

로컬에서 빌드하기

목차

사전 설정

- 1. MySQL

- 2. Docker 2. Ganache 설정 3. Remix를 이용한 Contract 배포
- 4. [Frontend] .env 파일
- 3. [Backend] /main/resources 내 application.properties

사전 설정

1. MySQL

데이터베이스 생성

- charset: utf8
- collation: utf8_general_ci

2. Docker

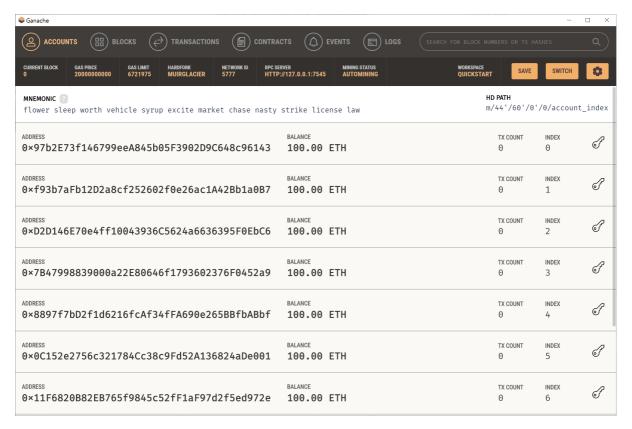
docker run --name sgbg-redis -p 6379:6379 redis

• token 저장을 위한 redis 컨테이너

2. Ganache 설정

두 가지 방법 중 한 가지 방법 선택

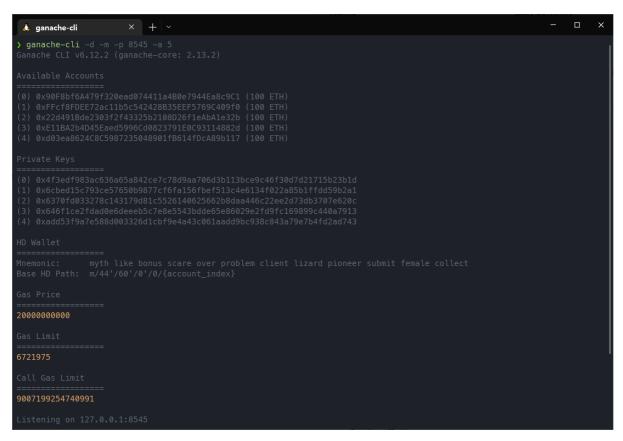
2.1 ganache client 사용하기



- [주의] 7545번 포트로 띄워짐
 - 。 application.properties에 기재하는 포트 번호 7545로 기재

2.2 (SGBG Default) ganache-cli 사용하기

ganache-cli -d -m -p 8545 -a 5



• 8545번 포트로 띄우기

3. Remix를 이용한 Contract 배포

smart_contract/cash.sol 배포 후, admin address와 private key 저장해두기

4. [Frontend] .env 파일

REACT_APP_KAKAO_KEY={KAKAO_KEY}
REACT_APP_KAKAO_API_KEY={KAKAO_API_KEY}

- KAKAO KEY: 카카오 로그인 CLIENT KEY
- KAKAO_API_KEY: 카카오 맵 API KEY

3. [Backend] /main/resources 내 application.properties

```
spring. data source.url=jdbc:mysql://127.0.0.1:3306/bcssafy?useUnicode=true\&characterEncoding=utf8\&serverTimezone=Asia/Seoul\&zeroDateTimeBehavior=convertToNull\&rewriteBatspring.datasource.driver-class-name=com.mysql.cj.jdbc.briverspring.datasource.hikari.username=root
spring.datasource.hikari.password=12345
# = LOGGING
# Logging pattern for the console
logging.pattern.console=%d{HH:mm:ss} %clr(%5p) [%c] %m%n
# Logging pattern for file
#logging.file = logs/backend.log
spring.web3j.client-address=http://127.0.0.1:8545\\ spring.web3j.admin-client=true
# = Ethereum CA & EOA
#eth.admin.wallet.filename=admin.wallet
# Kakao REST API
kakao.client.id=b833a7474795b71dfb16b78fbcac80be
kakao.client.secret=ciUaftxmuIwrIWhekVk5c0mtH9H4KsNc
kakao.redirect.uri=http://localhost:3000/login
# Spring doc
springdoc.swagger-ui.path=/index.html
springdoc.swagger-ui.groups-order=desc
springdoc.swagger-ui.operations-sorter=method
springoc.swagger-ul.operations-sorter=method springdoc.swagger-ui.disable-swagger-default-url=true springdoc.swagger-ui.display-request-duration=true springdoc.api-docs.path=/api-docs springdoc.default-consumes-media-type=application/json springdoc.default-produces-media-type=application/json springdoc.paths-to-match=/**
# Redis for Token
spring.redis.host=localhost
spring.redis.port=6379
# Admin Address
# Admin Address
eth.cash.contract={배포한_cash.sol_contract_address}
eth.admin.address={cash.sol에서_설정한_admin_address}
eth.admin.private={cash.sol에서_설정한_admin_private_key}
```