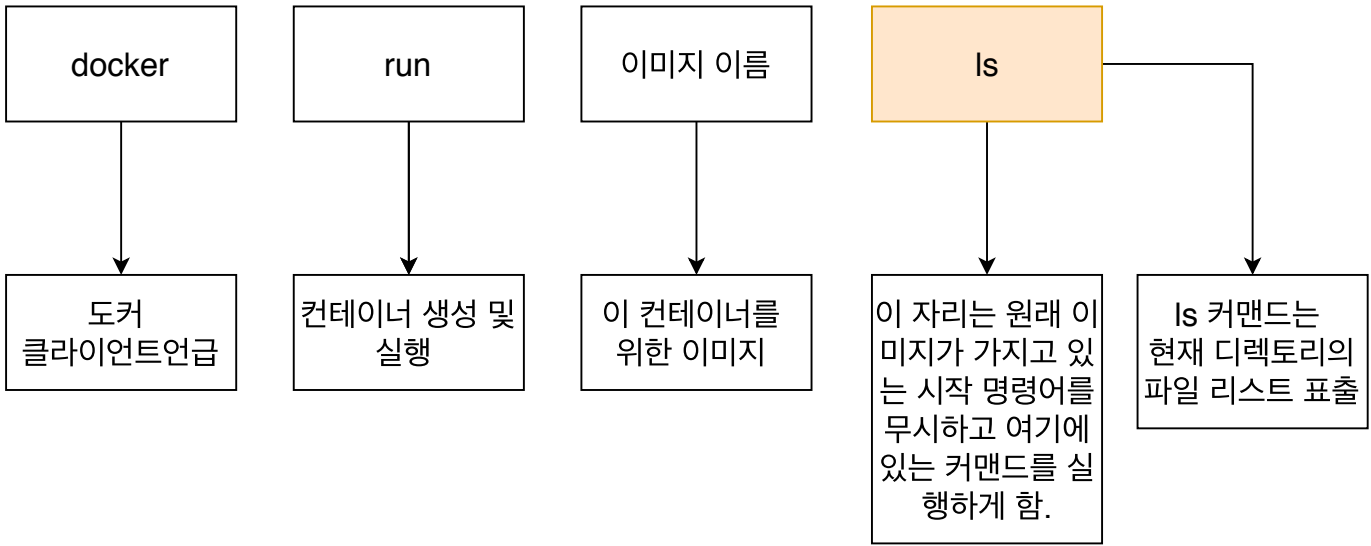
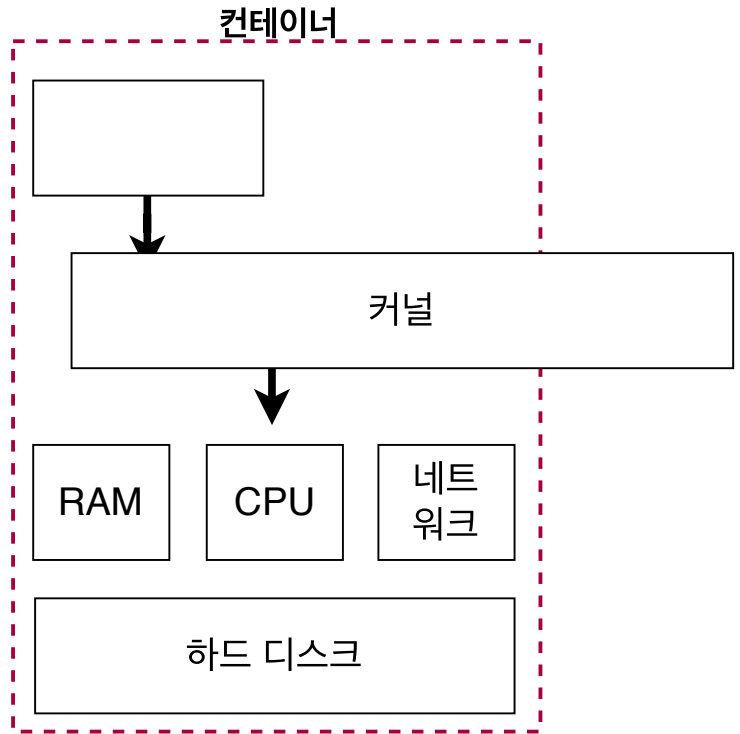
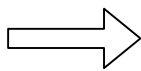
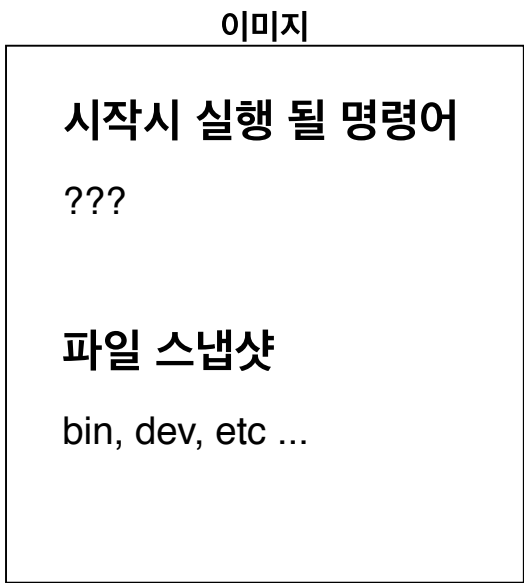


