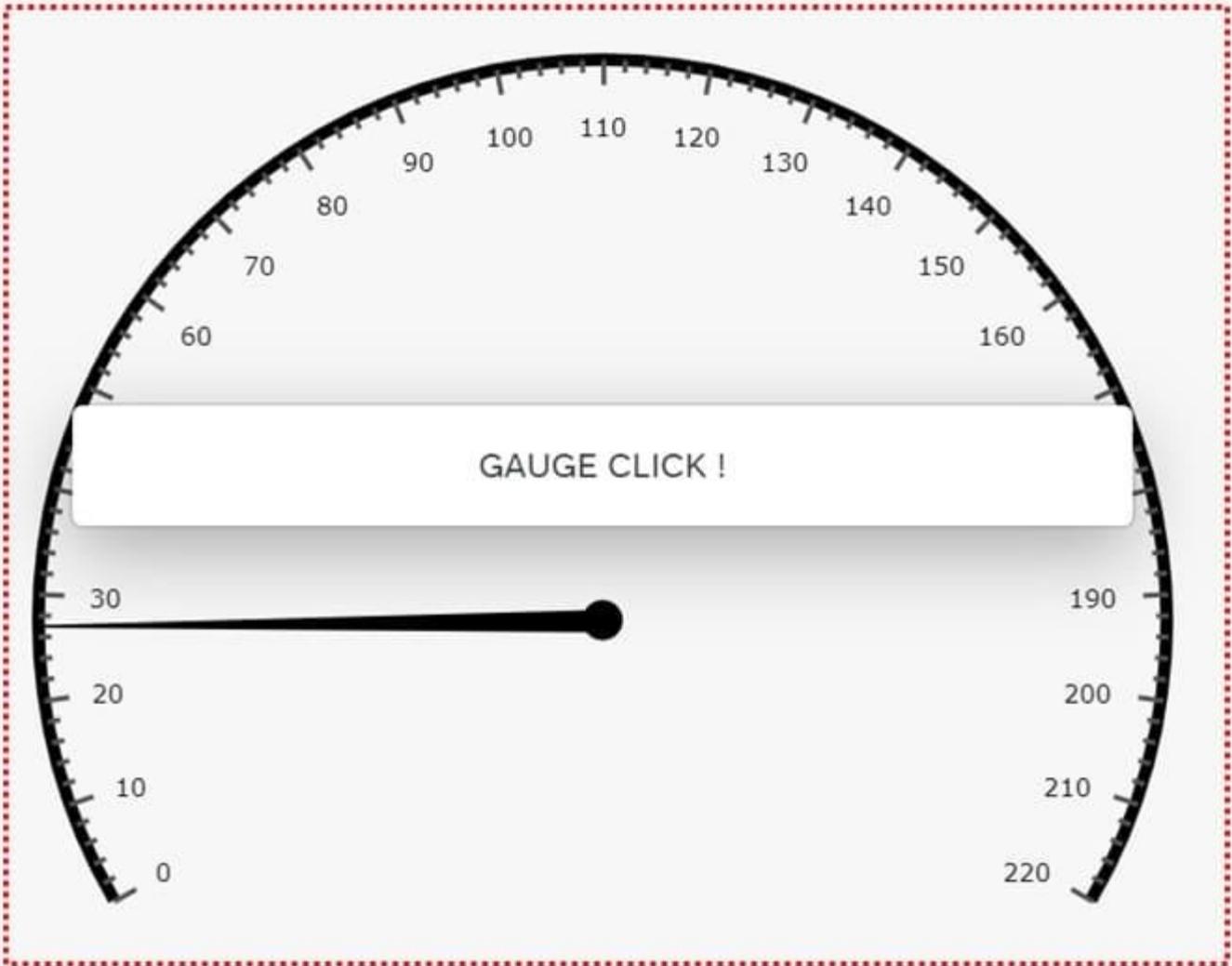


■ -값이 증가하면 ui가 왼쪽으로 이동하고 +값이 증가하면 ui가 오른쪽으로 이동.

5. Angulargauge Events - GaugeClick.

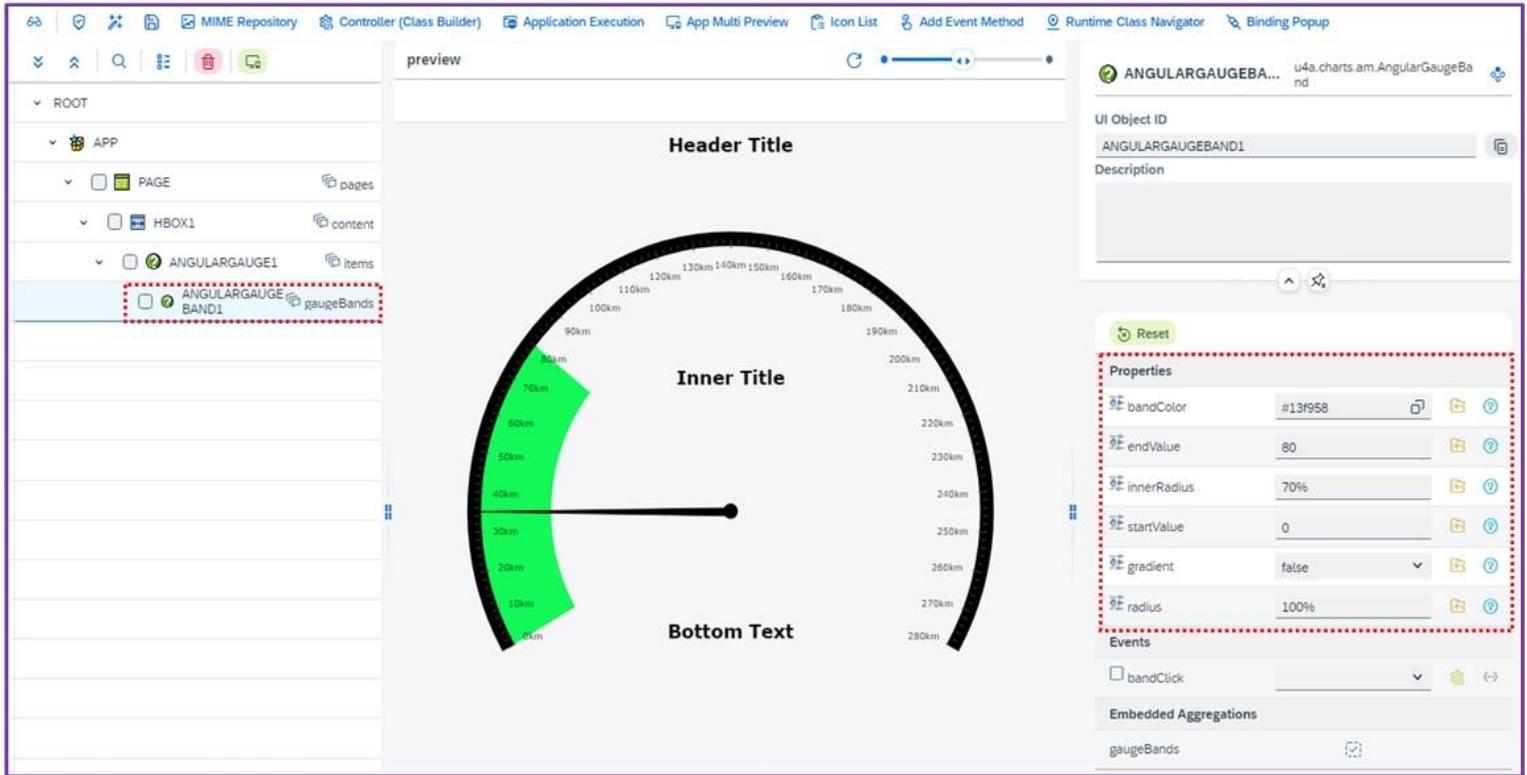
by amCharts

Gauge 1



▣ Angulargauge Ui Click 시 발생하는 Event 설정.

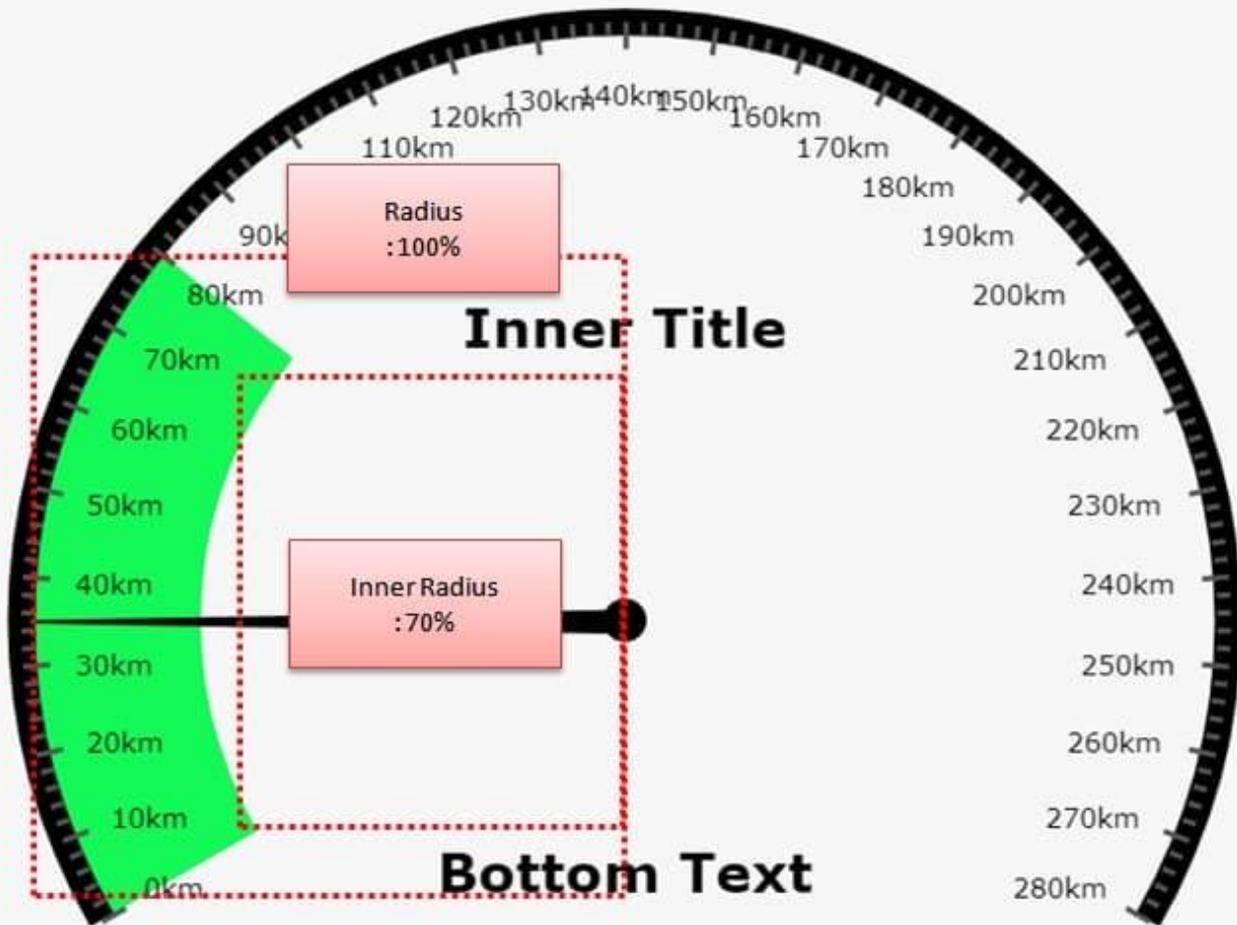
6. Angulargauge Aggregation - AngulargaugeBand.

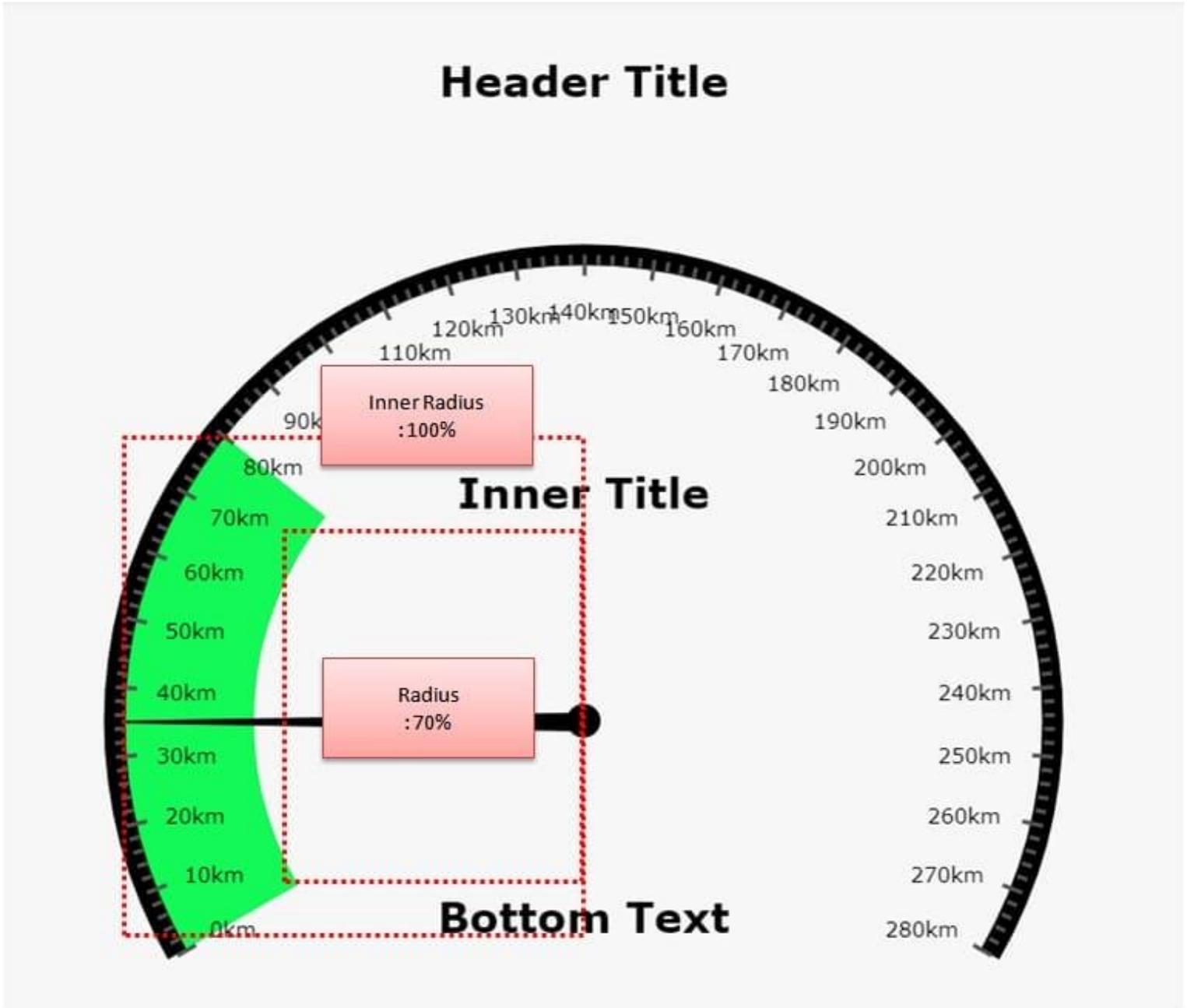


PROPERTIES	MEAN
bandColor	AngularGaugeBand Color
endValue	AngularGaugeBand가 AngularGauge에 표시될 마지막 위치
innerRadius	AngularGaugeBand가 AngularGauge에 표시될 영역의 기준점 위치, AngularGauge의 가운데 위치를 0%, 테두리 라인까지 100%를 기준으로 계산, ※ radius property와 한쌍으로 입력해야 함.
startValue	AngularGaugeBand가 AngularGauge에 표시될 처음 위치
gradient	AngularGaugeBand의 색상을 Gradation 효과를 적용할지 여부
radius	AngularGaugeBand가 AngularGauge에 표시될 영역의 기준점 위치, AngularGauge의 가운데 위치를 0%, 테두리 라인까지 100%를 기준으로 계산, ※ radius property와 한쌍으로 입력해야 함.

7. AngulargaugeBand Property - Radius, InnerRadius (한쌍으로 함께 사용).

Header Title





▣ 왼쪽 이미지를 예시로 든다면 InnerRadius가 70%인 영역부터 시작해서 Radius가 100%인 영역까지 AngularGaugeBand가 표시.

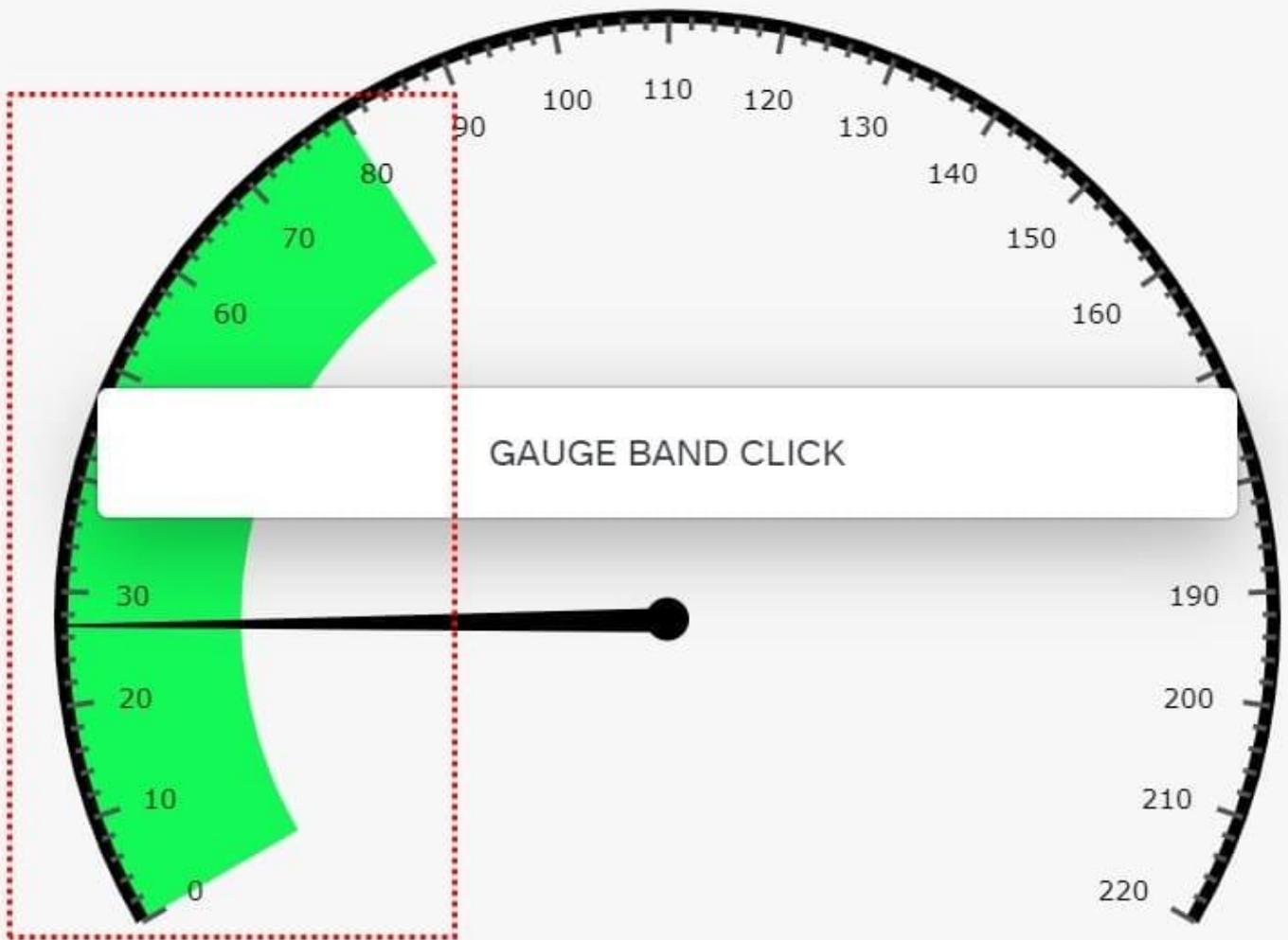
오른쪽 이미지를 예시로 든다면 Radius가 70%인 영역부터 시작해서 InnerRadius가 100%인 영역까지 AngularGaugeBand가 표시,

결국 두 Property가 하는 역할은 동일하며 Value가 더 작은 Property부터 시작해서 Value가 더 큰 Property까지 표시.

8. AngularGaugeBand Events - BandClick.

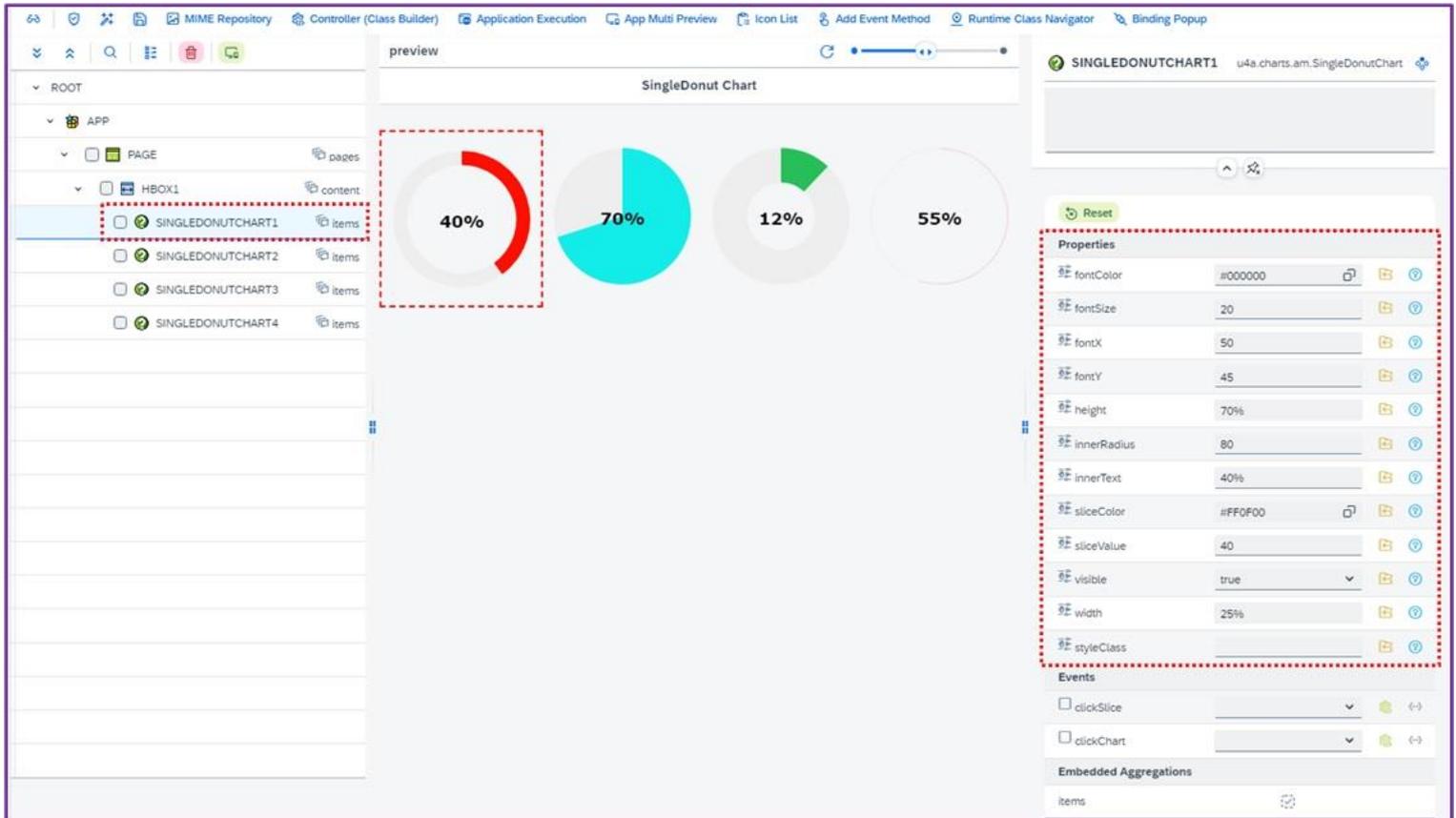
by amCharts

Gauge 1



■ AngularGaugeBand Ui Click 시 발생하는 Event 설정.

7.3 SingleDonutChart



PROPERTIES	MEAN
fontColor	Inner Text Property에 입력한 Text의 Color
fontSize	Inner Text Property에 입력한 Text의 Size
fontX	Inner Text Property에 입력한 Text의 가로 위치
fontY	Inner Text Property에 입력한 Text의 세로 위치
height	Chart 영역의 높이
innerRadius	Chart 내부 영역의 원형 크기 숫자가 작을수록 차트 가운데 영역의 크기가 작음, 숫자가 클 수록 차트 가운데 영역의 크기가 큼.
innerText	Chart 가운데 영역에 보여지는 Text
sliceColor	Chart 영역의 Color

sliceValue	Chart의 전체 영역에서 "sliceColor"의 색상이 적용될 부분 min: 0이며 0보다 작은값 입력시 0만큼 자동적용, max:100 이며 100보다 큰 값 입력시 100만큼 자동 적용
visible	Chart를 보이게 할 지 여부 결정
width	Chart 가로 넓이
styleClass	Chart CSS 설정

2. SingleDonutChart Property 위치.

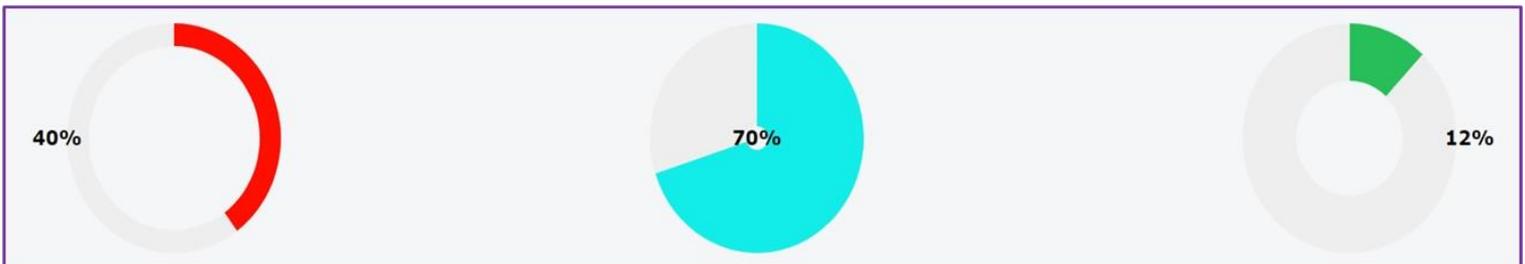


3. SingleDonutChart Properties - InnerRadius.



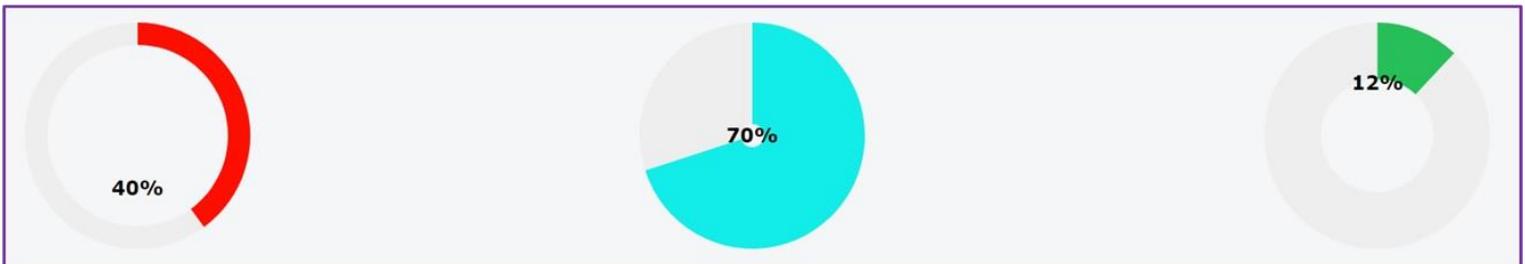
■ min: 0, max: 100이며 InnerRadius 값을 통해 위와 같이 DonutChart를 다양한 형태로 보여줄 수 있음.

4. SingleDonutChart Properties - fontX 조절 화면.



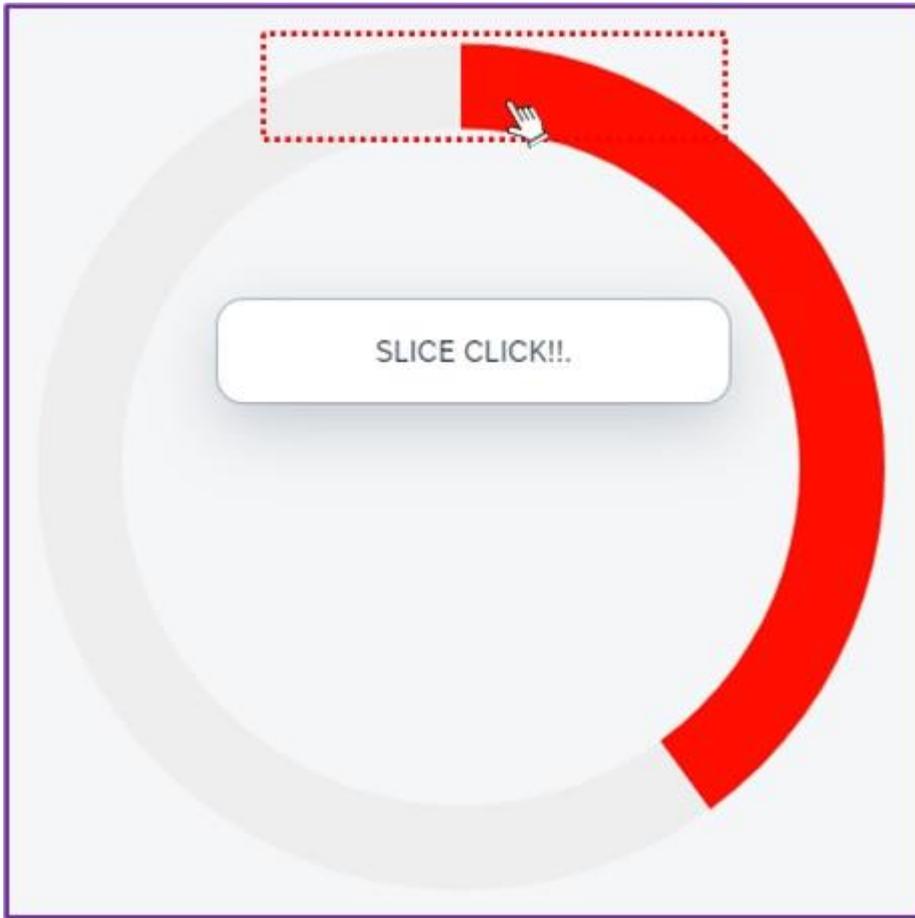
■ fonX의 값이 50 이하면 Text가 왼쪽으로 이동, 50이면 중앙, 50 이상이면 오른쪽으로 이동.

5. SingleDonutChart Properties - fontY 조절 화면.



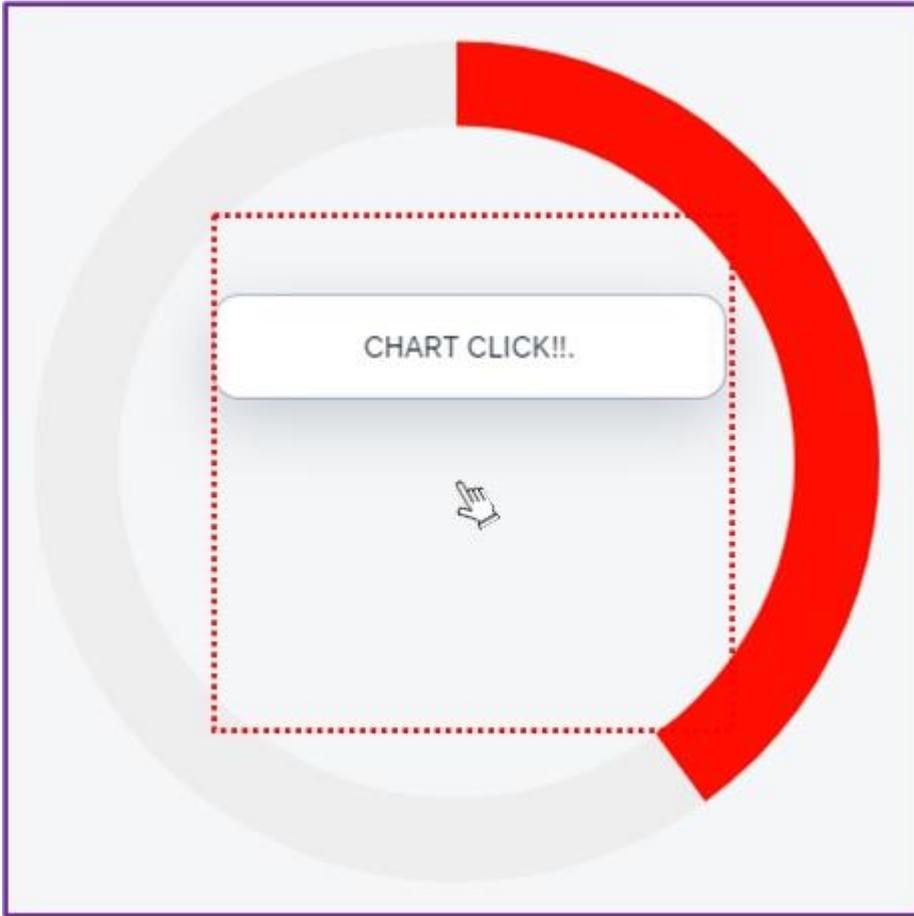
■ fonY의 값이 45 이하면 Text가 위쪽으로 이동, 45라면 중앙, 45이상이면 아래쪽으로 이동.

6. SingleDonutChart Events - clickSlice .



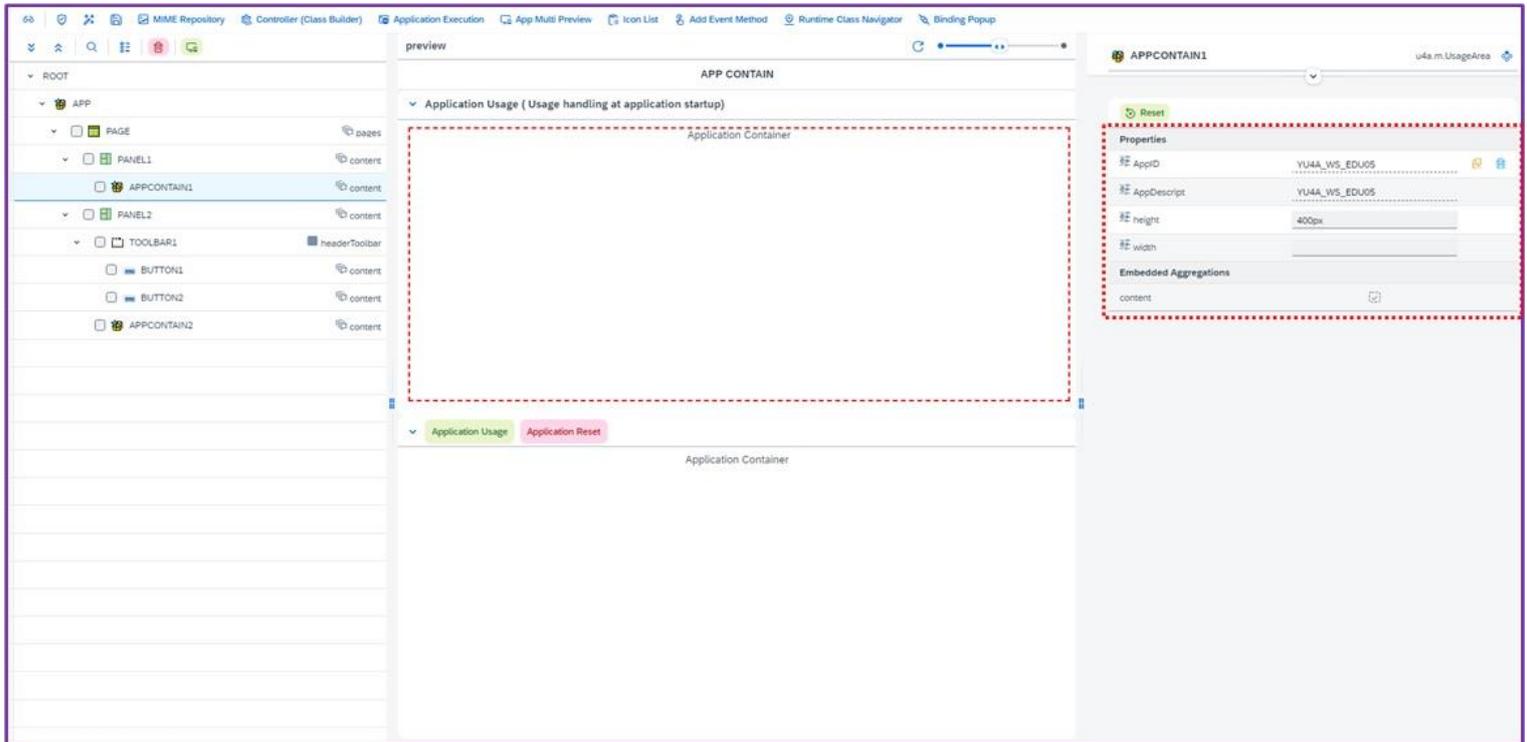
■ Chart Ui 가장자리 영역 Click 시 발생하는 Event 설정.

7. SingleDonutChart Events - clickChart .



- ▣ Chart Ui 내부 영역 Click 시 발생하는 Event 설정.

7.4 AppContain



PROPERTIES	MEAN
AppID	Rendering 하려는 Application ID 이며 Click 하면 해당 Application으로 이동
AppDescript	Rendering 하려는 Application에 대한 설명
height	Rendering 되는 Application이 보여질 영역의 세로 높이.
width	Rendering 되는 Application이 보여질 영역의 가로 넓이.

2. Rendering_Usage_App Method.

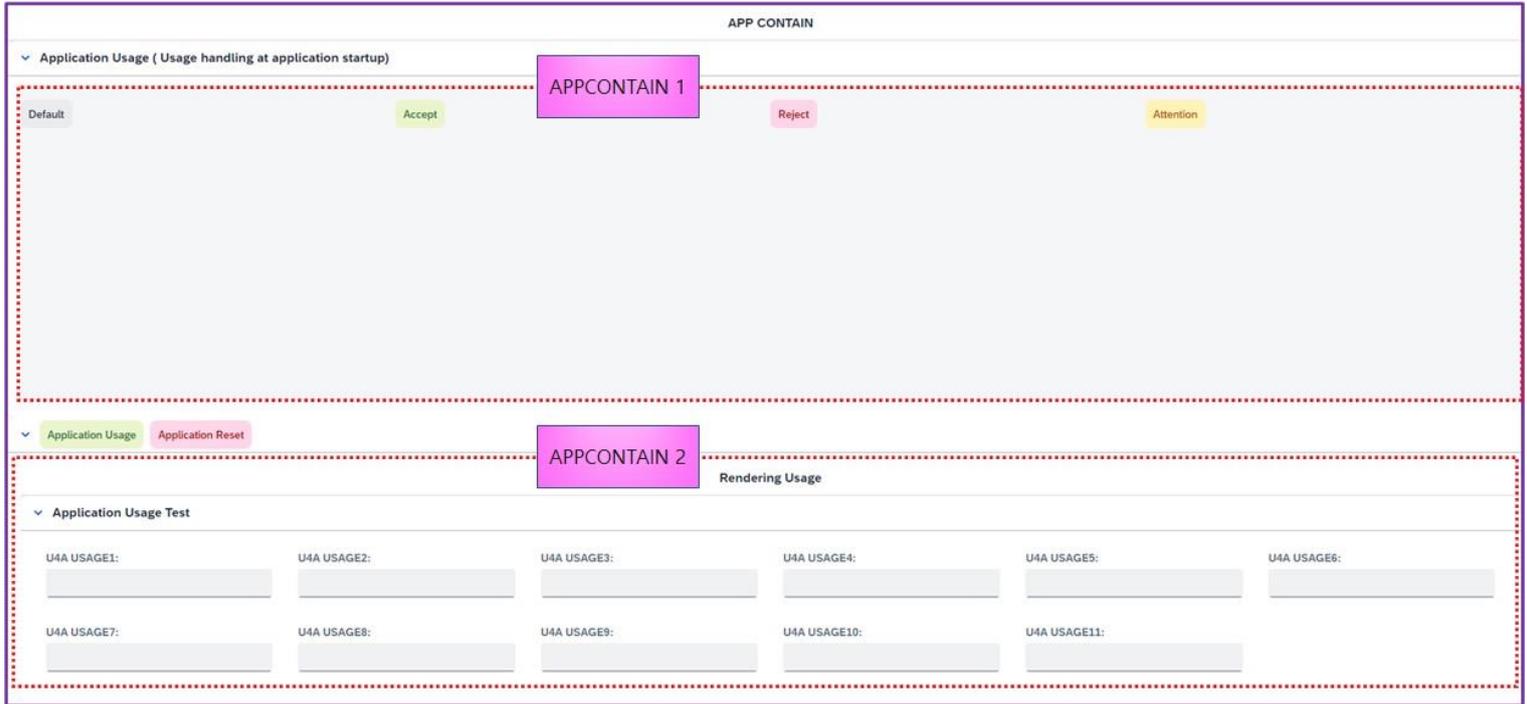
▣ AppContain을 통해 Application Rendering을 하려면 Rendering_Usage_App Method를 사용.

Rendering_Usage_App Method에 대한 상세설명 및 사용법은 App 개발하기 Part2 - Rendering_Usage_App을 통해 확인.

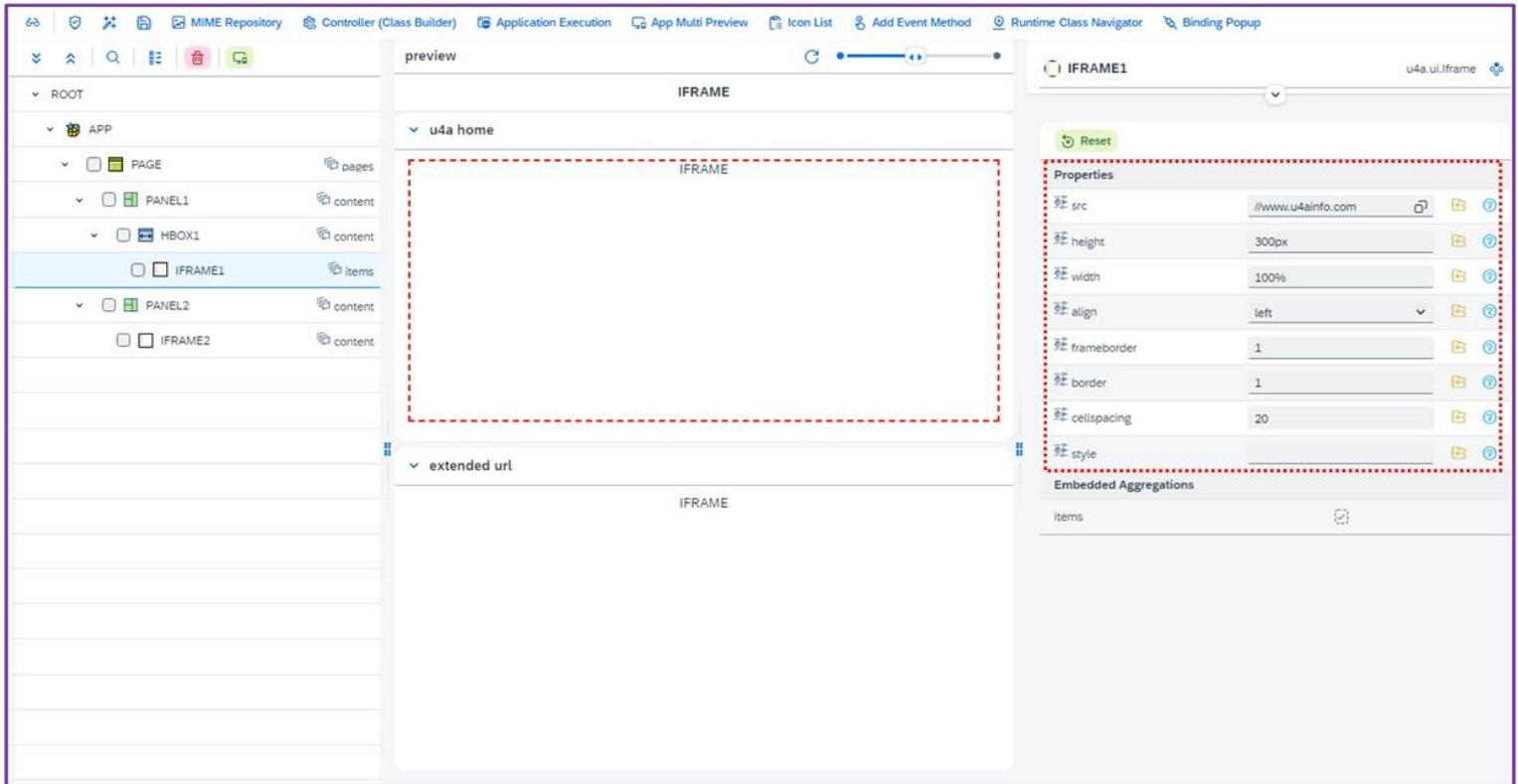
3. Reset_Usage_App Method.

- 현재 AppContain에 연결되어 있는 Usage_App 정보를 Reset 하려면 Reset_Usage_App Method를 사용.
Reset_Usage_App Method에 대한 상세설명 및 사용법은 App 개발하기 Part2 - Reset_Usage_App을 통해 확인.

4. AppContain을 통해 Application Rendering Usage 결과 화면.

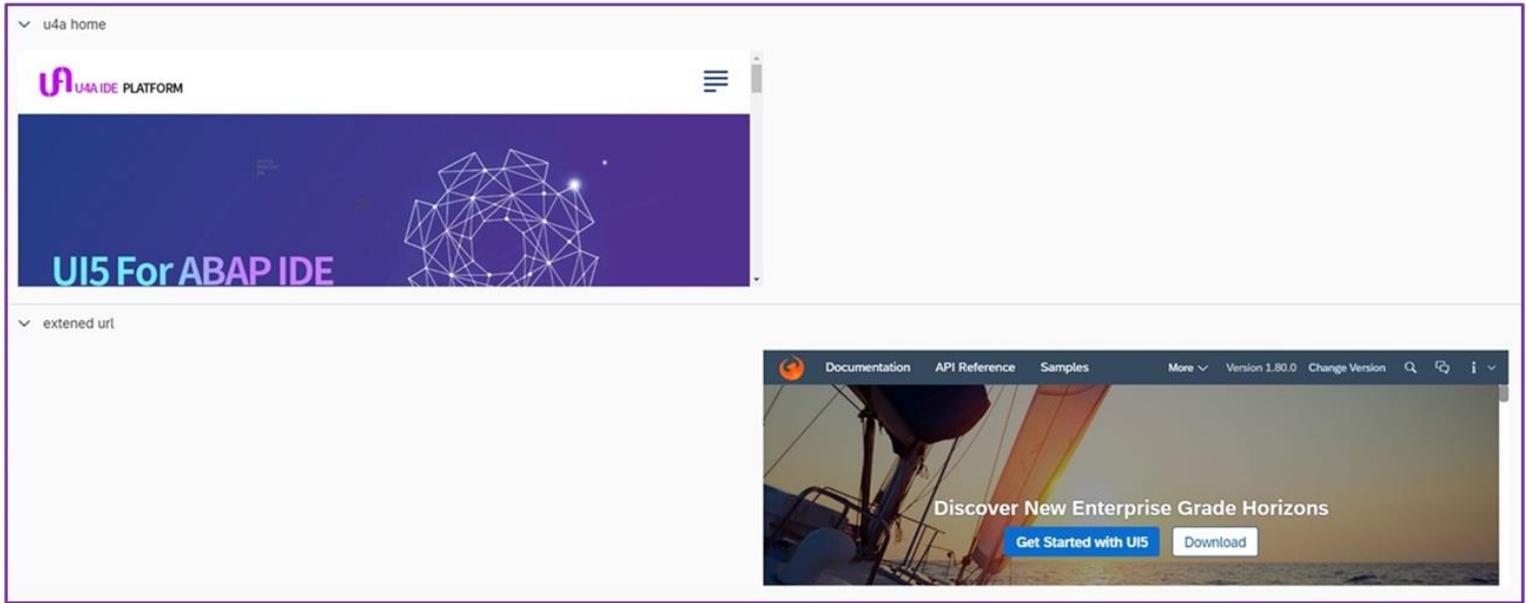


7.5 Iframe



PROPERTIES	MEAN
src	삽입 하려는 WebPage URL 주소
height	Ifram 세로 높이 설정
width	Ifram 가로 넓이 설정
align	Ifram 정렬 (left, right, middle, top, bottom)
frameborder	Ifram 경계선 유무 설정 (0 : 경계선 없음 1 : 경계선 있음)
border	업데이트 예정
cellspacing	업데이트 예정
style	CSS 직접 입력하여 사용

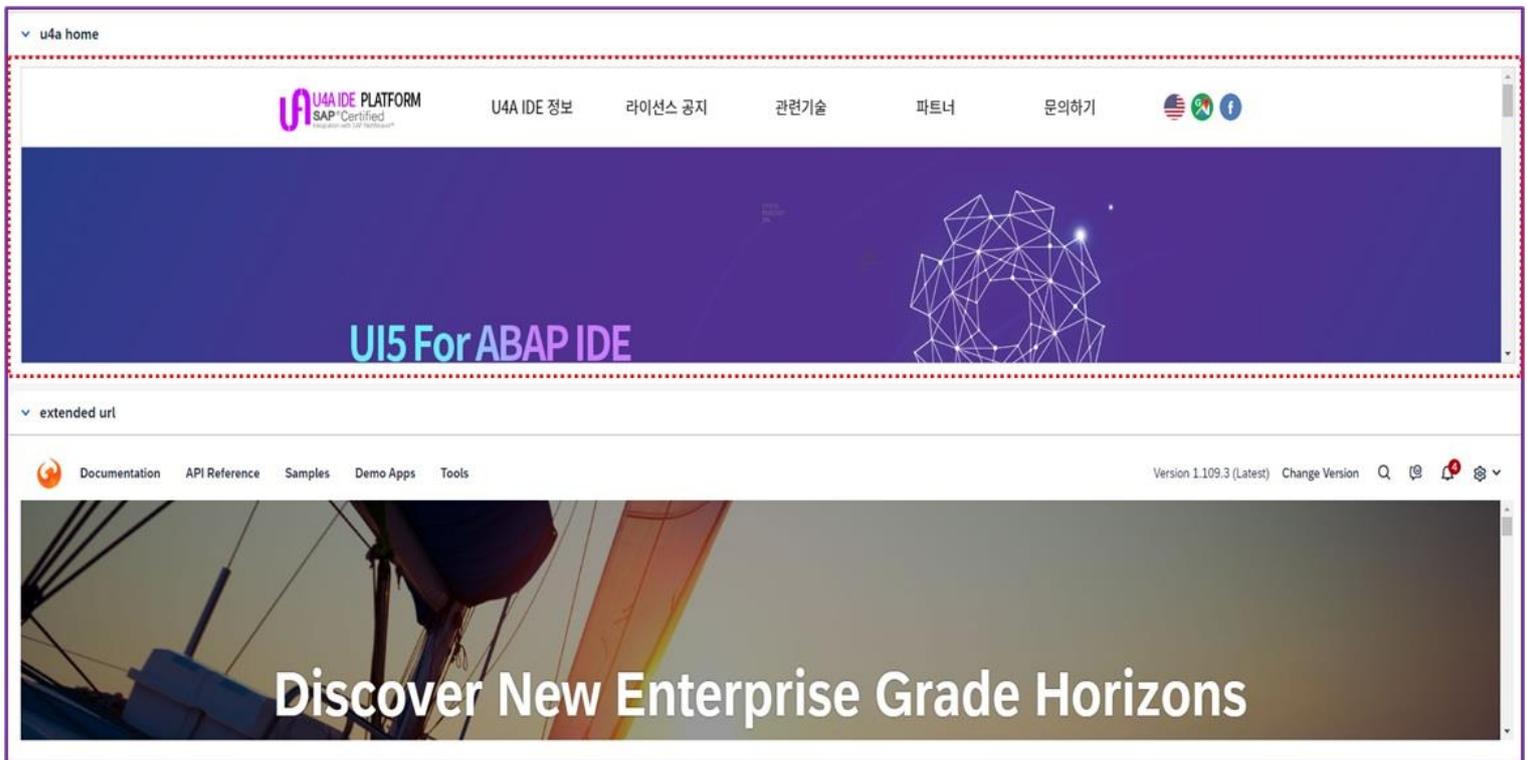
2. Iframe Property - Align.



