💻

[포팅 메뉴얼]

프로젝트 기술 스택

가. 이슈관리 : Jira

나. 형상관리 : GitLab

다. 커뮤니케이션 : Mattermost, Notion, webex

라. 개발환경

1. OS : windows11

2. IDE

- 인텔리제이 : IntelliJ IDEA 2022.1.3 (Ultimate Edition)
- visual studio code : 1.7.0

3. Database : MySQL Workbench 8.0 CE

4. Server : GCP `1.2.5.RELEASE`

5. 상세 내용

- Backend
  - Springboot : `org.springframework.boot` `2.7.1`
  - Java : 8
  - JDK : 1.8
  - Gradle : `gradle-7.4.1-bin`
  - lombok : 1.18.24
  - Java Data JPA : `spring-boot-starter-data-jpa`
  - MailSender : `1.15`
  - oauth2 : `spring-boot-starter-oauth2-client`
  - jwt : `jjwt-api` `0.11.2`
  - TemplateEngine : `thymeleaf-spring5` `3.0.11.RELEASE`
  - batch : `spring-boot-starter-batch`

- Frontend
  - node.js : 16.16.0(LTS)
  - npm : 8.11.0
  - vue.cli : 5.0.4
  - vuex : ^3.4.0
  - vuetify : ^2.6.0
  - HTML5, CSS3, JavaScript(ES6)
  - babel : ^7.12.16
  - animate.css : ^4.1.1
- 배포 및 서버
  - Jenkins
  - Nginx

백엔드 빌드 방법

1. Command 빌드 방법

- git clone “GIT REPOSITORY”
- cd S07P12A702/backend

- 빌드 : (Window) ./gradlew.bat → ./gradlew build, (MacOS) ./gradlew build
- cd build/libs
- 서버 실행 : java -jar octopus-o.o.1-SNAPSHOT.jar
- 서버 중지 : (Window) ctrl+c, (MacOS) ls -arlth
- 빌드 삭제 : (Window)./gradlew.bat clean build, (MacOS)./gradlew clean build

2. IntelliJ 빌드 방법

- git clone “GIT REPOSITORY”
- cd S07P12A702/backend
- backend 디렉토리 오른쪽 버튼 클릭 → Open Folder as IntelliJ IDEA Project
- src/main/resources/application-data.properties 추가

▼ application-data.properties

```
server.port = 7070(원하는 서버 포트)

spring.datasource.url=jdbc:mysql://localhost:3306/octopus?autoReconnect=true
spring.datasource.username=root(mysql db의 이름)
spring.datasource.password=Qoguswnd201!(mysql db 비밀번호)
spring.datasource.driver-class-name = com.mysql.cj.jdbc.Driver

jwt.header=Authorization
jwt.secret=c2lsdmVybmluZS10ZWNoLXNwcmluZy1ib290LWp3dC10dXRvcmlhbC1zZWNyZXQtc2lsdmVybmluZS10ZWNoLXNwcmluZy1ib290LWp3dC10dXRvcmlhb
jwt.token-validity-in-seconds=86400

spring.cloud.gcp.project-id=octopus-358301
spring.cloud.gcp.credentials.location=classpath:octopus-358301-c7e28c86140f.json

### SMTP ###
spring.mail.host=smtp.gmail.com
spring.mail.port=587
spring.mail.username=octopus.ssafy@gmail.com
spring.mail.password=isnmngtgqxtzseiu

spring.mail.properties.mail.smtp.auth=true

spring.mail.properties.mail.smtp.starttls.enable=true
spring.mail.properties.mail.smtp.starttls.required=true

bucketname = octopus-702

### Oauth2 ###
spring.security.oauth2.client.registration.kakao.client-id=9a4a29dd046d8945a94faa4566beb2f9
spring.security.oauth2.client.registration.kakao.client-secret=bs09zPBKZqXGtLsGSiqfirooN6up1yCX
spring.security.oauth2.client.registration.kakao.redirect-uri=http://localhost:8080/main
spring.security.oauth2.client.registration.kakao.client-authentication-method=POST
spring.security.oauth2.client.registration.kakao.authorization-grant-type=authorization_code
spring.security.oauth2.client.registration.kakao.scope=profile_nickname, profile_image, account_email
spring.security.oauth2.client.registration.kakao.client-name=Kakao

spring.security.oauth2.client.provider.kakao.authorization-uri = https://kauth.kakao.com/oauth/authorize
spring.security.oauth2.client.provider.kakao.token-uri = https://kauth.kakao.com/oauth/token
spring.security.oauth2.client.provider.kakao.user-info-uri = https://kapi.kakao.com/v2/user/me
spring.security.oauth2.client.provider.kakao.user-name-attribute = id

### Custom Oauth2 Data ###
kakao.token-uri=https://kauth.kakao.com/oauth/token
kakao.client-id=9a4a29dd046d8945a94faa4566beb2f9
kakao.client-secret=bs09zPBKZqXGtLsGSiqfirooN6up1yCX
kakao.redirect-uri=http://localhost:8080/main
```

