

쿠버네티스 port-forward 공부



좋습니다 👍 지금부터 차근차근 해볼게요.

`kubect1 port-forward`는 쿠버네티스 클러스터 내부에서만 접근 가능한 리소스(Pod, Service 등)를 로컬 PC 포트에 연결해서 접근할 수 있게 해주는 명령어예요.

예를 들어, 클러스터 안에서만 동작하는 MySQL, 웹 서비스 같은 걸 내 노트북 브라우저나 DBeaver 같은 툴로 직접 테스트할 수 있죠.

기본 구조

```
kubect1 port-forward <리소스 종류>/<이름> <로컬포트>:<타겟포트>
```

예시:

```
kubect1 port-forward pod/myapp-pod 8080:80
```

- **pod/myapp-pod** → 연결할 Pod
- **8080** → 내 로컬에서 접속할 포트
- **80** → Pod 안 컨테이너가 열고 있는 포트

→ 이제 브라우저에서 `http://localhost:8080`으로 접속하면, 실제로는 Pod의 `80` 포트에 연결돼요.

특징

1. 임시 연결

- `kubect1 port-forward`를 실행하는 동안만 동작해요. (터미널 끄면 연결 종료)

2. 네트워크 경로 단축

- 클러스터 외부 노출(Ingress, NodePort 등)을 안 해도, 개발자 로컬에서 바로 접근 가능.

3. 리소스 종류

- Pod, Service 둘 다 가능.
- Service를 걸면, 해당 Service 뒤에 연결된 Pod들 중 하나로 트래픽이 라우팅됨.

제가 먼저 하나 여쭙보고 싶습니다:

👉 지금은 `port-forward`를 이론 위주로 배우고 싶으신 건가요, 아니면 **실습 예제 (예: nginx, mysql 같은 pod)**를 직접 따라하면서 이해하고 싶으신 건가요?



좋습니다 😊

그럼 작은 **nginx** 웹서버를 하나 띄워서 `kubectl port-forward` 실습을 해보는 흐름으로 가보죠.

1단계: nginx Pod 만들기

```
kubectl run nginx-pf --image=nginx --port=80
```

- `nginx-pf`라는 이름의 Pod을 생성
- nginx 컨테이너가 80번 포트를 사용

👉 여기까지 실행하면 `kubectl get pods`로 `nginx-pf`가 **Running** 상태인지 확인하세요.

2단계: port-forward 실행

```
kubectl port-forward pod/nginx-pf 8080:80
```

- 로컬 포트 `8080` → Pod 안 컨테이너 포트 `80`
- 이 명령이 실행된 터미널은 계속 잡혀 있어야 연결이 유지돼요.

3단계: 로컬에서 접속 확인

브라우저에서:

```
http://localhost:8080
```

→ nginx 기본 환영 페이지가 보이면 성공 🎉

4단계: Service를 통해 연결하기 (선택)



Pod 말고 Service를 만들어도 돼요.

```
kubectl expose pod nginx-pf --port=80 --target-port=80 --name=nginx-svc
```

그리고:

```
kubectl port-forward svc/nginx-svc 9090:80
```

→ 이번엔 `http://localhost:9090` 접속

제가 궁금한 건 🙋 지금 학습 환경이 **minikube / kind / 회사 온프레미 클러스터** 중 어디인지예요?
(환경에 따라 접속 테스트 방식이 조금 달라질 수 있거든요.)