

# 1 개발자를 위한 Kubernetes 활용 소개

## 01 개발자를 위한 Kubernetes 활용 Orientation

# Kubernetes란?

01.

개발자를 위한  
Kubernetes 활용  
Orientation

**쿠버네티스**(Kubernetes, 쿠베르네테스, "K8s")는 **컨테이너**화된 애플리케이션의 자동 배포, 스케일링 등을 제공하는 **관리시스템**으로, 오픈 소스 기반이다. 원래 구글에 의해 설계되었고 현재 리눅스 재단에 의해 관리되고 있다. 목적은 여러 클러스터의 호스트 간에 애플리케이션 **컨테이너**의 **배치, 스케일링, 운영을 자동화**하기 위한 플랫폼을 제공하기 위함이다. 도커를 포함하여 일련의 컨테이너 도구들과 함께 동작한다.

출처 : <https://ko.wikipedia.org/wiki/%EC%BF%A0%EB%B2%84%EB%84%A4%ED%8B%B0%EC%8A%A4>

# Kubernetes 공식 홈페이지

01.

개발자를 위한  
Kubernetes 활용  
Orientation



<https://kubernetes.io/ko/>

# 개발자를 위한 Kubernetes 활용 컨셉

01.

개발자를 위한  
Kubernetes 활용  
Orientation

1. 클라우드 인프라 기반 컨테이너 플랫폼 환경 **설계** 및 **구축** 수행
2. 컨테이너 플랫폼 기반 **개발** 환경 구성
3. 컨테이너 플랫폼의 안정적인 **운영** 방안 마련

# 개발자를 위한 Kubernetes 활용 파트 구성

01.

개발자를 위한  
Kubernetes 활용  
Orientation

1. 개발자를 위한 Kubernetes 활용 소개
2. Terraform을 활용한 AWS EKS 생성
3. AWS EKS 기본 Plugin 구성 방법
4. Kubernetes Manifest 작성을 위한 Helm 활용
5. Kubernetes 배포를 위한 GitHub와 ArgoCD 활용
6. Kubernetes 안정성 강화 방법
7. Kubernetes Custom 관리 방법
8. Kubernetes 보안 강화 방법
9. Kubernetes 트러블 슈팅 방법
10. Go를 활용한 Kubernetes CLI 개발
11. Kubernetes 활용 미니프로젝트 수행 하기

# Terraform을 활용한 AWS EKS 생성

01.

개발자를 위한  
Kubernetes 활용  
Orientation

1. Terraform을 활용한 AWS EKS 생성 소개
2. 실습 환경 구성을 위한 AWS 네트워크 및 EKS 설계
3. [실습] AWS 웹 콘솔을 활용한 AWS EKS 생성
4. Terraform 소개 (Terraformer 포함)
5. [실습] Terraform을 활용한 AWS EKS 생성
6. [실습] Terraformer를 활용한 AWS EKS 관리

# Terraform, Terraformer 제공 사이트

01.

개발자를 위한  
Kubernetes 활용  
Orientation



<https://www.terraform.io/>



<https://github.com/GoogleCloudPlatform/terraformer>

# AWS EKS 기본 Plugin 구성 방법

01.

개발자를 위한  
Kubernetes 활용  
Orientation

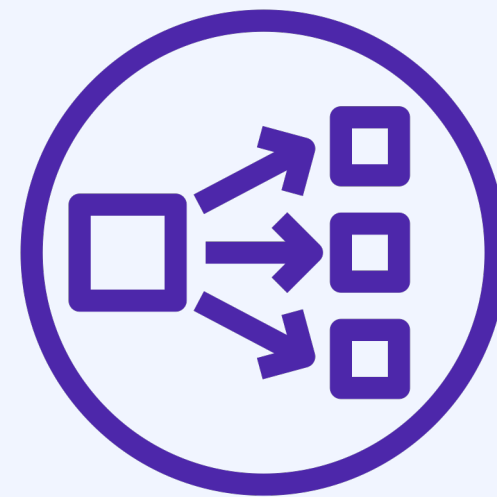
1. AWS **EKS 기본 설정** 방법 소개
2. [실습] **EKS Node 및 POD IP 대역 분리**(Secondary CIDR) 적용
3. [실습] AWS ALB Controller를 활용한 **Ingress ALB** 생성
4. [실습] AWS ALB Controller를 활용한 **Service NLB** 생성
5. [실습] AWS EBS CSI Driver를 활용한 **Block Storage PV** 생성
6. [실습] AWS EFS CSI Driver를 활용한 **File Storage PV** 생성
7. [실습] Cluster Autoscaler를 활용한 **EKS노드 오토스케일링**