- src/main/java/OctopusApplication 실행

프론트엔드 빌드 방법

1. node_modules를 위한 기본 install

```
npm install
```

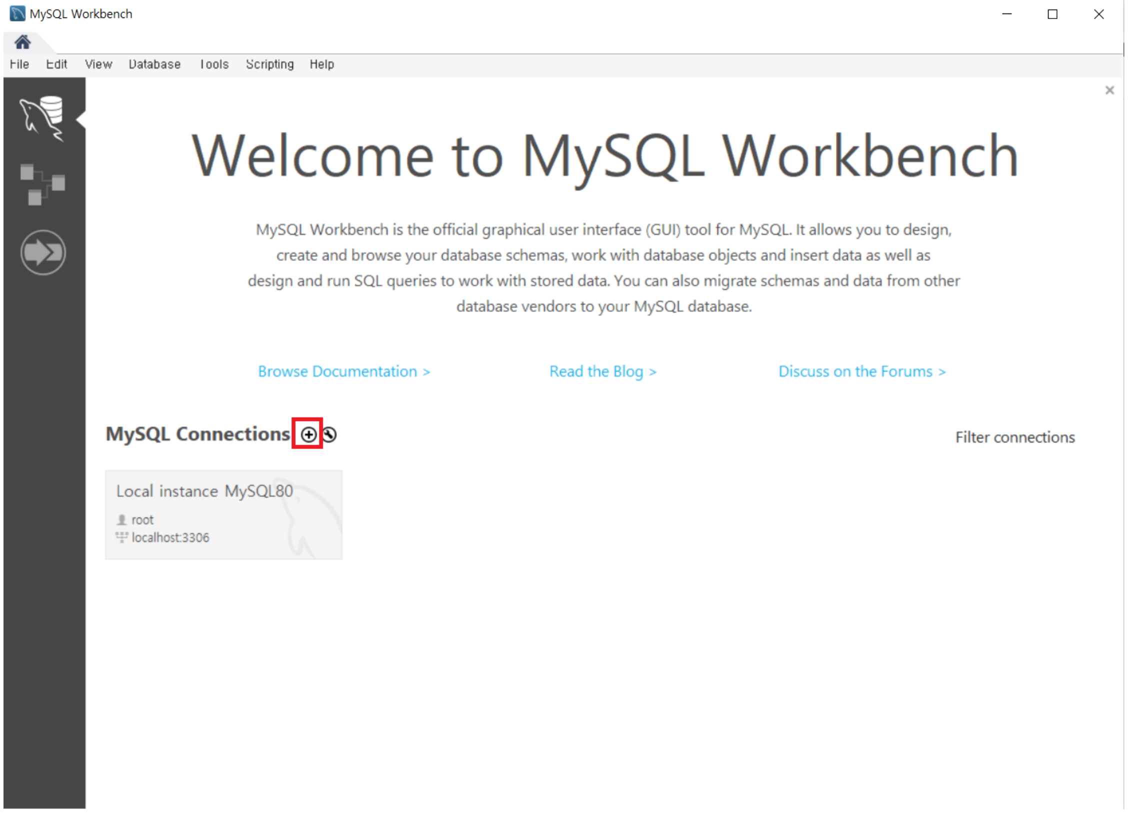
2. 현재 상태로 빌드하기

```
npm run build
```

배포 명령어 정리

MySQL 워크벤치 사용방법

1. MySQL WorkBench 추가하기

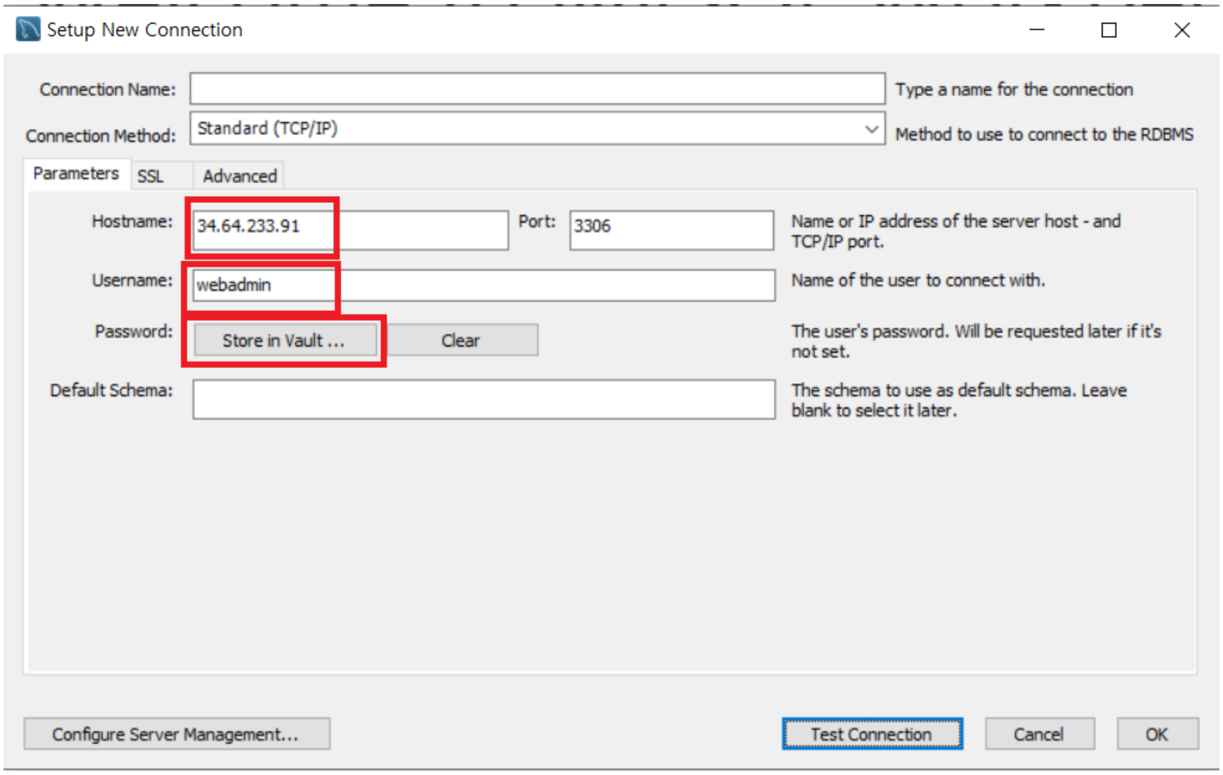


Welcome to MySQL Workbench

MySQL Workbench is the official graphical user interface (GUI) tool for MySQL. It allows you to design, create and browse your database schemas, work with database objects and insert data as well as design and run SQL queries to work with stored data. You can also migrate schemas and data from other database vendors to your MySQL database.