☐ 상단 Ifram : align : left, 하단 Ifram : align : right 으로 설정하였고 Application 실행 하여 상단 Ifram은 좌측 정렬, 하단 Ifram은 우측 정렬 된 것을 확인.

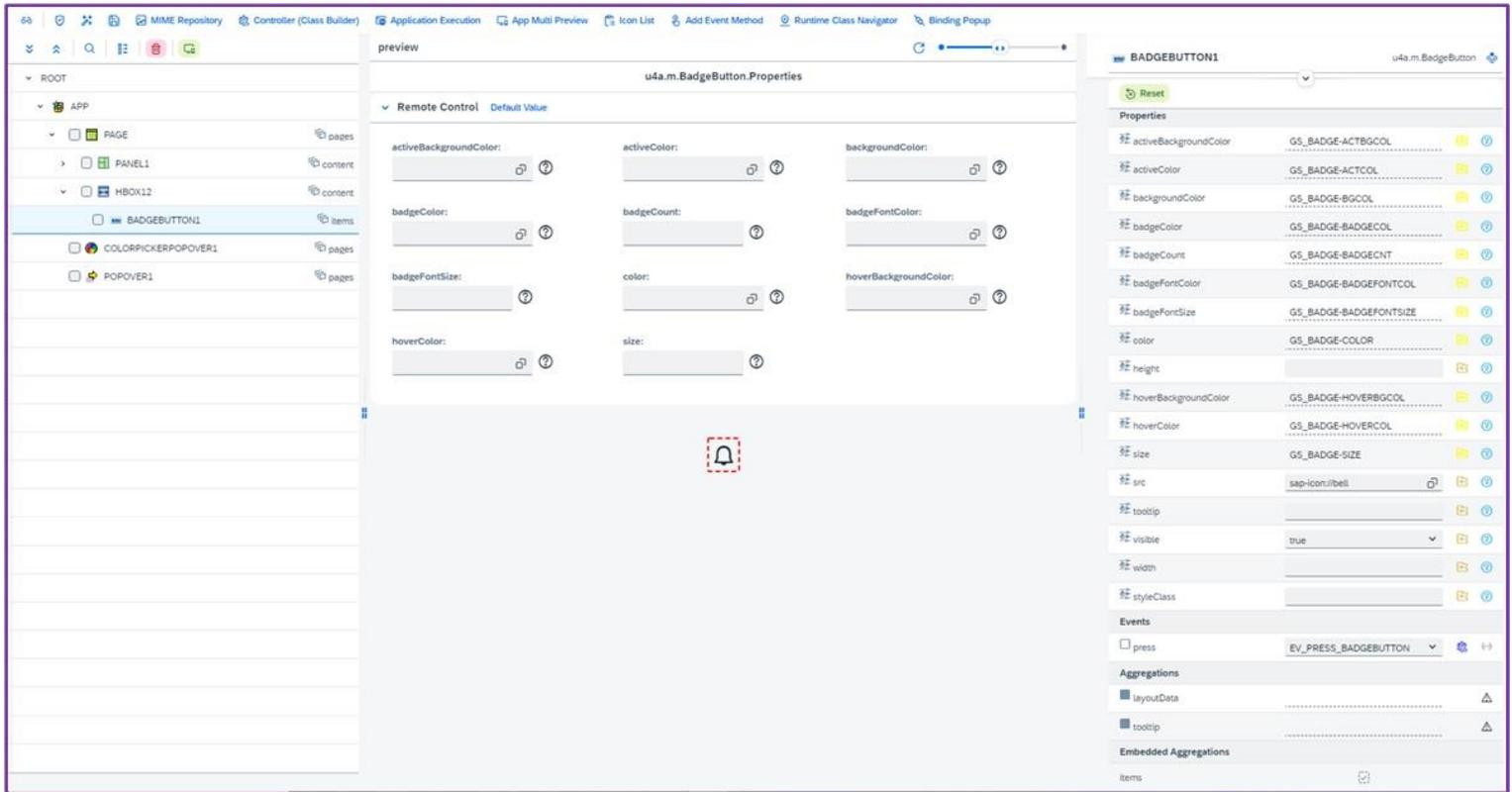
(단 Ifram의 height 또는 width의 값에 따라 정렬이 되지 않는 경우가 있으므로 고려하여 사용 Ex) width : 100%인 경우 left, right 정렬 불가능)

3. Iframe Property - FrameBorder.



▣ 상단 Ifram : `frameBorder : 1`, 하단 Ifram : `frameBorder :0` 으로 설정하였고 Application 실행 하여 하단 Ifram과 비교했을 때 상단 Ifram은 경계선이 생긴 것을 확인.

7.6 BadgeButton



PROPERTIES	MEAN
activeBackgroundColor	Badge Button Click 했을 때 Button 배경 Color 설정
activeColor	Badge Button Click 했을 때 Button Icon Color 설정
backgroundColor	Button 배경 Color 설정
badgeColor	badgeCount 숫자 backgroundColor 설정
badgeCount	Buuton 우상단에 보일 숫자 값 설정 (값을 지정 후 숫자 확인)
badgeFontSize	badgeCount 숫자 Color 설정
badgeFontSize	badgeCount 숫자 Size 설정
color	Button Icon Color 설정
height	Button 세로 높이 설정
hoverBackgroundColor	마우스를 Button 위에 올렸을 때 Button backgroundColor 설정,

hoverColor	마우스를 Button 위에 올렸을 때 Button Icon Color 설정
size	Button Size 설정
src	Button Icon 주소
tooltip	마우스를 Button 위에 올렸을 때 Text Box 생성 여부 및 Text 설정 (Text 미입력시 Text Box 미생성)
visible	Button 보이게 할 지 여부 설정
width	Button 가로 넓이 설정
styleClass	Badge Button 에 적용할 CSS 설정

2) Badge Button Property - backgroundColor, color.

Remote Control Default Value

activeBackgroundColor: ?

activeColor: ?

backgroundColor: ?

badgeColor: ?

badgeCount: ?

badgeFontColor: ?

badgeFontSize: ?

color: ?

hoverBackgroundColor: ?

hoverColor: ?

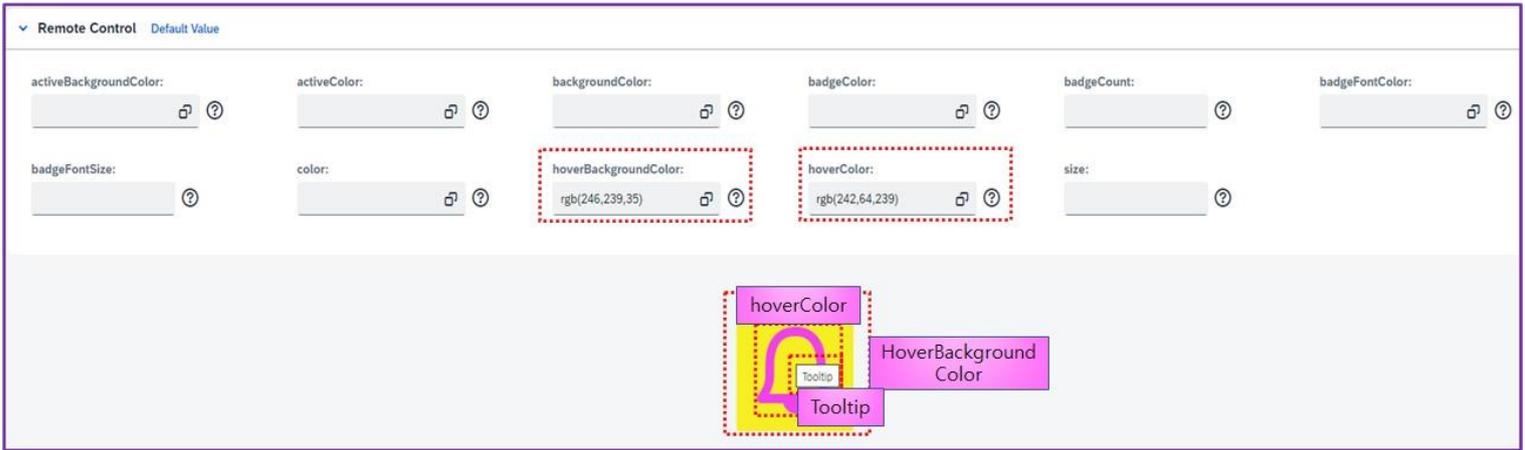
size: ?

Color

Background Color

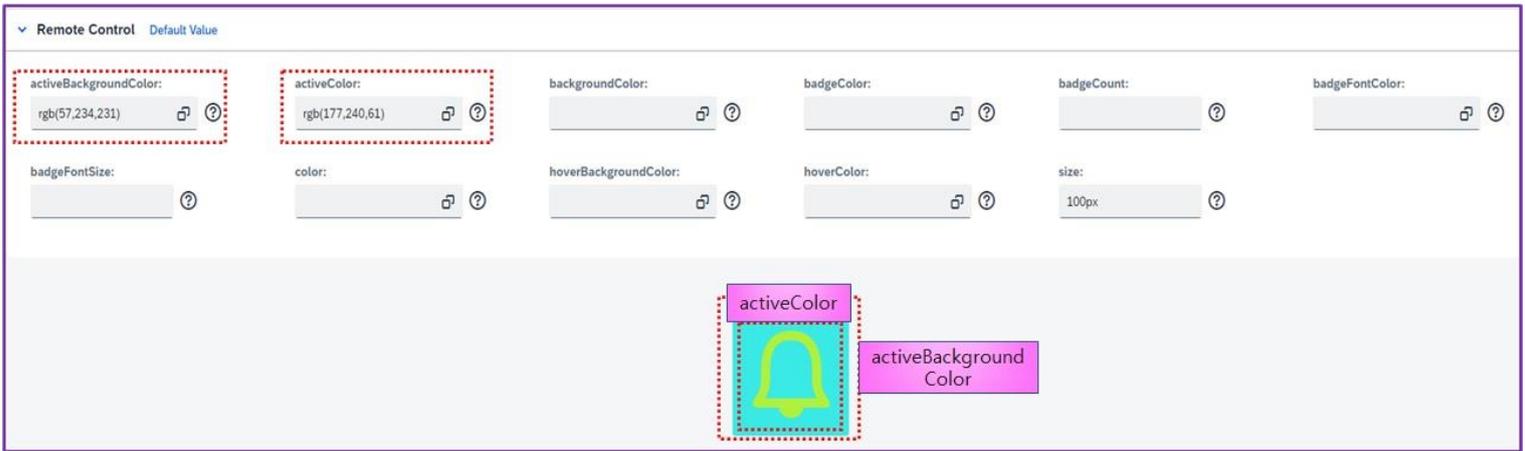
▣ 아무런 Action을 취하지 않았을 때 보이는 Properties.

3) Badge Button Property - hoverBackgroundColor, hoverColor, Tooltip.



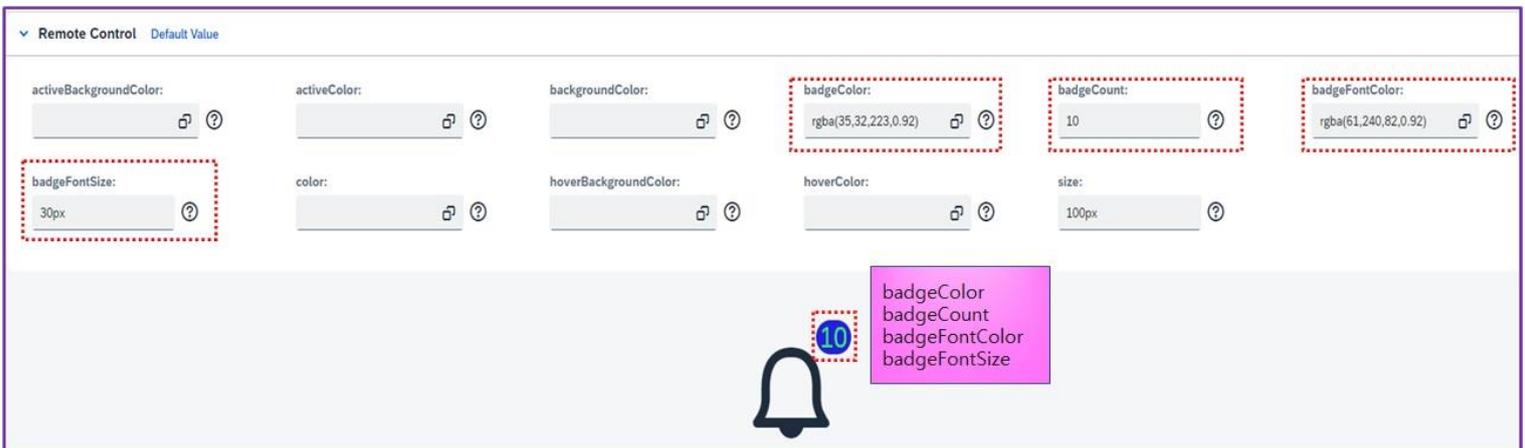
■ Button 위에 마우스를 올렸을 때 보이는 Properties.

4) Badge Button Property - activeBackgroundColor, activeColor.



■ Button을 마우스로 Click 했을 때 보이는 Properties.

5) Badge Button Property - badgeColor, badgeCount, badgeFontColor, badgeFontSize.



▣ badgeCount 과 관련된 Properties.

6) Badge Button Events - Press.

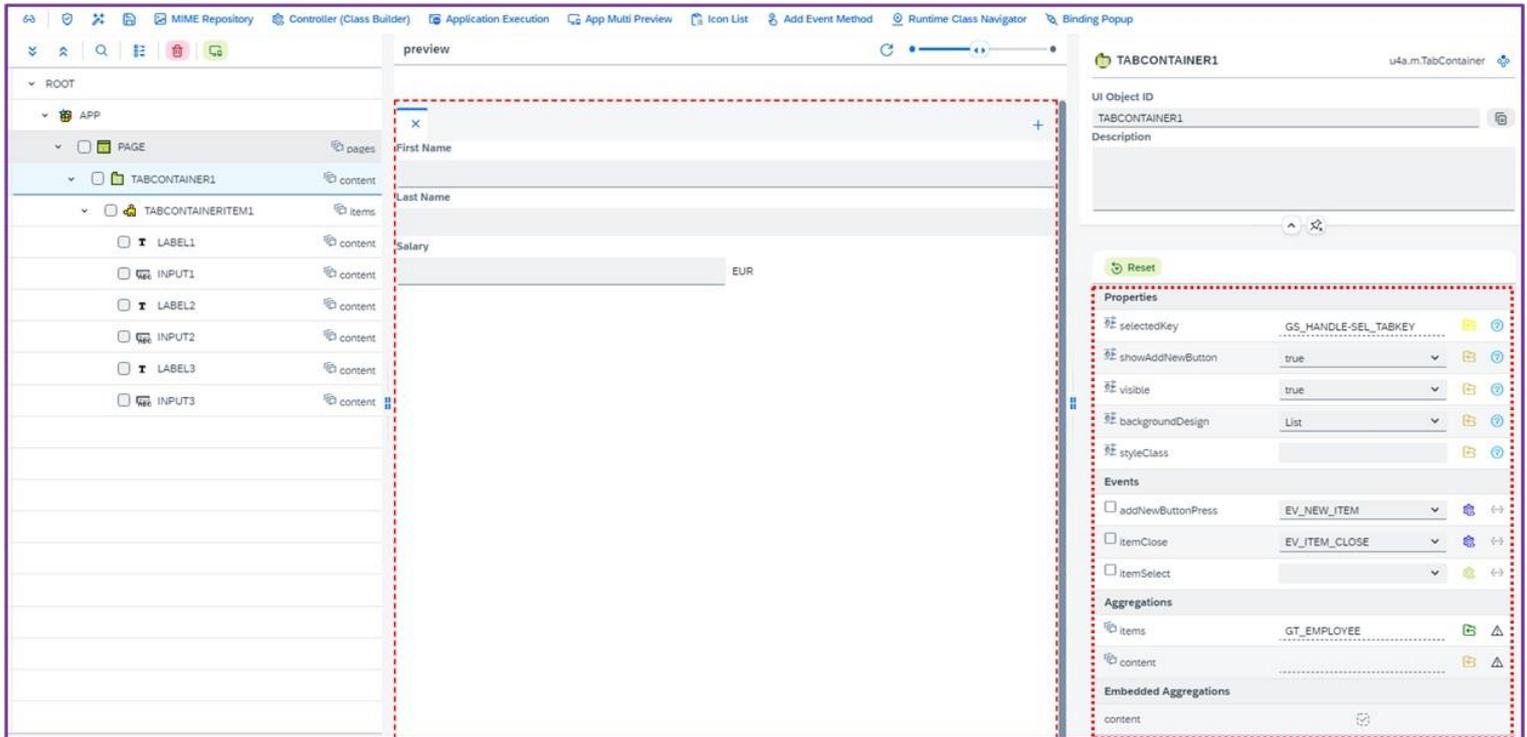
The screenshot displays the 'Remote Control' panel in the U4A IDE. At the top, it shows 'Default Value' for various properties. Below this, there are two rows of property controls, each with a text input field, a lock icon, and a help icon. The properties and their values are:

- activeBackgroundColor: (empty)
- activeColor: (empty)
- backgroundColor: (empty)
- badgeColor: rgba(35,32,223,0.92)
- badgeCount: 10
- badgeFontColor: rgba(61,240,82,0.92)
- badgeFontSize: 30px
- color: (empty)
- hoverBackgroundColor: (empty)
- hoverColor: (empty)
- size: 100px

Below the property list is a visual representation of a 'BadgeButton'. It consists of a white rounded rectangle with the text 'BadgeButton Click!' inside. Above the button is a blue circular badge containing the number '10'. The entire button and badge are enclosed in a red dashed rectangular border.

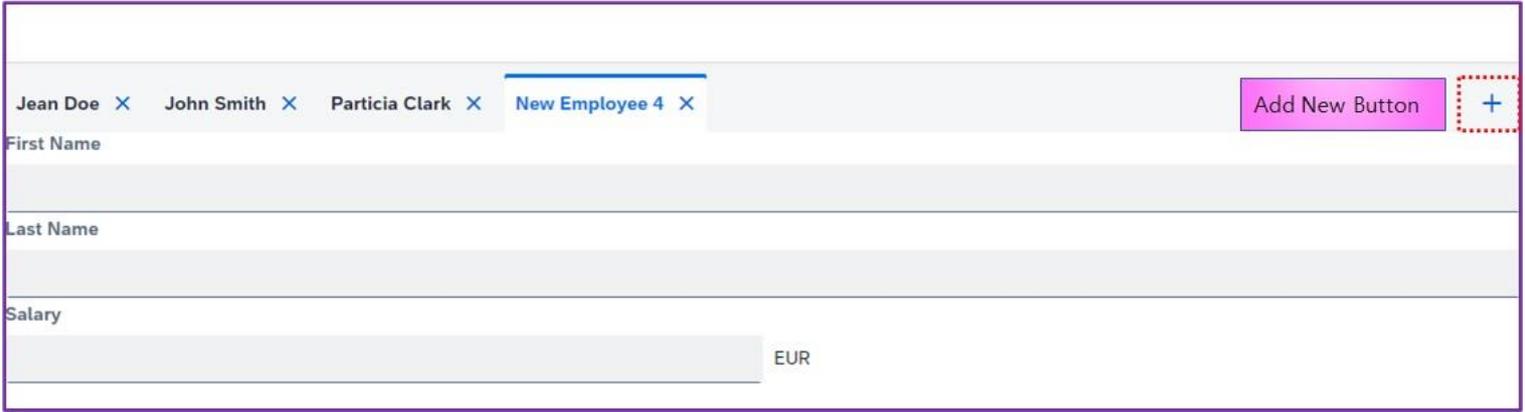
▣ Badge Button Click 할 때 동작하는 Event.

7.7 TabContainer



PROPERTIES	MEAN
selectedKey	처음 실행 시 선택 되어질 TabContainerItem의 key 값을 입력
showAddNewButton	TabContainer에 추가 버튼을 보일지 여부, true 이면 TabContainer 우측에 "+" 버튼이 생성 된다.
visible	TabContainer 보이게 할지 여부 설정
styleClass	TabContainer에 적용할 CSS 설정

2. TabContainer Property - ShowAddNewButton.

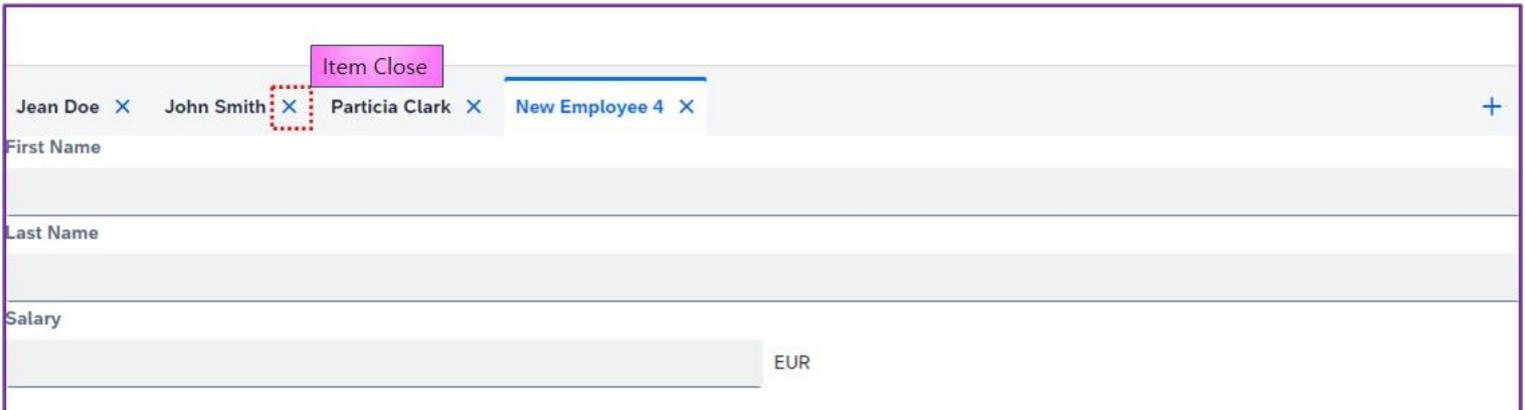


▣ "+" Button을 보이게 할 지 여부 결정 .

3. TabContainer Events - addNewButtonPress.

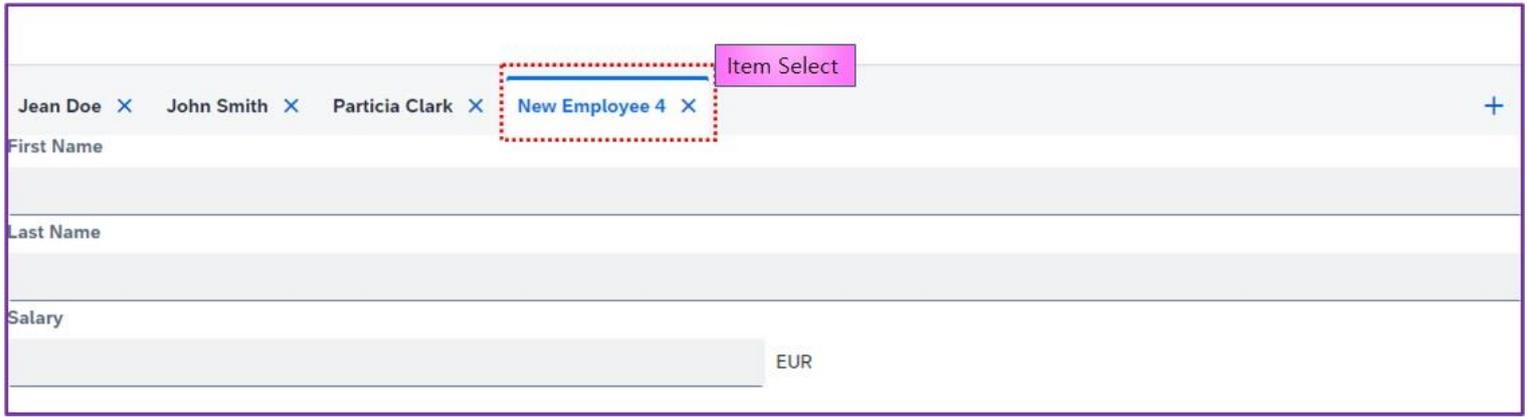
▣ 위 Image의 "+" Button을 Click하면 addNewButtonPress Event 발생.

4. TabContainer Events - itemClose.



▣ Tab Item의 'x' Button Click 하면 itemClose Event 발생 .

4. TabContainer Events - itemSelect.



■ Tab Item을 Click 하면 itemSelect Event 발생 .

7.8 SessionWorker

■ 목적 : U4A에서 제공하는 UI Object 중 SESSIONWORKER은 정해진 시간마다 Event를 작동시킬 수 있다.

1. SESSIONWORKER은 APP의 하위 항목에서만 만들 수 있기 때문에 APP 오른쪽 클릭 후 pages내에 생성한다.

UI Object Select - App

Aggregation Name: pages

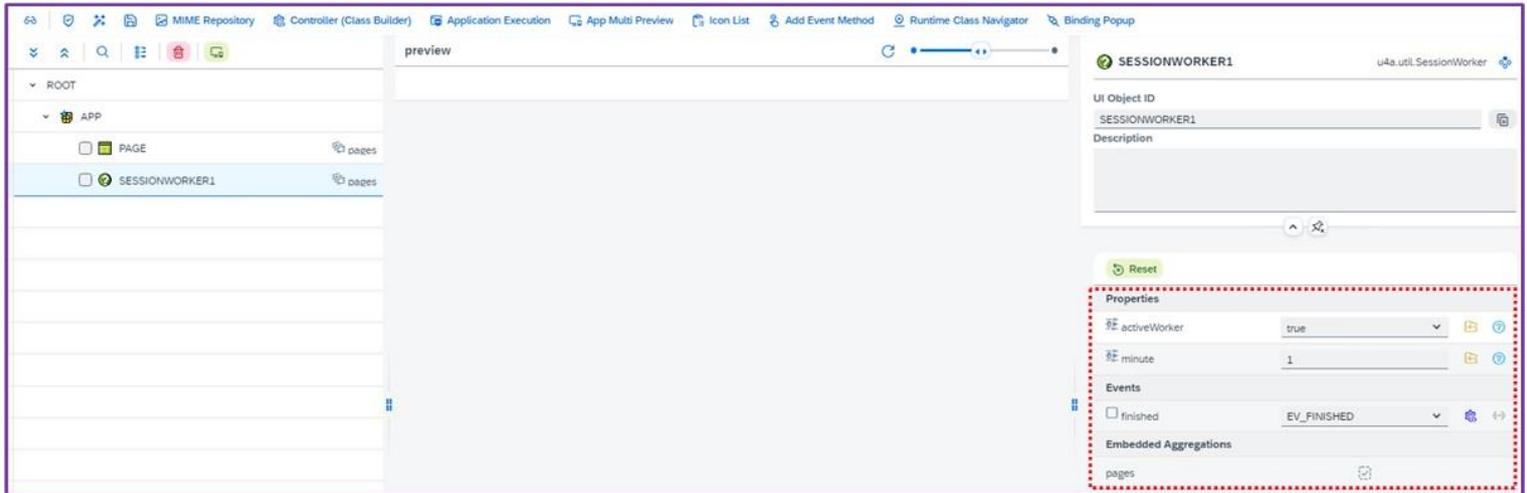
Generated Cnt: 1 Max : 100

UI Object: SessionWorker

Img	UI Object	UI Object(Fullname)	UI Key	UI Info
	SessionWorker	u4a/utli/SessionWorker	U001782	?

Confirm Cancel

2. SESSIOWOKER의 Attributes는 하기와 같이 구성되어 있다.



Attributes	Means
activeWorker	SEESIONWORKER의 작동 유무를 설정 한다.
minute	분단위 기준으로 Event를 작동 시킨다.(정수로만 값을 받을 수 있다.)
finished(Events)	Properties에서 설정한 기준에 따라 작동하는 Event이다.

3. SEESIONWORKET의 Event를 만든다.

[첨부ABAP SOURCE (EV_FINISHED)]

METHOD EV_FINISHED.

"메세지 호출

CALL METHOD /U4A/CL_UTILITIES=>M_MESSAGES_TOAST

EXPORTING

IO_VIEW = /U4A/IF_SERVER~AR_VIEW

" [U4A] UI Element Super Class

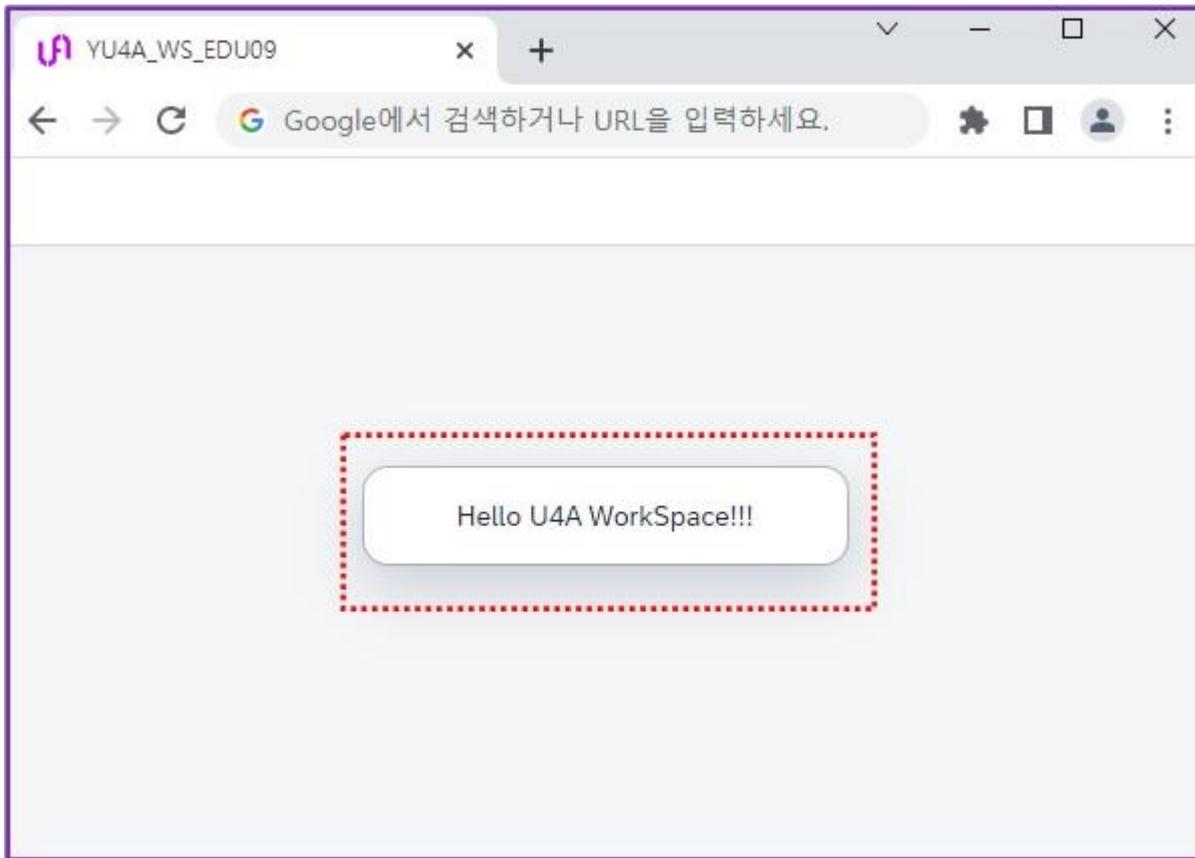
I_MSGTX = 'Hello U4A WorkSpace!!!'.

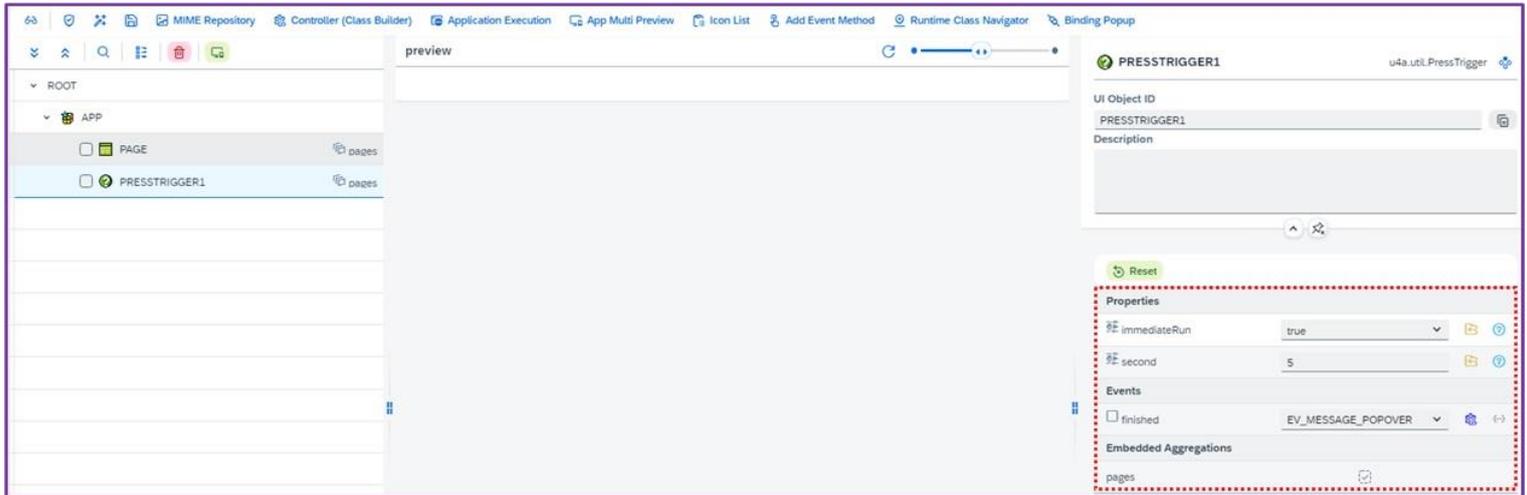
" MESSAGE TEXT

ENDMETHOD.

4. Web에서 지정한 시간마다 Event가 작동하는 것을 확인 할 수 있다.

U4A IDE HELP





3. Presstrigger의 Event를 만든다.

[첨부ABAP SOURCE (EV_MESSAGE_POPOVER)]

METHOD EV_MESSAGE_POPOVER.

"메세지 호출

CALL METHOD /U4A/CL_UTILITIES=>M_MESSAGETOAST

EXPORTING

IO_VIEW = /U4A/IF_SERVER~AR_VIEW

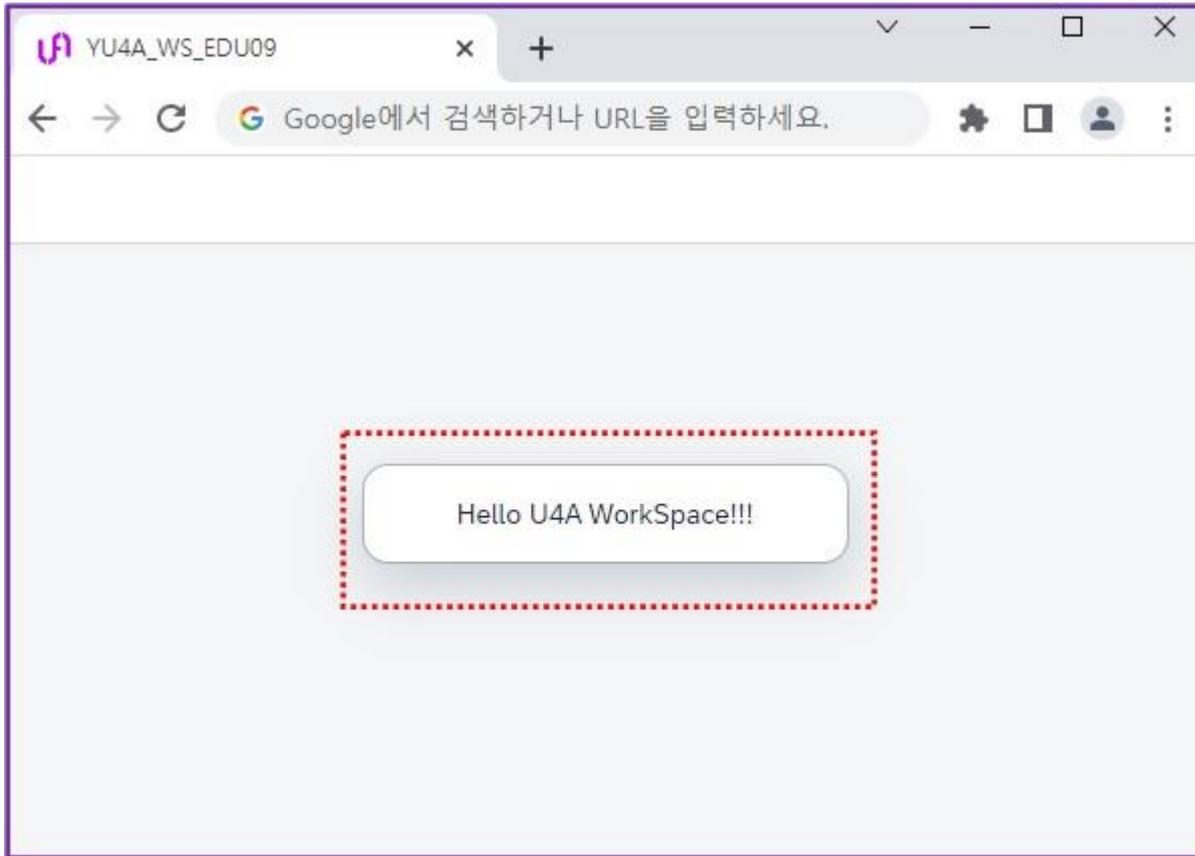
" [U4A] UI Element Super Class

I_MSGTX = 'Hello U4A Workspace!!!'.

" MESSAGE TEXT

ENDMETHOD.

4. Web에서 아무런 입력 없으면 지정 시간 후 Event가 발생하는 것을 확인 할 수 있다.

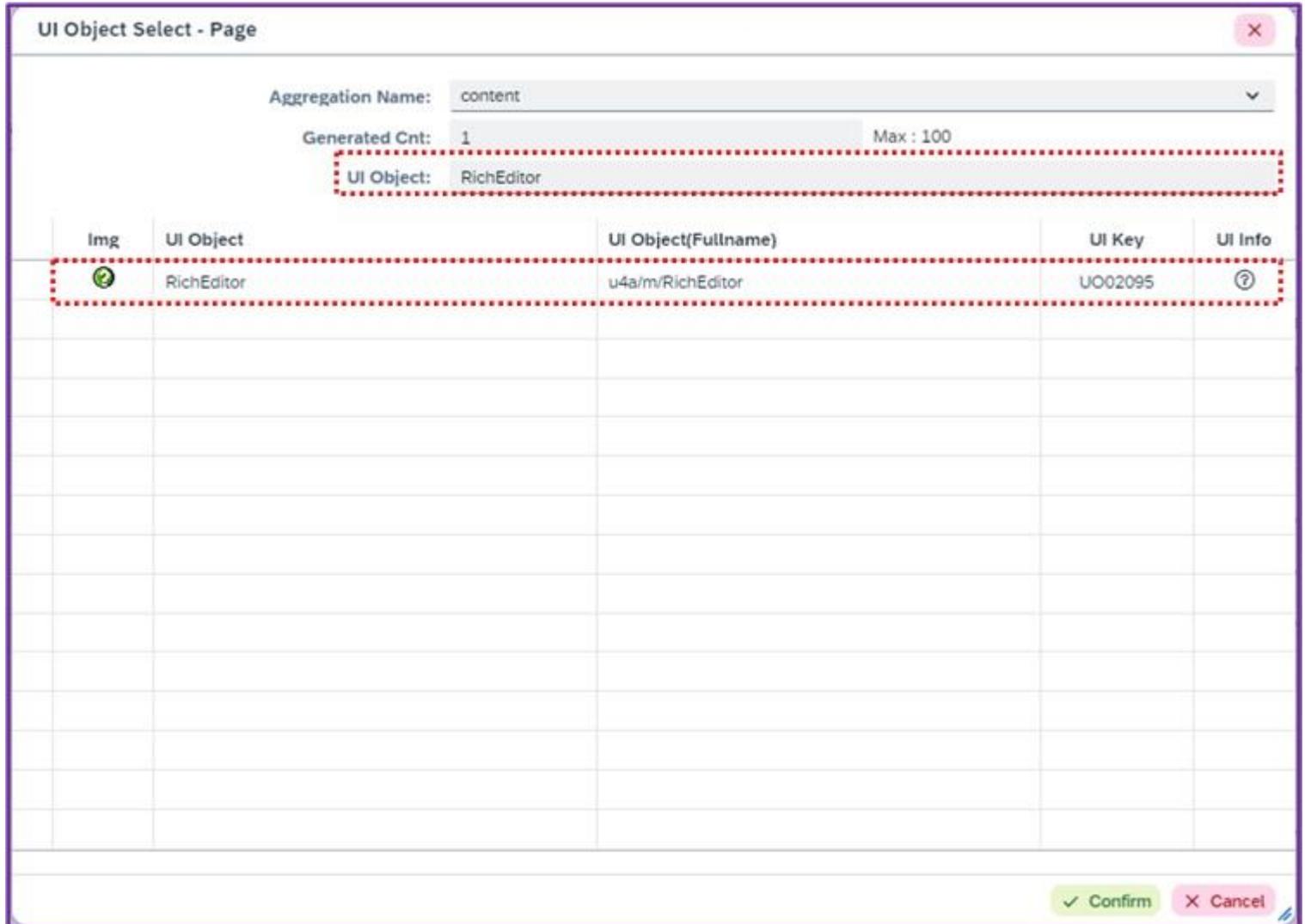


 Event 한번 작동 이후 Presstrigger의 IMMEDIATERUN은 false로 바뀌기 때문에 Event 후 IMMEDIATERUN을 다시 true로 바꿔주면 계속 작동한다.

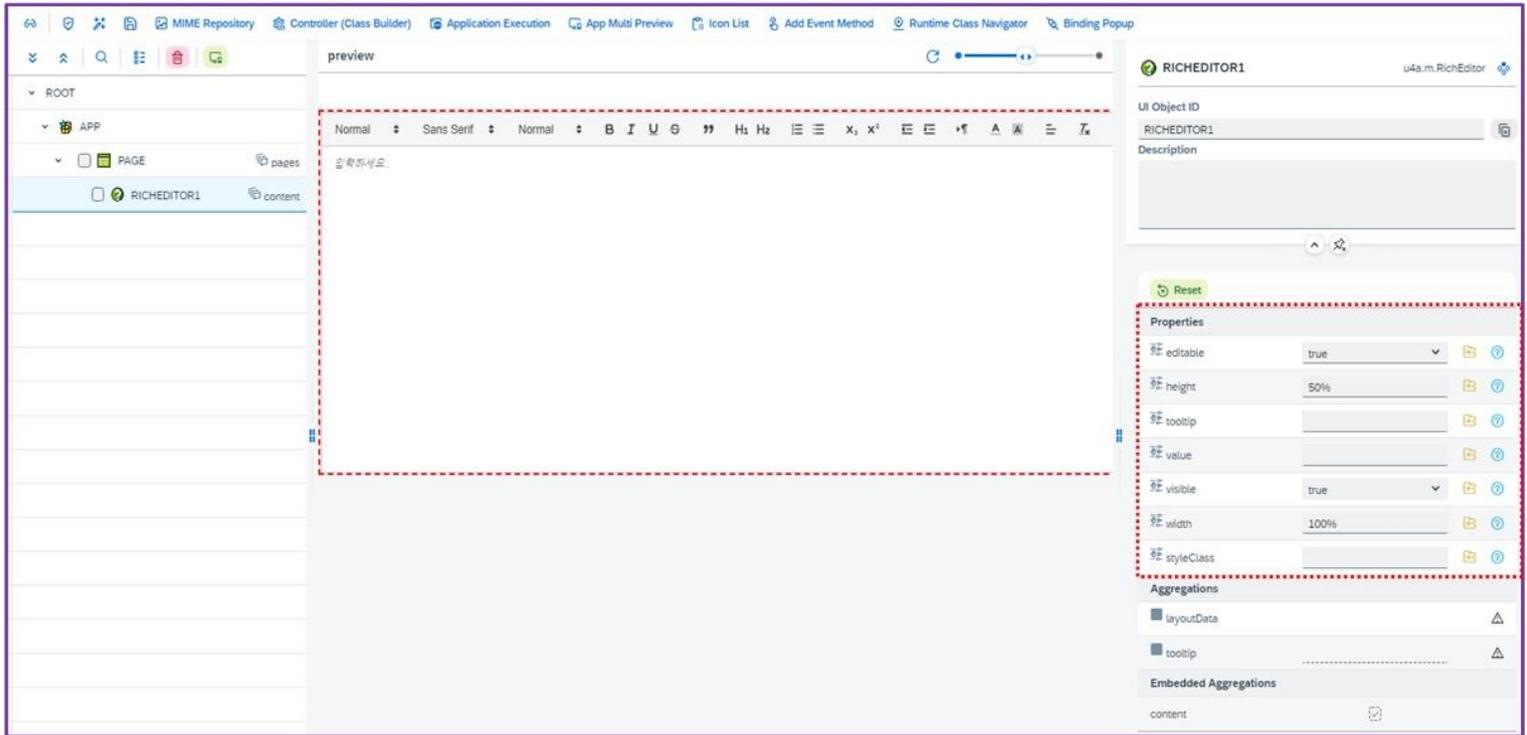
7.10 Richeditor

■ 목적 : Richeditor는 Text의 design을 바꿀 수 있는 TextArea UI로 UI5에서 제공하는 RichTextEditor보다 더 많은 기능을 제공한다.

1. 원하는 Aggregaion을 선택하고 RichEditor를 생성한다.

