# SWPP Practice Session #0
# Docker

Sep. 7, 2022

# Background Survey

https://forms.gle/983bkK2dLNLkkXMC8

# SWPP Practice Session Overview

⭐ In September, we will have an **offline** practice session every Wednesday (19:00 ~ 20:50). ⭐

- If you can’t attend the practice session due to COVID-19 issues, please let us know in advance.

# SWPP Practice Session Overview

As notified in today's lecture, during the semester, the students are expected to apply the principles to systems by working on team projects on web services.

In the practice sessions, we will teach stuff to build a web service starting from the basics (including languages, web frameworks, deployment, etc.).
Check our GitHub page for the schedule.

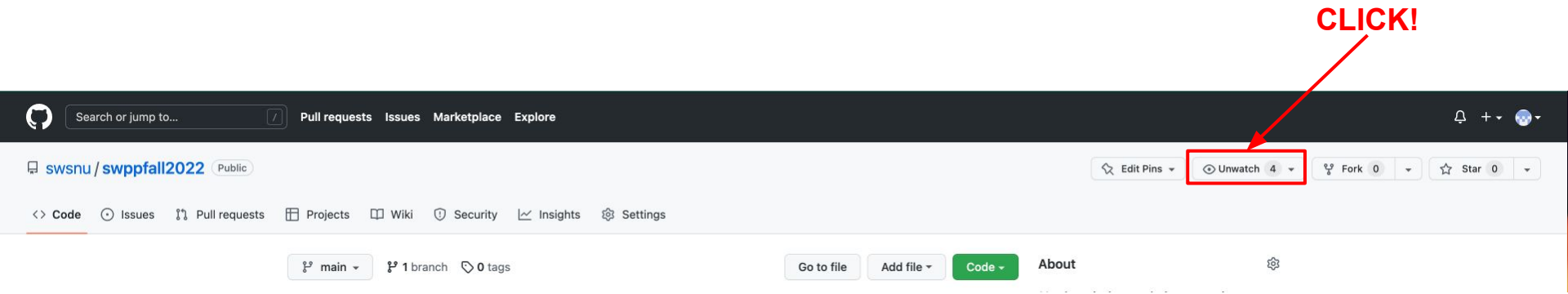
We may have some in-class exercises during the practice sessions. **Your attendance will be checked with the in-class exercises.**

# Please **Watch** swppfall2022 Repo!

All class-wide announcements will be posted on GitHub.

Make sure you watch the class repo to receive notifications.

- https://github.com/swsnu/swppfall2022



# *Docker* container as a grading environment

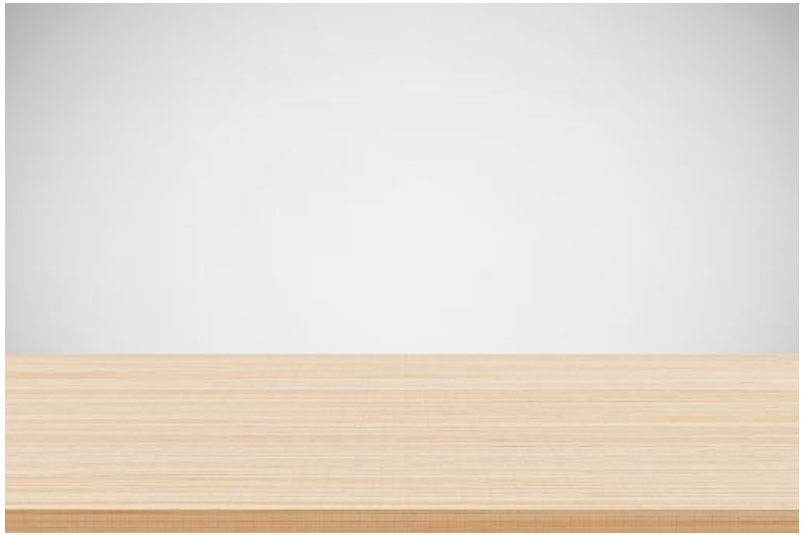
- We will provide the testing environment where your codes will be graded since each student has a different development environment (e.g., OS).
- Especially, a *Docker* container will be used for grading **all your homework**.
- We suggest you go through the following slides so that TAs can run your codes properly in the same environment.

