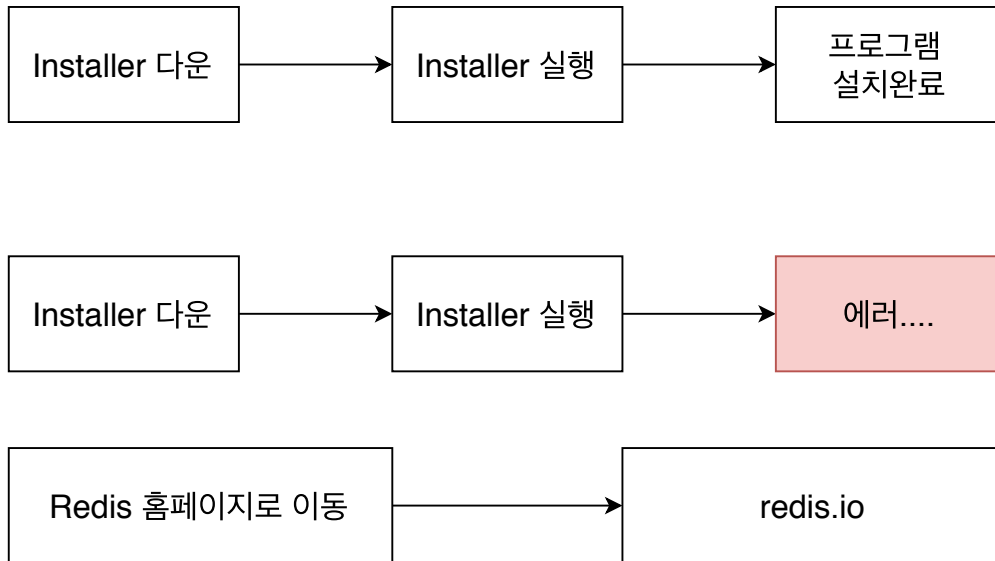
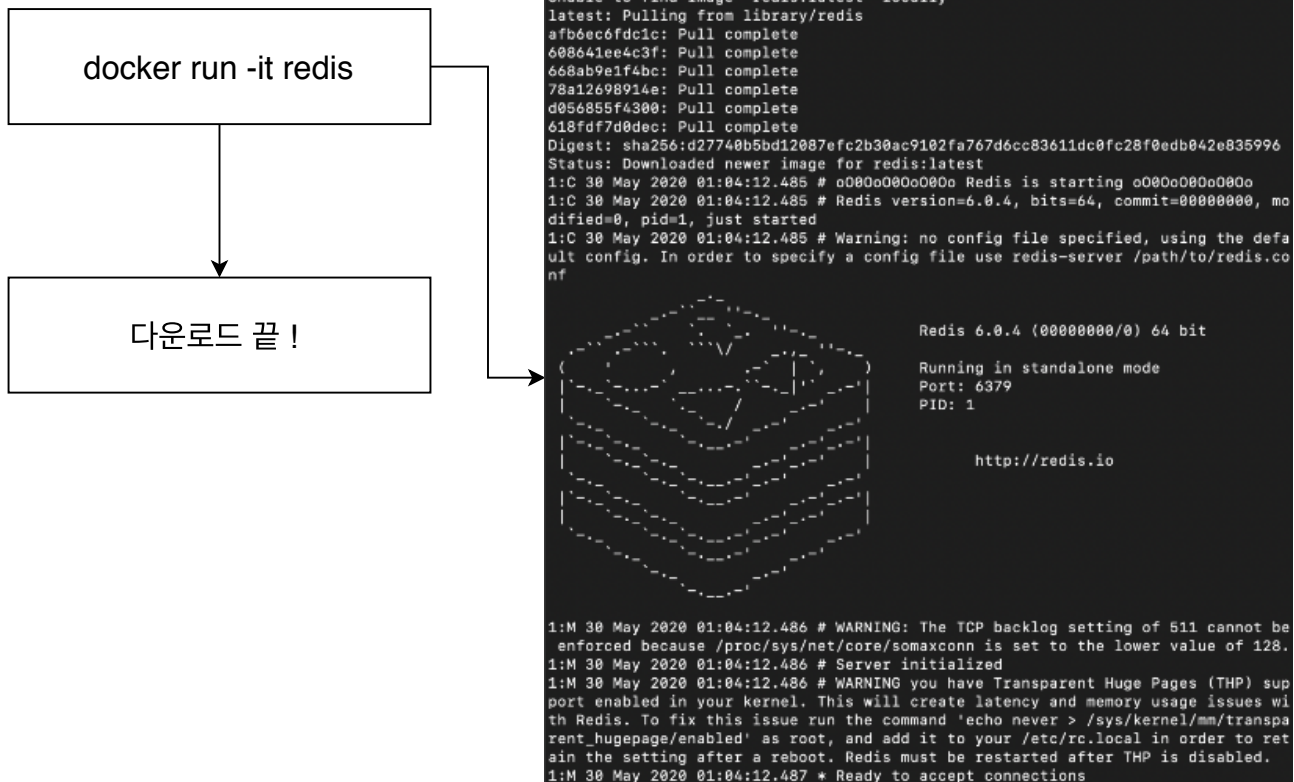