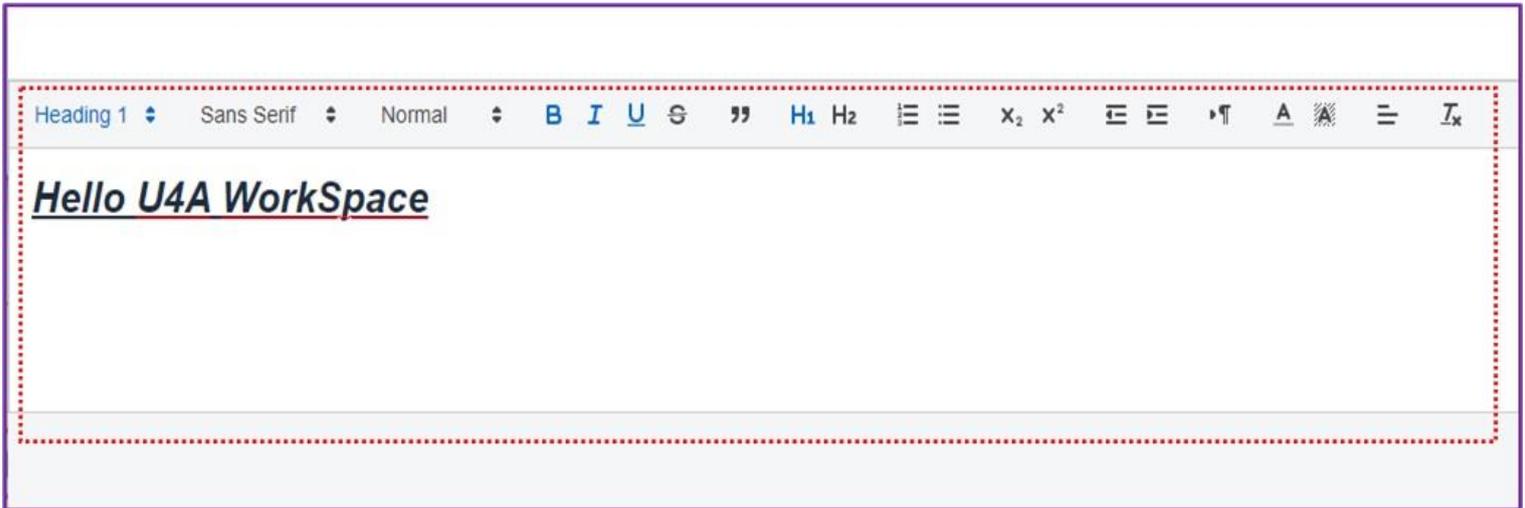


2. Richeditor를 생성하면 아래와 같이 생긴 UI Object가 나타난다.

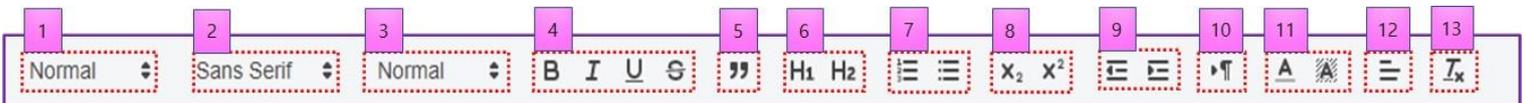


Attributes	Means
editable	text 입력하는 곳을 수정 가능/불가능으로 바꿀 수 있다.
height	UI Object의 높이를 조절한다
value	입력한 text가 담기게 된다.
visible	UI Object가 보이거나 보이지 않게 만든다.
width	UI Object의 가로폭을 조절한다.

3. Web화면에서 UI Object가 생성된 것을 확인 할 수 있다.



4. Richeditor Icon의 기능



- 1) 글자의 크기와 굵기 변경
- 2) 글자 font 변경
- 3) 글자 크기 변경
- 4) 글자 굵기, 기울기, 밑줄, 취소선
- 5) 글자를 주석 형태로 변경
- 6) 제목형태로 변경 1)의 Heading1과 Heading2과 같은 형태
- 7) 맨 앞에 줄마다 번호 또는 점 추가
- 8) 일반글자보다 하단에 작게 만들거나 상단에 작게 만듦
- 9) 줄의 들여쓰기/내어쓰기
- 10) 일반문자 + 특수문자 ([ex.1234!@#\\$](#)) 구조일 경우 특수문자 + 일반문자([ex.\\$#@!1234](#))로 변경
- 11) 글자 색 및 글자 배경색 변경
- 12) 좌,우,가운데 정렬
- 13) 글자에 적용된 모든 기능 제거

7.11 CountdownTimer

UI Object Select - Page

Aggregation Name: content

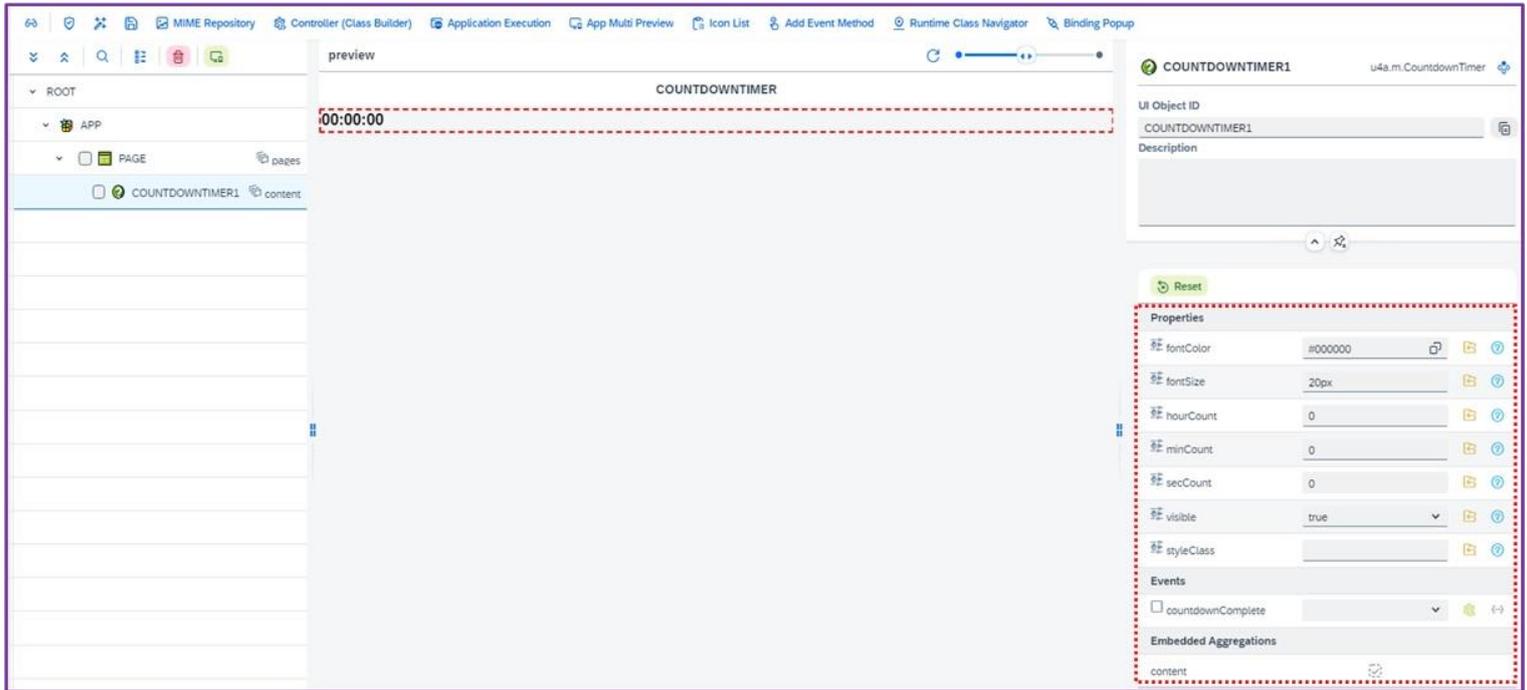
Generated Cnt: 1 Max : 100

UI Object: CountdownTimer

Img	UI Object	UI Object(Fullname)	UI Key	UI Info
	CountdownTimer	u4a/m/CountdownTimer	U001779	

Confirm Cancel

2. CountdownTimer을 생성하면 아래와 같이 생긴 UI Object가 나타난다.



Attributes	Means
fontColor	CountdownTimer의 색을 바꾼다
fontSize	CountdownTimer의 숫자 크기를 바꾼다.
hourCount	시간 단위로 변경 할 수 있다.
minCount	분 단위로 변경 할 수 있다.
secCount	초 단위로 변경 할 수 있다.
visible	CountdownTimer을 보이기/숨김 설정 할 수 있다.
countdownComplete (Events)	Countdown이 끝나면 Event를 작동시킨다.

3. Event를 만들고 Web화면에서 Countdown이 끝나면 Event가 작동하는 것을 확인 할 수 있다.

[첨부ABAP SOURCE (EV_COUNT_DOWN)]

METHOD EV_COUNT_DOWN.

"메세지 출력

```
CALL METHOD /U4A/CL_UTILITIES=>M_MESSAGE_TOAST
```

```
EXPORTING
```

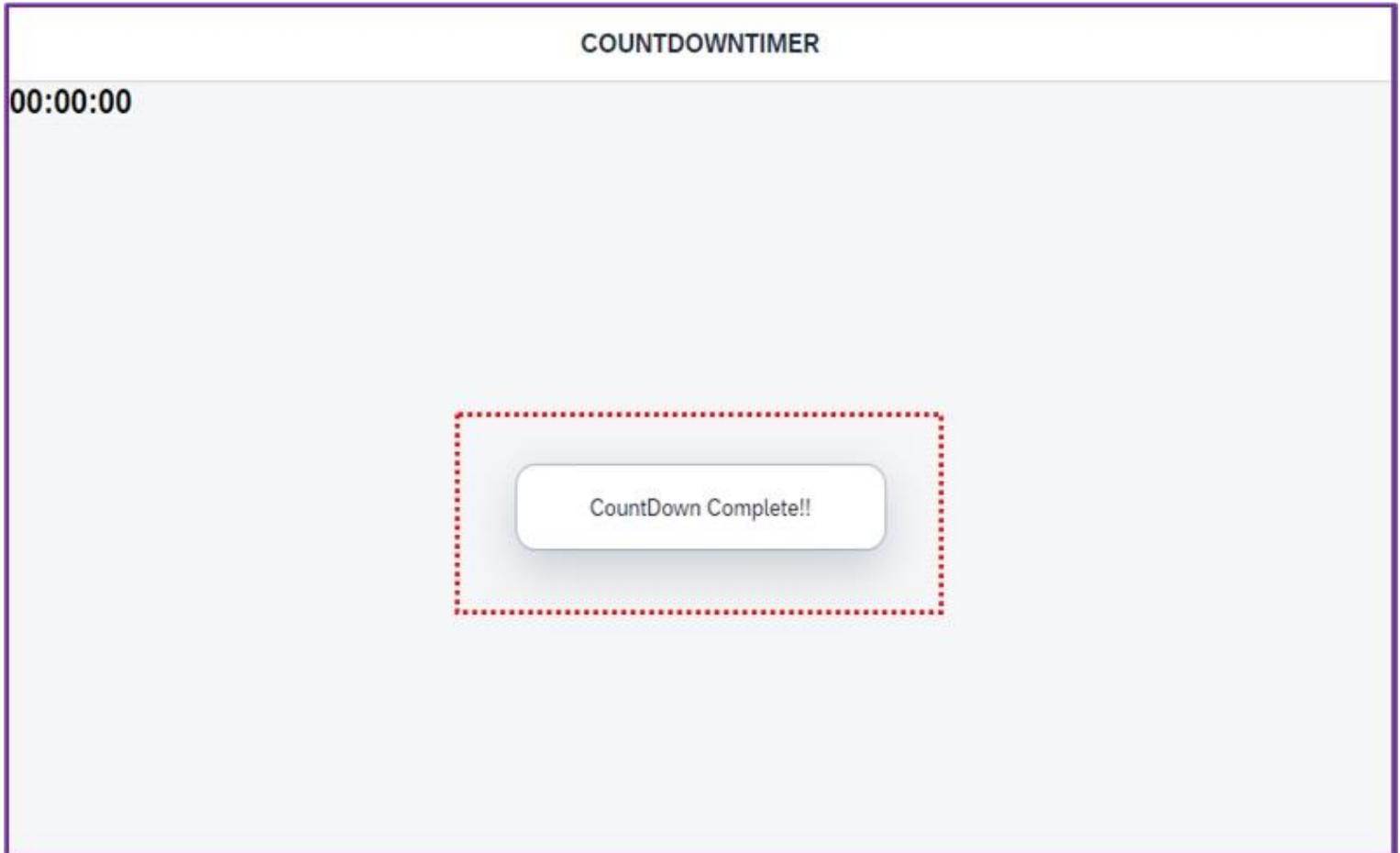
```
IO_VIEW = ME->/U4A/IF_SERVER~AR_VIEW
```

```
" [U4A] UI Element Super Class
```

```
I_MSGTX = 'CountDown Complete!!'.
```

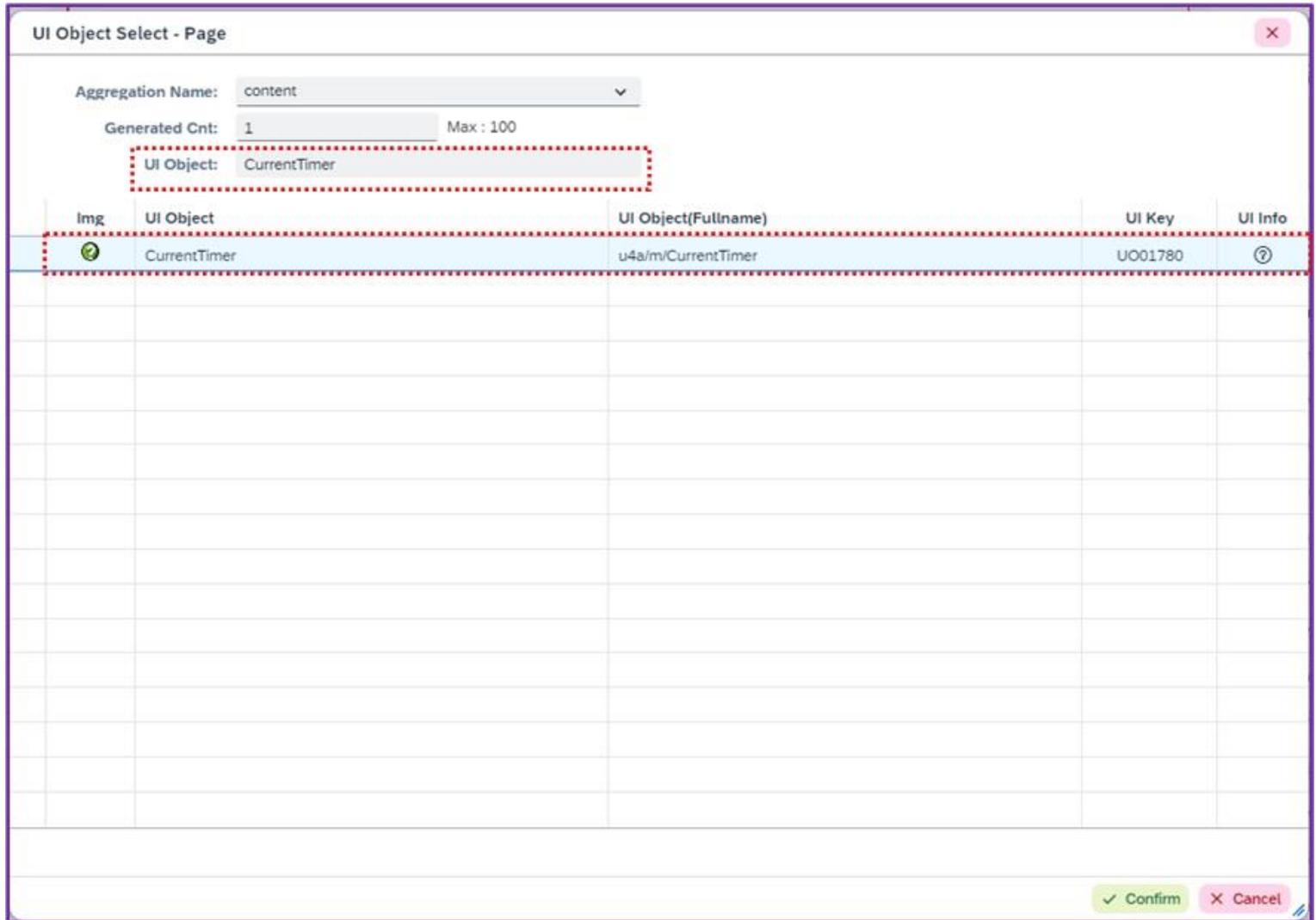
```
" MESSAGE TEXT
```

```
ENDMETHOD.
```

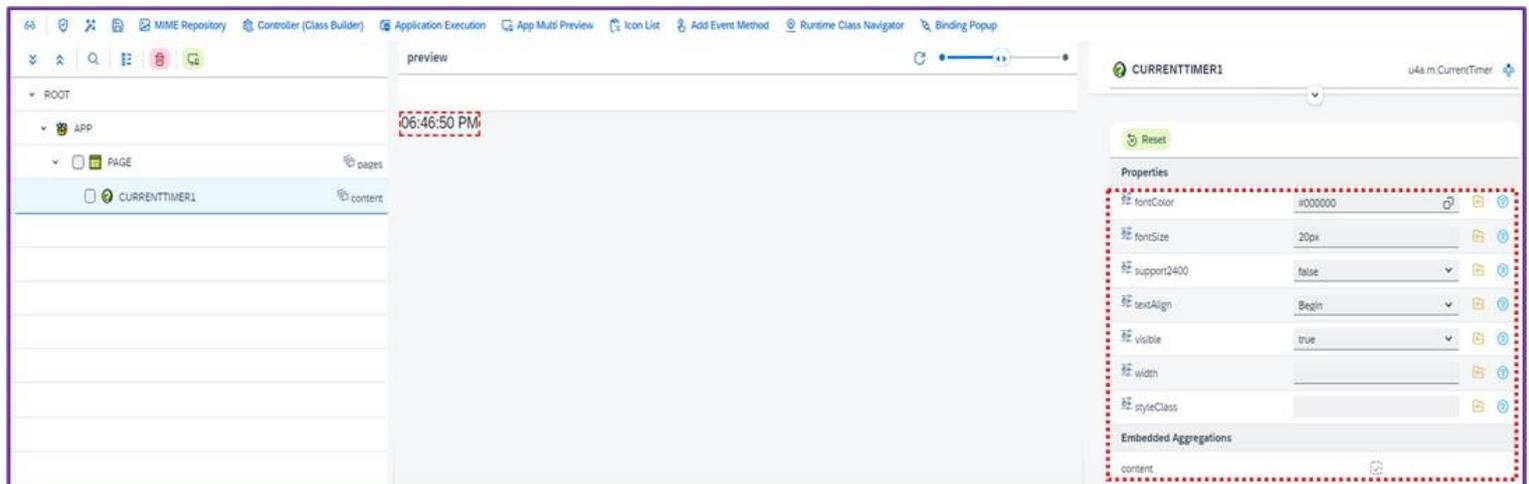


시간 설정 후 시스템을 로딩 하는 시간이 있기 때문에 실제 UI Object가 보여주는 시간과 2초 정도 차이가 날 수 있음.

7.12 CurrentTimer



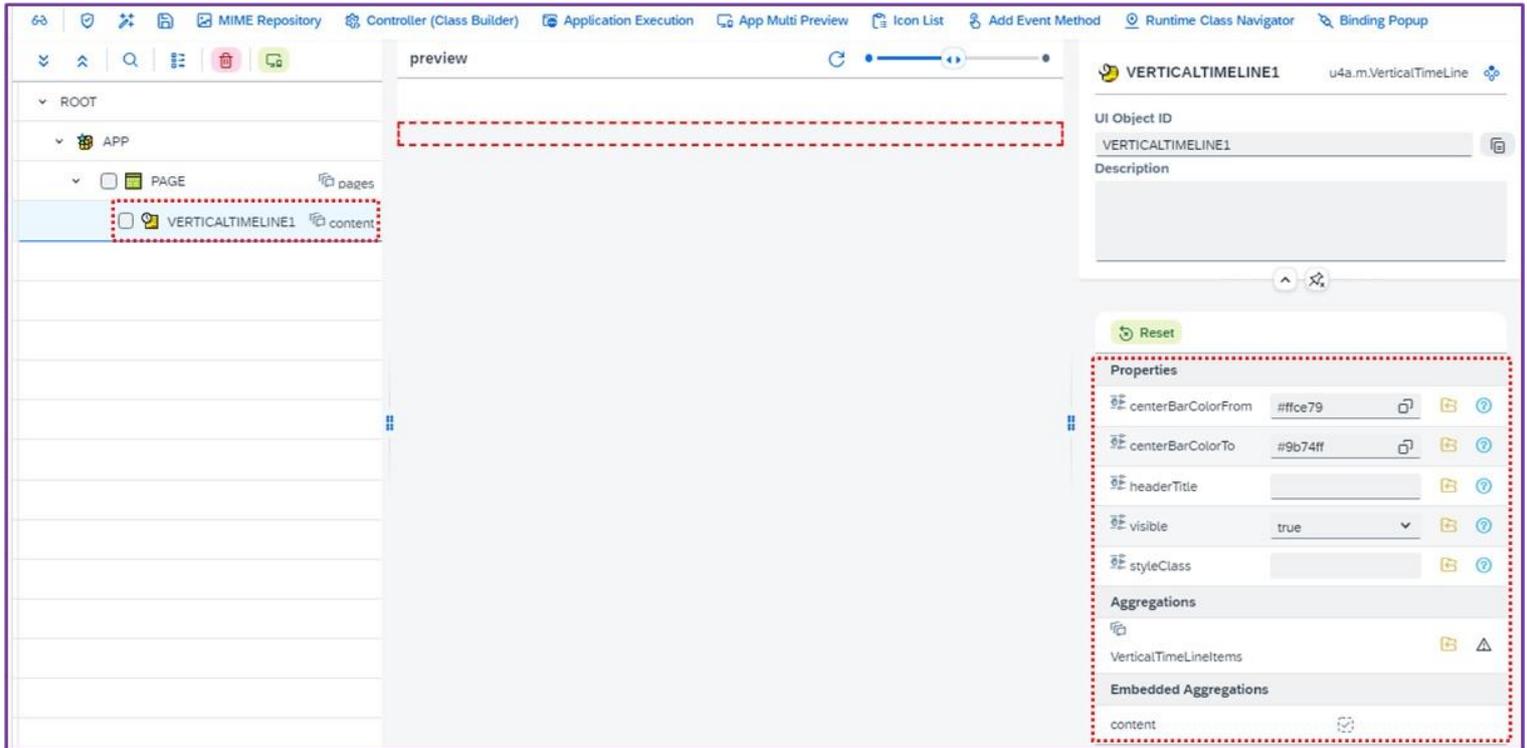
2. CURRNTTIMER를 생성하면 아래와 같이 생긴 UI Object가 나타난다.



Attributes	Means
fontColor	CurrentTimer의 색을 바꾼다
fontSize	CurrentTimer의 크기를 바꾼다.
support2400	false : 12시간을 기준으로 AM/FM 표기 true : 24시간을 기준으로 표기
textAlign	현재 시간의 위치 변경
visible	CurrentTimer을 보이기/숨김 설정 할 수 있다.
width	현재 시간의 폭 변경

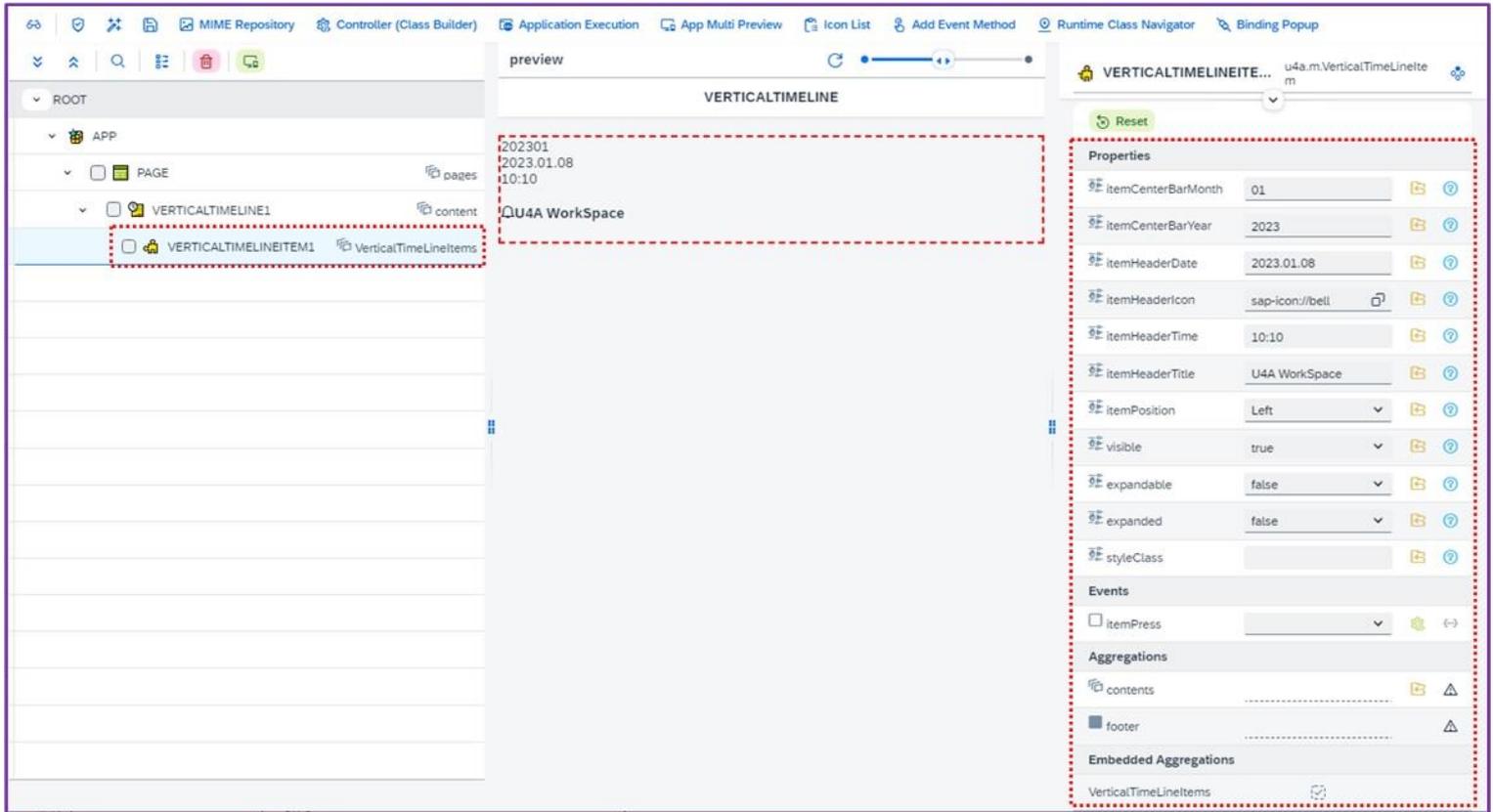
3. Web화면에서 CURRENTTIMER가 생성된 것을 확인 할 수 있다.





Attributes	Means
centerBarColorFrom	VerticalLine의 Gradient color의 start 색상
centerBarColorTo	VerticalLine의 Gradient color의 end 색상
headerTitle	VerticalTimeline UI의 최상단 제목
visible	VerticalTimeline를 보이기/숨김 설정

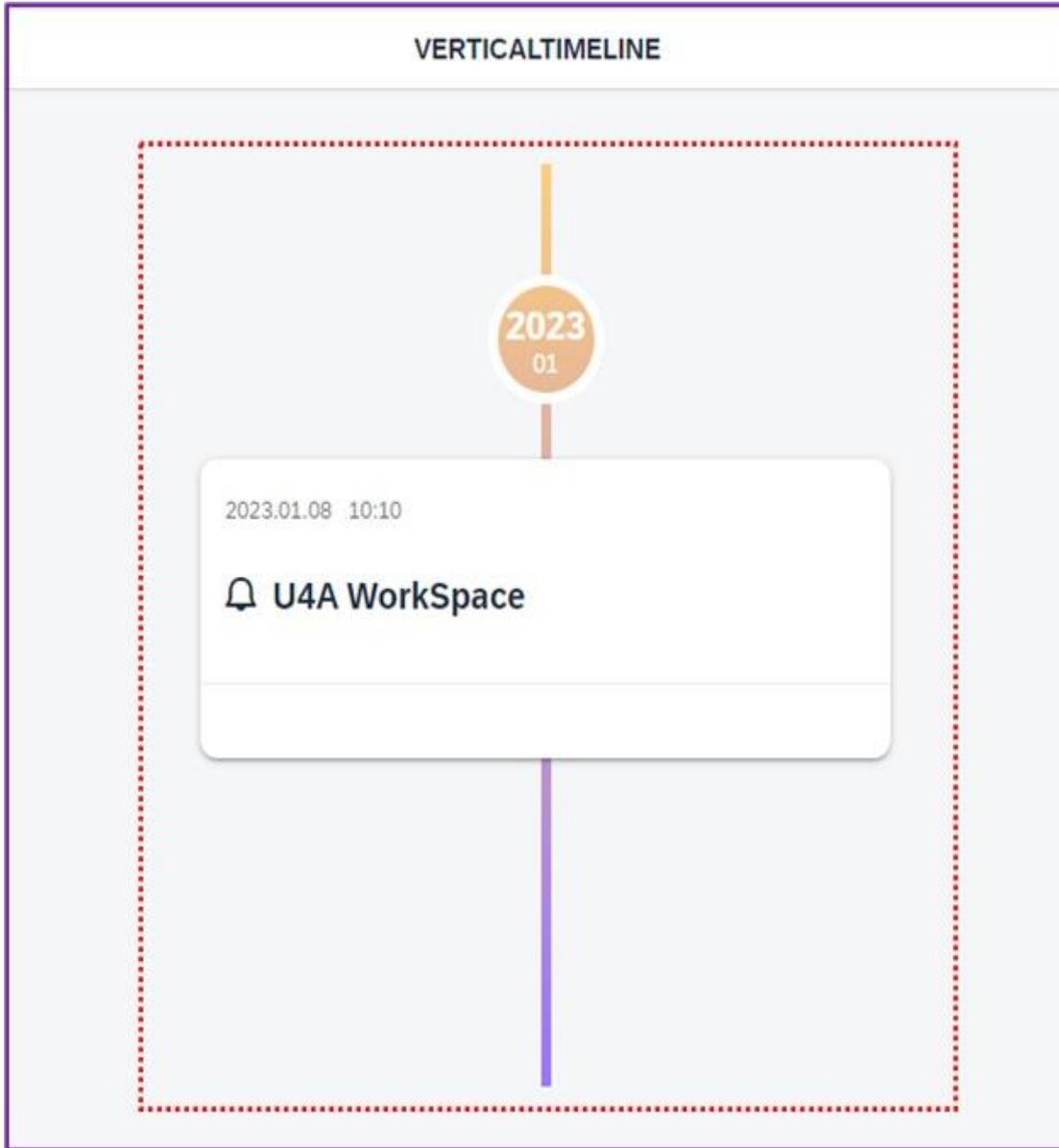
3. VERTICALTIMELINE에 연대를 입력 할 수 있는 UI Object (VERTICALTIMELINEITEM)를 추가 한다.



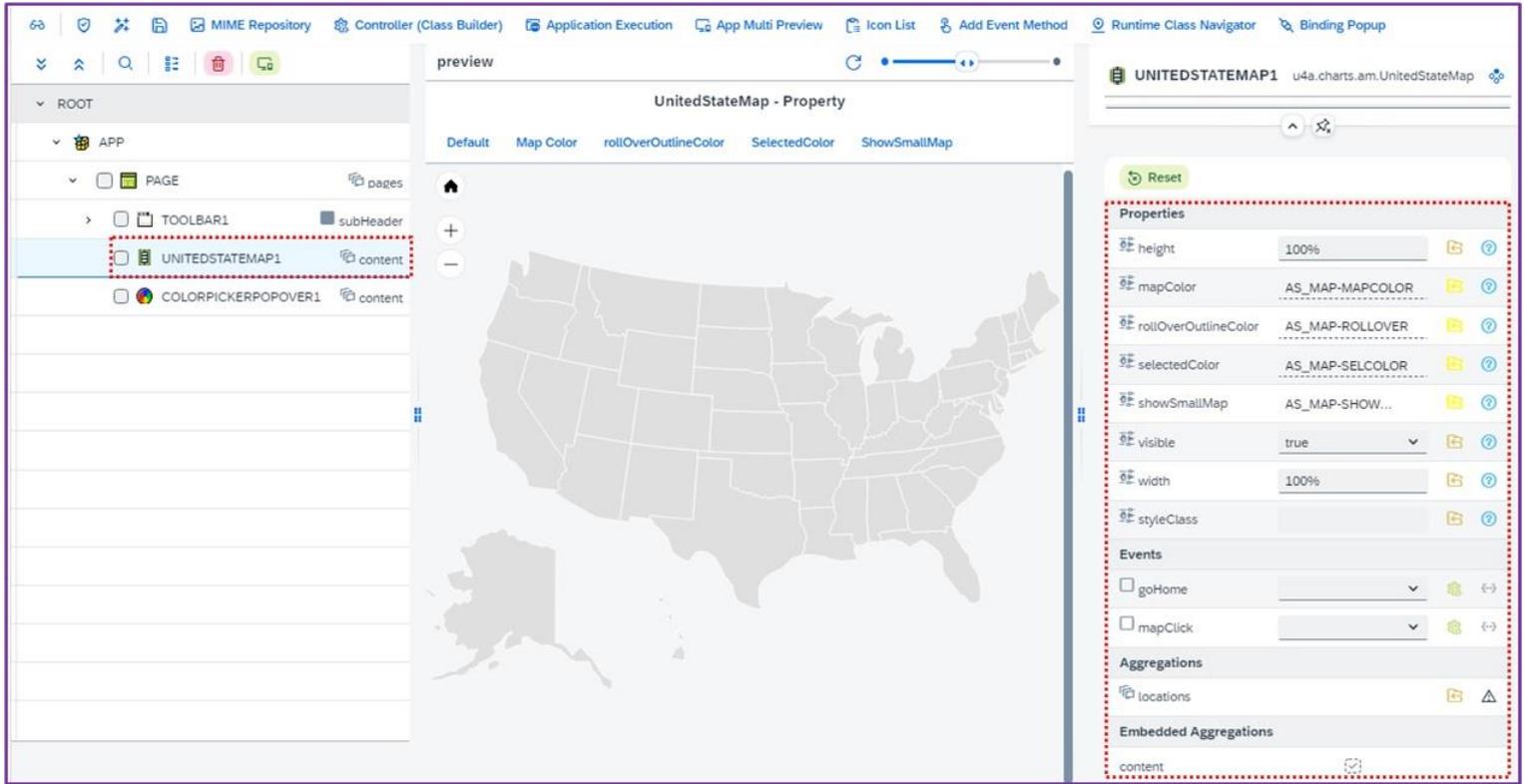
Attributes	Means
itemCenterBarMonth	VerticalTimeline 가운데 라인에 위치한 원형 내의 하단부분에 작은 글씨로 표시된다.
itemCenterBarYear	VerticalTimeline 가운데 라인에 위치한 원형 내의 상단 부분에 큰 글씨로 표시된다.
itemHeaderDate	VerticalTimelineItem의 상단 영역에 날짜 등을 표기할 수 있다.
itemHeaderIcon	itemHeaderTitle 영역의 좌측에 Icon을 표시한다.
itemHeaderTime	VerticalTimelineItem의 상단 영역에 시간 등을 표기할 수 있다.
itemHeaderTitle	VerticalTimelineItem의 상단 영역에 제목 등을 표기할 수 있다.
itemPosition	화면상에서 VerticalTimeline의 가운데 라인 영역을 기준으로 VerticalTimelineItem의 위치를 지정한다.
visible	VerticalTimelineItem를 보이기/숨김 설정

expandable	VerticalTimelineItem의 접기/펼치기 기능 사용 여부
expanded	true: VerticalTimelineItem 가 펼쳐진 상태로 Display false: VerticalTimelineItem 가 닫혀진 상태로 Display

4. Web에서 VERTICALTIMELINE이 조회된다.

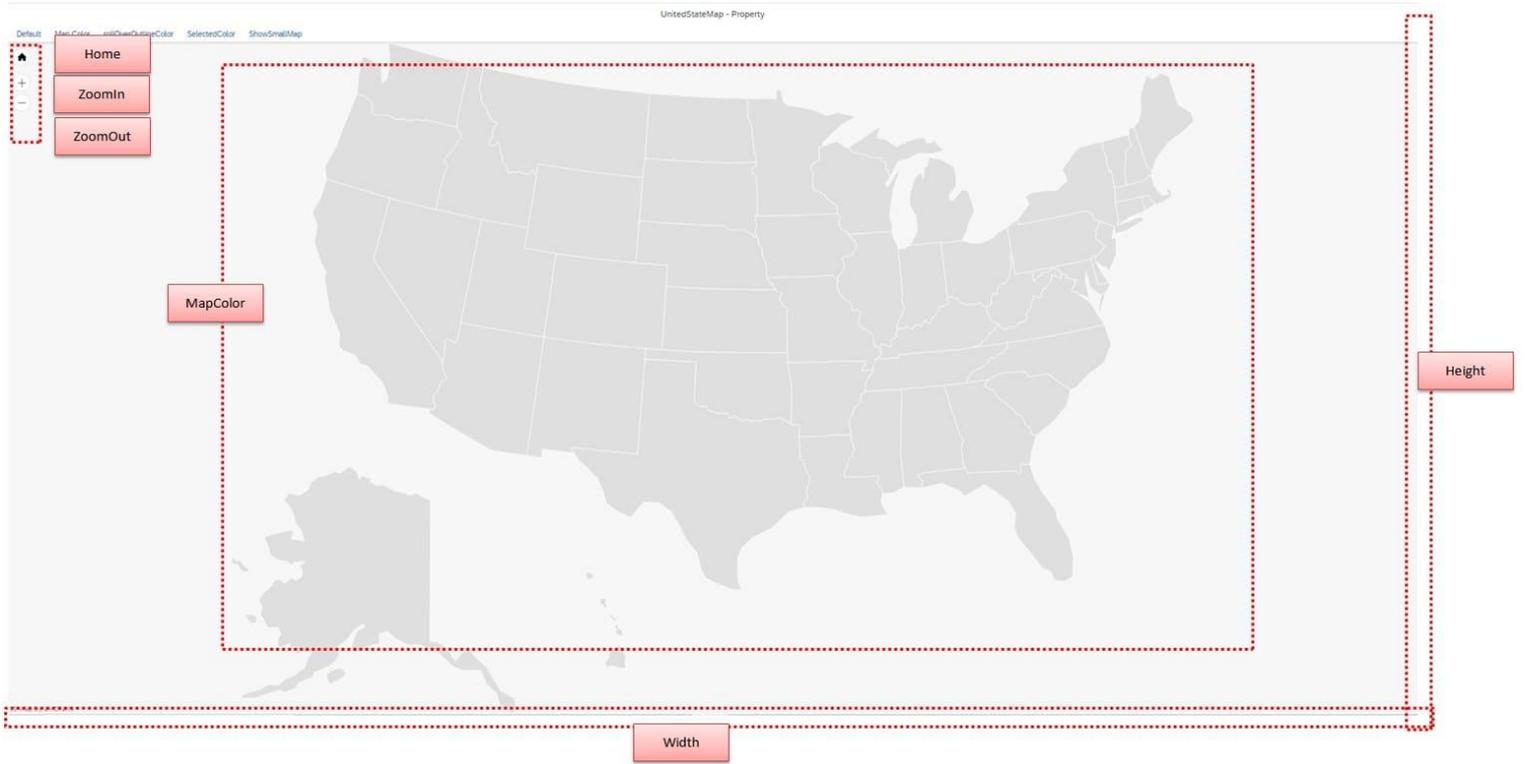


7.14 UnitedStateMap

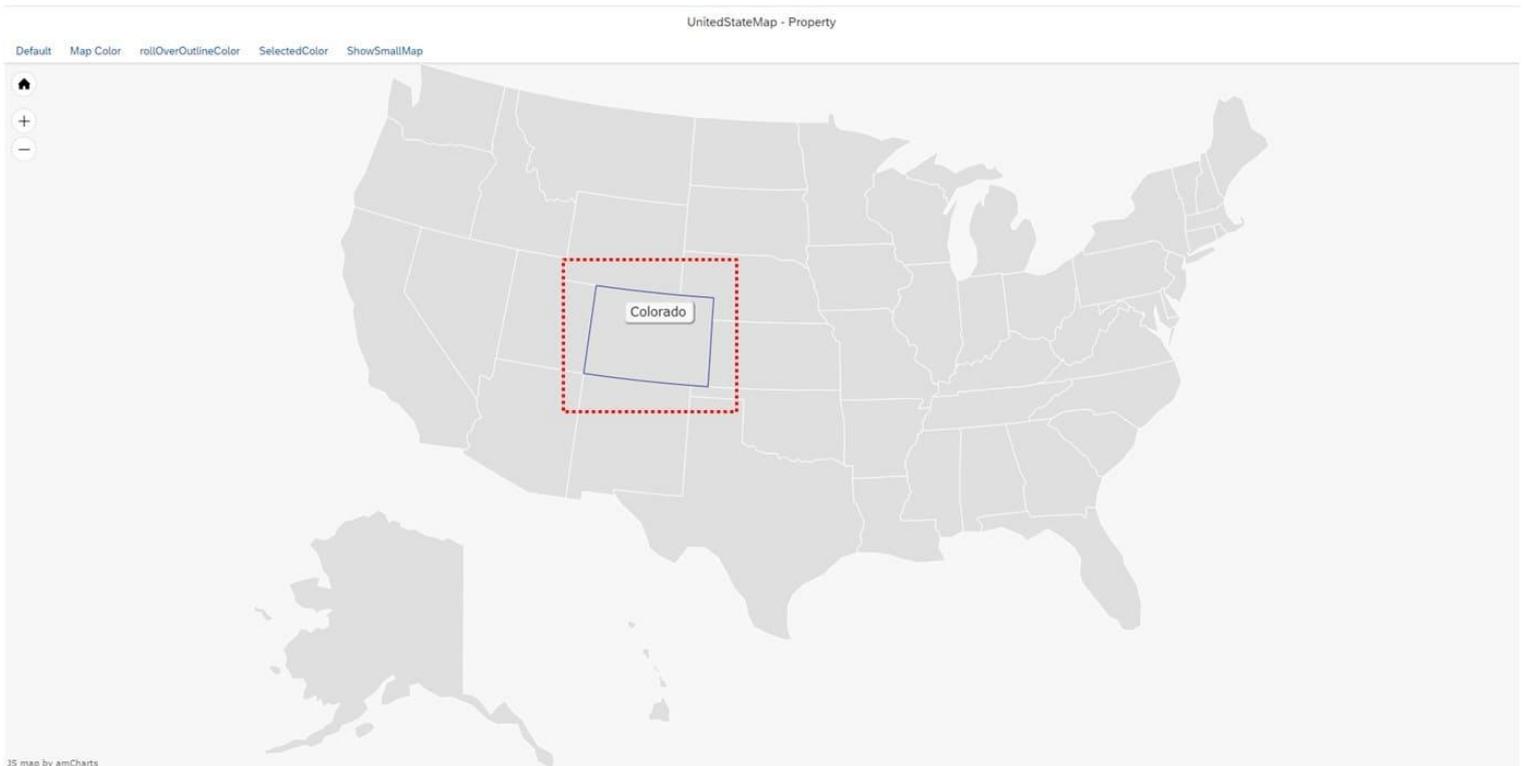


PROPERTIES	MEAN
height	UnitedStateMap 세로 높이
mapColor	UnitedStateMap UI Color
rollOverOutlineColor	지역 선택 했을 때 표시 되는 테두리 Color
selectedColor	지역 선택 했을 때 표시 되는 Color
showSmallMap	우측 상단에 축소된 형태의 UnitedStateMap 생성
visible	UnitedStateMap을 보여줄 지 여부
width	UnitedStateMap 가로 넓이
styleClass	UnitedStateMap 적용 할 CSS

2. UnitedStateMap 실행 화면.

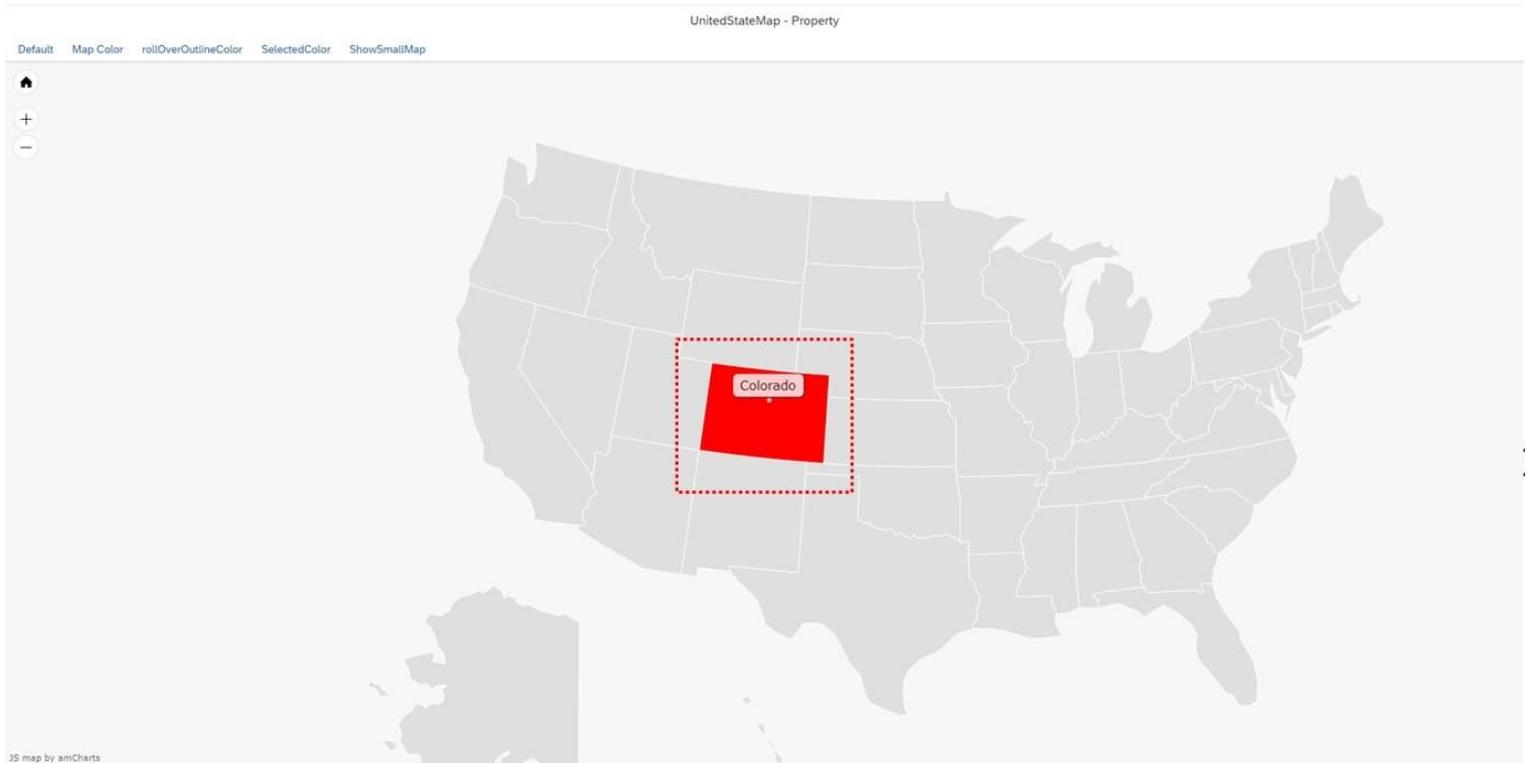


3. UnitedStateMap Properties - rollOverOutlineColor.



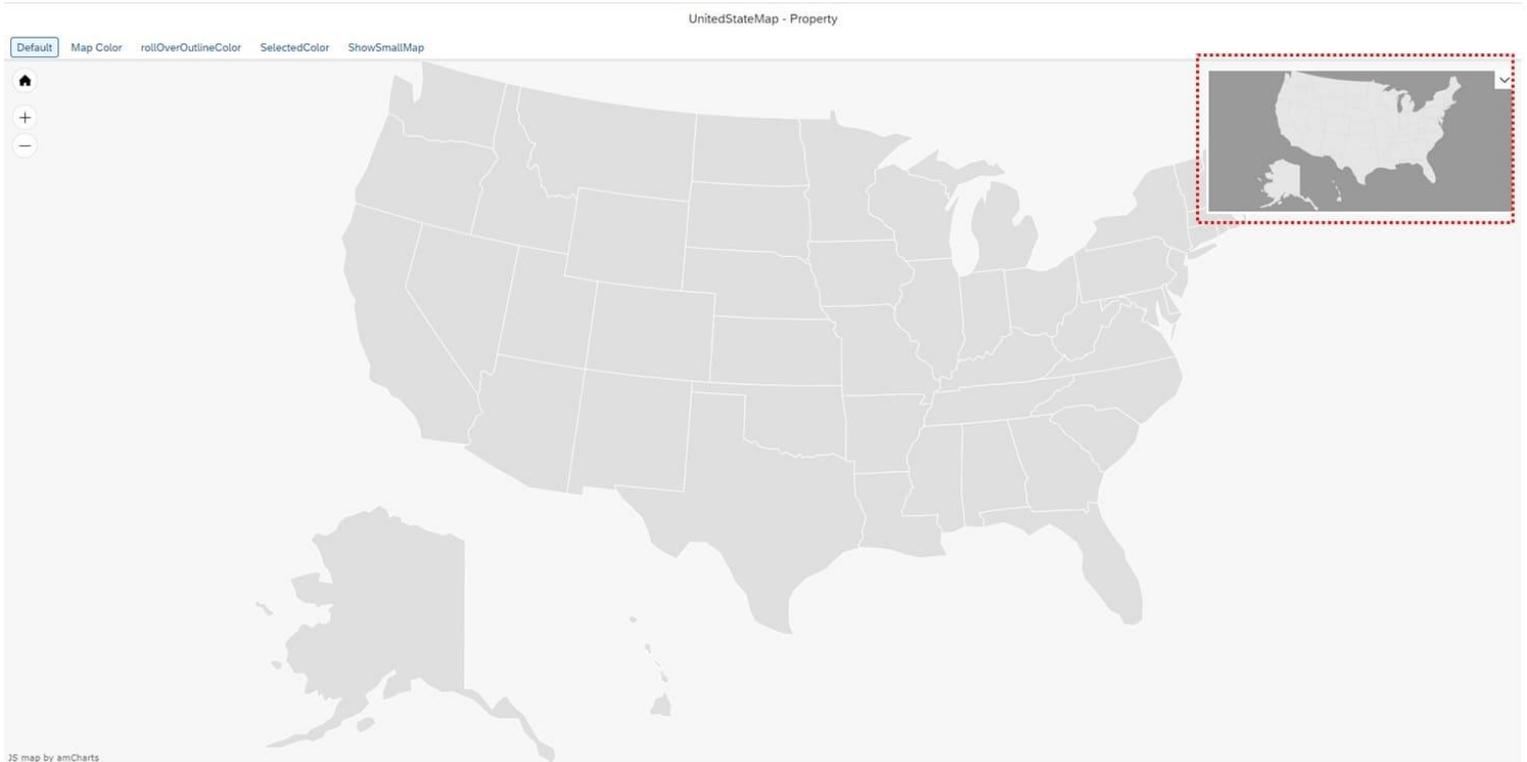
▣ 마우스 커서를 올려둔 지역은 위 Image와 같이 테두리를 통해 표시되며, rollOverOutlineColor는 이 때 생기는 테두리의 Color를 설정.

