



관리자페이지 - 전체

마지막 수정일자 2023-09-26

목차

- 1. 위젯 관리
 - ▼ 1.1. 마스터
 - (1) 개요
 - (2) 위젯 안내
 - ▼ 1.2. 속성
 - (1) 개요
 - (2) 기본 동작
 - (3) 속성 작성
 - ▼ 1.3. 쿼리
 - (1) 개요
 - (2) 쿼리 작성
 - ▼ 1.4. 설명
 - (1) 개요
 - (2) 설명 작성
 - ▼ 1.5. 권한
 - (1) 개요
 - (2) 권한 관리
- 2. 보드 관리
 - ▼ 2.1. 마스터
 - (1) 개요
 - (2) Master Manager - 보드
 - ▼ 2.2. 인트로 보드
 - (1) 개요
 - (2) 인트로 설정
 - ▼ 2.3. 보드 속성
 - (1) 설정 방법
 - (2) 속성 종류
- 3. 페이지 관리
 - 3.1. 마스터
 - (1) 개요
 - (2) Master Manager - 메뉴
 - 3.2. 페이지 편집
 - (1) 개요
 - (2) 페이지 툴바

▼ 1. 위젯 관리

▼ 1.1. 마스터

▼ (1) 개요

ZZ.ZNBMWIDGET 테이블 혹은 위젯관리 페이지의 Widget Master 위젯 (WIDGETMAST)을 사용하여 위젯 마스터 관리를 할 수 있다.

Widget Master 위젯의 총 8가지의 탭을 이용하여 쿼리 작성 및 실시간 반영, 위젯 단위로 사용자 권한 관리 등 위젯을 관리 편의 기능을 제공한다.

The screenshot shows a software application window titled "Widget Master". At the top, there's a toolbar with icons for search, refresh, and save. Below the toolbar is a header bar with the title "Widget Master" and the subtitle "WIDGETMAST ADMIN". The main area contains a form with various input fields. Key fields include "Name" (Widget Master), "사용종료일" (9999-12-31), "기본넓이" (40), "위젯리소스" (masterwidget.js), and "서브릿ID" (dev.WIDGETMAST). There are also dropdowns for "사용시작일" (2022-02-12), "기본높이" (6), "위젯유형" (ADMIN), "데이터출처" (REAL), and "속성ID" (empty). A note field "피라미터" is present at the bottom. On the right side of the form, there are several small icons.

Widget Master 위젯

▼ Master Table

- 주요 정보 ([ZZ.ZNBMWIDGET](#) 위젯 마스터 테이블 기준)

컬럼	설명	비고
WIDGET_ID	위젯ID	'W' + 랜덤 문자 조합 9글자
WIDGET_NM	위젯명	99 자 이내의 문자열
VALI_FRDD	유효시작일	yyyymmdd 형식의 문자열 날짜데이터
VALI_TODD	유효종료일	yyyymmdd 형식의 문자열 날짜데이터
POS_W	기본넓이	0 ~ 12 사이의 숫자 데이터
POS_H	기본높이	0 ~ 100 사이의 숫자 데이터
WIDGET_TYPE	위젯유형	'OBJECT' 'BATCH' 'ADMIM'
WIDGET_SRC_NM	위젯 소스 파일명 (리소스)	필요한 경우 리소스 파일명을 기입 SELECT * FROM ZZ.ZNBMCODE WHERE GRUP_NB_CD='DATASRCYPE'; 조회 데이터 중 NB_CD 컬럼 데이터 (STS REAL STSREAL BATCH 등)
DATA_SRC_TYPE	데이터 출처 유형	SZ.SMSMSVCI 참조 ZZ.ZNBMCDATA 참조 ZZ.ZNBMQSTMT 참조
SVC_CD	서브릿ID (서비스코드)	SZ.SMSMSVCI 참조
WIDGET_PROP_ID	속성ID	ZZ.ZNBMCDATA 참조
QSTMT_ID	쿼리ID	ZZ.ZNBMQSTMT 참조
WIDGET_DESC_ID	설명ID	'D' + 랜덤 문자 조합 9글자 (ZZ.ZNBMCDATA 참고)
WIDGET_IMGE_ID	이미지ID	'b' 나 'W' + 랜덤 문자 조합 9글자 (ZZ.ZNBMCDATA 참고)
DATA_ADD_YN	동적 JS 로딩 여부	페이지 로드 시점에 속성 템 스크립트 로드 여부 ('N' 설정 시 초기 시점에 로드하지 않음)
DRILL_DOWN_YN	드릴다운 기능여부	기본값 'N'
CONST_PARAMS	상수형 파라메터	

▼ WIDGET_TYPE

위젯 유형

유형	설명
OBJECT	일반 위젯
ADMIN	관리자용 위젯

유형	설명
REPORT	배치 레포트발송 위젯
BATCH	배치 위젯

▼ WIDGET_SRC_NM

위젯 소스 파일명

- 속성 탭에서 작성한 위젯 리소스가 아닌 외부 파일이 존재할 때 해당 컬럼에 파일명을 작성한다.
- 현재 속성 탭 이외의 파일은 사용할 수 없다.

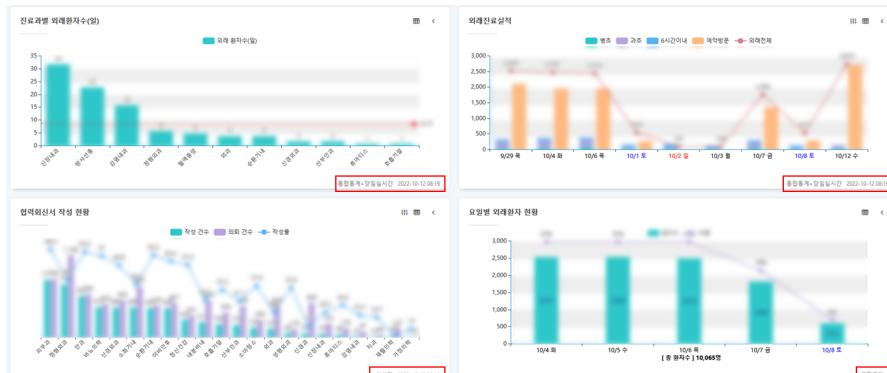
▼ DATA_SRC_TYPE

데이터 출처 유형

- `ZZ.ZNBMCODE` 참조

GRUP_NB_CD	NB_CD	NB_NM
"DATASRCTYPE" 고정		
	STSREAL	통합통계+당일실시간
	STS	통합통계
	REAL	실시간
	BATCH	배치
	UPLOAD	사용자업로드
	PART	부서통계
	EDW	EDW

- 위젯의 `div.card-footer` 요소에서 확인 가능



▼ SVC_CD

서비스코드

- nU와 같이 서비스 코드를 사용한다.
- `SZ.SMSMSVCI` 참조
- 일반적으로 '쿼리'탭의 쿼리문을 실행할 경우 서비스 코드 `DRZNBNB20204`를 사용한다.
- “BATCH” 유형의 위젯일 경우 해당 값은 없다. (-)

▼ WIDGET_PROP_ID

속성ID

- ZZ.ZNBMCMDATA 참조

CDATA_ID	CDATA_TYPE	CDATA
WIDGET_PROP_ID	"WIDGET_PROP" 고정	CLOB 속성 탭 스크립트

▼ QSTMID

쿼리ID

- ZZ.ZNBMQSTMT 참조

QSTMID_GRUP_CD	QSTMID	QSTMSEQ	QSTM
쿼리그룹ID	QSTMID (형태 : 쿼리그룹 ID.WIDGET_ID)	0 고정	CLOB 쿼리 탭 스크립트

▼ WIDGET_DESC_ID

설명ID

- ZZ.ZNBMCMDATA 참조

CDATA_ID	CDATA_TYPE	CDATA
WIDGET_DESC_ID	"WIDGET_DESC" 고정	CLOB HTML 형태로 변환된 설명 탭 스크립트

▼ WIDGET_IMGE_ID

이미지ID

- ZZ.ZNBMBDATA 참조

BDATA_ID	BDATA_TYPE	BDATA
WIDGET_IMGE_ID	"WIDGETIMGE" 고정	BLOB BLOB 형태로 변환한 이미지 데이터

▼ DATA_ADD_YN

동적 JS 로딩 여부 (Y | N)

- 값이 Y이고 WIDGET_PROP_ID (속성ID) 가 없고 WIDGET_SRC_NM (위젯리소스) 이 있는 경우, 위젯 리소스에 해당하는 파일이 로딩된다.
하지만 외부 JS 파일은 사용 불가능 하므로 '속성'탭의 스크립트를 읽어온다면 Y와 N의 차이는 없다.

▼ DRILL_DOWN_YN

드릴다운 기능 여부 (Y | N)

- 구현 예정

▼ CONST_PARAMS

상수형 파라미터

- 위젯 내에서 파라미터로 사용할 수 있다.
- '속성'탭에서 해당 파라미터 사용 시 PhisWidget 위젯 메소드 중 하나인 `getWidgetInfo()` 를 사용하여 파라미터를 확인할 수 있다.

```
const phisWidget = this;
phisWidget.getWidgetInfo('const_params');
```



1.2. 속성 - (1).1 공통규칙, (1).2 속성종류의 Method 참고

▼ (2) 위젯 안내

▼ 1. 위젯 상단 기능

The screenshot shows the 'Widget Master' application window. At the top, there's a toolbar with several buttons: 1. A red circle labeled '1' covers the 'Widget' button in the 'A.위젯' tab. 2. A red circle labeled '2' covers the 'SaveAs' button. 3. A red circle labeled '3' covers the 'Select...' button in the search bar. 4. A red circle labeled '4' covers the '추가' (Add) button.

Datetime	내용	widget_id	widget_nm (비고)
2022-10-04 08:31:22	속성저장	Wzb3QR756	1정동-정나윤
2022-10-04 08:31:14	SaveAs	Wzb3QR756	1정동-정나윤
2022-10-04 08:30:23	위젯저장	Wek4ammLE88	1정동-정나윤
2022-10-04 08:30:00	쿼리저장	Wek4ammLE88	1정동-정나윤
2022-10-04 08:29:54	속성저장	WekAmmLE88	1정동-정나윤
2022-10-04 08:29:33	SaveAs	WekAmmLE88	1정동-정나윤
2022-09-28 17:13:45	속성저장	Wz6bNwau1Y	1정동-정나윤
2022-09-28 17:11:54	쿼리저장	Wz6bNwau1Y	1정동-정나윤
2022-09-28 16:58:27	속성저장	Wz6bNwau1Y	1정동-정나윤
2022-09-28 10:25:04	속성저장	WxHdMxEa1Y	1정동-정나윤batch
2022-09-28 08:33:13	속성저장	Wz6bNwau1Y	1정동-정나윤
2022-09-28 08:32:31	쿼리저장	WxHdMxEa1Y	1정동-정나윤batch
2022-09-28 08:32:24	속성저장	WxHdMxEa1Y	1정동-정나윤batch
2022-09-28 08:31:50	위젯저장	WxHdMxEa1Y	1정동-정나윤batch
2022-09-28 08:31:39	SaveAs	WxHdMxEa1Y	1정동-정나윤
2022-09-28 08:29:22	쿼리저장	Wz6bNwau1Y	1정동-정나윤
2022-09-28 08:27:36	속성저장	Wz6bNwau1Y	1정동-정나윤
2022-09-28 08:27:15	위젯저장	Wz6bNwau1Y	1정동-정나윤
2022-09-28 08:26:53	SaveAs	Wz6bNwau1Y	1정동-정나윤
2022-09-27 17:50:23	속성저장	WMSVPhIFD5	1정동-정나윤

▼ ① A.위젯 콤보박스

- 위젯을 조회하여 마스터 정보를 확인한다. (위젯 ~ 권한 템)
- 위젯ID, 위젯명, 위젯 타입으로 검색이 가능하다.

▼ ② 새로고침 아이콘

- 'A.위젯'에 선택되어있는 위젯의 정보, 속성, 쿼리 등의 데이터를 갱신하고 싶은 경우에 클릭한다.

▼ ③ SaveAs 버튼

- '다른 이름으로 저장' 기능으로 'A.위젯' 정보를 복사하여 새로운 위젯으로 생성한다.
- 신규 위젯을 생성할 때 사용할 수 있다.

1. 새로 부여되는 항목

- 위젯 □ 속성 □ 설명 □ 이미지 ID는 랜덤 아이디 부여
- 쿼리ID는 'A.위젯'의 쿼리그룹ID + 랜덤으로 부여된 위젯ID로 설정 (예 : dev.Wizb3QR756)
- 위젯명은 '개발중-사용자명'으로 설정
- 사용시작일과 종료일은 '9999-12-31'로 부여 (수정이 반드시 필요)

2. 'A.위젯' 정보를 가져가는 항목

- 속성 □ 쿼리 □ 설명 □ 이미지 정보 (권한 정보는 복사되지 않는다.)
- 아래 사진 '위젯' 탭의 빨간 네모박스 안의 정보

The screenshot shows the 'Widget' configuration interface. At the top, there are tabs for '작업' (Job), '검색' (Search), '위젯' (Widget), '속성' (Properties), '쿼리' (Query), '설명' (Description), '이미지' (Image), and '권한' (Permissions). The 'Widget' tab is selected. Below the tabs, there's a message: '최초저장 : 안치원 2022-02-12 10:18 최종저장 : 안치원 2022-02-12 10:18'. The main area contains several input fields:

- ID: WIDGETMAST
- Name: Widget Master
- 사용시작일: 2022-02-12
- 사용종료일: 9999-12-31
- 기본넓이: 6
- 기본높이: 40
- 위젯유형: ADMIN
- 위젯리소스: masterwidget.js
- 데이터출처: REAL
- 서브위젯ID: -
- 속성ID: -
- 쿼리ID: dev.WIDGETMAST
- 설명ID: -
- 이미지ID: -
- 동작JS: Y
- 드릴다운: N
- 파라메터: (empty)

A large red box surrounds the entire 'Basic Properties' section, which includes the first six items listed above.

3. 주의사항

- 'A.위젯'의 속성 ~ 이미지 ID가 없는 경우 SaveAs 결과 (ID :)

	아이디 없음
속성 □ 설명	'A.위젯' 해당 항목 그대로 복사 (ID : <input type="text"/>)
이미지	랜덤ID 부여, Default 이미지 생성
쿼리	SaveAs 불가능

▼ ④ 추가 버튼

- 현재 페이지에 위젯을 추가 □ 실행하여 확인할 수 있다.
- 위젯이 없는 빈 공간에 기본 넓이와 높이로 설정한 크기로 추가된다.
- '속성' 탭에서 스크립트 저장 시 추가한 위젯이 변경된 스크립트로 자동 새로고침 된다.

▼ 2. 탭 별 기능

The screenshot shows the 'Widget Master' application interface. At the top, there's a navigation bar with various buttons and a search bar. Below the navigation bar is a table containing log entries for widgets.

Datetime	내용	widget_id	widget_nm (비고)
2022-10-04 08:31:22	속성저장	Wizb3QR756	속성등록
2022-10-04 08:31:14	SaveAs	Wizb3QR756	작업등록
2022-10-04 08:30:23	위젯저장	WekAmmLE88	위젯등록
2022-10-04 08:30:04	위젯저장	WekAmmLE88	위젯등록
2022-10-04 08:29:56	속성저장	WekAmmLE88	속성등록
2022-10-04 08:29:33	SaveAs	WekAmmLE88	작업등록
2022-09-28 17:13:45	속성저장	Wz6bNwauY	속성등록
2022-09-28 17:11:54	위젯저장	Wz6bNwauY	위젯등록
2022-09-28 16:58:27	속성저장	Wz6bNwauY	속성등록
2022-09-28 10:25:04	속성저장	WxHdMxEaYI	속성등록
2022-09-28 08:33:13	속성저장	Wz6bNwauY	속성등록
2022-09-28 08:32:31	위젯저장	WxHdMxEaYI	위젯등록
2022-09-28 08:32:24	속성저장	WxHdMxEaYI	속성등록
2022-09-28 08:31:50	위젯저장	WxHdMxEaYI	위젯등록
2022-09-28 08:31:39	SaveAs	WxHdMxEaYI	작업등록
2022-09-28 08:29:22	위젯저장	Wz6bNwauY	위젯등록
2022-09-28 08:27:36	속성저장	Wz6bNwauY	속성등록
2022-09-28 08:27:15	위젯저장	Wz6bNwauY	위젯등록
2022-09-28 08:26:53	SaveAs	Wz6bNwauY	작업등록
2022-09-27 17:50:23	속성저장	WM5VPnID0s	속성등록

▼ ① 작업

- 나의 위젯

나의 위젯:

- 하나의 위젯을 '나의 위젯'으로 설정 후 저장하여 해당 위젯을 언제든지 간편하게 마스터 조회할 수 있다.
- 위젯ID를 이용하여 저장하며, 위 사진과 같이 위젯ID 입력 후 '저장' 버튼을 클릭하면 세션에 해당 위젯 ID가 저장된다.
- 저장된 '나의 위젯'은 '검색' 버튼을 클릭하여 조회한다.

- B.위젯 / 위젯복사

- 'A.위젯' 정보를 'B.위젯'으로 복사한다. (**A.위젯** → **B.위젯**)
- ① B.위젯을 선택하고 ② '위젯복사' 버튼을 클릭하면 'B.위젯'으로 'A.위젯'의 정보가 이동 후 복사된다.
- 복사되는 정보는 아래와 같다.

- 표

- 작업 내역을 세션에 기록한다.
- SaveAs, 작업탭-위젯복사, 위젯 속성 쿼리 설명 이미지 권한 저장 등작의 기록을 볼 수 있다.
- 최대 30개까지 기록되며, 행 더블 클릭 시 해당 위젯이 조회된다.

Datetime	내용	widget_id	widget_nm (비고)
2022-10-05 14:48:02	위젯저장		
2022-10-05 14:22:38	위젯복사		
2022-10-05 14:22:00	속성저장		
2022-10-05 11:07:53	위젯복사		
2022-10-05 11:04:34	위젯복사		
2022-10-05 11:03:40	위젯저장		
2022-10-05 11:03:31	SaveAs		
2022-10-05 11:01:16	위젯복사		

▼ ② 검색

- 검색조건을 지정하여 조건에 맞는 위젯을 검색하는 탭이다.
- ① 검색조건을 선택한다. ② 검색내용을 입력하고 ③ 위젯검색 버튼 클릭 혹은 Enter 키를 입력하여 검색한다.
- 검색 결과로 나온 리스트 중 행을 더블클릭하면 해당 위젯이 조회된다.

The screenshot shows a search interface with a dropdown menu open under the '검색조건' (Search Condition) button. The menu items are: 전체(유효기간) (All (Effective Period)), 속성에서 찾기 (Search from Attribute), 쿼리에서 찾기 (Search from Query), and 설명에서 찾기 (Search from Description). The '쿼리에서 찾기' option is selected and highlighted with a red circle and the number 1. The main search table has columns: 시작일 (Start Date), 종료일 (End Date), 위젯유형 (Widget Type), 서브위IDGET (Sub-Widget ID), 속성ID (Property ID), and 이미지 (Image). A search input field '검색내용: inst_cd' contains 'inst_cd'. A search button 'Q. 위젯검색' is highlighted with a red circle and the number 2. A cursor is shown clicking on the search button, indicated by a red circle and the number 3.

▼ ③ 위젯

- 위젯 마스터 정보를 보여준다.
- ① 마스터 정보 편집 후 ② '위젯저장' 버튼을 클릭하여 저장한다.
- ZZ.ZNBMWIDGET 참조 (1.1. 마스터 - 개요 - Master Table 내용 참고)

항목명	컬럼명	설명
ID	WIDGET_ID	위젯ID
Name	WIDGET_NM	위젯명
사용시작일	VALI_FRDD	유효시작일
사용종료일	VALI_TODD	유효종료일
기본넓이	POS_W	위젯 넓이

항목명	컬럼명	설명
기본높이	POS_H	위젯 높이
위젯유형	WIDGET_TYPE	위젯 실행 유형
위젯리소스	WIDGET_SRC_NM	위젯 소스 파일명
데이터출처	DATA_SRC_TYPE	데이터 출처 유형
서브밋ID	SVC_CD	서비스코드
속성ID	WIDGET_PROP_ID	스크립트 CDATA 저장ID
쿼리ID	QSTMT_ID	쿼리 CDATA 저장ID
설명ID	WIDGET_DESC_ID	설명 CDATA 저장ID
이미지ID	WIDGET_IMGE_ID	이미지 BDATA 저장ID
동작JS	DATA_ADD_YN	동작 JS 로딩 여부
드릴다운	DRILL_DOWN_YN	드릴다운 기능 여부
파라메터	CONST_PARAMS	상수형 파라메터

The screenshot shows the 'Widget Master' configuration page with the '속성' (Properties) tab selected. The form contains the following fields:

- ID: WIDGETMAST
- Name: Widget Master
- 사용시작일: 2022-02-12
- 사용종료일: 9999-12-31
- 기본넓이: 6
- 기본높이: 40
- 위젯유형: ADMIN
- 위젯리소스: masterwidget.js
- 데이터출처: REAL
- 서브밋ID: -
- 쿼리ID: dev.WIDGETMAST
- 설명ID: -
- 이미지ID: -
- 동작JS: Y
- 드릴다운: N
- 파라메터: (empty)

⚠ 속성 □ 쿼리 □ 설명 □ 이미지 ID 항목의 두 가지 버튼



속성 □ 쿼리 □ 설명 □ 이미지를 사용하고 싶지 않을 때 사용한다.
클릭하면 해당 아이디는 - 가 되고 위젯 저장하면 해당 탭은 비활성화 된다.



아이디가 - 일 때 + 표시로 바뀌며 저장 시 새로운 ID를 랜덤으로 부여한다. (기존에 있던 각 탭 데이터 초기화)
쿼리ID의 경우 쿼리그룹 변경 팝업에서 원하는 쿼리그룹ID를 더블 클릭하여 편집한다.

▼ ④ 속성

- 위젯 화면, 동작 등을 “속성” 탭에 스크립트를 작성하여 구현한다.
- “위젯관리” 페이지 Widget Master (WIDGETMAST) 위젯의 “속성” 탭에서 스크립트를 작성 및 저장한다.

- 속성ID(`WIDGET_PROP_ID`)와 함께 `ZZ.ZNBMCDATA` 참조 (1.2. 속성 참고)

CDATA_ID	CDATA_TYPE	CDATA
WIDGET_PROP_ID	"WIDGET_PROP" 고정	CLOB 속성 템 스크립트

▼ ⑤ 쿼리

- “위젯관리” 페이지 내의 Widget Master (WIDGETMAST) 위젯의 “쿼리” 탭에서 각 위젯에 대한 쿼리 작성 및 저장한다.
- 쿼리ID(`QSTM_ID`)와 함께 `ZZ.ZNBMQSTM` 테이블 참조 (1.3. 쿼리 참고)

QSTM_GRUP_CD	QSTM_ID	QSTM_SEQ	QSTM
DAO코드	QSTM_ID (형태 : DAO코드.WIDGET_ID)	0	CLOB 쿼리 템 스크립트

▼ ⑥ 설명

- 위젯에 대한 설명을 작성하고 그 내용은 해당 위젯에서 확인할 수 있다.
- 사용자의 근무지, 부서, 직종, 직책 등에 따라 사용자에게 정보만 전달할 수 있다.
- 설명ID(`WIDGET_DESC_ID`)와 함께 `ZZ.ZNBMCDATA` 테이블 참조 (1.4. 설명 참고)

CDATA_ID	CDATA_TYPE	CDATA
WIDGET_DESC_ID	"WIDGET_DESC" 고정	CLOB HTML 형태로 변환된 설명 템 스크립트

▼ ⑦ 이미지

- Widget Gallery 위젯에서 위젯 추가 시 볼 수 있는 위젯 이미지이다.
- `ZZ.ZNBMCDATA` 테이블 참조

! 현재 이미지 템에서는 파일 업로드 기능을 제공하고 있지 않다.
필요한 경우 `ZZ.ZNBMCDATA` 테이블에 직접 **BLOB** 형태의 데이터를 추가한다.

작업	검색	위젯	속성	쿼리	설명	이미지	권한
최초저장 : 안치원 2022-02-07 09:49	최종저장 : null 2019-09-04 14:25					Select file or Drop file here	(3)
일자	접수건 수	예약접 수	당일 접수	진료 완료	수납 증	병원 조진	병조 진출
09/04	2265	2080	175	830	570	282	15%
09/03	2302	1825	367	1203	1202	206	13%
09/02	2545	2063	482	2545	2545	399	15% 0%
09/01	197	0	197	197	197	124	6%
08/31	4875	4764	4764	4079	4079	275	4800

컬럼	설명	비고
BDATA_ID	Bdata ID	'b' + 랜덤 문자 9글자 (위젯 마스터 정보의 이미지ID)
BDATA_TYPE	Bdata 유형	"WIDGETIMAGE" 고정
BDATA	Bdata	데이터 타입 : BLOB
BDATA_NM	Bdata명	위젯명

▼ ⑧ 권한

- 권한관리는 데이터의 성격에 따라 권한 관리자가 "사용자"의 "위젯" 혹은 "보드" 열람 범위를 지정하는 동작이다.
- `ZZ.ZNBMAUTH` 테이블 참조 (1.5. 권한 참고)

컬럼	설명
NBRD_ID	보드ID (<code>ZZ.ZNBMNBRD</code> 참조)
WIDGET_ID	위젯ID (<code>ZZ.ZNBMWIDGET</code> 참조)
SORT_NO	순서 (MAX+1)
INCLUDE_YN	Y : 포함조건 N : 제외조건
USER_GRUP_CD	사용자계정그룹코드
WRKPLC_CD	근무지코드
JOBKND_CD	직종코드
JOBRSP_CD	직책코드
DUT_INST_CD	근무등록기관코드
ORG_INST_CD	조직기관코드
USER_TYPE_CD	사용자유형코드
OCCGRP_CD	직군코드
PRFSHP_FLAG_CD	교직구분코드
JOBPOS_CD	직위코드

▼ 1.2. 속성

▼ (1) 개요

위젯 화면, 동작 등을 "속성" 탭에 스크립트를 작성하여 구현한다.

"위젯관리" 페이지 Widget Master (WIDGETMAST) 위젯의 "속성" 탭에서 스크립트를 작성 및 저장한다.

Javascript (es6 이하 권장), Html5, Css3 및 위젯함수, 공통함수 API 등으로 스크립트를 실행하여 위젯을 생성한다.



아래에서 다루는 속성은 `WIDGET_TYPE` 이나 `"OBJECT"`이나 `"ADMIN"`인 위젯으로 제한한다.

모든 타입에 적용할 수 있는 공통함수는 별도로 다룬다.

(3) 속성작성 - 3. 공통함수 참고)

Widget Master

A위치: Widget Gallery WWZrKZ5SJQ OBJECT

작업 검색 위젯 속성 뷰리 설명 이미지 권한

최초저장: 2022-06-27 10:22 최종저장: 2022-08-17 16:00

```

1 // 속성 2022-03-01
2 console.log('## Widget Gallery (ID: WWZrKZ5SJQ)');
3
4 const phisWidget = this;
5
6 let elGallery;
7
8 const widgetProp = {
9   title: 'Widget Gallery',
10  widgetHTML: 'CARD-FOOTERLESS',
11  enabledSlimScroll: false,
12  onBeforeLoad() {
13    const elWidget = phisWidget.getContentEl();
14    elGallery = phisQ(`<div class="ph-widget-gallery">
15      <div class="ph-header">
16        <div class="status"></div>
17      </div>
18      <div class="ph-list"></div>
19    </div>`);
20
21  const reqData = {
22    page_id: phisWidget.getWidgetInfo('page_id'),
23    nbrd_id: location.search.split('?bid=')[1].split('&ver=')[0],
24    admin_level: 'SUPER',
25  };
26
27  gfn_submit('DRZN810301', 'req', reqData,
28  function(responseInfo, isSuccess, errMsg) {
29    if (isSuccess) {
30      elWidget.append(elGallery);
31      phis_gethtmlTemp(responseInfo.gallery, elGallery);
32      elWidget.dxScrollView({
33        scrollByContent: true,
34        scrollByThumb: true,
35      });
36    } else {
37      phisWidget.notify('danger', errMsg);
38  }
39}
40
41}
42

```

▼ (2) 기본 동작

Widget Master

A위치: Widget Gallery WWZrKZ5SJQ OBJECT

작업 검색 위젯 속성 뷰리 설명 이미지 권한

최초저장: 2022-06-27 10:22 최종저장: 2022-09-23 13:40

Widget 속성 편집창

속성 2022-09-23 13:40:20
속성 2022-06-17 16:05:04
속성 2022-06-27 15:29:46
속성 2022-06-27 15:29:40
속성 2022-06-27 15:29:34

```

1 const phisWidget = this;
2
3 let elGallery;
4
5 const widgetProp = {
6   title: 'Widget Gallery',
7   widgetHTML: 'CARD-FOOTERLESS',
8   enabledSlimScroll: false,
9   onBeforeLoad() {
10    const elWidget = phisWidget.getContentEl();
11    elGallery = phisQ(`<div class="ph-widget-gallery">
12      <div class="ph-header">
13        <div class="status"></div>
14      </div>
15      <div class="ph-list"></div>
16    </div>`);
17
18  const reqData = {
19    page_id: phisWidget.getWidgetInfo('page_id'),
20    nbrd_id: location.search.split('?bid=')[1].split('&ver=')[0],
21    admin_level: 'SUPER',
22  };
23
24  gfn_submit('DRZN810301', 'req', reqData,
25  function(responseInfo, isSuccess, errMsg) {
26    if (isSuccess) {
27      elWidget.append(elGallery);
28      phis_gethtmlTemp(responseInfo.gallery, elGallery);
29      elWidget.dxScrollView({
30        scrollByContent: true,
31        scrollByThumb: true,
32        showScrollbar: 'onscroll',
33      });
34    } else {
35      phisWidget.notify('danger', errMsg);
36    }
37}
38
39}
40
41}
42

```

① 에디터 박스에 속성을 작성한다.