# What is Docker? Why Docker?

- *Docker* provides an isolated environment, called a container, for each application.
- Docker enables you to separate your applications from your infrastructure.
- So, even when you are using *Windows*, you can run your program on any other environments (e.g., *Ubuntu, Alpine* …).
- For us, we can share the environment through the container; a container with *node(v14.17.6) on Linux* will be used.
    - https://hub.docker.com/_/node

# What is Docker? Why Docker?

- Start from an empty desk? 😧



# What is Docker? Why Docker?

- Start from a prepared desk for *each app*! 😎 (+ save + share)


# What is Docker? Why Docker?

**NOTE**: This material covers only the minimum requirements for checking the assignments.

If you are interested, you can check more on the details in the following links:

- **Introductions**
  - English: Introduction to Docker containers
  - Korean: Docker 컨테이너 소개
- **Practice**
  - English: Build a containerized web application with Docker
  - Korean: Docker를 사용하여 컨테이너화된 웹 애플리케이션 빌드

# Install Docker (Ubuntu)

```
$ sudo apt-get update
$ sudo apt-get install apt-transport-https ca-certificates curl
software-properties-common
$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
$ sudo add-apt-repository \
   "deb [arch=amd64] https://download.docker.com/linux/ubuntu \
   $(lsb_release -cs) \
   stable"
$ sudo apt-get update
$ sudo apt-get install docker-ce
```

Recommended version for Docker Engine: 20.10.XX

# Install Docker

```
$ sudo docker version
```

```
# output:
Client: Docker Engine - Community
 Version:           20.10.05
 API version:       1.41
 Go version:        go1.13.15
 Git commit:        55c4c88
 Built:             Tue Mar  2 20:13:00 2021
 OS/Arch:           darwin/amd64
 Experimental:      true

Server: Docker Engine - Community
 Engine:
  Version:          20.10.05
  API version:      1.41 (minimum version 1.12)
  Go version:       go1.13.15
  ...
```

# Install Docker

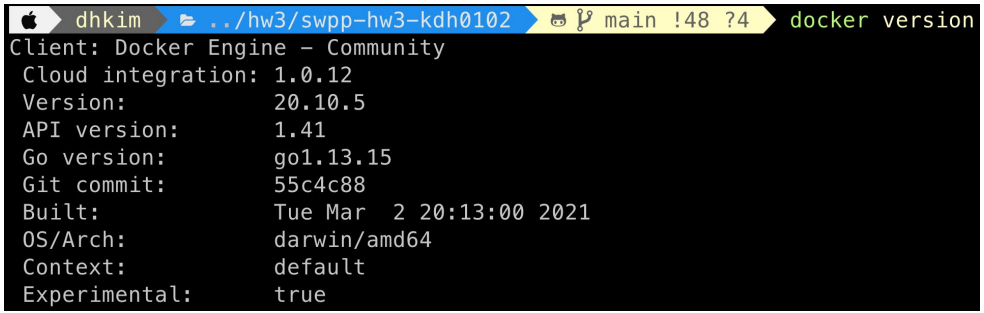
For Mac: https://docs.docker.com/desktop/mac/install/
For Windows: https://docs.docker.com/desktop/windows/install/

# Simple Docker Tutorial

# 0. Check if *Docker* is installed

- Both the following two commands should show the proper messages:

```
$ docker version
```




```
$ docker ps -a
```

# 1. Pull *Docker* image

- Pull image from remote => check image on local

```
$ docker pull snuspl/swpp:session1_multi_extend
```

```
jaewoo@maengjacBookAir swpp % docker pull snuspl/swpp:session1_multi_extend

session1_multi_extend: Pulling from snuspl/swpp
dfd1fb90fd33: Pull complete
ddcd91967bc4: Pull complete
c5145c2d06e4: Pull complete
510f607bbfa7: Pull complete
acf12c903f74: Pull complete
cd27bcdd46e1: Pull complete
dfecd824708c: Pull complete
4e901650996e: Downloading [=====>                                             ]  50.65MB/430.1MB
54e043d24fe1: Download complete
3f4d346805d4: Downloading [==================>                                ]  7.007MB/20.09MB
32af3c4803c3: Download complete
4f4fb700ef54: Waiting
```

```
$ docker images
```

```
jaewoo@maengjacBookAir swpp % docker images
REPOSITORY                    TAG                   IMAGE ID       CREATED         SIZE
snuspl/swpp                   session1_multi_extend   a880886f1db8   3 hours ago     817MB
```

# 2. Run *Docker* container

- Run a *docker* container with the downloaded image.
- Use `docker run` command as follows:

```
$ docker run -it \
-p 127.0.0.1:3000:3000 \
-v ${PWD}:/home \
--name session1 \
snuspl/swpp:session1_multi_extend \
/bin/bash
```

Bind host port 3000 to container port 3000.