4. UnitedStateMap Properties - selectedColor



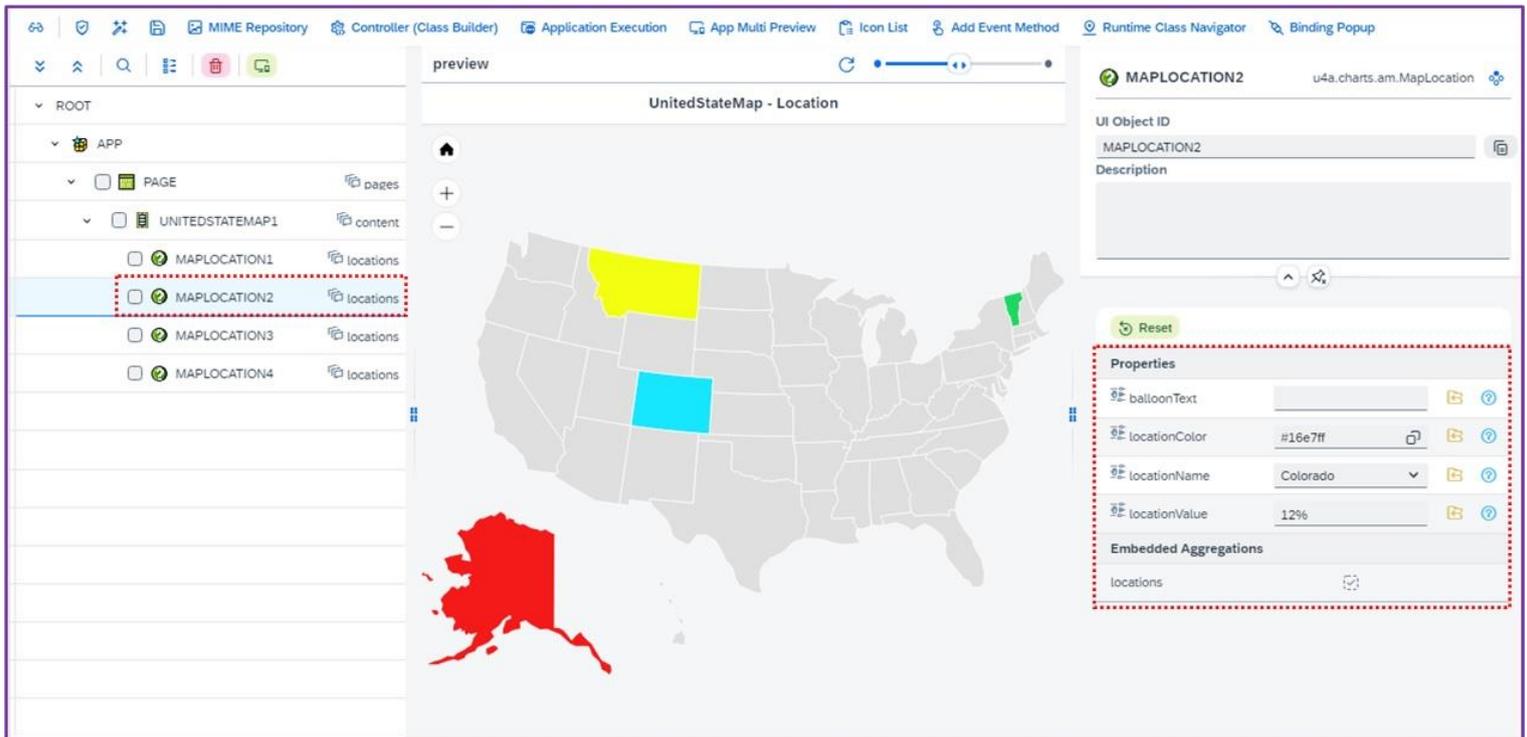
☐ 마우스로 Click한 지역은 위 Image와 같이 Color가 부여되어 표시되며, selected Color는 이 때 부여되는 Color를 설정.

5. UnitedStateMap Properties - showSmallMap.



☐ 우측 상단에 SmallMap이 생성된 것을 확인, SmallMap은 접기, 펼치기가 가능하며 SmallMap에서 지역을 Click 하면 UnitedStateMap 화면이 해당 지역으로 이동.

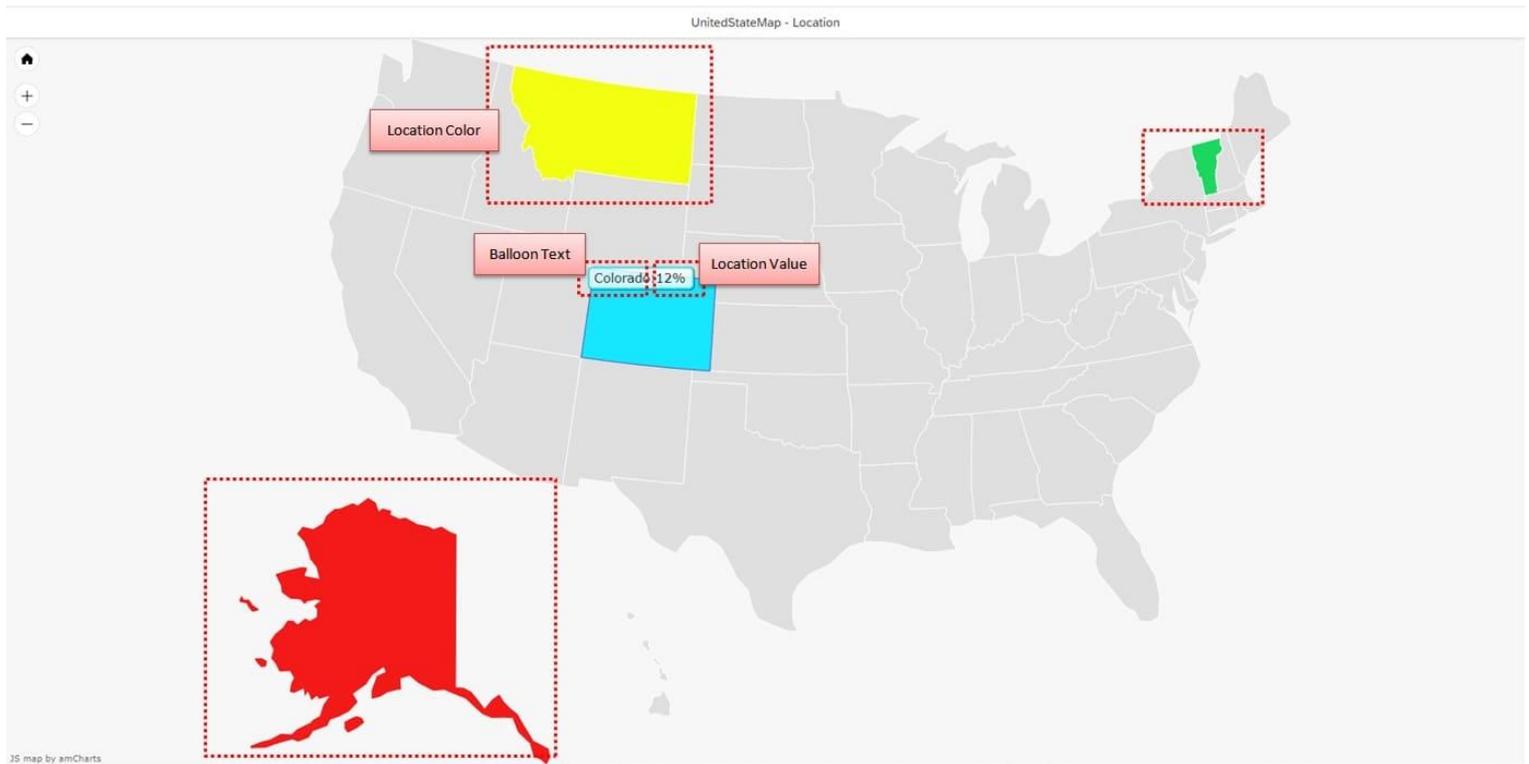
6. UnitedStateMap Aggregation - MapLocation.



▣ MapLocation은 UnitedStateMap의 Aggregation으로만 Insert 가능 .

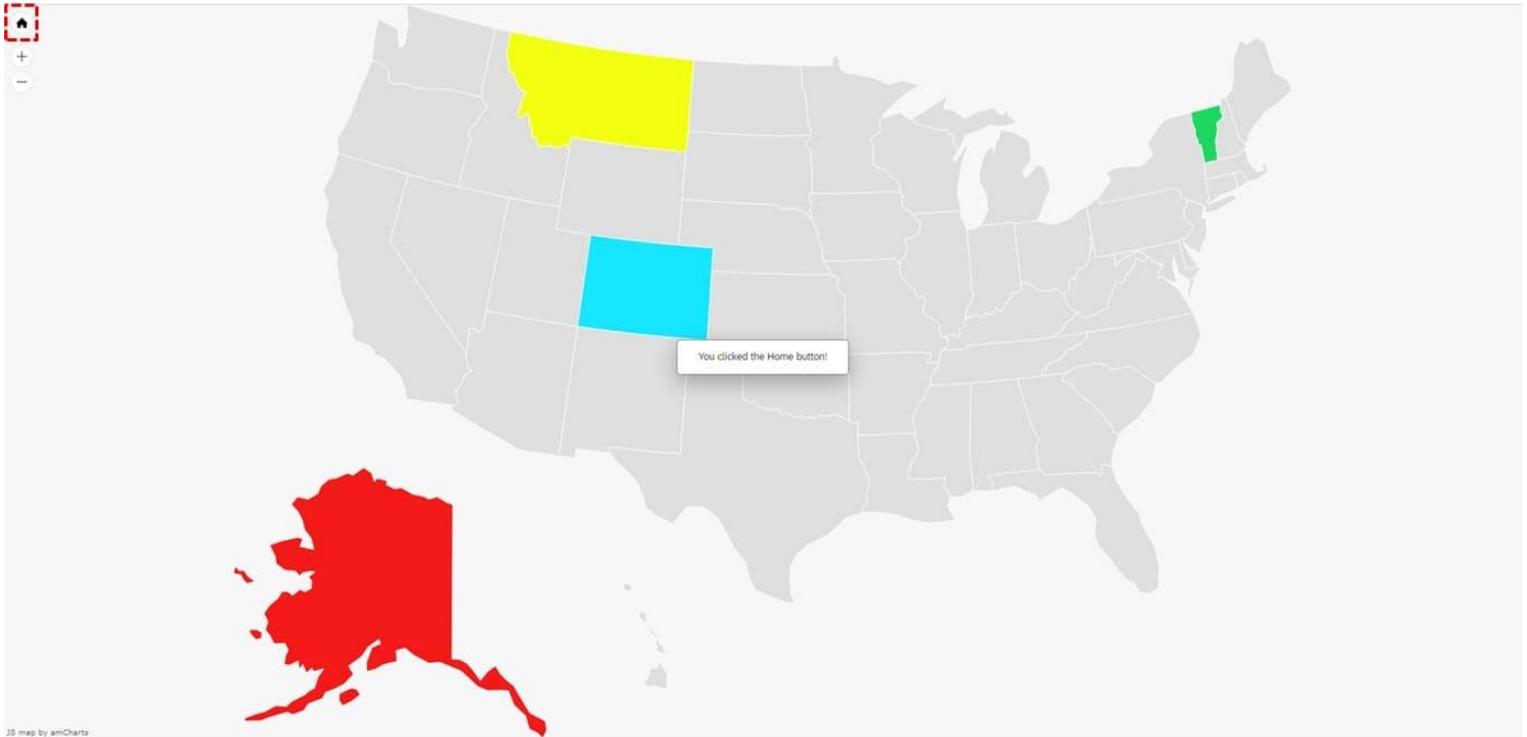
PROPERTIES	MEAN
balloonText	마우스 커서 올렸을 때 보이는 Text (값을 입력하지 않으면 locationName이 기본 값으로 설정)
locationColor	지정한 Location에 해당하는 영역 Color
locationName	지정하려는 Location 명 (USA의 실제 주 중에서 선택)
locationValue	지정한 Location에 부여하는 Value

7. UnitedStateMap 실행 했을 때 MapLocation Properties.



▣ MapLocation으로 설정한 4군데 Location이 UnitedStateMap UI에 표현된 것을 확인.

8. UnitedStateMap Events - goHome.



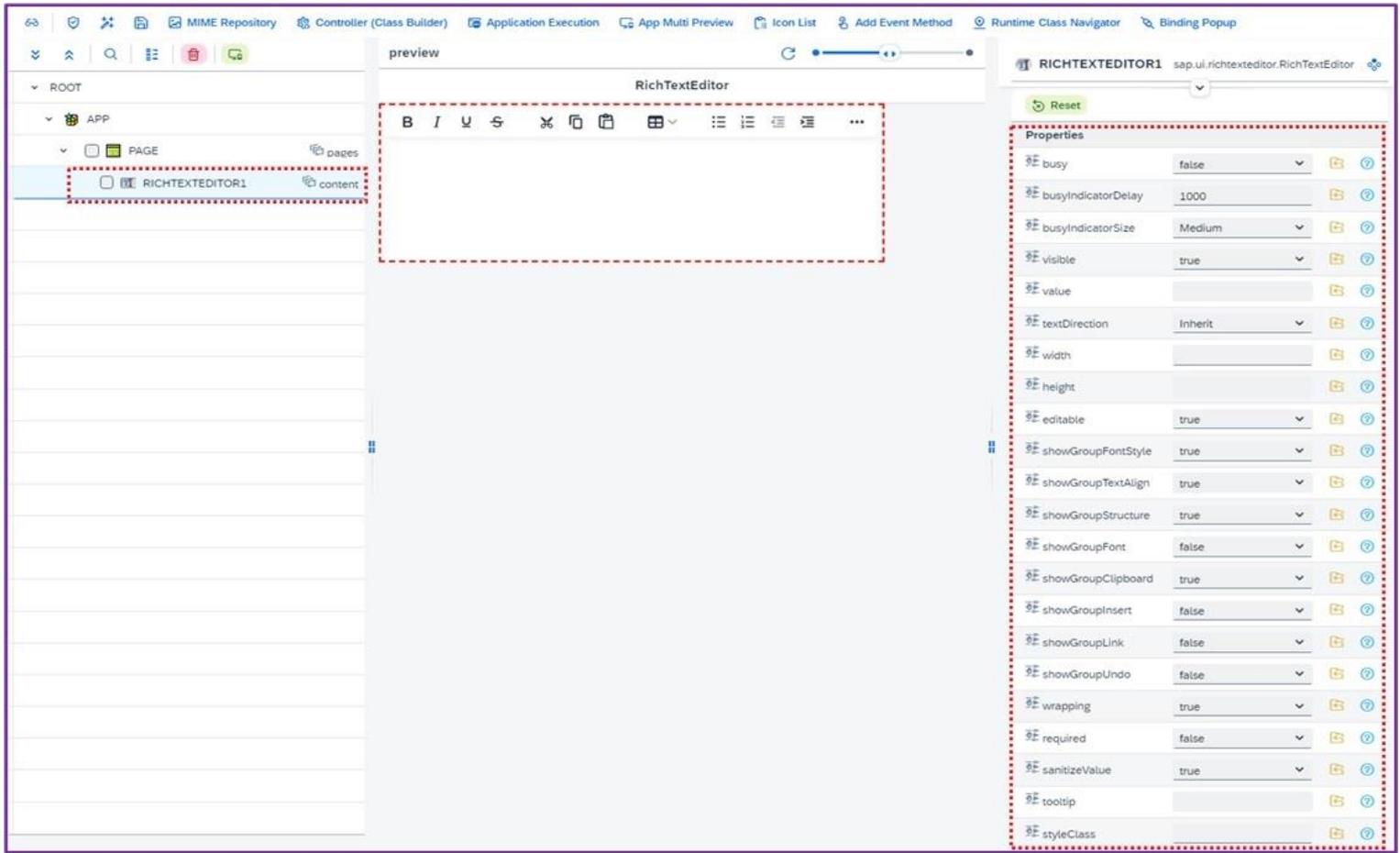
☐ goHome Event는 좌측 상단의 Home Button Click 할 때 동작, Method 미설정 시에는 Reset 기능이 기본적으로 있으며 특정 Event를 원한다면 Method 설정 가능.

9. UnitedStateMap Events - mapClick.



▣ mapClick Event는 UnitedStateMap의 Location을 Click 할 때 동작,

7.15 RichTextEditor

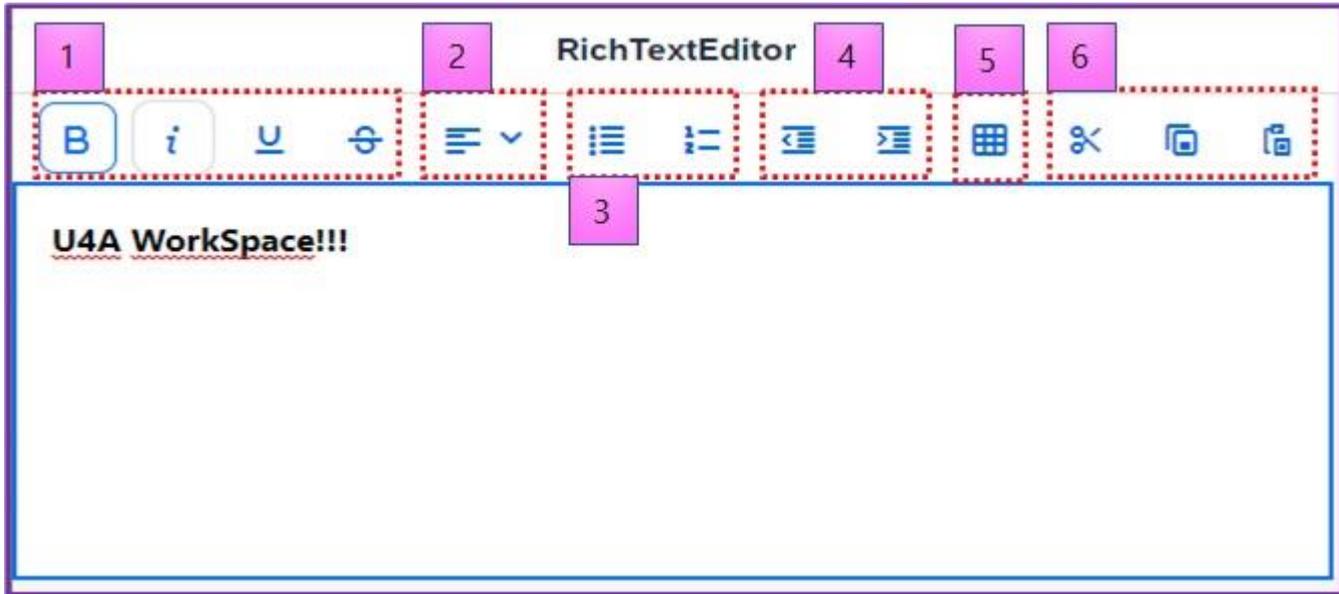


2. RichTextEditor Properties.

Attributes	Means
visible	UI를 보여줄지 안보여줄지 여부 결정 (True / False)
value	Text 입력란에 입력 할 Text 설정
textDirection	입력한 Text 정렬 (LTR: 왼쪽 정렬, RTL:오른쪽 정렬, Inherit: 양쪽 정렬)
width	RichTextEditor UI의 가로 길이 설정
height	RichTextEditor UI의 세로 높이 설정
editable	RichTextEditor 수정 가능 여부 설정 (True / False)

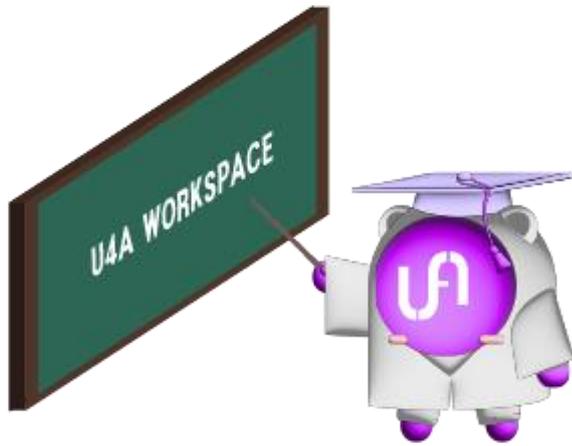
showGroupFontStyle	Text의 굵기, 밑줄 등 설정 Button 보여줄지 여부 설정 (True / False)
showGroupTextAlign	Text 정렬 Button 보여줄지 여부 설정 (True / False)
showGroupStructure	문단번호, 들여쓰기/내어쓰기 Button 보여줄지 여부 설정 (True / False)
showGroupFont	Text의 FontStyle, 색상, 크기 Button 보여줄지 여부 설정 (True / False)
showGroupClipboard	잘라내기, 복사, 붙여넣기 Button 보여줄지 여부 설정 (True / False)
showGroupInsert	Image, 감정표현 삽입 Button 보여줄지 여부 설정 (True / False)
showGroupLink	Link 삽입, 삭제 Button 보여줄지 여부 설정 (True / False)
showGroupUndo	되돌리기 Button 보여줄지 여부 설정 (True / False)
wrapping	UI의 가로 Width보다 Text의 길이가 길어질 때 줄 바꿀지 여부 설정
required	Value 값 필수 여부 설정 (True / False)
sanitizeValue	Html 실행 여부 설정 (True / False)
tooltip	UI에 마우스 올렸을 때 보여 줄 Text 설정
styleClass	CSS 설정

3. Web화면에서 UI Object가 생성된 것을 확인 할 수 있다.



- 1) 글자 굵기, 기울기, 밑줄, 취소선
- 2) 좌,우,가운데 정렬
- 3) 맨 앞의 줄마다 점 또는 번호 추가
- 4) 줄의 들여쓰기/내어쓰기
- 5) Text Area에 표 생성
- 6) 잘라내기, 복사, 붙여넣기

8. APP 개발 하기 Part 4



8.1 sapUiMargin

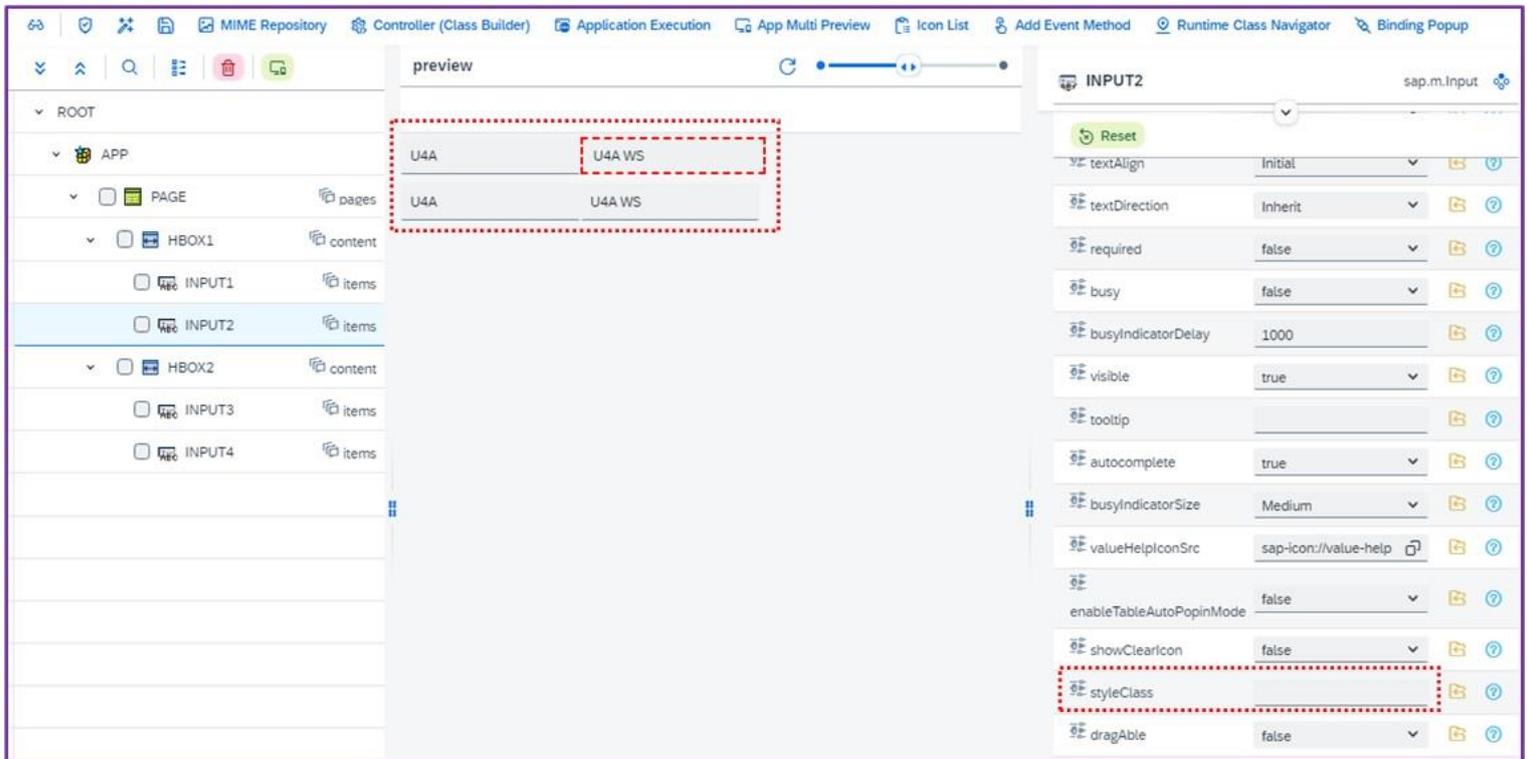
2) sapUiSmallMargin

3) sapUiMediumMargin

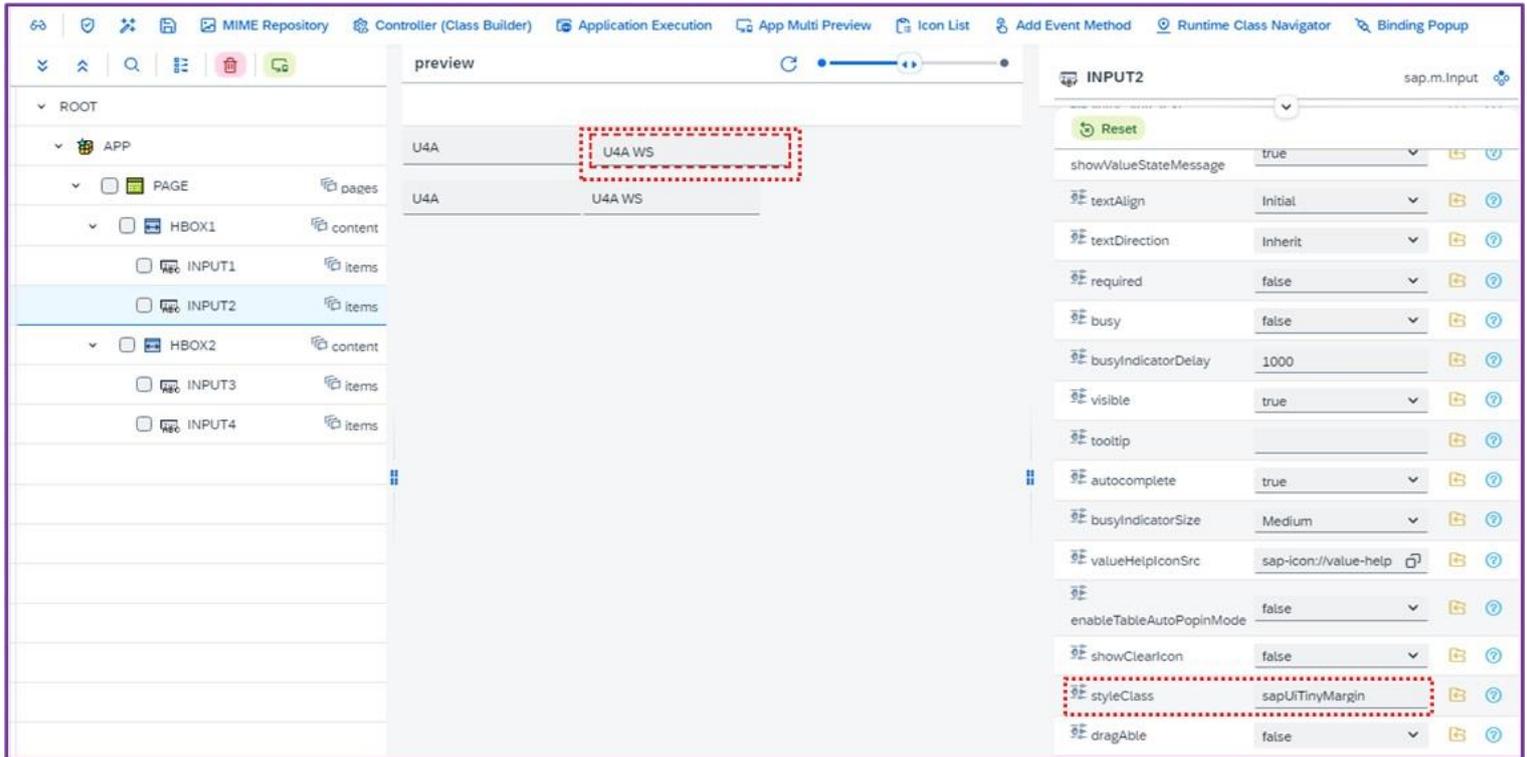
4) sapUiLargeMargin

※ Tiny -> Small -> Medium -> Large 순서대로 더 큰 여백을 만들 수 있음 .

1-1 sapMargin 을 적용하지 않았을 때 UI .

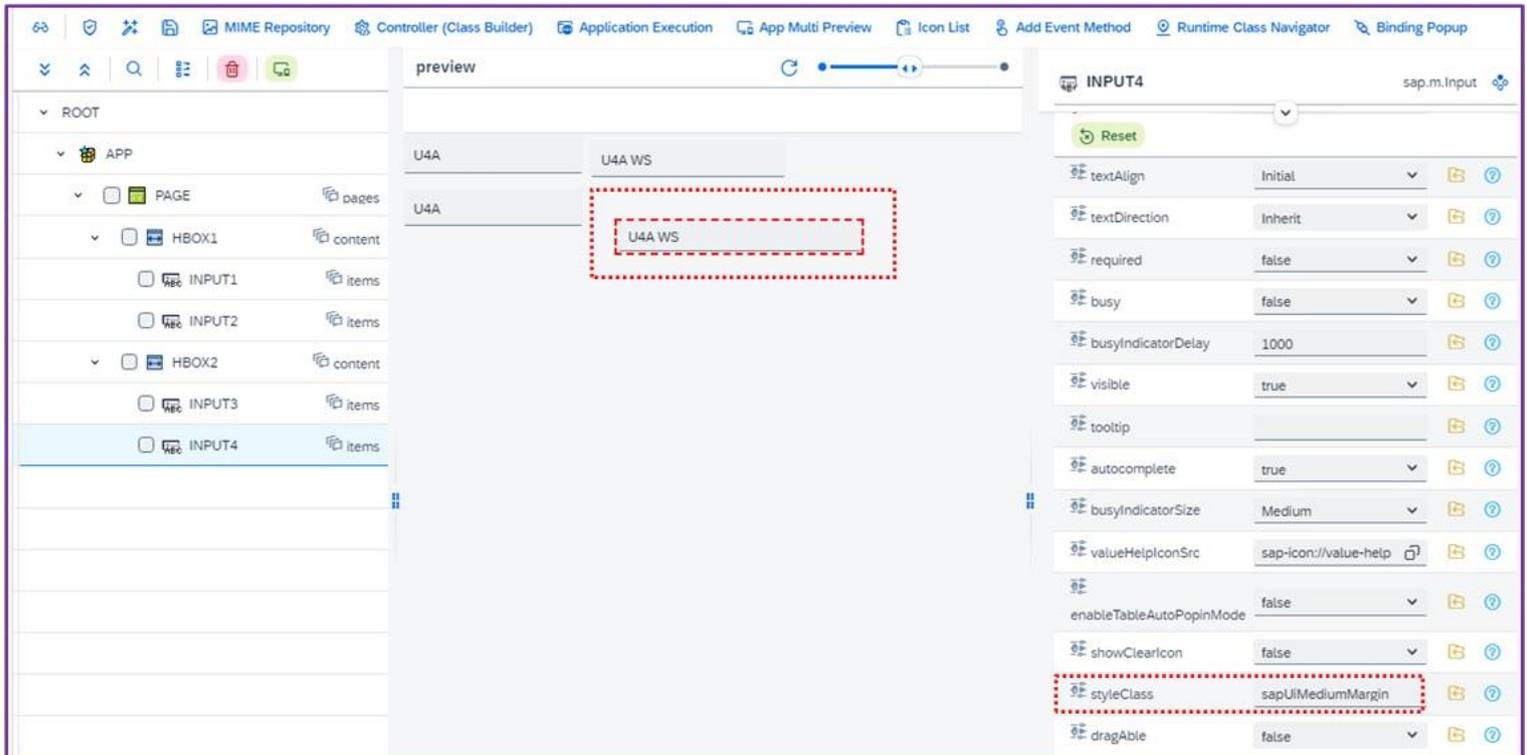


1-2 sapUiTinyMargin 을 적용했을 때 UI .



상하좌우에 미세한 여백이 생긴 것을 확인 할 수 있음 .

1-3 sapMediumMargin 을 적용했을 때 UI .



상하좌우에 sapUiTinyMargin 보다 더 큰 여백이 생긴 것을 확인 할 수 있음 .

2. Single- Sided Margins

Object간 특정 측면에 여백을 주고 싶을 때 사용 .

1)

sapUiTinyMarginTop sapUiTinyMarginBottom sapUiTinyMarginBegin sapUiTinyMarginEnd

2)

sapUiSmallMarginTop sapUiSmallMarginBottom sapUiSmallMarginBegin sapUiSmallMarginEnd

3)

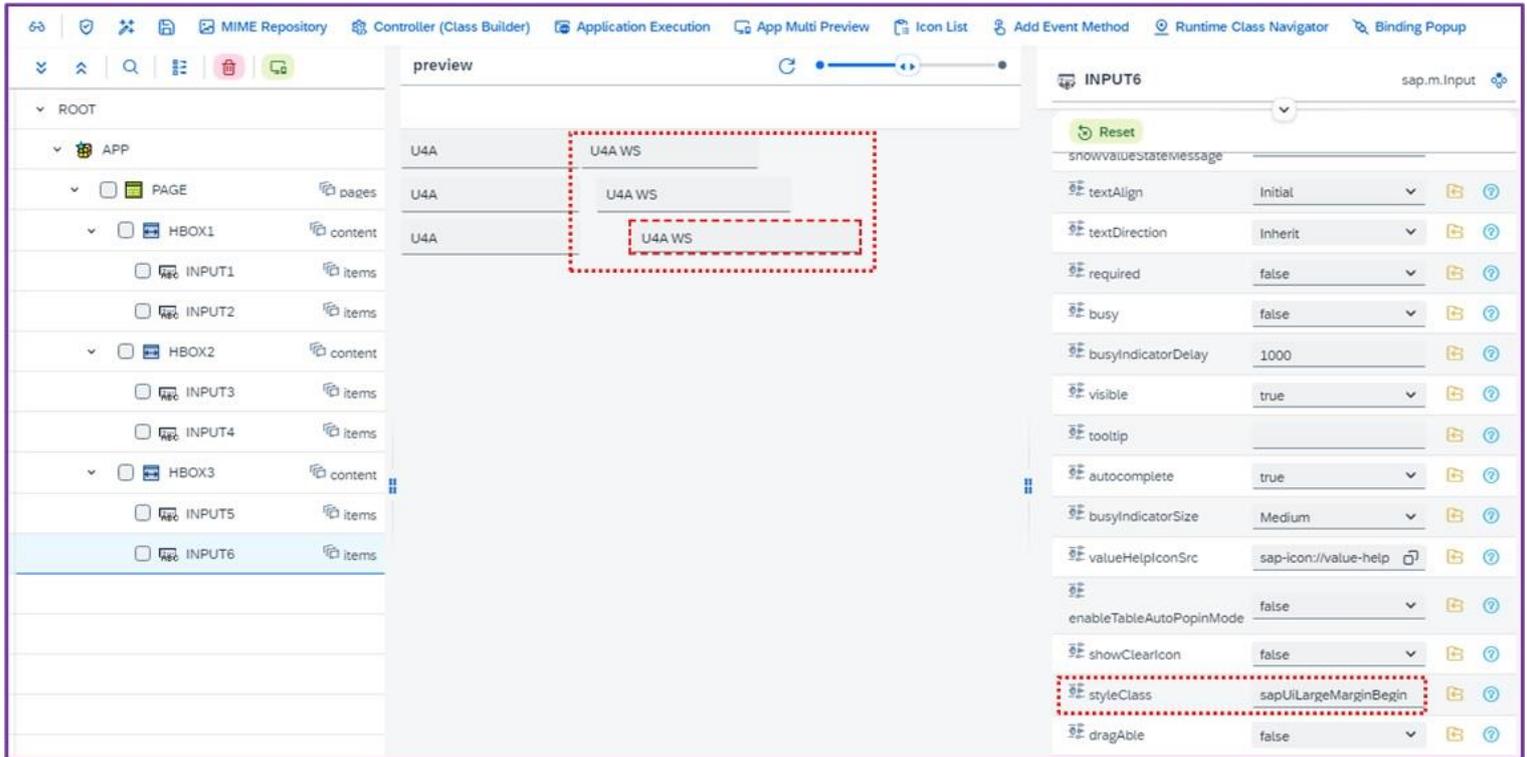
sapUiMediumMarginTop sapUiMediumMarginBottom sapUiMediumMarginBegin sapUiMediumMarginEnd

4)

sapUiLargeMarginTop sapUiLargeMarginBottom sapUiLargeMarginBegin sapUiLargeMarginEnd

※ Tiny -> Small -> Medium -> Large 순서대로 더 큰 여백을 만들 수 있음 .

2-1 sapMarginBegin 을 적용했을 때 UI .



UI의 좌측에 여백이 생긴 것을 확인할 수 있으며 여백의 크기 조절 가능, 다른 측면도 같은 방식으로 적용 가능함 .

3. Two- Sided Margins

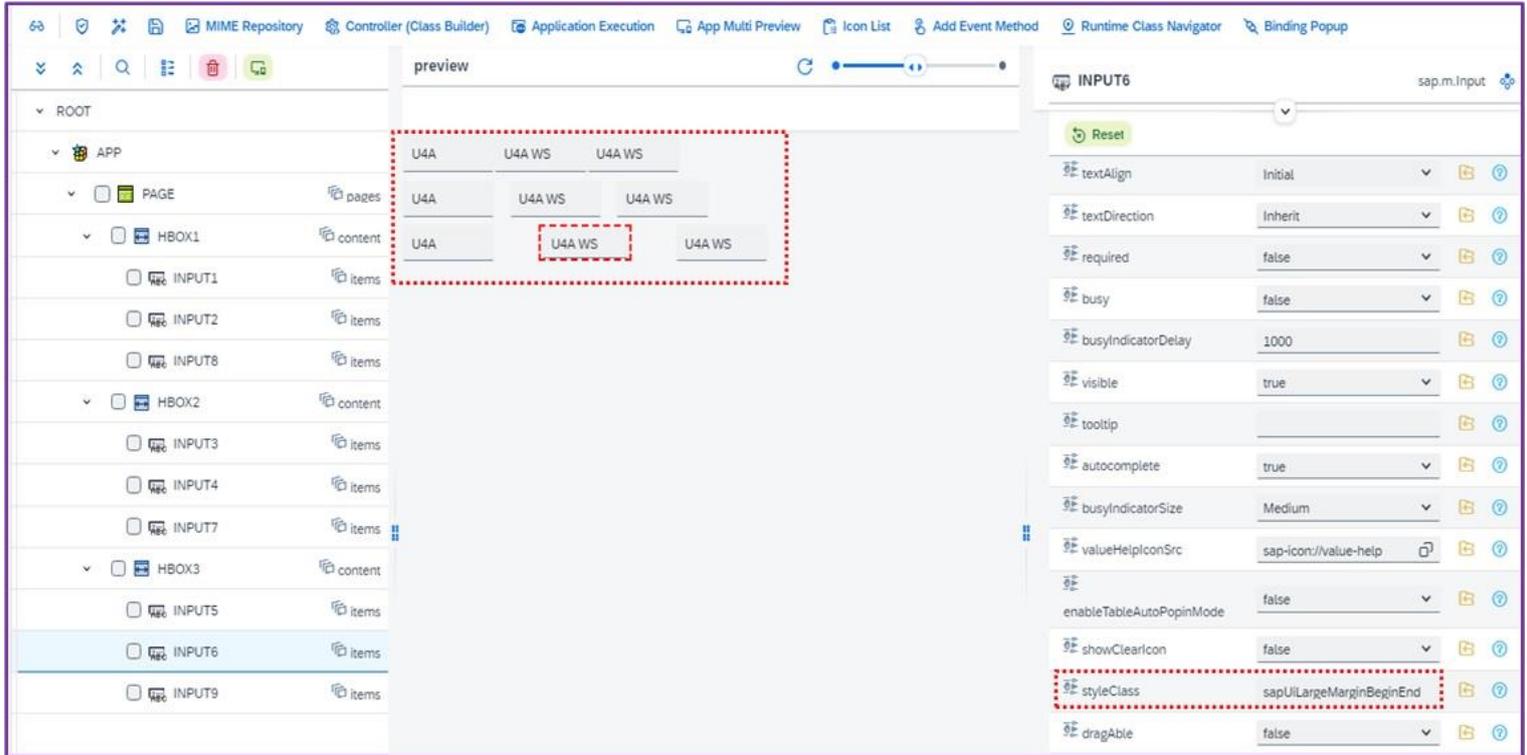
Object간 양쪽 측면에 여백을 주고 싶을 때 사용 .

1) `sapUiTinyMarginBeginEnd` `sapUiSmallMarginBeginEnd` `sapUiMediumMarginBeginEnd` `sapUiLargeMarginBeginEnd`

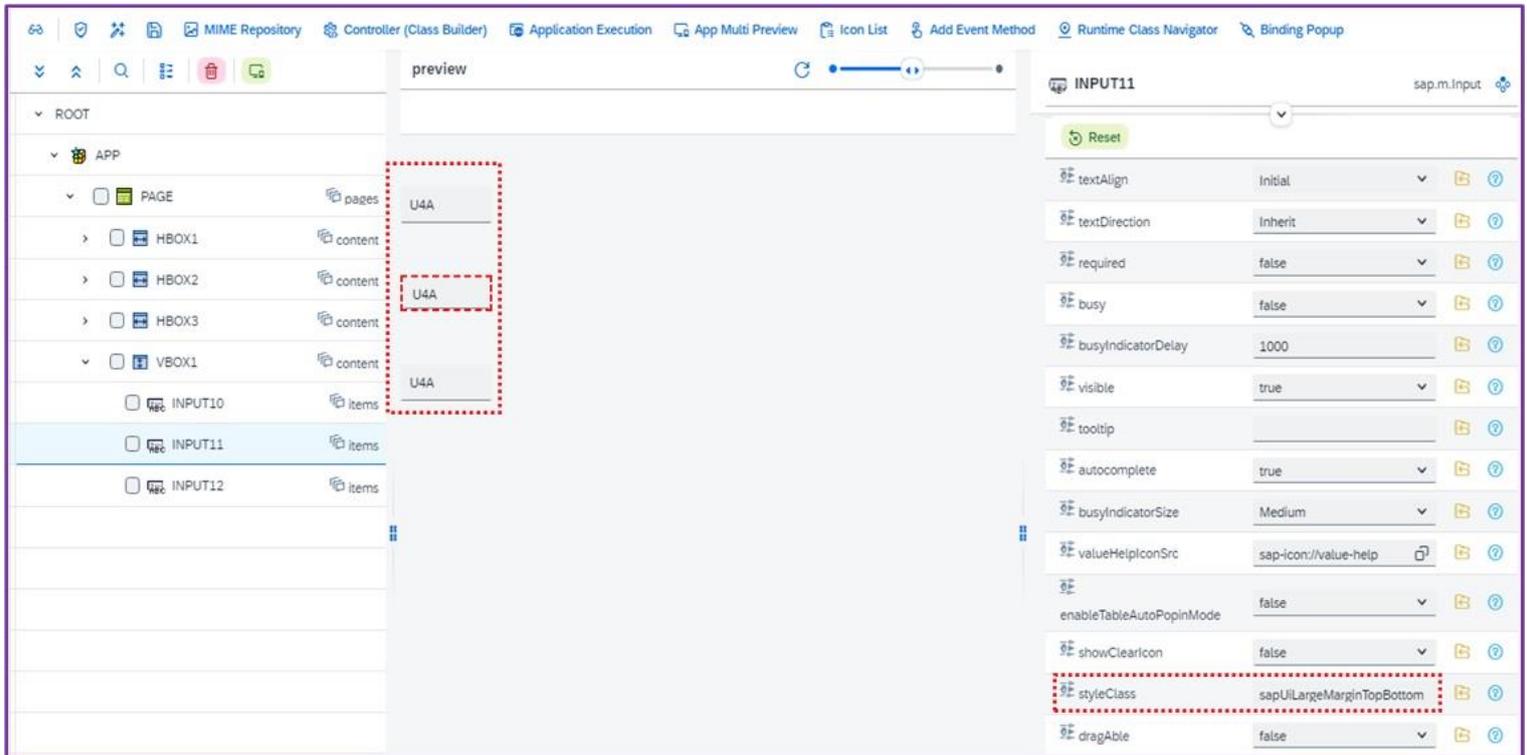
2) `sapUiTinyMarginTopBottom` `sapUiSmallMarginTopBottom` `sapUiMediumMarginTopBottom` `sapUiLargeMarginTopBottom`

※ Tiny -> Small -> Medium -> Large 순서대로 더 큰 여백을 만들 수 있음 .

3-1 sapMarginBeginEnd 을 적용했을 때 UI .



UI 의 좌우에 여백이 생긴 것을 확인할 수 있으며 여백의 크기 조절 가능.



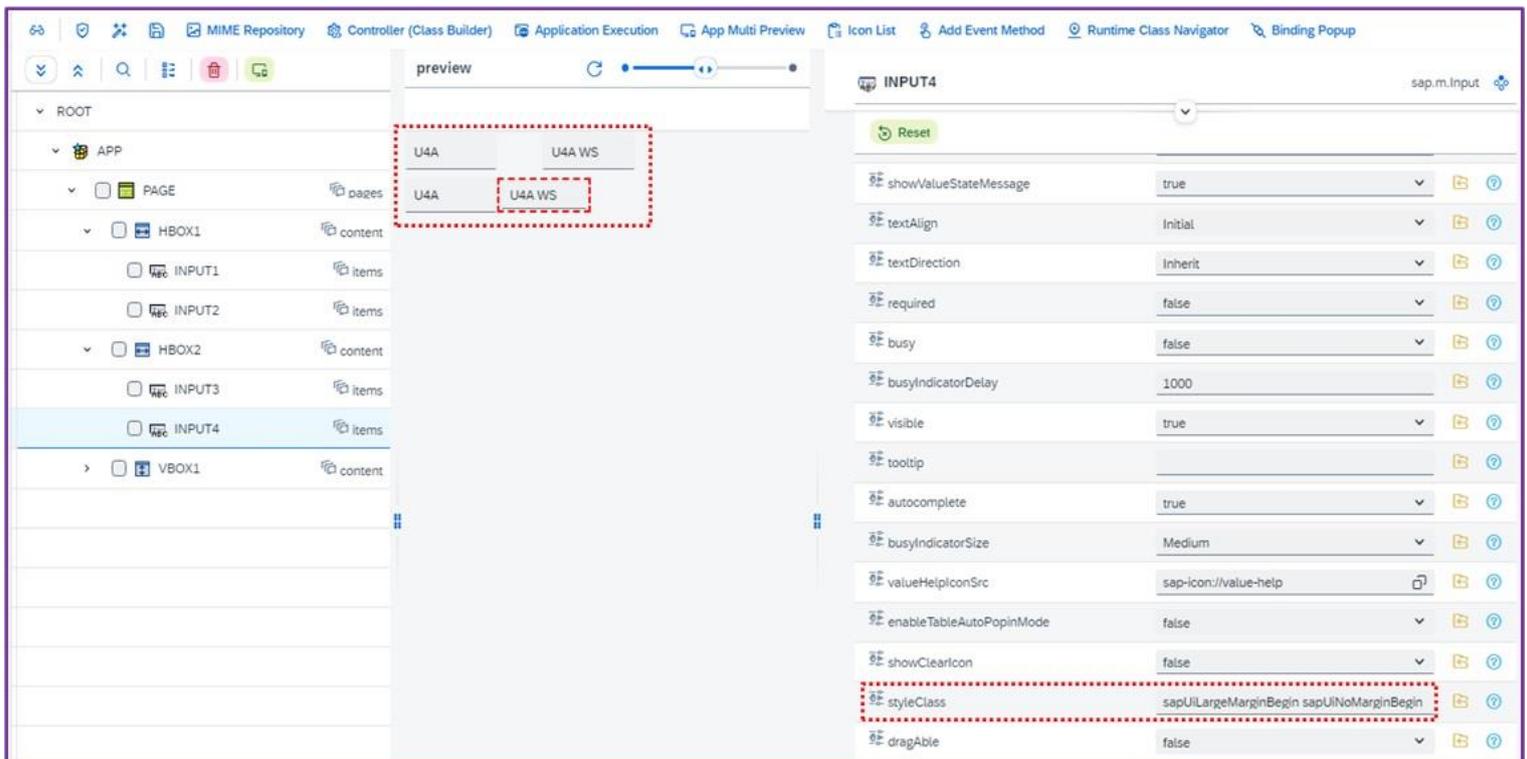
UI 의 상하에 여백이 생긴 것을 확인할 수 있으며 여백의 크기 조절 가능 .