스크립트 작성 유의사항

- Widget Master 위젯의 '속성' 탭에서 스크립트 작성 시, 요소를 전역 선택자로 선택할 수 없다.
위젯 단위 속성 적용 시, 동일 페이지 내 다른 위젯에 스크립트가 적용되는 것을 방지하기 위함이다.
 - \$.document.getElementBy, document.getElementsBy, document.querySelector 등을 적용할 수 없음

② '속성저장'을 클릭한다.

③ '변경이력' 버튼을 클릭하여 변경이력 팝업으로 해당 위젯의 속성 과거 저장 이력을 확인 할 수 있다. (더블 클릭으로 각 이력 확인)

▼ (3) 속성 작성

▼ 1. 공통 규칙

| 아래는 속성작성 시 유형이 "OBJECT" 인 위젯에 적용되는 규칙이다.

1. 속성 탭 내에서의 `this` 는 PhisWidget 인스턴스 객체이다.



"속성" 탭 내 작성된 소스는 함수명이 "phsGetWidgProp_" + 위젯속성ID로 조합된 함수로 스코핑 된 후,
위젯로드 시 apply 메소드로 실행된다.

this 객체는 apply 메소드 실행 시 PhisWidget 인스턴스 객체로 전달된다.
(= this 객체를 통해 PhisWidget 메소드를 실행할 수 있음)

2. 속성을 변수명 `widgProp`에 key-value (속성명-속성값) 형태로 데이터를 할당한다.



"속성" 탭 내에서 변수명 `widgProp` 변수는 선언없이 데이터만 할당하면 된다.

(= 속성 스크립트 실행 전에 `widgProp` 이 var 타입으로 선언되어 있는 상태)

때문에 `widgProp` 변수를 재선언하지 않고, "속성명-속성값" 을 "key-value" 형태로 할당해주기만 한다.

- “속성” 탭 내 스크립트 최종형태 예시

```

function phisGetWidgProp_DzfcluUGUm() {
    // WWZrKZ5SJQ Widget Gallery
    console.log('function phisGetWidgProp_DzfcluUGUm start. ' + this.getWidgetInfo('widget_nm'));
    var widgProp = null;
    try {
        /** 구현부 시작 *****/
        const phisWidget = this;
        widgProp = { ... }
        function phis_getHtmlTemp(data, elTarget) { ... }
        /** 구현부 종료 *****/
    } catch(err){
        console.error(err);
        throw err;
    }
    return widgProp;
}

```

```

1: const phisWidget = this;
2: let elGallery;
3: ...
4: widgProp = {
5:     title: 'Widget Gallery',
6:     widgetHtml: 'CARD-FOOTERLESS',
7:     enableSlideCell: false,
8:     onBeforeLoad() {
9:         ...
10:     },
11:     onAfterLoad() {
12:         elGallery = phisXQC('div#ph-widget-gallery');
13:         elGallery.innerHTML = phisWidget.getContextEl();
14:         elGallery.innerHTML += phis_getHtmlTemp(data, elTarget);
15:         ...
16:     }
}

```

“속성” 탭 내 스크립트 소스는 try-catch 문의 try 블록으로 로드된다.

스크립트 최종 로드형태

```

function phisGetWidgProp_DzfcluUGUm() {
    // WWZrKZ5SJQ Widget Gallery
    console.log('function phisGetWidgProp_DzfcluUGUm start. ' + this.getWidgetInfo('widget_nm'));
    var widgProp = null;
    try {
        /** 구현부 시작 *****/

```

```

// PhisWidget 인스턴스객체
const phisWidget = this;

// 위젯속성
widgProp = {
    title: 'Widget Gallery',
    widgetHtml: 'CARD-FOOTERLESS',
};

// 기타 스크립트
function phis_getHtmlTemp(data,
elTarget) { ... };

```

```

// PhisWidget 인스턴스객체
const phisWidget = this;

// 위젯속성
widgProp = {
    title: 'Widget Gallery',
    widgetHtml: 'CARD-FOOTERLESS',
};

// 기타 스크립트
function phis_getHtmlTemp(data, elTarget) { ... };

```

["속성” 탭 내 작성된 스크립트]

```

/** 구현부 종료 *****/
} catch(err){
    console.error(err);
    throw err;
}
return widgProp;
}

```

▼ 2. 속성 종류

widgProp 변수에 설정한 속성을 위젯 속성이라 한다.

widgProp에 위젯속성을 “속성명-속성값”(key-value) 형태의 Object 데이터로 할당한다.

▼ 1) Configuration

로드할 위젯의 고정동작을 설정한다.

속성명	기능 설명	데이터형 및 기본값
title	위젯 타이틀 설정	[String] ZZ.ZNBMWIDGET 테이블 WIDGET_NM 의 컬럼값
widgetHtml 필수	위젯 컨테이너 형태 설정	[String Function] null
viewOption 필수	위젯 데이터를 보여줄 위젯 뷰 설정	[JSON Array] []
viewIndex	viewOption 설정 컨텐츠가 2개 이상일 때 가장 먼저 보일 컨텐츠 index	[Number] 0
customizeData	submit 으로 가져온 데이터를 가공	[String Function] null
toolBox	위젯 부가기능을 툴박스로 적용	[Array] ["reload", "info", "max"]
srchCond	위젯 데이터를 필터할 필터요소 설정	[JSON Array] []
visibleLoadingBar	위젯 로딩바UI 적용	[Boolean] true
enabledInitDataLoad	위젯 데이터 초기에 로딩 여부	[Boolean] true
enableChangePid	위젯 내 환자정보 변경여부	[String Function] false
enabledSlimScroll	위젯 스크롤 적용여부	[Boolean] false
initMaxWidget	로드 초기에 위젯 최대화 (페이지당 위젯 1개만 설정)	[Boolean] false
responsiveHeight	window.innerHeight 에 따라 반응형으로 창크기 조절	[Number] 0
behindMaxWidget	위젯 최대화 시 팝업창 맨 앞으로 보내기 설정	[Boolean] false

▼ title

위젯 타이틀을 설정한다.

```
widgProp = {
    title: "위젯 타이틀이다.",
    ...
}
```

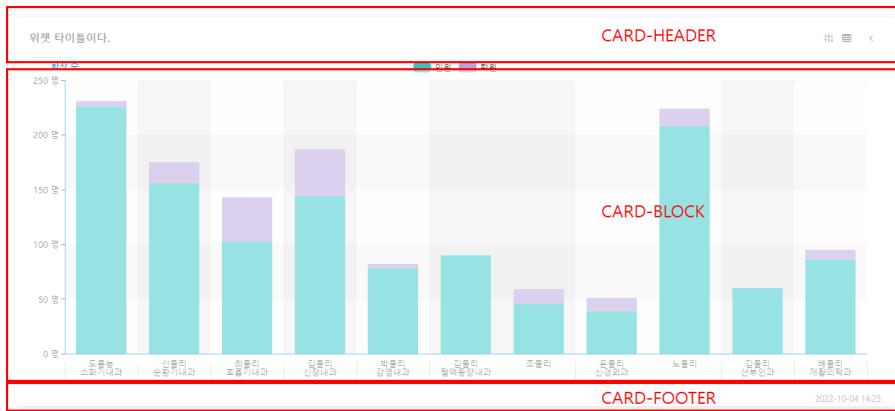
- div.widget-header h5 요소의 text 로 title 속성값을 적용한다.
- 속성값이 없는 경우, ZZ.ZNBMWIDGET 테이블 WIDGET_NM 의 컬럼값으로 기본 설정된다.



▼ widgetHtml 필수

위젯 컨테이너 형태를 설정한다.

위젯 구성 시 필수 적용 값이며, Default 값이 없으므로 반드시 적용해야 한다.



위젯 컨테이너 기본구조

[widgProp.widgetHtml = "CARD" 인 위젯 컨테이너 dom 구조]

```
<!--
widgProp = {
    widgetHtml: "CARD", // "CARD" 설정 시 div.card-header, block, footer 요소 모두 구성
    ...
}
-->

<div class="card widget-card grid-stack-item-content ... 생략" >
    <div class="card-header widget-header">
        <!-- CARD-HEADER 내용 -->
    </div>
    <div class="card-block">
        <!-- CARD-BLOCK 내용 -->
    </div>
    <div class="card-footer">
        <!-- CARD-FOOTER 내용 -->
    </div>
</div>
```

• CARD

- 구성요소: div.card-header, div.card-block, div.card-footer
- `ZZ_ZNBMWIDGET` 테이블 `DATA_SRC_TYPE` 의 컬럼값을 div.card-footer 요소에 표현

! div.card-footer 요소에 최종으로 입력되는 값은 `SELECT * FROM ZZ.ZNBMCODE WHERE GRUP_NB_CD='DATASRCTYPE'` 조회 데이터 중, `NB_NM` 컬럼값에 해당한다.

• CARD-Footerless

- 구성요소: div.card-header, div.card-block
- div.card-footer 요소가 없는 형태

• CARD-HEADER

- 구성요소: div.card-header

! div.card-block 와 div.card-footer 요소가 없기 때문에, div.card-block 요소를 구성하기 위한 이벤트 실행 시 에러가 발생할 수 있다.

- CARD-BLOCK

- 구성요소: div.card-block

▼ viewOption 필수

위젯 데이터를 보여줄 위젯 뷰를 설정한다.

viewOption 속성값에 따라 위젯 데이터를 차트, 그리드, 기타 형태로 나타낼 수 있다.

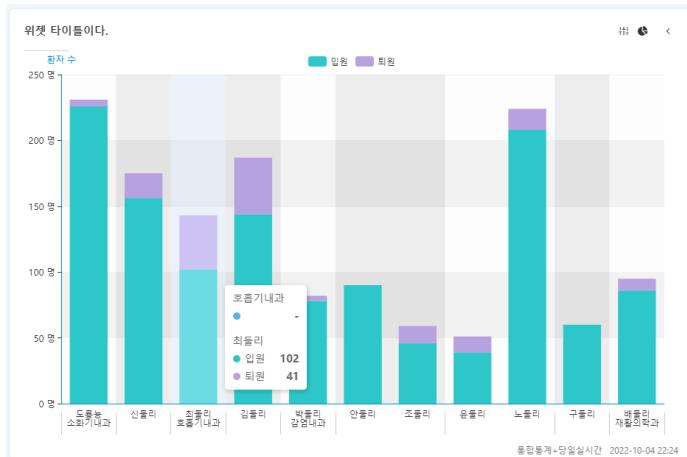
- 필수 속성

속성명	기능 설명	속성값
name 필수	위젯 데이터를 보여줄 위젯 뷰의 형태	"chart": 차트 (막대, 원형, 선형 등) "grid": 그리드 형태로 "treelist": 트리 그 리드 "prog": 스크립트로
option	viewOption.name 설정에 따른 뷰 옵션	아래 개별 속성 참고

- viewOption.name 값에 따른 개별 속성

▼ 1. "chart"

차트 형태의 위젯 뷰



- Apache Echarts 5.1.1 버전 API로 차트를 구현하므로, 차트 생성과 관련한 다양한 예제는 아래 링크를 참고

<https://echarts.apache.org/examples/en/index.html#chart-type-line>

Examples - Apache ECharts

ECharts, a powerful, interactive charting and visualization library for browser

🔗 <https://echarts.apache.org/examples/en/index.html>

- chart 속성

```

widgProp = {
    viewOption: [
        {
            name: 'chart',
            option: {
                phisStartXpos:0, // 초기에 X축에 표시되는 항목의 시작위치
            },
            drill: {
                name: 'thru', // 혹은 'down'
                /* 아래 속성은 drill.name='thru' 일 때만 적용 */
                targetWidget: [ '위젯Id1', '위젯Id2', ... ],
                firstDrillData: { ... },
                handler(p) { ... }
            } ...
        } ...
    ]
}

```

속성명	기능 설명	데이터형 및 설정값	설정값 설명	세부 설정값
option.phisStartXpos	X축에 표시되는 항목의 시작 위치	Number	dataZoom[0].startValue 값으로 설정 됨	-
drill	차트 click 이벤트로 데이터 drillThru			
		name	'thru' 설정 시 데이터 drillThru 동작, 'down' 설정 시 기본 동작으로 데이터 로드	-
		targetWidget	데이터 thru 할 위젯ID 를 String in Array 타입으로 기입 (ex. targetWidget: ['위젯 Id1', '위젯 Id2', ...])	-
		firstDrillData	로드 초기 시점에 drillThru 로 전달할 데이터	{ rowIndex: -1 (가장 마지막데이터), 특정 index, colName: 전달 데이터 key, }
		handler	firstDrillData 의 데이터 가공이 필요한 경우 적용하는 이벤트	-

- 내부에서 기본으로 적용되는 Echart 속성

```

const ECHART_OPTION = {
    tooltip: {
        trigger: 'axis',
        axisPointer: {
            type: 'shadow'
        }
    },
    legend: {
        align: 'left'
    },
    grid: {
        borderWidth: 0,
        top: '30px',
        bottom: '20px',
        left: '50px',
        right: '20px'
    },
    xAxis: {
        type: 'category'
    },
    yAxis: {
        type: 'value',
        show: true
    }
}

```

```

        },
        dataZoom: [
            {
                type: 'inside',
                show: false,
                xAxisIndex: [0],
            }
        ]
    }
}

```

▼ 2. "grid" / "treelist"

그리드, 트리그리드 형태의 위젯 뷰

위젯 타이틀이다.			
진료과	진료의	처방	재원
소화기내과	도봉형	226	5
순환기내과	신율리	156	19
호흡기내과	최율리	102	41
신장내과	김율리	144	43
감염내과	박율리	78	4
혈액증상내과	안율리	90	0
혈액증상내과	조율리	46	13
신경외과	송율리	39	12
신경외과	노율리	208	16
신부인과	구율리	60	0
재활의학과	비율리	86	9

위젯 타이틀이다.			
진료과	진료의	처방	재원
소화기내과	도봉형	226	5
순환기내과	신율리	156	19
호흡기내과	최율리	102	41
신장내과	김율리	144	43
감염내과	박율리	78	4
혈액증상내과	안율리	90	0
혈액증상내과	조율리	46	13
신경외과	송율리	39	12
신경외과	노율리	208	16
신부인과	구율리	60	0
재활의학과	비율리	86	9

왼쪽은 "grid" 위젯 뷰, 오른쪽은 "treelist" 위젯 뷰 예시

- DevExtreme 19.2 버전 API 의 속성 중 dxDataGrid, dxTreeList 로 그리드를 구현하는 방법은 생성 예제는 아래 링크를 참고

- dxDataGrid

https://js.devexpress.com/Documentation/ApiReference/UI_Components/dxDataGrid/

Documentation: DevExtreme - JavaScript Data Grid

The DataGrid is a UI component that represents data from a local or remote source in the form of a grid. This UI component offers such basic features as sorting, grouping, filtering, as well as more advanced capabilities, like state storing, client-side exporting, master-detail interface, and many others.

 https://js.devexpress.com/Documentation/ApiReference/UI_Components/dxDataGrid/

- dxTreeList

https://js.devexpress.com/Documentation/ApiReference/UI_Components/dxTreeList/

Documentation: DevExtreme - JavaScript Tree List

The TreeList is a UI component that represents data from a local or remote source in the form of a multi-column tree view. This UI component offers such features as sorting, filtering, editing, selection, etc.

 https://js.devexpress.com/Documentation/ApiReference/UI_Components/dxTreeList/

- grid 속성

```

widgProp = {
    viewOption: {
        name: 'grid',
        option: {
            columns: [
                {
                    caption: '컬럼타이틀1',
                    width: '30%', // 문자기입(px)
                    dataField: '컬럼데이터key1'
                } ...
            ]
        } ...
    }
}

```

- treelist 속성

```

widgProp = {
    viewOption: {
        name: 'treelist',
        option: {
            keyExpr: '컬럼데이터key1',
            // keyExpr: 트리 기준노드 컬럼데이터key
            columns: [
                {
                    caption: '컬럼타이틀1',
                    width: '30%', // 문자기입(px)
                    dataField: '컬럼데이터key1'
                } ...
            ]
        }
}

```

```
    } ...  
    } ...
```

- 내부에서 기본으로 적용되는 dxDataGrid, dxTreeList 속성

```
const DXGRID_OPTION = {  
  phisShowRowNumber: false, // nu option 행번호 표시여부  
  showBorders: true,  
  allowColumnResizing: true, // 사용자가 열 크기를 조정할 수 있는지 여부  
  columnAutoWidth: true,  
  columnResizingMode: 'widget',  
  hoverStateEnabled: true,  
  cellHintEnabled: true, // 내용이 잘린 셀 위로 마우스 포인터를 가져갈 때 표시되는 힌트를 사용  
  selection: {  
    mode: "single"  
  },  
  scrolling: {  
    // mode: "virtual",  
    rowRenderingMode: "standard"  
  },  
  paging: {  
    enabled: false  
  },  
  export: {  
    enabled: false,  
    fileName: 'downLoad'  
  },  
  noDataText: ""  
}
```

▼ 3. "prog"

사용자가 원하는 형태로 데이터를 시각화 한 위젯 뷰

chart, grid, treelist 포함, Javascript 를 활용하여 사용자가 원하는 뷔를 자유롭게 생성 가능



- "prog" 설정 시에만 제공되는 속성

```
widgProp = {  
  viewOption: [  
    {  
      name: 'prog',  
      initProg(contEl) { ... },  
      loadProgData(reqData) { ... },  
    }, ...  
  ]  
}
```

속성명	기능 설명	파라미터	파라미터 설명
initProg	데이터 조회 전, onBeforeLoad 이벤트 발생 이전에 발생할 위젯 뷰 영역 div.phis_widg_cont 요소를 파라미터로 제공하므로 요소에 append 하여 뷰 구성 가능	div.phis_widg_cont 요소	
loadProgData	onLoaded 이벤트 발생 이후, 위젯 쿼리Id 로 데이터를 조회하여 바인드 하기 위해 request 전달 데이터를 파라미터로 제공함+`~~`	쿼리 조회를 위한 req 데이터	
		{ qstmt_id	ZZ.ZNBMWIDGET.QSTMT_ID 참조
		, data_src_type	ZZ.ZNBMWIDGET.DATA_SRC_TYPE 참조
		, svc_cd }	ZZ.ZNBMWIDGET.SVC_CD 참조

▼ viewIndex

페이지 로드 직후에 위젯에 먼저 보일 컨텐츠의 인덱스를 설정한다.

viewOption 속성값이 JSON Array 타입이고 배열 길이가 2 이상인 경우에 적용된다.

- viewIndex 속성 적용예시

```
widgProp = {
  viewOption: [
    {
      name: 'chart',
      option: { .. }
    },
    {
      name: 'grid',
      option: { .. }
    },
    ...
  ],
  viewIndex: 1,
  ...
}
```

▼ customizeData

위젯 쿼리 조회 데이터 중 format 이 필요한 날짜 데이터를 가공한다.

“월/일 + 요일” 데이터, '/' 로 format 된 날짜 데이터를 가공 및 추가하여 최종 바인드한다.



왼쪽은 type: "month" 속성 적용 화면, 오른쪽은 type: "dateDay" 속성 적용 화면

- customizeData 설정값 및 기능

```
widgProp = {
  customizeData: [
```

```

{
    type: 'dateDay',
    dataField: 'orddd',
    // 최종 dataField 명은 "disp_" + "dataField 속성값" = 'disp_orddd'
    // prefix 로 'disp_' 가 조합되는 형태가 Default
},
{
    type: 'month',
    dataField: 'opcnfm_ym',
    newDataField: 'disp_opcnfm_ym',
},
...
]
...
}

```

◦ BEFORE - 조회 데이터 (원형)

```

[
    {
        "orddd": "20220905",
        "opcnfm_ym": "202109",
        ...
    },
    {
        "orddd": "20220906",
        "opcnfm_ym": "202109",
        ...
    },
    ...
]
...

```

◦ AFTER - 바인드 데이터 (가공)

```

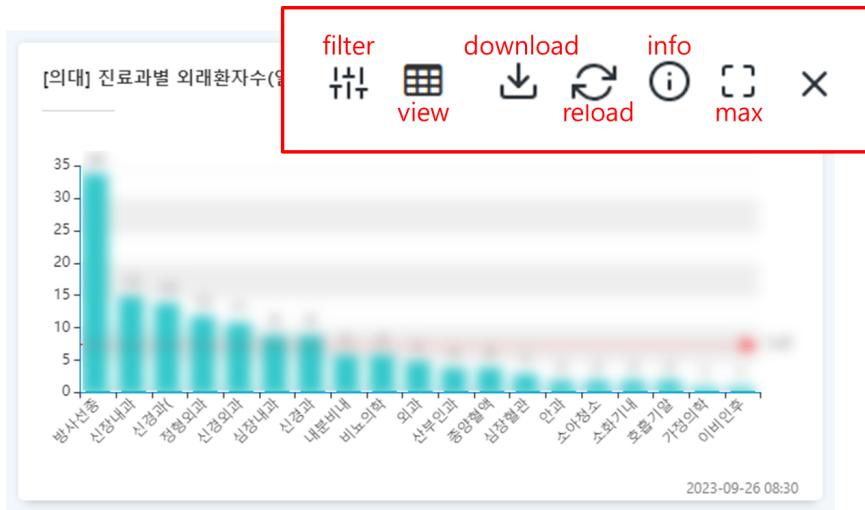
[
    {
        "orddd": "20220905",
        "opcnfm_ym": "202109",
        "disp_opcnfm_ym": "2021/09",
        "disp_orddd": "9/5 월" ...
    },
    {
        "orddd": "20220906",
        "opcnfm_ym": "202110",
        "disp_opcnfm_ym": "2021/10",
        "disp_orddd": "9/6 화" ...
    },
    ...
]
...
```

속성명	설명 및 설정값	설정값 설명
type	날짜 데이터 유형을 설정	
	"dateDay"	"월/일" + "요일" 형태
	"month"	"YYYY/MM" 형태
dataField	타겟 데이터의 JSON Key	쿼리 조회 데이터 중, 날짜 포맷으로 가공할 대상 데이터의 Key 설정
newDataField	가공 결과 데이터를 바인드 데이터로 추가 시 적용할 데이터의 Key 설정	최종 dataField 명은 "disp_" + "dataField 속성값" = 'disp_orddd' (prefix 로 'disp_' 가 조합되는 형태가 Default)

▼ toolBox

위젯의 부가기능을 툴박스의 기능으로 적용한다.

설정값이 없어도 "reload", "info", "max" 기능은 기본 적용된다.



- toolbox 설정값 및 기능

```
widgProp = {
    toolbox: ['filter', 'download', 'view'], // "reload", "info", "max" 는 기본설정값
    ...
}
```

툴바 타입	기능	비고
reload	위젯 데이터 재조회 및 컨텐츠 뷰에 재바인드 기능 버튼 추가	※ widgProp.viewOption.name 속성값이 'prog' 이면 loadProgData 함수 파라미터로 위젯 데이터를 전달하기만 하고, 재조회 및 재바인드는 동작하지 않는다.
info	위젯 ID 확인 및 “설명” 탭으로 작성된 내용 (1.4. 설명 참고)을 팝업창으로 확인 기능 버튼 추가	-
max	위젯 전체화면 모드로 변경, div.phis-board-main > div 영역 크기로 위젯을 최대화 하는 기능 버튼 추가	※ 위젯 최대화 시, 보드 상단 툴바의 페이지 수 정모드는 비활성화 한다.
view	위젯 컨텐츠 뷰가 'chart' 와 'grid' 로 토글하여 적용하는 기능 버튼 추가	※ widgProp.viewOption.name 속성값으로 'chart' 와 'grid' 가 모두 적용되어야 한다.
download	위젯 컨텐츠 뷰가 'grid' 인 경우(viewOption 의 name='grid')에 데이터를 엑셀다운로드 기능 버튼 추가	※ widgProp.viewOption.name 속성값으로 'grid' 가 적용되어야 한다.
filter	위젯 데이터를 필터링하여 컨텐츠 뷰 데이터로 적용하는 기능 버튼 추가	※ widgProp.srchCond 속성값이 적용되어야 한다.

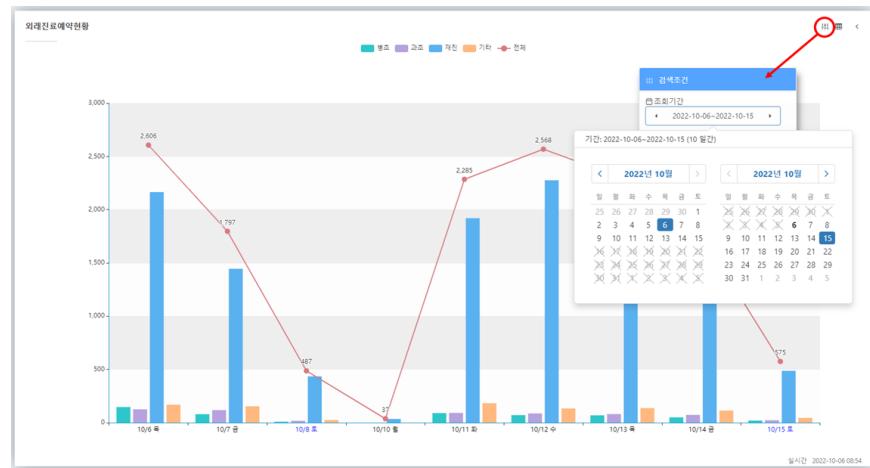
▼ srchCond

쿼리로 조회 된 데이터를 필터링하여 위젯 뷰에 바인드한다.

srchCond 속성으로 적용된 필터 타입에 따라 생성된 필터UI로 필터링을 적용할 수 있다.



필터UI를 적용하기 위해서는 srchCond 와 함께 **toolBox : ['filter']** 를 반드시 적용해야 한다.



