

배포

1. EC2 설정

- XShell 다운
- 새 세션 만들기
- **연결-호스트**에 AWS EC2 인스턴스의 **퍼블릭 도메인**을 입력
- **연결-사용자인증**의 public key 설정 클릭
- AWS IAM 설정 .pem 파일을 사용자 키로 지정하고 확인
- 새 세션 더블클릭
- EC2 instance에 맞춰서 유저이름 입력
- 세션 연결 성공

2. Java

- `sudo apt-get install java-1.8.0-openjdk`

3. MySQL

- `sudo apt-get install mysql-server`
- `sudo systemctl start mysql`
- `sudo systemctl enable mysql`

4. FrontEnd

- React 프로젝트 폴더로 이동
- `sudo npm install`
- `sudo npm run build`

5. BackEnd

- SpringBoot 프로젝트 폴더로 이동
- src/main/resources에 aws.properties와 datasource.properties 생성
- src/main/resources에 아래 파일들 설정

▼ application.properties

```
spring.jpa.hibernate.ddl-auto=update  
  
spring.jpa.show-sql=true
```

```
spring.servlet.multipart.max-file-size=-1
spring.servlet.multipart.max-request-size=-1

spring.jwt.secret=MySecretKey!QSW@DERCZ^3yrH.&3
```

▼ aws.properties

```
cloud.aws.s3.bucket=cocktailus
cloud.aws.region.static=ap-northeast-2
cloud.aws.credentials.access-key=AKIART7AMTPA6KTRNJGK
cloud.aws.credentials.secret-key=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
cloud.aws.stack.auto=false

logging.level.com.amazonaws.util.EC2MetadataUtils=error
```

▼ datasource.properties

```
spring.datasource.url=jdbc:mysql://localhost:3306/cocktailer?serverTimezone=UTC
spring.datasource.username=root
spring.datasource.password=1234
spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver
```

- `sudo vim /etc/systemd/system/springboot.service`

```
#springboot.service
[Unit]
Description=Cocktailus Service
After=network.target

[Service]
ExecStart=/bin/bash -c "exec java -jar @build경로@"

User=root
Group=root

[Install]
WantedBy=multi-user.target
```

- `[Service]` 의 `ExecStart=/bin/bash -c "exec java -jar @build경로@"` 의 `@build경로@` 에 SpringBoot build jar파일 경로 넣기
- SpringBoot 프로젝트 폴더에서 `sudo gradle build`
- `sudo systemctl start springboot.service`
- `sudo systemctl enable springboot`

6. Nginx

- `sudo apt-get install nginx`
- `sudo vim /etc/nginx/sites-enabled/default`

```
server {  
    listen 80 default_server;  
    listen [::]:80 default_server;  
  
    root /var/www/html;  
    index index.nginx-debian.html;  
    root /var/www/html;  
  
    index index.nginx-debian.html;  
    server_name _;  
  
    location / {  
        root    @build경로@;  
        index   index.html index.htm;  
        try_files $uri /index.html;  
    }  
}
```

location /의 `root @build경로@;` 의 @build경로@에 React 프로젝트 build 폴더 넣기

- `sudo service nginx start`