

000-0000 Frontend 00`

- 1. 0000
 - Production DB 000 .env.dabory 0 00
 - 000 DB API 0 DB 00 00
 - Docker0 0000 00
 - 000/0000 0000 00
 - 0000(Dummy-Theme)00 00
 - 0000(Production-Theme)0000
 - Dabory Linux 00 SSH 0 000 00
- 2. Restfull API 00
 - 000 Gate Token 00
 - 000 API 000 00 00
 - PAP(pick-act-page) API 00
- 3. Frontend 00 00
 - .env 0 00
 - .env.dabory 0 Gate Token
 - 000 Frontend 00 00
 - 000 Theme 000 00
 - gate-token-get 00 000
- 4. Parameter 00 00
 - 000 00000 00
 - ListType1 00000 000 000
- 5. 00 000 00 00
 - 00 000 00

- 資料庫 安裝

- 6. 安裝 資料庫

- theme 安裝 安裝
- git 安裝 Standard/theme 安裝 安裝
- theme 安裝 .env 安裝 安裝
- 安裝 安裝 安裝 安裝
- ERP 安裝 安裝 安裝 安裝

- 7. 安裝 資料庫

- 安裝 安裝

- 8. 安裝 安裝 安裝

10.

--

--	--	--	--

1. 環境

Production DB 環境 .env.dabor

環境 Front-end環境 Laravel環境 環境 環境 環境 環境 env 環境 環境 環境 環境 環境 環境. 環境 Back-end API 環境 環境

- **env.dabory** 環境 backend api 環境 環境 環境 環境. api server 環境 db server 環境 環境 env. dabory 環境 環境 環境. 環境 環境.
- **env.dabory** 環境 環境
 - *_API_URL = 環境 API環境 URL:PORT
 - *_API_CLIENT_ID = API CLIENT ID
 - *_API_CLIENT_SECRET = API Client Secret
 - *_API_BEFORE_BASE64 = API Keypair

Front-end環境 db 環境 環境 sodium環境 環境 環境 環境 環境 環境 API 環境 環境 環境 環境 環境.

.env.dabory環境 dabory sso(<https://www.daborysso.com/>)環境 環境 環境 my-app 環境 App Manger 環境 env.dabory 環境 環境 環境 env.dabory 環境 環境 環境 api環境 環境 環境.

1.

DB API DB

2. Restfull api 1.

2 1 clientId beforebase64 json type API api response ga db api .

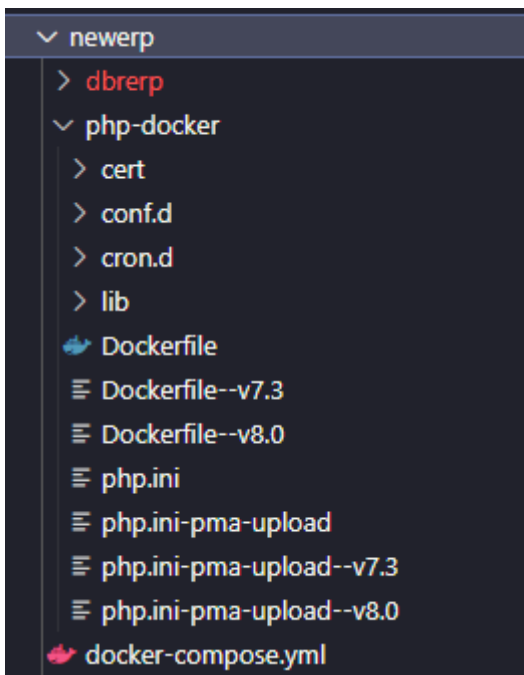
1. 安装

Docker 安装

docker 安装

安装 docker 需要安装 php-docker 包。 (git clone php-docker url)

docker 包在 Dockerfile 和 docker-compose.yml 文件中。 Dockerfile 文件中 php 7.3 版本



- php 8.0 版本
copy Dockerfile--v8.0 Dockerfile
copy php.ini-pma-upload--v8.0 php.ini-pma-upload
安装 Dockerfile 文件。

安装 docker 需要安装 dbrep 包。 (git clone dbrep url) 安装 dbrep 包。

- /dbrep/.env
- dbrep/.env.dabory

安装 dbrep 包。 安装 .gitignore 文件。 安装 .gitignore 文件。 (ventor, theme) 安装 up.sh 文件。 安装 docker 包。

1. 安装

ssh/ssh-keygen ssh-keygen

ssh/ssh-keygen ssh-keygen

*[.ssh] 目录 目录 目录

在本地 目录 目录 目录 目录. git clone 目录 目录 目录 目录 gitignore 目录 目录 目录. 目录 [

- /dabory/themes
- /vendor

在本地 目录 目录 ussh 目录 目录 目录 目录. ./ussh 目录 目录 目录 目录 目录.