- srchCond 공통 속성

```
widgProp = {
  srchCond: [
    {
      type: '필터타입',
      key: '필터 컴포넌트Id',
      default: '기본값' // 기본값(숫자),
      label: '컴포넌트 라벨',
      labelStyle: '컴포넌트 라벨 style 속성값',
      ...
    }, ...
  ], ...
}
```

속성명	기능 설명	속성값
type 필수	필터 타입 설정	[String] filter_y, filter_ym, filter_ymd, filter_ymd2, filter_wk, filter.cbo, filter.cbo2, filter.rdo, filter.chk, filter.lbl, filter.txt
key 필수	필터UI에 설정될 컴포넌트의 Id (div.phis-filter 요소의 id 속성값으로 적용)	[String] 사용자 지정
default	필터 컴포넌트에 적용될 초기 기본값	[String Number] 필터 타입에 따라 다름
label	필터UI에서 컴포넌트 라벨로 설정될 텍스트	[String] 사용자 지정
labelStyle	컴포넌트 라벨에 적용할 라벨 스타일 속성값 (div.phis-filter-label.filter_label 요소의 style 속성값으로 적용)	[String] style 속성 key-value

- srchCond 설정 가능한 필터

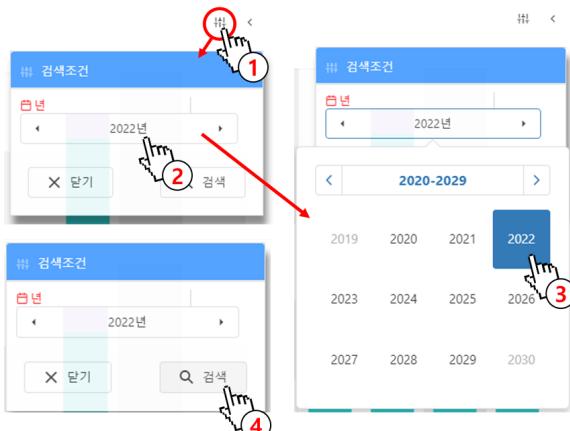
필터 종류	필터명	필터 기준	결과값	결과값 예제
캘린더	filter_y	년	[String] "YYYY" 년도 4자리	{ key_calendar_y : "2022" }
	filter_ym	년, 월	[String] "YYYYMM" 년도 6자리	{ key_calendar_ym : "202209" }
	filter_ymd	년, 월, 일	[String] "YYYYMMDD" 년도 8자리	{ key_calendar_ymd : "20221006" }
	filter_wk	주간 (년, 월, 일)	[String] (설정key1) "YYYYMMDD" 년도 8자리, (설정key2) 설정key1의 날짜 + 7일	{ key_calendar_w1: "20220926", key_calendar_w2: "20221001" }

필터 종류	필터명	필터 기준	결과값	결과값 예제
	filter_ymd2	기간 (년, 월, 일)	[String] (설정key1) "YYYYMMDD" 년도 8자리, (설정key2) "YYYYMMDD" 년도 8자리	{ key_calendar_ymd21: "20220930", key_calendar_ymd22: "20221006" }
콤보	filter.cbo	사용자 선택값 1개	[String] 사용자가 선택한 콤보Value 1개	{ key.cbo : "A" }
	filter.cbo2	사용자 선택값 1개	[String] 사용자가 선택한 콤보Value 1개 (※ filter.cbo 와의 차이점은 아 래 설명참고)	{ key.cbo2 :"2010500000" }
라디오	filter.rdo	사용자 선택값 1개	[String] 사용자가 선택한 라디오Value 1개	{ key.rdo : "A" }
체크박스	filter.chk	사용자 선택값 1개	[String] 사용자가 선택한 체크박스Value 1개 (※ filter.chk 내 items 속성값으 로 설정)	{ key.chk : ['A', 'B', 'C'] }
라벨	filter.lbl	필터기능 없음	결과값 없음	-
텍스트	filter.txt	사용자 기입값 1개	[String] 사용자가 기입한 텍스트	{ key.txt : "입력값" }

▼ 캘린더 종류 필터

- **filter_y**

- filter_y 속성 및 UI 화면



filter_y 설정에 따른 필터UI 화면

```
widgProp = {
  srchCond: [
    {
      type: 'filter_y',
      key: 'key_calendar_y',
      default: 0,
      label: '년',
      labelStyle: 'width:200px;color:#FF0000;border-right:1px solid lightgray;',
      // format: "Year"
    },
    ...
  ],
  ...
}
```

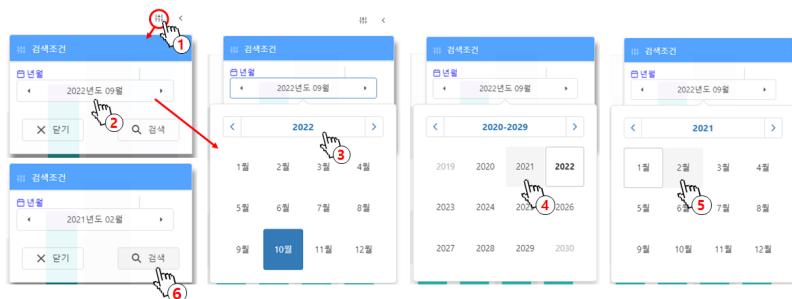
속성명	기능설명	설정값	설정예시
default	년도 데이터를 기본값으로 설정	Number 오늘 기준 년도 + default 설정값으로 계산한 값 (Default 0)	default 가 -1이고 오늘 날짜가 2022년이면, "2021"을 기본값으로 설정
format	Textbox 컴포넌트에 출력되는 날짜 데이터의 포맷 설정	String 년도 데이터 뒤에 적용될 문구 (Default "년")	-

- filter_y 필터 결과값

```
{ key_calendar_y : "YYYY 형식 년 데이터" } // "2022"
```

- filter_ym

- filter_ym 속성 및 UI 화면



filter_ym 설정에 따른 필터UI 화면

```
widgProp = {
  srchCond: [
    {
      type: 'filter_ym',
      key: 'key_calendar_ym',
      default: 0,
      label: '년월',
      format: '년도^월', // ('^'는 구분자)
      labelStyle: 'width:200px;color:blue;border-right:1px solid lightgray;'
    }, ...
  ], ...
}
```

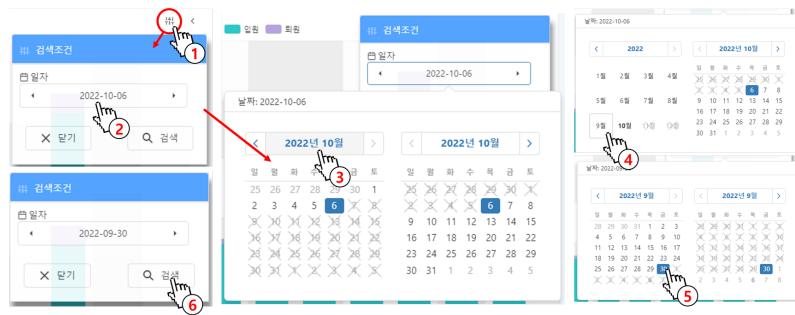
속성명	기능설명	설정값	설정예시
default	년, 월 데이터를 기본값으로 설정	Number 오늘 기준 월 + default 설정값으로 계산한 값 (Default 0)	default 가 -1이고 오늘 날짜가 2022년 10월이면, "202210"을 기본값으로 설정
format	Textbox 컴포넌트에 출력되는 날짜 데이터의 포맷 설정	String 년도 데이터 뒤에 적용될 문구 (Default "년^월")	-

- filter_ym 필터 결과값

```
{ key_calendar_ym : "YYYYMM 형식 년, 월 데이터" } // "202109"
```

• filter_ymd

◦ filter_ymd 속성 및 UI 화면



filter_ymd 설정에 따른 필터UI 화면

```
widgProp = {
    srchCond: [
        {
            type: 'filter_ymd',
            key: 'key_calendar_ymd',
            default: 0,
            label: '일자', // label 제외한 너비로 지정
            ...
        },
        ...
    ],
    ...
}
```

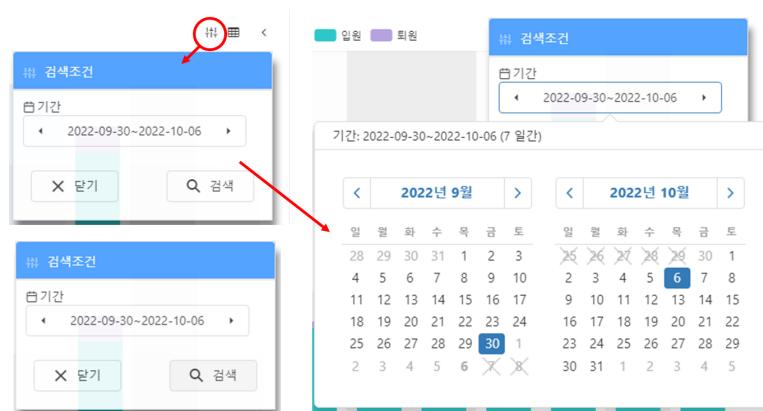
속성명	기능설명	설정값	설정예시
default	날짜 데이터를 기본값으로 설정	Number 오늘날짜 + default 값으로 계산한 값 (Default 0)	default 가 -1이고 오늘 날짜가 2022년 10월 06일 이면, "20221005" 를 기본값으로 설정

◦ filter_ymd 필터 결과값

```
{ key_calendar_ymd : "YYYYMMDD 형식 날짜 데이터" } // "20220930"
```

• filter_wk

◦ filter_wk 속성 및 UI 화면



filter_wk 설정에 따른 필터UI

```

widgProp = {
    srchCond: [
        {
            type: 'filter_wk',
            key: 'key_calendar_w',
            default: -1,
            label: '주',
            ...
        },
        ...
    ],
    ...
}

```

속성명	기능설명	설정값	설정예시
default	주간을 기본 데이터로 설정	Number 오늘날짜가 기준이 되는 주간의 시작날짜 + (default 설정값 * 7) 을 계산한 값 (Default 0)	default 가 -1 이고 오늘 날짜가 2022년 10월 06일 이면, 2022년 10월 6일이 포함된 주간의 시작 날짜인 20221003 + (-1 * 7) 결과값 "20220926"과 "20221001"을 각각 기본 값으로 설정

- filter_wk 필터 결과값

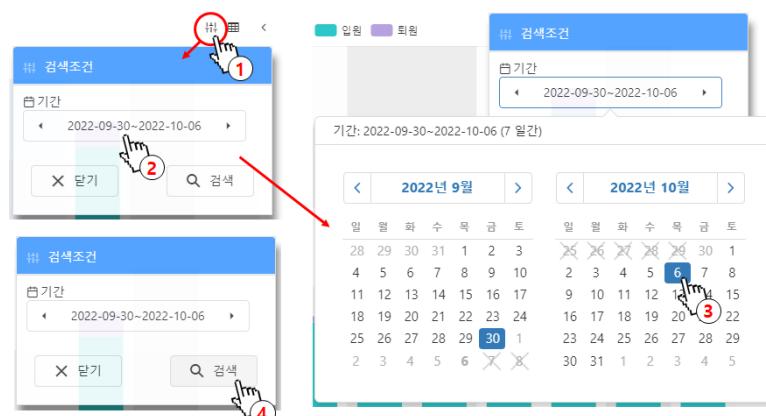
```

{
    key_calendar_w1: "YYYYMMDD 형식 날짜 데이터", // "srchCond.key 속성값" + "1" 조합이 from 데이터의 Key
    key_calendar_w2: "YYYYMMDD 형식 날짜 데이터" // "srchCond.key 속성값" + "2" 조합이 from 데이터의 Key
}

```

- filter_ymd2

- filter_ymd2 속성 및 UI 화면



filter_ymd2 설정에 따른 필터UI

```

widgProp = {
    srchCond: [
        {
            type: 'filter_ymd2',
            key: 'key_calendar_ymd2',
            default: '-6,0', // 기간을 설정해야 하므로 , 를 구분자로 하여 설정
            label: '기간',
            ...
        },
        ...
    ],
    ...
}

```

속성명	기능설명	설정값	설정예시
default	지정 기간을 기본 날짜 데이터로 설정	<code>String</code> , 를 구분자로 설정된 default 값을 각각 오늘날짜 + default 설정값으로 계산한 값 (<code>Default</code> 0,0)	default 가 "-6,0"이고 오늘 날짜가 2022년 10월 06일이면, "20220930"과 "20221006"을 기본값으로 설정

- filter_ymd2 필터 결과값

```
{
    key_calendar_ymd21: "YYYYMMDD 형식 날짜 데이터", // "srchCond.key 속성값" + "1" 조합이 from 데이터의
    key_calendar_ymd22: "YYYYMMDD 형식 날짜 데이터" // "srchCond.key 속성값" + "2" 조합이 to 데이터의 K
}
```

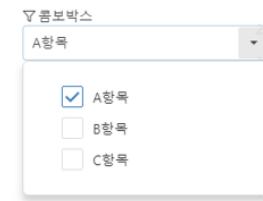
▼ 콤보 필터

- filter.cbo, filter.cbo2

- filter.cbo, filter.cbo2 속성 및 UI 화면



filter.cbo(searchEnabled 속성 설정), filter.cbo2 설정에 따른 필터UI



filter.cbo 설정에 따른 필터UI

filter.cbo2 는 filter.cbo 에 `searchEnabled : true` 를 설정한 상태이다.
때문에 필터 기능 및 UI 가 동일하다.

```
/*1*/ // type: 'filter.cbo'
widgProp = {
  srchCond: [
    {
      type: 'filter.cbo',
      key: 'key.cbo',
      items: keyCboData,
      value: 'A, B',
      multiSelect: true,
      searchEnabled: true,
      change_callback({el, val}) { ... },
      direction: 'vertical',
      label: '콤보박스',
    }, ...
  ],
  ...
}
```

```
/*2*/ // type: 'filter.cbo2'
widgProp = {
  srchCond: [
    {
      type: 'filter.cbo2',
      key: 'key.cbo.2',
      items: keyCboData,
      value: 'B',
      multiSelect: true,
      change_callback({el, val}) { ... },
      direction: 'vertical',
      label: '콤보박스',
      label: '콤보박스',
    }, ...
  ],
  ...
}
```

```
const keyCboData = [
  { value: 'A', label: 'A항목' },
  { value: 'B', label: 'B항목' },
  { value: 'C', label: 'C항목' }
];
```

```

    { value: 'B', label: 'B항목' },
    { value: 'C', label: 'C항목' }
];

```

속성명	기능 설명	설정값
items	콤보 아이템으로 나타낼 데이터를 설정 (label: 리스트 아이템에 드러낼 값, value: 리스트 아이템 값)	[JSON Array]
multiSelect	콤보 아이템을 2개 이상 선택할 수 있는 기능 설정. getDefaultValue로 필터 결과값 반환 시, true 설정은 여러 개 데이터를 ';' 구분자로 조합한 문자형 데이터	[Boolean] true 설정 시 아이템을 2개 이상 선택 (Default false)
searchEnabled	콤보 아이템을 검색할 수 있는 UI 및 기능 설정	[Boolean] true 설정 시 아이템을 검색할 수 있는 검색창UI 보임 (Default false)
change_callback	콤보 선택 값이 변경되는 시점에 발생하는 이벤트 설정	[Function] Param1: selectBox 컨테이너 요소, Param2: selectBox 값
direction	label과 selectBox 위치를 수평 방향으로 변경	('vertical' 'horizontal') 'horizontal' 설정 시 label과 selectBox를 수평으로 나란히 정렬

- filter.cbo, filter.cbo2 필터 결과값

```

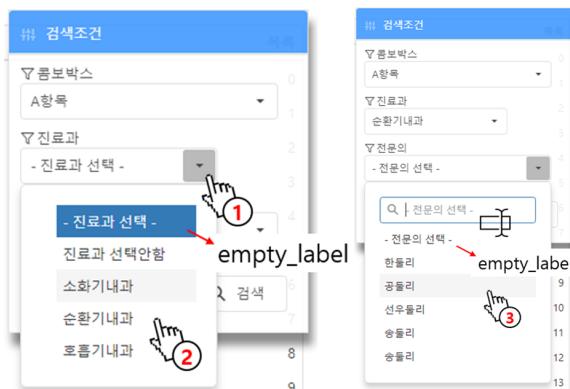
/*1 결과*/ // multiSelect: false,
{ key.cbo: "A" } // type: 'filter.cbo', value: 'A'
{ key.cbo_2: "A" } // type: 'filter.cbo', value: 'A'

/*2 결과*/ // multiSelect: true
{ key.cbo: "A,B" } // type: 'filter.cbo', value: 'A,B'
{ key.cbo_2: "A,B" } // type: 'filter.cbo', value: 'A,B'

```

- 콤보 필터 Master-Detail 예시

- 콤보 필터 Master-Detail은 2개 이상의 콤보가 데이터에 따라 상-하 관계로 연결된 것이다.
- 상위 콤보의 선택 값을 하위 콤보 조회 데이터의 조건 파라미터로 전달한다.
- 하위 콤보는 상위 콤보 값을 조건으로 데이터가 조회된 후 바인드 된다.



Master-Detail 관계의 두 콤보필터

```

widgProp = {
    srchCond: [
        /* Master 역할 필터 */
        {
            type: 'filter_cbo',
            key: 'key.cbo',
            datasource: {
                init: true,
                submit_id: 'DRZNBNB20204',
                qstmt_id: 'dev.WqCCzeqSrW',
                query_flag: 'master_list',
                // 쿼리 피라미터
                add_dept: '진료과 선택안함',
                add_code: '00'
            },
            label: '진료과',
            empty_label: '- 진료과 선택 -',
        },
        /* Detail 역할 필터 */
        {
            type: 'filter_cbo2',
            key: 'key.cbo2',
            datasource: {
                init: false, // false 설정
                submit_id: 'DRZNBNB20204',
                qstmt_id: 'dev.WqCCzeqSrW',
                query_flag: 'detail_list',
                // Master 데이터 중 전달할 Vo
                param_nm: 'value'
            },
            label: '전문의',
            empty_label: '- 전문의 선택 -',
        },
    ] ...
}

```

```

<isEqual property="query_flag" compare="master_list"><![CDATA[
SELECT
    #add_dept# as label,
    #add_code# as value
FROM DUAL UNION ALL
SELECT
    '소화기내과', 'A1'
FROM DUAL UNION ALL
SELECT
    '순환기내과', 'A2'
FROM DUAL UNION ALL
SELECT
    '호흡기내과', 'A3'
FROM DUAL
]]></isEqual>

<isEqual property="query_flag" compare="detail_list"><![CDATA[
WITH TEMP AS (
    SELECT
        '해당없음' as label,
        '00' as value_master,
        '00000000' as value
    FROM DUAL UNION ALL
    SELECT
        '이돌리', 'A2', '126578953'
    FROM DUAL UNION ALL
    SELECT
        '최돌리', 'A3', '126578954'
    FROM DUAL UNION ALL
    SELECT
        '한돌리', 'A1', '126578955'
    FROM DUAL UNION ALL
    SELECT
        '구돌리', 'A2', '126578956'
    FROM DUAL UNION ALL
    SELECT
        '곽돌리', 'A3', '126578957'
    FROM DUAL UNION ALL
    SELECT
        '공돌리', 'A1', '126578958'
    FROM DUAL UNION ALL
    SELECT
        '기돌리', 'A3', '126578959'
    FROM DUAL UNION ALL
)

```

```

SELECT
    '장돌리', 'A2', '126578950'
FROM DUAL UNION ALL
SELECT
    '선우들리', 'A1', '212657895'
FROM DUAL UNION ALL
SELECT
    '송돌리', 'A1', '312657895'
FROM DUAL UNION ALL
SELECT
    '강돌리', 'A3', '412657895'
FROM DUAL UNION ALL
SELECT
    '사돌리', 'A2', '512657895'
FROM DUAL UNION ALL
SELECT
    '박돌리', 'A2', '612657895'
FROM DUAL UNION ALL
SELECT
    '송돌리', 'A1', '712657895'
FROM DUAL
)
SELECT * FROM TEMP
WHERE value_master=#value#
]]>/isEqual>

```

▼ 라디오 필터

- filter_rdo

- filter_rdo 속성 및 UI 화면



filter_rdo 설정에 따른 필터UI

```

widgProp = {
    srchCond: [
        {
            type: "filter_rdo",
            key: "key_rdo",
            default: 'A',
            items: [
                {
                    value: 'A',
                    label: 'A항목'
                },
                {
                    value: 'B',
                    label: 'B항목'
                },
                {
                    value: 'C',
                    label: 'C항목'
                }
            ],
            label: "라디오버튼입니다."
        },
        ...
    ],
    ...
}

```

속성명	기능 설명	설정값
items	라디오 아이템으로 나타낼 데이터를 설정 (label: 라디오UI 에 드러낼 값, value: 라디오 아이템 값)	[JSON Array]

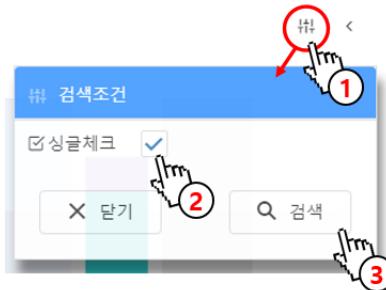
- filter_rdo 필터 결과값

```
{ key_calendar_ym : "YYYYMM 형식 년, 월 데이터" } // "202109"
```

▼ 체크박스 필터

- filter_chk

- filter_chk 속성 및 UI 화면 (싱글 체크)



filter_chk 설정에 따른 필터UI

```
widgProp = {
  srchCond: [
    {
      type: "filter_chk",
      key: "key_chk2",
      default: true,
      label: "멀티체크"
    },
    ...
  ],
  ...
}
```

- filter_chk 속성 및 UI 화면 (멀티 체크)



filter_chk 설정에 따른 필터UI

```
widgProp = {
  srchCond: [
    {
      type: "filter_chk",
      key: "key_chk",
      default: 'A',
      width: 255,
      items: [
        { value: 'A', label: 'A항목' },
        { value: 'B', label: 'B항목' },
        { value: 'C', label: 'C항목' }
      ]
    }
  ]
}
```

```

        label: 'C항목'
    }],
    label: "멀티체크",
    ],
    ...
],
...

```

◦ 멀티체크 기능관련 속성

속성명	기능 설명	설정값
width	filter_chk 속성 items 있는 경우 콤보의 너비를 변경 (#'srchCond.key 속성값' + '_dropdown' 요소의 width 속성으로 적용)	[Number] px 단위 숫자 (px 생략, Default : 200) [String] % 단위 붙은 너비
items	2개 이상의 값을 멀티체크 하는 경우에 대상 데이터를 설정 (label: 콤보UI에 나타낼 값, value: 콤보 아이템 값)	[JSON Array]

▼ 라벨 필터

- filter_lbl



filter_lbl 설정에 따른 필터UI

```

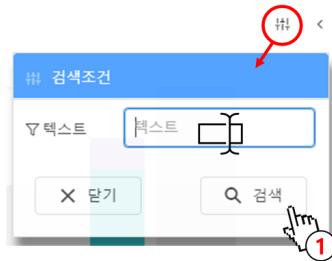
widgProp = {
  srchCond: [
    {
      type: "filter_lbl",
      key: "key_lbl",
      style: 'color:green;margin:1rem;font-weight:bold;font-size:25px;',
      label: "'확인' 버튼을 눌러 필터를 적용하세요."
      ...
    },
    ...
]
}

```

속성명	기능 설명	설정값
style	filter_lbl 태입 라벨에 적용할 스타일, label.filter_label 요소의 style 속성으로 적용 됨	[String] style 속성key-value

▼ 텍스트 필터

- filter_txt



filter_txt 설정에 따른 필터UI

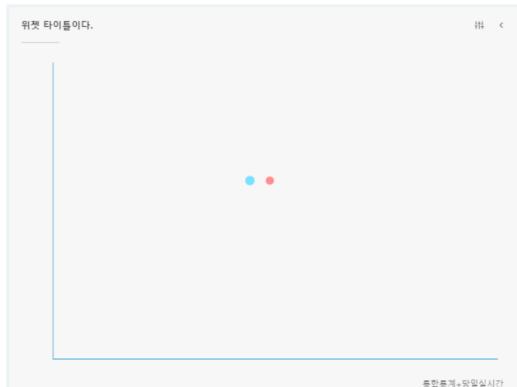
```
widgProp = {
  srchCond: [
    {
      type: "filter_txt",
      key: "key_txt",
      placeholder: "텍스트",
      width: 180,
      direction: 'horizontal',
      label: "텍스트"
    ...
  }, ...
}
```

속성명	기능 설명	설정값
placeholder	placeholder 기능 적용	[String]
width	textbox 영역 너비 변경	[Number] px 단위 숫자 (px 생략, Default : 200) [String] % 단위 붙은 너비
direction	label과 textbox 위치를 수평 방향으로 변경	['horizontal' 'vertical'] Default 'vertical'

▼ visibleLoadingBar

| 위젯 초기 로드 시, 로딩바UI를 적용/해제한다.

로딩바UI가 위젯 보다 앞에 위치 (z-index 상위값) 하므로, 로딩 중의 사용자 조작으로 인한 에러를 방지할 수 있다.

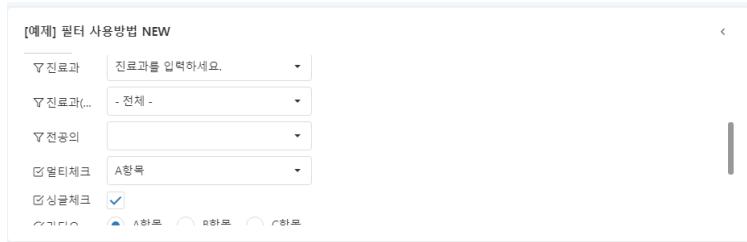


visibleLoadingBar 속성 적용화면

```
widgProp = {
  visibleLoadingBar: false, // default: true
} ...
```

▼ enabledSlimScroll

위젯 컨텐츠 영역보다 위젯 뷰가 더 큰 경우에 스크롤을 생성한다.
스크롤이 생기지 않는 동작이 기본값(false)이므로, 스크롤을 생성하기 위해서는 true를 설정한다.



enabledSlimScroll 속성 적용화면

```
widgProp = {  
    enabledSlimScroll: true, // default: false  
} ...
```

▼ 2) Event

위젯 추가/삭제, 사이즈 변경, 테마 변경 동작 시 발생한다.

```
widgProp = {  
    onBeforeLoad() { ... },  
    onLoaded() { ... },  
    onBeforeLoadData({viewObj, srchData, contData}) { ... },  
    onBeforeLoadDataSource({viewName, viewObj, srchData, contData}) { ... },  
    onAfterLoadData({viewObj, srchData, contData}) { ... },  
    onResize() { ... },  
    onChangeTheme() { ... },  
    destory() { ... },  
}..
```

속성명	기능 설명	파라미터	파라미터 정보
onBeforeLoad	위젯 컨텐츠(div.phis_widg_cont) 로드 이전 시점에 설정할수 실행	-	-
onLoaded	위젯 컨텐츠(div.phis_widg_cont) 로드 직후 시점에 설정할수 실행	-	-
onBeforeLoadData	위젯 컨텐츠(div.phis_widg_cont) 로드 후, 위젯 데이터 조회 직전 시 점에 설정할수 실행 ※ widgProp.viewOption.name='prog' 설정 시에는 이벤트 발생하지 않음	[Object]	{ viewObj : 컨텐츠에 생 성된 인스턴스, srchData: 데 이터 조회정보, contDat: 조 회된 데이터 }
		viewObj	- "chart": Echarts인스턴스 - "grid" "treelist": DevExpress인스턴스
		srchData	데이터 조회정보 (qstmt_id, data_src_type)
		contData	srchData로 조회된 데이터
onBeforeLoadDataSource	위젯 컨텐츠(div.phis_widg_cont) 위젯 데이터 조회 이후, 데이터 바 인드 직전 시점에 설정할수 실행 (※ widgProp.viewOption.name='prog' 설정 시에는 이벤트 발생하지 않음)	[Object]	{ viewName: "chart" "grid" "treelist", viewObj : 컨텐 츠에 생성된 인스턴스, srchData: 데이터 조회정 보, contDat: 조회된 데이터 }

속성명	기능 설명	파라미터	파라미터 정보
		viewName	widgPropviewOption.name 설정 값 ("chart" "grid" "treelist")
		viewObj	- "chart": Echarts 인스턴스 - "grid" "treelist": DevExpress 인스턴스
		srchData	데이터 조회정보 (qstmt_id, data_src_type)
		contData	srchData 로 조회된 데이터
onAfterLoadData	위젯 컨텐츠(div.phis_widg_cont) 위젯 데이터 바인드 및 drillThru 데 이터 전달 이후 시점에 설정함수 실행 (※ widgProp.viewOption.name='prog' 설정 시에는 이벤트 발생하지 않음)	[Object]	{ viewObj : 컨텐츠에 생성 된 인스턴스, srchData: 데 이터 조회정보, contData: 조 회된 데이터 }
		viewObj	- "chart": Echarts 인스턴스 - "grid" "treelist": DevExpress 인스턴스
		srchData	데이터 조회정보 (qstmt_id, data_src_type)
		contData	srchData 로 조회된 데이터
onResize	페이지 수정모드 실행 중, 위젯 컨 테이너의 크리를 변경 시 설정함수 실행	-	-
onChangeTheme	보드 우측 상단 톰니마퀴 아이콘 (i.icon-settings.setting) 클릭으로 열리는 메뉴 중, 테마변경 스위치 로 테마 변경 시 설정함수 실행	-	-
destroy	위젯 좌측 상단의 쓰레기통 아이콘 ([phis-id="div_removeWidget"])을 눌러 위젯 삭제 시 설정함수 실행	-	-



이벤트 발생 시점 및 순서

• 위젯 추가

- 시점 : 페이지 상에 위젯 첫 로드
- 순서 (1)
 - `widgProp.viewOption.name='chart' 혹은 'grid' 혹은 'treelist'` 설정의 경우
 - `onBeforeLoad → onLoaded → onBeforeLoadData → onBeforeLoadDataSource`
→ `onAfterLoadData`
- 순서 (2)
 - `widgProp.viewOption.name='prog'` 설정의 경우
 - `onBeforeLoad → onLoaded`

```
▼ === 위젯 신규 추가 ===  
↳ undefined  
[콘솔] onBeforeLoad undefined  
[콘솔] onLoaded undefined  
[콘솔] onBeforeLoadData ► {viewObj: ECharts, srchData: {}, contData: null}  
  
[콘솔] onBeforeLoadDataSource chart ► ECharts {_eventProcessor: ECEventProcessor  
► (11) [{}], {}, {}, {}, {}, {}, {}, {}, {}, {}, {}, {}]  
[콘솔] onAfterLoadData ► {viewObj: ECharts, srchData: {}, contData: Array(11)}
```

• 위젯 뷰 타입 변경

- 시점 : toolBox: [view] 설정 및 'chart'/grid' 토글버튼 클릭하여 위젯 컨텐츠 변경
- 순서 : `onLoaded → onBeforeLoadData → onResize`

```
▼ === 위젯 chart/grid 툴 ===  
↳ undefined  
[콘솔] onLoaded undefined  
[콘솔] onBeforeLoadDataSource grid ► t {_events: {}, _eventsStrategy: t, callBase: u  
► (11) [{}], {}, {}, {}, {}, {}, {}, {}, {}, {}, {}, {}]  
[콘솔] onResize undefined
```

• 위젯 삭제

- 시점 : 페이지 수정모드 실행 후 사용자가 위젯의 쓰레기통 아이콘([phish-id="div_removeWidget"])을 눌러 위젯을 삭제
- 순서 : `destroy`

```
▼ === 보드 삭제 ===  
↳ undefined  
[콘솔] destroy undefined
```

• 위젯 컨테이너 크기 변경

- 시점 : 페이지 수정모드 실행 후 사용자가 위젯 컨테이너 크기를 변경
- 순서 : `onResize`

```
▼ === 위젯 resize ===  
↳ undefined  
[콘솔] onResize undefined  
  
> console.groupEnd('==> 위젯 resize ==>')
```

- 위젯 테마 변경

- 시점 : 보드 상단 를바로 테마변경 실행 시 위젯 테마 변경
- 순서 : **onChangeTheme**

```
▼ === 보드 Theme 변경 ===
<- undefined
② [콘솔] onChangeTheme undefined
```

▼ 3) Method

위젯 단위로 생성된 **PhisWidget** 인스턴스 객체의 내부함수 를 위젯 메소드라 한다.