MySQL WorkBench에 Connection을 추가하기 위해 +버튼을 눌러줍니다.

2. GCP의 MySQL과의 연결


Setup New Connection

Connection Name: Type a name for the connection

Connection Method: Standard (TCP/IP) Method to use to connect to the RDBMS

Parameters | SSL | Advanced

Hostname: 34.64.233.91 Port: 3306 Name or IP address of the server host - and TCP/IP port.

Username: webadmin Name of the user to connect with.

Password: Store in Vault ... Clear The user's password. Will be requested later if it's not set.

Default Schema: The schema to use as default schema. Leave blank to select it later.

Configure Server Management... Test Connection Cancel OK

- username : `webadmin`
- password : `octopus702`

Nginx Default 값

가. nginx 설치 및 세팅

```
sudo apt-get update
sudo apt-get install nginx -y
```

나. nginx 프록시 서버 설정

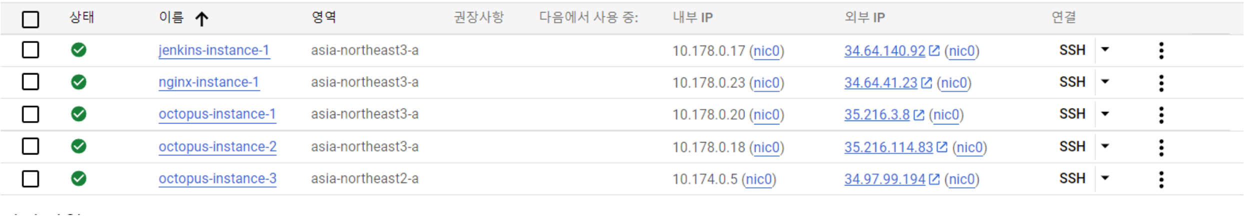
```
cd /etc/nginx/sites-available
```

sites-available.default

```
upstream octopus-app{
  server 10.178.0.20:8080 weight=100 max_fails=3 fail_timeout=3s; #instance 1
  server 10.178.0.18:8080 weight=100 max_fails=3 fail_timeout=3s; #instance 2
}

server {
        listen 80 default_server;
        listen [::]:80 default_server;

        root /var/www/html;

        index index.html index.htm index.nginx-debian.html;
        server_name _;

        location / {
                # First attempt to serve request as file, then
                # as directory, then fall back to displaying a 404.
                # try_files $uri $uri/ =404;
                try_files $uri $uri/ /index.html;
        }

        location /api {
                proxy_pass http://octopus-app;
                proxy_http_version 1.1;
                proxy_set_header Upgrade $http_upgrade;
                proxy_set_header Connection 'upgrade';
                proxy_set_header Host $host;
                proxy_cache_bypass $http_upgrade;
        }

        location /kauth.kakao.com {
                proxy_pass https://kauth.kakao.com/oauth/authorize?client_id=9a4a29dd046d8945a94faa4566beb2f9&redirect_uri=http://34.64.41.

                proxy_http_version 1.1;
                proxy_set_header Upgrade $http_upgrade;
                proxy_set_header Connection 'upgrade';
                proxy_set_header Host $host;
                proxy_cache_bypass $http_upgrade;

        }
}
```

