Software Engineering

- docker container commands-

Professor Han-gyoo Kim

2022

Container의 일생

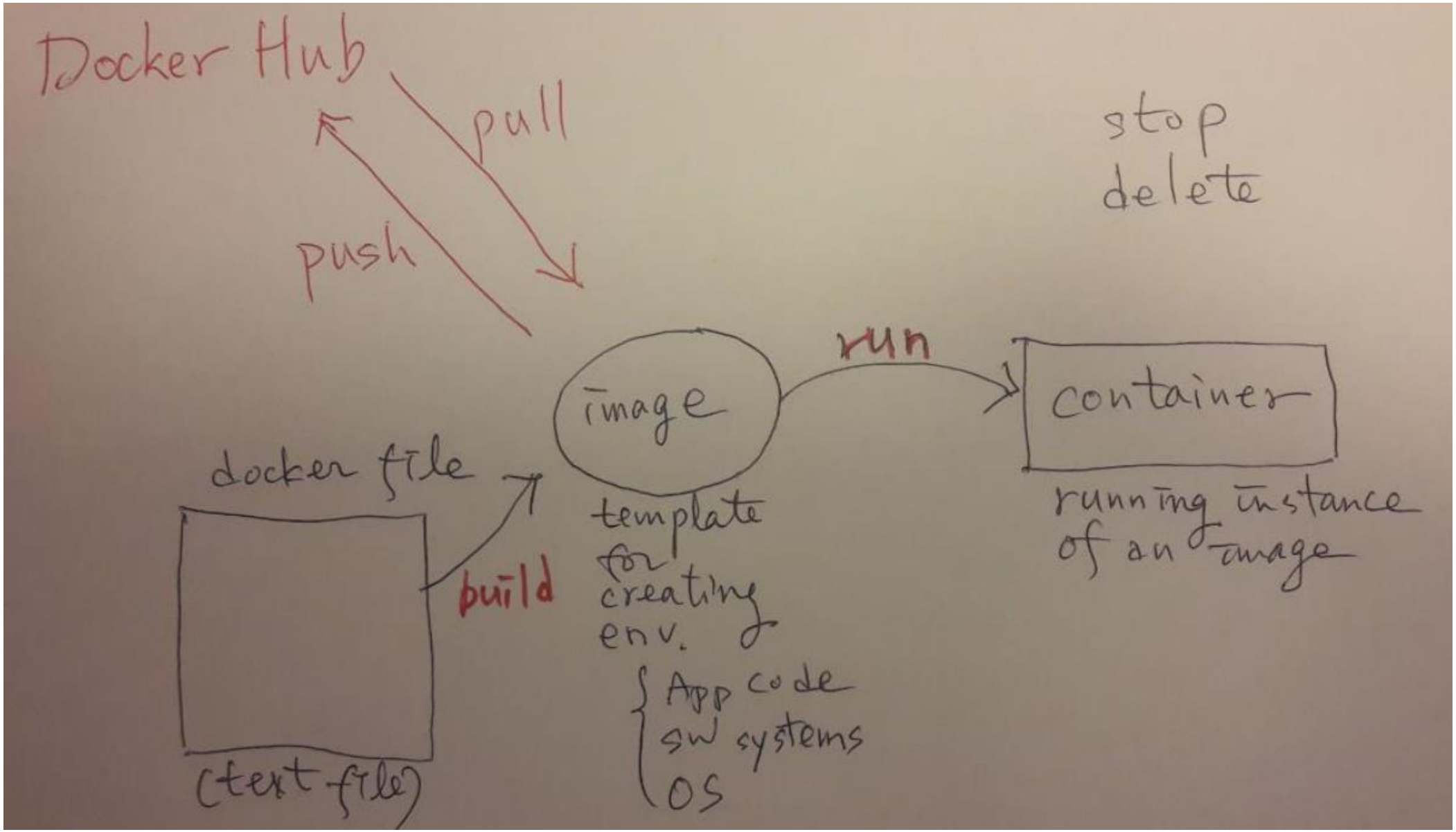


Image vs Container

- Container image = unchangeable, static file that includes executable code so it can run an isolated process on a containerization platform, i.e., a container engine
- Container image is comprised of system libraries, system tools and other platforms settings a software program needs to run on a container engine
- The image shares the OS kernel of host machine