- 在本地 OS 目录 [window](#) 目录.

在本地 目录 目录. /home/目录/目录.daboryhost.com 目录

git clone 'repository url' 目录 目录 目录 目录.

在本地 目录 目录 目录 目录 目录 目录.

1. 安装

安装(Dummy-Theme)包

.env 文件中 DBR_THEME 设置为 erponly。erponly 为 /pro 目录下的 erponly 子目录 user page 重定向到 /dabory/themes 目录下 [erponly](#) 子目录 theme 目录下。安装 erponly 包后，安装 erponly 包。

1. 设置

设置(Production-Theme)设置

• 设置(Production-Theme)设置

.env设置 ERP_THEME, PRO_THEME 设置 设置. 设置 设置 设置 设置 设置 设置 设置. /themes/pro 设置 设置 (

1.

Dabory Linux SSH

* ussh linux

ussh

```
baeksan@DESKTOP-AJSCJCD MINGW64 /d
$ cd ssh/

baeksan@DESKTOP-AJSCJCD MINGW64 /d/ssh
$ ls
dabory_key.pem  dbr_list.txt  ussh*
```

(1). ussh

(2). pem key

(3). txt

- txt .ussh

ex) list_file=\$svr_str3"_list.txt"

ussh

./ussh serverhost username

ex) ./ussh eyeFr textmsg

linux

* window wsl git bash

2. Restfull API

Gate Token

. Gate Token?

SSO ClientId PublicKey sodium key pair sso

(Front-end) - - - - > {Client Id, BA64 Key} - - - - > (Back-end)

gateToken

(Back-end) - - - - > {gate token} - - - - > (Front-end)

Front-end ClientId db BA64 Key Back-end Back-end Gate Token Token Gate token API

[Gate Token]

<https://manual.dabory.com/books/frontend-tutorial/page/envdabory-gate-token>

API

. API

- ERP API POST method
- **API GateToken**
 - API DB PKI BeforeBase64 Key (<https://kbsodium.gitbook.io/doc>)
 - DaborySSO Key Pair API SSO
 - Frontend Server ClientId DaborySSO App
 - Client(Frontend) GateToken API DB
- API Gate Token Header

PAP(pick-act-page) API

PAP(pick-act-page) API?

PAP(pick-act-page) API는 Table 에 있는 데이터를 API 를 통해 가져오고, API 를 통해 데이터를 (Pick),

- **Pick api** ()
 - ID 를 통해 데이터를 가져오는 API.
- **Act api** ()
 - Insert, Update, Delete 에 대한 API.
 - ID 를 통해 Insert(), Update(), Delete() 에 대한 API.
 - Delete 에 대한 ID 를 통해 데이터를 삭제하는 API.
- **Page api** ()
 - Select 에 대한 2개 이상의 데이터를 가져오는 API.
 - QueryCnt, Query, Fields, Asc, Desc, Limit, Offset, ReturnJson 에 대한 데이터를 가져오는 API.
 - ReturnJson 에 대한 데이터를 Base64 인코딩하여 반환하는 API.

*API Json Structure

```
"PageVars": {  
  "QueryCnt": "string", //Response 에 Where 에 Row Count - Request 에  
  "Query": "string", //Query Where Clause 에 (And / Or )  
  "Fields": "string", // 에 Empty (all)  
  "Asc": "string", // ORDER BY *** ASC  
  "Desc": "int", // ORDER BY *** DESC  
  "Limit": "int", // Row , Default  
  "Offset": "int", // Where rows  
  "ReturnJson": "string", //Base64 인코딩  
}
```

3. Frontend

.env

- .env

Laravel .env (<https://laravel.kr/docs/5.7/configuration>)