위젯 메소드로 위젯 정보를 확인하거나, 위젯 동작을 명령할 수 있다.

```
const phisWidget = this; // PhisWidget 인스턴스 객체
phisWidget.getWidgetEl();
...
```

- 위젯 동작을 실행하는 경우 (반환값 없음)

메소드명	기능 설명	실행 동작
dispose	위젯 컨텐츠 인스턴스 제거 (viewOption.name 이 "chart", "treelist", "grid" 설정인 경우만 동작)	"chart": Echarts인스턴스.dispose() 실행 "grid" "treelist": DevExpress인스턴스.dispose() 실행
popupFilter	위젯 필터 팝업창을 엽	

- 위젯 정보를 확인하는 경우 - 파라미터 설정 없음

메소드명	기능 설명	반환값
getWidgetEl	위젯 컨테이너 요소를 확인	[JQuery<Element>] div.phis-widget 요소
getContentEl	위젯 컨텐츠 요소를 확인 (속성 스크립트 로드완료 시점 이후에 동작)	[JQuery<Element>] div.phis_widg_cont 요소
getViewObj	위젯 컨텐츠 인스턴스 객체 확인 (widgPropviewOption.name='chart' 혹은 'grid' 혹은 'treelist' 설정인 경우만 동작)	- "chart": Echarts 인스턴스 - "grid": DevExpress 인스턴스 - "treelist": DevExpress 인스턴스

- 위젯 정보를 확인하는 경우 - 파라미터 설정 필요



데이터 타입 중 'Any' 는 모든 형태의 데이터 지정이 가능함을 의미한다.

메소드명	기능 설명	설정 파라미터	반환값 및 동작
getWidgetInfo	위젯 정보(설정VO) 를 반환	(Param1) [String] 위젯 데이터 VO 변수명 (= ZZ_ZNBMWIDGET 테이블 Vo명, ex. "upage_id", "widget_id", "widget_desc" ...)	설정한 VO 의 값 반환
setWidgetStyle	위젯 타이틀 동적 변경		

메소드명	기능 설명	설정 파라미터	반환값 및 동작
		(Param1) [String] "title"	
		(Param2) [String] 첫번째 파라미터가 "title" 인 경우, 위젯 타이틀로 설정할 문자	"위젯요소.card-header h5" 타겟에 설정한 문구로 text 변경
find	위젯요소 () 를 타겟으로 선택자 요소를 find		
		(Param1) [String] phis-id 요소 속성값 (ex. @찾으려는 phis-id 속성값)	find 결과 요소 (JQuery<Element>)
loadingBar	위젯에 로딩UI 보임/숨김		
		(Param1-1) [Boolean] true: UI 보임 / false: UI 숨김 Default	true 시 UI 보임, false 시 UI 숨김
		(Param1-2) [Number] UI 보일 시간 설정 (단위: milli sec)	설정 시간 만큼 보임 후 숨김
		(Param2) [JQuery<Element>] 위젯el.widget-card Default UI 설정 컨테이너	
notify	알림 창 열/닫음		
		(Param1) [String] 1. 알림 창 종류: "danger" "warning" "info" Default 2. 알림 창 닫힘 설정: "closeAll"	설정한 타입으로 알림창 열림/닫힘
		(Param2) [String] 설정 문구	설정한 문구로 알림 창이 열림
		(Param3) [String] DevExpress.ui.notify 설정 값	설정 값으로 알림 창이 열림
message	"info" 알림 창 열 = notify("info", "파라미터") 동작과 동일		
		(Param1) [String] "info" 알림 창 설정 문자	설정한 문구로 "info" 알림 창이 열림
popupWidget	위젯을 팝업창으로 엽		
		(Param1) [String] 팝업창으로 열 위젯ID (ZZ.ZNBMWIDGET 테이블 WIDGET_ID 의 컬럼값)	설정한 위젯이 팝업창으로 열림 Promise<T>
		(Param2) [Object] dxPopup 옵션 및 위젯 옵션	설정한 위젯이 dxPopup 옵션에 따라 팝업창으로 열림 Promise<T>
popupWidget2	위젯을 팝업창으로 엽 (2)		
		(Param1) [String] 팝업창으로 열 위젯ID (ZZ.ZNBMWIDGET 테이블 WIDGET_ID 의 컬럼값)	
		(Param2) [Object] dxPopup 옵션 및 popupWidget2 옵션	
		(Param3) [Function] 콜백 함수	
popoverWidget	위젯을 팝오버로 엽		
		(Param1) [String] 팝오버창으로 열 위젯ID (ZZ.ZNBMWIDGET 테이블 WIDGET_ID 의 컬럼값)	
		(Param2) [Object] dxPopover 옵션 및 popoverWidget 옵션	
accAuthorized	환자정보 접근 시 사유입력 / 열람요청 팝업창을 엽	accAuthorized 내용 하다 확인	

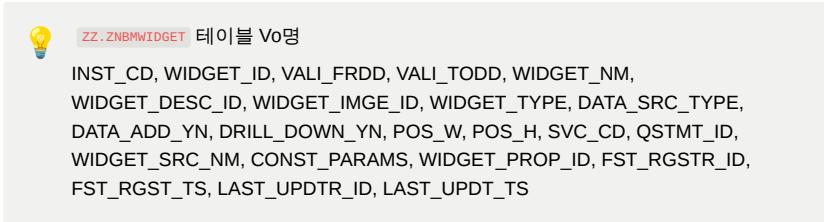
메소드명	기능 설명	설정 파라미터	반환값 및 동작
localStorage	위젯 단위 localStorage 값 관리		
		(Param1) [String] localStorage 메소드 첫번째 파라미터로 설정할 key (최종 key의 형태 : '위젯_Id + 설정한 key + 사용자Id')	getItem 실행 후 반환된 데이터
		(Param2) [String] localStorage 메소드 두번째 파라미터로 설정할 value (1. null / 2. null 이외 값)	1. removeItem 실행 / 2. setItem 실행
getDefaultValue	위젯속성 srchCond 로 설정 된 필터 데이터 확인		
		(Param1) [JSON Array] 위젯속성 srchCond 로 설정한 필터 타입 ([{ type: 'filter_wk' }, { type : 'filter.cbo' }] ...)	필터에 설정된 데이터 Object
addToolBtn	toolBox 에 사용자 지정 버튼 추가		
		(Param1) [Object] { class : 버튼 icon class 속성값 title : 버튼 title 속성값 handleClick : 버튼 클릭 이벤트 }	phis-id="phis-tool-box-1" 에 설정 값으로 생성한 i 태그 append
sendDrillThru	데이터 전달 및 타겟 위젯 리로드	sendDrillThru 내용 하단 확인	
info	위젯 및 보드 정보 확인	info 내용 하단 확인	

▼ getWidgetInfo(Param1)

| 위젯에 적용된 데이터 (`ZZ.ZNBMWIDGET`) 를 확인한다.

- 설정 파라미터

- (Param1) [String] 파라미터로 설정할 수 있는 Key (= `ZZ.ZNBMWIDGET` 테이블 Vo명)



▼ setWidgetStyle(Param1, Param2)

| 위젯 타이틀을 동적으로 변경한다.

- 설정 파라미터

- (Param1) 'title' (고정값, 이외의 값은 현재 미지원)
- (Param2) [String] 위젯 타이틀 요소 (div.card-header h5) 에 적용될 text

▼ find(Param1)

위젯 컨테이너 하위 요소 중 '@phis-id 속성값' (= [phis-id="속성값"])에 해당하는 요소를 JQuery find 메소드로 찾고, 결과를 리턴한다.

- 설정 파라미터

- (Param1) **[String]** @ + 'phis-id 속성값' 형태 선택자 (= \$('위젯의 div.phis-widget').find([phis-id="phis-id 속성값"]));

▼ loadingBar(Param1, Param2)

위젯에 로딩UI를 적용한다.

- 설정 파라미터

- (Param1)

- [Boolean]** 위젯에 로딩UI를 보일 때 true를 설정하고, 숨길 때 false를 설정하여 실행

- ex) 조회 전 - true, 조회 후 - false로 로딩UI 보임/숨김

- [Number]** 위젯에 로딩UI가 보일 시간 설정 (단위: milli sec)

- (Param2) **[JQuery<Element>]** 필수값 아님. div.widget-card 하위 요소 중 UI를 보일 특정 요소가 있는 경우에 설정

▼ notify(Param1, Param2, Param3)

위젯에 phisToast 토스트 알림창UI(DevExpress.ui.notify)를 적용한다.

- 설정 파라미터

- (Param1), (Param2)

- [String]** phisToast 메소드와 동일 (각각 알림창 종류, 알림창 문구)

- (Param3) **[Object]** DevExpress.ui.notify 옵션 중 position.of 기본값은 \$('위젯의 div.phis-widget')

▼ message(Param1)

위젯에 notify('info', '설정문구')를 적용한 동작과 같다.



phisWidget.message('문구');

- 설정 파라미터

- (Param1) **[String]** 토스트 알림창으로 보일 알림 문구

▼ popupWidget2(Param1, Param2, Param3)

위젯을 팝업UI(dxPopup) 컨텐츠 위젯으로 연다.



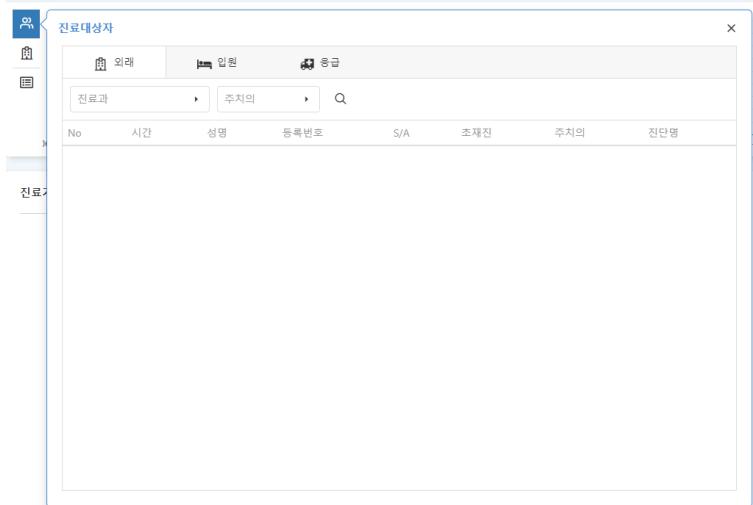
phsWidget.popupWidget2('위젯ID');

- 설정 파라미터

- (Param1) `[String]` 팝업창으로 열 위젯ID (ZZ.ZNBMWIDGET 테이블 WIDGET_ID 의 컬럼값)
- (Param2) `[Object]` dxPopover 옵션 및 popupWidget2 옵션
- (Param3) `[Function]` 콜백 함수

▼ popoverWidget(Param1, Param2, Param3)

| 위젯을 팝오버UI (dxPopover) 컨텐츠 위젯으로 연다.



- 설정 파라미터

- (Param1) `[String]` 팝오버창으로 열 위젯ID (ZZ.ZNBMWIDGET 테이블 WIDGET_ID 의 컬럼값)
- (Param2) `[Object]` dxPopover 옵션 및 popoverWidget 옵션

▼ accAuthorized(Param1)

| 환자정보 접근 시 사유입력/열람요청 기능을 동작한다.

| 메소드가 Promise 객체로 실행되므로 메소드 완료 시점 (= then) 시점에 파라미터로 인증 여부(Boolean) 를 반환한다.

⚠️ 접근사유입력, 접근권한요청 위젯이 팝업하여 동작하므로 관련 위젯이 생성되어 있는 상태여야 한다.

- 접근사유입력
 - wACMREASON (고정 아이디)

The screenshot shows a '접근사유입력' (Access Reason Input) window. At the top, there's a warning message: '의료정보를 이용함에 있어 멀시된 목적외에는 이용할 수 없으며, 법에 정한 경우를 제외하고 환자의 동의 없이 개인정보를 사용하거나 제3자에게 제공할 수 없습니다.' (When using medical information, it cannot be used for purposes other than those specified, except as explicitly allowed by law, without the patient's consent or providing it to a third party.) Below the message, there's a note: '△ 접근 사유가 의무기록의 추가 작성, 수정인 경우 관련부서로 문의 후 의무기록정정 절차에 따라 진행 하시기 바랍니다.' (When adding or modifying a medical record, please inquire with the relevant department and follow the procedures for correcting the medical record.) The main form has fields for '사유' (Reason), '사용자HD' (User HD), 'NUSCARD' (NUSCARD), and '비밀번호' (Password). Buttons for '확인' (Confirm) and '취소' (Cancel) are at the bottom.

- 접근권한요청
 - wACAUTHREQ (고정 아이디)

The screenshot shows a '접근권한요청' (Access Permission Request) window. It contains fields for '메시지' (Message), '등록번호' (Registration Number), '요청상태' (Request Status), '요청일' (Request Date), '요청자' (Requester), '연락처' (Contact), '사유' (Reason), '기타사유' (Other Reasons), and '요청기간' (Request Period). Buttons for '확인' (Confirm) and '취소' (Cancel) are at the bottom.

- 설정 파라미터
 - (Param1) **[Object]** 접근 요청 시 필요한 환자 정보
 - accdataflag : 환자 정보를 접근하는 위젯ID
 - pid, orddd, cretno, ordtype 정보

▼ **sendDrillThru(Param1, Param2)**

| 위젯 데이터를 전달하여 타겟 위젯에 데이터를 바인드 후 로드한다.

- 설정 파라미터

- (Param1) [Object] { targetWidget: 데이터를 전달 받을 대상 위젯Id 목록 Array }
 - 파라미터 undefined로 지정 시, 보드 내 전체 위젯에 Param2의 값으로 재조회 (enabledInitDataLoad: false 적용된 위젯)
- (Param2) Param1에 설정한 대상 위젯 로드 시점에 전달할 데이터

- 예제

```
const phisWidget = this;
phisWidget.sendDrillThru(
{
    targetWidget: ['데이터 전달대상 widget_id1', '데이터 전달대상 widget_id2', ...]
},
/*
    Param1에 설정한 대상위젯에 전달할 데이터로, String Number 등등 데이터형 관계없이 적용가능
*/
{
    exampleKey: '전달이 데이터1'
});
/* 
const reqData = $.extend(param2, {
    'qstmt_id': 위젯 'qstmt_id',
    'data_src_type': 위젯 'data_src_type'
}); // 기본 request 정보와 사용자 전달데이터 merge

(1) viewObjNm = 'prog' 인 경우
widgProp = {
    ... 생략 ...
    loadProgData(reqData) { ... }
};

(2) viewObjNm ≠ 'prog' 인 경우
widgProp = {
    ... 생략 ...
    onBefore.loadData({ srchData }) { ... }
};
gfn_submit('위젯 svc_cd', 'req', reqData, submitCallback함수, null, false);
*/
```



sendDrillThru로 '대상위젯'이 데이터를 전달 받는 시점 및 동작

- (Param1)에서 설정한 대상 위젯이 viewObjNm = 'prog'인 경우
 - loadProgData 함수의 1번째 파라미터로 전달
- (Param1)에서 설정한 대상 위젯이 viewObjNm ≠ 'prog'인 경우
 - 1. onBefore.loadData 함수의 { srchData } 파라미터로 전달
 - 2. srchData 파라미터 (참조객체)와 (Param2)에 설정된 파라미터를 병합
 - 병합된 참조객체로 내부에서 재조회 발생하여, 데이터 바인드 및 로드

▼ info(Param1)

| 소스 내부에서 차트, 그리드 위젯에 적용하는 Echart, dxDataGrid 기본값을 확인한다.

- 설정 파라미터

- (Param1) ['defaultEChartOption' | 'defaultDxGridOption'] 각각 EChart, dxDataGrid 기본적용 속성 값

- 예시

```

const phisWidget = this;
const gridOption = Object.assign(
    phisWidget.info('defaultDxGridOption'),
    {
        columnAutoWidth: false,
        repaintChangesOnly: true,
        dataSource: [], ...
    }
);
elTarget.dxDataGrid(gridOption);

```

▼ 3. 공통 함수

PhisBoard 플랫폼에서 유ти리티 성격으로 제공하는 API를 공통 함수라 한다.

공통 함수는 phiscommon.js, syscommon.js, znbhelper.js 등 PhisBoard 라이브러리 내부 스크립트로 제공된다.

- 유형별 함수 목록

▼ 서버

메소드명	기능
gfn_submit <small>중요</small>	서버로 데이터를 요청하고 결과를 리턴
gfn_getUserinfos	세션 스토리지에 저장된 사용자 정보 중 입력된 파라미터를 Key로 하는 데이터가 있는 경우 결과로 리턴

▼ gfn_submit 중요

- 설정 파라미터

Param1	서비스 코드 <small>SVC_CD</small> <small>필수</small>	<small>ZZ.ZNBWIDGET</small> 테이블 <small>SVC_CD</small> Vo 값
Param2	서버 침조명 <small>REQ_NM</small> <small>필수</small>	데이터 요청 시 서버에서 침조하는 이름 (<small>Default</small> 'req')
Param3	요청 데이터 <small>필수</small>	데이터 요청 시 필요한 request 객체
Param4	요청 후 콜백 함수 <small>필수</small>	서버 요청 결과를 받을 콜백함수
Param5	시스템 코드	타기관 서비스 호출 일 경우 시스템 코드를 입력 (<small>Default</small> null)
Param6	시스템 메세지 팝업 사용 여부	true 설정 시 시스템 메세지 팝업 사용, false 시 시스템 메세지 팝업 사용하지 않음 (<small>Default</small> false)
Param7	비동기 통신 여부	true 설정 시 비동기, false 설정 시 데이터 조회까지 await 후 callback 을 실행하는 동기로 통신 (<small>Default</small> true)

```

const callDRZNB10301 = (responseInfo, isSuccess, errMsg) => {
    if (isSuccess) {
        // 조회 성공, responseInfo 조회 결과 데이터 바인드
    } else {
        // 조회 실패, errMsg 에러 메시지 출력
    }
};

gfn_submit('DRZNB10301', 'req', reqData, callDRZNB10301, null, false);

```

▼ gfn_getUserinfos

Session Storage에 저장된 사용자 정보 중 입력된 파라미터를 Key로 하는데 이터가 있는 경우 결과로 리턴 한다.

- 설정 파라미터

- (Param1) 세션 스토리지에 저장된 사용자 정보 Key

▪ 주요 정보

KEY	설명	예시
userid	계정 아이디	NUBOARD
usernm	사용자 이름	서환자
dutinstcd	기관코드	012
dutinstnm	기관명	서울00병원
dutplceinstcd	실제 근무하고 있는 기관 코드	012
dutplcecd	실제 근무하고 있는 근무지 코드	1000000000
jobkindcd	직종코드	null
jobposcd	직위코드	null
systemcd	시스템 코드	HIS012

▪ 그 외 정보



1.3. 쿼리 - (2) 쿼리 작성 - 시스템 정보 변수 참고

▼ 날짜, 시간

메소드명	기능
gfn_GetDateFormat	Date 타입을 파라미터에 입력된 포맷을 적용한 String 타입으로 변환하여 리턴
gfn_ToDate	포맷이 적용된 String 타입 날짜 데이터를 Date 타입으로 변환하여 리턴
gfn_GetNewDate	PC 의 현재 날짜, 시간을 서버 시간으로 변경하여 리턴
gfn_GetCurrentDate	현재 날짜를 'YYYYMMDD' 포맷을 적용한 String 타입으로 변환하여 리턴
gfn_GetCurrentTime	현재 날짜를 'YYYYMMDD' 포맷을 적용한 String 타입으로 변환하여 리턴
gfn_GetAddDate	날짜 데이터는 년, 월, 일을 계산, 시간 데이터는 시, 분, 초를 계산하여 Date 타입으로 리턴
gfn_GetDateInterval	기준 날짜와 대상 날짜 사이의 기간 차(일수)를 계산하여 리턴

▼ gfn_GetDateFormat

• 설정 파라미터

- (Param1) [String] 날짜, 시간 포맷 적용

• 날짜 포맷

설정값	기능
YYYY	년
MM	월
MON	월 텍스트 (ex. January ...)
DD	일
DAY	일 텍스트 (ex. Sunday ...)

• 시간 포맷

설정값	기능
HH	시 (1~12)
hh	시 (0~24)
mm	분
ss	초
SS	밀리 세컨드
a	주간/야간 텍스트 (AM/PM)

• 반환 데이터

- [String] Param1로 설정한 포맷이 적용된 날짜, 시간 데이터

• 예시

```
/* gfn_GetNewDate() 결과값 */ // Fri Oct 07 2022 16:20:10 GMT+0900 (한국 표준시)
```

```
/*1*/ gfn_GetNewDate().gfn_GetDateFormat('YYYYMMDD');           /* 1 결과*/ // '20221007'  
/*2*/ gfn_GetNewDate().gfn_GetDateFormat('YYYYMMDD HH:mm');    /* 2 결과*/ // '20221007 04:20'  
/*3*/ gfn_GetNewDate().gfn_GetDateFormat('YYYYMMDD hh:mm');     /* 3 결과*/ // '20221007 16:20'  
/*4*/ gfn_GetNewDate().gfn_GetDateFormat('YYYYMMDD hh:mm a');   /* 4 결과*/ // '20221007 16:20 PM'
```

▼ gfn_ToDate

- 설정 파라미터
 - (Param1) **[String]** String 타입의 날짜 데이터에 적용된 포맷
- 반환 데이터
 - **[Date]** Param1로 설정한 포맷을 기준으로 Date 타입으로 변환된 날짜 데이터
- 예시

```
/*1*/ '20221007'.gfn_ToDate('YYYYMMDD'); /*1 결과*/ // Fri Oct 07 2022 00:00:00 GMT+0900 (한국 표준시)
```

▼ gfn_GetNewDate

- 설정 파라미터
 - 없음
- 반환 데이터
 - **[Date]** 현재 시간 날짜 데이터
- 예시

```
/*1*/ gfn_GetNewDate() /*1 결과*/ // Fri Oct 07 2022 16:20:10 GMT+0900 (한국 표준시)
```

▼ gfn_GetCurrentDate

- 설정 파라미터
 - 없음
- 반환 데이터
 - **[String]** 현재 시각 기준, 날짜 데이터 'YYYYMMDD' 포맷을 적용한 String 타입으로 변환한 데이터
- 예시

```
/* gfn_GetNewDate() 결과값 */ // Fri Oct 07 2022 16:20:10 GMT+0900 (한국 표준시)  
/*1*/ gfn_GetCurrentDate() /*1 결과*/ // '20221007'
```

▼ gfn_GetCurrentTime

- 설정 파라미터
 - 없음
- 반환 데이터
 - **[String]** 현재 시각 기준, 날짜 데이터 'HHMMSS' 포맷을 적용한 String 타입으로 변환한 데이터
- 예시

```
/* gfn_GetNewDate() 결과값 */ // Fri Oct 07 2022 16:20:10 GMT+0900 (한국 표준시)  
/*1*/ gfn_GetCurrentTime() /*1 결과*/ // '162010'
```

▼ gfn_GetAddDate

- 설정 파라미터

- (Param1) **[Number]** Param2 를 기준으로 계산할 기간 데이터
- (Param2) **[String]** 날짜, 시간 포맷 적용

- 날짜 포맷

설정값	기능
Y	년 +/-
M	월 +/-
D	일 +/-

- 시간 포맷

설정값	기능
h	시 +/-
m	분 +/-
s	초 +/-

- 반환 데이터

- [Date]** 계산된 날짜 데이터

- 예시

```
/* gfn_GetNewDate() 결과값 */ // Fri Oct 07 2022 16:20:10 GMT+0900 (한국 표준시)
/*1*/ gfn_GetNewDate().gfn_GetAddDate(-365, "D") /*1 결과*/ // Thu Oct 07 2021 16:19:19 GMT+0900 (한국
/*2*/ gfn_GetNewDate().gfn_GetAddDate(5, "h")      /*2 결과*/ // Fri Oct 07 2022 21:19:40 GMT+0900 (한국
```

▼ gfn_GetDateInterval

- 설정 파라미터

- (Param1) 기준 날짜
- (Param2) 계산 대상 날짜

- 반환 데이터

- [Number]** 계산 대상 날짜 (Param2) - 기준 날짜(Param1) 의 일수 차이

- 예시

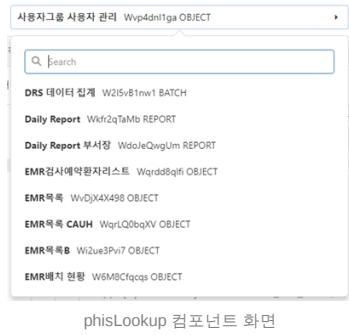
```
/*1*/ gfn_GetDateInterval('2022-10-05', '2022-10-01') /*1 결과*/ // -4
/*2*/ gfn_GetDateInterval('2022-10-01', '2022-10-05') /*2 결과*/ // 4
/*3*/ gfn_GetDateInterval('20221001', '20221001')      /*3 결과*/ // 0
```

▼ 컴포넌트 (DevExtreme, Ace Editor)

메소드명	기능
phisLookup	데이터를 리스트 아이템UI로 구현
phisToast	토스트 알림창UI 구현
phisPopup	팝업창UI 구현
phisModal	모달 (배경을 선택할 수 없는 형태) 팝업창UI 구현
phisInitAceEditor	ace-editor 로 SQL, Javascript 등 에디터 구현

▼ phisLookup

| 데이터를 리스트 아이템UI로 구현



phisLookup 컴포넌트 화면

- 설정 파라미터

- (Param1) `[jQuery<Element>]` 필수 dxLookup 컴포넌트로 변환할 대상 요소
- (Param2) `[Object]` 리스트 아이템UI 관련 세부 옵션 (dxLookup 옵션)

- 기본 옵션

```
{
    displayExpr: 'text', // 기본값
    valueExpr: 'id', // 기본값
    noDataText: '- 검색결과 없음 -',
    closeOnOutsideClick: true,
    dropDownOptions: {
        closeOnOutsideClick: true,
    },
    showPopupTitle: false,
    showCancelButton: false,
    dropDownCentered: true,
    searchTimeout: 300,
    popupHeight() {
        return window.innerHeight * 0.4;
    }
}
```

- dxLookup 옵션

https://js.devexpress.com/Documentation/ApiReference/UI_Components/dxLookup/

Lookup

The Lookup is a UI component that allows an end user to search for an item in a collection shown in a drop-down menu.

https://js.devexpress.com/Documentation/ApiReference/UI_Components/dxLookup/

- (Param3) `[Function]` 리스트 아이템 dataSource 적용 함수

Param2 의 dxLookup 옵션 중 dataSource 속성 값에 직접 바인드 하는 경우 설정하지 않아도 된다.

- 예시

```
const phisWidget = this;
phisLookup(
    phisWidget.find('@cbo_srch'),
    { ... phisLookup 옵션 ... },
    () => { ... 데이터 조회 Function ... }
);
```

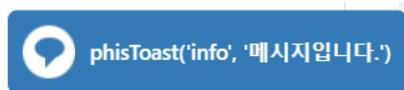
▼ phisToast

| 토스트 알림창UI 구현

- 설정 파라미터

- (Param1) `'info' | 'success' | 'warning' | 'error'` 토스트 알림창 종류

- info `Default`



phisToast 컴포넌트 'info' 타입 화면

- success



phisToast 컴포넌트 'success' 타입 화면

- warning



phisToast 컴포넌트 'warning' 타입 화면

- error



phisToast 컴포넌트 'error' 타입 화면

- (Param2) `[String]` 토스트 알림창에 보일 문구

- (Param3) `[Object]` 토스트 알림창UI 관련 세부 옵션 지정 (DevExpress.ui.notify 옵션)

- 기본 옵션

```
{  
    type: ['info', 'warning', 'error', 'success'].indexOf(type) !== -1 ? type : 'info',  
    message: msg ? msg : undefined,  
    width: 'auto',  
    height: 'auto',  
    displayTime: 5000,  
    closeOnClick: true,  
    maxCount: 5,  
    position: {  
        at: 'bottom right',  
        my: 'bottom right',  
        of: window,  
        offset: {  
            x: -10,  
            y: 0  
        }  
    },  
    animation: {  
        show: {  
            type: "slide",  
            from: {  
                left: -50  
            },  
            to: {  
                left: 0  
            },  
            duration: 200  
        }  
    }  
}
```

```
    }  
};
```

- DevExpress.ui.notify 옵션

<https://js.devexpress.com/Documentation/ApiReference/Common/Utils/ui/>

Documentation: DevExtreme - JavaScript Utils UI

```
import { Component } from '@angular/core'; import config from 'devextreme/core/config'; import  
repaintFloatingActionButton from 'devextreme/ui/speed_dial_action/repaint_floating_action_button';  
@Component({ selector: 'app-root', templateUrl: './app.component.html', styleUrls: ['./app.component.css'] })
```

 <https://js.devexpress.com/Documentation/ApiReference/Common/Utils/ui/>

- 예시

```
const phisWidget = this;  
phisToast('error', '비밀번호를 확인해주세요.', {  
  position: 'bottom right',  
  positionOf: phisWidget.getContentEl(),  
});
```

▼ phisPopup / phisModal

팝업창UI 구현

phisModal 은 배경에 Backdrop이 나타나는 모달 팝업 창UI 의 형태



phisModal 메소드는 phisPopup 메소드와 같은 속성을 공유한다.

phisPopup 과 다른 점인 모달 기능은 shading 속성값 true 로 적용된다.

Grid]) => {



phisPopup 컴포넌트 화면

[[instGrid]] =>



phisModal 컴포넌트 화면

- 설정 파라미터

- (Param1) [String] 팝업창 타이틀 문구
- (Param2) [String | Function] 팝업창 컨텐츠 영역 (dxPopup 옵션 중 contentTemplate)

 Param1, 2 는 각각 Param3 의 title, contentTemplate 속성 값으로 직접 설정할 수도 있다.