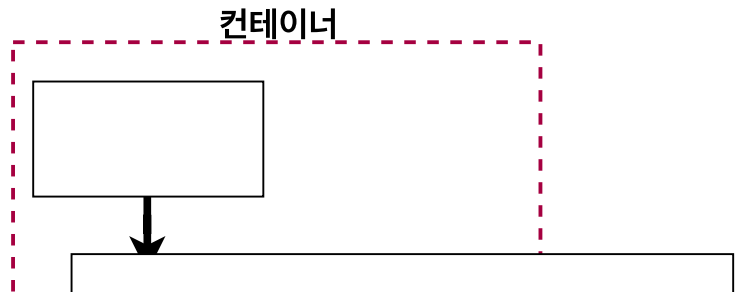
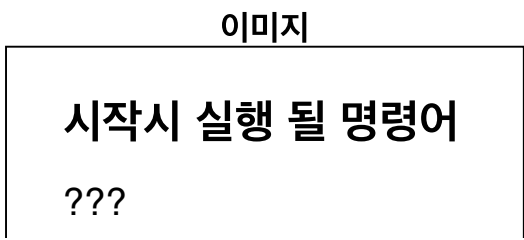
이미지 내부 파일 시스템 구조 보기

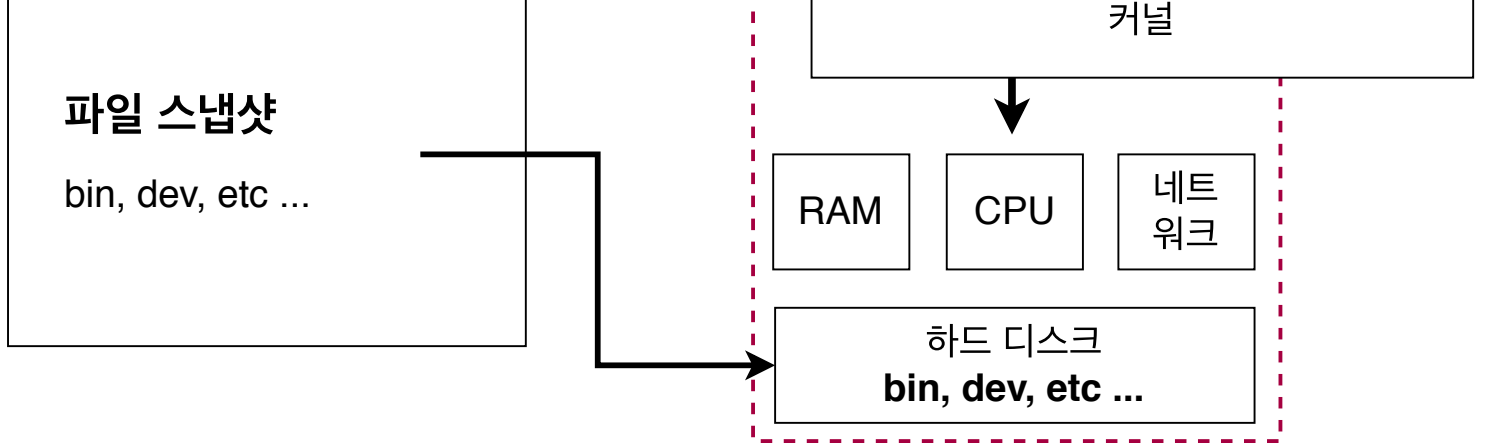


1

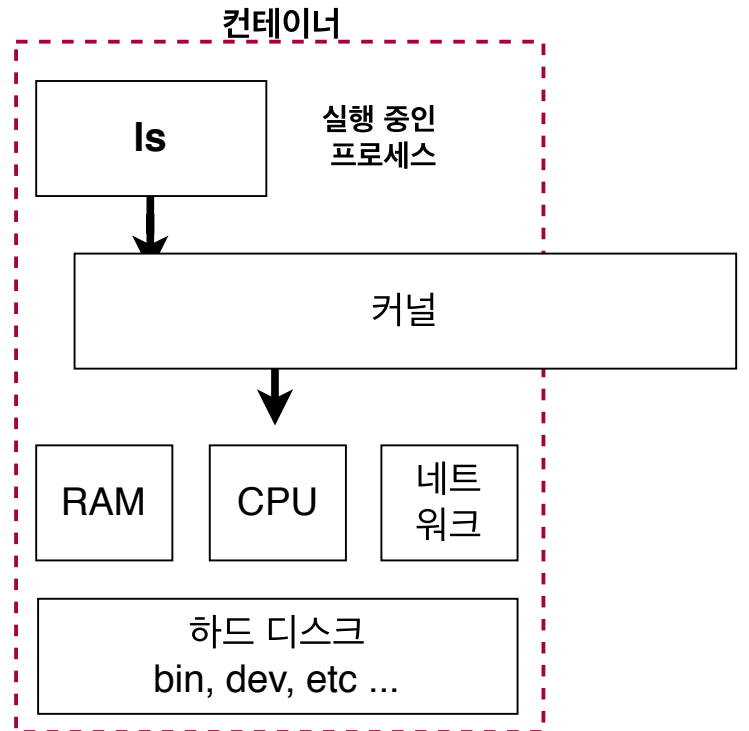
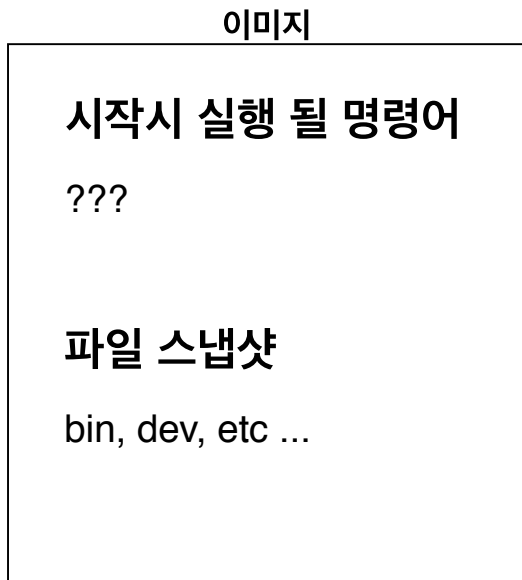