Laravel .env Example(https://zetawiki.com/wiki/.env,_env.example)

```
# [Dabory]
ERP_THEMES= (themes/erp )
PRO_THEME=PRO (themes/pro )
LOCALE_SEQUENCE=
APP_MOBILE_NO=
TAG_LINE=
CDN_TYPE= CDN (aws-s3 or local)
MEDIA_URL= base url (local or a3)
FAVICON_PATH= path
IS_SKIP_DBUPDATE=DBUPDATE or
IS_ON_MEMBER_SIGNUP=
## [MetaMask]
CHAIN_ID=0xE67B
CHAIN_NAME='Dang Smart Chain'
RPC_URLS=http://3.38.62.211:8545
NATIVE_CURRENCY_NAME=DGT
NATIVE_CURRENCY_SYMBOL=DGT
NATIVE_CURRENCY_DECIMALS=18
BLOCK_EXPLORER_URLS=https://explorer-staging.dangnn.co.kr
### [asset & cache]
BROWSER_CACHE= (false)
CSSJS_URL=local or asset url (local)
### [Printer]
REPORT_SERVER_URL= base url
### [Laravel]
APP_NAME=
APP_ENV=.env
APP_KEY=base64:.env
APP_DEBUG=.env
APP_URL=url
LOG_CHANNEL=stack
LOG_LEVEL=debug
DB_CONNECTION=
DB_HOST=
DB_PORT=
```

DB_DATABASE=xx
DB_USERNAME=xx
DB_PASSWORD=xx
BROADCAST_DRIVER=xx
CACHE_DRIVER=xx
FILESYSTEM_DRIVER=xx
QUEUE_CONNECTION=xx
SESSION_DRIVER=xx
SESSION_LIFETIME=xx
MEMCACHED_HOST=xx
REDIS_HOST=xx
REDIS_PASSWORD=xx
REDIS_PORT=xx
MAIL_MAILER=.env xx xx xx
MAIL_HOST=.env xx xx xx
MAIL_PORT=.env xx xx xx
MAIL_USERNAME=.env xx xx xx
MAIL_PASSWORD=.env xx xx xx
MAIL_ENCRYPTION=.env xx xx xx
MAIL_FROM_ADDRESS=.env xx xx xx
MAIL_FROM_NAME=.env xx xx xx
AWS_ACCESS_KEY_ID=.env xx xx xx
AWS_SECRET_ACCESS_KEY=.env xx xx xx
AWS_DEFAULT_REGION=.env xx xx xx
AWS_BUCKET=.env xx xx xx
AWS_USE_PATH_STYLE_ENDPOINT=.env xx xx xx
PUSHER_APP_ID=xx
PUSHER_APP_KEY=xx
PUSHER_APP_SECRET=xx
PUSHER_APP_CLUSTER=xx
MIX_PUSHER_APP_KEY=xx
MIX_PUSHER_APP_CLUSTER=xx
ANALYTICS_UA_ID=
ANALYTICS_VIEW_ID=
ANALYTICS_G_ID=
ANALYTICS_PROPERTY_ID=

.env.dabory Gate Token

2 Rest Api <https://manual.dabory.com/books/frontend-tutorial/chapter/restfull-api> Gate Token
dabory Gate Token

dabory Gate Token(DGT)

- [KeyPair/PublicKey PHP Code

<https://www.php.net/manual/en/function.sodium-crypto-box-seal.php>.

- [Sodium Ajax (260~262 Lines)

<http://git.daboryhost.com:10880/dabory/dbrerp/src/master/routes/web.php>).

- [DB how to use it in PHP (254~263 Lines)

<http://git.daboryhost.com:10880/dabory/dbrerp/src/master/resources/views/front/dabory/pro/my-app/popup/popup-form1/form-a/client-app-form.blade.php>).

- secret_key key_pair db (erp_center_db, erp_memory_db, erp_individual_dbs)
- db (driver, host, username, database, passwd, public_key) -> client config
- db

[DGT

- api DGT
- api client config 20 Sodium
- DGT status 200, DGT
- DGT
- DGT DGT

[Main Request]

- DGT

[DCS

- DGT DCS
- memory_db DGT DCS
- DCS DB(erp databases)
- db data

[]

Frontend

.Frontend

- 0.0.0
- app
- app.examples
- appslabke
- artisan*
- composer
- config
- dabory**
 - (...)
- public**
- ralouphe
- ramsey
- resources robots.txt
- (...)

