PAP-API

시작하기 전에

이 문서는 다보리 앱에서 개별 Table 단위의 데이터 처리를 위한 API 구조를 정의하는 방법을 설명하고 있습니다. PAP-API의 구조는 기본적으로 아래와 같습니다.

- **P**ick(단일 데이터 추출)
- **A**ct(데이터의 추가, 수정, 삭제)
- **P**age(다수의 row 추출)

API URL 네이밍 규칙

PAP-API 호출시 데이터를 사용하려는 해당 테이블명에 -pick, -act, -page를 붙여 공통적으로 api url을 네이밍하고 있습니다.

예를들어

- item 테이블에서 특정 상품 하나의 정보를 불러오고 싶다면 url은 item-pick
- member 테이블에서 특정 회원의 정보를 불러오고 싶다면 url은 member-pick
- item 테이블에 특정 상품을 추가, 삭제, 수정하고 싶다면 url은 item-act
- member 테이블에서 특정한 status(휴면회원)인 회원 page를 불러오고 싶다면 member-page
- member 테이블에 특정회원을 추가하고 싶다면 member-act (Id : 양수)
- member 테이블에 특정회원을 삭제하고 싶다면 member-act (Id : 음수)

pick-api 사용예제

쿼리의 WHERE 절에 유니크한 값인 Id로 조회하여 단일데이터 및 다수의 레코드를 추출할 수 있는 api입니다.

php

```
$itemPick = $this->callApiService->callApi([
  'url' => 'item-pick',
  'data' => [
    'Page' => [
                 ['UpdatedMd5' => $updatedMd,]
```

```
    ],
  ],
  'headers' => [
  'GateToken' => $this->gateToken['main']
  ]
]);
```

javascript

```
const response = await axios.post('/ajax/get-data', {
  url: 'item-pick',
  data: {
    Page: [
          { Id:  parseInt(window.User['SgroupId']) }
    ]
  }
});
```

act-api □□□□

Insert, Update, Delete□ □□ □□□ □□□ □□□□□. Id□ □□ □□ Insert(0), Update(□□), Delete(□□)□ □□□□□.

□□ Id□□ 0□□ □ □□□□ □□ Id□□ □□□□ □□□□ Update□□□.

Delete□ □□ □□ □□ ID □□□□ □□□ □ □□□□.

php

```
// insert
$itemAct = $this->callApiService->callApi([
          'url' => 'item-act',
          'data' => [
             'Page' => [
                [
                   'Id' => 0, // 0 : insert, □□: update, □□ : delete
                   'IgroupId' => 526,
```

```
                'ItemCode' => Str::limit($linkproMd5, 21, ''),
                'ItemSlug' => $linkproMd5,
                'ItemName' => $scrap['ItemName'],
                'SalesPrc' => (string)$scrap['SalesPrice'],
            ]
        ],
    ],
    'headers' => [
        'GateToken' => $this->gateToken['main']
    ]

// update
$itemAct = $this->callApiService->callApi([
    'url' => 'item-act',
    'data' => [
        'Page' => [
            [
                'Id' => 4, // 0 : insert, □□: update, □□ : delete
                'IgroupId' => 526,
                'ItemCode' => Str::limit($linkproMd5, 21, ''),
                'ItemSlug' => $linkproMd5,
                'ItemName' => $scrap['ItemName'],
                'SalesPrc' => (string)$scrap['SalesPrice'],
            ]
        ],
    ],
    'headers' => [
        'GateToken' => $this->gateToken['main']
    ]


// delete
$itemAct = $this->callApiService->callApi([
    'url' => 'item-act',
    'data' => [
        'Page' => [
            [ 'Id' => -4 ] // 0 : insert, □□: update, □□ : delete
        ],
    ],
    'headers' => [
```

```
            'GateToken' => $this->gateToken['main']
        ]
```

javascript

```
const response = await axios.post('/ajax/get-data', {
        url: 'item-act',
        data: {
          Id : 0,
          ItemCode: $(item_form).find('#item-code-txt').val(),
          IgroupId: Number($(item_form).find('#igroup-id-txt').data('id')),
          ItemName: $(item_form).find('#item-name-txt').val(),
          SubName: $(item_form).find('#sub-name-txt').val(),
          ItemSlug: $(item_form).find('#item-slug-txt').val(),
        }
});

const response = await axios.post('/ajax/get-data', {
        url: 'item-act',
        data: {
          Id : 3,
          ItemCode: $(item_form).find('#item-code-txt').val(),
          IgroupId: Number($(item_form).find('#igroup-id-txt').data('id')),
          ItemName: $(item_form).find('#item-name-txt').val(),
          SubName: $(item_form).find('#sub-name-txt').val(),
          ItemSlug: $(item_form).find('#item-slug-txt').val(),
        }
});

const response = await axios.post('/ajax/get-data', {
        url: 'item-act',
        data: {Id : -3}
});
```

page-api □□□□

□□□ SELECT □□□ □□□□ □□□□ 2□□ □□□ □□□ □□□□□.

Query, Asc, Desc, Limit, Offset □□ □□□□□ □□ request□ data□ □□□□ □□□□□.

php

```
$this->callApiService->callApi([
  'url' => 'app-guest-page',
  'data' => [
    'PageVars' => [
      'Query' => "app_name = '$appName' and is_on_use = 1",
      'Limit' => 1,
    ]
  ],
'headers' => [
  'GateToken' => $this->gateToken['main']
]
```

javascript

```
let response = await get_api_data('setting-search-page', {
  QueryVars: {
    QueryName: 'igroup',
    FilterName: 'dbr_igroup.id',
 },
  PageVars: {
    Limit: 9999,
    Offset: 0,
  }
})

// get_api_data(url, data)
```

Revision #15
Created 26 February 2024 08:30:18 by Admin
Updated 5 March 2025 13:05:54 by □□□