- (Param3) [Object] 팝업창UI 관련 세부 옵션 지정 (dxPopup 옵션)

- 기본 옵션

```
const headerHeight = $('div.phis-header-toolbar').height();
const viewPortHeight = window.innerHeight - headerHeight - 20;

{
    contentTemplate() {
        return $(`<div>`); // contents 등록하지 않는 경우
    },
    position: {
        at: 'top center',
        my: 'top center',
        of: $('div.phis-board-main'),
    },
    width: 'auto',
    height: `${viewPortHeight}px`,
    showTitle: !gfn_isNull(title), // Param1 (타이틀 문구) 없으면 보이지 않음
    showCloseButton: true,
    shading: false,
    dragEnabled: true,
    visible: true,
    closeOnOutsideClick : true,
    scrollToTop: true,
    /*
        phisModal 메소드의 경우에만 shading 속성이 true 이다.
        shading: true,
    */
}
}
```

- dxPopup 옵션

https://js.devexpress.com/Documentation/ApiReference/UI_Components/dxPopup/

Popup

 https://js.devexpress.com/Documentation/ApiReference/UI_Components/dxPopup/

- phisPopup, phisModal 별도 옵션

- dxPopup 옵션 외 소스 내부에서 제공하는 phisPopup 옵션

속성명	속성값
appendTarget	[JQuery<Element>] 팝업창을 띄울 기준 div (<code>default : 'div.main-body'</code>)
scrollToTop	[Boolean] 팝업 시 스크롤 최상단 이동여부 (<code>default : true</code>)

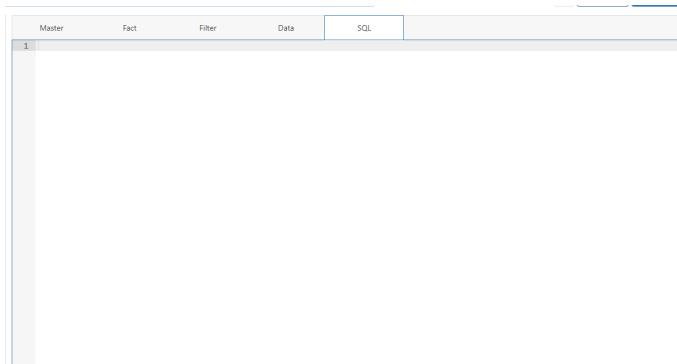
- 예시

```
const content = '<div>... phisModal 컴포넌트 요소 ...</div>';
phisModal('', content, {
    ... 생략 ...
    title: '[사용자 추가] 검색결과에서 더블클릭하면 추가됩니다.',
    position: {
        at: 'center',
        my: 'center',
        of: window,
    }
});
```

```
// ※ phisPopup 도 동일하게 적용
```

▼ phisInitAceEditor

ace-editor 로 SQL, Javascript, Html 등 에디터 구현



- 설정 파라미터
 - (Param1) [String] ace editor 컨테이너 div
 - (Param2) [String] ace editor 설정 타입
 - 'javascript', 'sql', 'html' 등
 - (Param3) [Object] ace editor 세부 옵션 지정 (ace editor 옵션)
 - 기본 옵션

```
{  
    fontSize: '1rem',  
    showPrintMargin: false,  
    useSoftTabs: true,  
    tabSize: 4,  
    css: {  
        marginTop: 0  
    }  
}
```

▪ ace editor 옵션

<https://ace.c9.io/api/modules/Ace.html>

Ace | Ace API Reference

Documentation for Ace API Reference

<https://ace.c9.io/api/modules/Ace.html>

- 예시

```
const phisWidget = this;  
phisInitAceEditor(phisWidget.find('@prop_js'), 'javascript');
```

▼ DOM 요소

메소드명	기능
\$.phisFind	'@phis-id속성값' (= [phis-id=""])에 해당하는 요소를 JQuery find 메소드로 찾고, 결과를 리턴
phisJQ	Html 태그를 String 입력 시 JQuery 요소로 바꾸어 결과값 리턴
phisJQAsync	phisJQ 를 async 처리 (=비동기), jquery 요소로 바꾸어 Promise. then 시점에 파라미터로 리턴

▼ \$.phisFind

- 설정 파라미터
 - (Param1) **[String]** @ + 'phis-id 속성값' 형태 선택자
- 반환 데이터
 - 선택자에 해당하는 요소 JQuery<Element> 형으로 리턴
- 예시

```
const phisWidget = this;
phisWidget.phisFind('@lst_toolbar');
```

▼ phisJQ / phisJQAsync

- 설정 파라미터
 - (Param1) **[String]** JQuery Element로 변환할 HTML Element 문자열 템플릿
- 반환 데이터
 - phisJQ
 - **[JQuery<Element>]** JQuery<Element> 변환된 요소를 리턴
 - phisJQAsync
 - **[Promise<T>]** then 시점에 파라미터로 JQuery<Element> 변환된 요소를 리턴
 - phisJQ 메소드 async 적용 메소드이다. (비동기 처리)
 - (장문의) 템플릿이 요소로 변환 완료된 직후에 변환된 요소를 선택자로 적용해야 하는 경우, JQuery Element 유무를 확인하여 적용할 수 있다.
- 예시

```
/*1*/ // 1. phisJQ
phisJQ('<div phis-id="textBox1" style="width: 250px;">');
/*2*/ // 2. phisJQAsync
phisJQAsync('<div phis-id="textBox1" style="width: 250px;">').then(jqEl => {
    jqEl.dxTextBox({... 옵션 ...});
});
```

▼ 데이터

메소드명	기능
gfn_isNull	입력된 파라미터가 null 인지 체크하고 맞으면 true 아니면 false 리턴
phisFormat	입력된 파라미터에 따라 데이터 포맷을 변경하여 리턴 및 동작 실행
phisMasking.name	입력된 파라미터 (이름 등 민감정보)의 첫번째, 마지막 자릿수를 제외한 나머지 문자를 모두 'o'로 치환하여 리턴
phisRandomString	입력된 파라미터 길이만큼 랜덤하게 선택된 문자 + 숫자 조합의 문자를 리턴

▼ gfnIsNull

- 설정 파라미터
 - (Param1) null 여부를 확인하려는 대상 데이터
- 반환 데이터
 - **[Boolean]** null 인 경우 true, null 이 아닌 경우 false
- 예시

```
/*1*/ const targetData1 = {};
/*2*/ const targetData2 = undefined;
```

```

/*3*/ const targetData3 = '';
/*4*/ const targetData4 = ' '; // 빈 문자열
/*5*/ const targetData5 = 0;

gfn_isNull(targetData1); /*1 결과*/ // false, null 아님
gfn_isNull(targetData2); /*2 결과*/ // true, null 및 음수
gfn_isNull(targetData3); /*3 결과*/ // true, null 및 음수
gfn_isNull(targetData4); /*4 결과*/ // false, null 아님
gfn_isNull(targetData5); /*5 결과*/ // false, null 아님

```

▼ phisFormat

- 설정 파라미터

- (Param1) `['3comma' | 'dateDay' | 'chartDayColor']` format 종류
- (Param2) format 종류별로 변환할 대상 데이터

Format	Param2 설명
3comma	[Number String] 3자릿 수 이상의 숫자 데이터
dateDay	[String] YYYYMMDD 형식의 날짜 데이터
month	[String] YYYYMM 형식의 날짜 데이터
chartDayColor	[String] 차트 Axis Label 데이터
gridDayColor	[0 4 8] 그리드 셀 데이터에 적용할 색상 (0 : 테마에 따른 기본색상 4 : 'blue' 색상 적용코드 8 : 'red' 색상 적용코드)

- (Param3) 'chartDayColor', 'gridDayColor' 의 적용 옵션

Format	Param3 설명	Param3 속성	속성 설명
chartDayColor	[Object] 색상을 적용 할 데이터 정보		
		dataField	색상을 적용할 데이터 컬럼KEY
		data	차트 바인드 데이터
gridDayColor	[Object] 색상을 적용 할 데이터 정보		
		text	색상을 적용할 셀 데이터
		container	셀 데이터 컨테이너 td 요소 (dxDataGrid 속성 중 cellTemplate 메소드 첫번째 파라미터)

- 반환 데이터 및 적용 동작

Format	반환 데이터	적용 동작
3comma	[String] ','로 3자리마다 구분된 숫자 데이터 (ex. 1,000)	-
dateDay	[String] 'MM/DD' + '날짜에 해당하는 요일'로 조합된 문자 데이터 (ex. 1/1 토)	-
month	[String] '/'를 구분자로 년도 4자리와 월 2자리가 조합된 문자데이터 (ex. 2022/01)	-
chartDayColor	-	[Object] 차트 Axis Label 에 토요일은 'blue', 일요일 및 공휴일은 'red' 색상을 적용
gridDayColor	-	[Object] 그리드 셀 데이터에 코드에 따라 색상을 적용

- 예시

```

/*1*/ // '3comma' 적용
phisFormat('3comma', 123456789); /*1 결과*/ // '123,456,789'

```

```

/*2*/ // 'dateDay' 적용
phisFormat('dateDay', '20220505'); /*2 결과*/ // '5/5 목'

/*3*/ // 'month' 적용
phisFormat('month', '202205'); /*3 결과*/ // '2022/05'

/*4*/ // 'chartDayColor' 적용
// value 같은 xAxis.axisLabel.textStyle.color = (value, index) 로 리턴되는 파라미터
phisFormat('chartDayColor', value, {
    dataField: 'disp_enddd',
    data: [
        // 차트 바인드 데이터
        { disp_enddd: "10/11 화", dutflag: 0, enddd: "20221011", gcnt: 141, ...},
        { disp_enddd: "10/12 수", dutflag: 0, enddd: "20221012", gcnt: 152, ...}
    ]
});

/*5*/ // 'gridDayColor' 적용
phisFormat('gridDayColor', 0, {
    // 셀 텍스트
    // text는 columns[인덱스].cellTemplate = (container, option)로 리턴되는 파라미터 option 해당 셀의 데이터
    text: option.data.컬럼 dataField,
    // 셀 td 요소
    // container 값은 columns[인덱스].cellTemplate = (container, option)로 리턴되는 파라미터
    container: container
});

```

▼ phisMasking.name

- 설정 파라미터
 - (Param1) `[String | Number]` 첫번째, 마지막 자릿수를 제외한 나머지 문자를 모두 'o'로 치환할 문자
- 반환 데이터
 - `[String]` 파라미터로 입력된 데이터의 첫번째, 마지막 자릿수를 제외한 나머지 문자가 모두 'o'로 치환된 문자
- 예시

```

phisMasking.name('고로케'); // '고○케'
phisMasking.name('선우용녀'); // '선○○녀'

```

▼ phisRandomString

- 설정 파라미터
 - (Param1) `[Number | String]` 필요한 랜덤 문자의 수 (or String 형 숫자)
- 반환 데이터
 - `[String]` Uint32Array 타입으로 랜덤하게 선택된 문자 + 숫자 조합의 문자
- 예시

```

phisRandomString(1); // 'f'
phisRandomString('5'); // 'c379ot'
phisRandomString(5); // 'c48jx'

```

▼ 1.3. 쿼리

▼ (1) 개요

“위젯관리” 페이지 내의 Widget Master (WIDGETMAST) 위젯의 “쿼리” 탭에서 각 위젯에 대한 쿼리 작성 및 저장한다.

작성된 쿼리는 별도의 배포 과정 또는 컴파일 필요 없이 적용, 위젯 실행하여 확인이 가능하다.

과거 저장했던 변경 이력을 확인할 수 있다. ([ZZ.ZNBMQSTMT](#) 테이블 참조)

The screenshot shows the Widget Master interface with the following details:

- A.위젯:** 당일 환자현황 WyknnKTGYI OBJECT
- 최초저장:** 2022-02-07 09:49 **최종저장:** 2022-06-15 09:12
- Query History:**

```

1 <![CDATA[
2 select a.orddd as orddd --진료일자
3 , to_char(to_date(a.orddd, 'yyyymmdd'), 'mm/dd') as disp_orddd
4 , to_char(sysdate, 'HH24:MI') as ordtm --조회시각
5 , count(*) as totcnt --접수건수
6 , sum(case when substr(a.fstacptdt,1,8) < a.orddd then 1 else 0 end) as rsrvcnt --예약접수
7 , count(*) - sum(case when substr(a.fstacptdt,1,8) < a.orddd then 1 else 0 end) as todaycnt --당일접수
8 , SUM(case when a.drcptyn = 'Y' then 1 else 0 end) as cnfmcnt -- 진료완료
9 , SUM(case when ( select count(*)
10          from pam.paohopni x
11         where x.instcd = a.instcd
12         and x.pid = a.pid
13         and x.orddd = a.orddd
14         and x.cretno = a.cretno
15         and x.rctexedcd = to_char(sysdate, 'yyyymmdd')
16         and x.unccrptflag != '1A'
17         and x.rctstat = 'Y' ) > 0 then 1
18        else 0
19       end
20      ) as rcptcnt --수납완료
21 , sum(decode(a.fsexamflag, 'F',1,0)) as fstcnt --방초진
22 , round(sum(decode(a.fsexamflag, 'F',1,0)) / count(*) * 100,1) || '%' as fstrate --방초진율
23 , ( SELECT count(*) AS cnt
24        FROM pam.pmhinpt y
25       , pam.pbmmdre x
26      WHERE y.instcd = a.instcd
27      and a.orddd between y.indd and y.dschdd
28      and y.mskind = 'M'
29      and y.histstat = 'Y'
30      and y.ordtype = 'I'
31      and y.indschapctstat = 'A'
32      and x.instcd = y.instcd

```

• [ZZ.ZNBMQSTMT](#) 테이블

컬럼명	데이터 타입	설명
QSTMT_GRUP_CD	VARCHAR2	쿼리그룹ID (dev, emr ...)
QSTMT_ID	VARCHAR2	쿼리 statement ID (쿼리그룹ID.위젯ID 의 형태) (※ 아래 ZZ.ZNBMCODE 에서 쿼리그룹ID 참고)
QSTMT_SEQ	NUMBER	0 : 유효 쿼리 1 이상 : 이력 기록
QSTMT	CLOB	쿼리문
ORG_QSTMT_SEQ	NUMBER	0 : qstmt_seq가 0 인 경우, 그 외는 원번호

• [ZZ.ZNBMCODE](#) 테이블

컬럼명	설명	비고
GRUP_NB_CD	"SYS_CONTENT.DAO" 고정	SELECT * FROM ZZ.ZNBMCODE WHERE grup_nb_cd='SYS_CONTENT.DAO';
NB_CD	쿼리그룹ID (Content 의 DAO)	
VALI_FRDD	유효시작일	yyyymmdd 형식의 날짜 데이터를 문자열로 설정
VALI_TODD	유효종료일	yyyymmdd 형식의 날짜 데이터를 문자열로 설정
NB_NM	Content 의 DAO 설명	
SORT_NO	순서	
NB_DESC	DAO version	
NB_ATTR	멀티 DB 구분	

▶ 쿼리그룹ID

nb_cd	nb_nm
dev	개발도구
emr	EMR

▶ 멀티 DB 구분

	설명
dsn	jndi명
dbms	ora : Oracle pg : PostgreSQL

nb_cd	nb_nm
edw	EDW
mngt	경영현황
mig	Data Trans / Mig
etc	기타

▼ (2) 쿼리 작성

▼ 1. 기본 작성

▼ 작성 방법

- 위젯 유형이 "OBJECT", "ADMIN" 이면 `SELECT` 쿼리만 적용 가능하고, "BATCH" 유형은 모든 `DML 쿼리`를 작성할 수 있다.
 - `INSERT INTO`
 - `MERGE INTO`
 - `DELETE FROM`
 - `UPDATE ... SET`



Batcher 항목 내용 참고

1. ibatis 문법을 따른다. (JDK 1.4)

- 파라미터 형태 : `#param#`, `$param`
- `<isNotEmpty>`, `<isEqual>` 등 ... 태그 사용 가능
- 예시 |

```
<isEqual property="query_flag" compare="getGrupList"><![CDATA[
SELECT INST_CD
, NB_CD AS GRUP_CD
, NB_NM AS GRUP_NM
, JSON_VALUE(NB_ATTR, '$$.GRUP') as GRUP /* 특수문자 #,$ 사용 시 문법 : ##, $$ */
, JSON_VALUE(NB_ATTR, '$$.REQ_FLAG') as REQ_FLAG
, JSON_VALUE(NB_ATTR, '$$.REQ_FLAG2') as REQ_FLAG2
FROM ZZ.ZNBMCODE
WHERE INST_CD = %dutplceinstcd#
... 생략 ...
]]></isEqual>
```

2. 하나의 Statement ID (`qstmt_id`) 만 읽어 들이기 때문에 두 개 이상의 쿼리 문을 적용하고 싶다면 `<isEqual>` 태그를 사용해야 한다.

이는 반드시 `query_flag`로 구분되어야 하고 그 값은 유일해야 한다.

EX) ① `<isEqual property="query_flag" compare="getData_1">`
 ② `<isEqual property="query_flag" compare="getData_2">`

- “쿼리” 탭 내 스크립트 최종 형태 예시

```

sql.xml 파일

<?xml version="1.0" encoding="EUC-KR"?>
<sqls>
<statement id="qstmt_id"><![CDATA[
----->
-----</statement>
</sqls>

```

```

A위치: [ ] SaveAs [ ] 주기
조작법 Q 검색 워크 속성 최근 O 설명 이미지 관리
최종저장: 앤디월 2022-10-14 16:10 최종저장: 앤디월 2022-10-14 16:10
1 <isEqual property="query_flag" compare="de_list"><![CDATA[
2 select dm_id as dm_id
3 , dm_nm as dm_nm
4 , TO_CHAR(LST.RGST_DT,'YYYY-MM-DD') AS updt
5 , TO_CHAR(LST.RGST_DT,'YYYY-MM-DD') AS updt
6 from zz_zmbedodata
7 where dm_id = #dm_id# and dm_nm = #dm_nm#
8 ORDER BY LST.RGST_DT DESC
9 </isEqual>
10 <isEqual property="query_flag" compare="de_info"><![CDATA[
11 select dm_id
12 , dm_nm
13 , dm_desc
14 , dm_tbl_nm
15 from zz_zmbedodata
16 where dm_id = #dm_id#
17 and dm_nm = #dm_nm#
18 </isEqual>
19 <isEqual property="query_flag" compare="de_item"><![CDATA[
20 SELECT "Y" USE_YN
21 , DM_ID
22 , DM_NM
23 , DM_DESC
24 , DM_TBL_NM
25 , DD_ID
26 , COL_NM
27 , DATA_TYPE
28 , DATA_LENGTH
29

```

▼ 작성 예시

- 쿼리문이 한 개일 경우

```

<! [CDATA[
    SELECT ...
    FROM TABLE_NAME ...
]]>

```

- 쿼리문이 두 개 이상일 경우

```

<isEqual property="query_flag" compare="getData_1"><![CDATA[
    SELECT ...
    FROM TABLE_NAME ...
]]></isEqual>

<isEqual property="query_flag" compare="getData_2"><![CDATA[
    SELECT ...
    FROM TABLE_NAME ...
]]></isEqual>

<isEqual property="query_flag" compare="getData_3"><![CDATA[
    SELECT ...
    FROM TABLE_NAME ...
]]></isEqual>

```

- 쿼리 실행 요청 시 속성 스크립트 작성 방법

- 공통 함수 `gfn_submit` 으로 데이터를 서버에 요청 및 반환한다.

1.2. 속성 - (3)속성 작성 - 3. 공통함수 참고

■ 속성 탭 작성

```

const reqData = {
  /* 필수 */
  qstmt_id : '워켓의 쿼리ID',
  query_flag : '<isEqual>의 property와 compare 속성',

  /* 옵션 */
  // 쿼리 파라미터, <isEqual> 내 다른 동적 태그 속성 ..
  param1 : 'param1 값',
  param2 : 'param2 값', ...
}

gfn_submit(
  '서비스코드', '서버참조명', '요청 데이터',
  '요청 후 콜백함수',
  '시스템코드',

```

```
'시스템 메시지 팝업사용여부'  
);
```

▪ 쿼리 탭 작성

```
<isEqual property="query_flag" compare="compare 속성">  
<!CDATA[  
    SELECT ...  
    FROM TABLE_NAME  
    WHERE COL1 = 'VALUE'  
    <isNotEmpty property="param1">  
        AND COL2 = #param2#  
    </isNotEmpty>  
]]>  
</isEqual>
```

• 속성 탭 작성 예시

```
const reqData = {  
    qstmt_id : 'dev.wliqdptiyd',  
    query_flag : 'getAuth',  
    nbrd_id : 'Bctvsc8gxa',  
    user_grup_cd : 'UG_tblkljh'  
};  
  
gfn_submit('DRZNB20204', 'req', reqData, () => {  
    ... Callback 할수 ...  
}, null, false);
```

• 쿼리 탭 작성 예시

```
<isEqual property="query_flag" compare="getAuth">  
<!CDATA[  
    SELECT *  
    FROM ZZ.ZNBMAUTH  
    WHERE INST_CD = #%systeminstcd#  
        AND NBRD_ID = #nbrd_id#  
    <isNotEmpty property="user_grup_cd">  
        AND USER_GRUP_CD = #user_grup_cd#  
    </isNotEmpty>  
]]>  
</isEqual>
```

▼ SQL 공통 함수



COM 테이블로 사용하는 공통함수는 프로젝트 환경에 따라 상이할 수 있음.



아래 표의 '기준일'은 모두 VARCHAR2 타입이며 '**YYYYMMDD**'의 형태이다.

데이터 타입이 따로 기재되지 않은 리턴 타입과 파라미터는 모두 **VARCHAR2** 타입이다.

함수명	리턴타입과 기능	파라미터
COM.FN_ZB_GETADDRESS	우편번호를 이용해 주소를 조회한다.	(Param1) 우편번호 앞 3자리 (Param2) 우편번호 뒤 3자리 (Param3) 우편번호 순서 (일련번호) NUMBER
COM.FN_ZS_GETDEPTNAME	부서의 한글명을 조회한다.	(Param1) 기관코드 (Param2) 부서코드 (Param3) 기준일 DEFAULT : 19000101 ~ 99991231
COM.FN_ZS_GETDEPTNM	부서의 한글명을 조회한다. (2)	" 기준일 DEFAULT값 없음
COM.FN_ZS_GETLATESTDEPTNM	부서의 한글명을 조회한다. (3)	" 기준일 DEFAULT값 없음
COM.FN_ZS_GETUSERNM	사용자의 한글명을 조회한다.	(Param1) 사용자ID (Param2) 기준일
COM.FN_ZZ_GETAGE	주민번호 / 생년월일로 만나이를 반환한다. (주민번호 블분명자에 대한 처리도 가능하다.)	(Param1) 주민번호 앞 6자리 (Param2) 주민번호 뒤 7자리 (Param3) 나이계산 기준일

함수명	리턴타입과 기능	파라미터
		(Param4) 계산 구분 (A : 주민번호 기준 B : 생년월일 기준)
COM.FN_ZZ_GETAGEAS	주민번호 / 생년월일로 만나이를 반환한다. (2) (20세 이상인 경우 나이만 반환, 20세 미만인 경우 개월/일수 반환)	“”
COM.FN_ZZ_GETAGE_EMR	주민번호 / 생년월일로 만나이를 반환한다. (3) (4세 이상인 경우 나이만 반환, 4세 미만인 경우 개월/일수 반환)	“”
COM.FN_ZZ_GETINHOSPTIME	입원 시간을 계산한다. (날짜 비교)	(Param1) 비교할 날짜(1) 20자리 (Param2) 비교할 날짜(2) 20자리 (Param3) 표시구분 (1 : 일,시,분 2 : 시,분)
COM.FN_ZZ_GETINHOSPTIME_KSS	입원 시간을 계산한다. (날짜 비교) (2)	“”
COM.FN_ZZ_GETSEX	주민 번호로 성별을 구분한다.	(Param1) 주민번호 앞자리 6글자 (Param2) 주민번호 뒷자리 6글자 (Param3) 표현방식 (1 : 남 · 여 2 : F · M 3 : 남자 · 여자 4 : Female · Male)
COM.FN_ZZ_MD5	문자를 md5 암호화한다.	(Param1) 암호화할 문자
COM.FN_ZZ_TO_TMSTMP	TIMESTAMP 문자 타입으로 된 날짜 를 TIMESTAMP 타입으로 바꾼다.	(Param1) 날짜 14자리 혹은 20자리 (그 외 자릿수가 아닐시 NULL)

- 공통 함수 작성 예시

```
/*
SELECT COM.FN_ZS_GETDEPTNAME(INSTCD, ORDDEPTCD, ORDDD)
  FROM PAM.PMOHOTPT ;

/*
SELECT NBRD_ID
      , NBRD_NM
      , COM.FN_ZS_GETUSERNM(FST_RGSTR_ID, TO_CHAR(FST_RGST_TS, 'YYYY-MM-DD HH24:MI')) AS FST_RFSTR_ID
      , TO_CHAR(FST_RGST_TS, 'YYYY-MM-DD HH24:MI') AS FST_RGST_TS
      , COM.FN_ZS_GETUSERNM(LAST_UPDTR_ID, TO_CHAR(LAST_UPDT_TS, 'YYYYMMDD')) AS LAST_UPDTR_ID
      , TO_CHAR(LAST_UPDT_TS, 'YYYY-MM-DD HH24:MI') AS LAST_UPDT_TS
  FROM ZZ.ZNBMNBRD ;
```

▼ 시스템 정보 변수

시스템 정보 변수는 프로젝트 환경에 따라 상이할 수 있음.

| 세션에 저장된 시스템 사용자 정보를 SQL문에서 파라미터로 사용할 수 있다.

파라미터 사용 시 포맷은 다음과 같다.
: #param#

해당 세션은 cmc_config.xml 파일에 설정되어있다. (...\\webhome\\app_home\\cmcnu\\WEB-INF\\classes\\resources\\cmc_config.xml)
아래의 표는 사용할 수 있는 세션 정보 파라미터 목록이다.

시스템 변수	설명	시스템 변수	설명
userid	사용자ID	usergrp	사용자 계정그룹 별칭명

usernm	사용자명	userkindcd	사용자유형 (공통코드로 관리)
userengnm	사용자 영문 이름	medispclno	전문의번호
dutinstcd	근무기관코드	deptabbr	부서 영문 약어
dutinstnm	근무기관명	deptengnm	부서코드의 영문명칭
dutunitcd	근무부서코드	jobkindcd	직종코드
dutunitnm	근무부서명	jobkindnm	직종
posinstcd	인사상 소속기관 코드	jobposcd	직위코드
posinstnm	인사상 소속기관명	jobposnm	직위
posdeptcd	인사상 소속부서코드	jobrespcd	직책코드
posdeptnm	인사상 소속부서명	jobrespnm	직책
orginstcd	인사상 원소속기관코드	prfshipflagcd	교직구분 코드
orginstnm	인사상 원소속기관명	prfshipflagnm	교직구분
orgdeptcd	인사상 원소속부서코드	grupjoblnkd	직군 (공통코드로 관리)
orgdeptnm	인사상 원소속부서명	licnsno	자격면허번호
ordsupdeptcd	진료상위부서	specordyn	선택진료여부 Y/N
dutplceinstcd	실제 근무하고 있는 기관코드	orddeptflag	진료부서를 구분 (진료 사용여부가 Y인 경우)
dutplceinstnm	실제 근무하고 있는 기관명	wardflag	병동구분
dutplcecd	실제 근무하고 있는 근무지코드	logindt	로그인한 날짜와 시간
dutplcenm	실제 근무하고 있는 근무지명	ipaddr	아이피 주소
systemcd	시스템 코드	mpphonno	핸드폰번호
systemnm	시스템 명	vpn	VPN client 여부 Y/N
systeminstcd	시스템이 속한 기관코드	kmiip	KMI 아이피
systeminstnm	시스템이 속한 기관명	kmiport	KMI 포트

- 시스템 변수 사용 예시

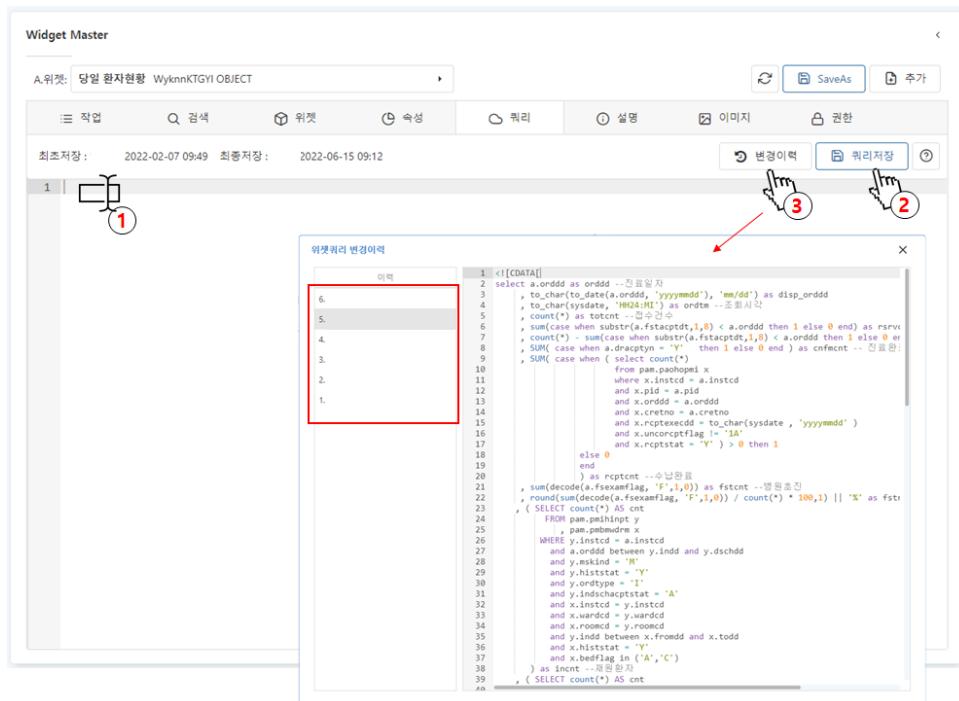
```

/* 1 */
SELECT *
  FROM ZZ.ZNBMWIDGET
 WHERE INST_CD = #%dutplceinstcd#


/* 2 */
SELECT *
  FROM ZZ.ZNBDPAGE
 WHERE PAGE_ID = #page_id#
   AND INST_CD = #%dutplceinstcd#
   AND USER_ID = #%userid#

```

▼ 2. 기본 동작



- ① 에디터에 쿼리를 작성한다.
- ② '쿼리저장'을 클릭한다.
- ③ '변경이력' 버튼을 클릭하여 변경이력 팝업으로 해당 위젯의 쿼리 과거 저장 이력을 확인할 수 있다. (더블 클릭으로 각 이력 확인)

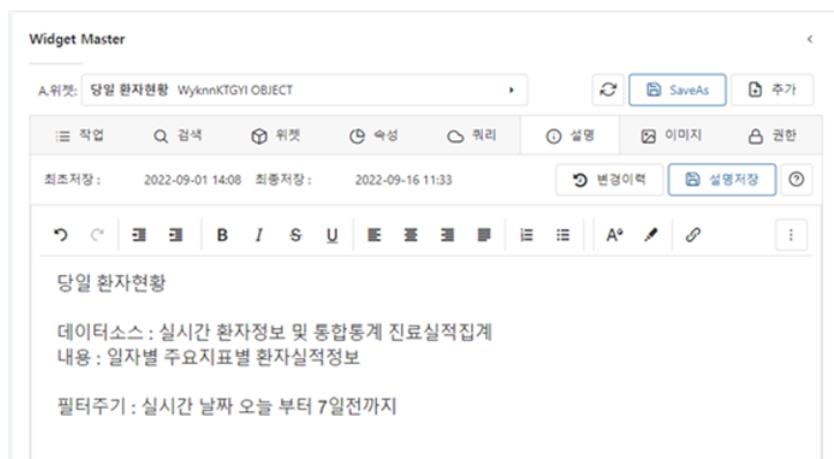
▼ 1.4. 설명

▼ (1) 개요

위젯에 대한 설명을 HtmlEditor로 작성할 수 있다.