Mount the current directory to the container path (*/home*).

Container name.

# 3. Basic Container Management

- check local containers
  (-a option: see even the stopped containers)

```
$ docker ps -a
```

- stop a running container

```
$ docker stop {container_name}
```

- start a stopped container

```
$ docker start {container_name}
```

# 3. Basic Container Management

- run commands in a running container
  (usually used to attach to running containers)

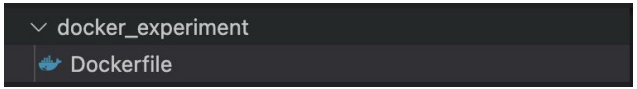
```
$ docker exec -it {container_name} /bin/bash
```

- remove a container

```
$ docker rm {container_name}
```

# 4. Docker Build

- Make a file named Dockerfile
  - **FROM**: basic docker image to start from
  - **RUN**: shell commands you want to run in the image
  - **WORKDIR**: note the starting directory in the image
  - more syntax (https://docs.docker.com/engine/reference/builder/)




```
1  FROM snuspl/swpp:session1_multi_extend
2
3  RUN apt-get install npm -y
4
5  WORKDIR /home
```

# 4. Docker Build

- Build an image with Dockerfile
  - navigate to the directory with Dockerfile

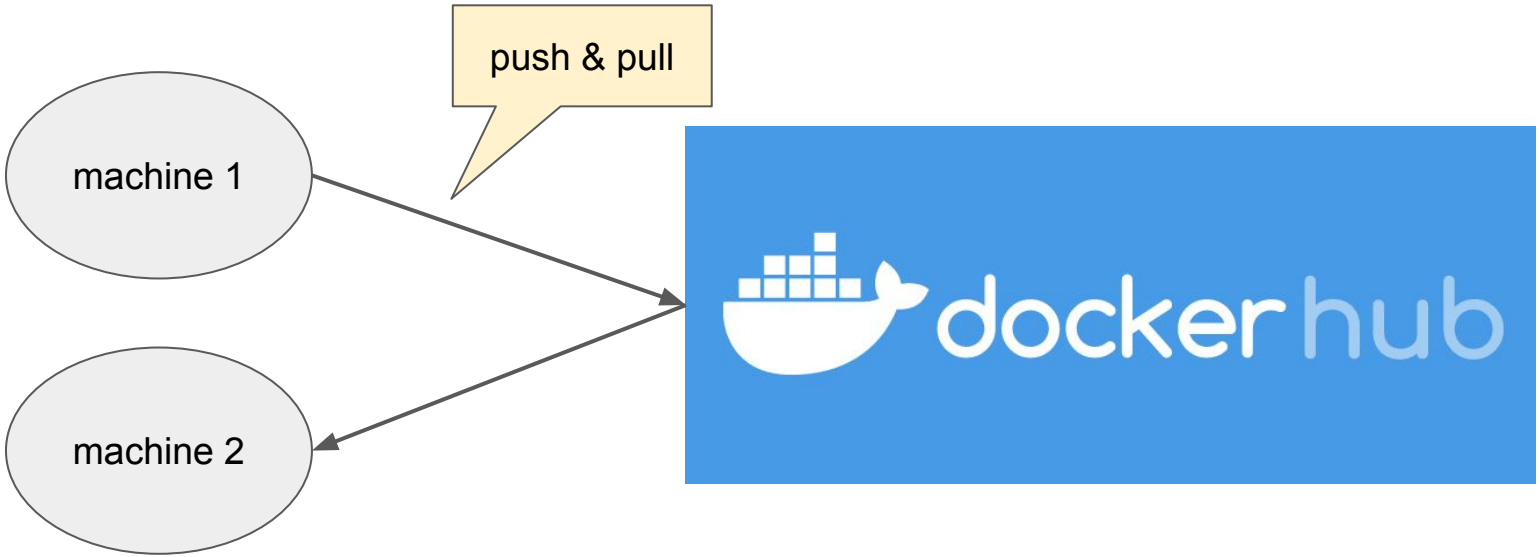
```
$ docker build -t {user_name}/swpp:session1_with_npm .
```

```
jaewoo@maengjaeuui-MacBookAir docker_experiment % docker build -t snuspl/swpp:session1_with_npm .
[+] Building 11.9s (4/6)
 => [internal] load build definition from Dockerfile                                                                0.0s
 => => transferring dockerfile: 110B                                                                                0.0s
 => [internal] load .dockerignore                                                                                   0.0s
 => => transferring context: 2B                                                                                     0.0s
 => [internal] load metadata for docker.io/snuspl/swpp:session1                                                     0.0s
 => [1/3] FROM docker.io/snuspl/swpp:session1                                                                       0.1s
 => [2/3] RUN apt-get install npm -y                                                                               11.7s
 => => # Get:31 http://ports.ubuntu.com/ubuntu-ports bionic-updates/main arm64 libubsan0 arm64 7.5.0-3ubuntu1~18.04 [117 kB]
 => => # Get:32 http://ports.ubuntu.com/ubuntu-ports bionic-updates/main arm64 libgcc-7-dev arm64 7.5.0-3ubuntu1~18.04 [833 kB]
 => => # Get:33 http://ports.ubuntu.com/ubuntu-ports bionic-updates/main arm64 gcc-7 arm64 7.5.0-3ubuntu1~18.04 [7772 kB]
 => => # Get:34 http://ports.ubuntu.com/ubuntu-ports bionic-updates/main arm64 gcc arm64 4:7.4.0-1ubuntu2.3 [5208 B]
 => => # Get:35 http://ports.ubuntu.com/ubuntu-ports bionic-updates/main arm64 libstdc++-7-dev arm64 7.5.0-3ubuntu1~18.04 [1471 kB]
 => => # Get:36 http://ports.ubuntu.com/ubuntu-ports bionic-updates/main arm64 g++-7 arm64 7.5.0-3ubuntu1~18.04 [8077 kB]
```

```
jaewoo@maengjaeuui-MacBookAir docker_experiment % docker images
REPOSITORY                    TAG                 IMAGE ID       CREATED          SIZE
snuspl/swpp                   session1_with_npm   5d049e73a2eb   28 seconds ago   1.04GB
snuspl/swpp                   session1            e61827f20122   4 hours ago      816MB
```