- dabory

- dabory Composing /erp para themes son

- public

- public
-

- para- -/dabory themed**dabory** .
- **dabory** **public** (Symbolic link) .

11 Theme 11 11

Theme 11 11

11 1111 1111 1111.

1. pro (111 111)

1111 11 111 11 111 1 11 1111 1111 11 111 111 11111.

- pro 111 11 11

└─themes

└─eyerecord

└─erp

└─shell

└─pro

└─app

└─Http

└─Services

└─para

└─resources

└─routes

└─shell

2. erp (테마 테마)

테마를 만들 때 테마를 만들기 전에 테마를 만든다.

erponly 테마 erp 테마를 만들기 전에 /pro (테마를 만들기) 테마를 만들기 전에 user page redirection 테마를 만들기 전에 테마를 만든다.

- erp 테마를 만들기 전에

└─themes

└─erponly

└─erp

└─shell

└─pro

└─routes

└─shell

└─qry

○ ()

local PC: \$ cd pro newerp-php73

\$ cd /Users/EricKim/docker-works/php/newerp-php73/dbrrerp/public/themes/pro

server : \$ cd pro newerp cd /home/newerp/public_html/public/themes/pro

local PC: \$ cd erp newerp-php73

\$ cd /Users/EricKim/docker-works/php/newerp-php73/dbrrerp/public/themes/erp

server : \$ cd pro newerp cd /home/newerp/public_html/public/themes/erp

gate-token-get 11 111

- api 1111 11 gate-token 1111

rest api 1111 1 11 PostMan 11 Thunder Client 1111 gate-token response 111111.

11 .env.dabory 1111 11111 MAIN_API_URL, MAIN_API_CLIENT_ID, MAIN_API_BEFORE_BASE64

111 11 1111 111111. Thunder Client 11 MAIN_API_URL 1111 111111 clientId BeforeBase64 11 gate-token
store() 11 test() 1111 111111 post 111 111. 11
body 11 json 1111 111 11 111111.

```
{  
  "ClientId": "MAIN_API_CLIENT_ID 11 value",  
  "BeforeBase64": "MAIN_API_BEFORE_BASE64 11 value"  
}
```

11 11 1111 api server 11 gate-toekn 1111 1 1111.

4. Parameter θ

4. Parameter 11 11



11 11 11 11 11 json 11 11 11 11 11 11 11 11 11 11.

11 11 11 11 11 json 11(11 11 11 11) 11 11 11 11

11 11 11 json 11 11. json 11 11 sql 11 11 11 11 11 11 11 11 11 11.

11 11 11 11 11 11 11.

/dabory/themes/\$theme/\$theme_type/para/ko_KR/{\$type}/{filePath}.json

\$theme = 11

\$theme_type= pro or erp

\$type = request or response

\$filePath = 11 + 11

json 11 11

- json 11 11

```
{
  "//list-type-1": "simple and plain list by query of table join",

  "General": {
    "Title": "Crawling Domains",
    "PageApi": "list-type1-page",
    "ActApi": "domain-act"
  },

  "QueryVars": {
    "QueryName": "pro/my-app/dbr-main/domain-input",
    "FilterName": "",
    "FilterValue": "",
    "FilterDate": "domain_date"
  },

  "HeadSelectOptions": [
    {
      "Value": "new",
      "Caption": "11 11"
    },
  ],
```

```

{
  "Value": "multi-delete",
  "Caption": "☐☐ ☐☐"
},
{
  "Value": "clear-all-filter",
  "Caption": "☐☐☐☐"
}
],

```

```

"//FormVars": "FormVars[0]-caption, FormVars[1]-searchPopup",

```

```

"FormVars": [
  {
    "ListButton": "☐☐",
    "DateRange": "",
    "DateNavi": "☐☐☐☐(☐☐)",
    "Date": "☐☐",
    "From": "☐☐",
    "To": "☐☐",
    "FirstRange": "",
    "SecondRange": "",
    "ThirdRange": "",
    "FourthRange": "",
    "AddTotalLine": "",
    "SelectPopup": "",
    "MultiPopup": "",
    "ChartPopup": "",
    "DownloadList": "",
    "ShowOnlyClosed": "",
    "Balance": "",
    "OrderBy": "☐☐ ☐☐",
    "FilterOption": "☐☐☐☐",
    "SimpleOption": "☐☐☐☐ ☐☐"
  }
],

