

docker run

1. (In CMD)

docker pull [IMG]

2. docker run -d [IMG]

3. VScode 연동

- vs code 확장자 설치에서 remote development 설치
- ctrl + shift + P (모든 명령 표시) 후 remote-containers.Attatch to Running container
- IMG 선택

- docker pull : 도커 이미지 가져오기
- docker run : Image로 container를 생성하는 명령

docker run 옵션

- d : detach → 프로세스를 실행시킴과 동시에 다른 작업을 병행할 수 있는 방식
- i : interactive → 표준 입력을 받음
- t : tty → 터미널로 연결가능한 컨테이너를 만듦
- it → 컨테이너를 종료하지 않은채로, 터미널의 입력을 계속해서 컨테이너로 전달하기 위해서 사용
- name → ID를 사용하면 불편하니, 이름을 부여해 컨테이너를 식별
- e → 컨테이너 내에서 사용할 환경 변수를 설정
- rm → 프로세스 종료시 컨테이너 자동 제거
- link → 컨테이너 연결
- p → 호스트와 컨테이너의 포트를 연결
- docker ps : 현재 가동중인 컨테이너 출력 (-a : 가동, 멈춘 컨테이너 다 출력)
- docker logs : docker container 로그 확인

컨테이너에 자바 다운로드 방법

1. jdk(개발 자바) 다운로드

2. `docker cp jdk-8u271-linux-x64.tar.gz dcd5ea03d3b:/`

--> 컨테이너에 jdk 파일 복사하기 → 안됨

3. `docker exec -it dcd5ea03d3b /bin/bash -l`

컨테이너로 들어가기

4. 압축 풀기

tar.gz 압축풀기 : `tar -xvzf (압축파일명).tar.gz`

5. 압축 해제된 폴더를 /usr/local/java에 mv로 옮긴다

/usr/local/java는 형식적으로 프로그램을 모아두는 곳

`mv (이동할 파일명) (이동할 위치)`

6. 환경변수세팅

`vi $HOME/.profile`

(i를 눌러서)

`export JAVA_HOME=/usr/local/java`

`export PATH=$JAVA_HOME/bin:$PATH`

(입력하고 esc누르고)

`:wq`

* java 설치 후 db로 접속하는 방법

`sqlplus sys/oracle as sysdba -> db로 접속 (cmd에서)`

vs code에 도커 연동

<https://opencvlib.weebly.com/blog/docker-vscode>

```
docker pull truevoly/oracle-12c
```

```
docker run -d truevoly/oracle-12c
```