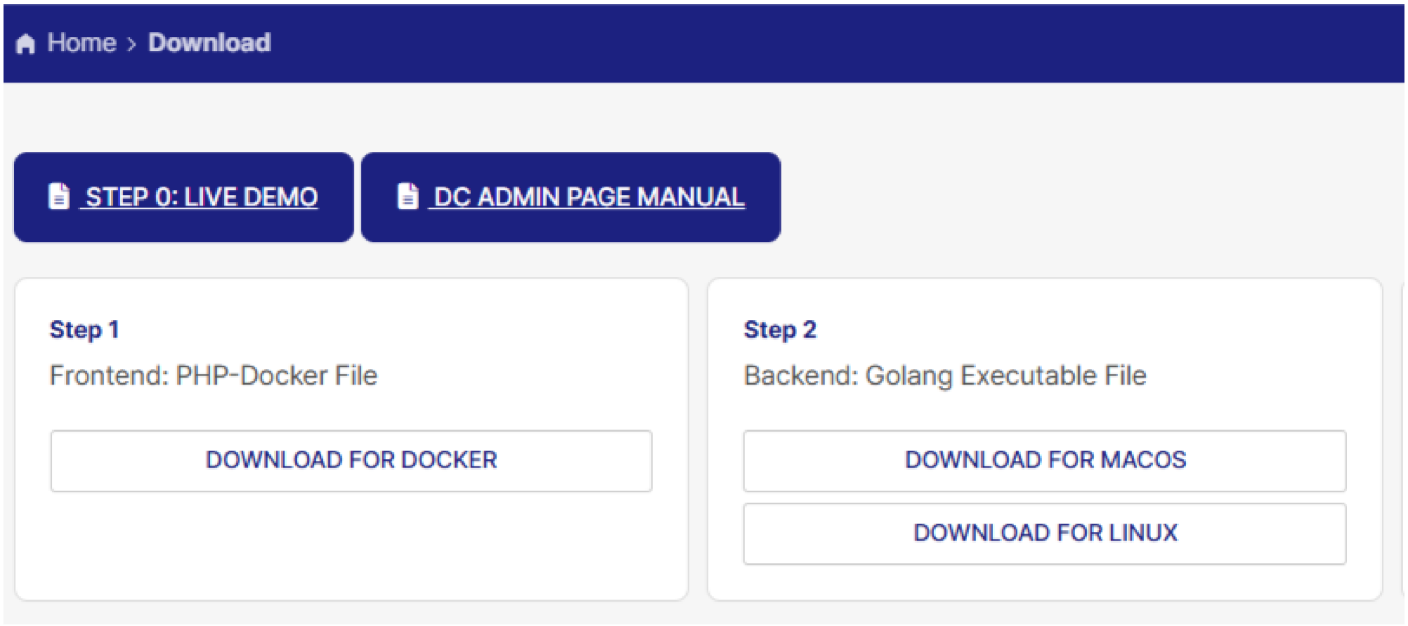
DC-Demo 백엔드 설치 가이드 (Local)

DC 백엔드 설치 방법 (Mac OS, Linux 방식 동일)

1. DC 백엔드 설치본은 백엔드 다운로드 링크를 클릭하여 환경에 맞는 **Backend: Golang Executable File**을 다운로드합니다.

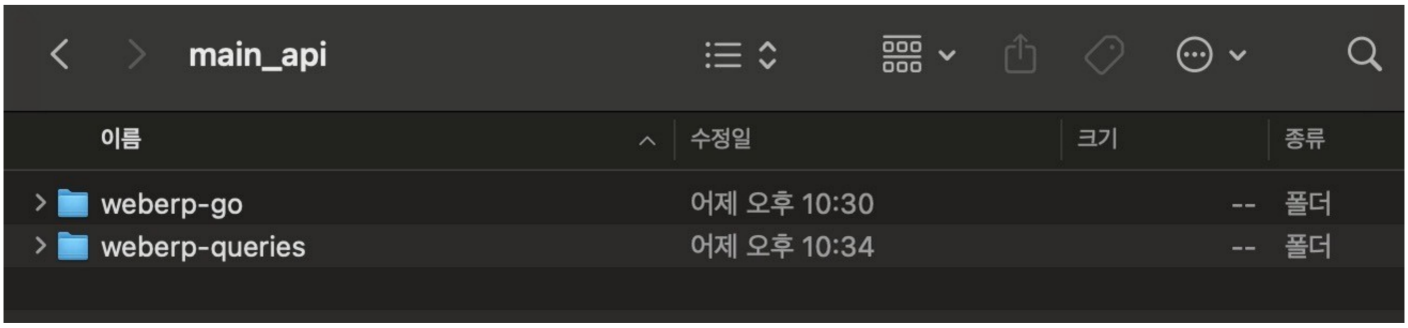


2. 다운`mv ~/Downloads/dc_macos_main_api.tar.gz /path/to/~` 명령어를 사용하여 설치 디렉토리로 이동 시킵니다.

3. tar 명령어를 이용해 압축을 해제합니다.

```
cd /path/to/~
tar -xvzf dc_macos_main_api.tar.gz
```

4. 압축 해제되면 **main_api**라는 파일안에 **weberp-go, weberp-queries** 파일 두개가 생성됩니다.



weberp-go (실행 파일) : GateToken을 발급받을 수 있는 백엔드 서버 실행 파일입니다.

weberp-go / conf : weberp-go를 실행시키기 위한 설정파일입니다.

weberp-queries : 쿼리 파일입니다.