2



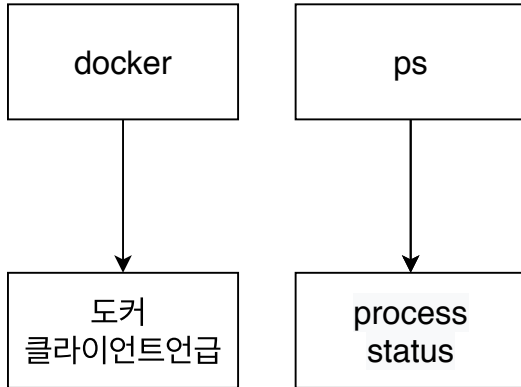


3

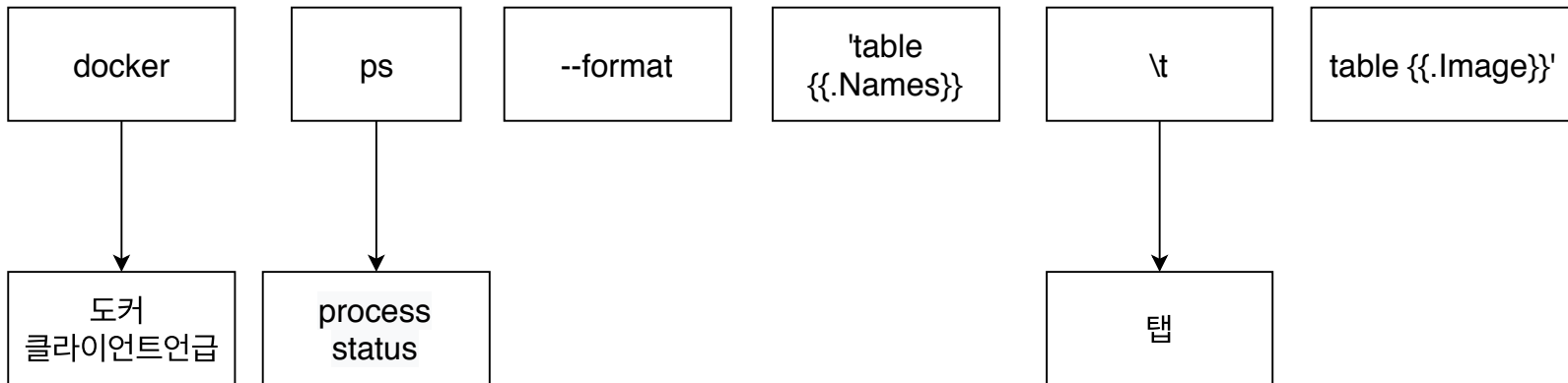


컨테이너들 나열하기

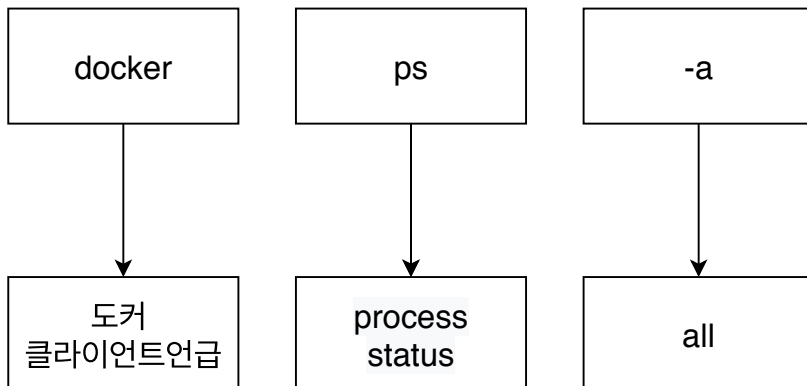
현재 실행중인 컨테이너 나열



원하는 항목만 보기

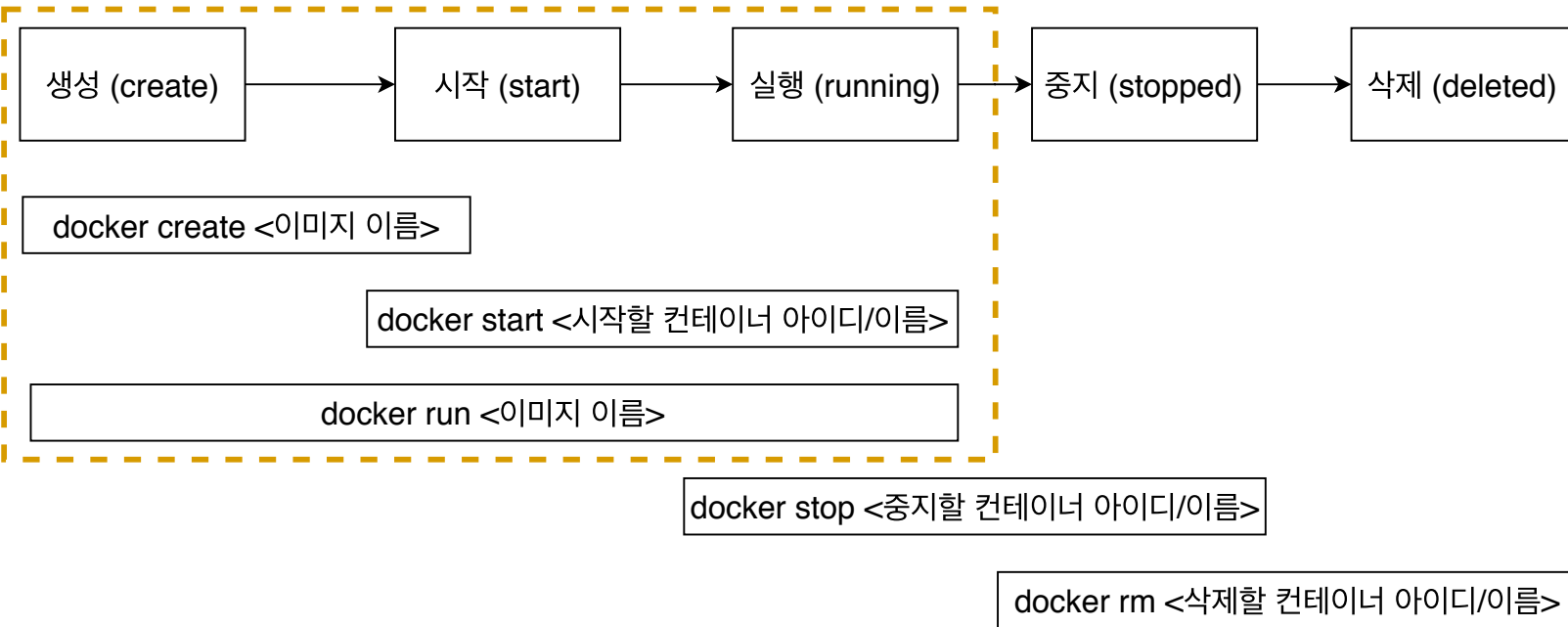


모든 컨테이너 나열