## 도커를 쓰는 이유



## 도커로 Redis 받는 과정



## 도커란 무엇인가?

### 일반적인 컨테이너의 개념

#### 컨테이너

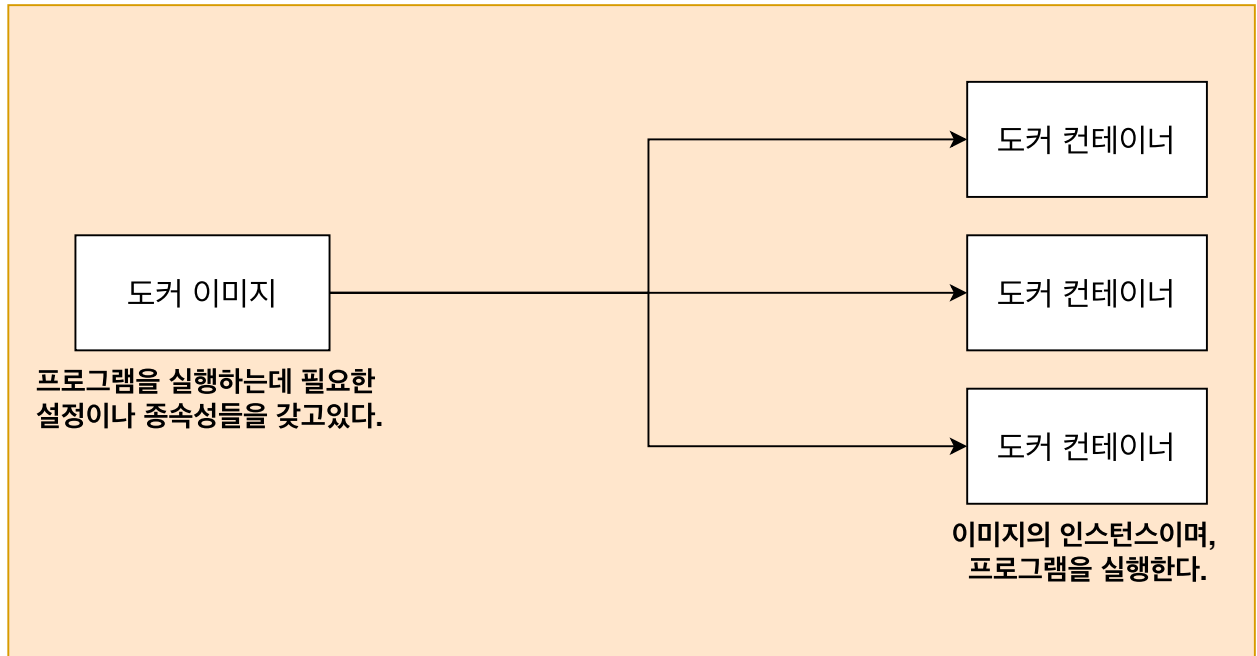


### 서버에서의 컨테이너의 개념

#### 컨테이너

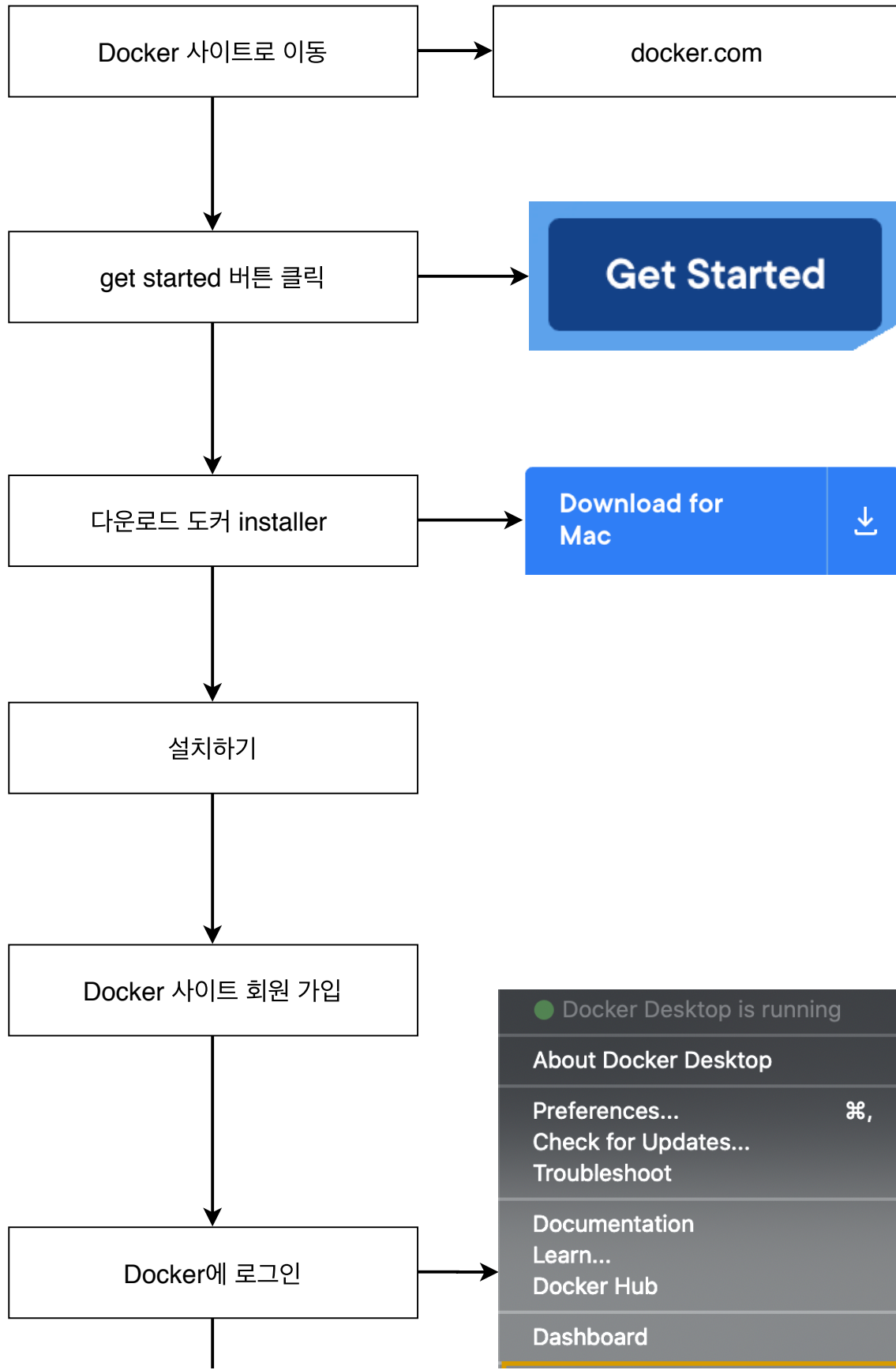


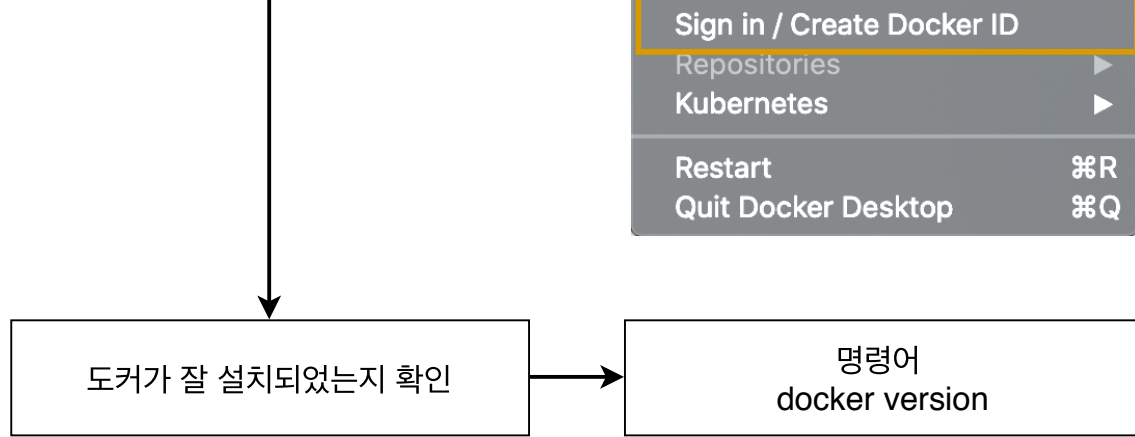
