

# 10. Go를 활용한 Kubernetes CLI 개발

## 05 Kubernetes CLI 빌드 및 실행

## 실습 내용

### 05. Kubernetes CLI 빌드 및 실행

#### 순서

1. go 언어 기반 Microservice 컨테이너 빌드 및 AWS ECR Push
2. go 언어 기반 모듈로 Kubernetes Custom CLI 빌드 및 실행
3. 생성된 K8s Manifest 배포 및 기동

#### 예제코드 경로

#### Chapter10

## 1. go 언어 기반 Microservice 컨테이너 빌드 및 AWS ECR Push

### (1) go 언어 기반 Microservice 컨테이너 빌드 명령어

- 소스코드 경로 : Chapter10 > **container**

```
$ docker build -t <AWS ECR Repository명>:<Tag명> .
```

### (2) AWS ECR 로그인

```
$ aws ecr get-login-password --region ap-northeast-2 | docker login \
  --username AWS --password-stdin <AWS ECR Repository명>
```

### (3) AWS ECR로의 빌드된 컨테이너 Push

```
$ docker push <AWS ECR Repository명>:<Tag명>
```

## 2. go 언어 기반 모듈로 Kubernetes Custom CLI 빌드 및 실행

### (1) go 언어 기반 모듈로 Kubernetes CLI 빌드를 위한 Dependency Import

- 소스코드 경로 : Chapter10 > **cdk8s**

**\$ cdk8s import**

### (2) go 언어 기반 모듈로 Kubernetes Custom CLI 빌드 및 실행

**\$ cdk8s synth**

### 3. 생성된 K8s Manifest 배포 및 기동

#### (1) Kubernetes CLI 빌드 및 실행후 생성된 K8s Manifest 배포 및 기동

- 소스코드 경로 : Chapter10 > **cdk8s**

```
$ kubectl apply -f dist/<지정한 애플리케이션명>.k8s.yaml
```

#### (2) K8s Manifest 배포 후 정상 기동 확인

```
$ kubectl get pods
```

```
$ kubectl get services
```

#### (3) 웹 애플리케이션 접속

```
http://<Service의 EXTERNAL-IP(DNS주소명)>:80/edit/ANewPage
```