API gateway 실습

목차

Step 1. Pull the Kong Gateway Docker image

Pull the following Docker image.

```
docker pull kong/kong-gateway:2.6.0.0-alpine
```

..... 오류! 책갈피가 정의되어 있지 않습니다.

Step 2. Create a Docker network

Create a custom network to allow the containers to discover and communicate with each other.

```
docker network create kong-ee-net
```

..... 오류! 책갈피가 정의되어 있지 않습니다.

Step 3. Start a database

Start a PostgreSQL container:

```
docker run -d --name kong-ee-database \
  --network=kong-ee-net \
  -p 5432:5432 \
  -e "POSTGRES_USER=kong" \
  -e "POSTGRES_DB=kong" \
  -e "POSTGRES_PASSWORD=kong" \
  postgres:9.6
```

Step 4. Prepare the Kong database

```
docker run --rm --network=kong-ee-net \
  -e "KONG_DATABASE=postgres" \
  -e "KONG_PG_HOST=kong-ee-database" \
  -e "KONG_PG_PASSWORD=kong" \
  -e "KONG_PASSWORD={PASSWORD}" \
  kong-ee kong migrations bootstrap
```

..... 오류! 책갈피가 정의되어 있지 않습니다.

Step 5. Start the gateway with Kong Manager

> ⚠ **Important:** The settings below are intended for non-production use **only**, as they override the default `admin_listen` setting to listen for requests from any source. **Do not** use these settings in environments directly exposed to the internet.
>
> If you need to expose the `admin_listen` port to the internet in a production environment, secure it with authentication.

```
  -e "KONG_PG_HOST=kong-ee-database" \
  -e "KONG_PG_PASSWORD=kong" \
  -e "KONG_PROXY_ACCESS_LOG=/dev/stdout" \
  -e "KONG_ADMIN_ACCESS_LOG=/dev/stdout" \
  -e "KONG_PROXY_ERROR_LOG=/dev/stderr" \
  -e "KONG_ADMIN_ERROR_LOG=/dev/stderr" \
  -e "KONG_ADMIN_LISTEN=0.0.0.0:8001" \
  -e "KONG_ADMIN_GUI_URL=http://{HOSTNAME}:8002" \
    -p 8000:8000 \
    -p 8443:8443 \
    -p 8001:8001 \
    -p 8444:8444 \
    -p 8002:8002 \
    -p 8445:8445 \
    -p 8003:8003 \
    -p 8004:8004 \
    kong-ee
```

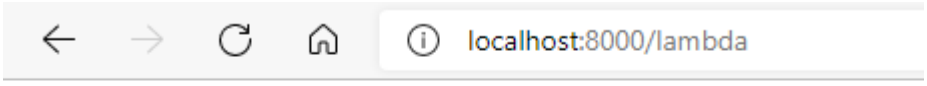
.....5

kong		latest	d5ef5ae48a05	about 1 month ago	141.63 MB
kong-ee	IN USE	latest	f73eac3208ed	26 days ago	314.63 MB
kong/kong-gateway	IN USE	2.6.0.0-alpine	f73eac3208ed	26 days ago	314.63 MB
pantsel/konga		latest	113950dafdbb	over 1 year ago	408.68 MB
postgres	IN USE	9.6	8f39a959063c	14 days ago	199.77 MB

..... **오류! 책갈피가 정의되어 있지 않습니다.**

```
C:₩Users₩dhd65>curl -I http://localhost:8001
HTTP/1.1 200 OK
Date: Wed, 10 Nov 2021 03:36:22 GMT
Content-Type: application/json; charset=utf-8
Connection: keep-alive
Access-Control-Allow-Origin: http://song:8002
X-Kong-Admin-Request-ID: cxisCVOe3jkVmPrpjlbACo5JUsQlEy9E
vary: Origin
Access-Control-Allow-Credentials: true
Content-Length: 16708
X-Kong-Admin-Latency: 5
Server: kong/2.6.0.0-enterprise-edition
```

.....6



"Hello from Lambda!"

.....오류! 책갈피가 정의되어 있지 않습니다.

1. Installation(docker)

Docker를 통한 Kong Api gateway에 관련된 시스템들을 컨테이너 형태로 실행하며 진행한다.