4. Removing Margins

sapUiMargin을 통해 생긴 특정 측면의 여백을 제거할 때 사용 .

- 1) sapUiNoMarginTop
- 2) sapUiNoMarginBottom
- 3) sapUiNoMarginBegin
- 4) sapUiNoMarginEnd

4-1 sapNoMarginBegin 적용했을 때 UI.



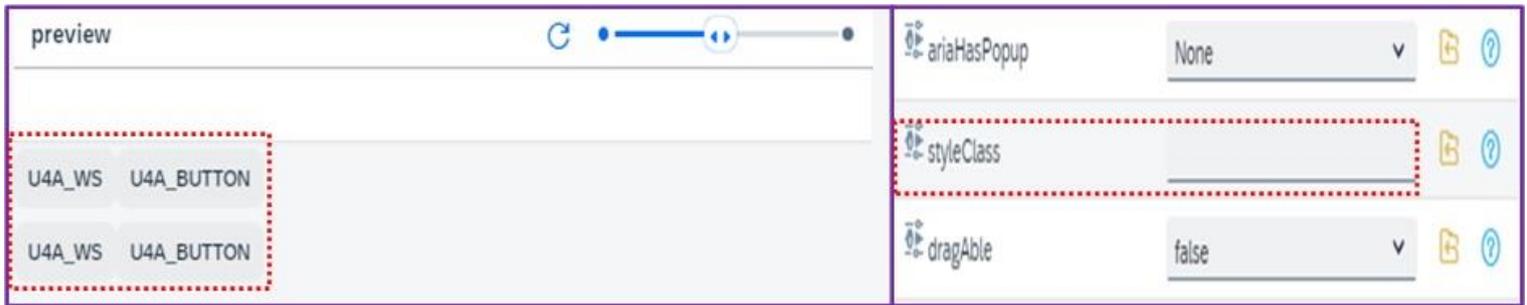
위 Input UI 가 여백이 좌측에 여백이 생긴 것과 달리 아래 Input UI 는 좌측 여백이 사라진 것을 확인할 수 있음.

8.2 sapUiSizeCompact and Cozy

■ 목적 : UI Object의 Content의 밀도 크기를 조절 할 수 있다.

1. sapUiSizeCompact (Object간 간격을 줄이고 밀도를 높여서 더 많은 Object를 보여준다)

1.1 sapUiSizeCompact "미적용"

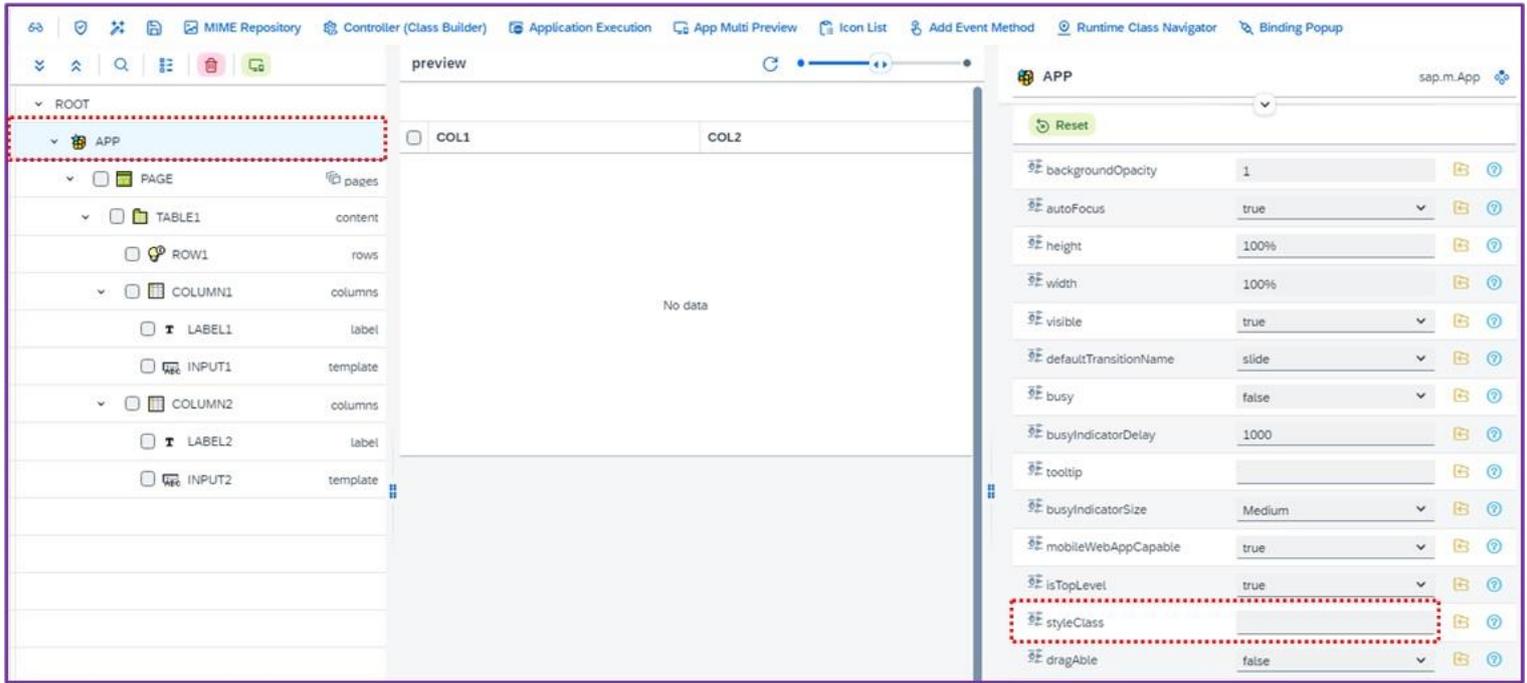


1.2 sapUiSizeCompact "적용"



2. sapUiSizeCozy

App Style Class 에서 적용해서 UI Table의 전체 크기를 늘릴때 사용한다.



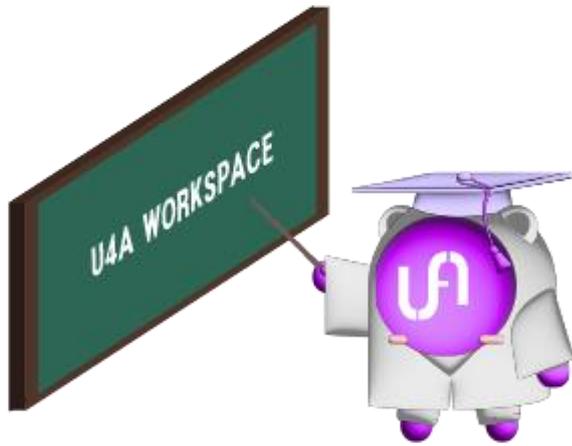
2-1. sapUiSizeCozy 적용 전 UI TABLE 화면

<input type="checkbox"/>	COL1	COL2
<input type="checkbox"/>	U4A	U4A_WS

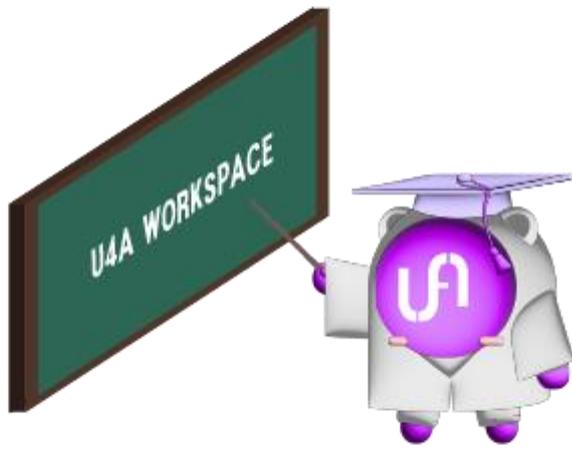
2-2. sapUiSizeCozy 적용 후 UI TABLE 화면

<input type="checkbox"/>	COL1	COL2
<input type="checkbox"/>	U4A	U4A_WS

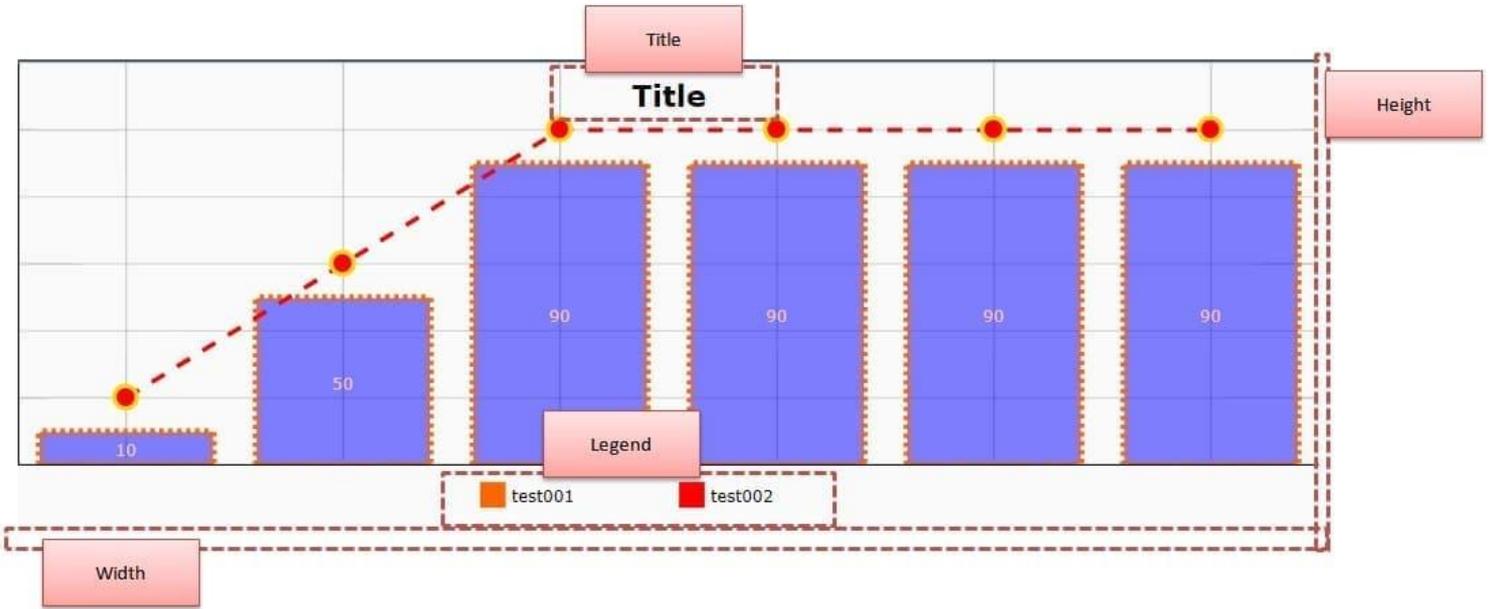
9. APP 개발 하기 Part 5



9.1 AmSerialChartComposite



9.1.1 Properties



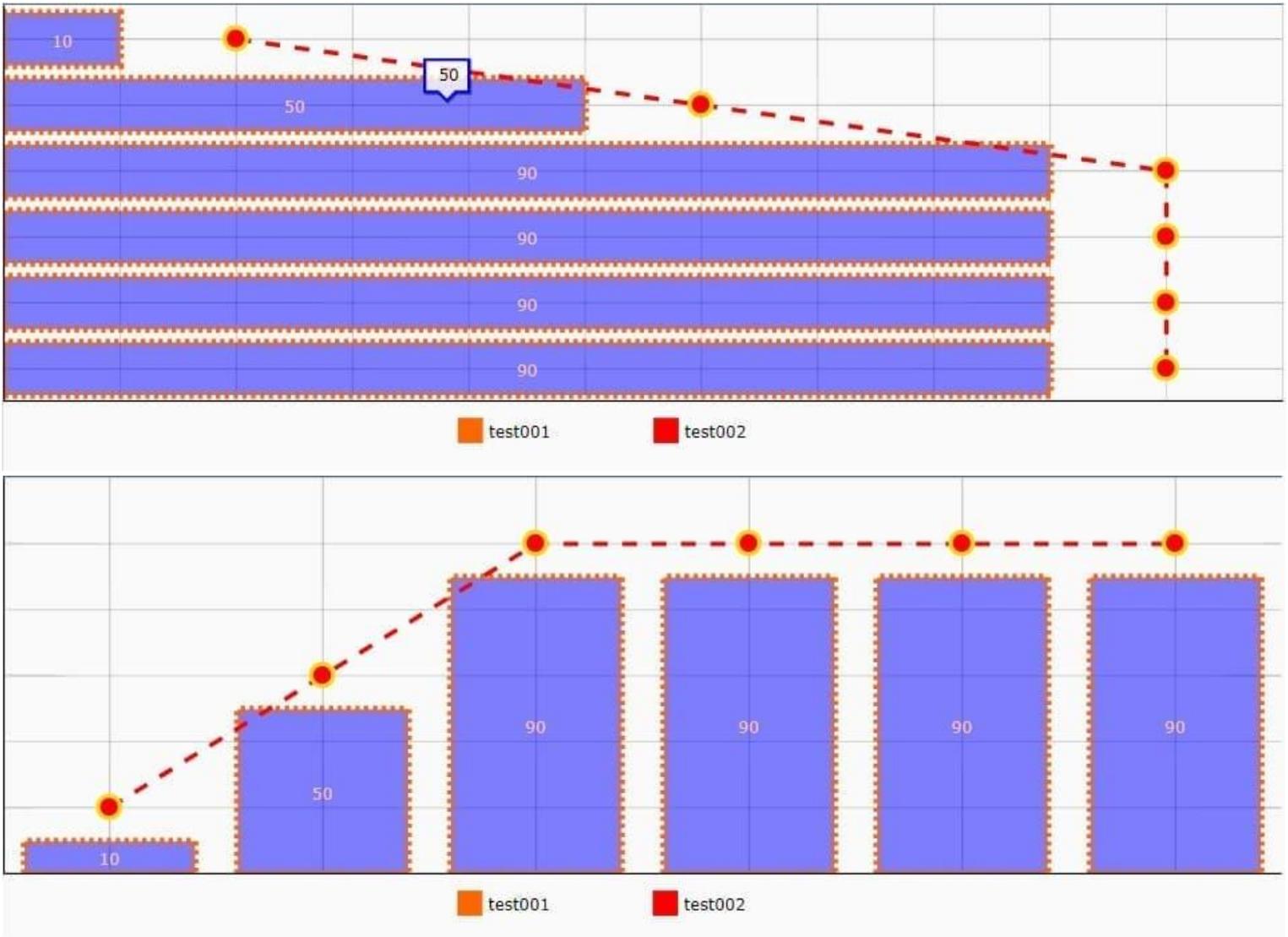
Properties			
width	GS_CHART-WIDTH		
height	GS_CHART-HEIGHT		
title	GS_CHART-TITLE		
visible	GS_CHART-VISIBLE		
styleClass	GS_CHART-STYLECLASS		
showLegend	GS_CHART-SHOWLEGEND		
rotate	GS_CHART-ROTATE		
startDuration	GS_CHART-STARTDURATION		
legendPosition	GS_CHART-LEGENDPOSITION		
legendAlign	GS_CHART-LEGENDALIGN		
columnWidth	GS_CHART-COLUMNWIDTH		
LegendFontSize	GS_CHART-LEGENDFONTSIZE		
marginTop	GS_CHART-MARGINTOP		
marginLeft	GS_CHART-MARGINLEFT		
marginRight	GS_CHART-MARGINRIGHT		
marginBottom	GS_CHART-MARGINBOTTOM		
autoMargins	GS_CHART-AUTOMARGINS		
angle	GS_CHART-ANGLE		
depth3D	GS_CHART-DEPTH3D		
plotAreaBorderAlpha	GS_CHART-PLOTAREABORDERALPHA		
plotAreaBorderColor	GS_CHART-PLOTAREABORDERCOLOR		
plotAreaFillAlphas	GS_CHART-PLOTAREAFILLALPHAS		
plotAreaFillColors	GS_CHART-PLOTAREAFILLCOLORS		

backgroundAlpha	GS_CHART-BACKGROUNDALPHA		
backgroundColor	GS_CHART-BACKGROUNDCOLOR		
borderAlpha	GS_CHART-BORDERALPHA		
borderColor	GS_CHART-BORDERCOLOR		
legendMarkerSize	GS_CHART-LEGENDMARKERSIZE		
legendValueWidth	GS_CHART-LEGENDVALUEWIDTH		
legendVerticalGap	GS_CHART-LEGENDVERTICALGAP		
legendHorizontalGap	GS_CHART-LEGENDHORIZONTALGAP		
showChartScrollbar	GS_CHART-SHOWCHARTSCROLLBAR		
updateOnReleaseOnly	GS_CHART-UPDATEONRELEASEONLY		
zoomOutText	GS_CHART-ZOOMOUTTEXT		
zoomOutButtonAlpha	GS_CHART-ZOOMOUTBUTTONALPHA		
zoomOutButtonColor	GS_CHART-ZOOMOUTBUTTONCOLOR		
zoomOutButtonPadding	GS_CHART-ZOOMOUTBUTTONPADDING		
zoomOutButtonRollOverAlpha	GS_CHART-ZOOMOUTBUTTONROLLOVERALPHA		
legendColor	GS_CHART-LEGENDCOLOR		
textColor	GS_CHART-TEXTCOLOR		

PROPERTIES	MEAN
WIDTH	Graph 전체 가로 길이 조절
HEIGHT	Graph 전체 세로 높이 조절
TITLE	Graph 타이틀 텍스트 설정
VISIBLE	Graph 보일 지 여부 설정 (TRUE: 보이게 설정 FALSE: 안보이게 설정)
STYLECLASS	Graph에 적용할 CSS 설정
SHOWLEGEND	범례 보일 지 여부 설정 (TRUE: 보이게 설정 FALSE: 안보이게 설정)
ROTATE	Graph 90도 회전 시킬지 여부 설정 (TRUE: 90도 회전 FALSE: 회전 안함)
STARTDURATION	Graph가 등장하는데 걸리는 시간 설정
LEGENDPOSITION	범례 위치 설정 (Top, Bottom, Left, Right)
LEGENDALIGN	범례 정렬 설정 (Left, Center, Right)
COLUMNWIDTH	Graph 너비 설정
LEGENDFONTSIZE	범례 Description Text Size 조절
MARGINTOP	Graph 상단 여백 설정
MARGINLEFT	Graph 좌측 여백 설정
MARGINRIGHT	Graph 우측 여백 설정
MARGINBOTTOM	Graph 하단 여백 설정
AUTOMARGINS	자동 여백 설정 (TRUE: 설정 FALSE: 설정 안함)
ANGLE	Graph를 바로보는 Angle 설정 (DEPTH3D 값이 0이면 ANGLE 변화 없음)
DEPTH3D	입체적으로 보이는 정도 조절
PLOTAREABORDERALPHA	그림 영역 테두리 투명도 설정
PLOTAREABORDERCOLOR	그림 영역 테두리 색상 설정

PLOTAREAFILLALPHAS	그림 영역 색상 투명도 설정
PLOTAREAFILLCOLORS	그림 영역 색상 설정
BACKGROUNDALPHA	Graph 배경 색상 투명도 설정
BACKGROUNDCOLOR	Graph 배경 색상 설정
BORDERALPHA	Graph 배경 테두리 투명도 설정
BORDERCOLOR	Graph 배경 테두리 색상 설정
LEGENDMARKERSIZE	범례 Icon Size 조절
LEGENDVALUEWIDTH	범례 가로 영역 조절 (범례 간 간격 조절할 때 사용 가능)
LEGENDVERTICALGAP	범례 세로 위치 조절
LEGENDHORIZONTALGAP	범례 가로 위치 조절
SHOWCHARTSCROLLBAR	상단에 Graph 특정 구간을 확대하여 볼 수 있는 Scrollbar 생성 (TRUE: 설정 FALSE: 설정 안함)
UPDATEONRELEASEONLY	Scrollbar를 움직일 때 마우스를 Drop시에만 동작하도록 설정 (TRUE: 설정 FALSE: 설정 안함)
ZOOMOUTTEXT	ShowChartScrollbar를 True로 설정 했을 때 ZoomOut Button에 대한 Description Text
ZOOMOUTBUTTONALPHA	ZommOut Button 및 Text 영역의 투명도 설정 (0으로 설정하면 색상 변화가 없음)
ZOOMOUTBUTTONCOLOR	ZommOut Button 및 Text 영역의 색상 설정
ZOOMOUTBUTTONPADDING	ZommOut Button 및 Text 영역의 크기 조절
ZOOMOUTBUTTONROLLO	ZommOut Button 위에 마우스를 올렸을 때 투명도 조절 (0으로 설정하면 마우스를 올려도 색상 변화가 없음)
LEGENDCOLOR	범례 Text 색상 설정

2. AmSerialChartComposite Property - Rotate .



Graph를 회전 시킬 지 여부를 설정.

위 Image와 같이 Rotate 값이 'True' 또는 'False'인지에 따라 Graph가 90도 회전 된 형태인 것을 확인 가능.

3. AmSerialChartComposite Property - Angle & Depth3D .

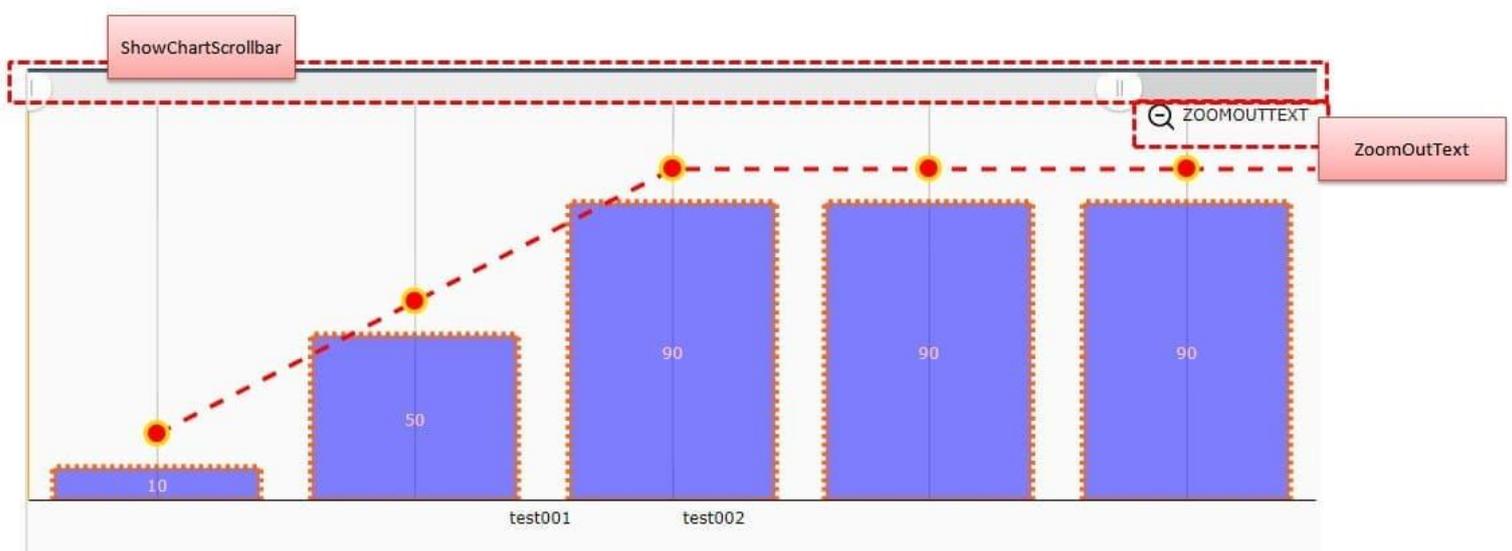


Depth3D를 통해 Grpah를 어느정도 입체적으로 보일 것인지 설정 가능.

Angle을 통해 입체적인 Graph를 바로보는 각도를 설정할 수 있음.

위 Image는 Angle, Depth3D 값을 각각 50으로 설정 하였으며 위와 같이 입체적인 그래프를 정면이 아닌 방향에서 바로 본 형태로 그려짐 .

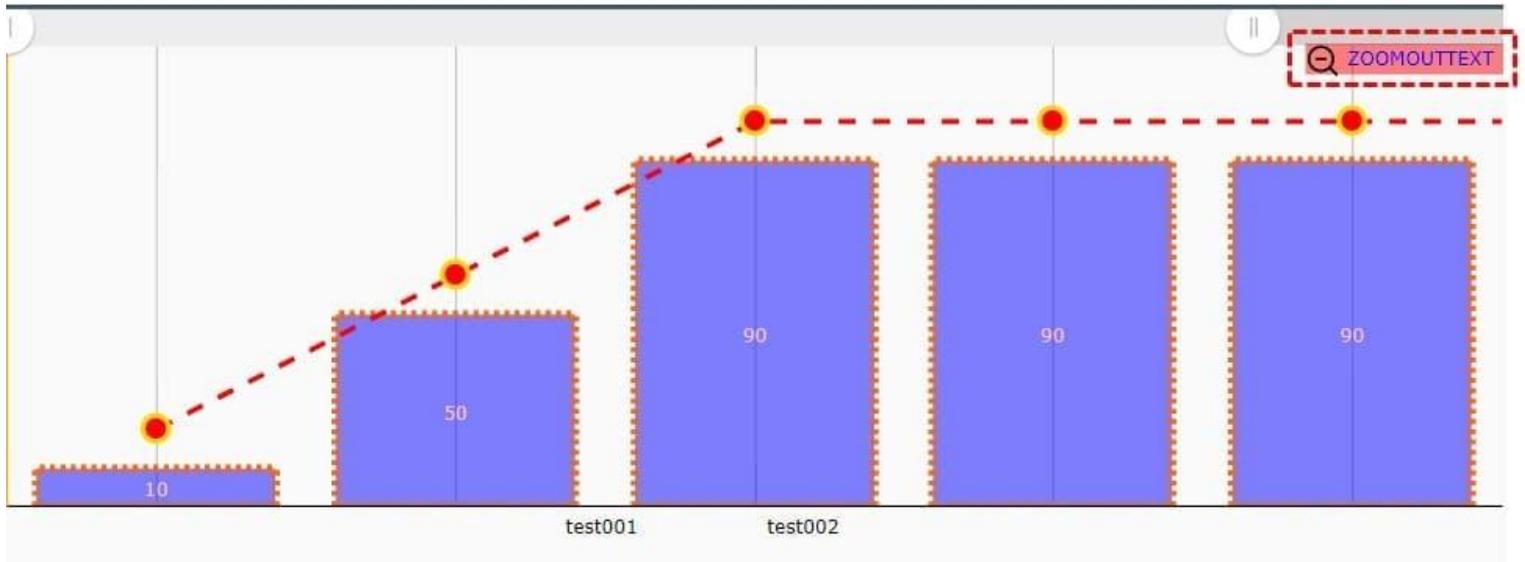
4. AmSerialChartComposite Property - ShowChartScrollbar & ZoomOutText .



ShowChartScrollbar에 true값을 주면 상단의 Scrollbar가 생성.

Scrollbar를 통해 화면을 확대할 수 있으며 확대 하는 순간 ZoomOutText가 오른쪽 상단에 생성되는 것을 확인.

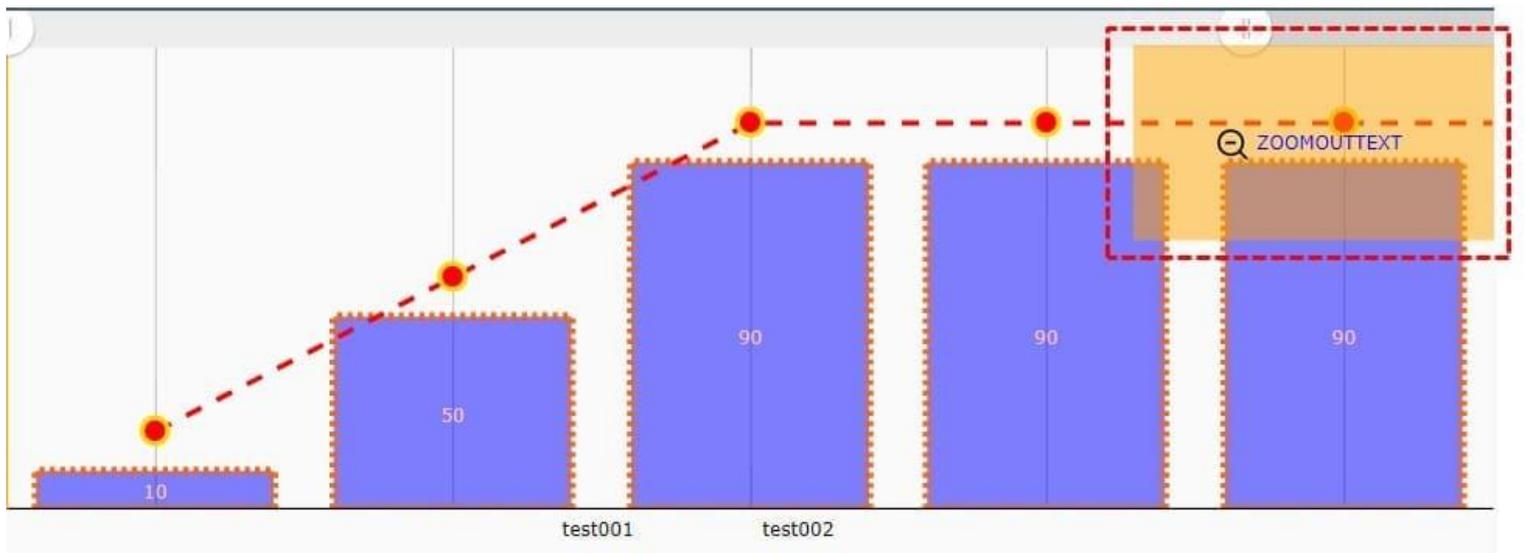
1) ZoomOutButtonColor & ZoomOutButtonAlpha & TextColor.

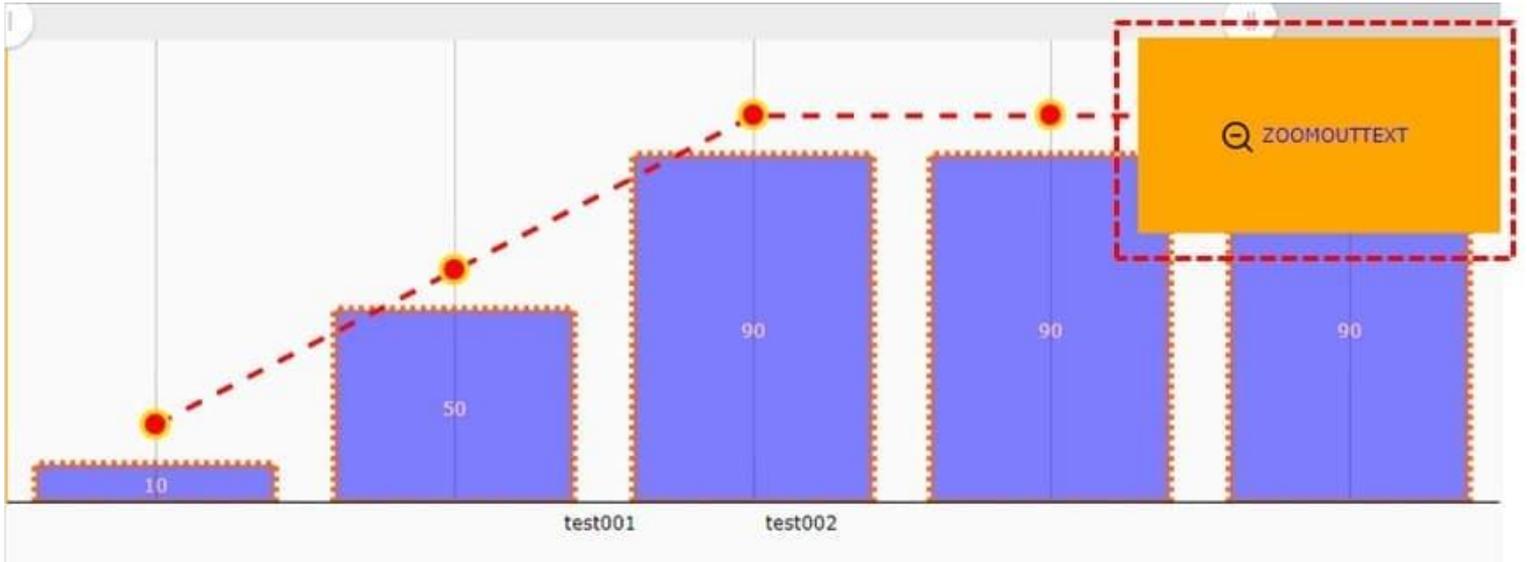


위 Image의 경우 ZoomOutButtonColor 값을 'red', ZoomOutButtonAlpha 값을 0.5로 설정하고 TextColor를 'blue'로 설정.

배경이 붉은색이며 투명도가 0.5정도에 'ZOOMOUTTEXT'라는 Text가 파란색으로 설정된 것을 확인.

2) ZoomOutButtonPadding & ZoomOutButtonRollo.

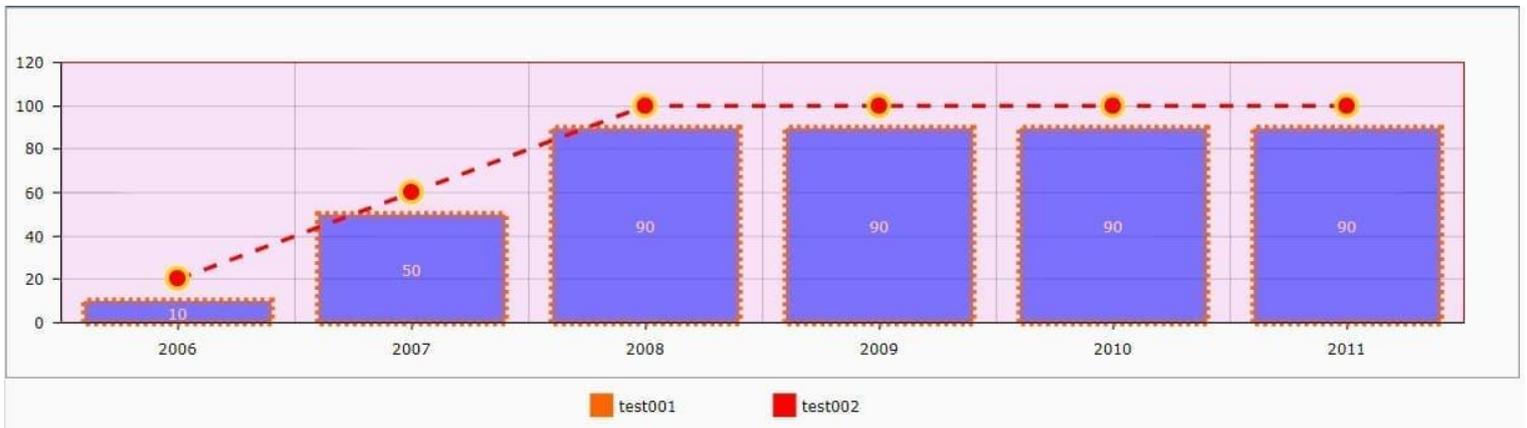




왼쪽 Image의 경우 ZoomOutButtonPadding 값을 '50'으로 설정하였으며 ZoomOutText 영역이 커진 것을 확인.

오른쪽 Image의 경우 ZoomOutButtonRollo를 '50'으로 설정하였으며 마우스 커서를 올리면 왼쪽 Image에 비하여 더 선명해 진것을 확인.

5. AmSerialChartComposite Property - PlotArea .

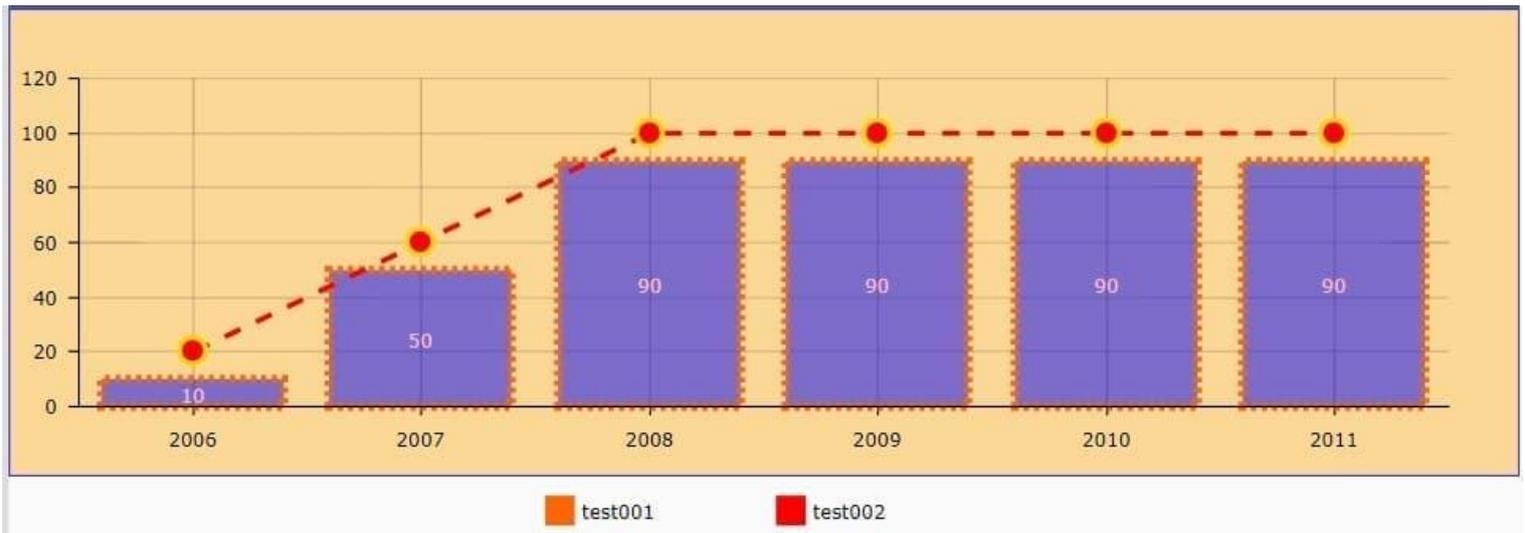


Graph의 그림 영역에 대한 설정 가능.

위 Image의 경우 Graph의 plotAreaFillColors 를 'violet'으로 설정하여 배경 색상이 보라색으로 바뀐 것을 확인.

Graph의 plotAreaBorderColor를 'red'로 설정한 경우 그림 영역에 붉은색 테두리가 생긴 것을 확인.

6. AmSerialChartComposite Property - BackGround .



Graph의 전체 배경 영역에 대한 설정 가능.

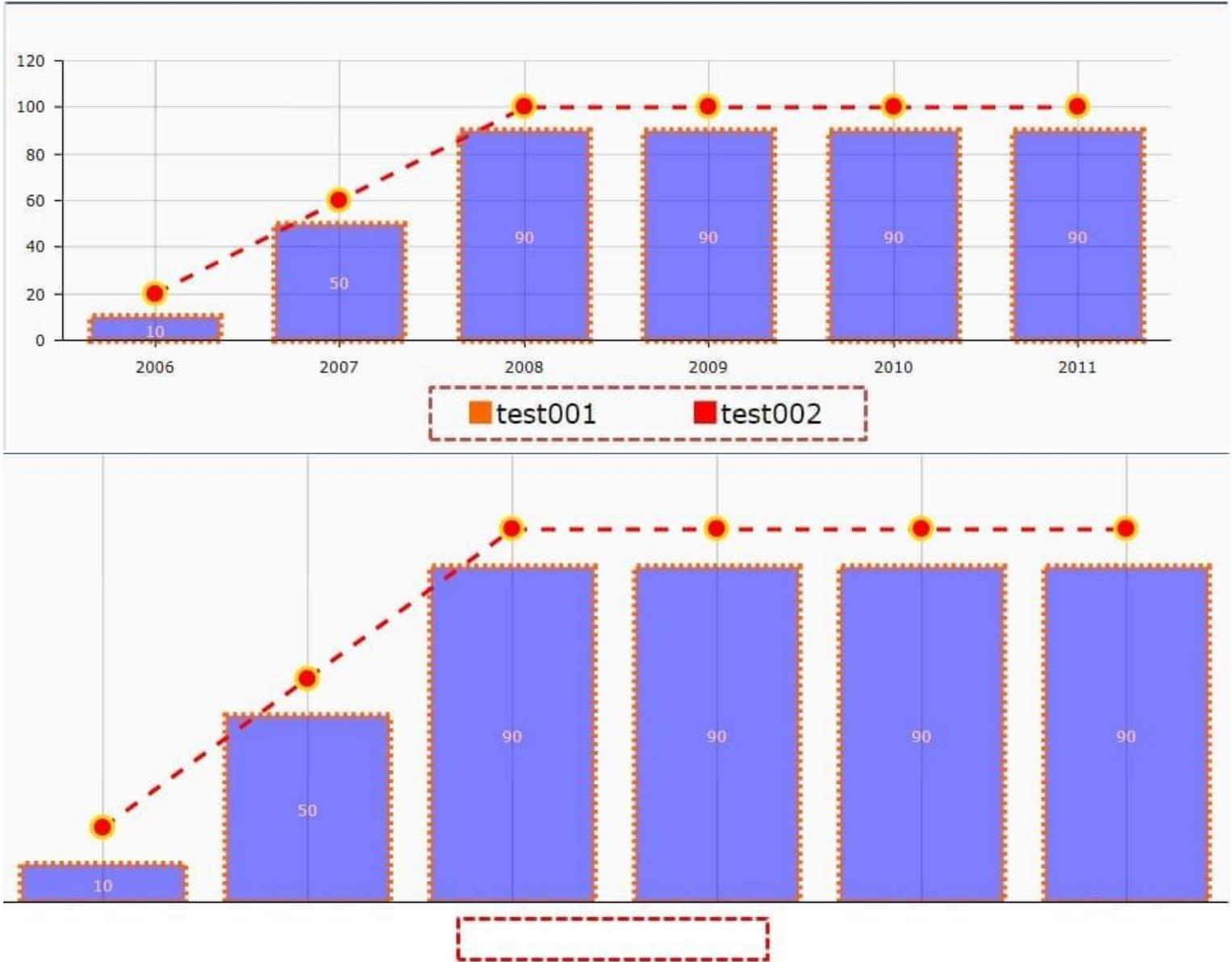
위 Image의 경우 Graph의 BackGroundColor를 'orange'으로 설정하여 전체 배경 색상이 주황색으로 바뀐 것을 확인.

Graph의 BackgroundColor 를 'blue'로 설정한 경우 전체 가장자리에 파란색 테두리가 생긴 것을 확인.

7. AmSerialChartComposite Property - Legend .

Graph의 Legend 영역에 대한 설정 가능.

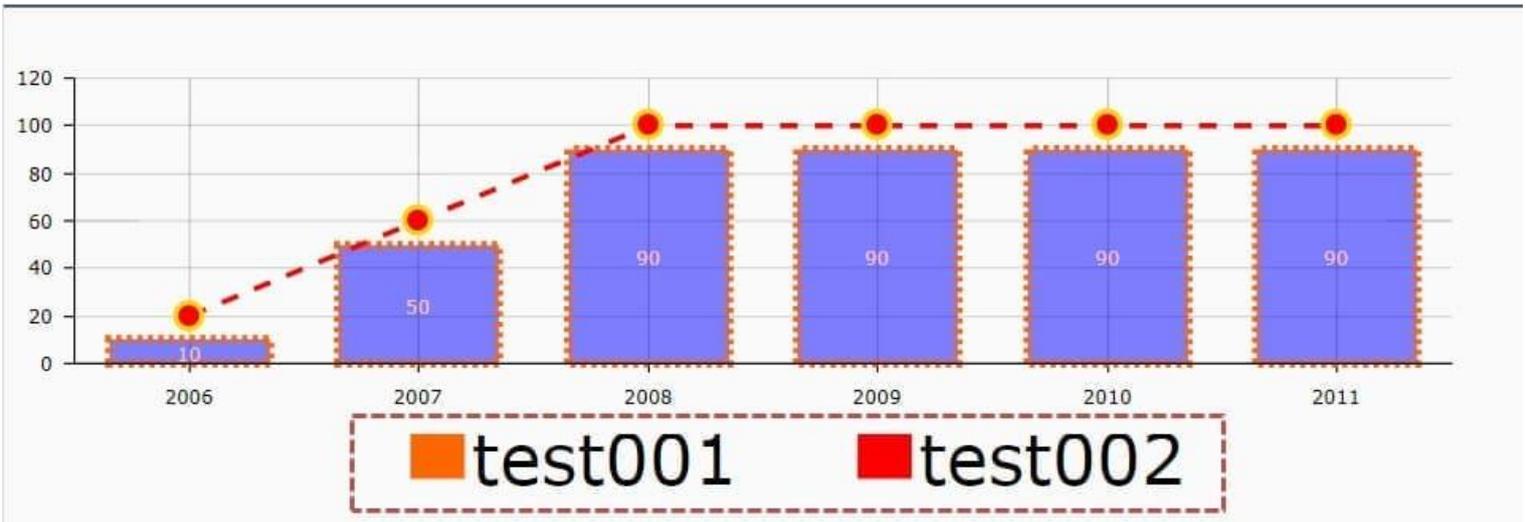
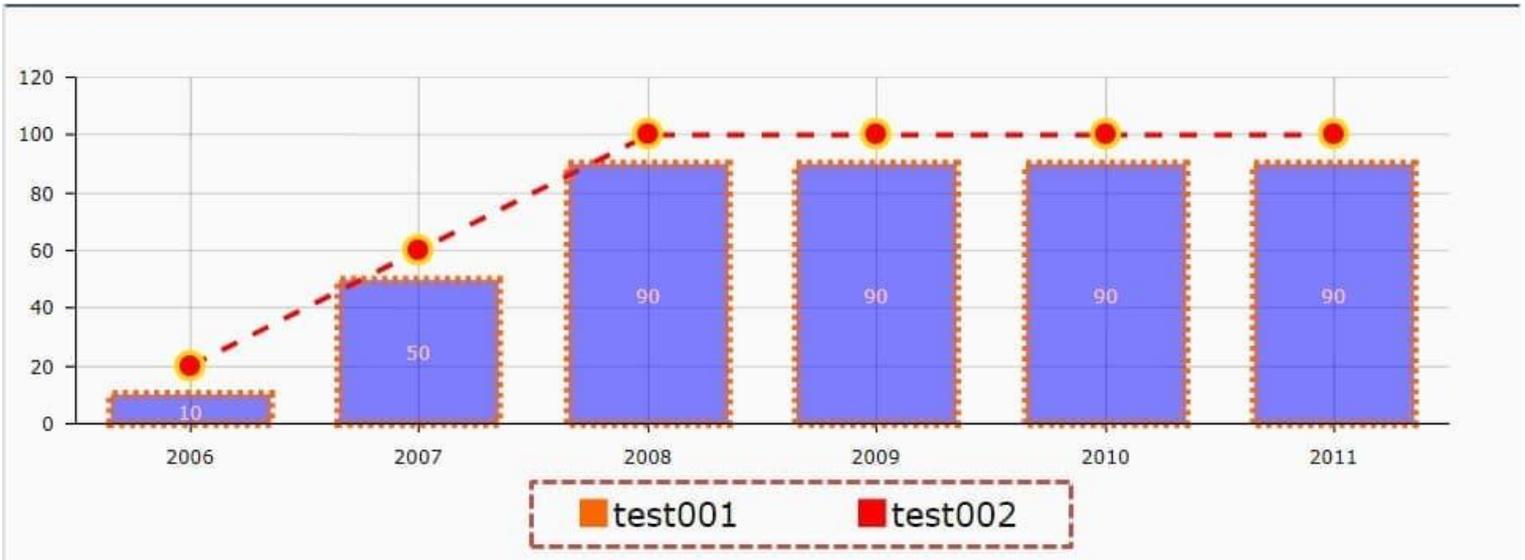
1) ShowLegend .



왼쪽 Image의 경우 ShowLegend 값을 'true'로 설정하였으며 Legend 영역이 보이는 것을 확인.

오른쪽 Image의 경우 ShowLegend 값을 'false'로 설정하였으며 Legend 영역이 사라진 것을 확인.

2) LegendMarkerSize & LegendFontSize.