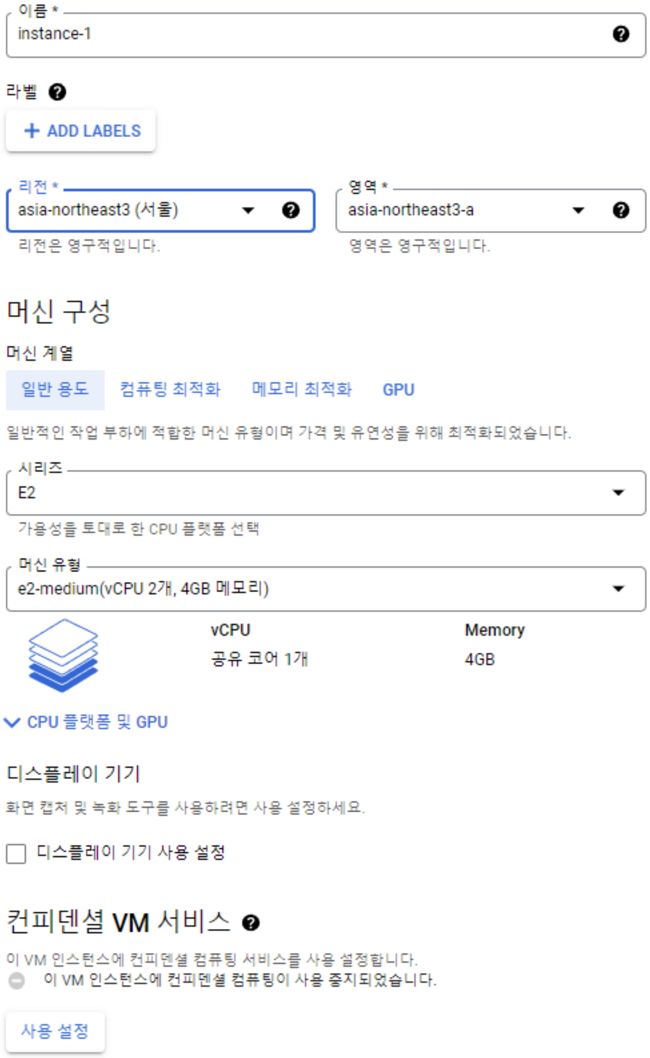
GCP 세팅

가. VM Instance



☐	상태	이름 ↑	영역	권장사항	다음에서 사용 중:	내부 IP	외부 IP	연결	
☐	✓	jenkins-instance-1	asia-northeast3-a			10.178.0.17 (nic0)	34.64.140.92 (nic0)	SSH ▾	⋮
☐	✓	nginx-instance-1	asia-northeast3-a			10.178.0.23 (nic0)	34.64.41.23 (nic0)	SSH ▾	⋮
☐	✓	octopus-instance-1	asia-northeast3-a			10.178.0.20 (nic0)	35.216.3.8 (nic0)	SSH ▾	⋮
☐	✓	octopus-instance-2	asia-northeast3-a			10.178.0.18 (nic0)	35.216.114.83 (nic0)	SSH ▾	⋮
☐	✓	octopus-instance-3	asia-northeast2-a			10.174.0.5 (nic0)	34.97.99.194 (nic0)	SSH ▾	⋮

나. VM Instance 생성



이름 *: instance-1

라벨

리전 *: asia-northeast3 (서울)

리전은 영구적입니다.

영역 *: asia-northeast3-a

영역은 영구적입니다.

머신 구성

머신 계열

일반적인 작업 부하에 적합한 머신 유형이며 가격 및 유연성을 위해 최적화되었습니다.

시리즈: E2

가용성을 토대로 한 CPU 플랫폼 선택

머신 유형: e2-medium(vCPU 2개, 4GB 메모리)

vCPU	Memory
공유 코어 1개	4GB

디스플레이 기기

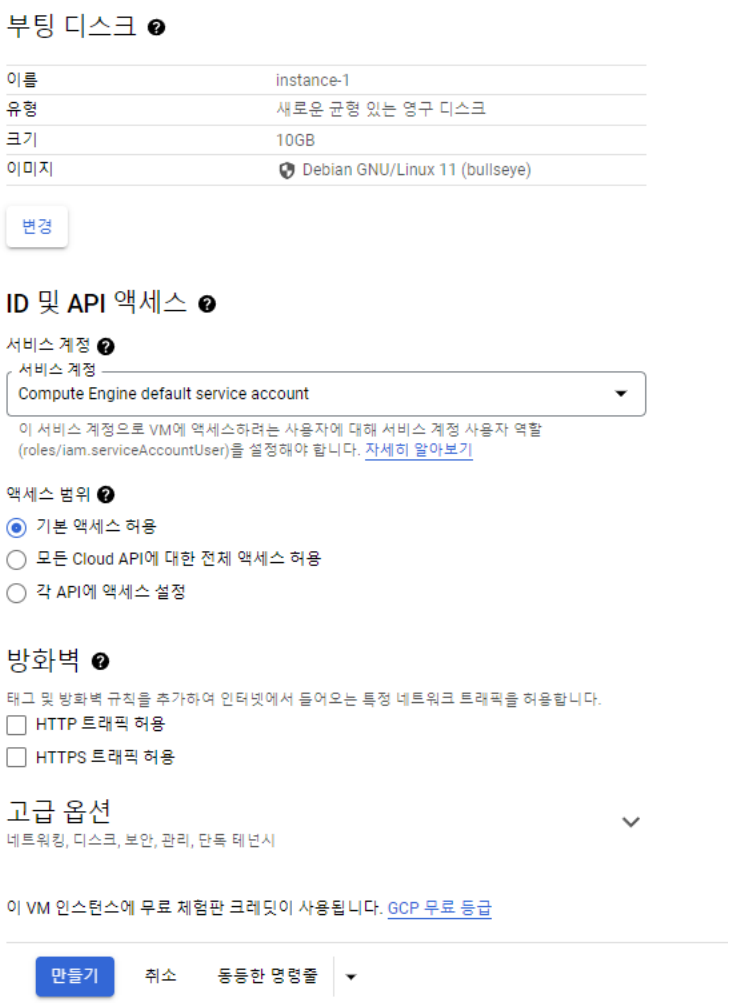
화면 캡처 및 녹화 도구를 사용하려면 사용 설정하세요.

□ 디스플레이 기기 사용 설정

컨피덴셜 VM 서비스

이 VM 인스턴스에 컨피덴셜 컴퓨팅 서비스를 사용 설정합니다.

이 VM 인스턴스에 컨피덴셜 컴퓨팅이 사용 중지되었습니다.



부팅 디스크

이름	instance-1
유형	새로운 균형 있는 영구 디스크
크기	10GB
이미지	Debian GNU/Linux 11 (bullseye)

ID 및 API 액세스

서비스 계정

서비스 계정: Compute Engine default service account

이 서비스 계정으로 VM에 액세스하려는 사용자에 대해 서비스 계정 사용자 역할 (roles/iam.serviceAccountUser)을 설정해야 합니다. 자세히 알아보기

액세스 범위

● 기본 액세스 허용

○ 모든 Cloud API에 대한 전체 액세스 허용

○ 각 API에 액세스 설정

방화벽

태그 및 방화벽 규칙을 추가하여 인터넷에서 들어오는 특정 네트워크 트래픽을 허용합니다.