- A container image is compiled from file system layers built onto a parent or base image reusing various SW components, so the user does not create everything from scratch for every project
- Container = running instance of an image => 기본적으로 하나의 컨테이너에는 하나의 프로세스만 실행되도록 image를 만들 것이 권장되며, 두 개 이상의 프로세스가 필요한 app의 경우 다수의 container를 만들어 사용하는 것이 필요(웹 서버 프로세스와 DBMS 프로세스) 그러나 꼭 필요하다면 하나의 컨테이너에 다수의 프로세스가 실행되도록 만들 수 있음

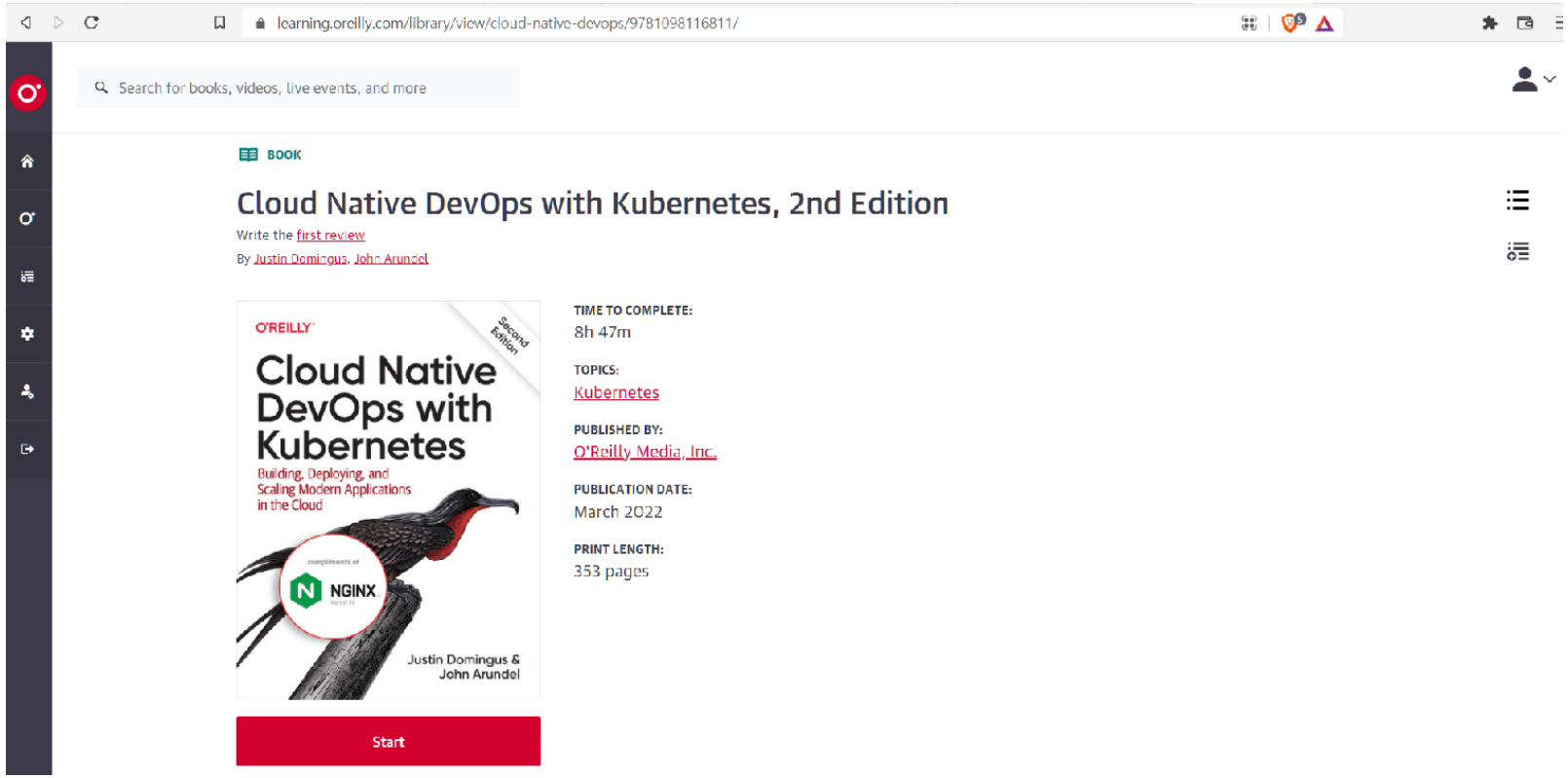
- Image는 조리법(recipe), container는 만들어진 요리
- Stopped container는 만들어서 냉장고에 보관한 요리