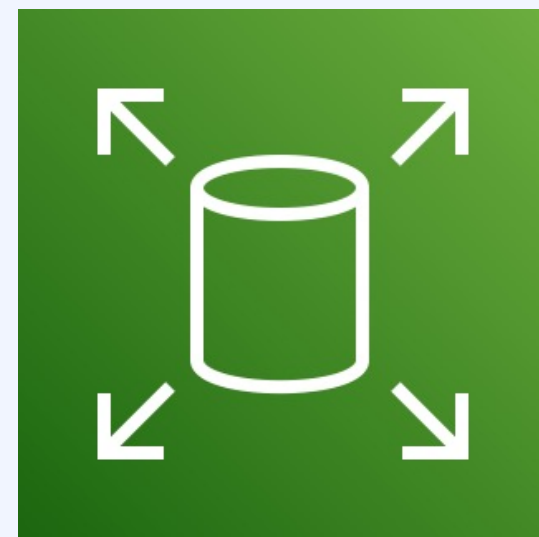
# AWS EKS Plugin 제공 사이트

01.

개발자를 위한  
Kubernetes 활용  
Orientation



<https://github.com/kubernetes-sigs/aws-load-balancer-controller>



<https://github.com/kubernetes-sigs/aws-ebs-csi-driver>



<https://github.com/kubernetes-sigs/aws-efs-csi-driver>

# Kubernetes Manifest 작성을 위한 Helm 활용

01.

개발자를 위한  
Kubernetes 활용  
Orientation

1. Kubernetes **Manifest 작성** 방법 소개
2. **Kustomize** 소개
3. [실습] **Kustomize**을 활용한 **Manifest 배포**하기
4. **Helm Charts** 소개
5. Helm **Values 및 Template** 소개
6. [실습] **Helm Charts**를 활용한 **Manifest 배포**하기

## Kustomize와 Helm의 제공 사이트

01.

개발자를 위한  
Kubernetes 활용  
Orientation



<https://kustomize.io/>



<https://helm.sh/>

# Kubernetes 배포를 위한 GitHub와 ArgoCD 활용

01.

개발자를 위한  
Kubernetes 활용  
Orientation

1. **Git**을 활용한 Kubernetes **배포** 방법 소개
2. [실습] **Github** Repository **생성 및 설정**
3. Kubernetes 배포 도구(**ArgoCD, ArgoRollout**) 소개
4. [실습] **ArgoCD** 구성
5. [실습] ArgoCD를 활용한 Kubernetes **Manifest 배포**
6. [실습] **ArgoRollout** 구성
7. [실습] Argo Rollout을 활용한 **무중단 배포**

# ArgoCD와 Argo Rollout의 제공 사이트

01.

개발자를 위한  
Kubernetes 활용  
Orientation



<https://argo-cd.readthedocs.io/en/stable/>



<https://argoproj.github.io/argo-rollouts/>

# Kubernetes 안정성 강화 방법

01.

개발자를 위한  
Kubernetes 활용  
Orientation

1. Kubernetes **안정성 강화** 방법 소개
2. [실습] **HPA** 소개 및 Metrics Server 설치
3. [실습] HPA 및 **오토스케일링** 적용
4. [실습] **Minio** 소개 및 설치
5. [실습] Minio 및 AWS **S3** 연결후 **업로드/다운로드** 적용
6. [실습] **Velero** 소개 및 설치
7. [실습] Velero **백업** 적용
8. [실습] Velero **복구** 적용

## minio와 velero의 제공 사이트

01.

개발자를 위한  
Kubernetes 활용  
Orientation



<https://min.io/>



<https://velero.io/>

# Kubernetes Custom 관리 방법

01.