□ HTTP 트래픽 허용

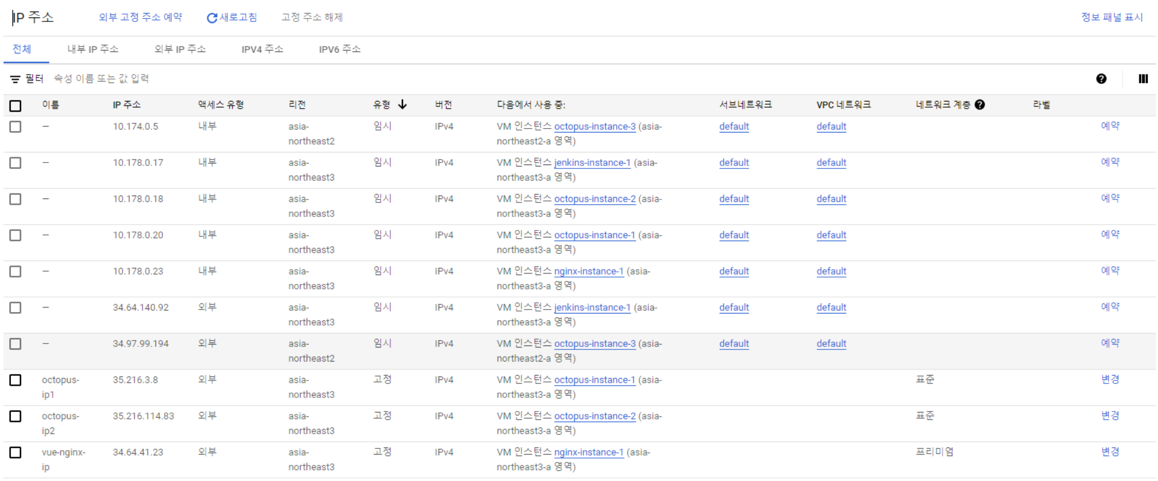
□ HTTPS 트래픽 허용

고급 옵션

네트워킹, 디스크, 보안, 관리, 단독 테넌시

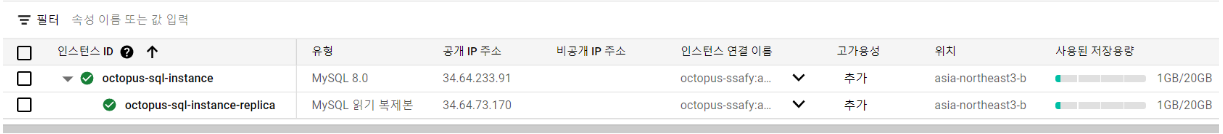
이 VM 인스턴스에 무료 체험판 크레딧이 사용됩니다. GCP 무료 등급

다. VPC IP 설정



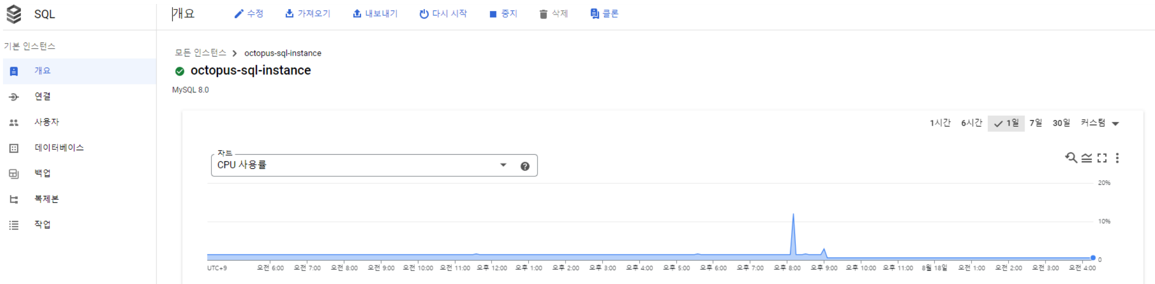
이름	IP 주소	액세스 유형	리전	유형	버전	다음에서 사용 중:	서브네트워크	VPC 네트워크	네트워크 계층	라벨	
–	10.174.0.5	내부	asia-northeast2	임시	IPv4	VM 인스턴스 octopus-instance-3 (asia-northeast2-a 영역)	default	default			예약
–	10.178.0.17	내부	asia-northeast3	임시	IPv4	VM 인스턴스 jenkins-instance-1 (asia-northeast3-a 영역)	default	default			예약
–	10.178.0.18	내부	asia-northeast3	임시	IPv4	VM 인스턴스 octopus-instance-2 (asia-northeast3-a 영역)	default	default			예약
–	10.178.0.20	내부	asia-northeast3	임시	IPv4	VM 인스턴스 octopus-instance-1 (asia-northeast3-a 영역)	default	default			예약
–	10.178.0.23	내부	asia-northeast3	임시	IPv4	VM 인스턴스 nginx-instance-1 (asia-northeast3-a 영역)	default	default			예약
–	34.64.140.92	외부	asia-northeast3	임시	IPv4	VM 인스턴스 jenkins-instance-1 (asia-northeast3-a 영역)	default	default			예약
–	34.97.99.194	외부	asia-northeast2	임시	IPv4	VM 인스턴스 octopus-instance-3 (asia-northeast2-a 영역)	default	default			예약
octopus-ip1	35.216.3.8	외부	asia-northeast3	고정	IPv4	VM 인스턴스 octopus-instance-1 (asia-northeast3-a 영역)			표준		변경
octopus-ip2	35.216.114.83	외부	asia-northeast3	고정	IPv4	VM 인스턴스 octopus-instance-2 (asia-northeast3-a 영역)			표준		변경
vue-nginx-ip	34.64.41.23	외부	asia-northeast3	고정	IPv4	VM 인스턴스 nginx-instance-1 (asia-northeast3-a 영역)			프리미엄		변경

다. SQL 인스턴스



필터 속성 이름 또는 값 입력

인스턴스 ID	유형	공개 IP 주소	비공개 IP 주소	인스턴스 연결 이름	고가용성	위치	사용된 저장용량
octopus-sql-instance	MySQL 8.0	34.64.233.91		octopus-ssafy:a...	추가	asia-northeast3-b	1GB/20GB
octopus-sql-instance-replica	MySQL 읽기 복제본	34.64.73.170		octopus-ssafy:a...	추가	asia-northeast3-b	1GB/20GB



