

Pro Page

1 () pro page

[- 1]

1. Controller

pro Controller dabory/themes/pro/app/Http/Controller/etc OneToOneController.php

- API

API App\Services\CallApiService

```
class OneToOneController extends Controller
{
    private $callApiService;

    public function __construct(CallApiService $callApiService)
    {
        $this->callApiService = $callApiService;
    }

    public function list()
    {
        $limit = (int)request('limit', 12);
        $page = (int)request('page', 1);

        $oneToOnePage = $this->callApiService->callApi([
            'url' => 'list-type1-page', // api url
            'data' => [ // list-type1-page data
                'QueryVars' => [ // api Query
```

```

        'QueryName' => 'pro:my-page/post-list', // pro/.../my-page/post-list/ 1to1 1to1 1to1 1to1.
        'SimpleFilter' => "post_code='1to1'", // where 1to1 1to1
        'SubSimpleFilter' => "image_type = 'thumb'",
        'IsntPagination' => false // pagination 1to1
    ],
    'ListType1Vars' => [
        'OrderBy' => request('sort', 'mx.created_on desc')
    ],
    'PageVars' => [
        'Limit' => $limit,
        'Offset' => ($page - 1) * $limit
    ]
    ];

```

```

// dump($oneToOnePage);

```

```

    $oneToOnePage['Page'] = new LengthAwarePaginator($oneToOnePage['Page'],
$oneToOnePage['PageVars']['QueryCnt'],
    $limit, $page, ['path' => request()->url()]);
    // dump($oneToOnePage);
    return view('views.etc.1to1-list', compact('oneToOnePage'));
}

```

```

public function store()
{
    $validator = Validator::make(request()->all(), [
        'post_title' => 'required',
        'post_contents' => 'required',
        'pc1' => 'required',
        'pc2' => 'required|email'
    ]);
}

```

```

if ($validator->fails()) {
    notify()->error(_e('Action failed'), 'Error', 'bottomRight');
    return redirect()->back()
}

```

```

        ->withErrors($validator)
        ->withInput();
    }

```

```

$response = $this->callApiService->callApi([
    'url' => 'post-act',
    'data' => [
        'Page' => [
            [
                'Id' => 0,
                'PostTitle' => request('post_title'),
                'PostContents' => request('post_contents'),
                'PostTypeid' => 6,
                'Pc1' => request('pc1'),
                'Pc2' => request('pc2'),
                'Status' => '2',
                'MemberId' => session('member')['MemberId'],
            ]
        ]
    ],
]);

```

```

if ($this->callApiService->verifyApiError($response)) {
    notify()->error($response['body'], 'Error', 'bottomRight');
    return redirect()->back();
}

```

```

//    notify()->success(_e('Action completed'), 'Success', 'bottomRight');
return redirect()->route('1to1.list');
}

```

```

public function show($id)
{
    $listType1Book = $this->callApiService->callApi([
        'url' => 'list-type1-book',
        'data' => [

```

```

'Book' => [
  [
    'QueryVars' => [
      'QueryName' => 'pro:my-page/post-details',
      'SimpleFilter' => "post_code='1to1' and mx.id = $id",
      'IsntPagination' => true,
    ],
    'PageVars' => [
      'Limit' => 1
    ]
  ],
  [
    'QueryVars' => [
      'QueryName' => 'pro:my-page/post-details-prenext',
      'SimpleFilter' => "mx.id = (select max(id) from pro_post where id < ${id} and post_type_id =
6)",
      'SubSimpleFilter' => "",
      'IsntPagination' => true,
    ],
    'PageVars' => [
      'Limit' => 1
    ]
  ],
  [
    'QueryVars' => [
      'QueryName' => 'pro:my-page/post-details-prenext',
      'SimpleFilter' => "mx.id = (select min(id) from pro_post where id > ${id} and post_type_id =
6)",
      'SubSimpleFilter' => "",
      'IsntPagination' => true,
    ],
    'PageVars' => [
      'Limit' => 1
    ]
  ],
]
];

```

```
// dd($listType1Book);

if ($this->callApiService->verifyApiError($listType1Book)) {
    notify()->error($listType1Book['body'], 'Error', 'bottomRight');
    return redirect()->back();
}

// dump($listType1Book);
return view('views.etc.1to1-details', compact('listType1Book'));
}
}
```

1:1 문의 관련 Controller

list-type1-page URL, data weberp-queries URL, QueryVars, SQL, ,

View

API, View, View

TEST

Controller, view, erp/pro

1:1 문의

pro



erp, 1:1 문의

일자범위

일

주

월

분기

반기

년

전체

날짜

2024-07-09

 ~

2024-07-09

검색 조건

=검색 조건=

상태별 검색

전체

	번호	제목	생성일	요청회원	답변자	완료일	상태
<input type="checkbox"/>	1	문의 테스트	24-07-03 10:21:40	83ffidca2117634626548	kimhi65@0c7d		미처리

1111 11111 1 111111.

1111

erp11111111

erp11111111