

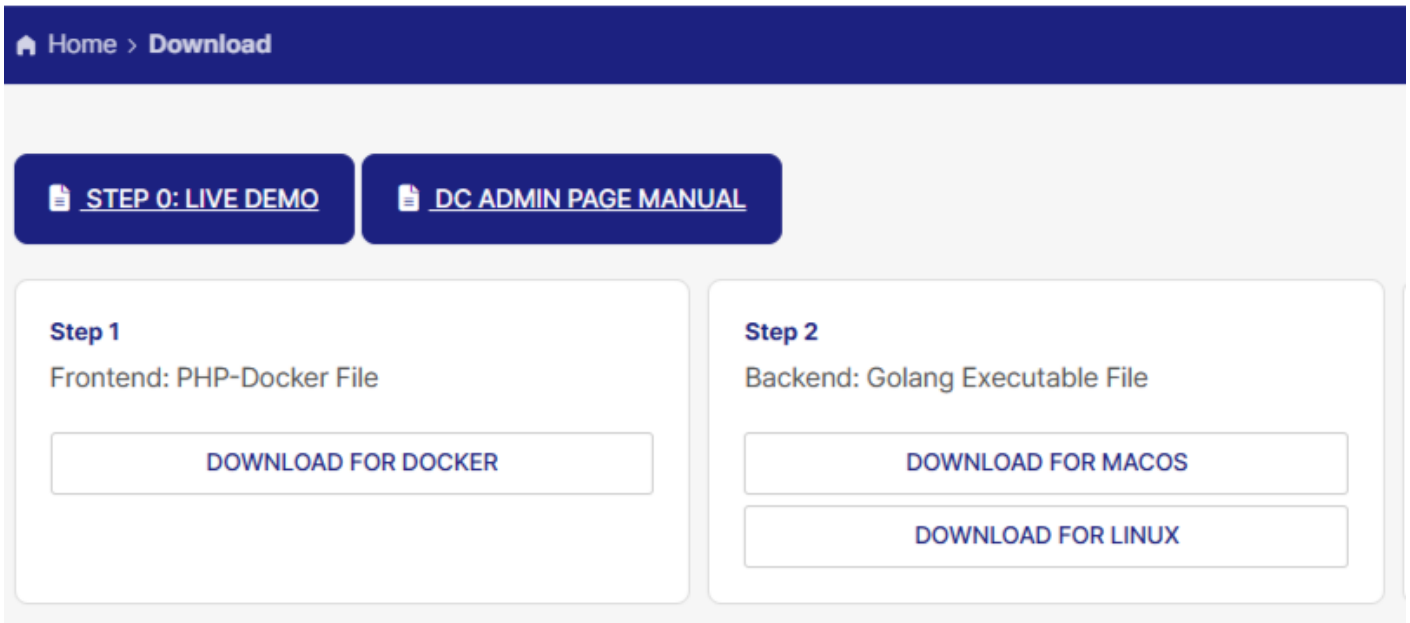
📦 Backend 📦 📦 📦

- DC-Demo 📦 📦 📦 (Local)

DC-Demo 배포 파일 다운로드 (Local)

DC 배포 파일 다운로드 (Mac OS, Linux 배포)

1. DC 배포 파일 다운로드 Backend: Golang Executable File 다운로드.



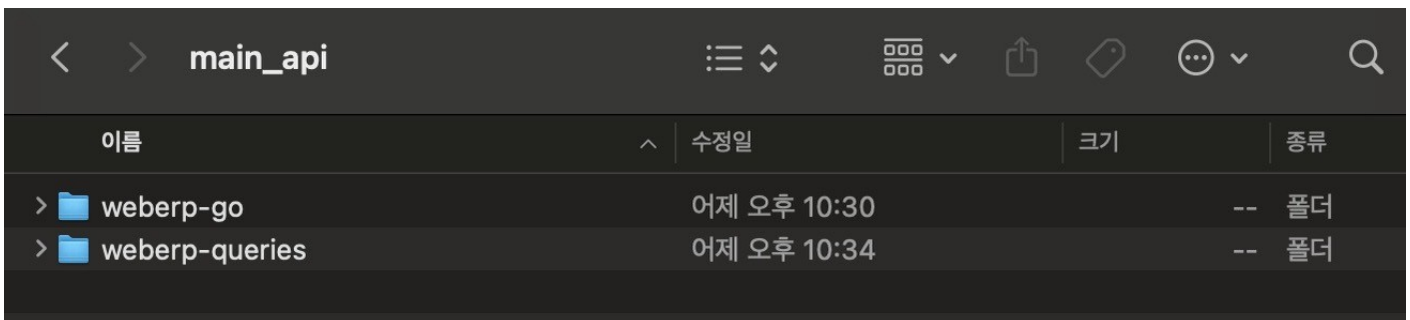
2. `mv ~/Downloads/dc_macos_main_api.tar.gz /path/to/~` 다운로드 받은 파일을 이동.