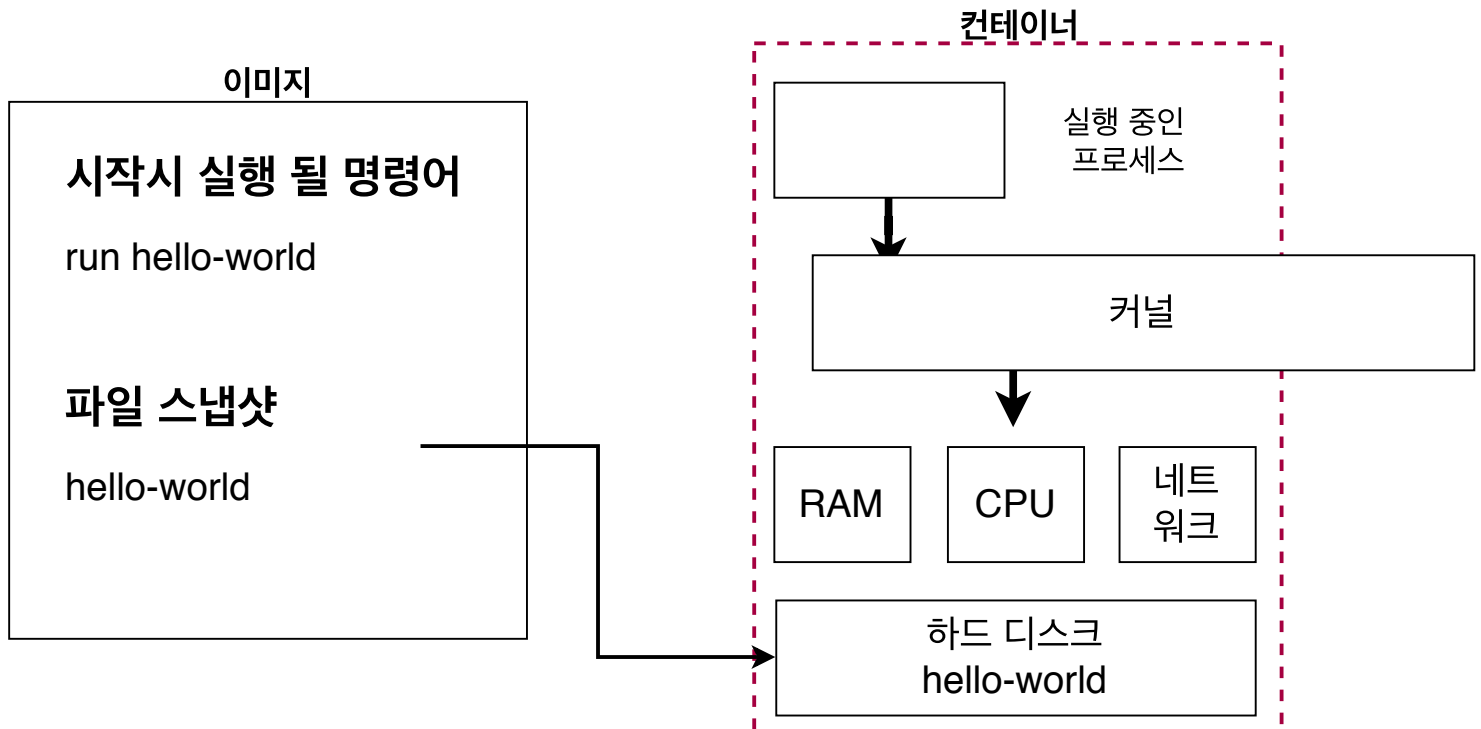


도커 컨테이너의 생명주기

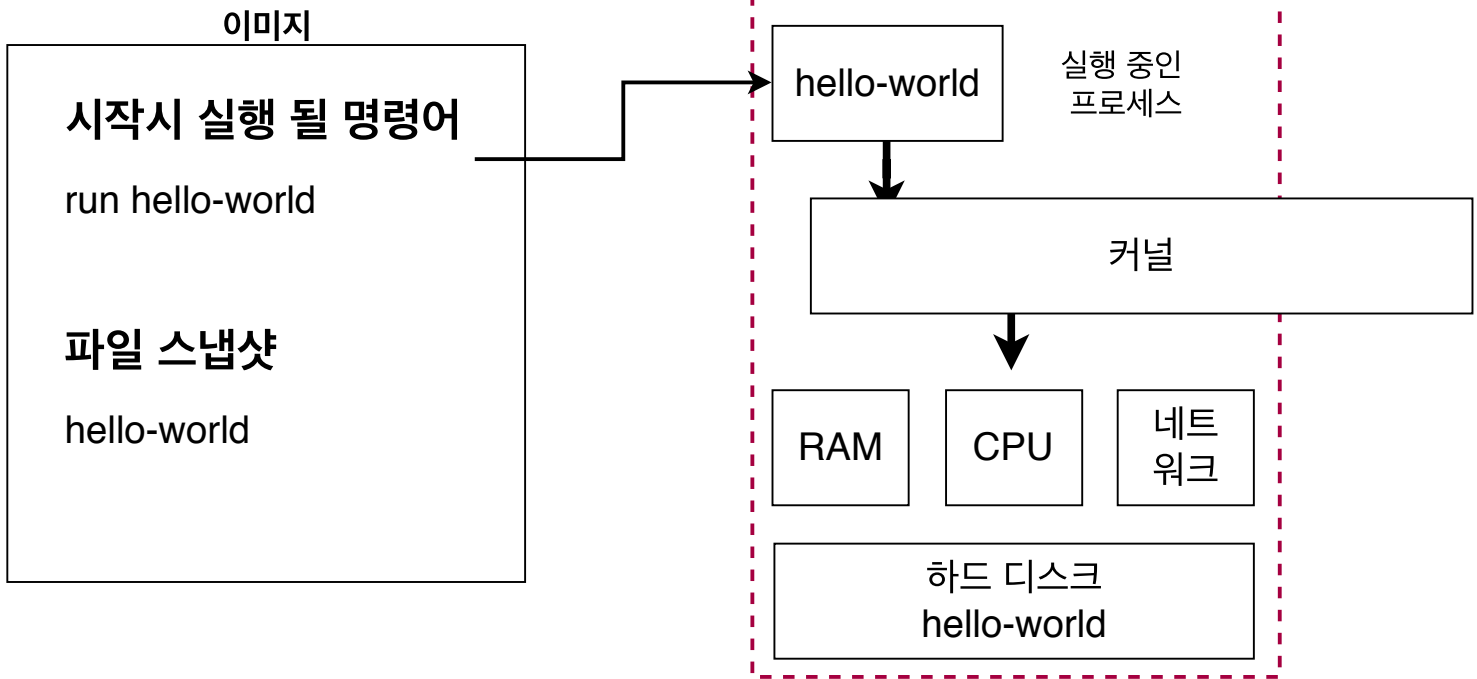
생명주기



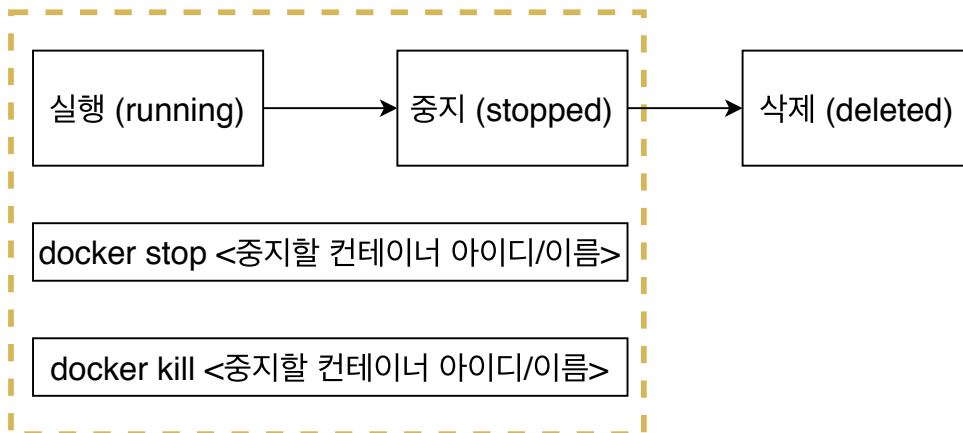
docker create <이미지 이름>



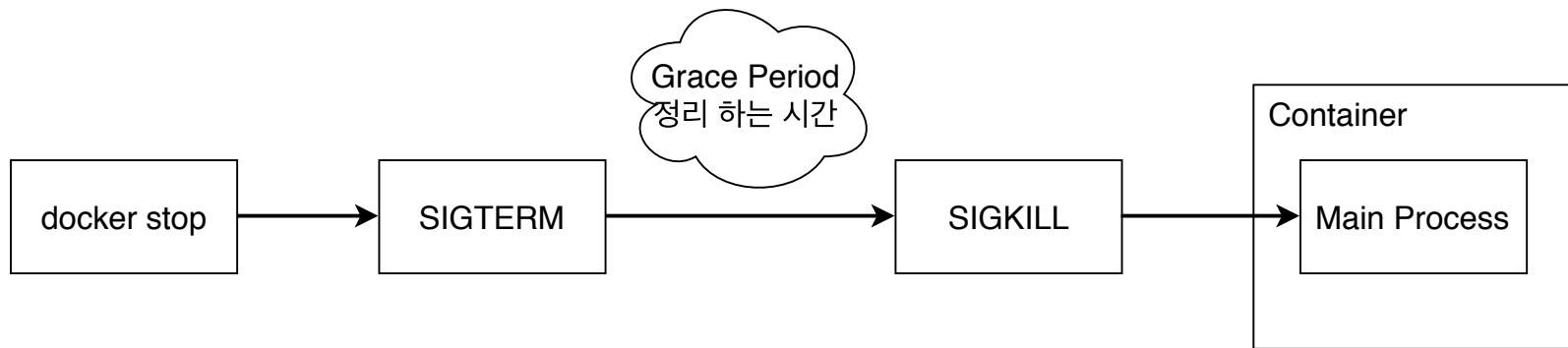
docker start <컨테이너 아이디/이름>