```

```

"FilterSelectOptions": [
  {
    "Value": "",
    "Caption": "=☐☐ ☐☐="
  },
  {
    "Value": "mx.domain_url",
    "Caption": "Domain URL"
  },
  {
    "Value": "mx.solution_code",

```

```
    "Caption": "Solution Code"
  },
  {
    "Value": "mx.search_tool",
    "Caption": "Search Tool"
  }
],
```

```
"SimpleSelectOptions": [
  {
    "Value": "",
    "Caption": "□□"
  },
  {
    "Value": "mx.status='0'",
    "Caption": "Waiting"
  },
  {
    "Value": "mx.status='1'",
    "Caption": "Confirmed"
  },
  {
    "Value": "mx.status='2'",
    "Caption": "GivingUp"
  },
  {
    "Value": "mx.status='9'",
    "Caption": "Exception"
  }
],
```

```
"//ListType1RangeVars": "[0]-filter, [1]-component, [2]-parameter",
```

```
"ListType1RangeVars": [
```

```
  {
    "FirstRange": "",
    "SecondRange": "",
    "ThirdRange": "",
    "FourthRange": ""
  },
```

```
  {
    "FirstRange": "",
    "SecondRange": "",
    "ThirdRange": "",
    "FourthRange": ""
  },
```

```
  {
    "FirstRange": "",
```

```

        "SecondRange": "",
        "ThirdRange": "",
        "FourthRange": ""
    }
],

 "// DateRangeOptions": "□:□□□□ 1~30, □□: □□□□ □□ 1□ ~ □□□□ □□, □□/□ □□ □□,□□:1990~3000",
 "DateRangeOptions": [
 ],

 "DateNaviOptions": [
     {
         "Value": "day",
         "Caption": "□"
     },
     {
         "Value": "week",
         "Caption": "□"
     },
     {
         "Value": "month",
         "Caption": "□"
     },
     {
         "Value": "quarterly",
         "Caption": "□□"
     },
     {
         "Value": "year",
         "Caption": "□"
     },
     {
         "Value": "all",
         "Caption": "□□"
     }
 ],

 "SelectPopupOptions": [

 ],

 "// OrderByOptions": "OrderBy Options//□□□ Value□ Default",
 "OrderByOptions": [
     {
         "Value": "mx.domain_url asc",
         "Caption": "□□□ □□"
     },
 ],

```

```

{
  "Value": "mx.solution_code asc",
  "Caption": "解題順序"
}
],

```

```

"//BalanceOptions": "Show OnlyBalanceRemained or All",

```

```

"BalanceOptions": [
  {
    "Value": "",
    "Caption": "全部"
  },
  {
    "Value": "c10 > 0",
    "Caption": "解題順序"
  }
],

```

```

"DisplayVars": {
  "IsListFirst" : true,
  "IsExcelColumn" : false,
  "InitLines" : 15,
  "HeadHeight": "130",
  "InitDateRange" : "all",
  "IsC1Popup": "1",
  "BodyHeight": "680"
},

```

```

"// ListVars": "ListVars[0]-caption, ListVars[1]-size(px), ListVars[2]-align->right(number)
left(left)",

```

```

"ListVars": [
  {
    "$Radio": "$Radio",
    "$Check": "$Check",
    "No": "No",
    "C1": "Domain URL*",
    "C2": "Host Hash",
    "C3": "Domain Date",
    "C4": "Owner Name",
    "C5": "Solution Code",
    "C6": "OwnerFile",
    "C7": "EncryptKey",
    "C8": "DbrHub URLs",
    "C9": "TimeZone",
    "C10": "解題順序",
    "C11": "Aggreg Hash",
    "C12": "Status"
  }
]

```

```

    },
    {
        "$Radio": "5",
        "$Check": "5",
        "No": "5",
        "C1": "20",
        "C2": "15",
        "C3": "15",
        "C4": "20",
        "C5": "15",
        "C6": "10",
        "C7": "10",
        "C8": "10",
        "C9": "10",
        "C10": "10",
        "C11": "10",
        "C12": "10"
    },
    {
        "$Radio": "center",
        "$Check": "center",
        "No": "center",
        "C1": "left",
        "C2": "center",
        "C3": "center",
        "C4": "center",
        "C5": "center",
        "C6": "center",
        "C7": "center",
        "C8": "center",
        "C9": "center",
        "C10": "status('crawled')",
        "C11": "center",
        "C12": "status('domain')"
    }
}
]
}