## 도커 이미지와 도커 컨테이너 정의



## MacOS를 위한 도커 다운받기

### 도커를 다운받는 순서

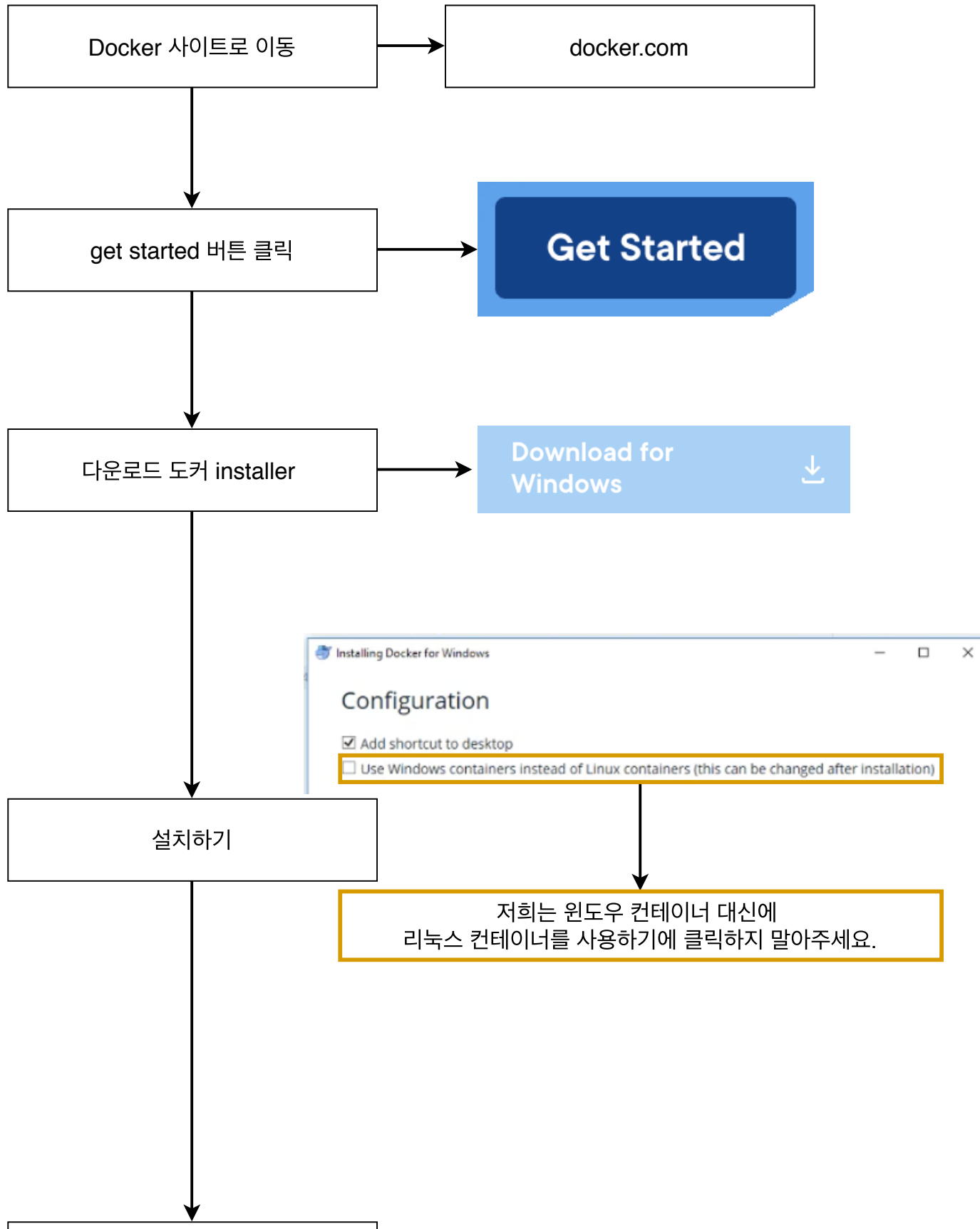




**이렇게 해서 MAC OS를 위한 도커를 다운받았습니다.**  
**이제는 Window OS를 위한 도커를 다운 받아 보겠습니다.**

## Window를 위한 도커 다운받기

### 도커를 다운받는 순서

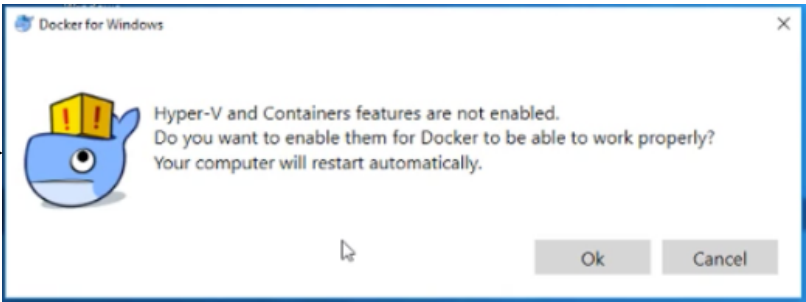