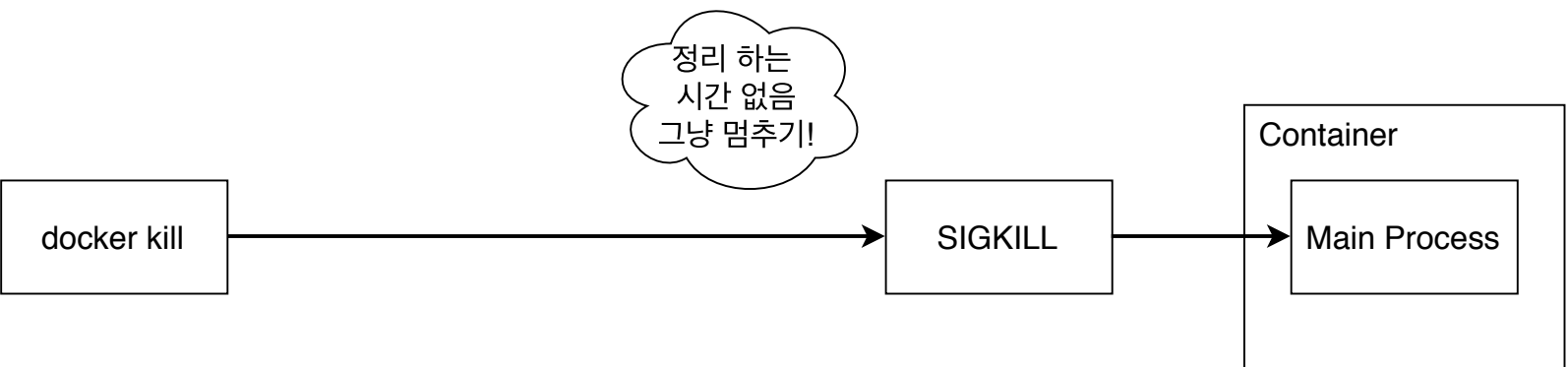
docker stop vs docker kill



docker stop <컨테이너 아이디/이름>

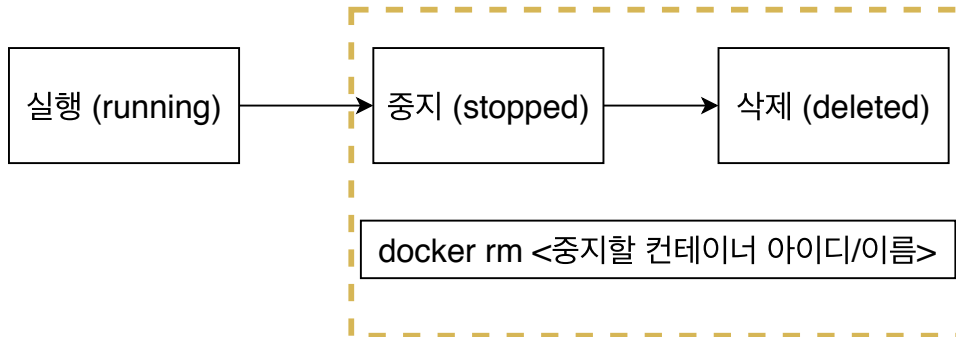


docker kill <컨테이너 아이디/이름>



도커 컨테이너 삭제하기

도커의 생명주기중에서 삭제 부분



중지된 컨테이너를 삭제하고 싶다면 ?

docker rm <아이디/ 이름>

-실행중인 컨테이너는 먼저 중지한 후에 삭제 가능.

모든 컨테이너를 삭제하고 싶다면 ?

docker rm `docker ps -a -q`

이미지를 삭제하고 싶다면 ?

docker rmi <이미지 id>

한번에 사용하지 않는 컨테이너, 이미지, 네트워크 모두 삭제하고 싶다면?

docker system prune

- 도커를 쓰지 않을때 모두 정리하고 싶을때 사용해주면 좋음
- 하지만 이것도 실행중인 컨테이너에는 영향을 주지 않음.