```

□□□ □□□ □□ □□

1. General

Title : □□□□

PageApi : page api□

ActApi : act api□

2. QueryVars -> □□ □□ □□□□

□□ □□□□ □□ □□□□ db □□□ □□□□

FilterDate : □□ □□ □□ □□ □□ □□ □□

Caption : option□

```
"key" : "value" □
```

FourthRange :

BodyHeight :

ListVars[2] -> align->right or left or center

ListType1 数据类型

- ListType1 数据类型

数据类型 数据类型 ListType1 数据类型

- ListType1 数据类型

数据类型 LT1 数据类型.

C5:파라미터명
/setup/struct-type1/struct-input
/list/list-type1/dbr-main/search/domain-input
/list/list-type1/dbr-main/search/solution-input
/list/list-type1/users-input
/list/list-type1/users-input
/list/list-type1/app-api-input
/list/list-type1/user-menu-input
/form/form-b/user-menu-perm
/list/list-type1/users-ext-input
/list/list-type1/pro/member-menu-input
/form/form-b/member-menu-perm
/list/list-type1/pro/member-ext-input
/list/list-type1/app-api-input

erp 1000 100000 - 100 1000 100 10 1000 10 10000.

dbr_user_menu para_name 10000 10 1000. 100000 10000

list-type1 para_name 1000 100 json 100 100 1000 10000.

para_name /list/list-type1/dowonload/menu/sort-menu /list/list-type1/dowonload/menu/ [sort-menu.json 100 1000 10 100 1000 1000 1000.

weberp-queries 100 100 100 1000

10 100 1000 1000 10 100, 10 1000 1000 C1~C20 1000 1000.

```
INSERT INTO dbt_list_type1 ( created_on, list_token, id,  
    c1, c2, c3, c4,  
    c5, c6, c7, c8 )
```

```
SELECT UNIX_TIMESTAMP(), ?, id,  
    menu_code, sort_type, bgcolor_code, dash_page_url,  
    component, parameter, icon, is_enabled
```

```
FROM dbr_sort_menu as mx
```

```
-- @where wngur6076, 2023/08/24 4:39 PM • t
```

```
-- @order
```

이 코드는 dbt_list_type1 테이블에 데이터를 삽입하고, dbr_sort_menu 테이블에서 데이터를 선택하여 json 형식으로 출력하는 SQL 쿼리입니다.

```

"ListVars": [
  {
    "$Radio": "$Radio",
    "$Check": "",
    "No": "번호",
    "C1": "메뉴코드",
    "C2": "소트구분",
    "C3": "메뉴배경색상",
    "C4": "대시보드 페이지 URL",
    "C5": "컴포넌트",
    "C6": "파라미터",
    "C7": "지정아이콘",
    "C8": "사용"
  },
  {
    "$Radio": "3",
    "$Check": "",
    "No": "6",
    "C1": "15",
    "C2": "15",
    "C3": "12",
    "C4": "17",
    "C5": "20",
    "C6": "20",
    "C7": "12",
    "C8": "8"
  },
  {
    "$Radio": "center",
    "$Check": "",
    "No": "center",
    "C1": "center",
    "C2": "center",
    "C3": "center",
    "C4": "center",
    "C5": "center",
    "C6": "center",
  }
]

```

이러한 경우 QueryName을 지정합니다.

```
{
  "General": {
    "Title": "바둑판 메뉴 불러오기",
    "PageApi": "list-type1-page",
    "ActApi": "list-type1-upload"
  },

  "QueryVars": {
    "QueryName": "download/menu/sort-menu",
    "FilterName": "",
    "FilterValue": "",
    "FilterDate": ""
  },
}
```

이러한 구조를 가진 데이터를 list에 담아 보겠습니다.

5□. □□ □□□ □□ □□

5. 1. 2. 3. 4.

1. 2. 3. 4. 5. 6. 7. 8. 9. 10. 11. 12. 13. 14. 15. 16. 17. 18. 19. 20.