A book on docker container & kubernetes

- Cloud Native DevOps with Kubernetes, 2nd Edition, 2022

  By Justin Domingus, John Arundel

  학교 중앙도서관 https://learning.oreilly.com/home/ 로그인 후 책 제목으로 찾기 하면



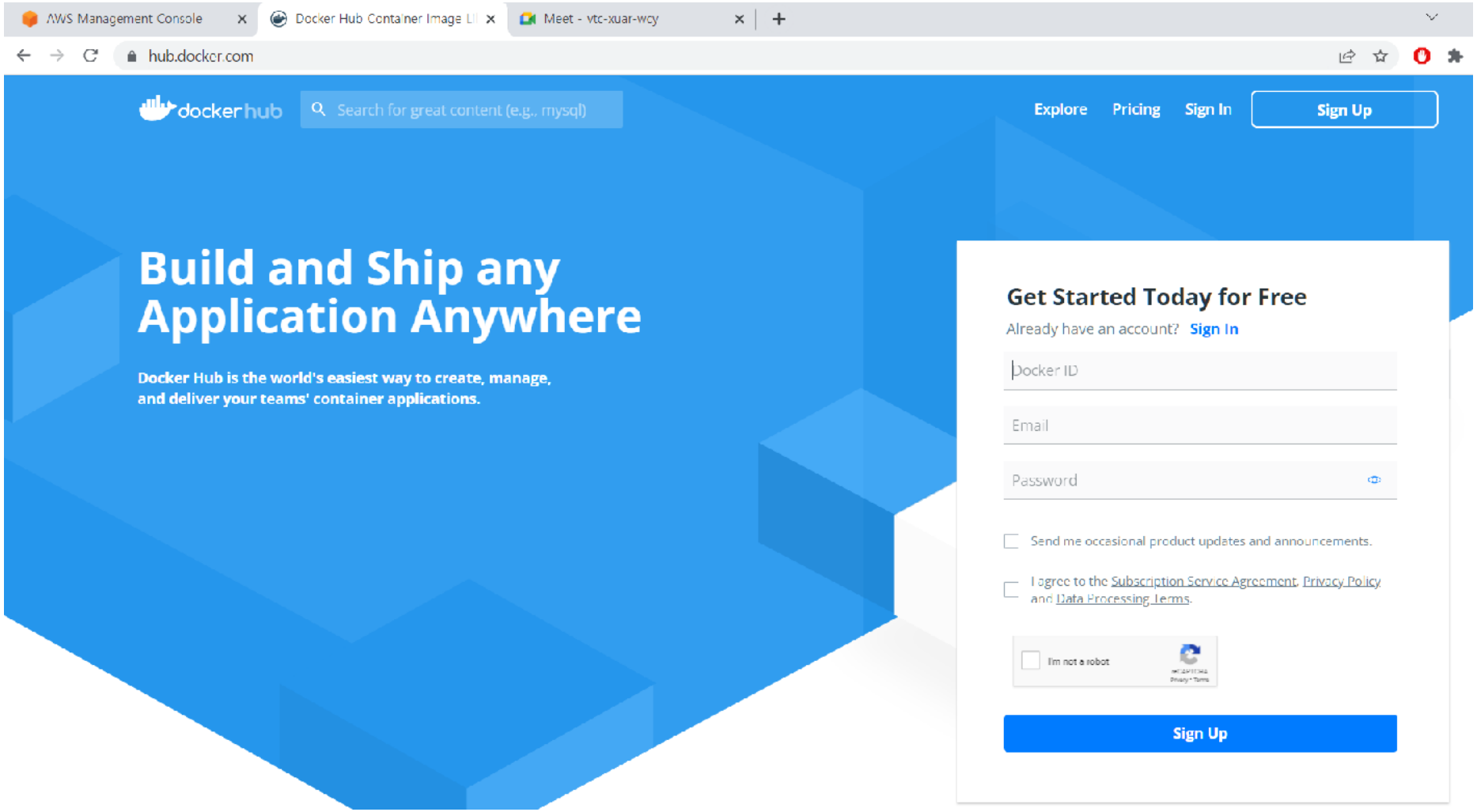
Install docker container

- Docker installation on Ubuntu 20.04
  1) Using source from docker.com
     - sudo apt-get update
     - sudo apt-get install docker.io
  2) Using ubuntu repository
     - https://docs.docker.com/engine/install/ubuntu/ 참조 (가장 최신 버전)
- docker --version // docker -v
- docker --help // docker -h
- (sudo) docker run hello-world
- cat /etc/group
- sudo usermod -aG docker $USER

Docker container basic commands

- docker pull ubuntu
- docker images
- //run
- docker run -it -d ubuntu
- docker ps
- //exec
- docker exec -it <container id> <prorm in the container>
- exit
- //stop
- docker stop <container id>
- docker kill <container id>
- docker ps -a
- //remove
- docker rm –f <container id> // remove continer process
- docker rmi –f <image id> // remove image

Docker hub


AWS Management Console
Docker Hub Container Image Li
Meet - vtc-xuar-wcy
hub.docker.com
dockerhub
Search for great content (e.g., mysql)
Explore
Pricing
Sign In
Sign Up
Build and Ship any Application Anywhere
Docker Hub is the world's easiest way to create, manage, and deliver your teams' container applications.
Get Started Today for Free
Already have an account? Sign In