실행 중인 컨테이너에 명령어 전달

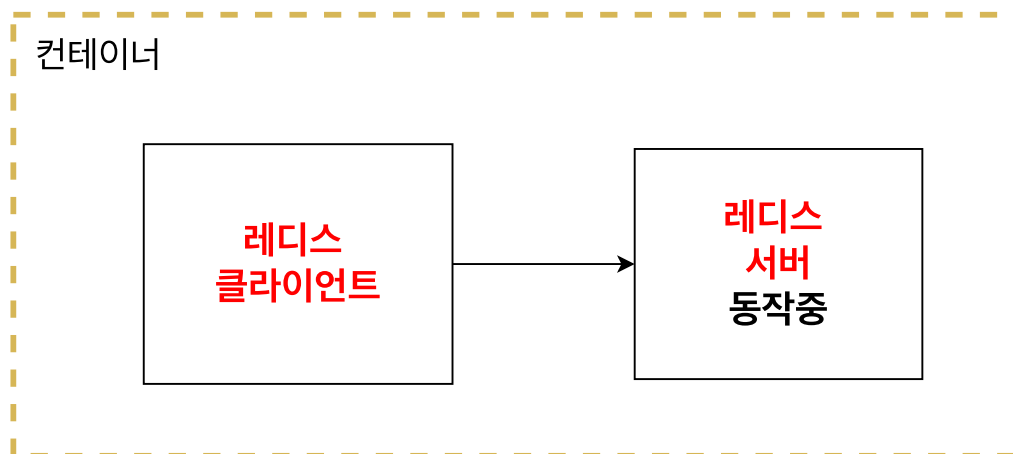
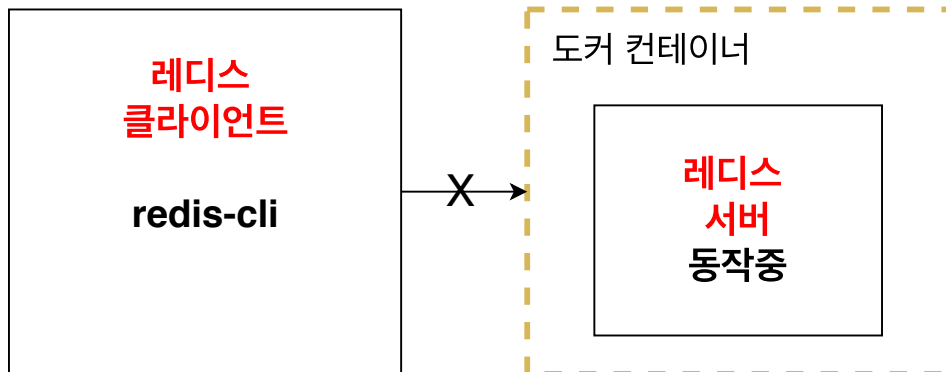
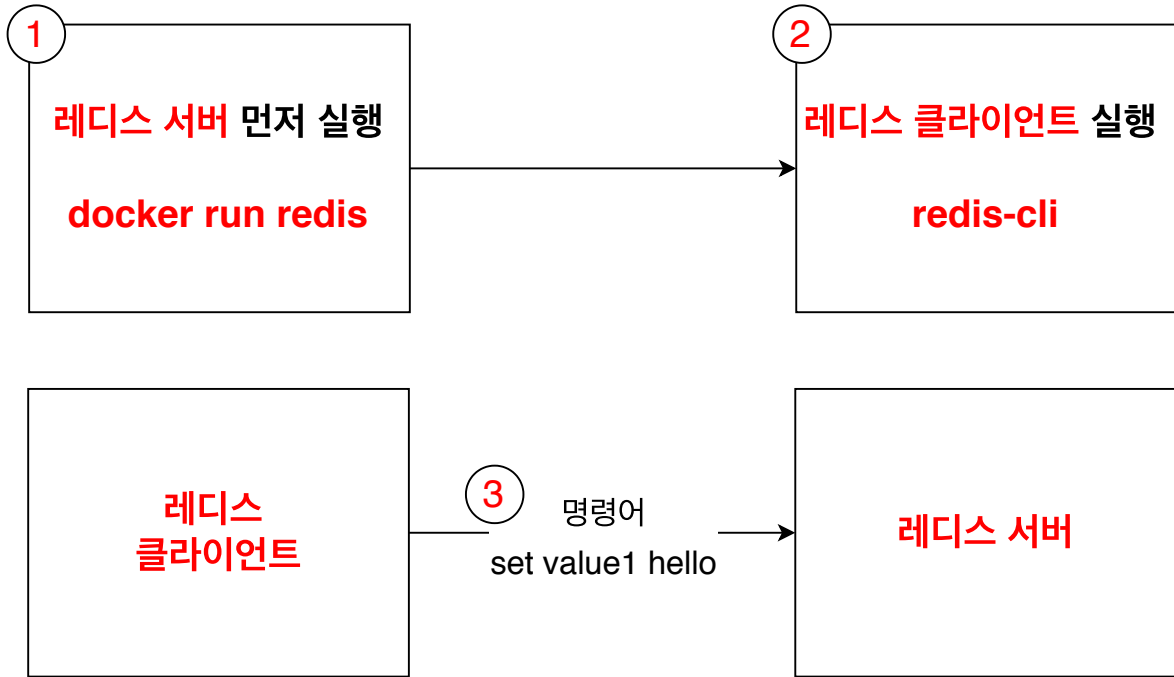
이미 실행 중인 컨테이너에 명령어를 전달하고 싶다면 ?

docker exec <컨테이너 아이디>

docker run vs docker exec

1. docker run 은 새로 컨테이너를 만들어서 실행