슈퍼 유저 메뉴

관리자 계정

구분설정 관리

잔액/재고/적립금 설정

메뉴 확장과 API 관리

사용자 메뉴

사용자 메뉴 권한

사용자 확장(ERP)

회원 앱 메뉴

회원 앱 권한

회원 확장(Pro)

Host API 등록

Host API 권한 설정

API App 회원 확장

Guest App 정보

시스템 관리(주의)

Query_name :

[illegible]

Component : /views/front/dabory/erp/popup-form1/form-a/menu-form

Parameter : /dabory/para/erp/ko_KR_list/popup/popup-form1/form-a/user-menu

Query_name : download/menu/user-menu

value, .

5□. □□ □□□ □□ □□

권한번호(지동재번)*

super-user

생성일자*

2024-05-13

분류메뉴

전제

권한ID

생성일자

수정일자

권한이름

권한설명

체크/언체크 적용

메뉴코드

분류유형

메뉴명

Skipped

Use

Mymenu

List

Read

Create

Update

Delete

Newtab

Component : /views/front/dabory/erp/perm/user-perm

Parameter : /dabory/para/erp/ko_KR/form/form-b/user-menu-perm

Query_name : user-menu-perm

6□. □□ □□□□□□

6. 如何安装主题

theme 如何安装

如何安装主题

1. Theme 如何安装

安装主题时，需要将主题文件复制到 /public/themes 目录下。

- pro 主题

`db-theme-pro` 主题安装 pro 主题。

`erponly` 主题安装

`git clone erponly`

安装主题时，需要将主题文件复制到 /public/themes 目录下。

- erp 主题

`db-theme-erp` 主题安装 erp 主题。

`optical_pos` 主题安装

`git clone optical_pos`

安装主题时，需要将主题文件复制到 /public/themes 目录下。

* `pro` 主题安装 pro 主题。 `erp` 主题安装 erp 主题。

git  Standard/theme  

```
1. cd $HOME/docker-works/php/ : docker-works/php/
2. all-gitpp {website-alias} : all-gitpp {website-alias}
```

1. `cd /home`: `/home` `XXXX` `XXXX`.
2. `all-gitpp-host {website-alias}`: `all-gitpp-host` `XX` `all-gitpp` `XXX` `XXXX`. `{website-alias}` `Git` `XXXX` `XX` `XXXX` `XXXX`.

☐ PC ☐

6. 設定ファイル

theme 設定ファイル .env の設定

Theme 設定 : .env の設定

設定ファイル .env の設定。

.env の DBR_THEME, PRO_THEME の設定は、設定ファイル .env の設定。

(設定ファイル .env の設定は、設定ファイル .env の設定。)

設定 DBR_THEME の設定は、設定 DBR_THEME の設定。

6.

pro :

point2u .

Welcome to Dabory Shop in Your Dream Online Store! 데스트탑 My Page 고객센터

Search Anything...

Q

Login Join 0 1

뷰티 패션 식품 생활/건강 가전 가구/침구/인테리어 자동차/공구 반려동물 취미/여행/스포츠/E쿠폰 Call Us (02)871-4134 Email Us kimhi65@gmail.com

SHOW :

12

 SHORT BY :

최신순

올리브영1위* [더프트앤도프트] 바디케어 & 핸드케어 5종선물세트 (바디케어3종+핸드케어2종)
28,210원

[더프트앤도프트] 바디케어3종세트 & 톤업 선크림 SPA 50+/ PA++++ (3종선물세트) 바디워시+타올+바디로션+선크림
33,280원

올리브영 1위* [더프트앤도프트] 핸드케어세트 & 톤업 선크림 SPA 50+ (3종선물세트) 핸드워시+핸드크림+선크림
31,590원

올리브영1위* [더프트앤도프트] 유브이 퍼펙션 톤업 선크림 50ml SPA 50+/ PA++++
26,520원

point2u , pro .

-

1. , .
2. route Controller .
3. view text css .

.

□□□...

[□□□□ □□□□ □□]

6. 1 1 1 1 1 1

ERP 1 1 1 1 1 1 1 1

7□. □□ □□□□□□

7. 表 数据库



表 数据库 (standard) 表 数据库 表 数据库

表 数据库 数据库 表 数据库. 数据库 数据库 数据库 表 数据库 数据库

表 theme 表 -> qry 表 表 数据库 表 表 数据库(表 表 sql 表 表)

数据库 表 query_name 表 数据库 表 数据库 数据库 表 表 数据库

8□. □□□ □□□ □□□□