3. tar 파일을 압축 해제.

```
cd /path/to/~
```

```
tar -xvzf dc_macos_main_api.tar.gz
```

4. 다운로드 받은 main_api 폴더 안에 weberp-go, weberp-queries 폴더 생성.



weberp-go (weberp-go) : GateToken 발급 받은 후 API 호출 가능.

weberp-go / conf : weberp-go의 설정을 관리하는 패키지.

weberp-queries : 쿼리 패키지.

Conf / local_config.json

Kafka 설정

- **KafkaOn** : Kafka를 사용할지 여부 (Yes/No).
- **KafkaTopic** : Kafka의 Topic명.
- **KafkaAddr** : Kafka의 주소 (IP와 Port).
- **KafkaPort** : Kafka의 포트.

gRPC 설정

- **gRpcOn** : gRPC를 사용할지 여부 (Yes/No).
- **AppName** : 애플리케이션명.
- **SiteName** : 사이트명.
- **gRpcProtocol** : gRPC의 프로토콜 (http 또는 tcp).
- **gRpcAddr** : gRPC의 주소 (0.0.0.0 또는 ::).
• **gRpcPort** : gRPC의 포트.

Rest 설정

- **RestOn** : REST API를 사용할지 여부.
- **RestConnect** : REST API의 연결 주소 (예: :8080).
- **RestUri** : REST API의 URI (/api).

개발 모드 설정 (개발용, 배포용 구분)

- **IsDevMode** : 개발 모드 여부.
- **IsSqlDebugMode** : SQL 디버그 모드 여부.
- **IsNormalDebugMode** : 일반 디버그 모드 여부.

DB 설정

- **IsYDBFixed** : YDB 고정 여부.
- **DBOptionString** : DB 옵션 문자열.

XDB 설정

- **XDBOn** : XDB를 사용할지 여부 (Yes/No).
- **XDBConnString** : XDB의 연결 문자열.

Crystal 설정

- **CrystalClientId** : Crystal의 클라이언트 ID.
- **CrystalBB64** : Crystal의 Base64 인코딩.

- **IsCrystalKeyPair** : Crystal (Yes / No)
- **CrystalKeyPair** :

() () ()

- **IsCacheKeyPair** : CacheKeyPair () ()
- **CacheKeyPairDir** : CacheKeyPair () () () ()
- **LocalKeyPair** :

SSO () ()

- **SsoConnString** : SSO () ()
- **SsoAppBase64** : SSO () () () () Base64 ()

DBU () ()

- **DbuConnString** : DBU () ()
- **DbuAppBase64** : DBU () () () () Base64 ()
- **DeviceAuthOn** : () () () () () ()
- **DbuByForceOn** : () DBU () () ()

() () () () ()

- **IsQryFromQDB** : QDB() () () () () ()
- **QueryDir** () () () () () (/Weberp-quary() () () () () ()
- **ThemeQryDir** : () () () () () (/Weberp-quary/themes() () () () () ()

() () () ()

- **DbuEncryptCode** : DBU () () ()
- **SslMode** : SSL () () () ()
- **SslConnect** : SSL () () ()
- **SslFullChain** : SSL () () () () () ()
- **SslPrivate** : SSL () () () ()

Blockchain ()

- **GethConnString** : Geth() () () () () ()
- **XrpConnBridge** : XRP() () () () () ()

() () () () () () () ()

- **DbType** : () () mysql ((postgresql)

Kafka consumer ()

- **IsKafkaConsumer** : Kafka Consumer () ()
- **KafkaConnString** : Kafka () () () (: 123.456.789.101:9092)

- # ElasticSearch

- ## Conf / Config_select.json

weberp-queries (□□ □□ □□)


$$(\square\square\square\square = \square\square\square\square\square\square, \square\square\square\square\square\square\square\square = \square\square\square\square\square\square\square\square\square\square)$$

Weberp-queries / themes

ex) **Weberp-queries / themes / erp / eye / eyetest.sql**

```
SELECT dbr_sorder.* FROM dbr_sorder WHERE dbr_sorder.sorder_no=?
```

Cache-key-pair

- 
 -  (Public/Private Key) 
 -     
- 
 -     

- 在配置文件中
 - 添加 `cache-key-pair` 配置项
 - 配置项的值为一个数组，包含两个字符串

Cache-key-pair 配置项

- 该配置项的值为一个数组，包含两个字符串
- 第一个字符串为缓存键名，第二个字符串为缓存值
- 该配置项的值可以为空数组，表示不使用缓存

5. weberp-go 在 `./weberp-go` 目录下运行。

在 `18080` 端口上运行。 (`local_config.json` 文件中 `RestConnect` 项设置为 `18080`)