

Kubernetes 명령어 정리

<기본적인 kubectl 문법>

`kubectl [command] [type] [name] [flags]`

```
command : 리소스 수행 동작 지정
          ex ) get, create, delete, describe, apply, expose, scale, edit, exec 등

type : 리소스의 타입 지정
      ex ) pod , node , deployment , service , namespace 등

name : 해당 리소스의 이름 지정
      ex ) kubectl delete pod exam-pod1
           kubectl get pod/exam-pod1 service/example-svc1

flags : 추가적인 option 지정
       ex ) kubectl get pod -n kube-system
           kubectl get pod -o wide
```

kubectl version --short

쿠버네티스 버전 확인

kubectl get [type]

리스트 조회

- `kubectl get pods -o wide` (더 자세한 내용 출력)
- `kubectl get pods -n kube-system` (-n : 특정 네임스페이스 지정)

kubectl describe [type]

리소스 상세 조회

kubectl create -f [yaml file]

yaml 파일을 이용하여 리소스 생성

kubectl apply -f [yaml file]

yaml 파일을 이용하여 리소스 생성

- create 명령어와 차이점은 apply는 object가 이미 존재하더라도 부분적인 spec 추가 구성이 가능하고 create는 object가 이미 존재하면 error가 발생한다.

kubectl run pod --image=[image name]

클러스터에 특정 이미지 리소스 생성

- `kubectl run pod --image=nginx : nginx 이미지로 리소스 생성`

kubectl expose [type]/[name] --type=[service type] --port [port]

서비스를 통한 외부로 애플리케이션 노출

- ex) `kubectl expose deployment/nginx-deployment --type=NodePort --port 8080`

kubectl exec -it [pod name] /bin/bash

파드 접속

- bash 셸을 유지하여 접속
- Pod 안에 컨테이너가 여러개일때 접속 방법

ex) `kubectl exec -it [pod name] -c [container name] /bin/bash`

kubectl logs [pod name]

파드 로그 확인

kubectl delete [type] [name]

리소스 삭제

- `kubectl delete -f [yaml file]` : 파일로 만든 리소스 삭제
- `kubectl delete pod,deployment --all` : 파드와 디플로이먼트 모두 삭제

kubectl edit [type] [name]

리소스 설정 변경

kubectl scale --replicas=[count] deploy/[deployment name]

디플로이먼트의 파드 개수 변경

kubectl get pod [pod name] -o yaml > [yaml file]

해당 파드의 설정을 yaml 파일로 저장

kubectl set image deployment [deployment name] [prev_image]=[next_image] --record

디플로이먼트의 모든 파드 이미지를 일괄 변경

- ex) `kubectl set image deployment nginx-deployment nginx=nginx:1.20.2 --record`
- 반대로 edit 명령어를 사용했을때는 파드 단일 변경 가능

kubectl rollout history deployment [deployment name]

디플로이먼트의 파드 업데이트 정보 확인

- ex) `kubectl rollout history deployment nginx-deployment`

kubectl rollout undo deployment [deployment name] --to-revision=[revision version]

디플로이먼트를 이전 버전의 세팅으로 롤백

- ex) `kubectrl rollout undo deployment nginx-deployment --to-revision=1`
- revision version의 경우 rollout history 정보 확인하여 선택