개발자를 위한  
Kubernetes 활용  
Orientation

1. Kubernetes **Custom 관리** 방법 소개
2. Custom Resource Definition(**CRD**) 소개
3. Kubernetes **Operator** 소개
4. [실습] Kubernetes **Operator** 적용

<https://kubernetes.io/ko/docs/concepts/extend-kubernetes/operator/>



# Kubernetes 보안 강화 방법

01.

개발자를 위한  
Kubernetes 활용  
Orientation

1. Kubernetes **보안 강화** 활용 소개
2. [실습] **kube2iam** 소개 및 설치
3. [실습] kube2iam를 활용한 AWS **IAM** 기반 **권한 관리**
4. [실습] **Falco** 소개 및 설치
5. [실습] Falco를 활용한 **런타임 보안 강화**
6. [실습] **OPA Gatekeeper** 소개 및 설치
7. [실습] OPA Gatekeeper를 활용한 **보안 정책 관리**
8. [실습] **cert-manager** 소개 및 설치
9. [실습] cert-manager를 활용한 **TLS 인증서 관리**
10. [실습] AWS **ACM** 활용 **TLS 인증서 관리**

# kube2iam, falco, opa, cert-mgr 제공 사이트

01.

개발자를 위한  
Kubernetes 활용  
Orientation



<https://github.com/jtblin/kube2iam>



<https://falco.org/ko/>



<https://github.com/open-policy-agent/gatekeeper>



<https://cert-manager.io/docs/>

# Kubernetes 트러블 슈팅 방법

01.

개발자를 위한  
Kubernetes 활용  
Orientation

1. Kubernetes **트러블 슈팅** 방법 소개
2. [실습] **Cluster 로깅** 및 이슈 사례 조치방법
3. [실습] **Node 로깅** 및 이슈 사례 조치방법
4. [실습] **POD 로깅** 및 이슈 사례 조치방법
5. [실습] **보안관련 로깅** 및 이슈 사례 조치방법
6. [실습] **DNS 로깅** 및 이슈 사례 조치방법

<https://kubernetes.io/docs/tasks/debug-application-cluster/troubleshooting/>

# Go를 활용한 Kubernetes CLI 개발

01.

개발자를 위한  
Kubernetes 활용  
Orientation

1. Go를 활용한 Kubernetes CLI 개발 소개
2. Go언어 필수 문법 소개
3. Kubernetes Custom CLI 설계
4. [실습] Kubernetes API 활용 모듈 구현
5. [실습] Kubernetes CLI 빌드 및 실행

<https://github.com/kubernetes/client-go>

# Kubernetes 활용 미니프로젝트 수행 하기

01.

개발자를 위한  
Kubernetes 활용  
Orientation

1. Kubernetes 활용 미니프로젝트 소개
2. [실습] CI 환경 설정 및 빌드 준비 하기
3. [실습] CD 환경 설정 및 배포 준비 하기
4. [실습] CI 빌드 수행 및 컨테이너 Push, Helm Charts Push 하기
5. [실습] CD 자동 Trigger 및 배포/기동 상태 확인 하기
6. [실습] Canary 배포 적용 및 무중단 버전 업데이트 하기

## 실습 환경 구축시 과금 관련 주의사항

### 주의사항

- 해당 강의의 실습은 AWS 클라우드의 유료 서비스를 포함하고 있습니다.
- AWS 프리티어에서 제공하는 서비스 초과 사용분과 프리티어에서 제공하지 않는 서비스에서 과금이 발생할 수 있습니다.
- 정확한 요금은 다음의 사이트에서 확인하시기 바랍니다.

<https://aws.amazon.com/ko/pricing/>

### AWS 프리티어 미포함 서비스

- 대상 서비스 : AWS EKS (Elastic Kubernetes Service)
  - 요금 산정 :
    - 1 Cluster x 0.10 USD per hour x 730 hours per month = **73.00 USD**
    - 3 Worker Nodes\* x 0.0468 USD per hour x 730 hours per month = **102.492 USD**
- \* Worker Node Spec (t3a.medium Type – 2Core CPU, 4GB Mem)

**AWS EKS 총 비용 (월별) : 73.00 + 102.492 = 175.492 USD**