Conf / local_config.json

Kafka 관련 설정

- **KafkaOn** : Kafka 사용 여부를 설정
- **KafkaTopic** : Kafka에서 사용할 토픽 이름
- **KafkaAddr** : Kafka 브로커의 주소(IP 또는 도메인)
- **KafkaPort** : Kafka 브로커가 사용하는 포트

gRPC 관련 설정

- **gRpcOn** : gRPC 사용 Yes / No).
- **AppName** : 애플리케이션 이름.
- **SiteName** : 사이트 이름.
- **gRpcProtocol** : gRPC에 사 http [] tcp 등).
- **gRpcAddr** : gRPC 서버의 IP 0.0.0.0 [] 모든 네트워크 인터페이스를 의미.
- **gRpcPort** : gRPC 서버 포트 번호.

Rest 관련 설정

- **RestOn** : REST API 사용 여부
- **RestConnect** : REST API 연결 포트(:8080][포트] 지정)
- **RestUri** : REST API의 기본 URI /api]:

개발 및 디버그 설정 (활성화된 경우, 로그 저장 위치를 명시)

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

DB 관련 설정

- **IsYDBFixed** :
- **DBOptionString** :

XDB 관련 설정

- **XDBOn** : XDB 사용 Yes / No)
- **XDBConnString** : XDB 연결 문자열

Crystal 관련 설정

- **CrystalClientId** : Crystal 서비스 클라이언트 ID
- **CrystalBB64** : Crystal의 Base64 키
- **IsCrystalKeyPair** : Crystal 키 활 Yes / No)
- **CrystalKeyPair** :

키페어 관련 설정

- **IsCacheKeyPair** : CacheKeyPair 사용 여부
- **CacheKeyPairDir** : CacheKeyPair 디렉터리 경로
- **LocalKeyPair** :

SSO 관련 설정

- **SsoConnString** : SSO 연결 주소
- **SsoAppBase64** : SSO 애플리케이션 Base64 값

DBU 관련 설정

- **DbuConnString** : DBU 연결 주소
- **DbuAppBase64** : DBU 애플리케이션 Base64 값
- **DeviceAuthOn** : 디바이스 인증 사용 여부
- **DbuByForceOn** : 강제 DBU 활성화 여부

쿼리 디렉터리 설정

- **IsQryFromQDB** : QDB에서 쿼리를 가져올지 여부
- **QueryDir** : 일반 쿼리 디렉터리 경로 (/Weberp-quary가 존재하는 경로로 지정)
- **ThemeQryDir** : 커스터마이징 쿼리 디렉터리 경로 (/Weberp-quary/themes가 존재하는 경로로 지정)

기타설정

- **DbuEncryptCode** : DBU 암호화 코드
- **SslMode** : SSL 모드 활성화 여부
- **SslConnect** : SSL 연결 포트
- **SslFullChain** : SSL 인증서 전체 체인 경로
- **SslPrivate** : SSL 개인 키 경로

BlockChain 설정

- **GethConnString** : Geth(이더리움) 연결 주소
- **XrpConnBridge** : XRP(리플) 연결 브리지 주소

데이터베이스 관련 설정

- **DbType** : 데이터 mysql (postgresql)

Kafka consumer 설정

- **IsKafkaConsumer**: Kafka Consumer 사용 여부
- **KafkaConnString**: Kafka 브로커의 주소 (예 : 123.456.789.101:9092)
- **MainProducdrTopic**: Kafka 프로듀서의 메인 토픽 이름
- **ConsumerTopics**: Kafka 컨슈머 토픽 이름
- **KafkaTimeout**: Kafka 연결 타임아웃 (밀리초)

  ElasticSearch 관련 설정

- **ElasticConnString**: ElasticSearch 연결 주소

Conf / Config_select.json

config_select.json : 프로젝트의 설정 파일을 선택하기 위한 설정 파일

weberp-queries (표준 쿼리 파일)

쿼리 파일이 따로 존재하는 이유

1. 쿼리가 빠져나와있어 다양한 커스터마이징이 가능합니다 **(**아래에서 설명**)**

2. 데이터베이스 구조가 다른 환경에서도 표준 쿼리는 동일하게 유지할 수 있습니다**.**

3. 표준 쿼리와 커스터마이징 된 쿼리 간 충돌을 최소화할 수 있습니다**.**

(표준 쿼리 = 기본 동작 유지, 커스터마이징 쿼리 = 각 프로젝트의 요구사항 반영)

Weberp-queries / themes

각 프로젝트 목적에 맞게 커스터마이징이 가능한 쿼리 파일입니다**.**

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

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

Cache-key-pair

다보리 컴포저블(Dabory Composable)에서 **앱별로 발급되는 암호화된 키 쌍(key pair)**을 관리하기 위해 사용됩니다.

- 앱 등록 시 키 발급
  - 각 앱은 고유한 키 쌍(Public/Private Key)을 받습니다

    - 이 키 쌍은 앱 보안과 인증을 위한 중요한 역할을 합니다
- 서버에 기본 저장
    - 발급된 키 쌍은 기본적으로 서버에 저장됩니다
- 서버 장애 대비
    - 서버에 `cache-key-pair` 폴더에 저장
    - 이로 인해 서버 문제 발생 시에도 키 값을 복구할 수 있습니다

Cache-key-pair 디렉토리 용도

- 키 복구: 서버 장애 시 캐싱된 키로 복구 가능
- 보안 강화: 고유 키로 데이터를 안전하게 암호화/복호화
- 효율성: 캐싱된 키를 활용해 서버 요청을 줄이고 성능 향상

5. weberp-go 실행 `./weberp-go`를 입력하여 서버를 구동시킵니다.

이제 **18080** 포트로 백엔드 서버가 실행할 수 있습니다. **(local_config.json** 파일에서 **RestConnect**로 서버 포트 변경 가능**)**

Revision #10
Created 16 December 2024 07:32:02 by 하늘
Updated 20 December 2024 02:22:10 by 하늘