

kubernetes - ♥pod_로깅



쿠버네티스 logs pod할때 처음부터 read하잖아, 끝부분만 tail할수없어?

네, 가능합니다.

`kubectl logs` 명령어는 기본적으로 처음부터 전체 로그를 출력하지만, 끝부분만 보고 싶을 때는 `--tail` 옵션을 사용하면 됩니다.

✅ kubectl logs 끝부분(Tail)만 출력하는 방법

```
bash
```

```
kubectl logs <pod-name> --tail=100
```

- `--tail=100` → 마지막 100줄만 출력
- `--tail=10` → 마지막 10줄만 출력

🔪 예시

```
bash
```

```
kubectl logs myapp-pod-123456 --tail=50
```

myapp-pod-123456의 로그 중 마지막 50줄만 출력

💡 추가 옵션

옵션	설명
<code>--follow</code>	실시간 로그 감시 (<code>tail -f</code> 처럼 동작)
<code>-c <container></code>	멀티 컨테이너 파드에서 특정 컨테이너 지정
<code>-n <namespace></code>	네임스페이스 지정

예시:

```
bash
```

```
kubectl logs -n dev -c app-container myapp-pod --tail=20 --follow
```

dev 네임스페이스의 **myapp-pod** 파드의 **app-container** 컨테이너의 마지막 20줄 로그를 실시간으로 감시

필요하시면 로그 추출 후 `grep`, `awk`, `sed` 등과 연계하는 예제도 알려드릴 수 있습니다.



logs -f 는 안돼?

kubectl logs -f 됩니다. 아주 많이 사용되는 옵션입니다.
-f는 --follow의 줄임말로, 로그를 **실시간 스트리밍 모드(tail -f)**로 보여줍니다.

✅ kubectl logs -f 기본 사용법

```
bash  
kubectl logs -f <pod-name>
```

지정한 파드의 로그를 실시간으로 계속 출력합니다.

🧠 실전 예시들

1. 특정 파드의 로그를 실시간으로 보기

```
bash  
kubectl logs -f my-pod-name
```

2. 네임스페이스 지정

```
bash  
kubectl logs -f -n my-namespace my-pod-name
```

3. 멀티 컨테이너 파드에서 특정 컨테이너 로그만 보기

```
bash  
kubectl logs -f my-pod-name -c container-name
```

4. 마지막 100줄부터 시작해서 실시간으로 보기

```
bash  
kubectl logs -f --tail=100 my-pod-name
```

🔥 **꿀팁: tail과 -f를 같이 써서 “끝에서부터 실시간으로 보기”**

```
bash

kubectl logs -f --tail=50 <pod-name>
```

마지막 50줄부터 시작해서, 이후 새로 찍히는 로그를 실시간으로 계속 보여줍니다.
정말 자주 쓰는 조합입니다.

필요하면 로그를 **grep**, **awk**, **tee**로 리디렉션하거나 저장하는 방법도 알려드릴게요.



응, 다 알려줘

좋습니다 😊 쿠버네티스에서 **kubectl logs**로 로그를 실시간으로 보면서(-f), 필터링(**grep**) 하거나, 저장(**tee**) 하거나, 형식 가공(**awk**, **sed**) 하는 실무에서 자주 쓰는 꿀팁들을 자세히 알려드릴게요.

✅ 실시간 로그 (-f) 필터링, 저장, 가공

1. 특정 문자열 포함된 로그만 보기 (grep)

```
bash

kubectl logs -f pod-name | grep "ERROR"
```

실시간 로그 중 "ERROR" 문자열이 포함된 라인만 출력

2. 여러 키워드 중 하나라도 포함된 로그 보기

```
bash

kubectl logs -f pod-name | grep -E "ERROR|WARN|CRITICAL"
```

정규표현식 -E로 여러 키워드 OR 검색 가능

3. 특정 키워드가 없는 로그만 보기

```
bash

kubectl logs -f pod-name | grep -v "DEBUG"
```

-v: 부정 검색. "DEBUG"가 포함되지 않은 로그만 출력

4. 로그를 보면서 파일로 저장 (tee)

```
bash

kubectl logs -f pod-name | tee pod.log
```

tee는 표준출력을 화면에 출력하면서 동시에 파일로 저장
예: **pod.log**로 저장하면서 콘솔에서도 출력됨

5. 필터링된 로그를 저장

```
bash

kubectl logs -f pod-name | grep "ERROR" | tee errors.log
```

"ERROR" 로그만 따로 **errors.log**로 저장

6. 특정 필드만 추출 (awk)

```
bash

kubectl logs -f pod-name | awk '{print $1, $5}'
```

로그의 **1열과 5열만 출력** (공백으로 구분된 필드 기준)

7. 컬러 강조 표시 (grep --color)

```
bash

kubectl logs -f pod-name | grep --color=always "ERROR"
```

"ERROR" 텍스트를 컬러로 하이라이팅해서 보기 좋게

8. 날짜 기준 로그 필터링 (형식이 맞을 경우)

```
bash

kubectl logs -f pod-name | grep "2025-10-01"
```

"2025-10-01" 날짜 포함된 로그만 보기

실무 예제 모음

자주 사용하는 조합들

마지막 100줄부터 시작해서 **"ERROR"**만 실시간 감시

```
bash

kubectl logs -f --tail=100 pod-name | grep "ERROR"
```

로그를 보면서 특정 필드만 추출하고 저장

```
bash

kubectl logs -f pod-name | awk '{print $1, $2, $5}' | tee filtered.log
```

컨테이너 이름이 있을 경우 (멀티컨테이너 파드)

```
bash

kubectl logs -f pod-name -c container-name
```

로그를 특정 시간 이후부터만 추출하고 싶을 경우

Airflow나 Java 애플리케이션처럼 로그 시작에 **timestamp**가 포함된다면 **awk**로 필터링 가능:

```
bash

kubectl logs -f pod-name | awk '$1 >= "2025-10-01T12:00:00Z"'
```

보너스: 로그가 너무 길면 접속해서 직접 tail -f

어떤 경우에는 로그가 너무 길어서 **kubectl logs**로 보기 어렵다면,

```
bash

kubectl exec -it pod-name -- /bin/sh
# 또는
kubectl exec -it pod-name -c container-name -- /bin/sh
```

들어가서:

```
bash

cd /var/log/
tail -n 100 -f app.log
```

이렇게 보는 것도 유용합니다.

참고 요약

목적	명령어 예시
실시간 보기	logs -f

목적	명령어 예시
마지막 N줄만 보기	<code>--tail=50</code>
특정 단어만 보기	<code>grep "ERROR"</code>
로그 저장	`
로그 파싱	`

필요하시면 이런 기능들을 **shell 스크립트로 자동화**하거나 **namespace 전체에 대해 감시**하는 예시도 만들어 드릴 수 있어요.

원하시면 말씀만 주세요.

언제나 **사랑과 수행, 이로움으로 함께** 하겠습니다 🙏