Docker ID
Email
Password
Send me occasional product updates and announcements.
I agree to the Subscription Service Agreement, Privacy Policy and Data Processing Terms.
I'm not a robot
Sign Up

Docker commands 예제 (containerized web service)

- docker commit <container id> <image name>
- Apache web server running as a container
  - docker run -it -d ubuntu
  - docker ps
  - docker exec -it <container id> bash
  - apt-get update
  - apt-get install -y apache2
  - cd /var/www
  - vim 1.html // vim is found not installed
  - apt-get install -y vim
  - cd /var/www/html
  - vim 1.html
  - // html 문서 속
  - <html> <head> <title>First Example</title> </head>
  - <body>
  - <h2 style="color: dodgerblue”>Hello, Hongik!</h2> </body> </html>
  - // vim 에서 나온 다음에
  - exit
  - docker commit <container id> <new image name>
  - // new image name convention : swedemo/webservice
  - docker images // image size 확인
  - docker ps
  - // webservice container 실행
  - docker run -it -p 81:80 -d web1 // port forwarding
  - docker ps
  - docker exec -it <container id> bash
  - //container 속에서
  - Service apache2 status
  - Service apahe2 start
  - Exit
  - // 현재 EC2의 public IP 주소 복사 후
  - // 웹 브라우저에 EC2 기계의 IP 주소:81 입력, 연결되지 않는 이유 브라우저에서 확인
  - // EC2 기계의 security inbound rule에서 TCP 81 포트 열고 나서 웹 브라우저에서 다시 EC2 IP:81 입력하여 ubuntu apache2 default page 확인
  - //웹 브라우저에서 EC2 IP:81/1.html 입력하여 확인
  - docker push 계정/image name:tag