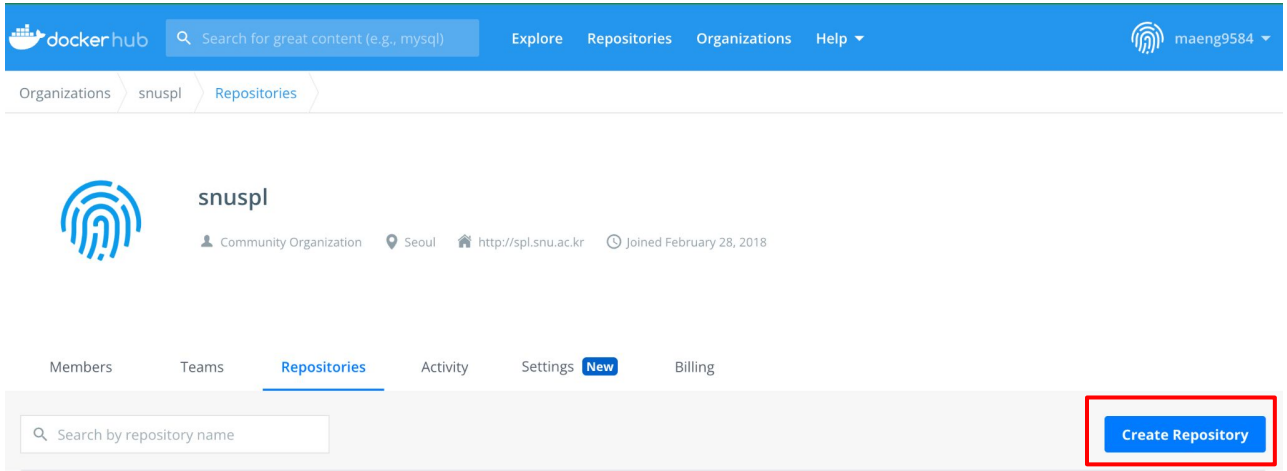
# 4. Docker Hub

- Remote storage that enables saving and sharing docker images!



# 4. Docker Hub

1. make an account at [docker hub]
2. create a repository



# 4. Docker Hub

3. login with docker hub account in terminal

```
$ docker login
```

4. push local image

```
$ docker push {docker_account}/{repository_name}:{tag}
# e.g.
$ docker push jaewoo/swpp:session1_with_npm
```