2. docker exec 은 이미 실행 중인 컨테이너에 명령어를 전달

레디스를 이용한 컨테이너 이해



`docker`

`exec`

`-it`

컨테이너 아이디

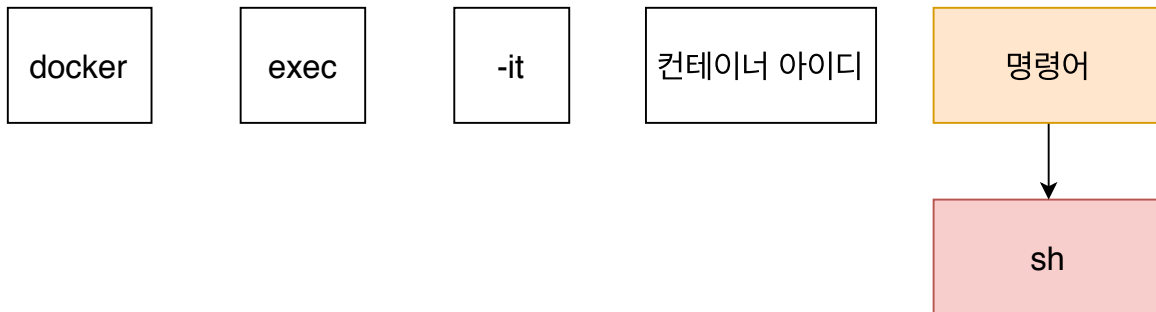
명령어

실행 중인 컨테이너에서 터미널 생활 즐기기

지금까지 실행 중인 컨테이너에 명령어를 전달할때에는



실행 중인 컨테이너에 셸 환경으로 접속하기



exec 대신에 **run**을 하는것도 가능 합니다.