작성된 설명은 HTML 형태로 DB에 저장된다.

위젯 toolBox의 '설명' 팝업으로 위젯 설명을 확인할 수 있다.



설명 데이터 저장 테이블 ZZ.ZNBMCDATA

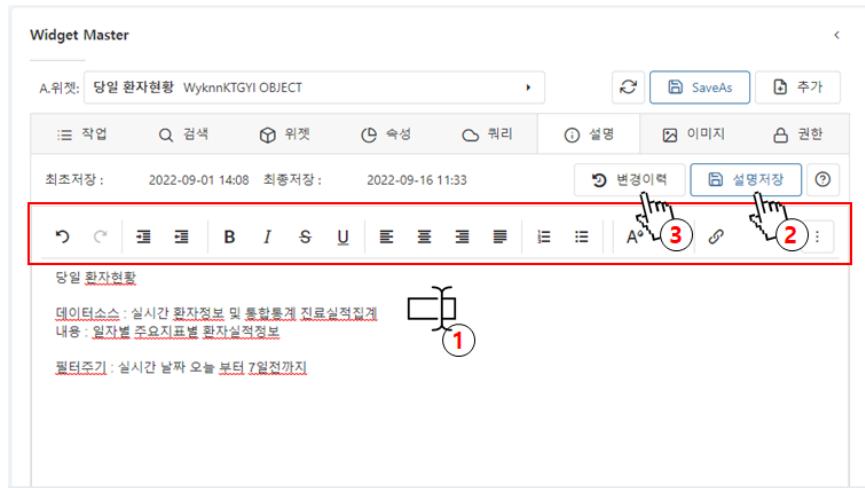
컬럼	설명
CDATA_ID	설명ID (ZZ.ZNBMWIDGET 참조)
CDATA_TYPE	"WIDGET_DESC" 고정
CDATA	설명 내용
CDATA_NM	위젯명 (ZZ.ZNBMWIDGET 참조)

설명 탭 이력 데이터 테이블 ZZ.ZNBMQSTM

컬럼	설명
QSTM_ID_GRUP_CD	"HIST_CDATA" 고정
QSTM_ID	설명ID (ZZ.ZNBMWIDGET 참조)
QSTM_SEQ	이력 순서
QSTM	설명 내용

▼ (2) 설명 작성

▼ 1. 기본 동작



- ① Editor에 위젯 설명을 작성한다.
- ② '설명저장' 버튼을 클릭하여 저장한다.
- ③ '변경이력' 버튼을 클릭하여 과거 저작 이력을 확인할 수 있다.

▼ 2. 내용 확인



- ① 설명을 작성한 위젯의 위젯 우측 상단에 있는 '더보기' 아이콘을 클릭한다.
- ② ToolBox 아이콘들 중 '설명' 아이콘을 클릭하여 뜨는 팝업에서 설명 내용을 확인한다.

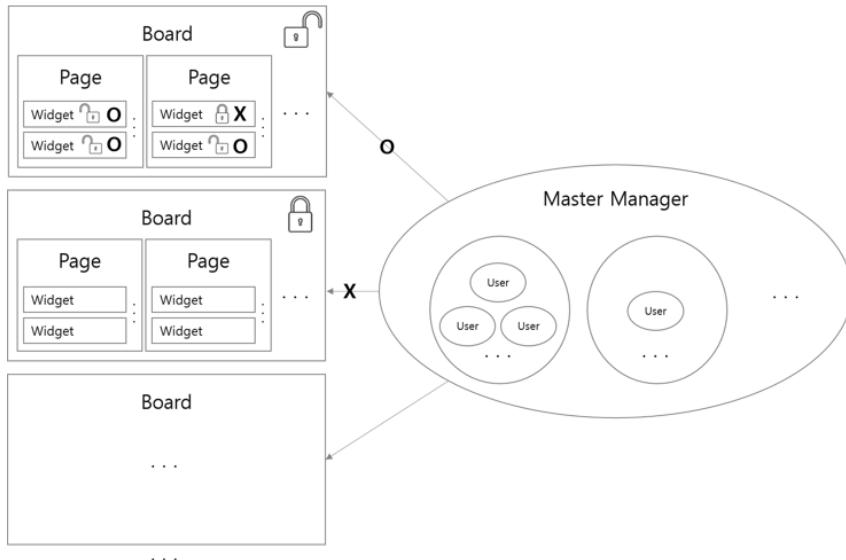
▼ 1.5. 권한

▼ (1) 개요

권한 관리는 데이터의 성격에 따라 권한 관리자가 "사용자"의 "위젯" 혹은 "보드" 열람 범위를 지정하는 동작이다.

사용자의 근무지, 부서, 직종, 직책 등에 따라 사용자에게 정보만 전달할 수 있다.

"관리자"에 대한 권한 추가 시, 2. 보드관리 - 2.2.(2). 인트로 설정 참고



Master Manager(권한 관리자)가 User(사용자)의 Board, Widget 열람을 제한/허용한다.

• 권한설정

권한설정은 생성한 보드, 위젯을 기준으로 사용자를 개별/일괄로 등록하여 관리한다.

! 아래 표는 권한 설정 및 조회 위젯을 설명한다. ([ZZ.ZNBMWIDGET](#) 기준)
표의 "위젯ID" 또는 "위젯명" 컬럼의 값은 권한관리자에 의해 변경될 수 있다. (절대값 아님)

◦ 개별 관리

▪ 개별권한 설정 및 조회 위젯 ([ZZ.ZNBMWIDGET](#) 기준)

위젯명	위젯ID	기능분류	순서	기능	필수 적용값
WidgetMaster	WIDGETMAST	권한 설정	1	"위젯" 기준으로 계정그룹, 부서, 직종, 직책 등의 조건의 권한을 "보드"에 적용한다.	위젯ID, 조건(허용/제한), 보드, 사용자정보(계정그룹, 근무지부서, 직종, 직책, 근무기관, 소속기관, 사용자유형, 직군, 교직구분, 직위)

◦ 일괄 관리

▪ 일괄권한 설정 및 조회 위젯 ([ZZ.ZNBMWIDGET](#) 기준)

위젯명	위젯ID	기능분류	순서	기능	필수 적용값
보드위젯권한조회 (권한A)	Wydrea8pos	보드/위젯 권한 조회	1	"보드" 기준으로 조회된 "위젯"별 권한 정보를 확인한다.	"보드"
			2	"위젯" 기준으로 조회된 "보드"별 권한 정보를 확인한다.	"위젯"
			3	"권한수정일" 기준으로 조회된 "보드"별 권한 정보를 확인한다.	"권한수정일"
			4	설정 기준없이 조회된 "권한명"(직책, 사용자유형)별 정보를 확인한다.	없음
			5	1, 2, 3의 기준이 종합적으로 적용된 권한 정보를 확인한다.	"보드" (or/and) "위젯" (or/and) "권한수정일"
사용자검색 (권한B)	W84grwpukl	사용자 권한 조회	1	사용자 정보(필수 적용값 참고)를 기준으로 조회된 사용자정보 ("사용자ID", "성명", "사용자그룹" 등)를 확인한다.	"사용자ID/성명" (or/and) "사용자그룹" (or/and) "근무지" "직책" "직종" "근무기관" "사용자유형" "직군" "교직구분" "직위"
			2	"보드" 와 "위젯"을 기준으로 조회된 사용자정보("사용자ID", "성명", "사용자그룹" 등)를 확인한다.	"보드" (or/and) "위젯" (or/and)
사용자권한조회 (권한C)	Wfowpcnigm	권한 검색	1	설정 기준없이 조회된 모든 보드 상의 전체 위젯을 확인한다.	없음
			2	"보드" 와 "위젯"을 기준으로 조회된 위젯 목록을 확인한다.	"보드" (or/and) "위젯" (or/and)
			3	"보드"와 "위젯"을 기준으로 보드위젯 권한조회(권한A)"에서 "권한" 정보를 확인한다.	"보드" (or/and) "위젯" (or/and)
			4	"보드"와 "위젯"을 기준으로 사용자검색(권한B)"에서 "사용자"의 권한 정보를 확인한다.	"보드" (or/and) "위젯" (or/and)
사용자그룹 관리	Whghb7g56t	사용자그룹 권한 추가	1	"추가" 선택 후 사용자 그룹 정보 ("그룹코드", "그룹명", "시작일자", "종료일자" 등)를 기입 후 "저장" 하여 사용자 그룹 권한을 추가한다	"그룹코드" "시작일자" "종료일자" "순서"
사용자그룹 사용자 관리	Wvp4dn1ga	그룹별 사용자 권한 추가	1	"사용자그룹" 선택 후 "사용자추가"로 사용자 정보를 검	"사용자그룹" "유효" "사용자ID" "사용자" "근무지" "시

위젯명	위젯ID	기능분류	순서	기능	필수 적용값
				색하고 "저장" 하여 사용자 권한을 추가한다.	작일자" "증료일자"

▼ (2) 권한 관리

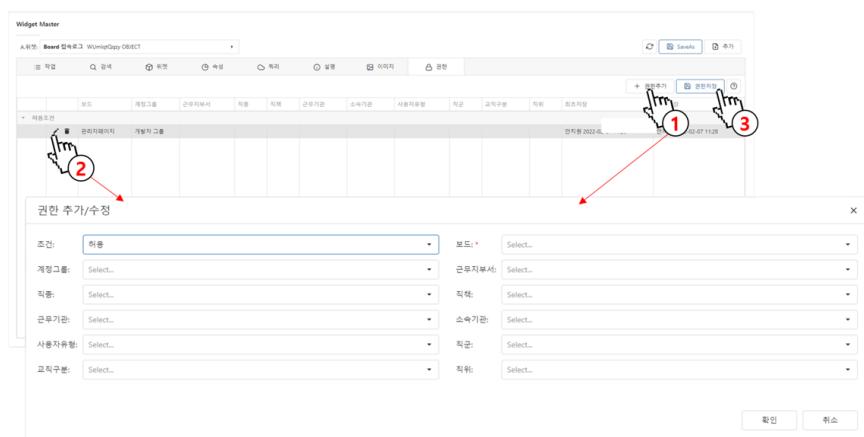
▼ 1. 위젯관리 페이지

▼ 개별관리

- WidgetMaster ('권한' 탭)

| 위젯 단위로 권한 정보를 추가하여 위젯 열림을 제한 할 수 있다.

▼ 기본동작



A. 위젯으로 선택한 위젯의 권한을 1 → 2 → 3의 순서에 따라 개별적으로 허용/제한할 수 있다.

① 새로운 권한 그룹을 추가하려는 경우 "권한추가" 버튼을 클릭하여 나타나는 팝업으로 수정한다.

② "속성저장" 버튼을 클릭하여 수정한 속성을 저장 및 적용한다.

③ 기존 허용된 그룹의 권한을 변경하려는 경우, 그리드의 펜 아이콘을 클릭하여 나타나는 팝업으로 수정한다.

(아래 '권한그룹 수정'에서 팝업 정보 참고)

▼ 권한그룹 수정

- 권한 추가/수정 팝업으로 권한그룹을 수정 및 추가할 수 있다.



아래 표는 권한 추가/수정 시 필요한 조건들이다.

항목	설명	항목 KEY (ZZ_ZNBMAUTH 테이블 컬럼)	COM_ZSUMUSRD 테 이블 컬럼	COM_ZSUMUSR 테이블 컬럼
조건	허용/제한 여부 지정	INCLUDE_YN	-	-
보드	위젯 접근 허용/제한 보드	NBRD_ID	-	-

항목	설명	항목 KEY (ZZ_ZNBMAUTH 테이블 컬럼)	COM.ZSUMUSRD 테이블 컬럼	COM.ZSUMUSRB 테이블 컬럼
계정그룹	위젯 접근 허용/제한 그룹	USER_GRUP_CD	-	-
근무지부서	위젯 접근 허용/제한 부서	WRKPLC_CD	DUTPLCECD	-
직종	위젯 접근 허용/제한 직종	JOBKND_CD	JOBKINDCD	-
직책	위젯 접근 허용/제한 직책	JOBRSP_CD	JOBRESPCD	-
근무기관	위젯 접근 허용/제한 근무 기관	DUT_INST_CD	-	DUTINSTCD
소속기관	위젯 접근 허용/제한 소속 기관	ORG_INST_CD	-	ORGINSTCD
사용자 유형	위젯 접근 허용/제한 사용자유형	USER_TYPE_CD	-	USERKINDCD
직군	위젯 접근 허용/제한 직군	OCCGRP_CD	-	GRUPJOBLNCD
교직구분	위젯 접근 허용/제한 교직 구분	PRFSHP_FLAG_CD	PRFSHIPFLAGCD	-
직위	위젯 접근 허용/제한 직위	JOBPOS_CD	JOBPOSCD	-

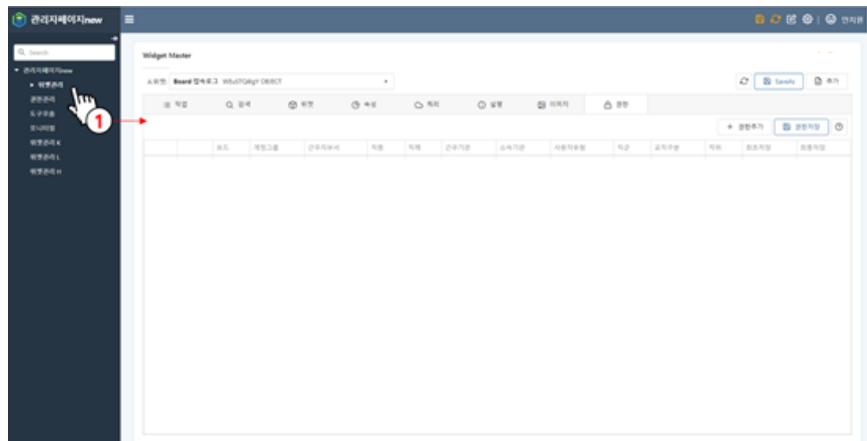
▼ 예시 케이스

새로만든 “신규위젯” 위젯을 “신규보드” 보드에 “우리병원” 소속기관의 “원장” 만 열람할 수 있도록 권한을 추가하는 경우

- 위젯정보
- 사용자정보

위젯이름	보드이름	소속기관	직책
신규위젯	신규보드	우리병원	원장

1. “관리자페이지” 보드의 “위젯관리” ① 페이지에 접속 한다.
2. “Widget Master (WIDGETMAST)” 위젯의 “A.위젯” 항목에서 ② “신규위젯”을 선택 및 ③ “권한” 탭으로 이동한다.
3. “권한” 탭 내 ④ “권한추가” 버튼을 클릭한다.
4. 팝업 오픈 후 “권한 추가/수정” 내에서 조건에 따라 ⑤ 값 지정 후 ⑥ “확인” 버튼을 클릭한다.
(지정값 예시: “조건” – “허용”, “보드” – “신규보드”, “소속기관” – “우리병원”, “직책” – “원장”)
5. 팝업 창 닫힌 후 그리드 상에 추가된 ⑦ 조건 확인 후 ⑧ “권한저장” 버튼을 클릭한다.



1. “관리자페이지” 보드의 “위젯관리” 페이지에 접속 ①



2. “Widget Master (WIDGETMASTER)” 위젯의 “A.위젯” 항목에서 “신규위젯”을 선택 ② 및 “권한” 탭으로 이동 ③



3. “권한” 탭 내 “권한추가” 버튼을 클릭 ④



4. 팝업 오픈 후 “권한 추가/수정” 내에서 조건에 따라 값 지정 ⑤ 후 “확인” 버튼 클릭 ⑥
(지정값 예시: “조건” – “허용”, “보드” – “신규보드”, “소속기관” – “우리병원”, “직책” – “원장”)



5. 팝업 창 닫힌 후 그리드 상에 추가된 조건 확인 ⑦ 후 “권한저장” 버튼 클릭 ⑧

▼ 2. 권한관리 페이지

▼ 일괄관리

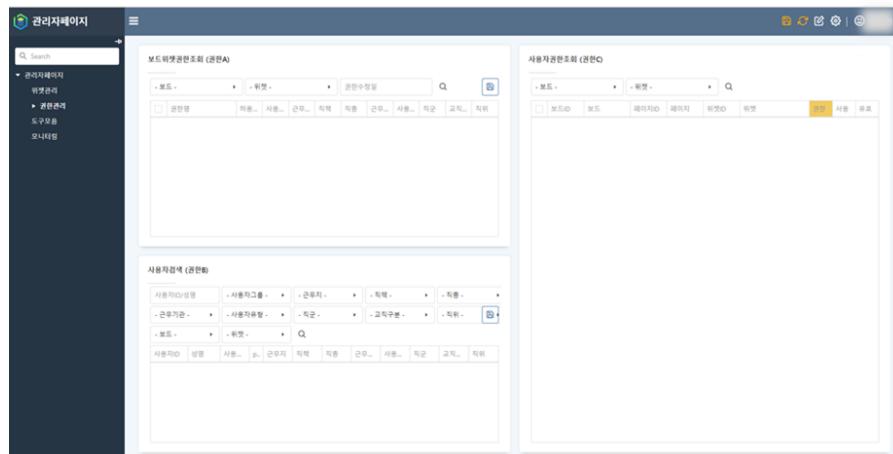
▼ 권한 A, B, C

권한관리 페이지는 위젯관리 페이지의 위젯마스터 권한 탭 이외에 추가적으로 권한 조회, 추가 기능을 제공한다.

사용자별 또는 보드, 위젯별로 권한 조회가 가능하며, 권한 추가 시 하나 이상의 권한을 일괄적으로 추가할 수 있다.

보드위젯권한조회(권한A), 사용자검색(권한B), 사용자권한조회(권한C) 위젯을 이용한다.

 “보드위젯권한조회(권한A)”, “사용자검색(권한B)”, “사용자권한조회(권한C)” 위젯을 편의상 각각 “권한A”, “권한B”, “권한C”로 별칭한다.



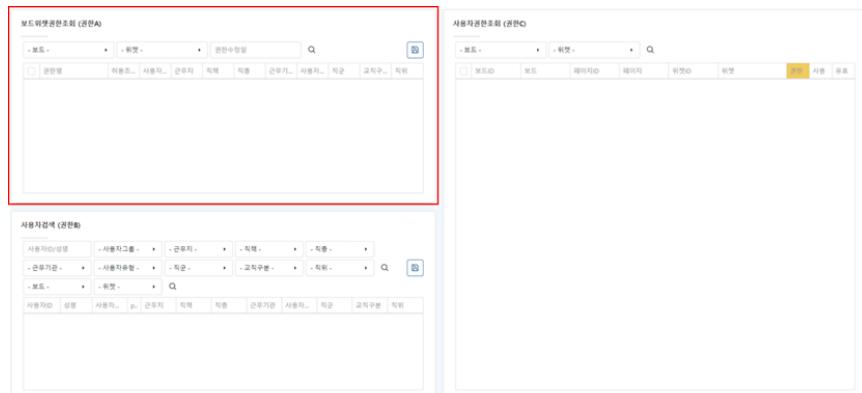
보드위젯권한조회(권한A), 사용자검색(권한B), 사용자권한조회(권한C) 위젯을 모두 이용할 수 있는 권한 관리 페이지

▼ 보드위젯권한조회(권한A)

보드와 위젯, 권한수정일을 검색 조건으로 위젯마스터의 권한 탭에 등록된 항목을 중복 제거 후 표시한다.

위젯 마스터의 권한 탭에서 가능했던 위젯별 조회뿐 아니라 보드와 권한수정일을 검색조건으로 권한을 조회할 수 있다.

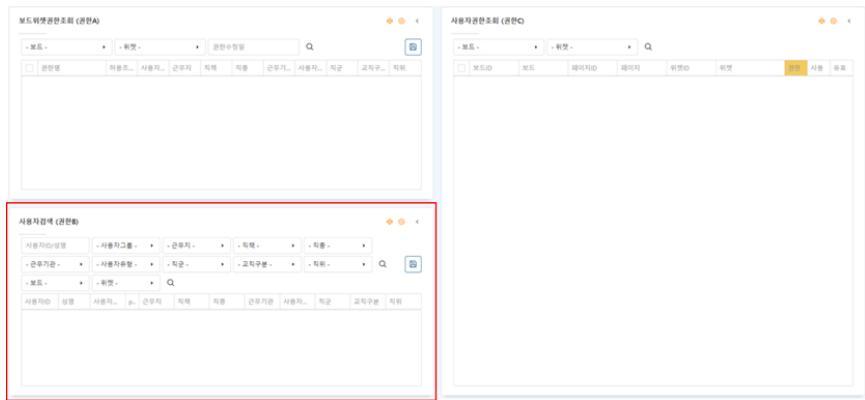
기존에 등록되어 있는 권한을 이용하여 (권한C) 위젯과 함께 새로운 권한을 추가할 수 있다.



▼ 사용자검색(권한B)

사용자ID, 사용자명, 사용자그룹, 근무지, 직책 등의 검색 조건으로 사용자를 조회한다.

(권한C) 위젯과 함께 사용하여 새로운 권한을 추가할 수 있다.



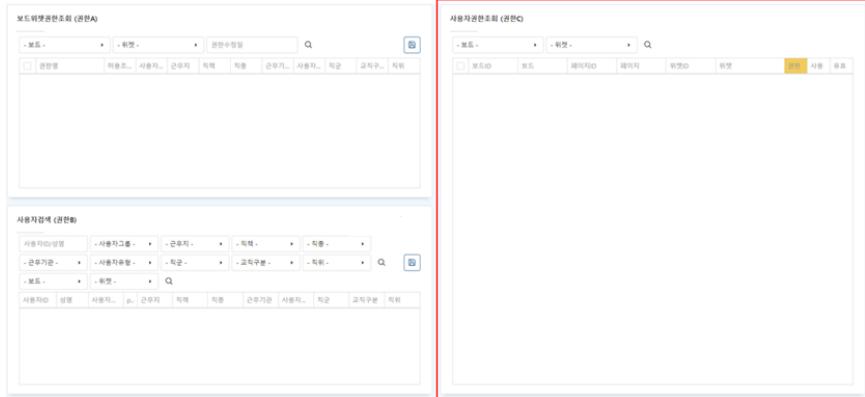
▼ 사용자권한조회(권한C)

전체 보드, 위젯을 조회할 수 있다.

(권한A) 위젯과 (권한B) 위젯과 함께 이용하여 권한을 조회하고 추가할 수 있다.

(권한A) 위젯과 (권한B) 위젯에서 조회한 내용에 대한 권한을 상세하게 확인할 수 있다.

! (권한A) 위젯 조회 내용 중 '사용자그룹'이 포함된 권한은 (권한C) 위젯에서 권한 상세 조회가 불가 하다.



▼ 기본동작

▼ 보드위젯권한조회(권한A)

- 권한 조회

보드위젯권한조회(권한A) 위젯 - 권한 조회

- ① 조회하고자 하는 보드 혹은 위젯 선택 후 검색 아이콘 클릭한다.
- ② 1의 검색 결과 중 조회하고자 하는 권한의 행을 더블클릭하여 해당 권한을 (권한C) 위젯에서 상세히 확인한다.
- 이 때 원하는 보드와 위젯만 조회하려면 (권한C) 위젯의 보드, 위젯 검색조건을 설정한 후 더블 클릭 한다.

주의 권한관리 페이지의 모든 검색 조건은 이름과 아이디로 검색이 가능하다.
검색 결과 중 '사용자그룹'이 포함된 권한은 (권한C) 위젯에서 권한 상세 조회가 불가 하다.

• 권한 추가

보드위젯권한조회(권한A) - 권한 추가

- ① (권한C) 위젯에서 권한을 부여하고자 하는 보드, 위젯에 체크한다.
- ② (권한A) 위젯에서 등록하고자 하는 권한에 체크한다.
- ③ (권한A) 위젯의 저장 아이콘을 클릭한다.



미등록 항목의 위젯에 권한 추가 시 유의사항

- 보드 미등록 위젯과 보드 등록 위젯을 같이 체크하여 권한을 추가할 수 없다.
일괄 추가를 원할 시 보드 미등록 혹은 등록끼리 디중 체크하여 저장 가능하다.
- 보드 미등록 위젯에 권한 추가 시 반드시 보드를 선택하여 권한 추가가 가능하다.
해당 위젯을 체크한 후 저장 버튼 클릭 시 '보드 일괄 등록' 팝업이 뜬다. 보드 선택 후 저장한다.

▼ 사용자검색(권한B)

- 사용자 조회

보드ID	보드	페이지ID	페이지	위젯ID	위젯	권한	사용	유효
BKdAaJToAr	경영현황	PPQqZPhYwex	외래	WEprymgug	월별 외래실적(인원...)	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>
BKdAaJToAr	경영현황	PPQqZPhYwex	외래	WwYHYIEZI	요일별 외래환자 현황	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>
BKdAaJToAr	경영현황	PPQqZPhYwex	외래	WzGfeVKKX	진료과별 외래환자...	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>
BKdAaJToAr	경영현황	PPQqZPhYwex	외래	WZSygwId0	진료과별 외래환자 현황	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>
BKdAaJToAr	경영현황	PPQqZPhYwex	외래	WhyQzITPu	협력회원사 작성 현황	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>
BKdAaJToAr	경영현황	PPQqZPhYwex	외래	WzGfeVKKX	진료과별 외래환자...	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>

사용자검색(권한B) - 사용자 조회

① 조회하고자 하는 사용자 검색 조건 혹은 보드, 위젯을 지정한 후 조회 아이콘 클릭한다.

(사용자 관련 조건 검색 시 직위 옆 검색 버튼, 보드 위젯별 검색 시 위젯 옆 검색 버튼을 클릭)

② 1의 검색 결과 중 조회하고자 하는 사용자의 행을 더블클릭하면 (권한 C) 위젯에서 상세히 확인할 수 있다.

(권한C) 위젯의 권한 컬럼의 체크 여부로 해당 사용자의 권한 유무를 알 수 있다.

이 때 원하는 보드와 위젯만 조회하길 원한다면 (권한C) 위젯의 보드와 위젯 콤보를 설정한 후 더블클릭한다.

- 권한 추가

사용자검색(권한B)

사용자ID/성명 개발자 그룹 중앙대학교병원 직책 직종

- 근무기관 - 가상사용자 - 직군 - 교직구분 - 직위 -

- 보드 - - 위젯 - Q

사용자ID 성명 사용자... p... 근무지 직책 직종 근무기관 사용자... 직군 교직구분 직위

사용자권한조회(권한C)

경영현황 - 위젯 - Q

보드ID	보드	페이지ID	페이지	위젯ID	위젯	권한	사용	유효
<input checked="" type="checkbox"/> BKbAaToAr	경영현황	PPQqZPhYwx	외래	WErppymug	월별 외래실적(인원...)	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> BKbAaToAr	경영현황	PPQqZPhYwx	외래	WwYHtYIEZI	요일별 외래환자 현황	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> AAjToAr	경영현황	PPQqZPhYwx	외래	WsGfeVKKSX	진료과별 외래환자...	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> BKbAaToAr	경영현황	PPQqZPhYwx	외래	WZ5ygw4d0	진료과별 외래환자...	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> BKbAaToAr	경영현황	PPQqZPhYwx	외래	WhyQzITPu	협력회신서 작성 현황	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

사용자검색(권한B) - 권한 추가

- ① (권한C) 위젯에서 권한을 추가할 보드, 위젯을 체크한다.
- ② (권한B) 위젯의 사용자 검색항목에서 권한을 부여할 사용자의 조건을 지정한다.
- ③ (권한B) 위젯의 저장 아이콘을 클릭한다.