Step 1. Pull the Kong Gateway Docker image

Pull the following Docker image.

```
docker pull kong/kong-gateway:2.6.0.0-alpine
```

그림 1 도커 이미지 받아오기

Step 2. Create a Docker network

Create a custom network to allow the containers to discover and communicate with each other.

```
docker network create kong-ee-net
```

그림 2 도커 네트워크 설정

Step 3. Start a database

Start a PostgreSQL container:

```
docker run -d --name kong-ee-database \
  --network=kong-ee-net \
  -p 5432:5432 \
  -e "POSTGRES_USER=kong" \
  -e "POSTGRES_DB=kong" \
  -e "POSTGRES_PASSWORD=kong" \
  postgres:9.6
```

Step 4. Prepare the Kong database

```
docker run --rm --network=kong-ee-net \
  -e "KONG_DATABASE=postgres" \
  -e "KONG_PG_HOST=kong-ee-database" \
  -e "KONG_PG_PASSWORD=kong" \
  -e "KONG_PASSWORD={PASSWORD}" \
  kong-ee kong migrations bootstrap
```

그림 3 Database 준비

Step 5. Start the gateway with Kong Manager

> ⚠ **Important:** The settings below are intended for non-production use **only**, as they override the default `admin_listen` setting to listen for requests from any source. **Do not** use these settings in environments directly exposed to the internet.
>
> If you need to expose the `admin_listen` port to the internet in a production environment, secure it with authentication.

```
  -e "KONG_PG_HOST=kong-ee-database" \
  -e "KONG_PG_PASSWORD=kong" \
  -e "KONG_PROXY_ACCESS_LOG=/dev/stdout" \
  -e "KONG_ADMIN_ACCESS_LOG=/dev/stdout" \
  -e "KONG_PROXY_ERROR_LOG=/dev/stderr" \
  -e "KONG_ADMIN_ERROR_LOG=/dev/stderr" \
  -e "KONG_ADMIN_LISTEN=0.0.0.0:8001" \
  -e "KONG_ADMIN_GUI_URL=http://{HOSTNAME}:8002" \
    -p 8000:8000 \
    -p 8443:8443 \
    -p 8001:8001 \
    -p 8444:8444 \
    -p 8002:8002 \
    -p 8445:8445 \
    -p 8003:8003 \
    -p 8004:8004 \
    kong-ee
```

그림 4 Kong Manager 실행

결과

kong		latest	d5ef5ae48a05	about 1 month ago	141.63 MB
kong-ee	IN USE	latest	f73eac3208ed	26 days ago	314.63 MB
kong/kong-gateway	IN USE	2.6.0.0-alpine	f73eac3208ed	26 days ago	314.63 MB
pantsel/konga		latest	113950dafdbb	over 1 year ago	408.68 MB
postgres	IN USE	9.6	8f39a959063c	14 days ago	199.77 MB

그림 5 이미지 파일

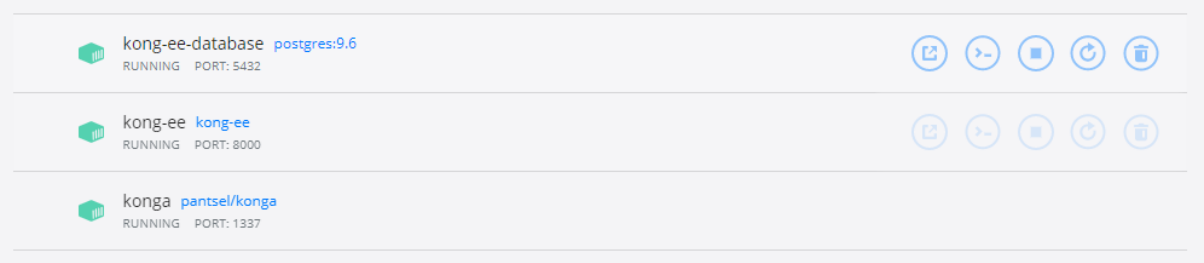


그림 6 컨테이너 실행 상태

그림 6은 현 실습을 실행하기 위하여 실행 중인 컨테이너들을 보여준다.

```
C:₩Users₩dhd65>curl -I http://localhost:8001
HTTP/1.1 200 OK
Date: Wed, 10 Nov 2021 03:36:22 GMT
Content-Type: application/json; charset=utf-8
Connection: keep-alive
Access-Control-Allow-Origin: http://song:8002
X-Kong-Admin-Request-ID: cxisCVOe3jkVmPrpjlbACo5JUsQlEy9E
vary: Origin
Access-Control-Allow-Credentials: true
Content-Length: 16708
X-Kong-Admin-Latency: 5
Server: kong/2.6.0.0-enterprise-edition
```