모든 인스턴스 > octopus-sql-instance

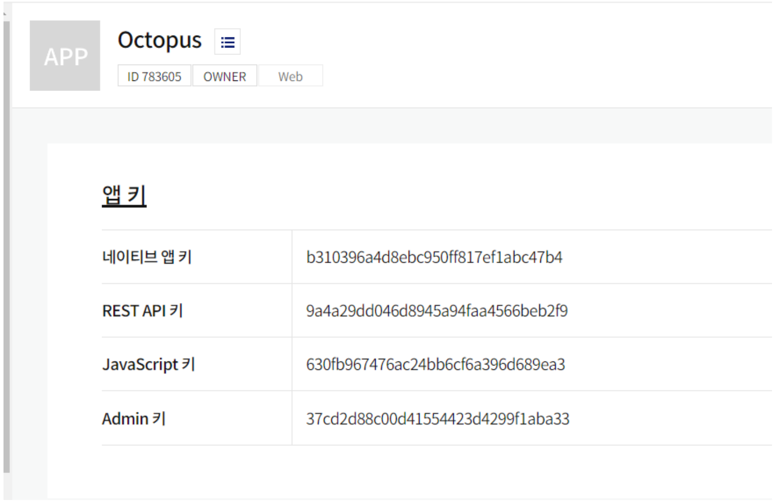
octopus-sql-instance

MySQL 8.0

차트: CPU 사용률

Kakao Dev 설정

가. REST-API Key 등록



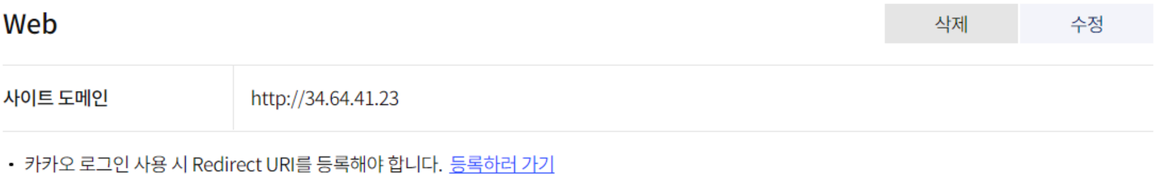
Octopus

ID 783605 | OWNER | Web

앱 키

네이티브 앱 키	b310396a4d8ebc950ff817ef1abc47b4
REST API 키	9a4a29dd046d8945a94faa4566beb2f9
JavaScript 키	630fb967476ac24bb6cf6a396d689ea3
Admin 키	37cd2d88c00d41554423d4299f1aba33

나. 카카오 플랫폼 도메인 등록

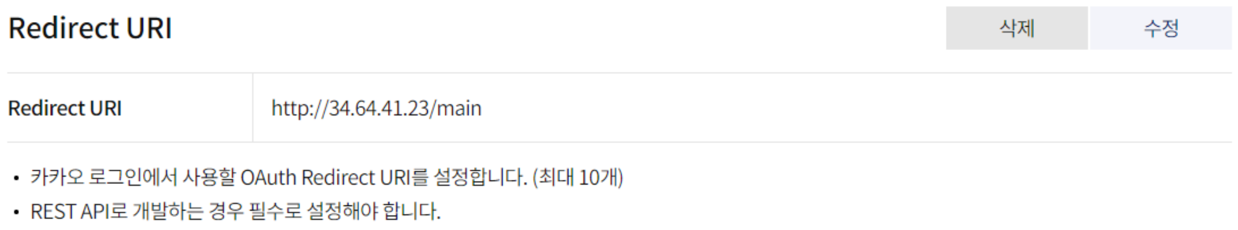


Web

사이트 도메인	http://34.64.41.23

- 카카오 로그인 사용 시 Redirect URI를 등록해야 합니다. 등록하러 가기

다. Redirect URI 등록



Redirect URI

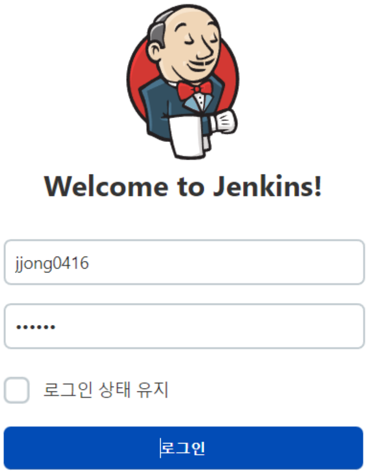
Redirect URI	http://34.64.41.23/main

- 카카오 로그인에서 사용할 OAuth Redirect URI를 설정합니다. (최대 10개)
- REST API로 개발하는 경우 필수로 설정해야 합니다.