! 저장 이후 해당 조건의 사용자를 조회 후 더블클릭 시 (권한C)에서 권한을 추가한 보드, 위젯의 권한 컬럼에 체크 표시로 권한 부여 확인할 수 있다.

▼ 사용자권한조회(권한C)

- 권한 조회

사용자권한조회(권한C)

보드 - 위젯 - Q

Search

보드ID	보드	페이지ID	페이지	위젯ID	위젯	권한	사용	유효
<input checked="" type="checkbox"/> BKbAaToAr	경영현황	PPQqZPhYwx	외래	WErppymug	월별 외래실적(인원...)	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> BKbAaToAr	경영현황	PPQqZPhYwx	외래	WwYHtYIEZI	요일별 외래환자 현황	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> BKbAaToAr	경영현황	PPQqZPhYwx	외래	WsGfeVKKSX	진료과별 외래환자...	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> BKbAaToAr	경영현황	PPQqZPhYwx	외래	WZ5ygw4d0	진료과별 외래환자...	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> BKbAaToAr	경영현황	PPQqZPhYwx	외래	WhyQzITPu	협력회신서 작성 현황	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> BKbAaToAr	경영현황	PPQqZPhYwx	외래	WsGfeVKKSX	진료과별 외래환자...	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> BKbAaToAr	경영현황	PPQqZPhYwx	외래	WyknKTGYI	당일 환자현황	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

사용자권한조회(권한C) - 권한 조회

! 전체 조회 시 다른 동작 없이 조회 아이콘을 클릭한다.
빠른 로딩을 위해 보드 또는 위젯 중 하나 선택하는 것을 권장한다.

- 권한, 사용, 유효 컬럼의 체크 선택/해제 시의 동작

체크	권한	사용	유효
선택	위젯열림 권한 있는 상태	개인별 페이지에 위젯 추가 된 상태 (개인별 페이지가 없는 보드는 표시되지 않음)	현재 날짜를 기준으로 사용유효한 위젯

체크	권한	사용	유효
해제	위젯열림 권한 없는 상태	개인별 페이지에 위젯 없는 상태	현재날짜를 기준으로 유효하지 않은 위젯

▼ 예시 케이스

▼ 직책으로 보드단위 권한추가

EX) 경영현황 보드에 중앙대의료원장 직책을 추가하는 경우

- 보드 및 직책 정보

보드	직책
경영현황	중앙대의료원장

- (권한C) 위젯에서 경영현황 ① 보드를 조회 및 검색결과에서 권한 추가할 ② 위젯을 체크선택(해당 예시의 경우 전체 추가)
- (권한B) 위젯에서 해당 사용자로 ③ 검색조건을 설정하고, ④ 저장 아이콘을 클릭
- 저장 완료 후 해당 ⑤ 사용자를 조회하여 권한 추가를 확인
- 검색결과 ⑥ 더블클릭 후, (권한C) 위젯의 ⑦ “권한” 컬럼에 체크된 것을 확인

This screenshot shows the 'Board Permission Search (All)' interface. At the top, there's a search bar with filters for 'Board' (Operating Status) and 'Role' (Central Medical Center Director). Below the search bar is a table titled 'Board Permission List (All)' containing a list of roles and their corresponding permissions. A red box highlights the 'Operating Status' board column, and a red circle with number 2 indicates the 'Check Selection' action being performed.

1. (권한C) 위젯에서 경영현황 보드를 조회 ① 및 검색결과에서 권한 추가할 위젯을 체크선택 ②(해당 예시의 경우 전체 추가)

This screenshot shows the 'Board Permission Search (All)' interface after performing the search. It displays a list of users with their roles and permissions. A red box highlights the 'Operating Status' board column, and a red circle with number 3 indicates the 'Search Condition Setting' action being performed. Another red circle with number 4 indicates the 'Save Icon Click' action being performed.

2. (권한B) 위젯에서 해당 사용자로 검색조건을 설정 ③ 하고, 저장 아이콘을 클릭 ④

3. 저장 완료 후 해당 사용자를 조회 ⑤ 하여 권한 추가를 확인

4. 검색결과 ⑤ 를 더블클릭 ⑥ 후, (권한C) 위젯의 “권한” 컬럼에 체크된 것을 확인 ⑦

▼ 보드로 보드단위 권한추가

EX) 경영현황 보드의 권한을 EDW 보드에 추가하는 경우

- 보드 정보

보드1	보드2
경영현황	EDW

- (권한C) 위젯에서 EDW ① 보드를 조회 및 권한 추가할 ② 위젯을 체크선택 (해당 예시의 경우 전체 추가)
- (권한A) 위젯에서 경영현황 ③ 보드를 조회 및 추가할 ④ 권한을 체크선택 후, (권한A) 위젯의 ⑤ 저장 아이콘을 클릭
- 저장 완료 후 해당 ⑥ 행을 더블클릭 하여 권한 추가를 확인 후, (권한C) 위젯의 ⑦ “권한” 컬럼에 체크선택 확인



예외적으로 권한 조건에 사용자그룹이 포함되었기 때문에 (권한C) 위젯에서 권한 유무를 확인할 수 없다.

사용자그룹이 없는 권한은 더블클릭으로 확인 가능하다.

1. (권한C) 위젯에서 EDW 보드를 조회 ① 및 권한 추가할 위젯을 체크선택 ② (해당 예시의 경우 전체 추가)

2. (권한A) 위젯에서 경영현황 보드를 조회③ 및 추가할 권한을 체크선택④ 후, (권한A) 위젯의 저장 아이콘을 클릭 ⑤

3. 저장 완료 후 해당 행을 더블클릭 ⑥ 하여 권한 추가를 확인 후, (권한C) 위젯에 '권한' 컬럼에 체크선택 확인 ⑦

▼ 사용자그룹 관리

| 사용자에게 부여할 보드, 위젯별 권한을 그룹화 하여 일괄로 관리한다.

- ZZ.ZNBMCODE** 참조

- 사용자그룹 조회 쿼리

```
SELECT *
FROM ZZ.ZNBMCODE
```

- 사용자그룹 조회 쿼리

```
SELECT *
FROM ZZ.ZNBMCODE
```

```

WHERE INST_CD = '기관코드'
AND GRUP_NB_CD LIKE 'UG_%'
AND NB_CD != '*'

```

```

WHERE INST_CD = '기관코드'
AND GRUP_NB_CD = '사용자그룹 그룹코드'
AND NB_CD != '*'

```

컬럼	설명
GRUP_NB_CD	사용자그룹 그룹코드
NB_CD	-
VALI_FRDD	유효시작일자
VALI_TODD	유효종료일자
NB_NM	사용자그룹명
SORT_NO	순서
NB_DESC	비고
NB_ATTR	-

컬럼	설명
GRUP_NB_CD	사용자그룹 그룹코드
NB_CD	사용자ID
VALI_FRDD	유효시작일자
VALI_TODD	유효종료일자
NB_NM	사용자명
SORT_NO	순서
NB_DESC	비고
NB_ATTR	-

• 사용자그룹, 사용자그룹 사용자관리 위젯

- 사용자그룹, 사용자그룹 내 사용자의 권한 조회, 추가, 수정 기능을 제공한다.

The screenshot displays two main sections of the application's user management interface:

- User Group Management (사용자그룹 관리):** This section shows a table of user groups. One row is visible, containing the group code "UG_DEV", name "개발자 그룹", start date "20190316", end date "99991231", sort order "0", and description "근무부서". Buttons for "추가" (Add) and "저장" (Save) are at the top right.
- User List Management (사용자그룹 사용자관리):** This section shows a table of users under a specific group. The table header includes columns for #, 유_ (User ID), 사용자_ (Name), 사용자_ (Name), 근무지_ (Workplace), 근무지_ (Workplace), 시작일_ (Start Date), 종료일_ (End Date), 비고 (Notes), 최종... (Last Update), 최종수정... (Last Update), 최종... (Last Update), and 최종등록... (Last Registration). A search bar at the top says "사유를 선택하세요." (Select reason). A button for "사용자 추가" (Add User) and a "저장" (Save) button are also present. The message "No data" is displayed in the center of the table area.

사용자그룹 관리, 사용자그룹 사용자관리 위젯

▼ 위젯소개

- 사용자그룹 관리

| 사용자그룹을 조회 □ 추가 □ 수정 할 수 있다.

사용자그룹 관리

그룹코드	그룹명	시작일자	종료일자	순서	비고	최종수...	최종수정일시	최초등...	최초
UG_DEV	개발자 그룹	20190316	99991231	0	근무부서		2019-03-16 14:...		2019

사용자그룹 관리 위젯

• 사용자그룹 사용자관리

| 선택한 사용자그룹의 사용자를 조회 · 추가 · 수정할 수 있다.

사용자그룹 사용자관리

사유를 선택하세요.										사용자 추가	저장	
#	유. 사용자...	사용자	근무지...	근무지	시작일...	종료일...	비고	최종...	최종수정...	최초...	최초등록...	
No data												

사용자그룹 사용자관리 위젯

▼ 기본동작

▼ 사용자그룹 관리

사용자그룹 관리

그룹코드	그룹명	시작일자	종료일자	↑	순서	비고	최종수...	최종수정일시	최초등...	최초
UG_DEV	개발자 그룹	20190316	99991231		0	근무부서		2019-03-16 14:...		2019
				1					2	4
UG_DEV	개발자 그룹	20190316	99991231		0	근무부서		2019-03-16 14:...		2019
UG_3kjm91p		20220718	99991231		0				3	

사용자그룹 관리 위젯 - 기본동작

| ① 사용자그룹 데이터 영역에서 수정할 셀을 클릭하고 수정한다.

⚠ “최조등록자”, “최조등록일시”, “최종등록자”, “최종등록일시” 컬럼 데이터는 수정 불가능 (저장 시 자동업데이트)

- ② 사용자그룹 추가 시 추가 버튼을 클릭한다.
- ③ 추가된 행에 사용자그룹 정보를 입력한다.
- ④ 저장 버튼을 클릭하여 추가 및 수정한 사용자그룹 정보를 저장한다.

▼ 사용자그룹 사용자관리

사용자그룹 사용자관리 위젯 - 기본동작

- ① 사용자그룹을 선택하여 해당 사용자그룹에 속한 사용자를 조회한다.
- ② 조회 결과 행 중에 수정하고자 하는 셀을 클릭하여 수정한다.

⚠ “비고”, “시작일자”, “종료일자” 컬럼 데이터만 수정 가능

- ③ 사용자 추가 시 “사용자 추가” 버튼을 클릭한다.
- ④ ③의 동작으로 오픈된 팝업에서 사용자 검색 후, 조회된 행 더블클릭하여 사용자를 추가한다.
- ⑤ “저장” 버튼을 클릭하여 추가 및 수정한 사용자그룹 사용자 정보를 저장한다.

▼ 2. 보드 관리

▼ 2.1. 마스터

▼ (1) 개요

마스터 관리를 통해 보드, 페이지, 공통코드 등을 편리하게 관리할 수 있다.

“마스터관리” 페이지의 Master Manager 위젯으로 관리한다. ([ZZ_ZNBMNBRD](#) 테이블 참조)



보드 마스터 관리와 인트로 보드 목록

보드 마스터 관리는 **ZZ.ZNBMNBRD** 테이블에 저장된 모든 보드를 조회한다. (# 인트로 화면 보드 목록)

만약 보드 마스터 관리로 보드를 추가했다면,

보드 마스터 관리에는 조회 되지만 인트로 화면의 보드 목록에는 나타나지 않는다.

사용자 권한 별로 보드의 열람 범위를 제한하고 있기 때문이므로, 추가한 보드에 사용자 권한을 부여해 주어야 한다.

(사용자별 권한 부여 방법은 [2.2. 인트로 보드](#) 부분을 참고)

- Master Table

| 보드 마스터 테이블로 보드를 추가 및 수정한다.

- ZZ.ZNBMNBRD** 테이블 참조

컬럼명	컬럼 설명	데이터형 및 기본값	NULLABLE	비고
NBRD_ID	보드 ID ('B' + 랜덤 문자 9글자)	VARCHAR2	NOT NULL	
VALI_FRDD	유효시작일	VARCHAR2	NOT NULL	
VALI_TODD	유효종료일	VARCHAR2	NOT NULL	
NBRD_NM	보드명	VARCHAR2		
NBRD_TYPE	보드유형	VARCHAR2		아래 세부 항목 참고
MENU_TYPE	메뉴유형	VARCHAR2		아래 세부 항목 참고
WIDGET_EDIT_YN	위젯편집가능여부	VARCHAR2 Y		아래 세부 항목 참고

- 편집 조건 중 세부 항목 (1) - 보드유형

컬럼명	설명	종류	종류설명	비고
NBRD_TYPE	보드유형			ZZ.ZNBMCODE 참조 (GRUP_NB_CD: "NBRDTYPE")
		DEFAULT	특정 목적이 아닌 일반적인 보드 구성 시 적용하는 유형	
		EIS	EIS 프로세스를 위한 별개의 보드 유형	
		LINK	외부 링크를 적용할 수 있는 유형	아래 추가 사항 참조
		* PATIENT		구버전 보드 적용 사항

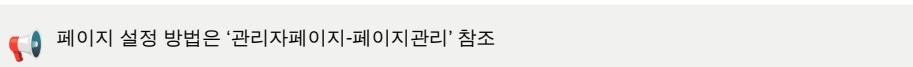


보드 유형이 **LINK** 일 경우 (**ZZ.ZNBMCODE** 테이블 참조 ⇒ GRUP_NB_CD : "EXTERNAL_LINK_URL")

```
SELECT * FROM ZZ.ZNBMCODE WHERE GRUP_NB_CD = 'EXTERNAL_LINK_URL' ;
```

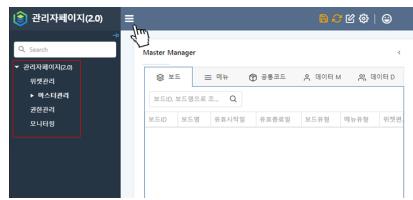
컬럼	입력 데이터
GRUP_NB_CD	"EXTERNAL_LINK_URL" 고정
NB_CD	보드ID
VALI_FRDD	유효시작일
VALI_TODD	유효종료일
NB_NM	URL 이름
NB_DESC	URL 설명
NB_ATTR	URL^key

◦ 편집 조건 중 세부 항목 (2) - 메뉴유형



컬럼명	설명	종류	종류설명	비고
MENU_TYPE	메뉴유형			ZZ.ZNBMCODE 참조 (GRUP_NB_CD: "MENUTYPE")
		FIX	고정형	Default 여러 페이지로 보드 구성하는 유형 (계층구조 설정 가능)
		NONE	메뉴없음	하나의 페이지로 보드 구성하는 유형
		USER	-	준비 중
		TAB	-	준비 중

- FIX : 관리자가 설정한 메뉴이므로 사용자 개인 설정은 불가
- NONE : FIX과 다르게 페이지 상단 아이콘 중 '메뉴' 버튼 없음



신 버전 보드_페이지 구성(FIX)



신 버전 보드_페이지 구성(NONE)

◦ 편집 조건 중 세부 항목 (3) - 위젯편집가능여부

- 개인화 페이지 아이콘 : 페이지 우측 상단 아이콘 모음에서 흰색 페이지 편집 아이콘

컬럼명	설명	종류	종류설명	비고
WIDGET_EDIT_YN	위젯편집 가능여부			개인화 페이지 설정 가능 여부 (관리자가 설정하는 기본 페이지와는 별개)
		Y	가능	사용자별 페이지 내 위젯 추가/삭제, 위젯 위치/크기 수정 가능
		N	불가능	관리자일시, 기본페이지 설정만 가능

- Y :

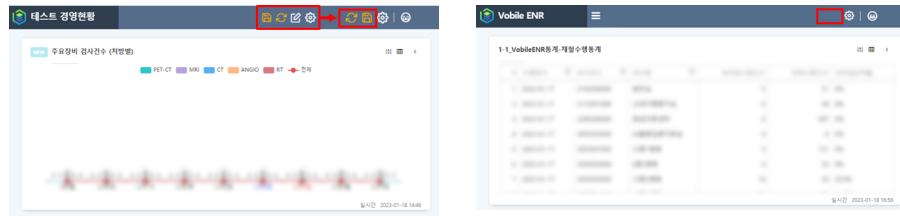


관리자의 경우, 페이지 편집 아이콘 클릭 시 개인화 페이지 저장 가능



일반사용자의 경우, 개인화 페이지 편집 아이콘 있음

- N :



관리자의 경우, 페이지 편집 아이콘 클릭 시 개인화 페이지 저장 불가

일반사용자의 경우, 개인화 페이지 편집 아이콘 없음



관리자 아이콘 (노란색)에 관한 안내는 '페이지 관리 - 3.2. 페이지 편집' 참조

▼ (2) Master Manager - 보드

보드를 조회하고 신규 생성, 수정 저장을 할 수 있다.

▼ 보드 조회

보드ID	보드명	유료 시작일	유료 종료일	보드 유형	예상 유형	비밀번호	최초 등록일시	최종수정일시
보드 ID, 보드명으로 조회합니다								
Ekd4eb18q	EMR View	20210917	99991231	EMR	NONE	풀기	20901920	2021-09-17 09:21
BUSIBOARD	Myboard	20190816	99991231	DEFAULT	USER	풀기	NUBOAL...	2019-09-22 18:27
BtfrTku6	김종관관리	99991231	99991231	DEFAULT	FIX	NUBOAL...	2021-04-07 07:57	NUBOAL...
BksAaTf0e	경영현황	20190614	99991231	DEFAULT	FIX	풀기	NUBOAL...	2019-06-14 21:20
BADMINPAGE	관리자페이지	20202016	99991231	DEFAULT	FIX	가능	안전팀	2020-10-03 16:14
BADMINPAGE	관리자페이지OLD	99991231	99991231	DEFAULT	FIX	가능	NUBOAL...	2019-07-30 14:09
Bdboards	리포트 생성기	20220518	99991231	DEFAULT	FIX	풀기	NUBOAL...	0002-01-31 23:08
SENDREPORT	리포트 전송 관찰설정	20200522	99991231	-	-	-	NUBOAL...	2020-05-22 08:40
BTBIMMS45	별별현황	99991231	99991231	DEFAULT	TAB	가능	NUBOAL...	2019-05-14 16:28
BntTest001	상황실	99991231	99991231	DEFAULT	FIX	가능	NUBOAL...	2019-12-27 16:09
BQ76PhTLQ	의료활동평가	99991231	99991231	DEFAULT	FIX	가능	NUBOAL...	2020-09-29 08:11

- ① 좌측 상단의 검색창에 조회하고자 하는 '보드명' 혹은 '보드ID'를 입력 후 돋보기 버튼을 클릭한다. (포함되는 검색어 조회 가능)
 - ② 표에서 보드 목록을 확인할 수 있으며 특정 보드 클릭 시 하단의 품에서도 데이터 확인 가능하다.
- * 전체 보드 목록 조회 시 검색어 입력 없이 돋보기 버튼을 클릭한다.

▼ 보드 편집

▼ 기존 보드 수정

보드ID	보드명	유효시작일	유효종료일	보드유형	메뉴유형	위젯판...	최초등록일시	최종수...	최종수정일시
B00eb1b8q	EMR View	20210917	99991231	EMR	NONE	불가	20901920 2021-09-17 09:21	20901920	2021-09-17 09:21
USERBOARD	Myboard	20190816	99991231	DEFAULT	USER	불가	NUBOA.. 2019-08-22 18:27	NUBOA..	2019-08-22 18:27
B00f7KXu6	김영준리	99991231	99991231	DEFAULT	RX	가능	NUBOA.. 2021-04-07 07:57	NUBOA..	2021-04-07 07:57
E0KAa1SuR	경영현황	20190614	99991231	DEFAULT	RX	불가	NUBOA.. 2019-06-14 21:20	NUBOA..	2022-07-13 17:45
BADMINPAGE	관리자페이지	20202016	99991231	DEFAULT	RX	가능	안치원 2022-12-02 16:14	안치원	2022-07-15 09:46
E0gBwBoards	관리자페이지OLD	99991231	99991231	DEFAULT	RX	가능	NUBOA.. 2019-07-30 14:09	NUBOA..	2022-07-15 11:37
SENDEREPORT	리포트_상용기	20220518	99991231	DEFAULT	RX	불가	NUBOA.. 0002-01-31 23:00	NUBOA..	0002-01-31 23:00
BTBNMMS45	병원현황	99991231	99991231	DEFAULT	TAB	가능	NUBOA.. 2019-06-14 16:28	NUBOA..	2019-06-14 16:28
BnITest001	상황실	99991231	99991231	DEFAULT	RX	가능	NUBOA.. 2019-12-27 16:09	NUBOA..	2022-08-04 17:26
BQ76PhnUkQ	의료질평가	99991231	99991231	DEFAULT	RX	가능	NUBOA.. 2020-09-29 08:11	NUBOA..	2020-09-29 08:11

보드ID: B00eb1b8q
유효시작일: 2021-09-17
보드유형: EMR
위젯판기능여부: 가능 (불가)

보드명: EMR View
유효종료일: 9999-12-31
메뉴유형: NONE

추가 (1) ①
저장 (2) ②
닫기 (3) ③

① 내용을 변경할 보드 클릭하여 아래 품에서 보드를 수정한다.

② 수정 완료 시 저장 버튼을 클릭한다.



위젯 내 보드ID 수정 불가

▼ 신규 보드 생성

보드ID	보드명	유효시작일	유효종료일	보드유형	메뉴유형	위젯판...	최초등록일시	최종수...	최종수정일시
B00eb1b8q	EMR View	20210917	99991231	EMR	NONE	불가	20901920 2021-09-17 09:21	20901920	2021-09-17 09:21
USERBOARD	Myboard	20190816	99991231	DEFAULT	USER	불가	NUBOA.. 2019-08-22 18:27	NUBOA..	2019-08-22 18:27
B00f7KXu6	김영준리	99991231	99991231	DEFAULT	RX	가능	NUBOA.. 2021-04-07 07:57	NUBOA..	2021-04-07 07:57
E0KAa1SuR	경영현황	20190614	99991231	DEFAULT	RX	불가	NUBOA.. 2019-06-14 21:20	NUBOA..	2022-07-13 17:45
BADMINPAGE	관리자페이지	20202016	99991231	DEFAULT	RX	가능	안치원 2022-12-02 16:14	안치원	2022-07-15 09:46
E0gBwBoards	관리자페이지OLD	99991231	99991231	DEFAULT	RX	가능	NUBOA.. 2019-07-30 14:09	NUBOA..	2022-07-15 11:37
SENDEREPORT	리포트_상용기	20220518	99991231	DEFAULT	RX	불가	NUBOA.. 0002-01-31 23:00	NUBOA..	0002-01-31 23:00
BTBNMMS45	병원현황	99991231	99991231	DEFAULT	TAB	가능	NUBOA.. 2019-06-14 16:28	NUBOA..	2019-06-14 16:28
BnITest001	상황실	99991231	99991231	DEFAULT	RX	가능	NUBOA.. 2019-12-27 16:09	NUBOA..	2022-08-04 17:26
BQ76PhnUkQ	의료질평가	99991231	99991231	DEFAULT	RX	가능	NUBOA.. 2020-09-29 08:11	NUBOA..	2020-09-29 08:11

보드ID: B00gkjh246
유효시작일: 2022-09-05
보드유형: DEFAULT
위젯판기능여부: 가능 (불가)

보드명: ① 추가 버튼 (1)
유효종료일: 9999-12-31
메뉴유형: RX
저장 (2) ②
닫기 (3) ③
신규추가

① 추가 버튼 클릭하여 새로운 품을 생성한다.

② 신규 생성할 보드 데이터를 작성한다.

③ 작성 완료 시 저장 버튼을 클릭한다.

▼ 2.2. 인트로 보드

▼ (1) 개요

로그인 이후 인트로 페이지에서 확인 가능한 보드를 설정한다.

‘권한 관리’와 함께 관리자가 사용자 별 보드 열람 범위를 지정한다. (1.위젯관리 - 1.5. 권한 참고)



▼ (2) 인트로 설정

▼ 1. 권한 별 보드 설정



“관리자” 권한 추가 시, 추가사항

(아래의 ※ Master Manager로 인트로 보드 속성 설정 방법 참고하여 추가 가능)

- ZZ.ZNBMCODE 테이블 확인

1. SUPER 레벨 추가

```
SELECT *
FROM ZZ.ZNBMCODE
WHERE INST_CD = '기관코드'
AND GRUP_NB_CD = 'ADMIN_LEVEL'
```

컬럼	설명	예시
INST_CD	기관코드	015
GRUP_NB_CD	ADMIN_LEVEL (그룹코드)	ADMIN_LEVEL
NB_CD	사용자 ID	11122233
VALI_FRDD	유효시작일자	20230101
VALI_TODD	유효종료일자	99991231
NB_NM	SUPER(관리자등급)	SUPER
SORT_NO	순서	999
NB_DESC	사용자 이름	관리자1
NB_ATTR	테스트 위젯 ID	-

2. UG_DEV (한시적 개발자 그룹) 추가

한시적 개발자 그룹은 ‘사용자그룹’에 포함되는 권한으로 ‘선택사항’이다.

(사용자그룹에 관한 설명은 [위젯관리-권한 - 1.5.\(2\) 권한 관리 - 사용자그룹 관리](#) 참고)

UG_DEV 권한은 개발자 그룹으로 관리자페이지를 포함한 대부분의 권한이 부여되어있다.

UG_DEV 이외의 권한을 부여하고 싶을 시, 따로 권한을 부여해야한다.

```
SELECT *
FROM ZZ.ZNBMCODE
WHERE INST_CD = '기관코드'
AND GRUP_NB_CD = 'UG_DEV'
```

컬럼	설명	예시
INST_CD	기관코드	015
GRUP_NB_CD	UG_DEV (그룹코드)	UG_DEV
NB_CD	사용자 ID	11122233
VALI_FRDD	유효시작일자	20230101
VALI_TODD	유효종료일자	99991231
NB_NM	사용자 이름	관리자1
SORT_NO	순서	999
NB_DESC	사용자 소속	정보운영팀
NB_ATTR	추가 정보	-

① 추가할 보드와 위젯의 정보 및 권한 확인

- 보드 정보 확인
 - 마스터관리 페이지 - Master Manager
 - ‘보드’ 탭 : 보드의 유효시작일과 유효종료일 확인

Master Manager								
보드	메뉴	공통코드	데이터 M	데이터 D				
보드ID, 보드명으로 조회합니다								
보드ID	보드명	유료시작일	유료종료일	보드유형	메뉴유형	위젯면...	최초등...	최초등록일시
Bk4leb10q	EMR View	20210917	99991231	MR	NONE	불가	20901920	2021-09-17 09:21
BUSERBOARD	MyBoard	20190816	99991231	DEFAULT	USER	불가	NUBOA...	2019-08-22 18:27
BdBR7TKU6	김영관리	99991231	99991231	DEFAULT	FIX	가능	NUBOA...	2021-04-07 07:57
BKQaauToAr	경영현황	20190614	99991231	DEFAULT	FIX	불가	NUBOA...	2019-06-14 21:20
BADMINPAGE	관리자페이지	20220216	99991231	DEFAULT	FIX	가능	안치원	2022-02-16 16:14
BADMINPAGE	관리자페이지OLD	99991231	99991231	DEFAULT	FIX	가능	NUBOA...	2019-07-30 14:09
BEdwBoard	리포트 생성기	20200518	99991231	DEFAULT	FIX	불가	0000-01-31 23:00	NUBOA...
SENDREPO	리포트전송 관찰설정	20200522	99991231	-	-	-	NUBOA...	2020-05-22 08:40
BTIRkH4s5	별첨인증	99991231	99991231	DEFAULT	TAB	가능	NUBOA...	2019-06-14 16:28
BnU7es001	상황실	99991231	99991231	DEFAULT	FIX	가능	NUBOA...	2022-08-04 17:26
8G76pnTLQ	의료질평가	99991231	99991231	DEFAULT	FIX	가능	NUBOA...	2020-09-29 08:11

• 위젯 정보 확인

◦ 위젯관리 페이지 - Widget Master

- '위젯' 탭 : 위젯의 유효시작일과 유효종료일 확인
- '권한' 탭 : 보드를 열람할 사용자의 권한 유무 확인 (반드시 하나 이상의 권한이 존재해야 한다.)

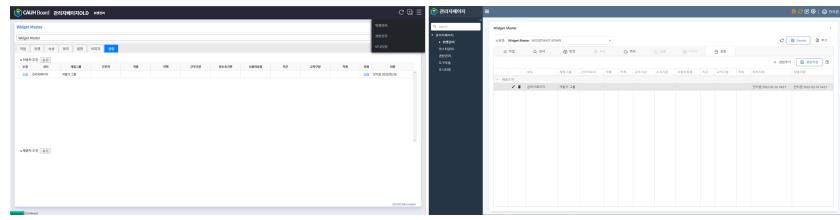
② 인트로 보드 속성 설정

- 인트로 페이지에서 확인할 수 있는 보드의 순서, 아이콘 등의 속성을 지정한다.
- `ZZ_ZNBMCODE` 테이블 또는 Master Manager 위젯의 '공통코드' 탭을 이용하여 인트로 보드 속성값을 설정한다.