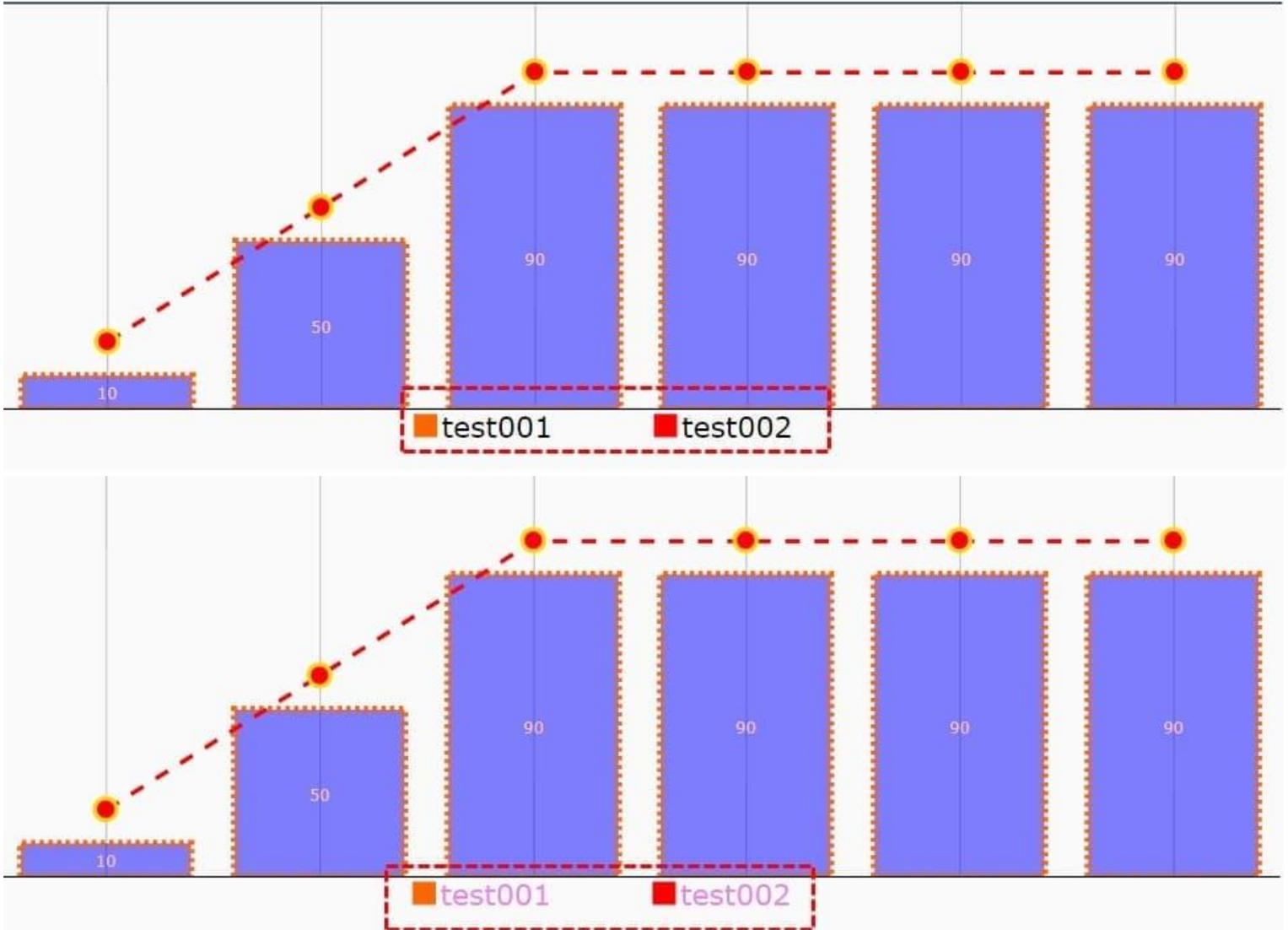


왼쪽 Image의 경우 LegendMarkerSize를 '50', LegendFontSize를 '50'으로 설정.

오른쪽 LegendMarkerSize를 '50'에서 '100'으로 설정했을 경우 범례 Icon의 크기가 바뀐 것을 확인.

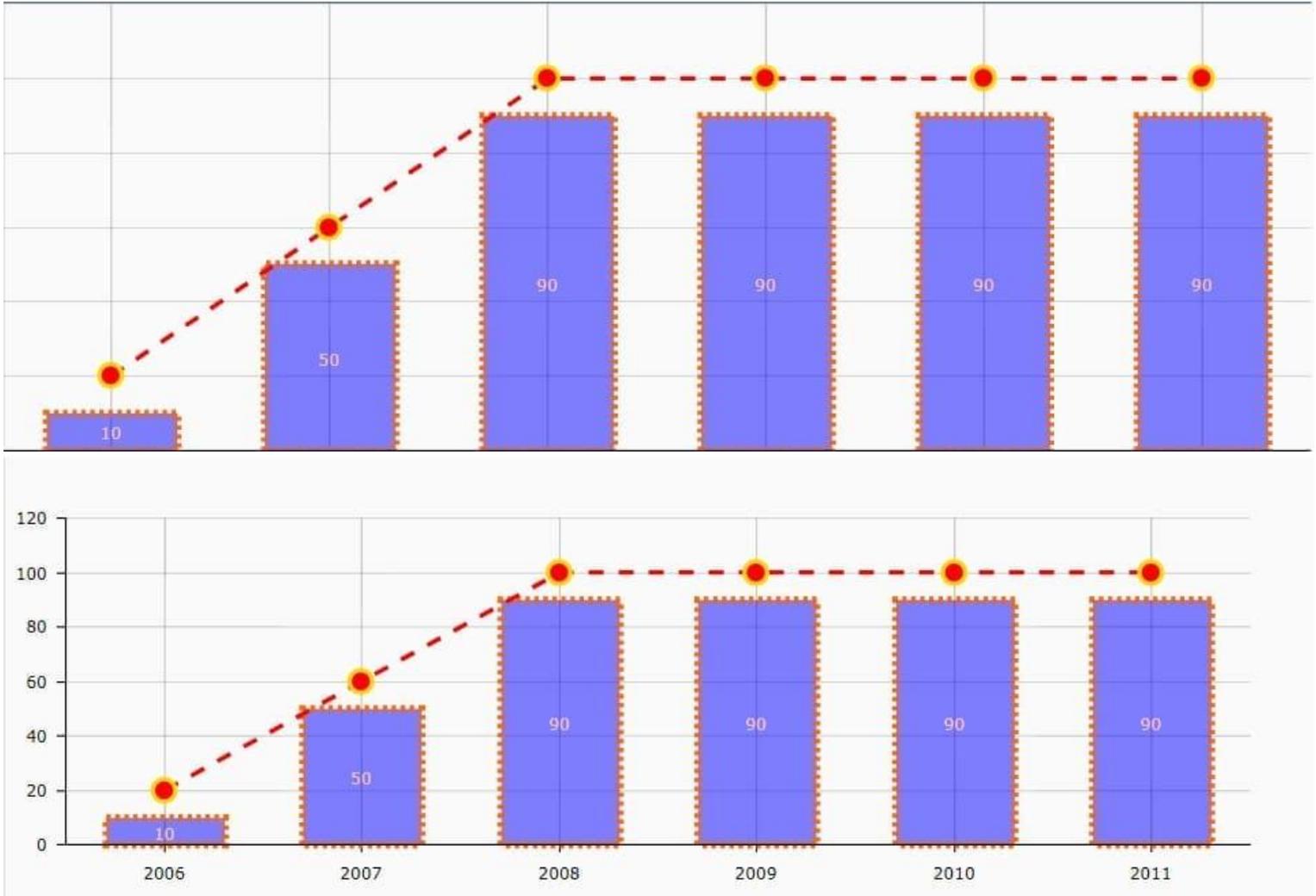
마찬가지로 Image에서 LegendFontSize를 '50'에서 '100'으로 설정했을 경우 범례 Description Text의 크기가 바뀐 것을 확인.

3) LegendColor .



LegendColor 값을 'violet'으로 설정하였으며 좌측 Image의 범례 Text가 우측 Image와 같이 보라색으로 변경된 것을 확인 .

8. AmSerialChartComposite Property - Margin .

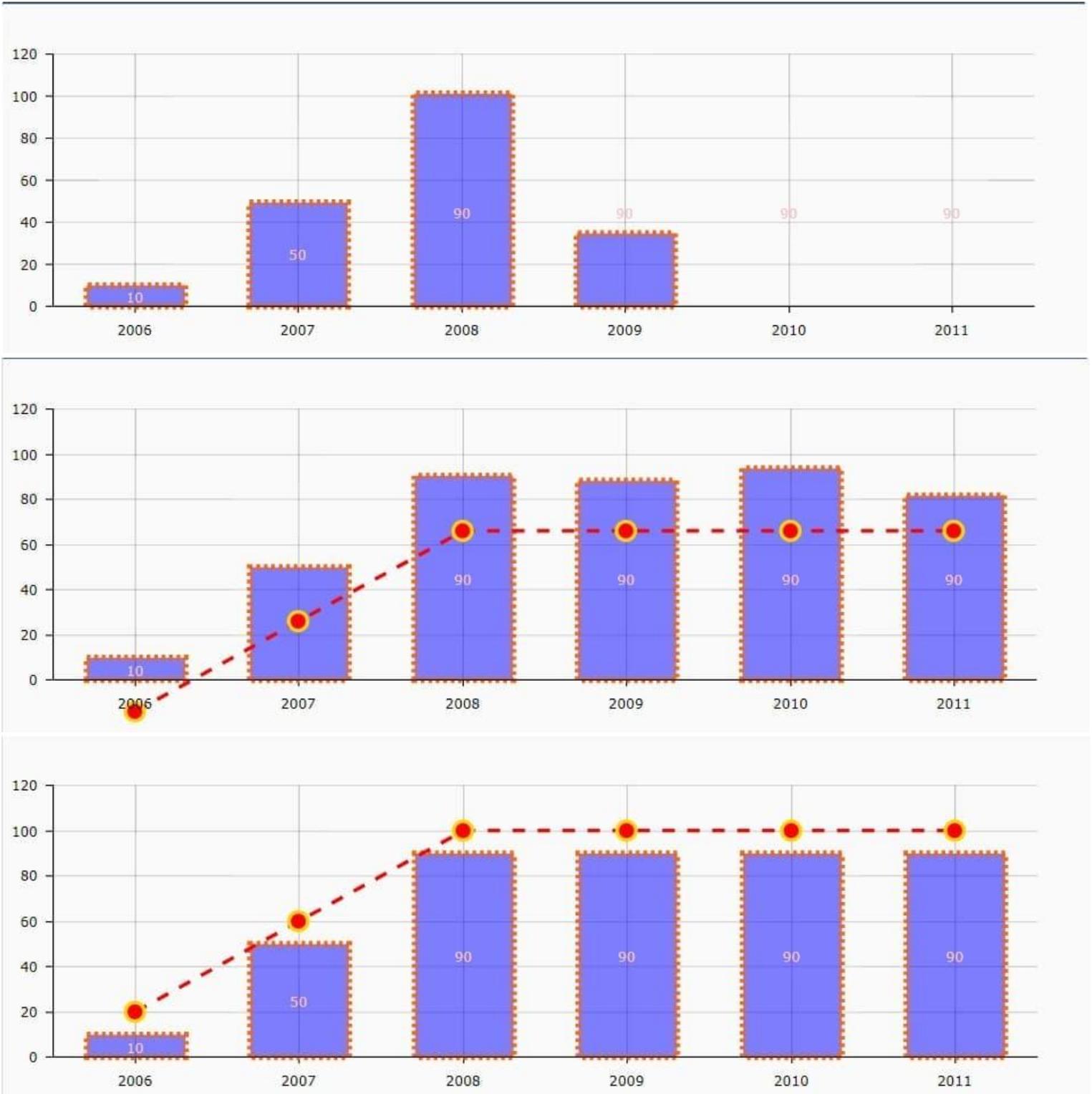


Graph의 여백에 대한 설정 가능 .

위 Image와 같이 상화좌우 Margin 값을 설정하지 않은 경우와 Margin 값을 설정한 경우 여백 여부에 의해 보이는 영역이 다른 것을 확인 .

AutoMargins에 'True' 값을 설정하는 경우 자동으로 적절한 여백이 설정 .

9. AmSerialChartComposite Property - StartDuration .



Graph가 등장하는데 걸리는 시간을 설정 가능.

StartDuration 값을 설정해주면 해당 값 만큼의 시간에 걸쳐 위 Image와 같이 좌측부터 우측 순서대로 순차적으로 Graph가 등장.

9.1.2 FieldCat

[첨부 ABAP SOURCE (SET_CHART_FCAT)]

```
METHOD SET_CHART_FCAT.
```

```
DATA : LS_FCAT TYPE /U4A/CL_U099988=>TY_FCAT.
```

```
*-----*
```

```
REFRESH ET_FCAT.
```

```
LS_FCAT-VALUEFIELD      = 'WEK01'.      "필드명
LS_FCAT-TYPE             = 'column'.      "필드타입("column", "step" "line", "smoothedLine")
LS_FCAT-LINEALPHA        = '1'.          "막대 테두리 색상 투명도(0~1 소숫점 허용가능).
LS_FCAT-LINECOLOR        = '#40E0C0'.    "막대 테두리 색상.
LS_FCAT-LINETHICKNESS    = '5'.          "라인 두께
LS_FCAT-BULLET           = 'round'.      "표시
```

```
점 type("none", "round", "square", "triangleUp", "triangleDown", "bubble", "custom").
```

```
LS_FCAT-BULLETSIZE      = '10'.          "표시점 크기
LS_FCAT-BULLETCOLOR     = '#40E0C0'.    "표시점 색상
LS_FCAT-LABELPOSITION   = 'right'.      "라벨 위
```

```
치.("bottom", "top", "right", "left", "inside", "middle")
```

```
LS_FCAT-SHOWVALUETEXT   = 'X'.          "막대의 값 TEXT 보일지 여부('X' = 값을 표현함)
LS_FCAT-FONTSIZE        = '15'.          "TEXT FONT크기(숫자)
LS_FCAT-COLOR           = '#40E0C0'.    "라벨 색상.
LS_FCAT-AXISVALUEFIELD  = 'WEK02'.      "AXIS를 사용할 다른 VALUEFIELD명.
```

```
APPEND LS_FCAT TO ET_FCAT.
```

```
CLEAR LS_FCAT.
```

```
ENDMETHOD .
```

PROPERTIES	MEAN
VALUEFIELD	Field 명
OPENFIELD	Low값 Field 명
TYPE	Field Type("column", "step", "line", "smoothedline")
TITLE	Field Desc
BALLOONTEXT	말풍선 Text
FILLCOLORS	막대 내부 색상
FILLALPHAS	막대 내부 색상 투명도 (0~1 소숫점 허용가능)
LINEALPHA	막대 테두리 색상 투명도 (0~1 소숫점 허용가능)
LINECOLOR	막대 테두리 색상
LINETHICKNESS	라인 두께
BULLET	표시점 type("none", "round", "square", "triangleUp", "triangleDown", "bubble", "custom")
BULLETSIZE	표시점 크기
BULLETCOLOR	표시점 색상
BULLETBORDERCOLOR	표시점 테두리 색상
BULLETBORDERALPHA	표시점 테두리 투명도
BULLETBORDERTHICKNESS	표시점 테두리 두께
DASHLENGTH	라인 점선(숫자가 클수록 점선 간격이 길어짐)
DASHLENGTHFIELD	라인 점건 바인딩 필드
COLOR	라벨 색상
LABELCOLORFIELD	라벨 색상 바인딩 필드

U4A IDE HELP

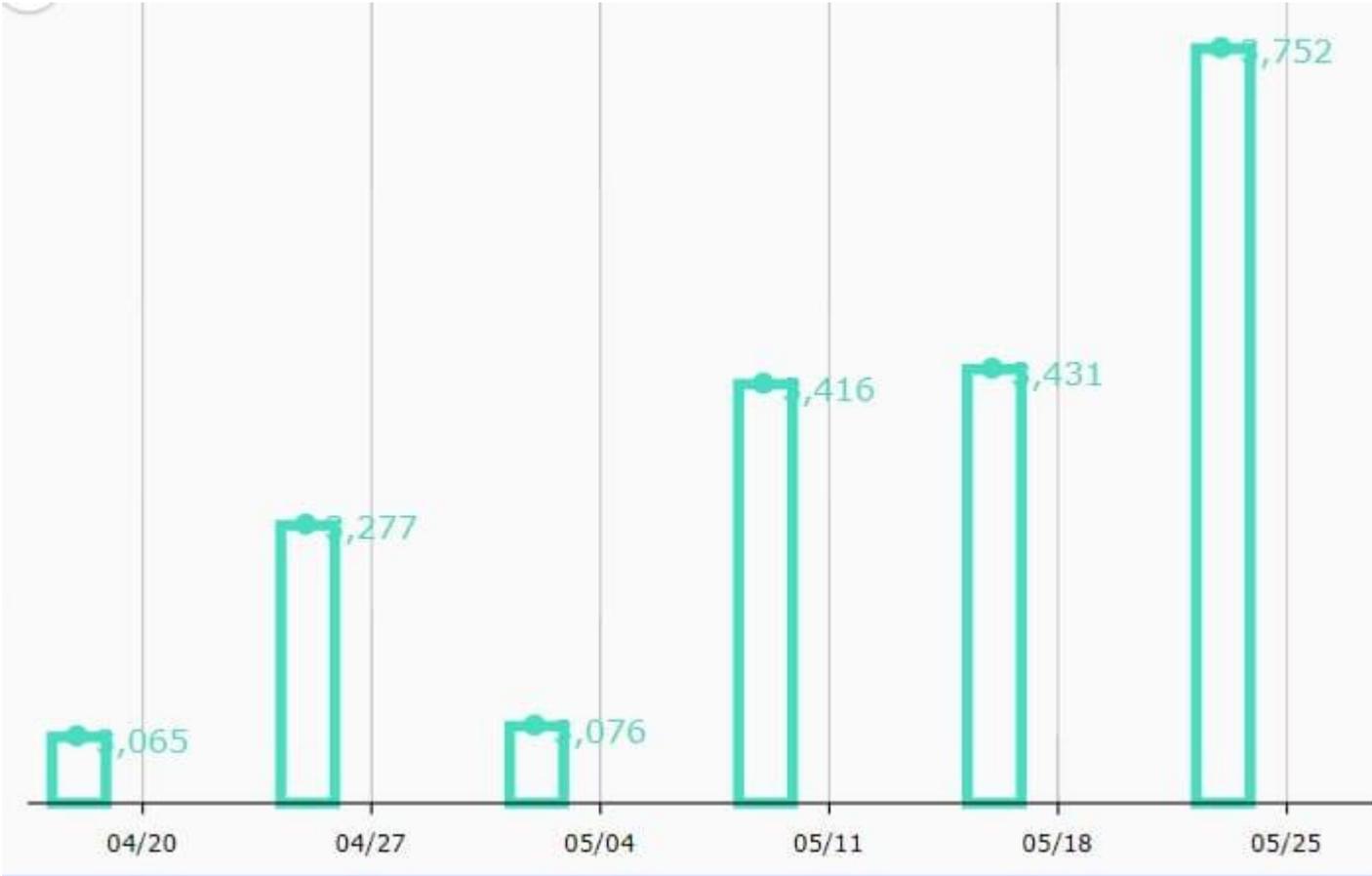
LABELPOSITION	라벨 위치.("bottom", "top", "right", "left", "inside", "middle")
LABELTEXT	라벨 Text(폐기처리됨 -> ShowValueText 필드 사용)
FONTSIZE	Text Font크기 (숫자)
AXISVALUEFIELD	Axis를 사용할 다른 ValueField명
LINECOLORFIELD	막대 테두리 색상 바인딩 필드
ALPHAFIELD	색상 투명도 바인딩 필드
CORNERRADIUSTOP	라운드 설정값(숫자)
TOPRADIUS	막대 원통으로 처리 값 - Chart의 Angle, Depth3D Property를 설정해야 적용 됨
SHOWVALUETEXT	막대의 값 Text 보일지 여부('X' = 값을 표현함)
HIDEBALLOON	말풍선 표시여부 ('X' = 말풍선 비활성)
DISCONNERCT	Category에 Graph 값이 없는 경우 표현 안함 여부('X' = 표현안함)
LABELTEXTFIELD	라벨 바인딩 필드
BULLETFIELD	표시점 Type 바인딩 필드
BULLETSIZEFIELD	표시점 크기 바인딩 필드
COLORFIELD	색상 바인딩 필드
FIXEDCOLUMNWIDTH	막대 두께 값(숫자)
HIDEBULLETSCOUNT	비활성처리 표시점 숫자(차트에 보여지는 표시점이 입력 숫자를 초과하면 비활성 처리됨)
LABELOFFSET	라벨 여백
LABELROTATION	라벨 기울기
LEGENDCOLOR	범례 색상
LEGENDALPHA	범례 색상 투명도
MARKERTYPE	범례 타입(square, circle, diamond, triangleUp, triangleDown, triangleLeft, triangleRight, bubble, line, none)

SHOWHANDONHOVER	막대(선에) 마우스를 갖다 대는 경우 마우스 아이콘이 손모양으로 변경할지 여부('X' = 손모양)
DESWITCHABLE	범례 아이콘 선택 가능 ('X' = 선택 불가능 처리)
INVISIBLEINLEGEND	범례 아이콘 비활성 여부 ('X' = 비활성)
XFIELD	x 차트 전용 X축 바인딩 필드
YFIELD	x 차트 전용 Y축 바인딩 필드

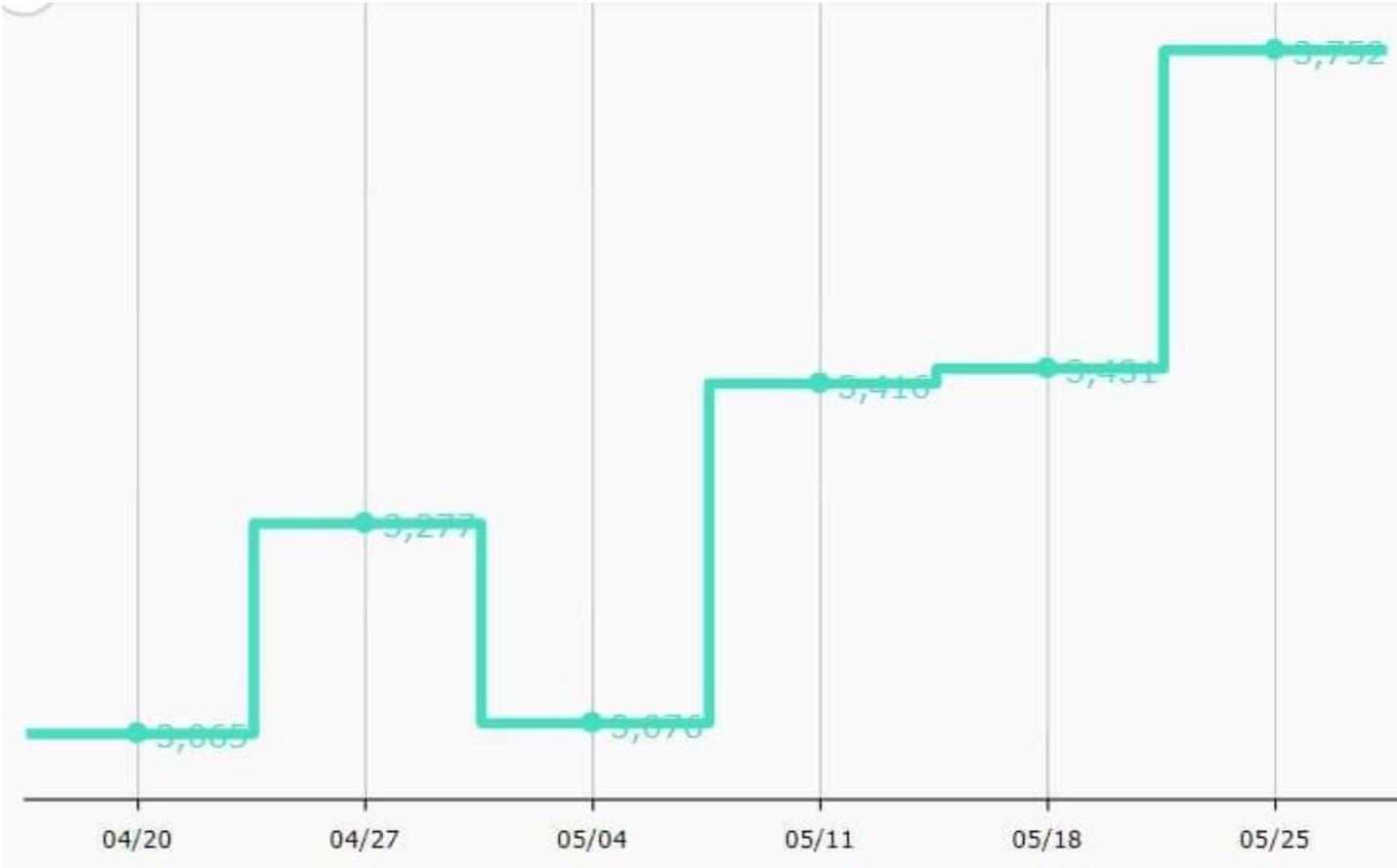
1. AmSerialChartComposite FieldCat - Type.



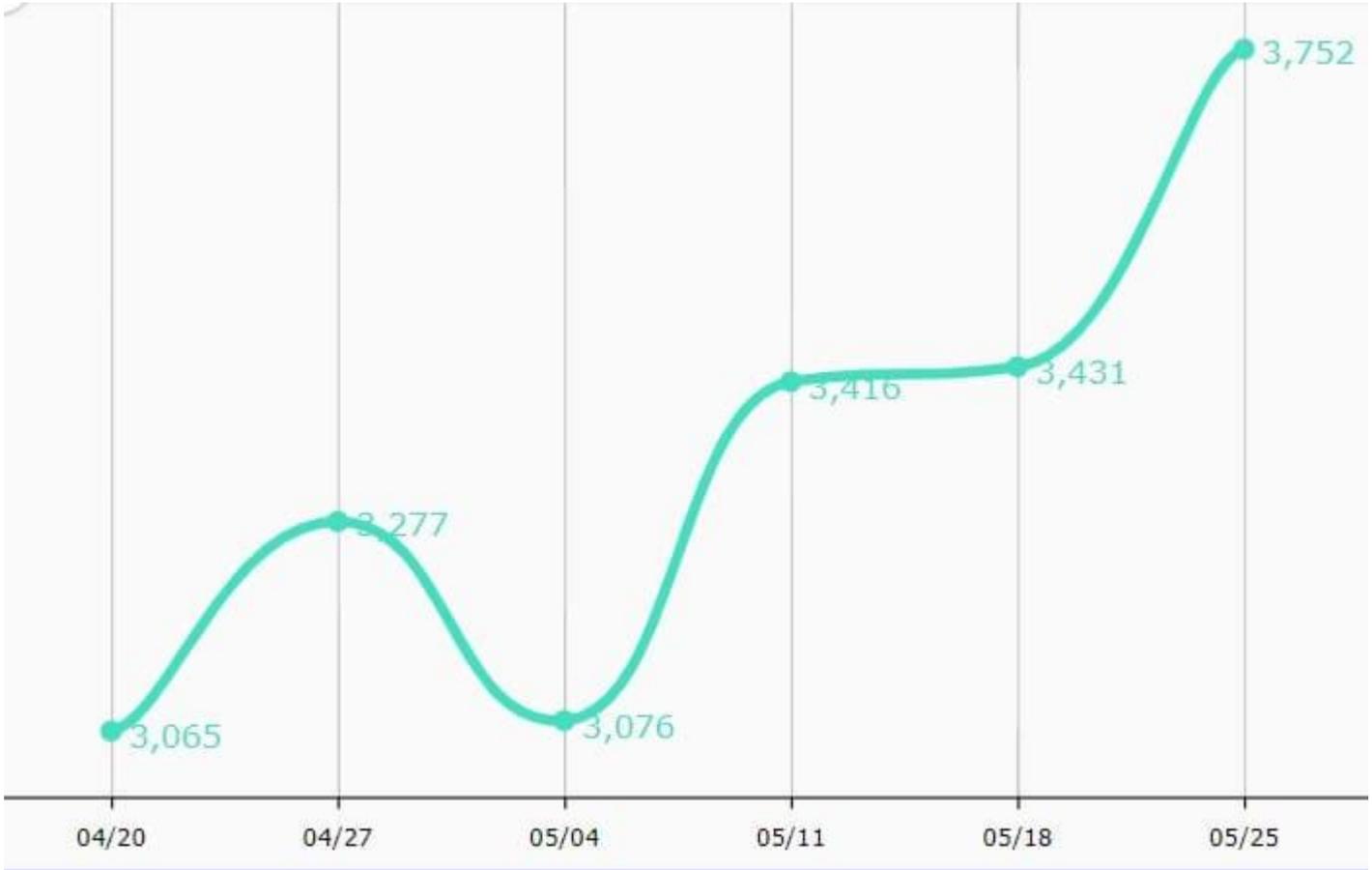
< line >



< column >



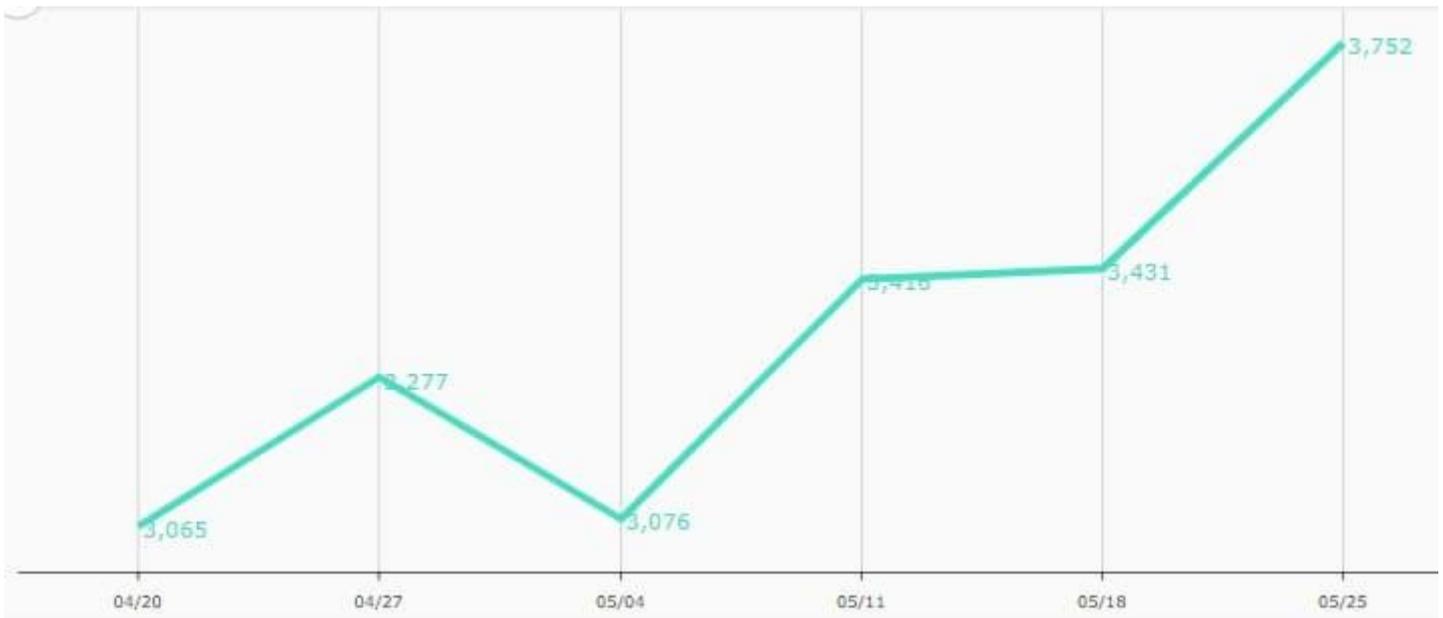
< step >



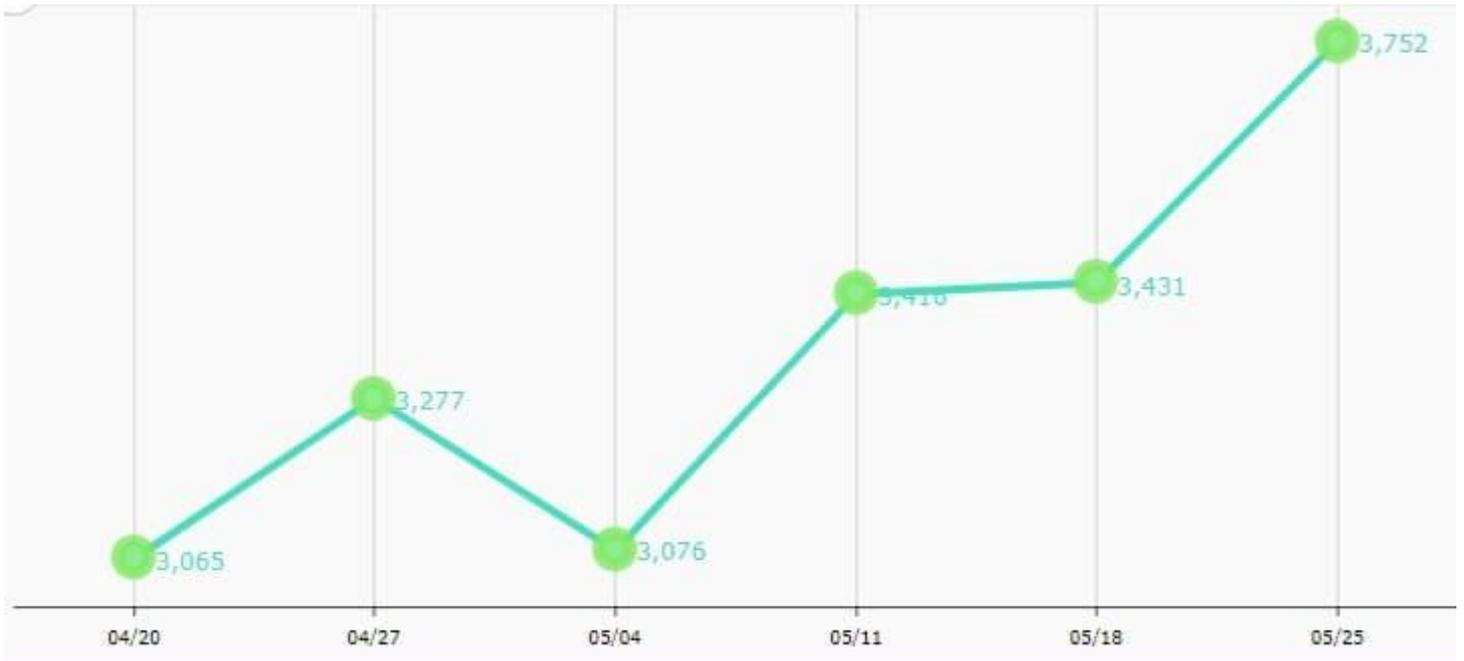
< smoothedline >

표현하고자 하는 Graph의 유형을 선택 가능.

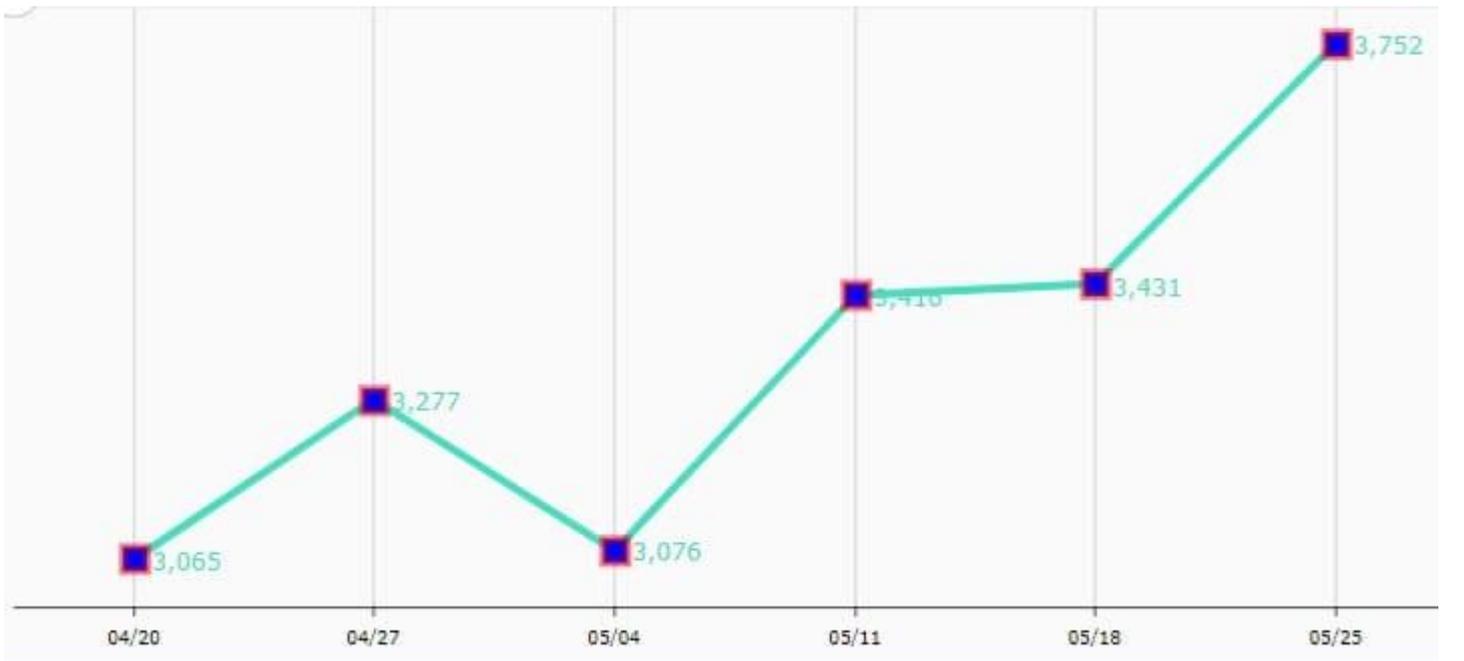
2. AmSerialChartComposite FieldCat - Bullet .



< none >

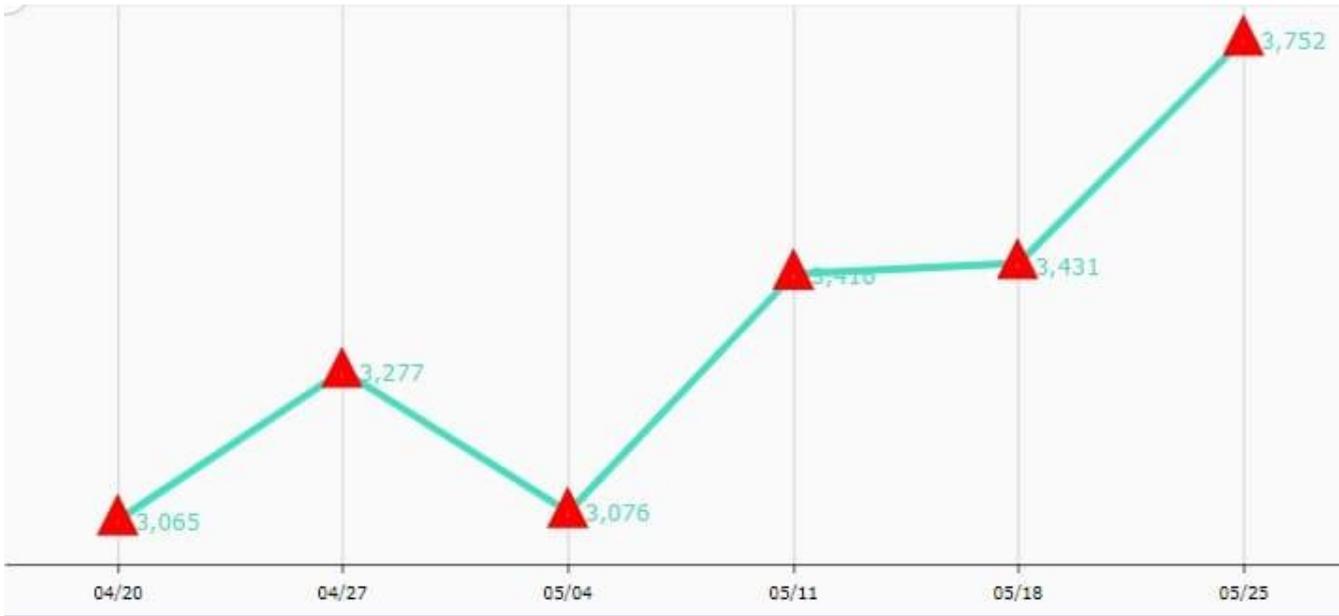


< round >

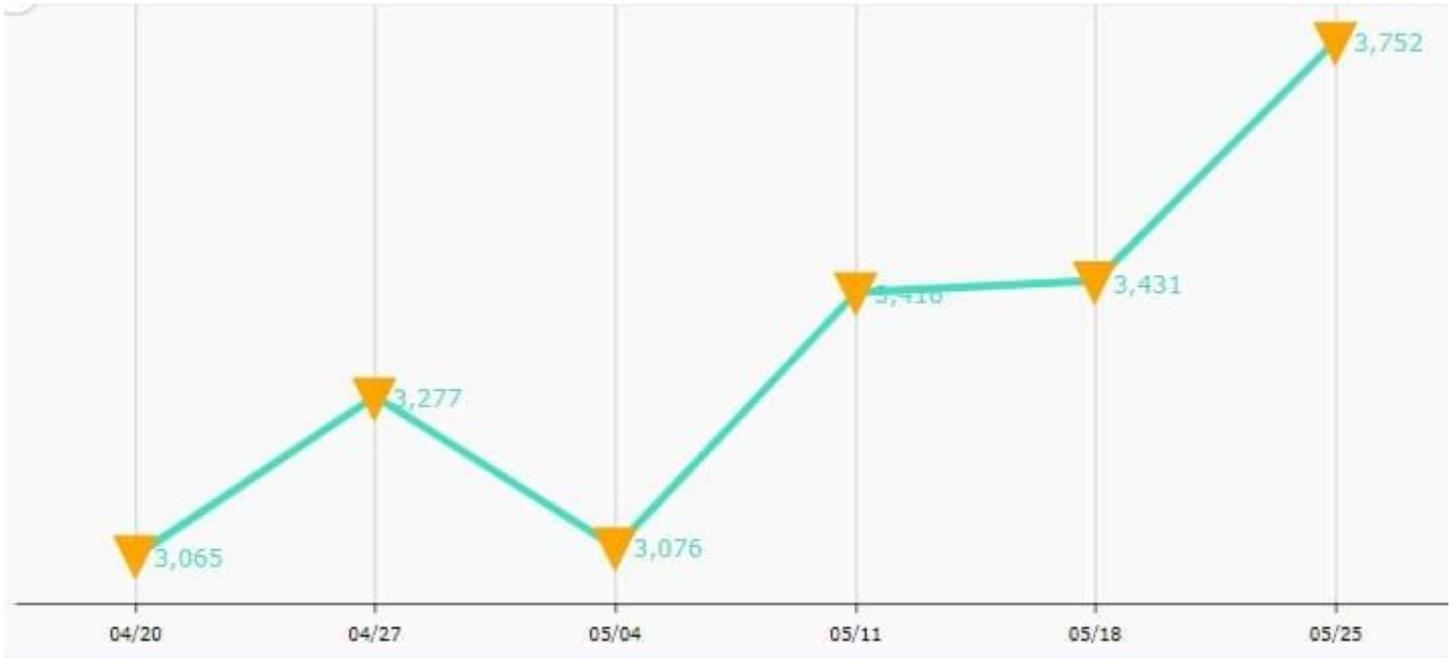


< square >

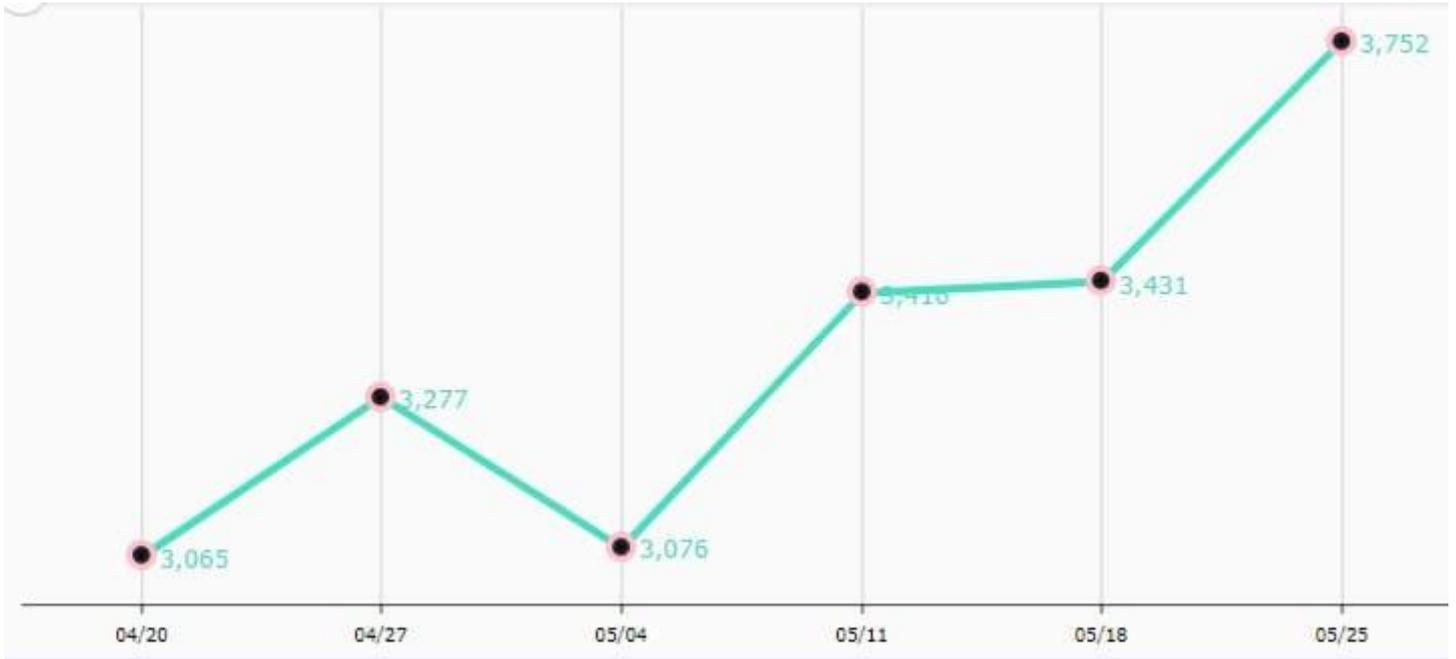
U4A IDE HELP



< triangleUp >



< triangleDown >



< bubble >

표시점의 색상 (BulletColor), 모양(Bullet), 크기(BulletSize), 테두리 색상(BulletBorderColor), 두께 (BulletBorderThickness), 투명도(BulletBorderAlpha) 설정 가능.

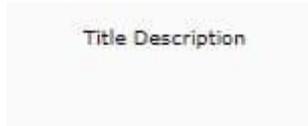
3. AmSerialChartComposite FieldCat - Legend.

< square, circle, diamond, triangleUp, triangleDown >



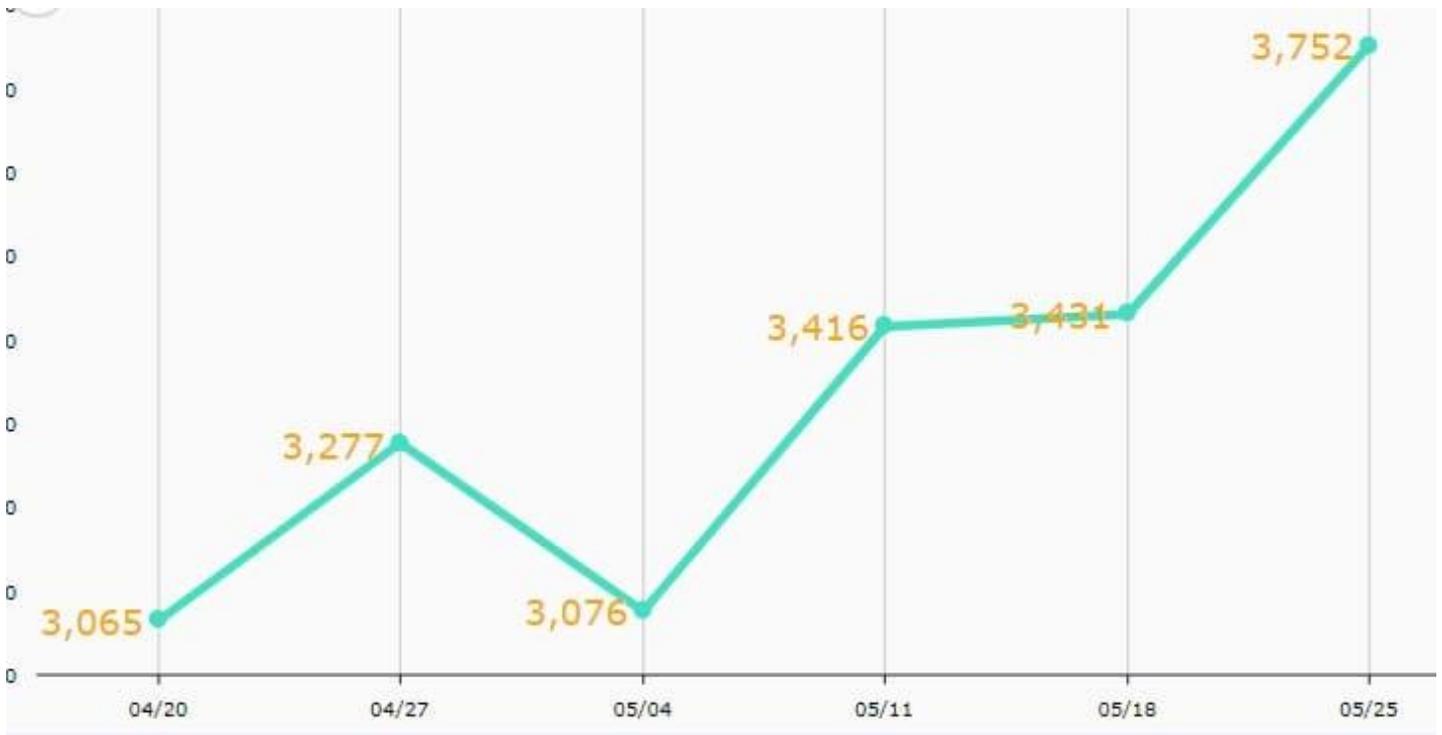
< triangleLeft, tringleRight, bubble, line, none >

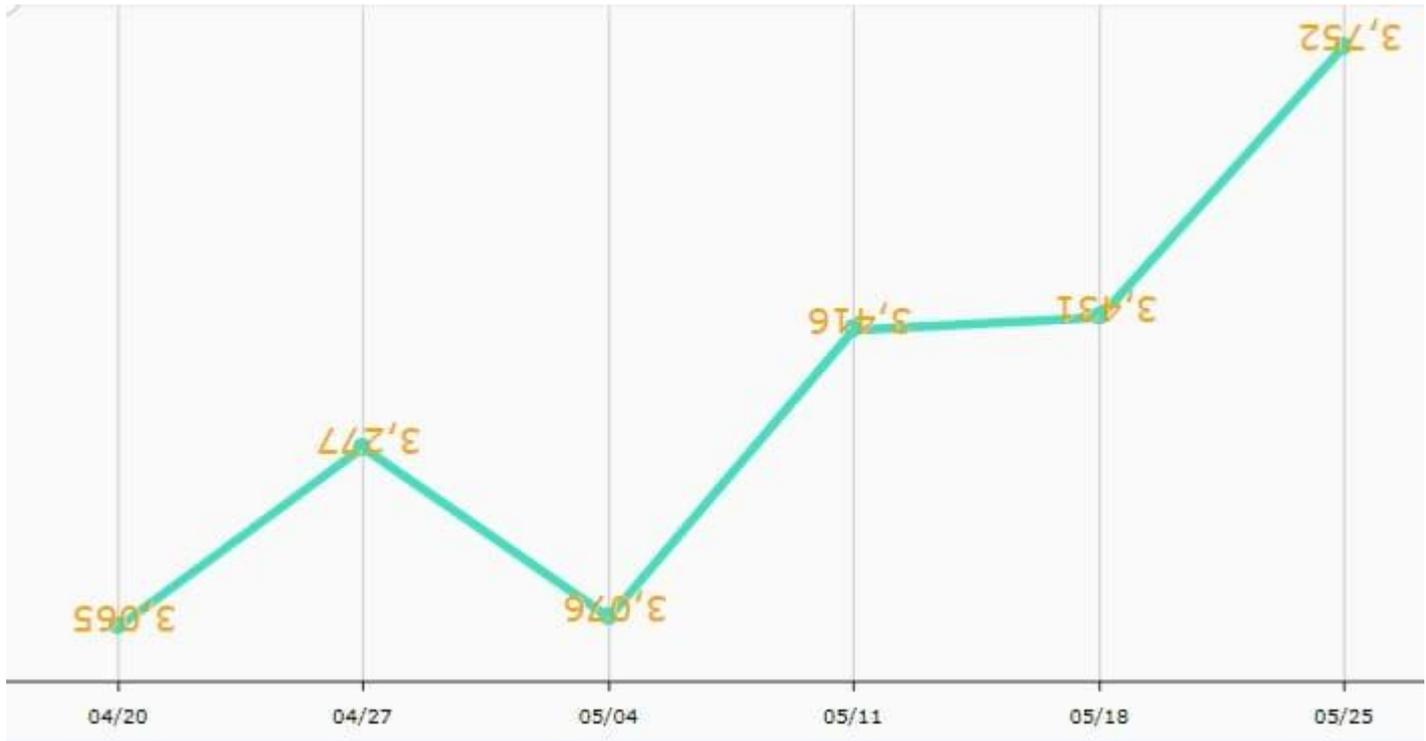




범례 Icon의 색상(LegendColor), 모양(MarkerType), 투명도(LegendAlpha), 선택 가능 여부(Deswitchable), 활성화 여부(InvisibleInLegend) Text(Title) 설정 가능.