그림 7 kong api gateway 실행 확인

2. Konga 사용

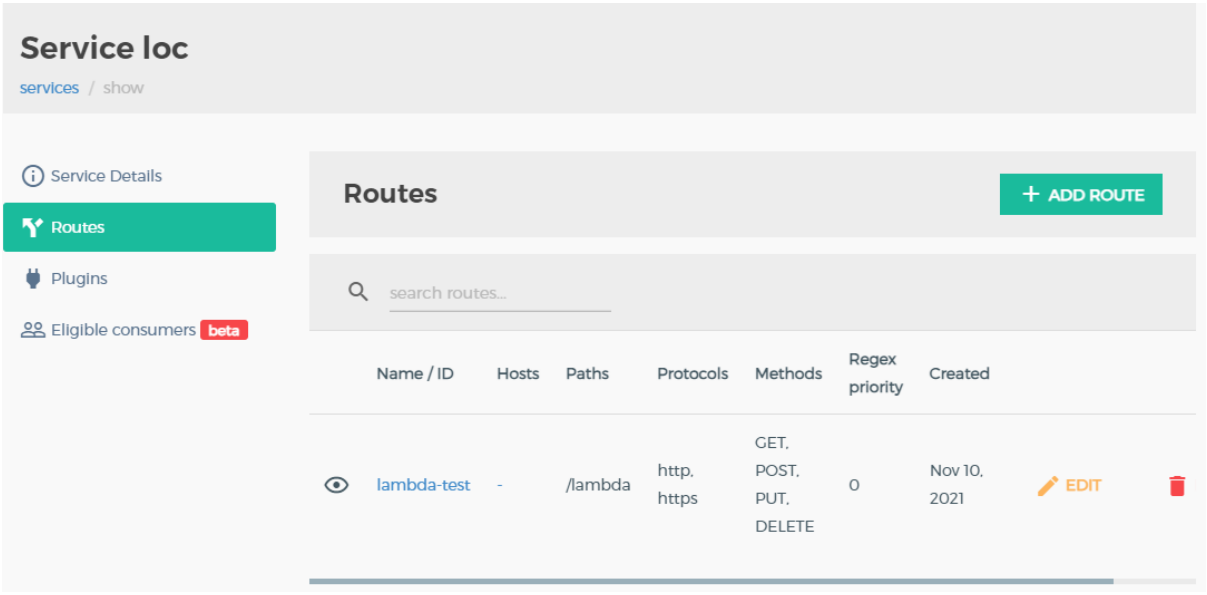


그림 8 Route 설정

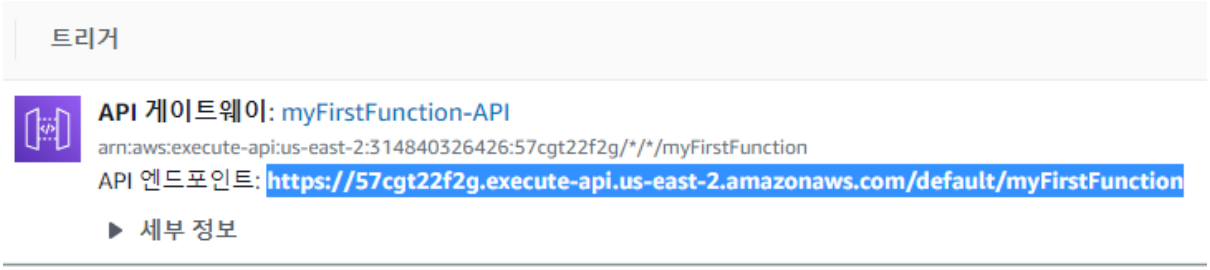
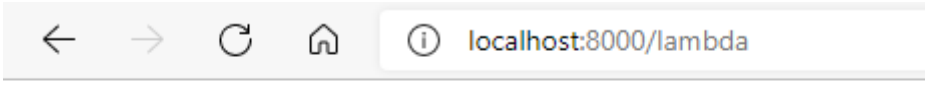


그림 9 lambda의 엔드포인트



"Hello from Lambda!"