Jenkins 설정

가. Jenkins 아이디/패스워드

```
아이디 : jjong0416
패스워드 : as5681
```



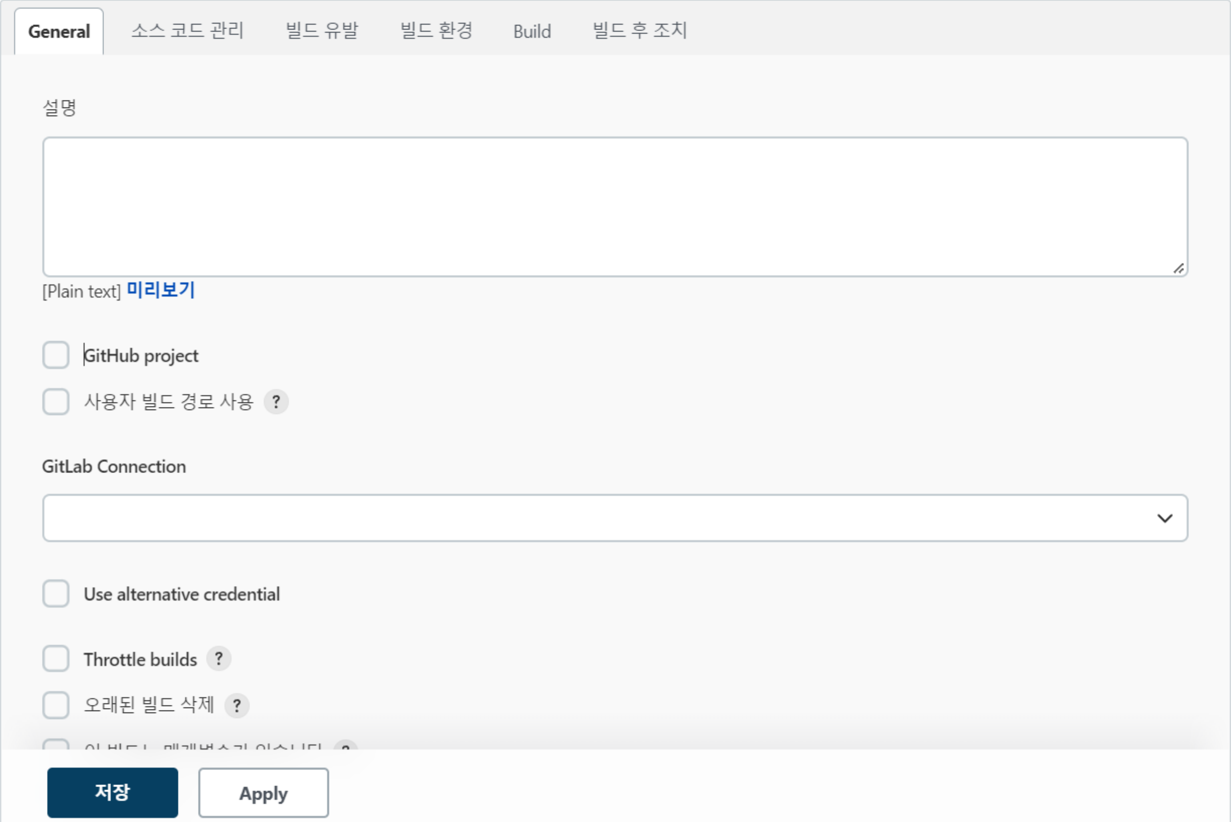
Welcome to Jenkins!

jjong0416

••••••

□ 로그인 상태 유지

나. Jenkins 설정



General | 소스 코드 관리 | 빌드 유발 | 빌드 환경 | Build | 빌드 후 조치

설명

[Plain text] 미리보기

□ GitHub project

□ 사용자 빌드 경로 사용 ?

GitLab Connection

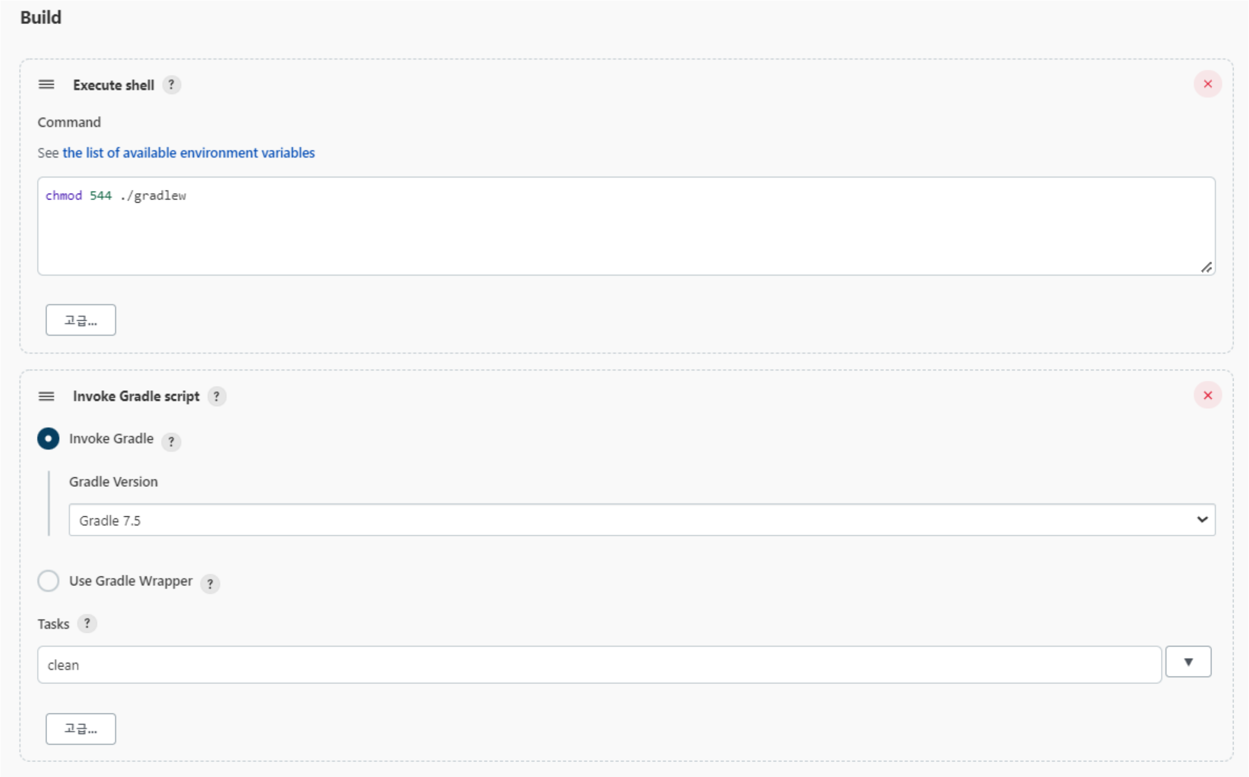
□ Use alternative credential

□ Throttle builds ?