4. AmSerialChartComposite FieldCat - Label .





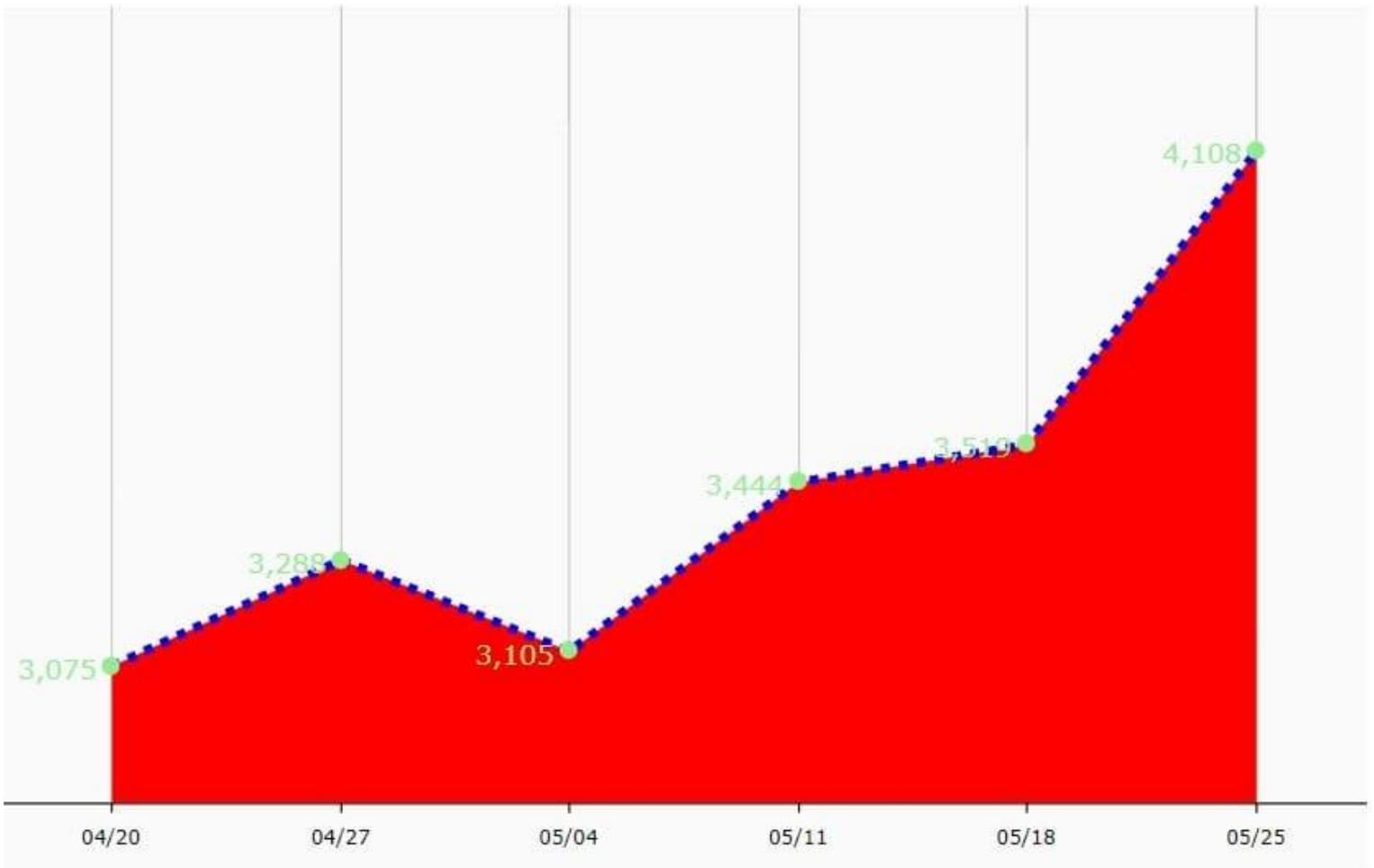
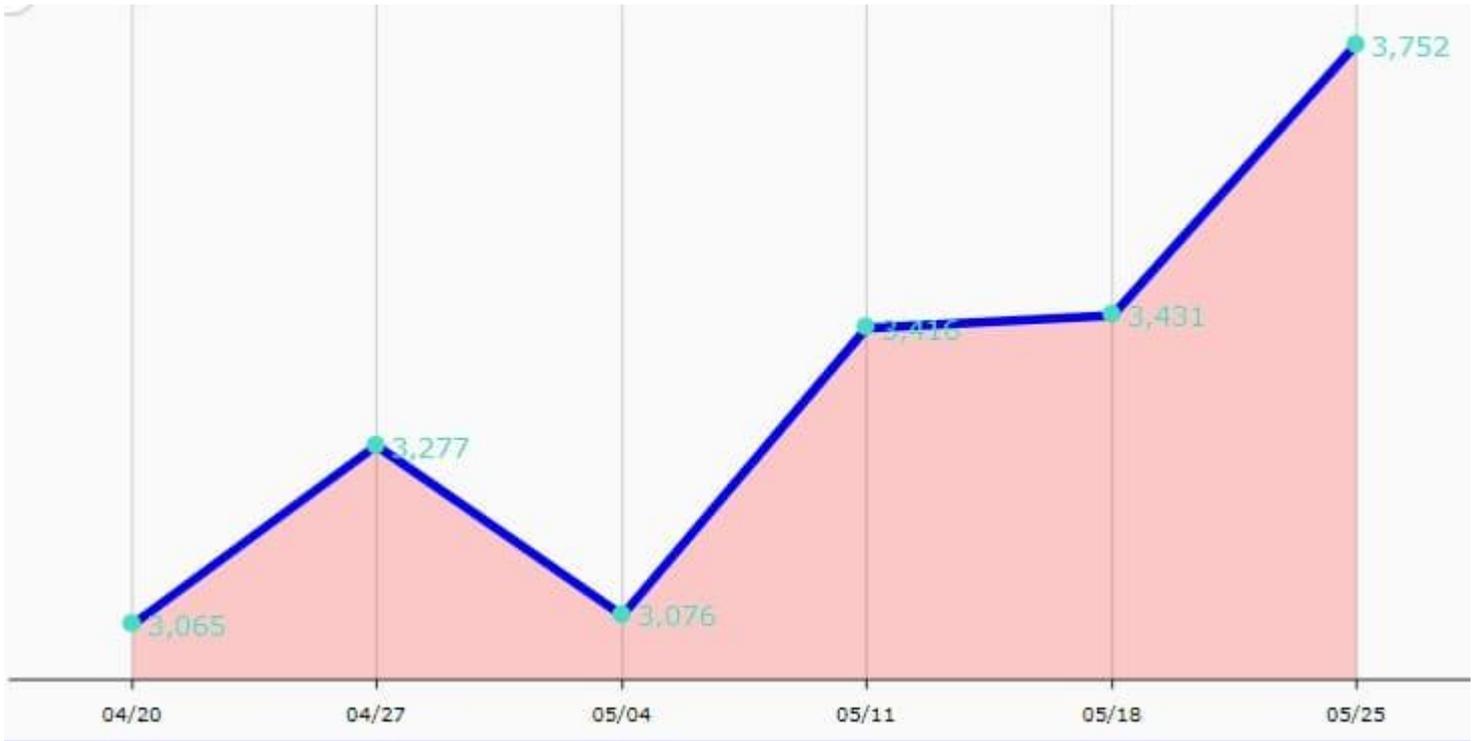
라벨 색상(Color), 위치(LabelPosition), 라벨 크기(FontSize) 설정이 가능하며 기울기(LabelRotation) 및 여백 설정(LabelOffset) 가능 .

왼쪽 Image의 경우 LabelRotation을 0으로 설정하였고 오른쪽 Image의 경우 LabelRotation을 180으로 설정 .

오른쪽 Image의 Label이 180도 만큼 회전 된 것을 확인 .

5. AmSerialChartComposite FieldCat - Fill.

U4A IDE HELP



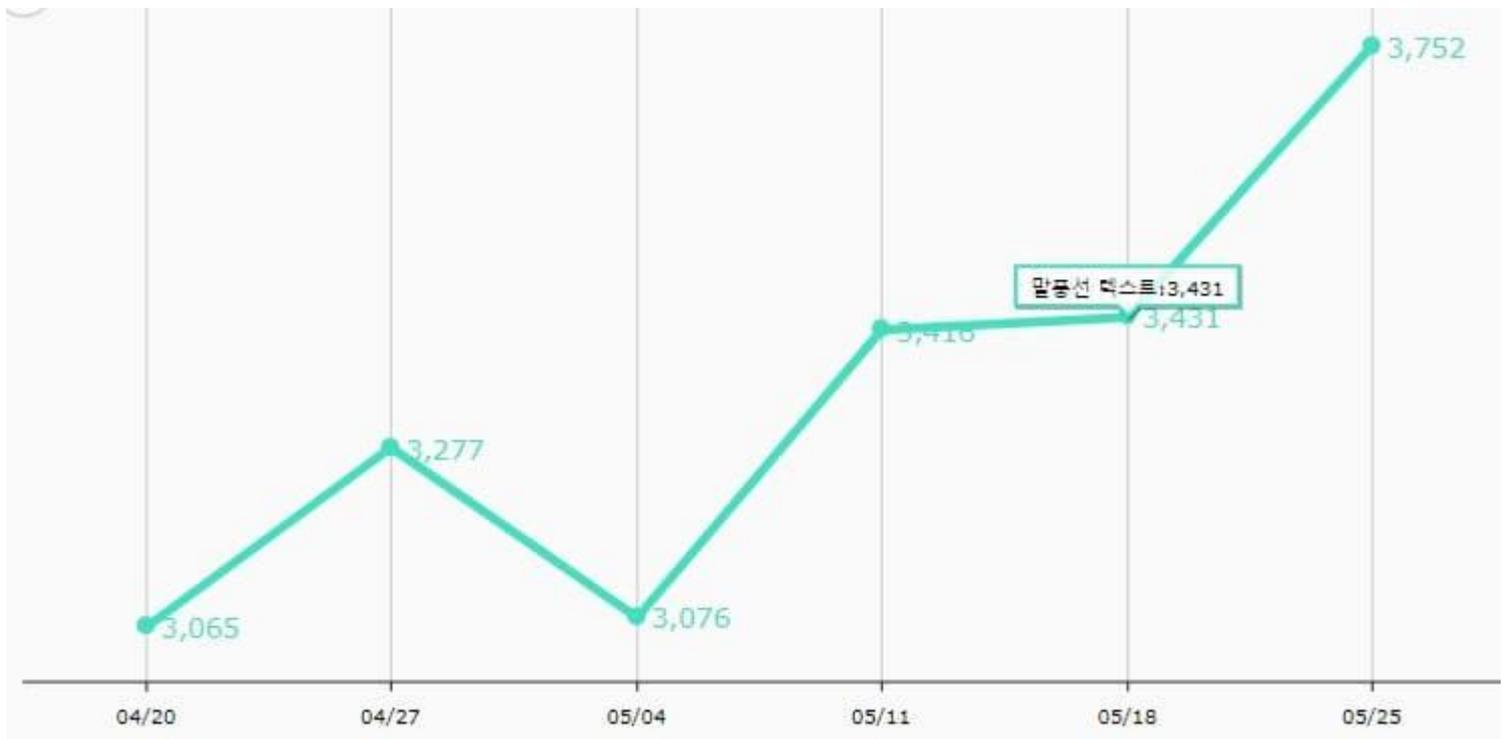
막대 내부 색상(FillColors), 막대 내부 색상 투명도(FillAlphas), 테두리 색상(LineColor), 막대 테두리 색상 투명도(LineAlpha), 막대 두께(FixedColumnWidth),

막대 테두리 두께(LineThickness), 테두리 점선 표현(DashLength), 설정 가능.

첫번째 Image의 경우 LineAlpha : 1, LineThickness : 5, DashLength : 0, FillColors : red, LineColor : blue, FixedColumnWidth : 10, FillAlphas : 0.3 .

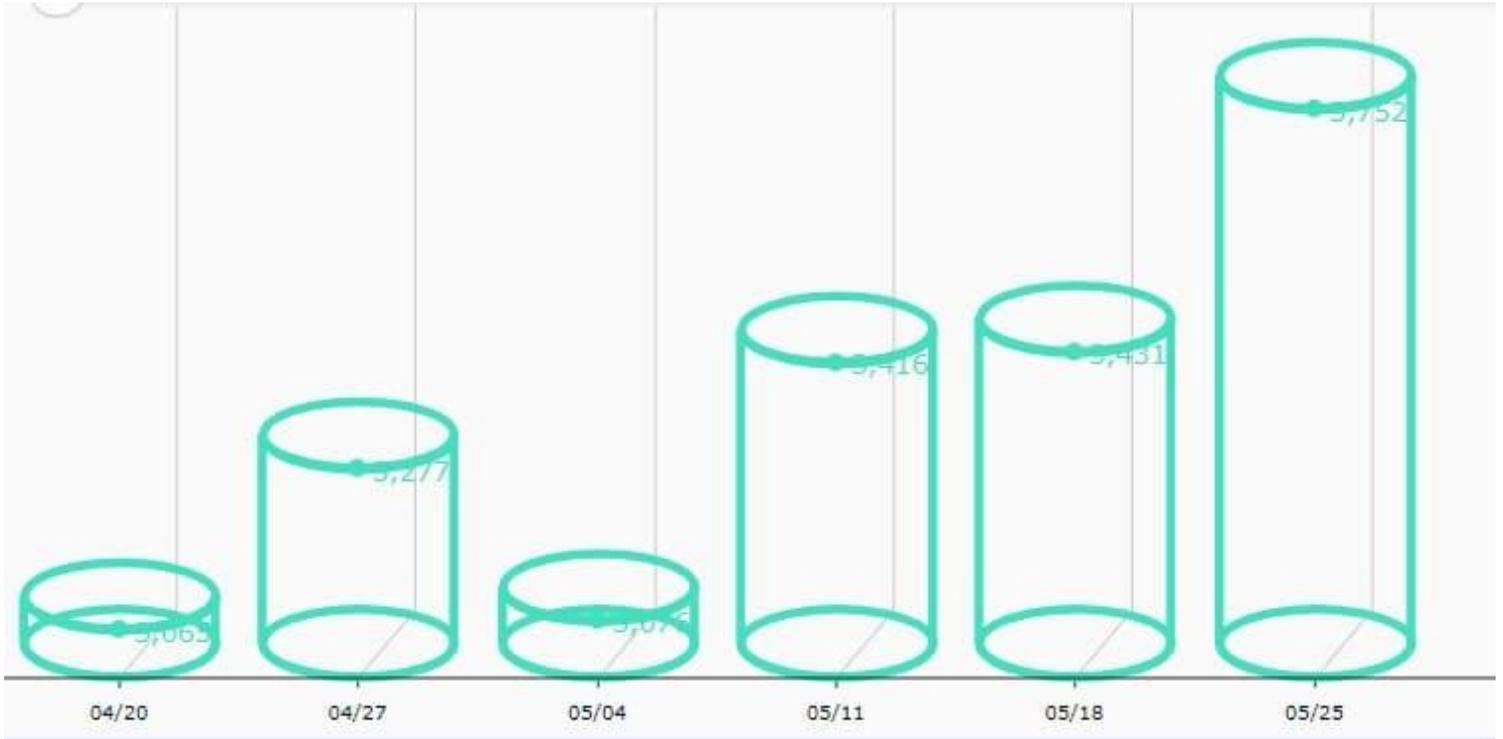
두번째 Image의 경우 LineAlpha : 1, LineThickness : 5, DashLength : 5, FillColors : red, LineColor : blue, FixedColumnWidth : 10, FillAlphas : 1 .

6. AmSerialChartComposite FieldCat - Ballon.



말풍선 생성여부(HideBalloon) 및 원하는 Text(BalloonText)를 설정할 수 있으며 Graph에 마우스 커서를 올리면 말풍선이 생기는 것을 확인 .

7. AmSerialChartComposite FieldCat - TopRadius .



Chart의 Angle, Depth3D Property를 설정해야 적용이 되며 Graph Type이 'Column' 인 경우에만 원통으로 표현되는 것을 확인 .

8. AmSerialChartComposite FieldCat - Binding Field.

Graph는 CategoryValue를 Header 성격으로 갖고 GraphValue를 Item 성격으로 가지고 있는 구조.

Dictionary: Display Structure

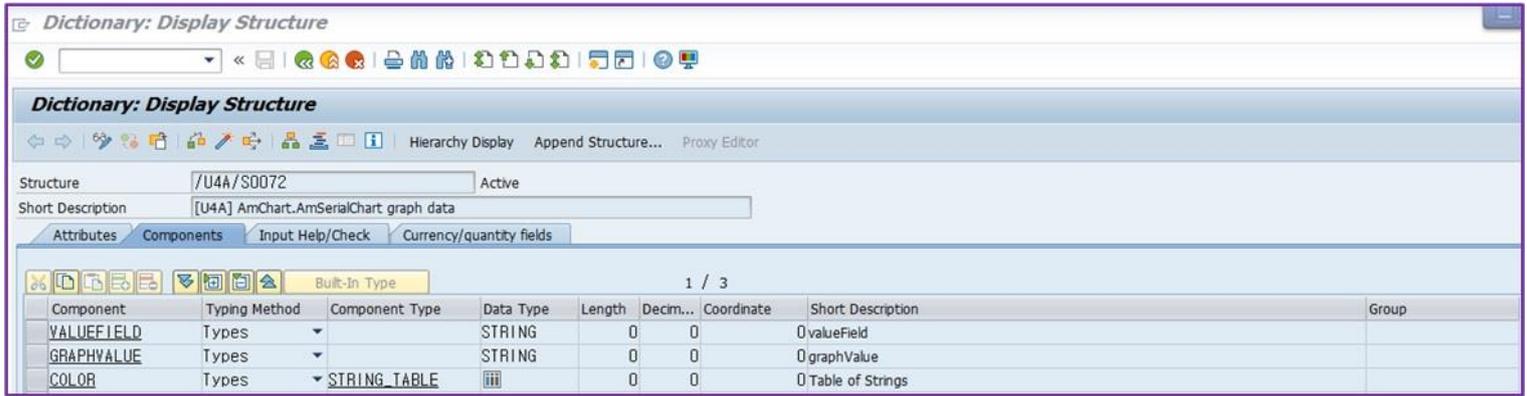
Structure: /U4A/S0071 Active
 Short Description: [U4A] AmChart.AmSerialChart 데이터 구조

Attributes Components Input Help/Check Currency/quantity fields

Built-In Type 1 / 2

Component	Typing Method	Component Type	Data Type	Length	Decim...	Coordinate	Short Description	Group
CATEGORYVALUE	Types		STRING	0	0		CategoryValue	
GRAPHVALUE	Types	/U4A/Y0072		0	0		[U4A] AmChart.AmSerialChart graph data	

FieldCat의 특정 Field에 대한 값을 구성하고자 하면 FieldCat에서 설정한 Field명과 동일하게 ValueField에 설정하고 원하는 값을 GraphValue에 삽입.



[첨부 ABAP SOURCE (SET_CHART_DATA)]

```
METHOD /U4A/IF_SERVER~HANDL_ON_INIT.

DATA : LO_CHART TYPE REF TO /U4A/CL_U099988,
       LT_FCAT  TYPE /U4A/CL_U099988=>TT_FCAT,
       LS_FCAT  LIKE LINE OF LT_FCAT,
       LT_DATA  TYPE /U4A/Y0071,
       LS_DATA  TYPE /U4A/S0071,
       LS_GRAPH TYPE /U4A/S0072.
```

```
LO_CHART ?= ME->/U4A/IF_SERVER~AR_VIEW-
```

```
>GET_UI_INSTANCE ( I_ID = 'AMSERIALCHARTCOMPOSITE1' ).
```

```
CHECK LO_CHART IS BOUND.
```

*- Chart FieldCat 구성 *

```
LS_FCAT-VALUEFIELD = 'A01'.
LS_FCAT-TYPE        = 'column'.
LS_FCAT-FILLCOLORS = 'blue'.
LS_FCAT-FILLALPHAS = '0.5'.
LS_FCAT-TITLE       = 'test001'.
LS_FCAT-LABELTEXT   = 'value'.
LS_FCAT-LABELPOSITION = 'middle'.
LS_FCAT-COLOR       = 'pink'.
LS_FCAT-SHOWVALUETEXT = 'X'.
LS_FCAT-FontSize    = '50'.
```

```

LS_FCAT-LABELCOLORFIELD = 'A02'.
LS_FCAT-DASHLENGTH      = 3.
LS_FCAT-LINEALPHA       = 1.
LS_FCAT-LINETHICKNESS   = 5.
APPEND LS_FCAT TO LT_FCAT.
CLEAR LS_FCAT.
    
```

*- Binding Field 값 구성 .

```

LS_GRAPH-VALUEFIELD = 'A01'.
LS_GRAPH-GRAPHVALUE = '50'.
APPEND LS_GRAPH TO LS_DATA-GRAPHVALUE.
CLEAR LS_GRAPH.
    
```

" ValueField Field명.

" Value.

```

LS_GRAPH-VALUEFIELD = 'A02'.
LS_GRAPH-GRAPHVALUE = 'red'.
APPEND LS_GRAPH TO LS_DATA-GRAPHVALUE.
CLEAR LS_GRAPH.
    
```

" Label Color Field명.

" Color.

```

LS_DATA-CATEGORYVALUE = '2006'.
    
```

" 연도

```

APPEND LS_DATA TO LT_DATA.
CLEAR LS_DATA.
    
```

```

LS_GRAPH-VALUEFIELD = 'A01'.
LS_GRAPH-GRAPHVALUE = '60'.
APPEND LS_GRAPH TO LS_DATA-GRAPHVALUE.
CLEAR LS_GRAPH.
    
```

" ValueField Field명.

" Value.

```

LS_GRAPH-VALUEFIELD = 'A02'.
LS_GRAPH-GRAPHVALUE = 'blue'.
APPEND LS_GRAPH TO LS_DATA-GRAPHVALUE.
CLEAR LS_GRAPH.
    
```

" Label Color Field명.

" Color.

```

LS_DATA-CATEGORYVALUE = '2007'.
    
```

" 연도.

```

APPEND LS_DATA TO LT_DATA.
CLEAR LS_DATA.
    
```

```

LS_GRAPH-VALUEFIELD = 'A01'.
LS_GRAPH-GRAPHVALUE = '70'.
APPEND LS_GRAPH TO LS_DATA-GRAPHVALUE.
CLEAR LS_GRAPH.
    
```

" ValueField Field명.

" Value.

```

LS_GRAPH-VALUEFIELD = 'A02'.
LS_GRAPH-GRAPHVALUE = 'green'.
    
```

" Label Color Field명.

" Color.

```
APPEND LS_GRAPH TO LS_DATA-GRAPHVALUE.
CLEAR LS_GRAPH.
```

```
LS_DATA-CATEGORYVALUE = '2008'. " 연도.
```

```
APPEND LS_DATA TO LT_DATA.
CLEAR LS_DATA.
```

*- 차트 Display .

```
CALL METHOD LO_CHART->SHOW
EXPORTING
```

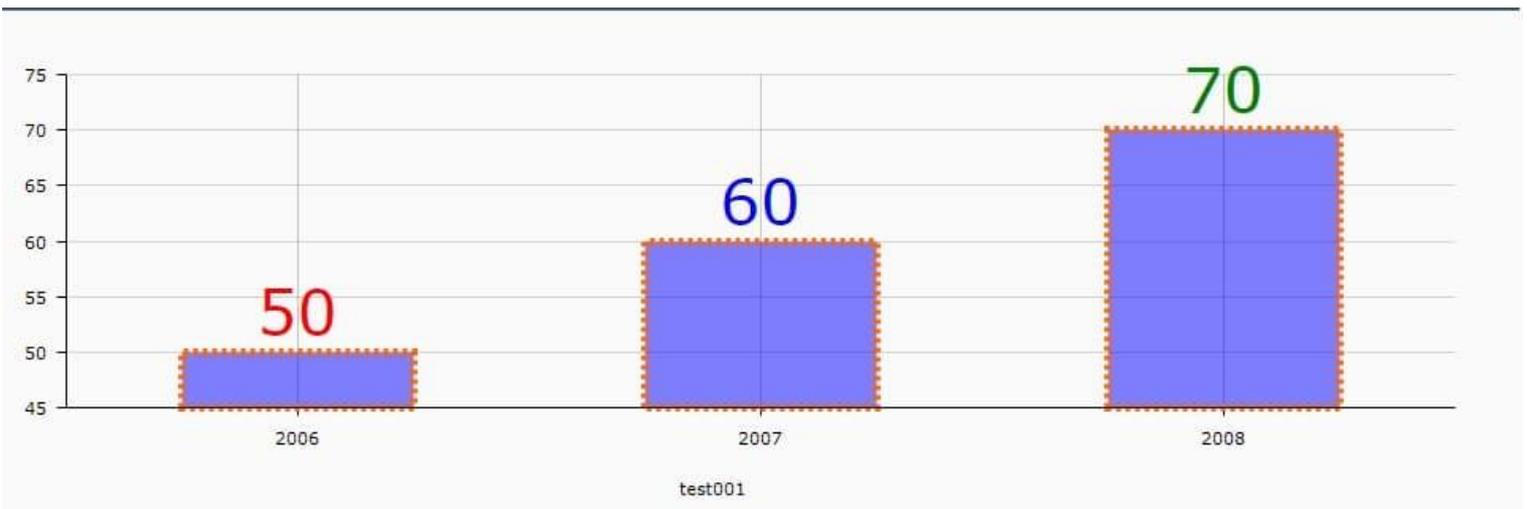
```
IT_FCAT = LT_FCAT " 차트 필드 정보
```

```
IT_DATA = LT_DATA " 차트 데이터
```

```
* IT_AXIS = IT_AXIS
```

```
* IS_CATEGORY = IS_CATEGORY " [U4A] AmChart.Category 구조
```

```
ENDMETHOD.
```



ValueField 값을 'A01'이라는 Field명으로 설정한 후 값을 구성하고 LabelColor 값을 'A02'라는 Field명으로 설정한 후 값을 구성.

위 Image에서 각 연도에 해당하는 Value값과 Label Color가 각각 설정된 것을 확인.

9.1.3 Axis

■ 목적 : Graph Axis의 속성을 정의 .



[첨부 ABAP SOURCE (SET_CHART_AXIS)]

```
method SET_CHART_AXIS.
```

```
DATA : LS_AXIS TYPE /U4A/CL_UO99988=>TY_AXIS.
```

```
*-----*
```

```
REFRESH ET_AXIS.
```

```
LS_AXIS-VALUEFIELD = 'WEK01'. "fcat 필드명.
LS_AXIS-STACKTYPE  = 'none'. "STACK유형("none", "regular", "100%", "3d").
LS_AXIS-POSITION   = 'left'. "위치("top", "bottom", "left", "right")
```

```

LS_AXIS-AXISALPHA      = '' .           "축 선의 투명도
LS_AXIS-AXISCOLOR     = '' .           "축 선 색상
LS_AXIS-AXISTHICKNESS = '' .           "축 선 두께
LS_AXIS-FONTSIZE      = '11' .        "TEXT FONT크기
APPEND LS_AXIS TO ET_AXIS .
CLEAR LS_AXIS .

endmethod.
    
```

PROPERTIES	MEAN
VALUEFIELD	Fcat Field 명
STACKTYPE	Stack 유형("none", "regular", "100%", "3d")
TITLE	축의 Title
TITLEALIGN	Title 위치("top", "center", "bottom", "left", "right")
TITLEBOLD	Title Bold 적용 여부('X' 적용, " 미적용)
TITLEROTATION	Title 글자 기울기
POSITION	위치("top", "bottom", "left", "right")
AXISALPHA	축 선의 투명도
AXISCOLOR	축 선 색상
AXISTHICKNESS	축 선 두께
COLOR	값 Text의 색상
FONTSIZE	Text Font 크기
DASHLENGTH	Graph Border 가로선 점선(숫자가 클수록 점선 간격이 길어짐, 왼쪽 축 기준만 적용됨)
GRIDALPHA	가로선의 투명도
GRIDCOLOR	가로선 색상
GRIDTHICKNESS	가로선 두께

LABELROTATION	값 기울기(Y축 영역에서는 속성 무시됨)
MAXIMUM	Graph 표현의 최대값
MINIMUM	Graph 표현의 최소값
INSIDE	Graph 숫자위치(X = 그래프 안으로 들어감)
UNIT	Y축 값 뒤에 표기 할 기호(% , \$ 등)
AUTOGRIDCOUNT	Y축 값의 눈금 자동 설정('X' = 자동설정 사용안함)
GRIDCOUNT	Y축 값의 눈금 갯수(autoGridCount = 'X' 일때만 적용)
GUIDE	Graph 평균선
LABELSDISABLED	Label 비활성화 여부('X' = 비활성화 처리)
BOLDLABELS	Label Bold 처리 여부('X' = BOLD 처리)
INCLUDEALLVALUES	확대 축소 시 Y축 최소, 최대값 고정 여부('X')
LABELOFFSET	Label 여백
OFFSET	Y축 여백
HIDEFIRSTLABEL	Y축 첫번째 값 표시 여부('X' = 표시안함)
HIDELASTLABEL	Y축 마지막 값 표시 여부('X' = 표시안함)
TITLEFONTSIZE	Title Font 크기
TOTALTEXTFIELD	Total 값 Binding Field(누적 막대 전용)
TOTALTEXTCOLOR	Total 값 색상 Field(누적 막대 전용)
TOTALTEXTOFFSET	Total 값 여백 Field(누적 막대 전용)
UNITPOSITION	Y축 값 뒤에 표기할 기호 위치(left, right)
TITLECOLOR	Title Font Color

1. AmSerialChartComposite Axis - StackType.

U4A IDE HELP

Chart



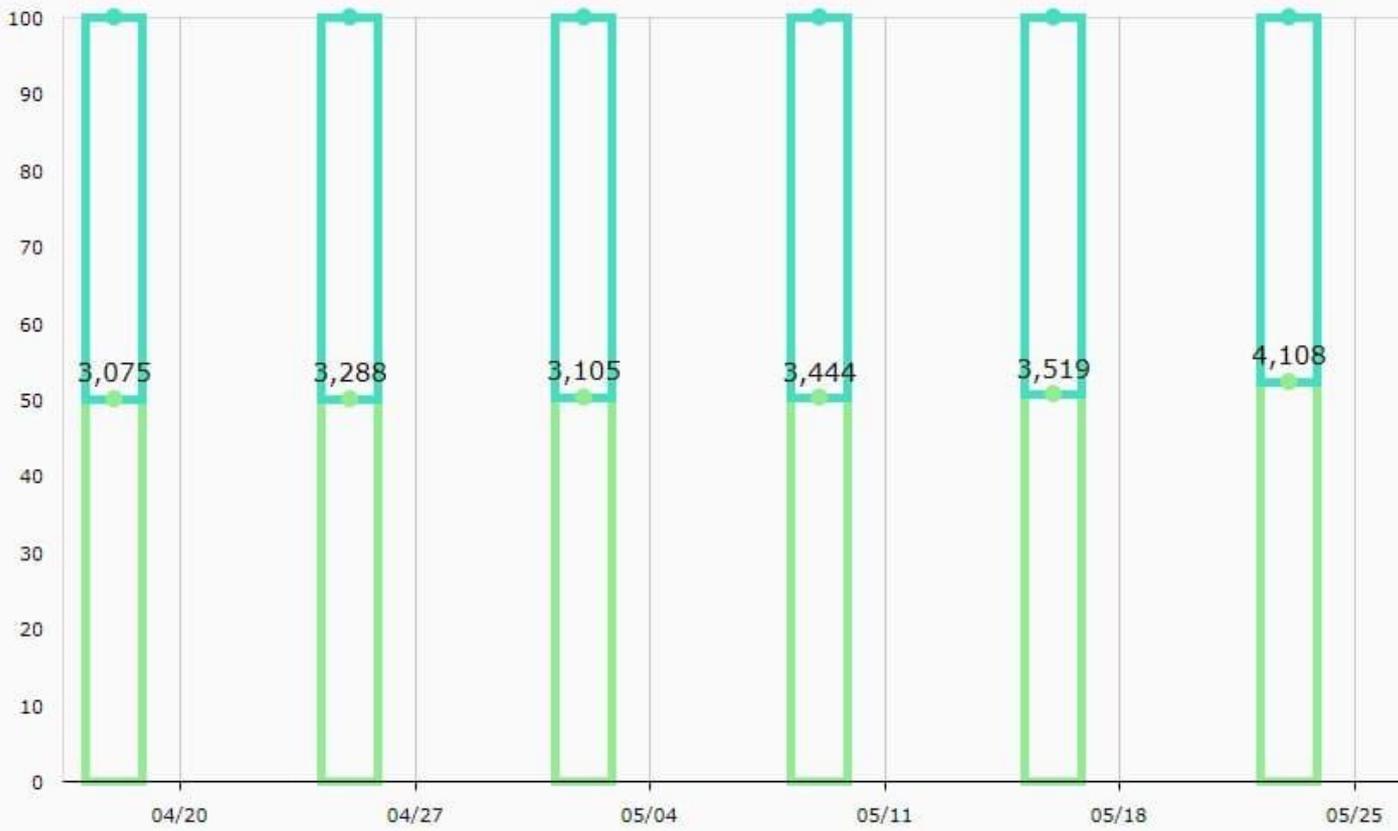
< none >

Chart

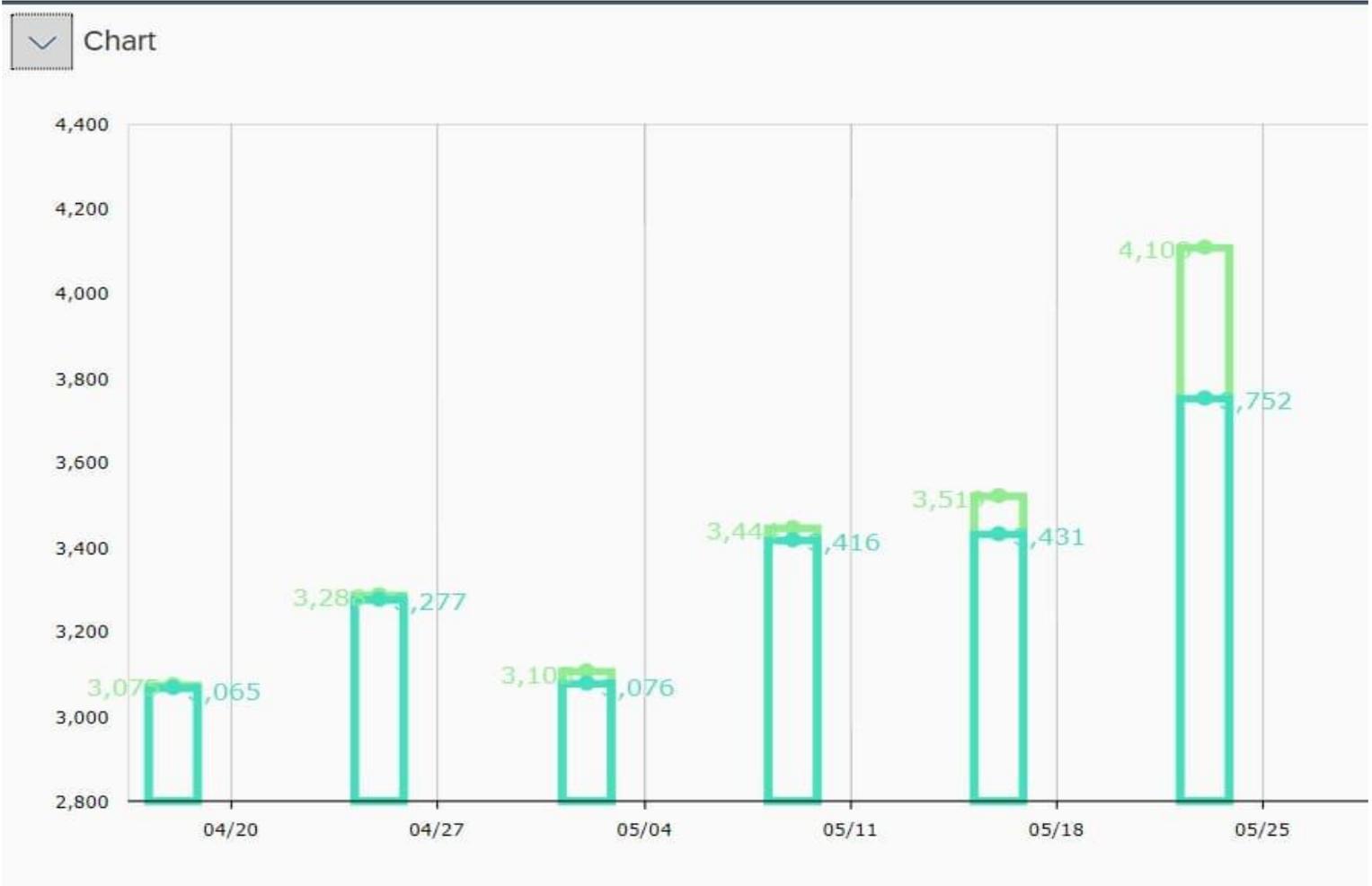


< regular >

Chart



< 100% >



< 3d >

2. AmSerialChartComposite Axis - Position.



축이 생성되는 위치를 설정 가능 .

Position을 'left', 'right'로 각각 생성하면 위 Image와 같이 양쪽에 축이 생성 되는 것을 확인.

3. AmSerialChartComposite Axis - Axis.



축 선 색상(AxisColor), 축 선 두께(AxisThickness), 축 선 투명도(AxisAlpha)를 설정 가능.

위 Image에서는 색상 : red 두께 : 5 투명도 : 1 로 설정하였으며 Axis가 존재하는 좌측에 붉은색 축이 생긴 것을 확인 .

4. AmSerialChartComposite Axis - X축.



가로선 색상(GridColor), 가로선 두께(GridThickness), 가로선 투명도(GridAlpha), 가로선 점선(DashLength)을 설정 가능.

위 Image에서는 색상 : blue 두께 : 1 투명도 : 1 로 설정하였으며 파란색 가로선이 생긴 것을 확인 .

5. AmSerialChartComposite Axis - Y축.



Y축 값 뒤에 표기 할 기호(Unit), Y축 값의 눈금 자동 설정(AutoGridCount), Y축 값의 눈금 갯수(GridCount),

Y축 여백(Offset), Y축 첫번째 값 표시여부(HideFirstLabel), Y축 마지막 값 표시 여부(HideLastLabel)를 설정 가능.

위 Image에서 좌측 Graph는 GridCount : 8, Unit : kg로 설정하였고 우측 Graph는 GridCount : 15, Unit: \$ 로 설정된 것을 확인 .

6. AmSerialChartComposite Axis - Title.



축의 Title 명(Title), Title 위치(TitleAlign), Title 굵기(TitleBold), Title 기울기(TitleRotation), Title Font 크기(TitleFontSize), Title Font 색상(TitleColor)를 설정 가능.

위 Image에서는 2개의 Title이 설정된 것을 확인 .

7. AmSerialChartComposite Axis - 표현 값.



Graph 표현의 최대값과 최소값을 설정 가능 .

위 Image에서는 좌측 Graph는 최소값 : 1000, 최대값 : 5000 으로 설정했으며 우측 Graph는 최소값 : 10000, 최대값 : 30000 으로 설정된 것을 확인 .

8. AmSerialChartComposite Axis - Total.

[첨부 ABAP SOURCE (SET_CHART_AXIS)]

```
METHOD SET_CHART_AXIS.
```

```
DATA : LS_AXIS TYPE /U4A/CL_UO99988=>TY_AXIS.
```



```
REFRESH ET_AXIS.
```

```
LS_AXIS-VALUEFIELD = 'WEK01'. "fcat 필드명.
LS_AXIS-STACKTYPE = 'regular'. "STACK유형("none", "regular", "100%", "3d").
LS_AXIS-POSITION = 'left'. "위치("top", "bottom", "left", "right")
LS_AXIS-FONTSIZE = '11'. "TEXT FONT크기
```

```

LS_AXIS-TOTALTEXTCOLOR = 'red'.
LS_AXIS-TOTALTEXTFIELD = 'A01'.
LS_AXIS-TOTALTEXTOFFSET = '10'.
APPEND LS_AXIS TO ET_AXIS.
CLEAR LS_AXIS.

LS_AXIS-VALUEFIELD = 'MON01'.
LS_AXIS-STACKTYPE = 'regular'.
LS_AXIS-POSITION = 'right'.
LS_AXIS-FONTSIZE = '11'.
LS_AXIS-TOTALTEXTCOLOR = 'blue'.
LS_AXIS-TOTALTEXTFIELD = 'A01'.
LS_AXIS-TOTALTEXTOFFSET = '50'.
APPEND LS_AXIS TO ET_AXIS.
CLEAR LS_AXIS.

ENDMETHOD.

```

"Total값 색상 필드(누적 막대 전용).

"Total값 Binding 필드(누적 막대 전용).

"Total값 여백 필드(누적 막대 전용).

"fcat 필드명.

"STACK유형("none", "regular", "100%", "3d").

"위치("top", "bottom", "left", "right")

"TEXT FONT크기

"Total값 색상 필드(누적 막대 전용).

"Total값 Binding 필드(누적 막대 전용).

"Total값 여백 필드(누적 막대 전용).

FieldCat 의 Binding Field를 구성할 때와 동일하게 'A01'이라는 Field명으로 설정한 후 값을 구성.

다음으로 Total과 관련된 TotalTextColor와 TotalTextOffset 값을 설정.

(Total Text에 모두 동일하게 적용되며 개별적으로 적용은 불가능 함.)

[첨부 ABAP SOURCE (SET_CHART)]

```

METHOD SET_CHART.

DATA : LO_CHART TYPE REF TO /U4A/CL_UO99988,
       LT_FCATT TYPE /U4A/CL_UO99988=>TT_FCATT,
       LT_DATA  TYPE /U4A/Y0071,
       LT_AXIS  TYPE /U4A/CL_UO99988=>TT_AXIS,
       LS_GRAPH TYPE /U4A/S0072.

DATA: LV_TOTXT TYPE STRING,
      LV_INDEX TYPE I.

```

U4A IDE HELP

FIELD-SYMBOLS: <FS_DATA>TYPE /U4A/S0071.

```
LO_CHART ?= ME->/U4A/IF_SERVER~AR_VIEW->GET_UI_INSTANCE( I_ID = 'CHART1' ).
```

```
CHECK LO_CHART IS BOUND.
```

```
"SET CHART FCAT DATA.
```

```
CALL METHOD ME->SET_CHART_FCAT
IMPORTING
  ET_FCAT = LT_FCAT.
```

```
"SET CHART AXIS DATA.
```

```
CALL METHOD ME->SET_CHART_AXIS
IMPORTING
  ET_AXIS = LT_AXIS.
```

```
"SET CHART DATA.
```

```
CALL METHOD ME->SET_CHART_DATA
IMPORTING
  ET_DATA = LT_DATA.
```

```
" TOTAL TEXT 설정 .
```

```
LOOP AT LT_DATA ASSIGNING <FS_DATA>.
```

```
  ADD 1 TO LV_INDEX.
```

```
" TOTAL TEXT 생성 .
```

```
LV_TOTXT = 'Total Text' && LV_INDEX.
```

```
LS_GRAPH-VALUEFIELD = 'A01'.
```

```
LS_GRAPH-GRAPHVALUE = LV_TOTXT.
```

```
APPEND LS_GRAPH TO <FS_DATA>-GRAPHVALUE.
```

```
CLEAR LS_GRAPH.
```

```
ENDLOOP.
```

```
"SHOW CHART.
```

```
CALL METHOD LO_CHART->SHOW
EXPORTING
```

```
  IT_FCAT = LT_FCAT
```

```
  IT_DATA = LT_DATA
```

```
  IT_AXIS = LT_AXIS.
```

```
" 차트 필드 정보
```

```
" 차트 데이터
```

```
ENDMETHOD.
```

Total Text Binding Field 값 설정.



Graph의 Total 과 관련된 합계 값 Text(ToTalTextField), 합계 값 색상(TotalTextColor), 합계 값 여백 (TotalTextOffset) 에 대한 설정이 된 것을 확인.

9.1.4 Guide

■ 목적 : Graph Guide의 속성을 정의 .

[첨부 ABAP SOURCE (SET_CHART_FCAT)]

METHOD SET_CHART_AXIS.

DATA : LS_AXIS TYPE /U4A/CL_U099988=>TY_AXIS.

DATA : LT_GUIDE TYPE /U4A/Y0074,
LS_GUIDE TYPE /U4A/S0074.

LS_GUIDE-GUIDE_VALUE	= '3700'.	" 평균선 값
LS_GUIDE-LINEALPHA	= '1'.	" 평균선 라인 투명도(0~1 소숫점 허용가능)
LS_GUIDE-LINECOLOR	= 'red'.	" 평균선 색상
LS_GUIDE-LINETHICKNESS	= '5'.	" 평균선 두께
LS_GUIDE-DASHLENGTH	= '60'.	" 평균선 점선 간격
LS_GUIDE-GUIDE_LABEL	= 'Guide01'.	" 평균선 Label
LS_GUIDE-FONTSIZE	= '20'.	" 평균선 Label font 크기
LS_GUIDE-INSIDE	= 'X'.	" 평균선 Label 위치('X' = 그래프 Y축 내부에 라벨 이 위치함, ' ' = 그래프 Y축 밖에 라벨이 위치함)
LS_GUIDE-GUIDE_POSITION	= 'left'.	" 평균선 Label 위치
("left", "right", "top", "bottom")		
LS_GUIDE-TICKLENGTH	= '5'.	" 평균선 내부 색상 투명도(0~1 소숫점 허용가능)
LS_GUIDE-FILLALPHA	= '0'.	" 평균선 내부 색상
LS_GUIDE-FILLCOLOR	= ''.	" 평균선 말풍선 Text
LS_GUIDE-BALLOONTEXT	= 'zztest1'.	" 평균선 표현 우선도('X' = Graph 앞에 표현 됨, ' ' = Graph 뒤에 표현됨)
LS_GUIDE-ABOVE	= 'X'.	" 평균선 표현 우선도('X' = Graph 앞에 표현 됨, ' ' = Graph 뒤에 표현됨)

LS_GUIDE-LABELROTATION = '45'.

리 안됨)

" 라벨 bold 여부('X' = bold 처리, ' ' = bold 처

APPEND LS_GUIDE TO LS_AXIS-GUIDE.

CLEAR ls_guide.

" AXIS를 구성하는 STRUCTURE의 GUIDE 필드에 삽입 .