그림 10 실행 결과

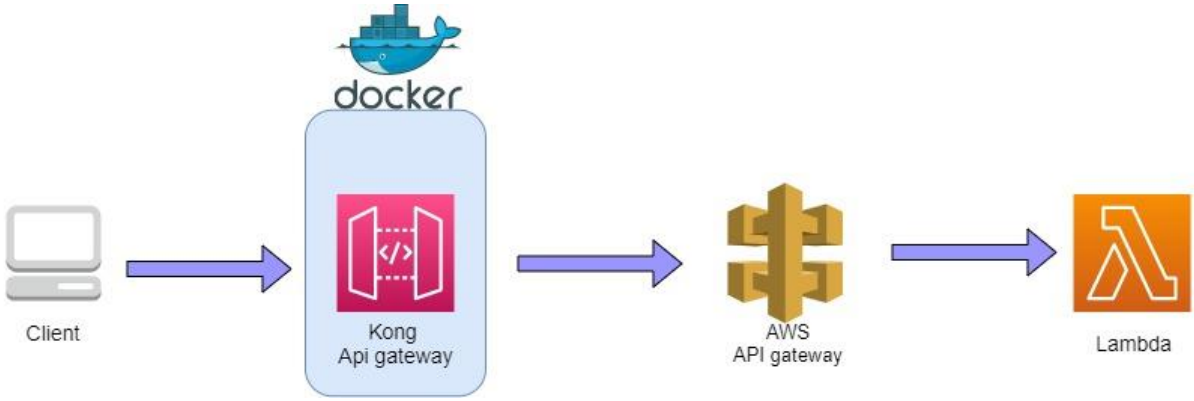


그림 11 실습 구조-1

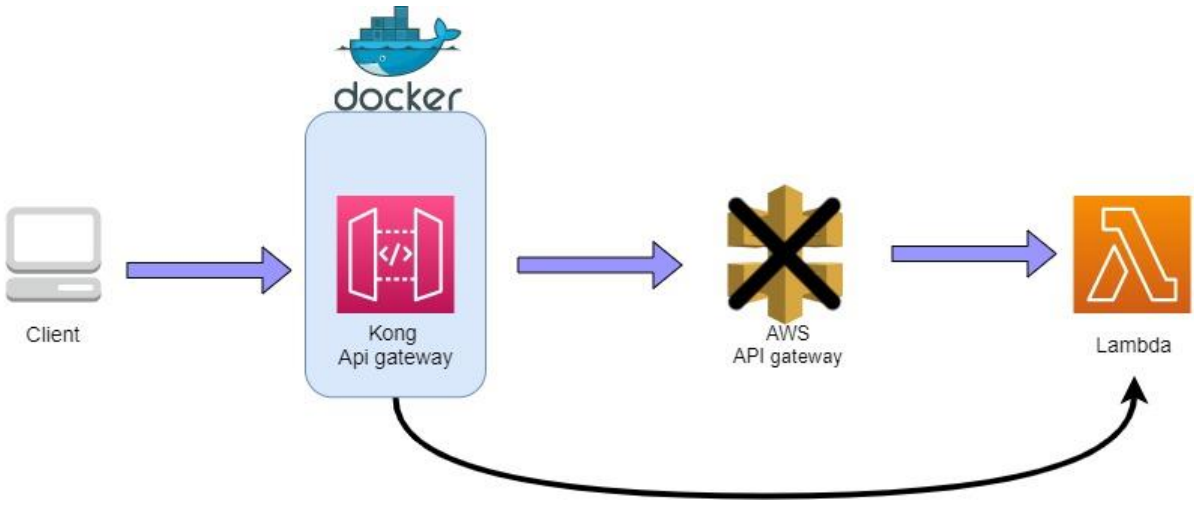


그림 12 실습 구조-2

그림 11과 그림 12는 현재 실습 구조를 보여주며, Kong Api gateway를 통하여 client의 호출이 AWS Api gateway에 도달한후 이후 람다에 호출이 가는 구조 혹은 end point를 가진 Kong Api gateway를 통하여 lambda에 호출을 보내는지 확실하지 않다.