□ 오래된 빌드 삭제 ?

□ 이 빌드는 매개변수가 있습니다 ?

저장 | Apply



Build

Execute shell ?

Command

See the list of available environment variables

```
chmod 544 ./gradlew
```

고급...

Invoke Gradle script ?

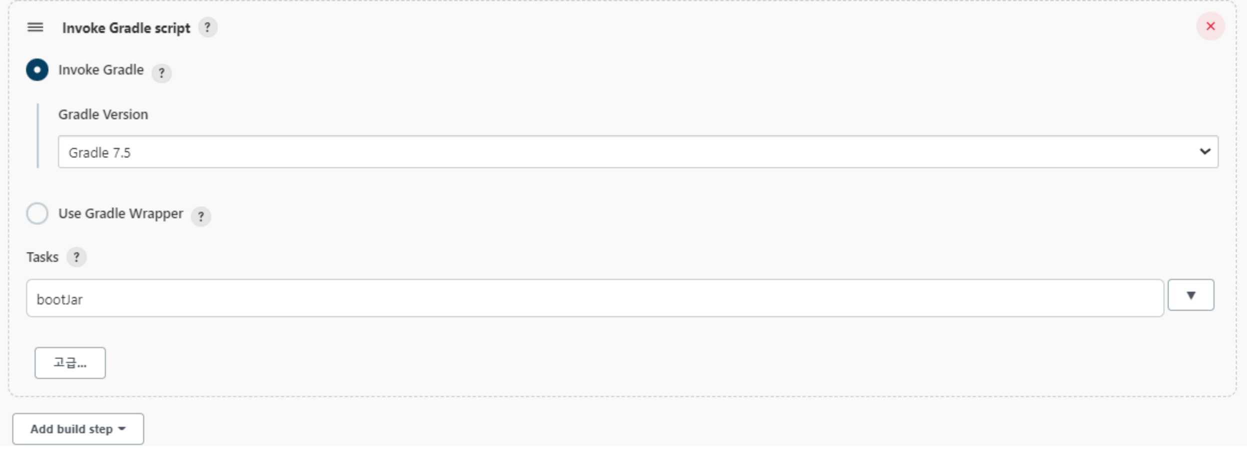
● Invoke Gradle ?

Gradle Version: Gradle 7.5

○ Use Gradle Wrapper ?

Tasks ?: clean

고급...



● Invoke Gradle

Gradle Version: Gradle 7.5

○ Use Gradle Wrapper

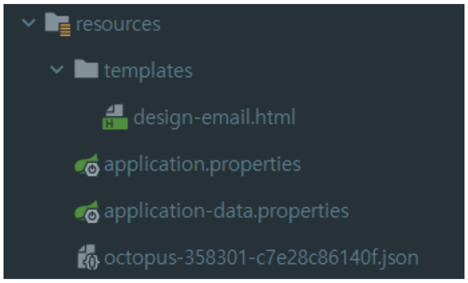
Tasks: bootJar

GCP Storage 설정

가. GCP Storage Json 발급

```
{
  "type": "service_account",
  "project_id": "octopus-358301",
  "private_key_id": "c7e28c86140fdf836631a5cbf89bb7917984dfad",
  "private_key": "-----BEGIN PRIVATE KEY-----\nMIIEvgIBADANBgkqhkiG9w0BAQEFAASCBKgwggSkAgEAAoIBAQCnq03DpOvcadqf\nzzw+69W84CZVDTOm81rlysw5Fl
  "client_email": "octopus702@octopus-358301.iam.gserviceaccount.com",
  "client_id": "101064939768886310194",
  "auth_uri": "https://accounts.google.com/o/oauth2/auth",
  "token_uri": "https://oauth2.googleapis.com/token",
  "auth_provider_x509_cert_url": "https://www.googleapis.com/oauth2/v1/certs",
  "client_x509_cert_url": "https://www.googleapis.com/robot/v1/metadata/x509/octopus702%40octopus-358301.iam.gserviceaccount.com"
}
```

나. key file resource에 등록



```
resources
  templates
    design-email.html
  application.properties
  application-data.properties
  octopus-358301-c7e28c86140f.json
```

다. application.properties 키 등록

```
spring.cloud.gcp.project-id=octopus-358301
spring.cloud.gcp.credentials.location=classpath:octopus-358301-c7e28c86140f.json
bucketname = octopus-702
```

Git Ignore

1. 백엔드

▼ git Ignore

```
HELP.md
.gradle
build/
!gradle/wrapper/gradle-wrapper.jar
!**/src/main/**/build/
!**/src/test/**/build/

### STS ###
.apt_generated
.classpath
.factorypath
.project
.settings
.springBeans
.sts4-cache
bin/
!**/src/main/**/bin/
!**/src/test/**/bin/

### IntelliJ IDEA ###
.idea
*.iws
*.iml
*.ipr
out/
!**/src/main/**/out/
!**/src/test/**/out/
**/application-oauth.properties
**/application-data.properties
**/application-jwt.properties

### NetBeans ###
/nbproject/private/
/nbbuild/
/dist/
/nbdist/
/.nb-gradle/

### VS Code ###
.vscode/
```

2. 프론트엔드

▼ git Ignore

```
.DS_Store
node_modules
/dist


# local env files
.env.local
.env.*.local

# Log files
npm-debug.log*
yarn-debug.log*
yarn-error.log*
pnpm-debug.log*

# Editor directories and files
.idea
.vscode
*.suo
*.ntvs*
*.njsproj
*.sln
*.sw?
```