ENDMETHOD.

PROPERTIES	MEAN
GUIDE_VALUE	평균선 값
LINEALPHA	평균선 Line 투명도(0~1 소숫점 허용가능)
LINECOLOR	평균선 색상
LINETHICKNESS	평균선 두께
DASHLENGTH	평균선 점선 간격
GUIDE_LABEL	평균선 Label
FONTSIZE	평균선 Label font 크기
INSIDE	평균선 Label 위치('X' = 그래프 Y축 내부에 라벨이 위치함, '' = 그래프 Y축 밖에 라벨이 위치함)
GUIDE_POSITION	평균선 Label 위치("left", "right", "top", "bottom")
FILLALPHA	평균선 내부 색상 투명도(0~1 소숫점 허용가능)
FILLCOLOR	평균선 내부 색상
BALLOONTEXT	평균선 말풍선 Text
ABOVE	평균선 표현 우선도('X' = Graph 앞에 표현됨, '' = Graph 뒤에 표현됨)
LABELROTATION	평균선 Label 기울기
BOLDLABEL	Label bold 여부('X' = bold 처리, '' = bold 처리 안됨)

1. AmSerialChartComposite Guide - Line.



평균선에 대한 값(Guide_Value), 평균선 Line 투명도(LineAlpha), 평균선 색상(LineColor), 평균선 두께 (LineThickness), 평균선 점선 간격(DashLength) 설정 가능 .

위 Image에서는 값 : 3700, 색상 : red, 두께 : 5, 투명도 : 1, 점선간격 : 60 으로 설정하였으며 설정한 값에 해당 하는 평균선이 생성 된 것을 확인 .

2. AmSerialChartComposite Guide - Above.





< ABOVE : 'X'
>
> < ABOVE : ''

평균선의 표현 우선도를 결정 .

왼쪽 Image의 경우 Above 값을 'X'로 설정하였으며 평균선이 Graph 보다 우선되어 표현되는 것을 확인 .

반대로 오른쪽 Image의 경우 Above 값을 ''로 설정하였으며 평균선이 Graph 뒤에 표현되는 것을 확인 .

3. AmSerialChartComposite Guide - Label.



평균선 Label 명(Guide_Value), 평균선 Label Font 크기(FontSize), 평균선 Label 위치(Guide_Position, Inside), 평균선 Label 기울기(LabelRotation)에 대해 설정 가능.

위 Image에서는 Font 크기 : 20, Label 위치 : left, Label 명 : Guide01 로 설정하였으며 평균선에 Label이 생성된 것을 확인 .

4. AmSerialChartComposite Guide - Fill.



평균선 내부 색상(FillAlpha), 평균선 내부 색상 투명도(FillColor) 에 대해 설정 가능.

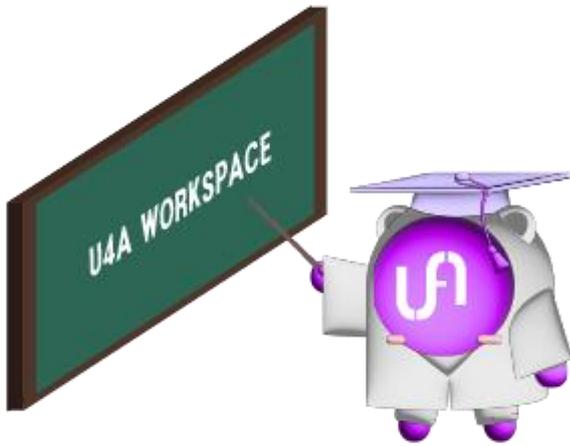
위 Image에서는 내부 색상 : blue 투명도 : 1로 설정하였으며 파란색 평균선이 생성된 것을 확인 .

5. AmSerialChartComposite Guide - BallonText.



평균선 위에 마우스 커서를 올렸을 때 'Balloon Text' 말풍선이 생성된 것을 확인 .

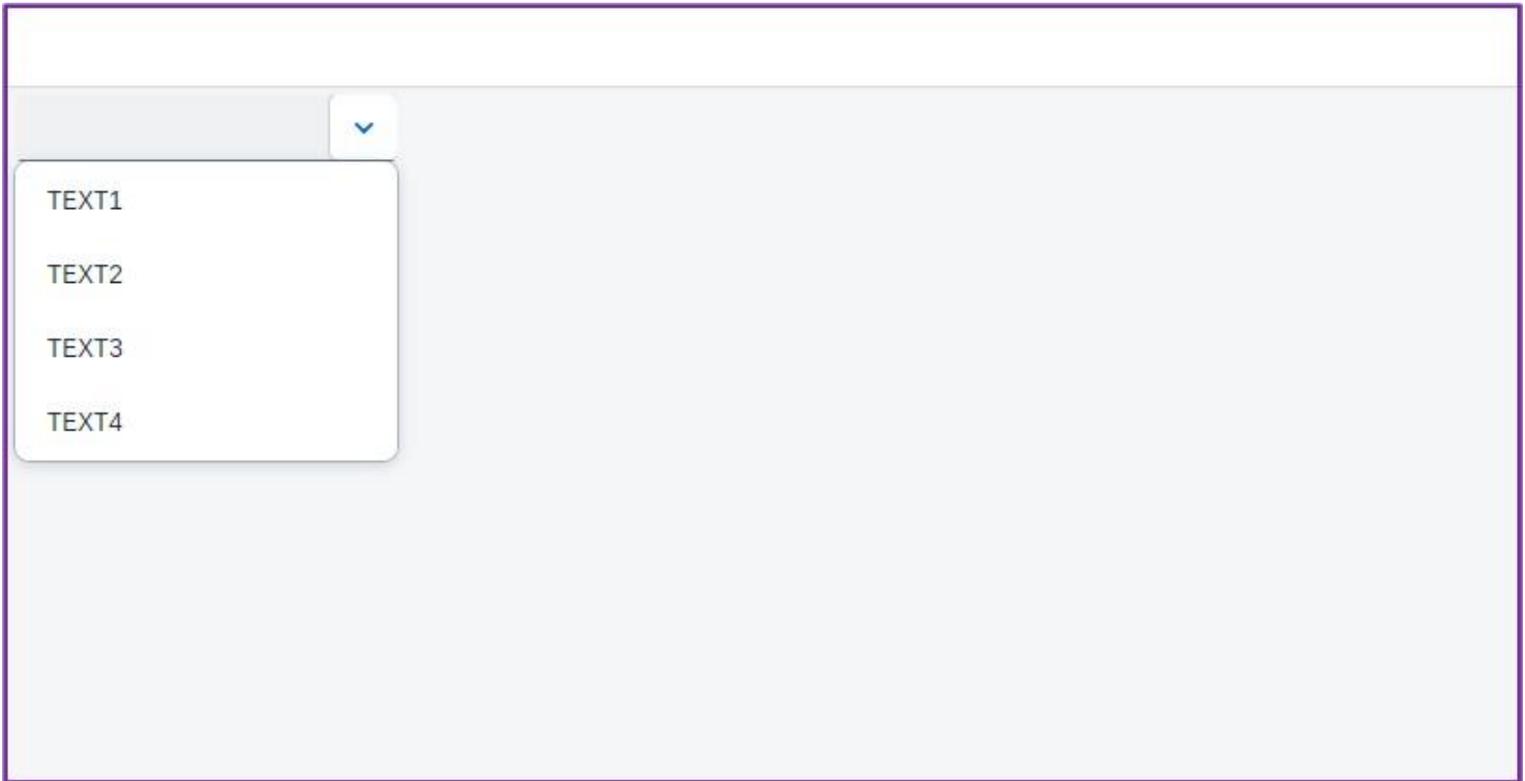
10. APP 개발 하기 Part 6



10.1 Dynamic ComboBox Create

■ 목적 : ComboBox UI 동적 생성 방법 소개 .

출력 시 화면 (예제 Sample)



1. 생성할 UI의 해당 Class를 찾아 선언 .

(참조 Class 조회는 App 개발하기 Part 6 참조)

```
DATA: LO_PAGE      TYPE REF TO /U4A/CL_UO00389,
      LO_COMBOBOX TYPE REF TO /U4A/CL_UO00256,
      LO_ITEM      TYPE REF TO /U4A/CL_UO00884.
```

2. ComboBox UI Object를 생성 .

*-- *ComboBox 오브젝트 생성* .

```
CREATE OBJECT LO_COMBOBOX
EXPORTING
```

```
ID = 'COMBOBOX1'.      " UI Instance Name
```

```
CHECK LO_COMBOBOX IS BOUND.
```

3. 생성한 Object가 포함될 영역의 Instance를 가져온다. (현재 Sample에서는 Page영역에 ComboBox UI를 추가).

```
*-- Page 오브젝트 생성 .
```

```
LO_PAGE ?= ME->AR_VIEW->GET_UI_INSTANCE( I_ID = 'PAGE' ).
```

```
CHECK LO_PAGE IS BOUND.
```

4. 상위 영역의 Aggregation에 생성한 ComboBox Object 추가 .

```
" ComboBox UI 삽입 .
```

```
CALL METHOD LO_PAGE->ADDCONTENT
```

```
EXPORTING
```

```
CONTENT = LO_COMBOBOX.      " content
```

5. ComboBox의 SelectedKey Property에 Binding 설정 .

```
" ComboBox SelectedKey Binding .
```

```
LO_COMBOBOX->BP00256_SELECTEDKEY = 'GS_DDLB-KEY'.
```

6. ComboBox의 Item 영역에 생성 될 UI Object 생성 및 Item Data 구성.

(현재 Sample에서는 Item UI를 ComboBox 하위에 추가)

```
*-- Item UI 삽입 .
```

```
CREATE OBJECT LO_ITEM
```

```
EXPORTING
```

```
ID = 'ITEM1'.      " UI Instance Name
```

```
CHECK LO_ITEM IS BOUND.
```

```
" DDLB List 구성 .
```

```
CALL METHOD SET_DDLB.
```

7. 추가 할 영역의 Aggregation 확인 후 생성한 Item Object를 붙여준다 .

```
" Item UI 삽입 .
```

```
CALL METHOD LO_COMBOBOX->ADDITEM
```

```
EXPORTING
```

```
ITEMS = LO_ITEM.      " items
```

7. Item의 key, Text Property 및 Model Binding (ComboBox의 Item Aggregation에 Binding한 Table 구조 Data를 Binding).

```
" Item Key Binding .
```

```
LO_ITEM->BP00884_KEY = 'GT_DDLB-KEY'.
```

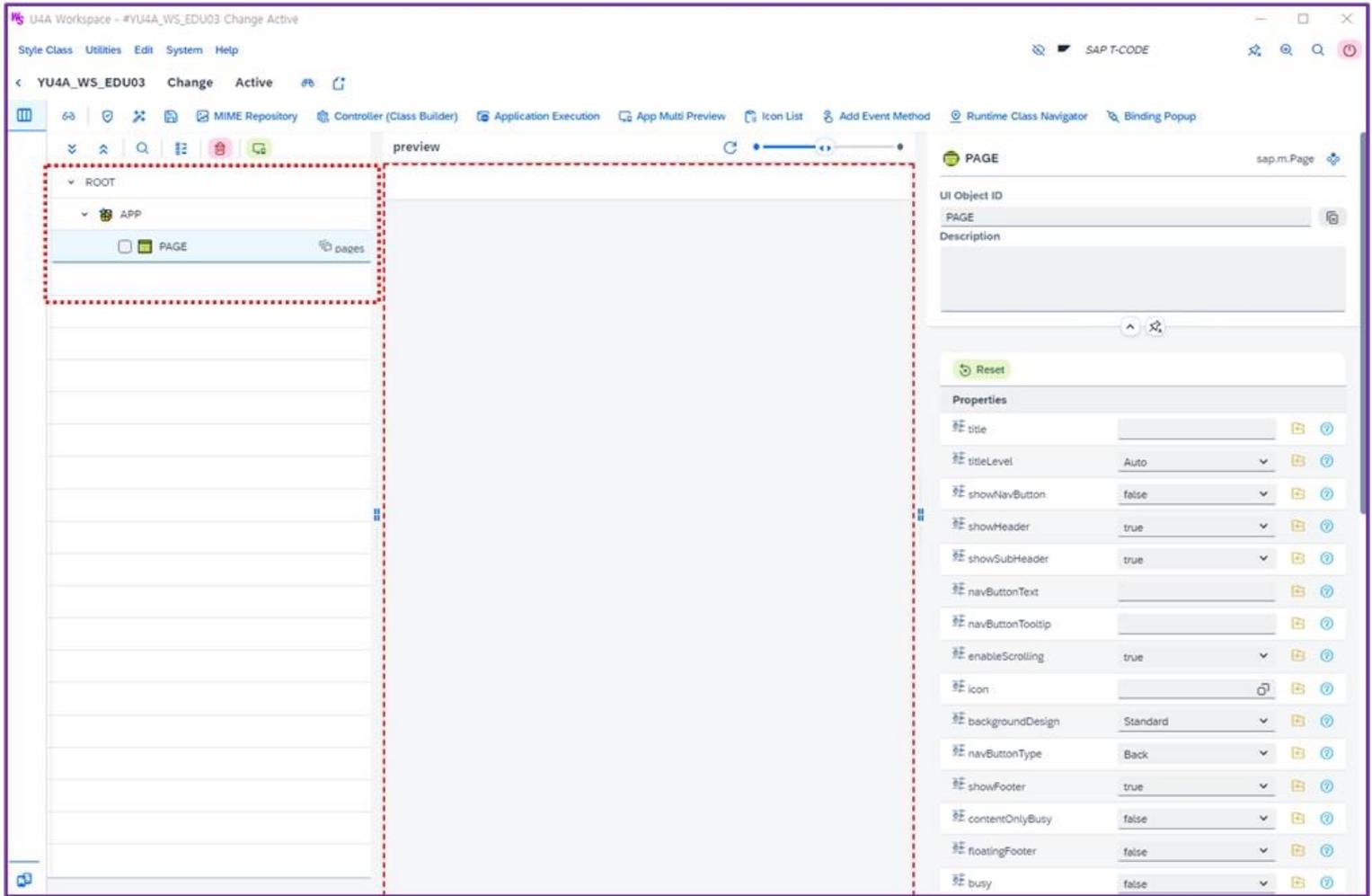
```
" Item Text Binding .
```

```
LO_ITEM->BP00884_TEXT = 'GT_DDLB-TEXT'.
```

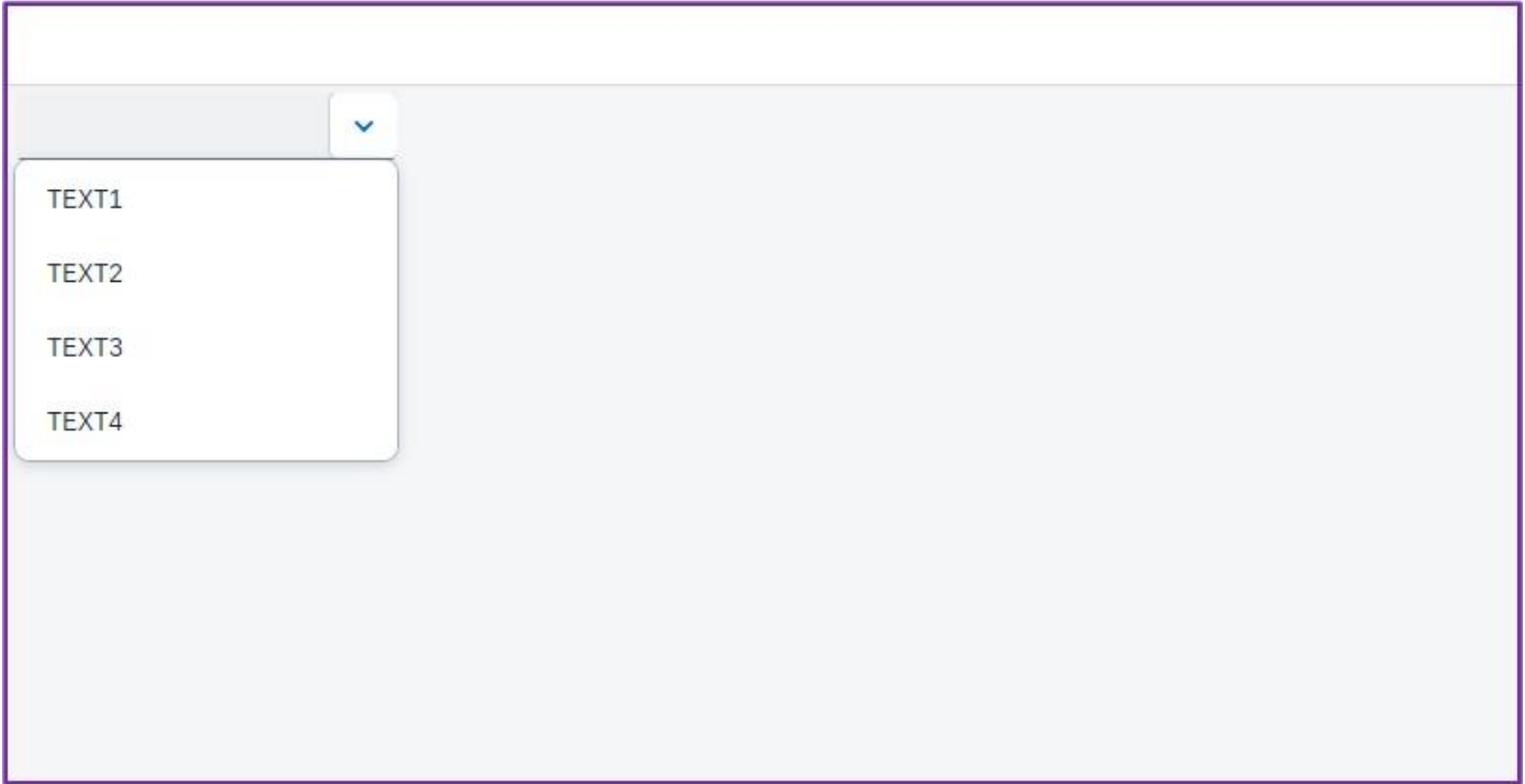
```
" Model Binding .
```

```
LO_ITEM->MODEL = 'GT_DDLB'.
```

8. UX Design Area 영역에 어떤 UI도 추가 되지 않은 것을 확인 .



9. 실행 했을 때 ComboBox UI가 생성된 것을 확인 .

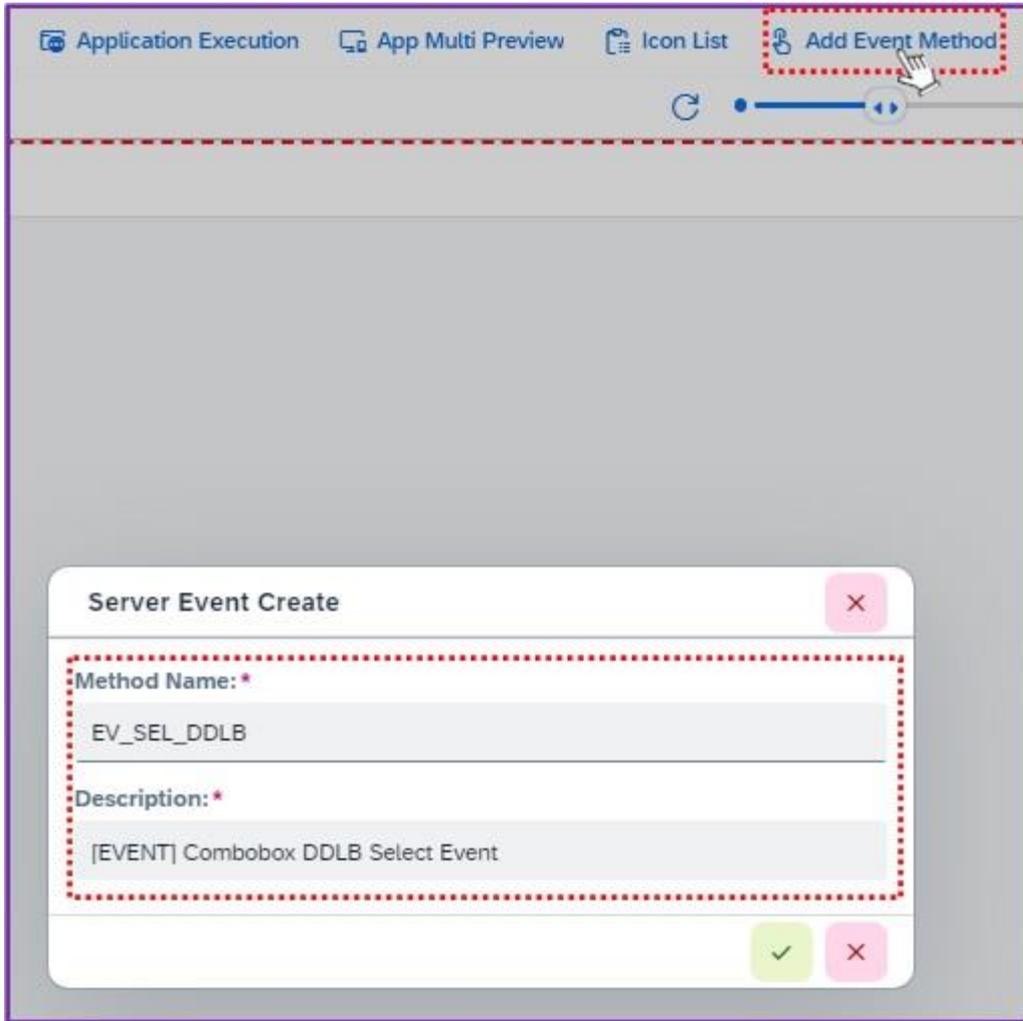


10. ComboBox Event Method 명 설정 (현재 Sample에서는 Change Event 설정).

"Change Event 설정 .

```
CALL METHOD LO_COMBOBOX->ATTACHCHANGE
EXPORTING
  SERVEREVENTNAME      = 'EV_SEL_DDLB' .  " serverEventName
```

11. ComboBox Change Event Method 생성 .



12. ComboBox Item 선택 시 호출 Method .

[첨부ABAP SOURCE (EV_SEL_DDLB)]

METHOD EV_SEL_DDLB .

DATA: LS_DDLB TYPE TY_DDLB .

DATA: LV_MSG TYPE STRING .

"선택한 Key에 해당하는 Item 조회."

```
READ TABLE GT_DDLB INTO LS_DDLB
  WITH KEY KEY = GS_DDLB-KEY.
```

```
CHECK SY-SUBRC EQ 0.
```

```
"출력 메시지 설정 .
```

```
CONCATENATE `KEY: ` LS_DDLB-KEY CL_ABAP_CHAR_UTILITIES=>NEWLINE `TEXT: ` LS_DDLB-
TEXT INTO LV_MSG.
```

```
" 메시지 출력 .
```

```
CALL METHOD /U4A/CL_UTILITIES=>M_MESSAGE_TOAST
```

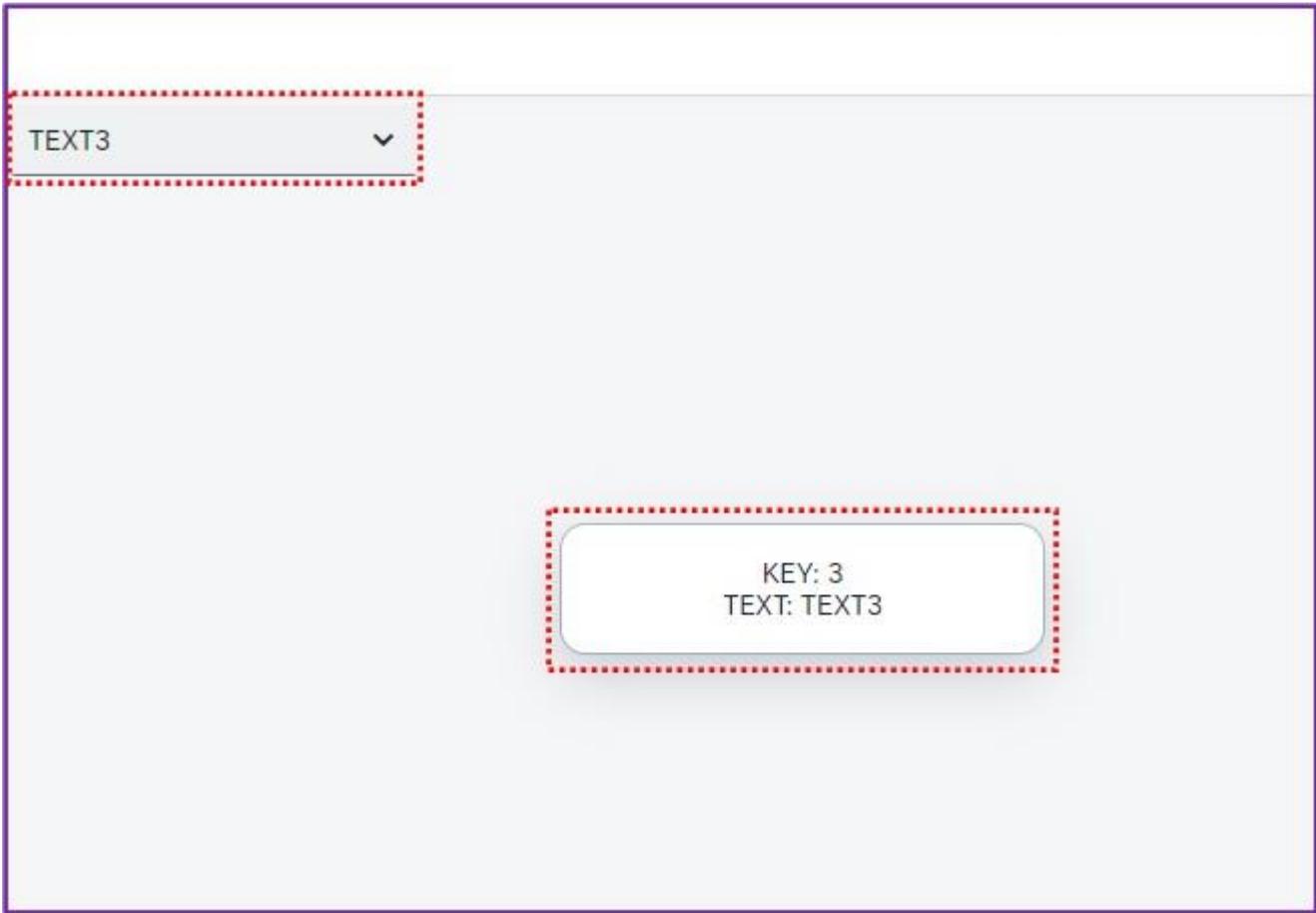
```
  EXPORTING
```

```
  IO_VIEW = ME->AR_VIEW " [U4A] UI Element Super Class
```

```
  I_MSGTX = LV_MSG. " MESSAGE TEXT
```

```
ENDMETHOD.
```

13. Item 항목을 선택하여 ComboBox Change Event가 정상적으로 동작하는 것을 확인 .



 UI Object 생성 시 부여하는 ID는 절대로 중복 되어서는 안된다.

10.2 Dynamic Column Create

■ 목적 : Table의 Column을 상황에 따라 동적으로 생성 하기 위한 방법.

1. Row에 Binding 하기 위한 Table Model 및 Column에 Binding 하기 위한 Table Model 선언 .
(Global로 사용할 것이기 때문에 Public section에 선언).

```
public section.
```

```
interfaces ZIF_U4A_SERVER .
```

```
TYPES:
```

```
" Column Table .
```

```
BEGIN OF GTY_COLUMN ,
    NAME TYPE STRING ,
END OF GTY_COLUMN .
```

```
TYPES:
```

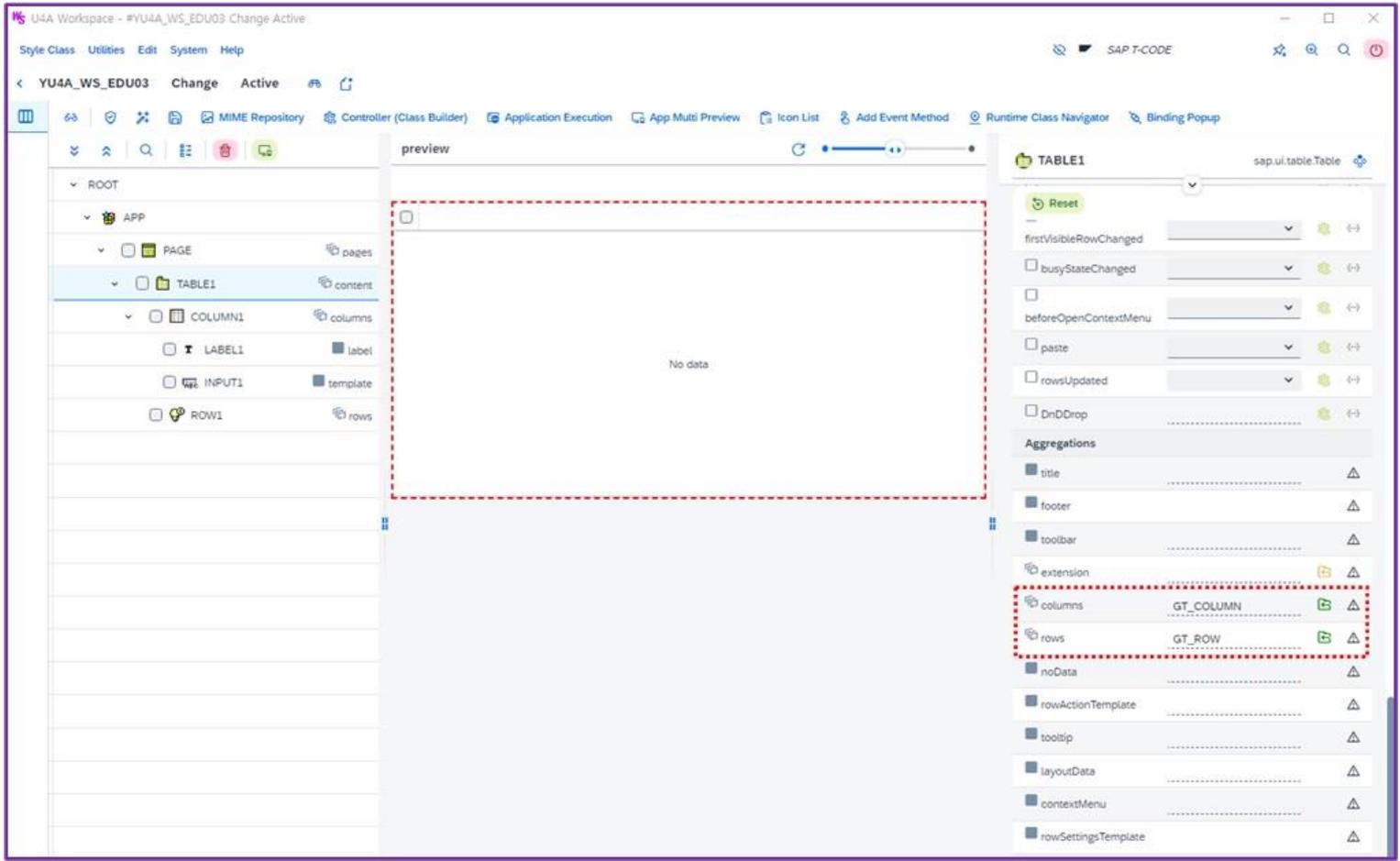
```
" Rows Table .
```

```
BEGIN OF GTY_ROW ,
    F01 TYPE STRING ,
    F02 TYPE STRING ,
    F03 TYPE STRING ,
    F04 TYPE STRING ,
    F05 TYPE STRING ,
```

```
END OF GTY_ROW .
```

```
DATA: GT_COLUMN TYPE TABLE OF GTY_COLUMN ,
      GT_ROW     TYPE TABLE OF GTY_ROW .
```

2. Row에 Binding 하기 위한 Table Model 및 Column에 Binding 하기 위한 Table Model 선언 .



3. Column, Rows Table Data 구성 .

(본 단원에서는 실행 시 Column이 구성 되도록 On_init영역에 해당 Method 설정).

[첨부ABAP SOURCE (SET_COL_ROW)]

```
DATA: LS_COLUMN TYPE GTY_COLUMN.
```

```
DATA: LS_ROW TYPE GTY_ROW.
```

```
DATA: LV_INDEX TYPE N LENGTH 2,  
      LV_ROW TYPE STRING,  
      LV_COLUMN TYPE STRING.
```

```
FIELD-SYMBOLS: <FS_ROW> TYPE STRING.
```

" COLUMN .

DO 5 TIMES.

" 현재 테이블 인덱스 .

LV_INDEX = SY-INDEX.

" COLUMN 명 생성 .

CONCATENATE 'F' LV_INDEX INTO LV_COLUMN.

" COLUMN 명 설정 .

LS_COLUMN-NAME = LV_COLUMN.

" COLUMN 데이터 쌓기 .

APPEND LS_COLUMN TO GT_COLUMN.

CLEAR LS_COLUMN.

ENDDO.

LOOP AT GT_COLUMN INTO LS_COLUMN.

" ROW 생성 .

CONCATENATE 'LS_ROW-' LS_COLUMN-NAME INTO LV_ROW.

ASSIGN (LV_ROW) TO <FS_ROW>.

CHECK <FS_ROW> IS ASSIGNED.

" 필드값 설정 .

<FS_ROW> = 'X'.

APPEND LS_ROW TO GT_ROW.

CLEAR LS_ROW.

ENDLOOP.

ENDMETHOD.

Column Table에는 Column명 생성 시 Rows Table의 Field명과 동일하게 설정. (본 예제에서는 F01,

F02)

Rows Table에는 Field값을 설정해주어야 함 . (본 예제에서는 'X' 값 설정)

4. 실행하여 Column이 정상적으로 생성되는지 여부 및 필드 값 정상적으로 입력됐는지 확인 .

<input type="checkbox"/>	F01	F02	F03	F04	F05
<input type="checkbox"/>	X				
<input type="checkbox"/>		X			
<input type="checkbox"/>			X		
<input type="checkbox"/>				X	
<input type="checkbox"/>					X

10.3 Dynamic Table Create

■ 목적 : Table 동적 생성 방법.

출력 시 화면 (예제 Sample)

	Col1	Col2	Col3	Col4	Col5	Col6	Col7	Col8	Col9	Col10
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="radio"/>	Text1	<input type="text" value="Input1"/>	Label & Tooltip1	<input type="text" value=""/>	<input checked="" type="checkbox"/>	Objectstatus1	<input type="text" value="2021-05-04"/>	<input type="text" value="Textarea1"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="radio"/>	Text2	<input type="text" value="Input2"/>	Label & Tooltip2	<input type="text" value=""/>	<input checked="" type="checkbox"/>	Objectstatus2	<input type="text" value="2021-05-04"/>	<input type="text" value="Textarea2"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="radio"/>	Text3	<input type="text" value="Input3"/>	Label & Tooltip3	<input type="text" value=""/>	<input checked="" type="checkbox"/>	Objectstatus3	<input type="text" value="2021-05-04"/>	<input type="text" value="Textarea3"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="radio"/>	Text4	<input type="text" value="Input4"/>	Label & Tooltip4	<input type="text" value=""/>	<input checked="" type="checkbox"/>	Objectstatus4	<input type="text" value="2021-05-04"/>	<input type="text" value="Textarea4"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="radio"/>	Text5	<input type="text" value="Input5"/>	Label & Tooltip5	<input type="text" value=""/>	<input checked="" type="checkbox"/>	Objectstatus5	<input type="text" value="2021-05-04"/>	<input type="text" value="Textarea5"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="radio"/>	Text6	<input type="text" value="Input6"/>	Label & Tooltip6	<input type="text" value=""/>	<input checked="" type="checkbox"/>	Objectstatus6	<input type="text" value="2021-05-04"/>	<input type="text" value="Textarea6"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="radio"/>	Text7	<input type="text" value="Input7"/>	Label & Tooltip7	<input type="text" value=""/>	<input checked="" type="checkbox"/>	Objectstatus7	<input type="text" value="2021-05-04"/>	<input type="text" value="Textarea7"/>
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="radio"/>	Text8	<input type="text" value="Input8"/>	Label & Tooltip8	<input type="text" value=""/>	<input checked="" type="checkbox"/>	Objectstatus8	<input type="text" value="2021-05-04"/>	<input type="text" value="Textarea8"/>

1. 출력할 Internal Table을 생성한다 (Global로 사용할 것이기 때문에 Public section에 선언).

public section.

```
interfaces ZIF_U4A_SERVER .
```

```
TYPES: BEGIN OF GTY_ROWS,
    COL1(1) TYPE C,           "Checkbox
    COL2(1) TYPE C,           "Radiobutton
    COL3     TYPE STRING,     "Text
    COL4     TYPE STRING,     "Input
    COL5     TYPE STRING,     "Label
    COL6     TYPE STRING,     "Combobox
    COL7     TYPE STRING,     "Icon
    COL8     TYPE STRING,     "Objectstatus
    COL9     TYPE D,          "Datepicker
    COL10    TYPE STRING,     "Textarea
END OF GTY_ROWS.
```

```
data:
```

```
GT_ROWS TYPE TABLE OF GTY_ROWS .
```

"Internal Table에 포함될 Combobox도 생성

```
TYPES: BEGIN OF GTY_DDL,
        KEY   TYPE STRING,
        VALUE TYPE STRING,
END OF GTY_DDL.
```

```
DATA: GT_DDL TYPE TABLE OF GTY_DDL.
```

2. 생성 할 해당 UI의 Class를 찾아 선언 (App개발하기 Part6 참조).

```
DATA: LO_PAGE TYPE REF TO /U4A/CL_U000389,
      LO_TAB  TYPE REF TO /U4A/CL_U001139,
      LO_COL  TYPE REF TO /U4A/CL_U001127,
      LO_LABEL TYPE REF TO /U4A/CL_U000319,
      LO_ROW  TYPE REF TO /U4A/CL_U001131.
```

*-- *TABLE FIELD OBJECT*

```
DATA: LO_CHKBOX TYPE REF TO /U4A/CL_U000253,
      LO_RADIO  TYPE REF TO /U4A/CL_U000410,
      LO_TEXT   TYPE REF TO /U4A/CL_U000452,
      LO_INPUT  TYPE REF TO /U4A/CL_U000314,
      LO_LABEL2 TYPE REF TO /U4A/CL_U000319,
      LO_COMBO  TYPE REF TO /U4A/CL_U000256,
      LO_LISTITEM TYPE REF TO /U4A/CL_U000887,
      LO_ICON   TYPE REF TO /U4A/CL_U000879,
      LO_OBJECTSTATUS TYPE REF TO /U4A/CL_U000368,
      LO_DATEPICKER TYPE REF TO /U4A/CL_U000261,
      LO_TEXTAREA TYPE REF TO /U4A/CL_U000453.
```

3. 생성한 Object가 포함될 영역의 Instance를 가져온다.

```
LO_PAGE ?= ME->AR_VIEW->GET_UI_INSTANCE( I_ID = 'PAGE' ).
CHECK LO_PAGE IS BOUND.
```

4. Table Object 생성.

```
CREATE OBJECT LO_TAB
EXPORTING
  ID = 'TABLE1'. " UI Instance Name

CHECK LO_TAB IS BOUND.
```

5. 상위 영역의 Aggregation에 생성한 Table을 추가 (Method 명 = ADD + 추가 할 Aggregation 영역).

(본 단원에서는 Page의 Content에 Table UI를 추가함 .)

```
LO_PAGE->ADDCONTENT (
  EXPORTING
    CONTENT = LO_TAB ). " content
```

6. 생성한 Table의 Property를 지정하는 경우 (set+ properties명 , 본 단원에서는 visiblerowcount를 적용)

```
CALL METHOD LO_TAB->SETVISIBLEROWCOUNT
  EXPORTING
    VISIBLEROWCOUNT = 8. " visibleRowCount
```

7. Column Object를 생성한 뒤 Table에 Column Object 추가.

```
CREATE OBJECT LO_COL
  EXPORTING
    ID = 'COLUMN' && SY-INDEX. " UI Instance Name

CHECK LO_COL IS BOUND.

CALL METHOD LO_TAB->ADDCOLUMN
  EXPORTING
    COLUMNS = LO_COL. " columns
```

8. Label Object를 생성한 뒤 Column에 Label Object 추가 및 Label Text를 입력.

(SETLABEL Method를 사용하여 Label Text값 설정).

```
CREATE OBJECT LO_COL
  EXPORTING
    ID = 'COLUMN' && SY-INDEX. " UI Instance Name

CHECK LO_LABEL IS BOUND.

CALL METHOD LO_COL->SETLABEL
  EXPORTING
```

```

LABEL = LO_LABEL.      " label

CALL METHOD LO_LABEL->SETTEXT
EXPORTING
    TEXT = 'Col' && SY-INDEX.      " text

```

9. 테이블에 추가할 Object 생성 (Text , Checkbox , Radiobutton , Input 등등).

하위 Logic은 7~8번에서 쓰인 내용이 포함되어있음.

```
DO 10 TIMES.
```

"Column 생성 (UI ID가 중복되면 안되기 때문에 Index로 아이디를 분기함)

```

CREATE OBJECT LO_COL
EXPORTING
    ID = 'COLUMN' && SY-INDEX.      " UI Instance Name

CHECK LO_COL IS BOUND.

CALL METHOD LO_TAB->ADDCOLUMN
EXPORTING
    COLUMNS = LO_COL.      " columns

CREATE OBJECT LO_LABEL
EXPORTING
    ID = 'LABEL' && SY-INDEX.      " UI Instance Name

CALL METHOD LO_COL->SETLABEL
EXPORTING
    LABEL = LO_LABEL.      " label

CALL METHOD LO_LABEL->SETTEXT
EXPORTING
    TEXT = 'Col' && SY-INDEX.      " text

```

"---필드 생성---

```
CASE SY-INDEX.
```

```
WHEN 1. "CHECKBOX
```

"-- 해당 Column의 Properties 지정

```

CALL METHOD LO_COL->SETHALIGN
EXPORTING

```

```
HALIGN = 'Center'. " hAlign
```

```
CALL METHOD LO_COL->SETWIDTH
EXPORTING
```

```
WIDTH = '80px'. "width
```

```
CREATE OBJECT LO_CHKBOX
EXPORTING
```

```
ID = 'CHECKBOX'. " UI Instance Name
```

```
CHECK LO_CHKBOX IS BOUND.
```

"해당 Field Binding

```
LO_CHKBOX->BP00253_SELECTED = 'GT_ROWS-COL1'.
```

"해당 Field Column에 추가

```
CALL METHOD LO_COL->SETTEMPLATE
EXPORTING
```

```
TEMPLATE = LO_CHKBOX. " template
```

WHEN 2. "RADIOBUTTON

"-- 해당 Column의 Properties 지정

```
CALL METHOD LO_COL->SETWIDTH
EXPORTING
```

```
WIDTH = '60px'. " width
```

```
CREATE OBJECT LO_RADIO
EXPORTING
```

```
ID = 'RADIOBUTTON'. " UI Instance Name
```

```
CHECK LO_RADIO IS BOUND.
```

"해당 Field Binding

```
LO_RADIO->BP00410_SELECTED = 'GT_ROWS-COL2'.
```

"해당 Field Column에 추가

```
CALL METHOD LO_COL->SETTEMPLATE
EXPORTING
```

```
TEMPLATE = LO_RADIO. " template
```

(위와같은 예시로 추가하고 싶은 object도 추가하면 됨.)

```
ENDCASE .
```

```
ENDDO .
```

10. 컬럼 생성 및 필드 생성을 마쳤다면 Row 생성 후 LO_ROW->MODEL Method 사용하여 Table Rows에 Binding .

LO_ROW->ADDROW Method를 사용해 Rows UI 생성 .

```
CREATE OBJECT LO_ROW
```

```
EXPORTING
```

```
  ID = 'ROW1' .      " UI Instance Name
```

```
CHECK LO_ROW IS BOUND .
```

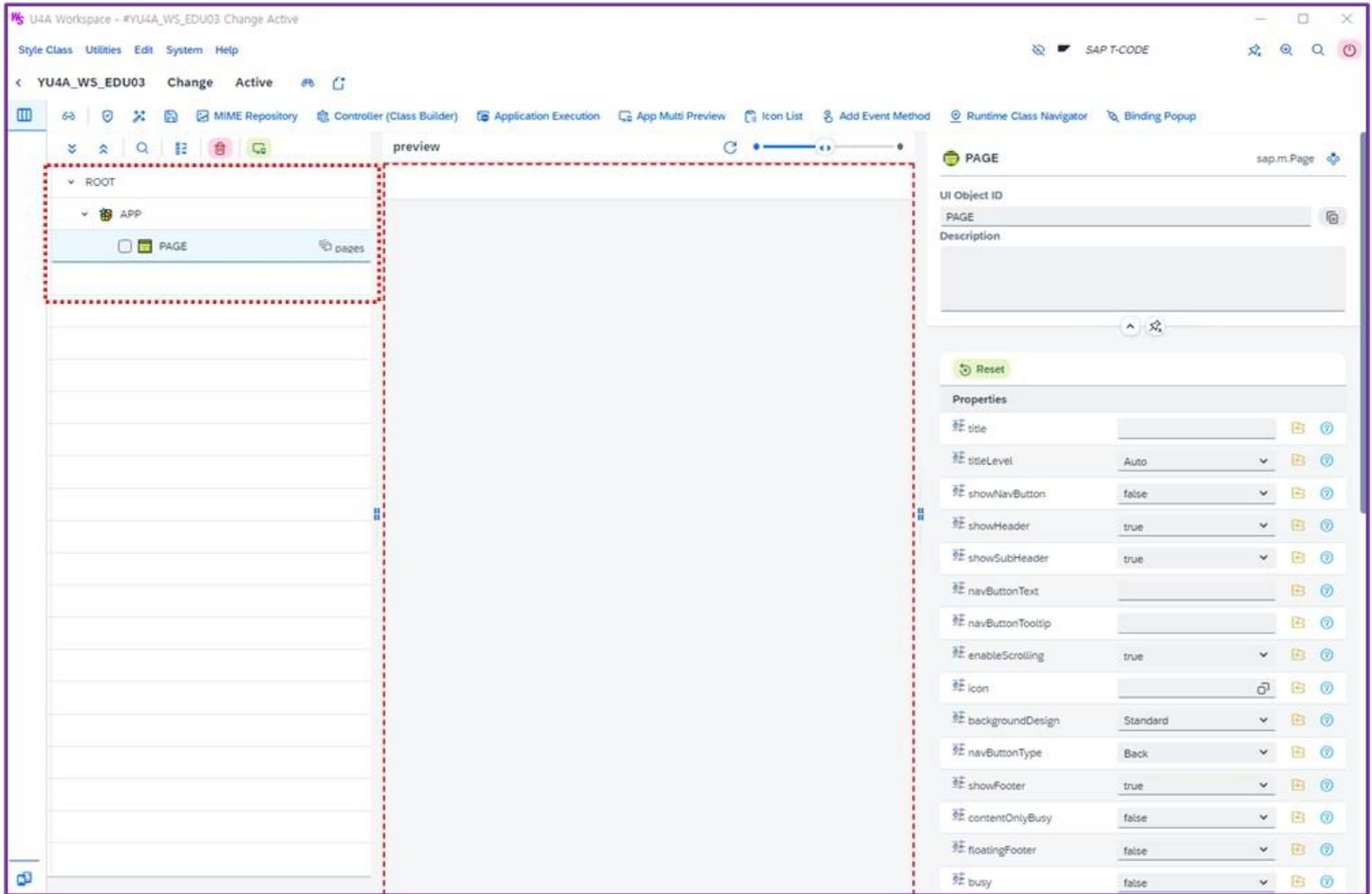
```
LO_ROW->MODEL = 'GT_ROWS' .      " GT_ROWS Binding
```

```
CALL METHOD LO_TAB->ADDROW
```

```
EXPORTING
```

```
  ROWS = LO_ROW .      " rows UI 생성 .
```

11. UX Design Area 영역에 어떤 UI도 추가 되지 않은 것을 확인 .



12. 예제대로 진행할 경우 하위와 같이 생성된다.

Col1	Col2	Col3	Col4	Col5	Col6	Col7	Col8	Col9	Col10
<input checked="" type="checkbox"/>	<input type="radio"/>	Text1	Input1	Label & Tooltip1	<input type="text"/>	<input checked="" type="checkbox"/>	Objectstatus1	2021-05-04	Textarea1
<input checked="" type="checkbox"/>	<input type="radio"/>	Text2	Input2	Label & Tooltip2	<input type="text"/>	<input checked="" type="checkbox"/>	Objectstatus2	2021-05-04	Textarea2
<input checked="" type="checkbox"/>	<input type="radio"/>	Text3	Input3	Label & Tooltip3	<input type="text"/>	<input checked="" type="checkbox"/>	Objectstatus3	2021-05-04	Textarea3
<input checked="" type="checkbox"/>	<input type="radio"/>	Text4	Input4	Label & Tooltip4	<input type="text"/>	<input checked="" type="checkbox"/>	Objectstatus4	2021-05-04	Textarea4
<input checked="" type="checkbox"/>	<input type="radio"/>	Text5	Input5	Label & Tooltip5	<input type="text"/>	<input checked="" type="checkbox"/>	Objectstatus5	2021-05-04	Textarea5
<input checked="" type="checkbox"/>	<input type="radio"/>	Text6	Input6	Label & Tooltip6	<input type="text"/>	<input checked="" type="checkbox"/>	Objectstatus6	2021-05-04	Textarea6
<input checked="" type="checkbox"/>	<input type="radio"/>	Text7	Input7	Label & Tooltip7	<input type="text"/>	<input checked="" type="checkbox"/>	Objectstatus7	2021-05-04	Textarea7
<input checked="" type="checkbox"/>	<input checked="" type="radio"/>	Text8	Input8	Label & Tooltip8	<input type="text"/>	<input checked="" type="checkbox"/>	Objectstatus8	2021-05-04	Textarea8

1. Checkbox 2. Radiobutton 3. Text 4. Input 5. Label 6. Combobox 7. Icon 8. Objectstatus 9. Datepicker 10. Textarea