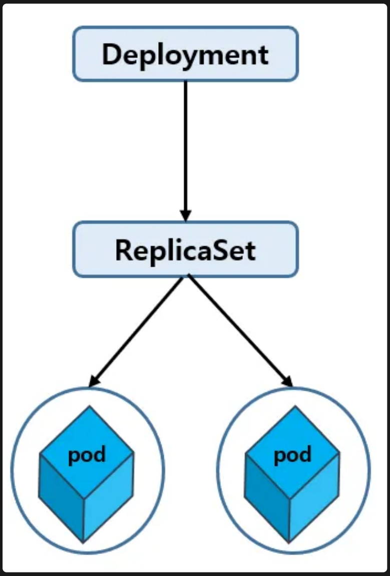
Deployment

1. Application 배포하기

- 쿠버네티스 클러스터에서 애플리케이션 배포 시 가장 많이 사용
- ReplicaSet을 컨트롤해서 Pod수를 조절



- deploy-nginx.yaml

```
apiVersion: apps/v1 kind: Deployment metadata: name: deploy-nginx spec:
replicas: 2 selector: matchLabels: app: webui template: metadata: name:
nginx-pod labels: app: webui tier: frontend spec: containers: - name:
nginx-container image: nginx:1.14
```

- 간단한 Deployment를 이용해 애플리케이션 배포하기
  - Deployment name : **weplat**
  - Container Image: **nginx:1.14**
  - Replicas : **2**

```
ubectl create deployment weplat --image=nginx:1.14 --replicas=2 --dry-
run=client -o yaml > deployment-1.yaml cat > deploy.yaml apiVersion:
apps/v1 kind: Deployment metadata: name: weplat spec: replicas: 2
selector: matchLabels: app: weplat template: metadata: labels: tier:
font-end app: weplat spec: containers: - image: nginx:1.14 name: web
resources: {} EOF kubectl apply -f deply.yaml kubectl get all
```

▶ **참고 :** Selector 와 Lables

> Note:
> A Deployment's rollout is triggered if and only if the Deployment's Pod template (that is, `.spec.template`) is changed, for example if the labels or container images of the template are updated.
> Other updates, such as scaling the Deployment, do not trigger a rollout.

▶ kubectl edit로 편집 가능한 항목

2. Pod Scale

- 배포 중인 애플리케이션 Pod를 확장하거나 축소
- 예 : weplat 에서 배포하고 있는 nginx web 컨테이너 수를 3개로 **확장**하시오.

```
kubectl scale deployment weplat --replicas=3
```

3. Rolling Update / Rollback

Rolling Update

- 동작중인 애플리케이션의 서비스 중단 없이 점진적으로 Pod내의 컨테이너 애플리케이션을 업데이트 진행
    - container version : nginx:1.14
    - new contaienr version : nginx:1.15
    - rolling update
- kubectl을 이용하여 롤링 업데이트 수행하기

  kubectl set image deployment <deploy_name> <container_name>=<new_version_image> --record

```
kubectl set image deployment.apps/weplat nginx=nginx:1.15 kubectl set
image deployment.apps/weplat nginx=nginx:1.16 kubectl rollout --help ...
Available Commands: history View rollout history pause Mark the provided
resource as paused restart Restart a resource resume Resume a paused
resource status Show the status of the rollout undo Undo a previous
rollout # update 일시중지 kubectl rollout pause deployment weplat kubectl
rollout resume deployment weplat kubectl rollout status deployment weplat
#rollback kubectl rollout undo deployment nginx-deployment kubectl rollout
undo deployment nginx-deployment --to-revision 1
```

- 예: weplat deploy의 컨테이너 이미지를 nginx:1.14에서 nginx:1.15 버전으로 rolling update 하시오.

```
kubectl set image deployment weplat app=nginx:1.15 --record kubectl
rollout history deployment bespin
```

- change-cause 를 적용해서 update

```
# --record 를 대신해서 change-cause를 미리 설정하고 rolling-update한다. kubectl
annotate deployments.apps weplat kubernetes.io/change-cause="version 1.18"
kubectl set image deployment weplat app=nginx:1.18 kubectl rollout history
deployment bespin
```

```
vi nginx-deployment.yaml apiVersion: apps/v1 kind: Deployment metadata:
name: weplat labels: app: nginx annotations: kubernetes.io/change-cause:
v.1.15 spec: replicas: 3 selector: matchLabels: app: nginx template:
metadata: labels: app: nginx spec: containers: - name: nginx image:
nginx:1.15 ports: - containerPort: 80
```

Roll Back

- 동작 중인 애플리케이션 서비스 중단 없이 이전 버전으로 되돌리기
- rolling update 진행 시 history가 기록되어 history 기반으로 rollback
- rollback

  kubectl rollout history deployment <deploy_name>
  kubectl rollout undo deployment <deploy_name>
  kubectl rollout undo deployment <deploy_name> - - to-revision=NUMBER

- 예: bespin deployment의 애플리케이션 버전을 nginx:1.14 버전으로 rollback 하시오.

```
kubectl rollout history deployment webserver kubectl rollout undo
deployment webserver --to-revision=1 ## --to-revision=1 생략시 이전버전으로
rollback ## 파드의 버전확인 kubectl describe pod webserver-XXX-YYY
```

4. 기출 문제 풀이

❓ 1. Deployment 생성하고 Scaling 하기 4%

- 작업 클러스터 : k8s

  `kubectl config use-context k8s`

- Create a deployment as follows:
- TASK:
  - name: `webserver`
  - `2` replicas
  - label: **`app_env_stage=dev`**
  - container name: `webserver`
  - container image: `nginx:1.14`
- Scale Up Deploument. : CLI
  - Scale the deployment webserver to 3 pods

▶ 답

❓ Pod Scale-out 2%

- 작업 클러스터 : k8s
- Expand the number of running Pods in " `eshop-order` " to 5.

▶ 답

❓ Rolling Update 4%

- 작업 클러스터 : k8s
- Create a deployment as follows:
- TASK:
  - name: `nginx-app`
  - Using container `nginx` with version `1.11.10-alpine`
  - The deployment should contain `3` replicas
- Next, deploy the application with new version `1.11.13-alpine`, by performing a rolling update
- Finally, `rollback` that update to the previous version 1.11.10-alpine

▶ 답