컬럼명	설명	비고
GRUP_NB_CD	그룹코드	"INTRO_NBRD_PROP" 고정
NB_CD	보드ID	<code>ZZ_ZNBMNBRD</code> 참조
VALI_FRDD	유효시작일	
VALI_TODD	유효종료일	
NB_NM	보드명	<code>ZZ_ZNBMMNBRD</code> 참조
SORT_NO	보드 순서	인트로에서 보이는 보드의 순서 설정
NB_DESC	보드 버전	아래 추가 안내사항 참고
NB_ATTR	보드 아이콘	아래 추가 안내사항 참고

◦ 추가 안내사항

- 보드 버전 (`NB_DESC`)

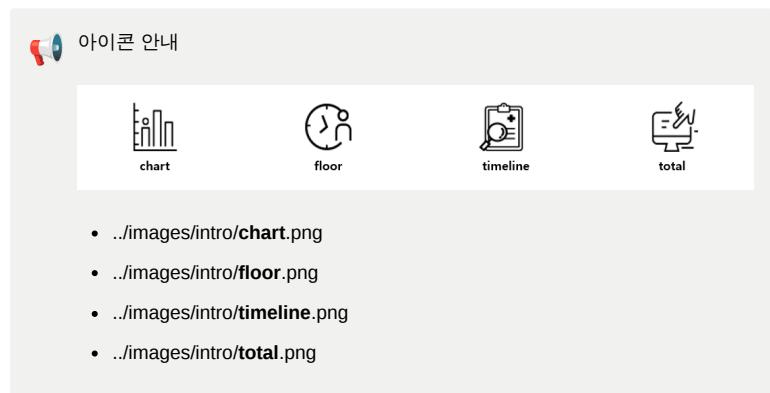


왼쪽이 구 버전 보드, 오른쪽이 신 버전 보드 화면

- 아래의 경로 그대로 `NB_DESC` 컬럼 값으로 설정 시, 신 버전으로 보드가 구성된다.
 - `../jsp/phisboardpage.jsp?bid=`

■ 보드 아이콘 (`NB_ATTR`)

- 아래 경로에 변경하고자 하는 이미지 아이콘 이름을 적용한다.
 - `../images/intro/_____png`



- 인트로 보드 속성 적용 예시

	예시1	예시2
그룹코드	INTRO_NBRD_PROP	INTRO_NBRD_PR...
보드ID	BADMINPAGE	BKbAaJToAr
유효시작일자	20220101	20220101
유효종료일자	99991231	99991231
보드명	관리자페이지	경영현황
보드 순서	1	2
보드 버전	<code>../jsp/phisboardpage.jsp?bid=</code>	<code>../jsp/phisboardpage.jsp?bid=</code>
보드 아이콘	<code>../images/intro/chart.png</code>	<code>../images/intro/total.png</code>

[인트로] 관리자페이지

[인트로] 경영현황

예시1과 예시2 설정에 따른 결과 화면
보드의 순서와 아이콘을 확인할 수 있다.



인트로 보드 속성을 설정하지 않고 “권한”만 추가해도 인트로 페이지에서 보드를 확인할 수 있다.

- 인트로 보드 속성 기본 형태

보드 버전	구 버전
보드 순서	9999
보드 아이콘	없음

▼ ※ Master Manager로 인트로 보드 속성 설정 방법

Master Manager 위젯의 ‘공통코드’ 탭을 이용하여 인트로 보드 속성 추가 및 편집 할 수 있다.

- 기본동작

The first screenshot shows the 'Common Code' tab of the Master Manager interface. A red box highlights the 'INTRO_NBRD_PROP' entry in the list, and a red circle labeled '1' points to the 'Edit' button next to it. The second screenshot shows the detailed view for 'INTRO_NBRD_PROP'. It lists various properties like '보드명' (Board Name), '코드설명' (Code Description), '시작일자' (Start Date), etc., with their values. A red box highlights the 'Edit' button at the top right, and a red circle labeled '4' points to it. A red circle labeled '2' points to the 'Delete' button. A red circle labeled '3' points to the 'Add Row' button.

① 그룹코드 중 “INTRO_NBRD_PROP”을 조회한다.

② 조회된 결과의 > 버튼을 클릭하여 하위 그리드를 열고, + 버튼을 클릭하여 행을 추가한다.

③ 새로 추가된 행에 데이터를 입력한다.

컬럼명	입력 데이터	컬럼명	입력 데이터
코드	보드ID	시작일자	유효시작일
코드명	보드명	종료일자	유효종료일
순서	보드 순서	코드속성	보드 아이콘
코드설명	보드 버전		

④ 데이터 편집 완료 시 저장 버튼을 클릭한다.

▼ 2. 보드 내 페이지 추가

Master Manager 위젯 혹은 [ZZ.ZNBMPAGE](#) 테이블을 이용하여 페이지를 추가한다.

3. 페이지관리 - 3.2 페이지 마스터 관리 참고

▼ 3. 페이지 내 위젯 추가

| 위젯갤러리 혹은 `ZZ.ZNBDPAGE` 테이블을 이용하여 페이지에 위젯을 추가한다.

 3. 페이지관리 - 3.3 페이지 별 편집 (1) User Table , (2).b 위젯갤러리 참고

▼ 2.3 보드 속성

| 보드의 화면, 동작 등을 데이터로 추가하여 구현한다.

`ZZ.ZNBMCDATA` 테이블 혹은 도구모음 페이지의 CDATA 관리 위젯을 이용한다.

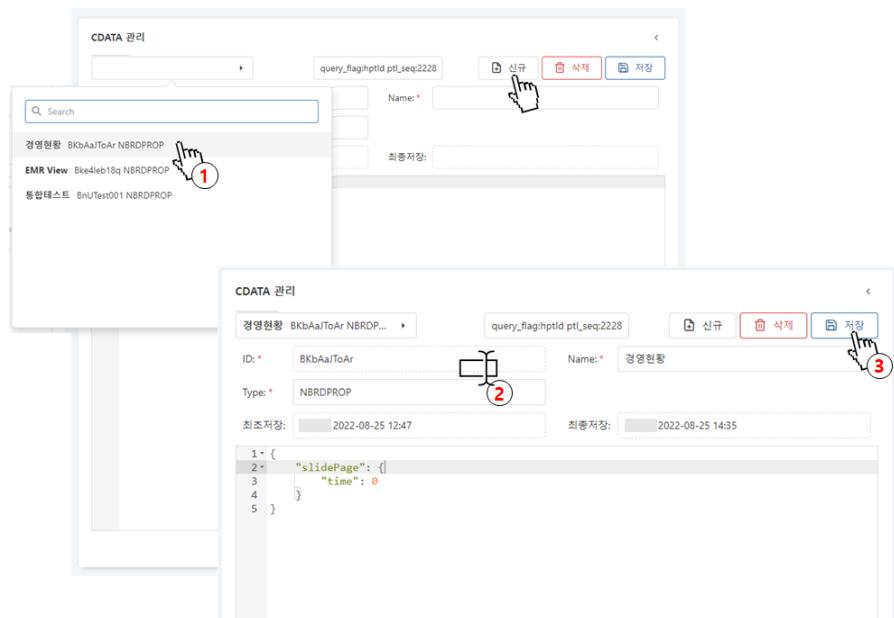
| 아래에서 다루는 속성은 신 버전으로 구성된 보드로 제한된다. (2.2. 인트로 보드 - (2)인트로 설정 참고)

▼ (1) 설정 방법

| 보드 속성은 CDATA 위젯을 이용하거나, `ZZ.ZNBMCDATA` 테이블에 데이터를 추가하여 설정할 수 있다.

▼ CDATA 위젯 관리

- CDATA 데이터를 신규 생성 및 수정한다.
 - 기본동작



① 기존 데이터 수정 시 콤보 박스에서 선택하고, 조회 / 신규 생성 시 우측의 신규 버튼 클릭한다.

② 품 내용과 CDATA를 입력한다.

항목	설명	항목	설명
ID	보드ID	Name	보드명
Type	"NBRDPROP" 고정	CDATA	보드 속성

③ 저장 버튼을 클릭한다.

▼ ZZ.ZNBMCMDA 테이블 데이터 추가

- 테이블 데이터로 직접 CDATA 를 추가 및 수정한다.

컬럼	데이터 타입	설명	예시
CDATA_ID	VARCHAR2	보드ID	BKbAaJToAr
CDATA_TYPE	VARCHAR2	"NBRDPROP" 고정	NBRDPROP
CDATA	CLOB	보드 속성 (데이터 형식 : Object)	{ "slidePage": { "time": 10 } }
CDATA_NM	VARCHAR2	보드명	경영현황

```
INSERT INTO ZZ.ZNBMCMDA
VALUES (
    '기관코드',
    '보드ID',
    'NBRDPROP',
    TO_CLOB('{ 보드속성 }'), '보드명', 'USERID', SYSTIMESTAMP, 'USERID', SYSTIMESTAMP);
```

▼ (2) 속성 종류

- ZZ.ZNBMCMDA 테이블의 CDATA 컬럼에 들어갈 보드 속성을 작성한다.

Key	데이터 타입	Value	설정값 데이터 타입	설명
"enableScreenLock"	Boolean	-	-	Session Timeout 동안 click 혹은 keydown 조작이 없을 때, 패스워드 재확인 화면이 나타난다.
"widgets"	Array	widget_id	String	페이지 상단 툴바에 추가할 위젯을 설정한다. (ex. EMR 진료상단정보 위젯)
		location	String	페이지 상단 툴바에 추가한 위젯의 위치를 설정한다.
"slidePage"	Object	time	Number	현재 페이지에서 다음 페이지로 슬라이드 될 시간을 설정한다.

○ 속성 Default 값

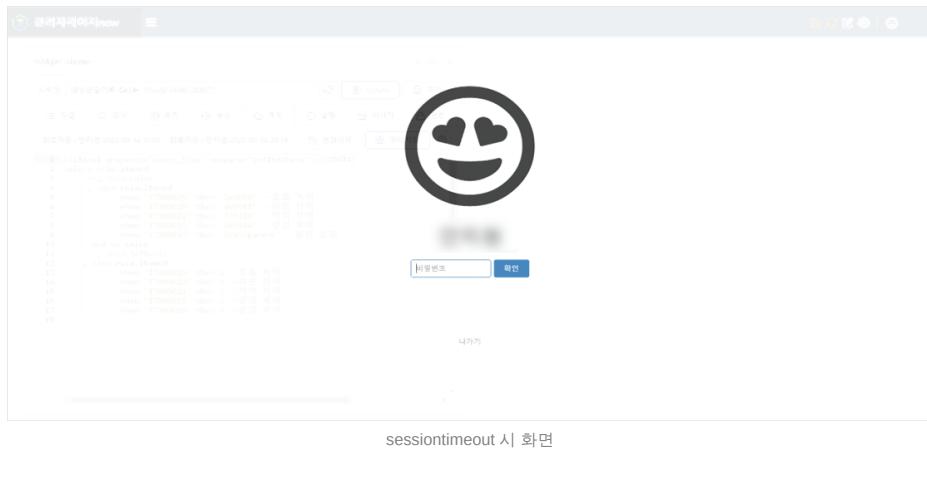
```
{
    "widgets": null,
    "enableScreenLock": false,
    "slidePage": {
        "time": 0
    }
}
```

▼ enableScreenLock

세션에 설정된 sessiontimeout 시간 동안 click 혹은 keydown 조작이 없을 때, 패스워드 재확인 화면 (ScreenLock)이 나타나도록 한다.

"enableScreenLock": true 일 때, ScreenLock이 가능하다.

```
{
    "enableScreenLock": true
}
```



sessiontimeout 시| 화면

! 설정된 timeout 시간은 cmc_config.xml 파일에서 확인할 수 있다.
본 단위로 설정되면 Default 값은 20분이다.
(...\\webhome\\app_home\\cmcnu\\WEB-INF\\classes\\resources\\cmc_config.xml)

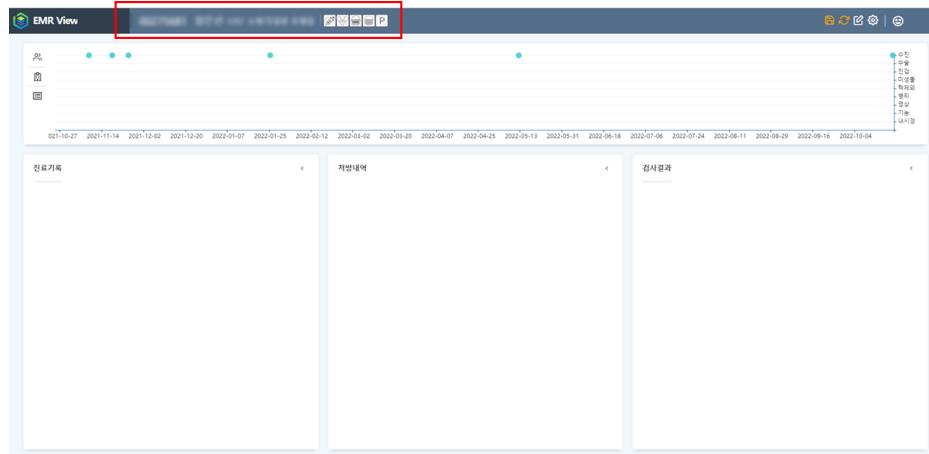
```
<entry key="session.timeout">10</entry> <!-- minute -->
```

▼ widgets

페이지 상단 Toolbar 공간에 위젯을 추가한다.

Key	Value	예시
"widget_id"	String	상단 Toolbar에 추가할 위젯ID "WbPVJ7fz7t"
"location"	String	추가할 위젯의 위치 topLeft topRight "topLeft"

```
{
    "widgets": [
        {
            "widget_id" : "WbPVJ7fz7t",
            "location": "topLeft"
        }
    ]
}
```



진료상단정보 위젯을 페이지 상단 Toolbar에 추가한 모습

▼ slidePage

설정 시간마다 현재 페이지 → 다음 페이지(`SORT_NO` 기준)로 페이지 이동한다.

“slidePage”의 설정값 “time”은 초 단위로 설정되며, 0초과 숫자로 설정한다.

Key	Value
“time”	Number 다음 페이지로 슬라이드 될 시간

```
{
  "slidePage": {
    "time": 10
  }
}
```

▼ 3. 페이지 관리

▼ 3.1. 마스터

▼ (1) 개요

페이지와 페이지 내 위젯을 편집할 수 있다.

`ZZ.ZNBMPAGE` 테이블 혹은 “마스터관리 페이지”의 Master Manager 위젯(MASTMANAGE)을 사용하여 페이지 마스터 관리를 할 수 있다.

`ZZ.ZNBDPAGE` 테이블 혹은 페이지 우측 상단 기능들을 통해 페이지별 위젯 관리를 할 수 있다.

- Master Table

페이지 마스터 테이블로 페이지를 추가 및 수정한다.

- `ZZ.ZNBMPAGE` 테이블 참조

컬럼명	설명	비고
PAGE_ID	페이지ID	
SUP_PAGE_ID	상위 페이지ID	<code>PAGE_ID</code> 와 계층 관계 / 메뉴의 단계(depth) 설정 가능 (4depth까지 권장)
NBRD_ID	보드ID	<code>ZZ.ZNBMNBRD</code> 참조
PAGE_NM	페이지명	
SORT_NM	표시순서	

- User Table

| 사용자 별 페이지 관리 테이블로 페이지 내 위젯을 사용자에 따라 추가 및 수정한다.

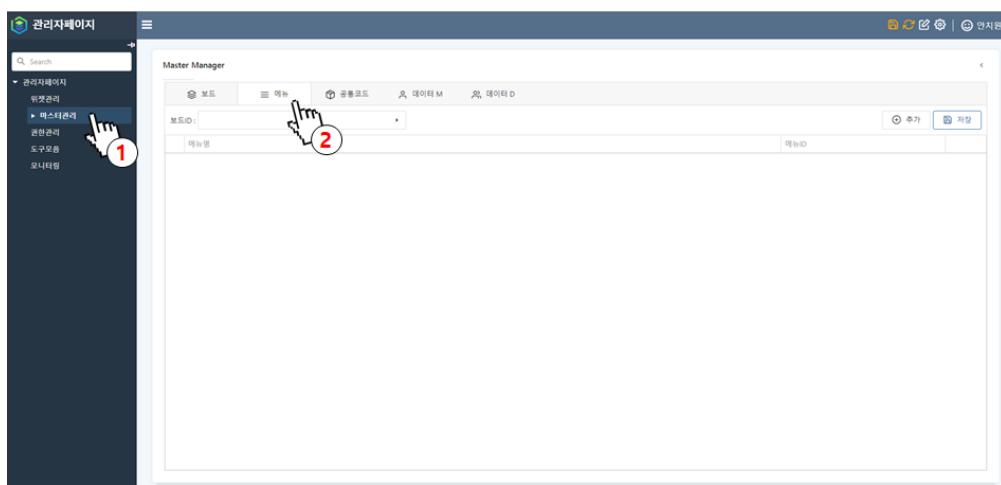
- **ZZ.ZNBDPAGE** 테이블 참조

컬럼	설명	비고
UPAGE_ID	사용자 페이지ID	
PAGE_ID	페이지ID	ZZ.ZNBMPAGE 참조
USER_ID	사용자ID	
WIDGET_ID	위젯ID	ZZ.ZNBMMWIDGET 참조
TITL_NM	타이틀명	위젯명 외에 각 위젯마다 타이틀 설정 가능
TITL_COLOR	타이틀색	
POS_X	위젯위치-X	
POS_Y	위젯위치-Y	
POS_W	위젯크기-가로	
POS_H	위젯크기-세로	
UWIDGET_PROP	사용자 위젯 설정	

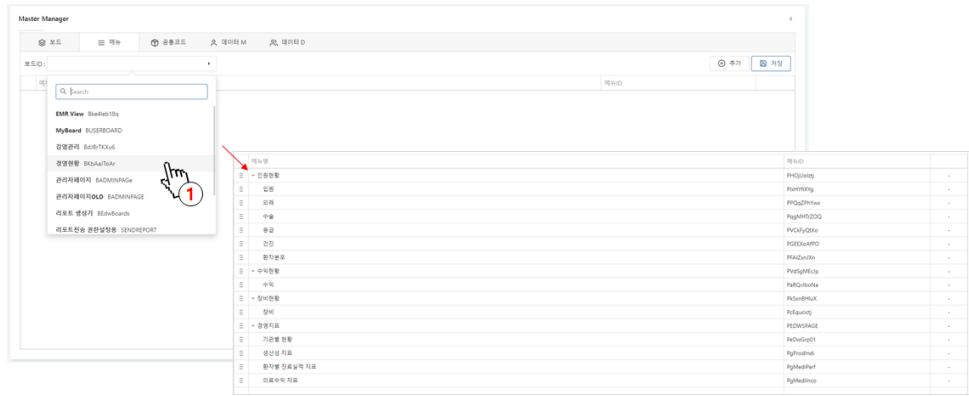
▼ (2) Master Manager - 메뉴

| 보드 별 페이지 목록을 조회하고 신규 생성, 수정 저장을 할 수 있다.

| 해당 위젯에서는 페이지를 “메뉴”로 지칭한다.



▼ 페이지 조회



① 보드ID 검색창에서 보드 선택한다. (보드ID 또는 보드명으로 검색 가능)

아래의 그리드에서 보드 별 페이지 조회 결과 확인

▼ 페이지 편집

▼ 기존 페이지 수정



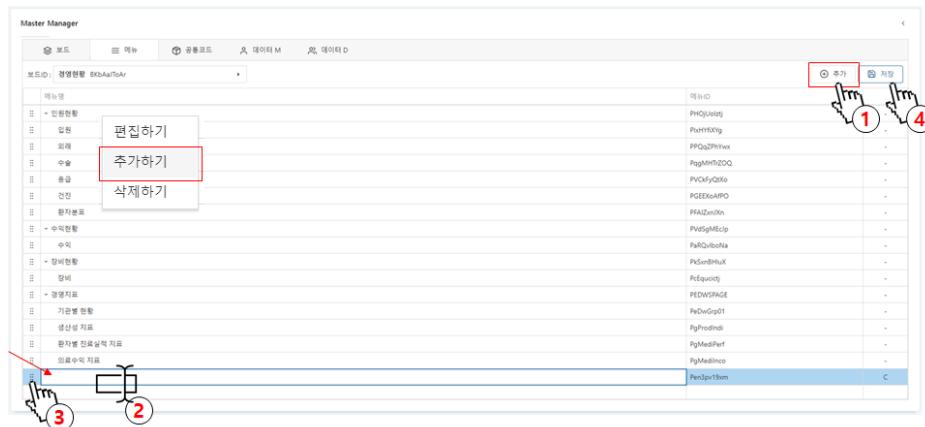
① 페이지 순서 변경 시 '메뉴명' 앞 셀에서 드래그 앤 드롭으로 이동한다.

② 편집할 페이지에서 우클릭 시 나타나는 “Context Menu”로 편집, 추가, 삭제한다.

③ ‘편집하기’ 클릭 시 메뉴명 셀이 수정 모드로 전환되고, 메뉴명을 수정한다. (위젯 내에서 메뉴ID는 수정불가)

④ 수정 완료 시 우측 상단의 저장 버튼을 클릭한다.

▼ 신규 페이지 생성

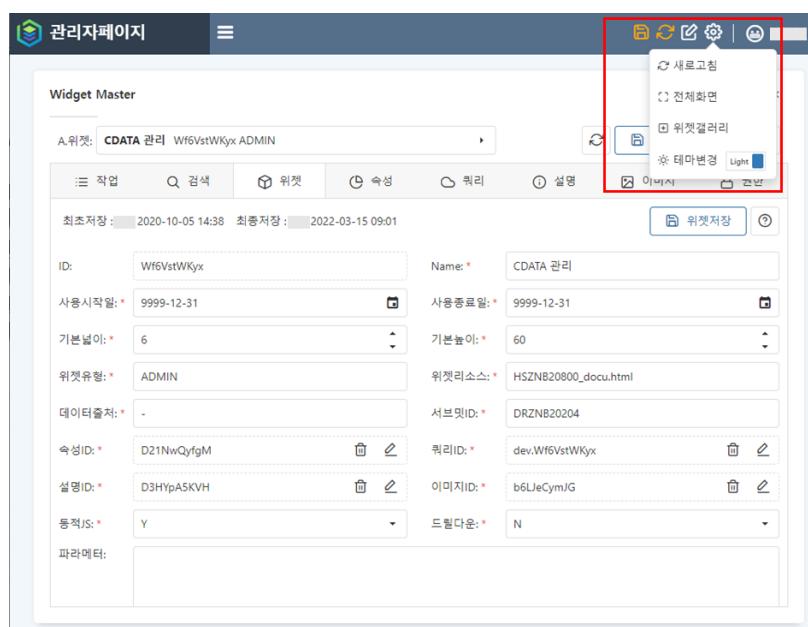


- ① 우측 상단 저장 버튼 옆 '추가' 버튼 클릭 혹은 그리드에서 우클릭 후 '추가하기'를 선택한다.
- ② 추가된 메뉴의 메뉴명을 입력한다.
- ③ 순서 변경 시 페이지 편집과 동일하게 드래그 앤 드롭으로 이동한다.
- ④ 추가 완료 시 저장 버튼을 클릭한다.

▼ 3.2. 페이지 편집

▼ (1) 개요

수정 □ 추가한 페이지에 사용자 권한에 따른 위젯을 추가하고 수정할 수 있다.
위젯갤러리, 테마 변경 등 페이지 설정 기능을 제공한다.



▼ (2) 페이지 툴바

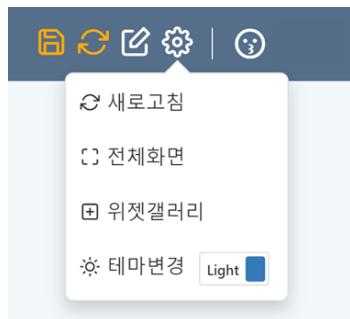
페이지 툴바(상단)의 아이콘을 활용해 간편하게 페이지 별 위젯을 관리할 수 있다.

관리자 권한 여부에 따라 기본 페이지 저장, 보드 속성 '위젯편집가능여부'에 따라 개인화 페이지 저장 가능여부가 결정된다.

! 주황색 아이콘은 관리자 ADMIN 권한 사용자에만 나타난다.
왼쪽부터 DEFAULT 페이지 설정, 강력 새로고침 순으로 기능을 제공한다.



▼ 1. 아이콘



페이지 저장

- 개인화페이지 저장
 - 사용자 별 페이지를 편집 ▶ 저장할 수 있다.
 - “페이지 편집” 아이콘을 먼저 클릭하면 “개인화페이지 저장” 아이콘으로 변경된다.
 - 편집 후 저장 아이콘을 클릭

기본페이지 저장 가능여부	
관리자	O
일반사용자	X

개인화페이지 저장 가능여부	
위젯편집 가능 (Y)	O
위젯편집 불가능 (N)	X

- 기본페이지 저장



- 관리자용 아이콘으로 일반 사용자에게는 보이지 않는다.

- 저장 전 “페이지 편집” 아이콘을 클릭하여 페이지 편집 모드로 전환 후, 저장 아이콘 클릭해야 한다.
- 보드 속성 중 “위젯편집 가능여부”가 “Y”일 때 (개인화 저장 가능), 기본페이지로 편집 저장하였을 경우 해당 위젯은 확인이 불가 하다.



ZZ.ZNBDPAGE 테이블에서 USERID가 “DEFAULT”로 저장된 데이터를 확인할 수 있지만 페이지에서는 사용자ID에 해당하는 위젯만 확인 가능하다.



새로고침

캐시 삭제를 위해 강력 새로고침 한다.



페이지 편집

페이지를 편집모드로 전환한다.

페이지 별 위젯의 크기나 위치 등을 편집할 수 있다.

페이지 저장 전 반드시 **페이지 편집모드로 전환** 후 저장해야 한다.

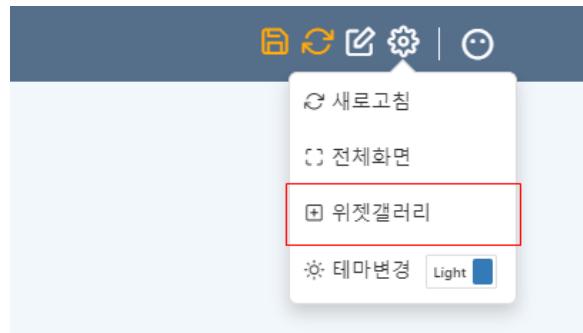


설정

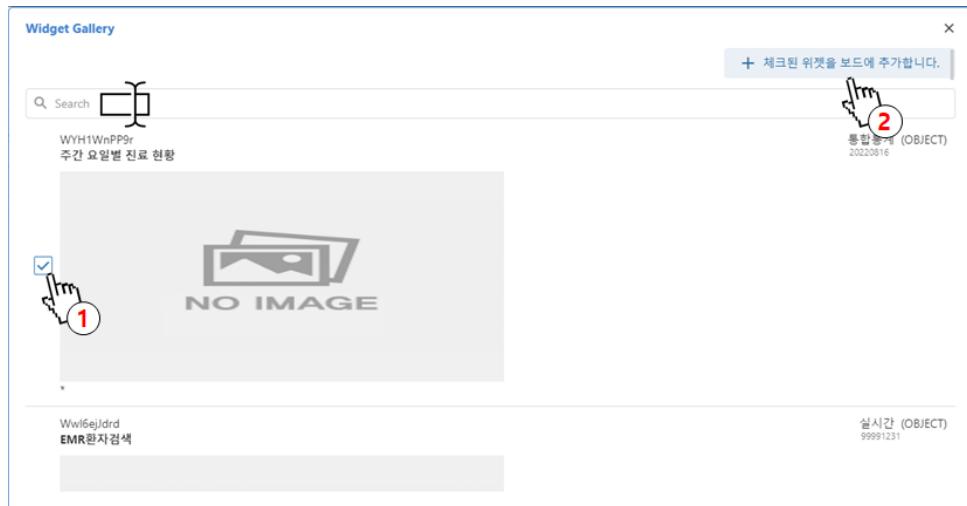
- 새로고침 : 관리자용 새로고침 아이콘과 동일 기능
- 전체화면 : 페이지를 전체화면으로 확인 가능 (F11과 동일 기능)
- 위젯갤러리 : 간편하게 해당 페이지에 위젯 추가하는 팝업 (위젯갤러리 참고)
- 테마변경 : Light 와 Dark 테마가 있으며 아이콘 클릭으로 자유롭게 변경 가능

▼ 2. 위젯갤러리

| 보드에 권한이 있는 위젯 목록을 확인하고 페이지에 추가한다.



페이지 우측 상단 '설정' 버튼 - 위젯갤러리 클릭



- ① 페이지에 추가할 위젯을 체크한다. (다중 체크 가능)
- ② 위젯 갤러리 팝업 우측 상단의 '체크된 위젯을 보드에 추가합니다.' 버튼을 클릭한다.



추가 안내사항

- 검색창을 통해 '위젯명' 혹은 '위젯아이디'를 검색하여 위젯을 찾을 수 있다.
- 위젯갤러리에서는 위젯 추가만 가능하다.
 - 외의 '타이틀명', '타이틀색', '사용자위젯설정'을 설정할 수는 없다.
 - 설정을 원할 시 `ZZ.ZNBDPAGE` 테이블에 직접 추가한다.