

# PAP-API

## API 概要

このAPIは Table 内のデータを API 経由で取得するための API です。PAP-API には以下の API があります。

- **Pick**(取得する ID)
- **Act**(取得する ID, 操作, 条件)
- **Page**(取得する row 数)

## API URL 一覧

PAP-API には以下の API があります。-pick, -act, -page は API URL の接尾辞です。

例

- item 取得 API URL item-pick
- member 取得 API URL member-pick
- item 操作 API URL item-act
- member 取得 API URL member-page
- member 操作 API URL member-act (Id : ID)
- member 操作 API URL member-act (Id : ID)

## pick-api 概要

この API は WHERE 条件で ID を取得するための API です。

php

```
$itemPick = $this->callApiService->callApi([
    'url' => 'item-pick',
    'data' => [
        'Page' => [
            ['UpdatedMd5' => $updatedMd5]
```

```

    ],
    ],
    '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 FROM 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  