Docker 사이트 회원 가입

설치 후 나타나는 아이콘 클릭

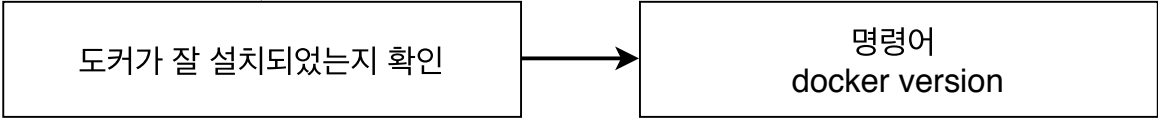


컴퓨터 재시작



Docker에 로그인

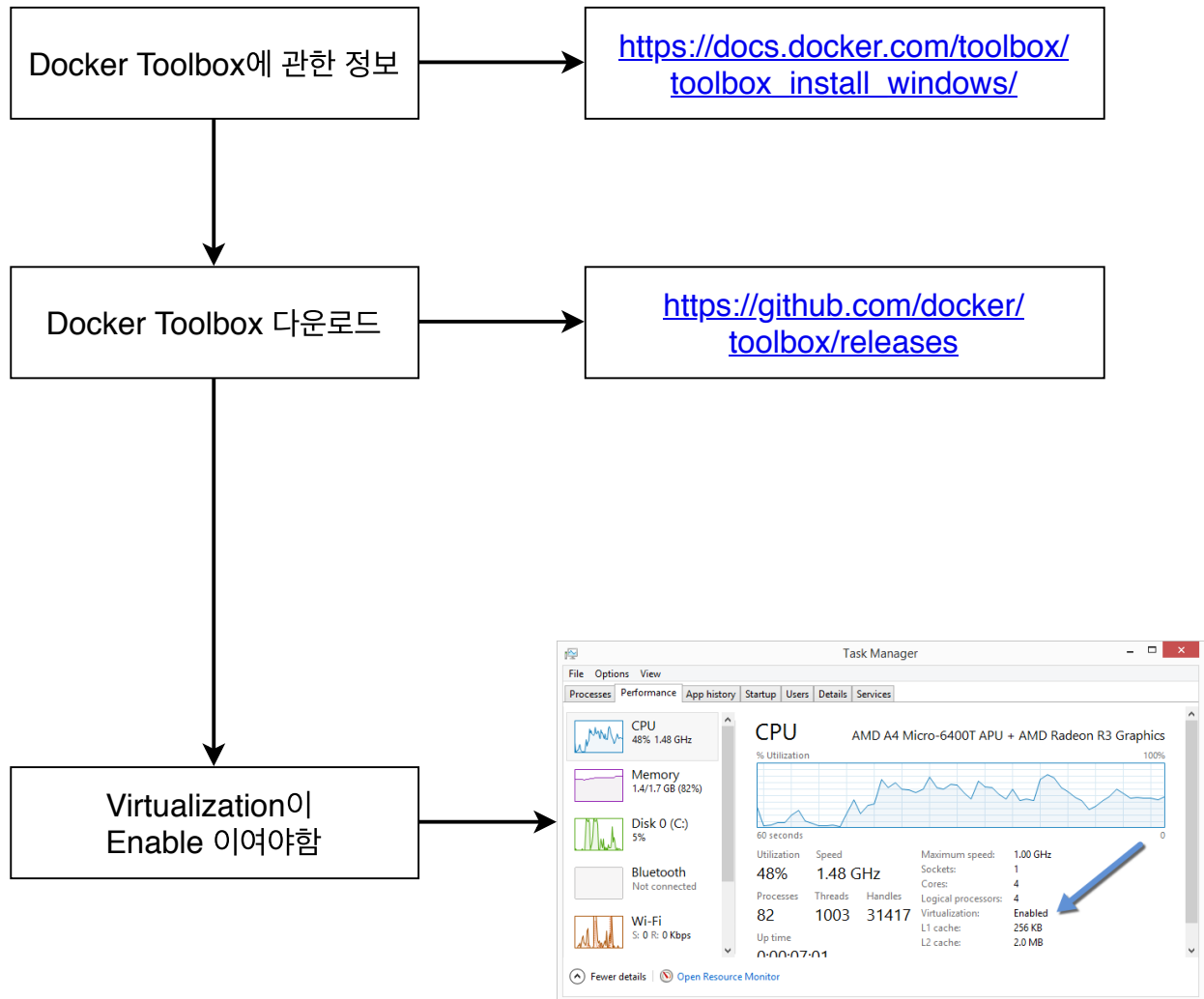






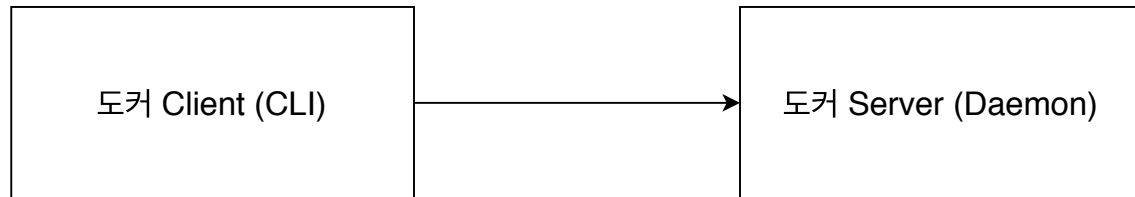
## Window를 Home 유저를 위한 도커 다운받기

Docker Community Edition for Windows를 사용하려면  
Hyper-V라는 것이 필요한데  
Home 버전에서는 Hyper-V 라는 것을 지원하지 않기에  
일반 윈도우 버전 도커를 사용할수 없습니다.  
그러기에 Docker Toolbox를 설치해서 사용해야 합니다.



**Docker Toolbox에서는 로컬호스트에 접근할때  
Localhost 대신에 192.168.99.100을 사용해야 합니다.**

## Docker를 사용할때의 흐름 감잡기



```
jaewon@jaewonui-MacBookPro ~ % docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
0e03bdcc26d7: Pull complete
Digest: sha256:6a65f928fb91fcfb963f7aa6d57c8eeb426ad9a20c7ee045538ef34847f44f1
Status: Downloaded newer image for hello-world:latest

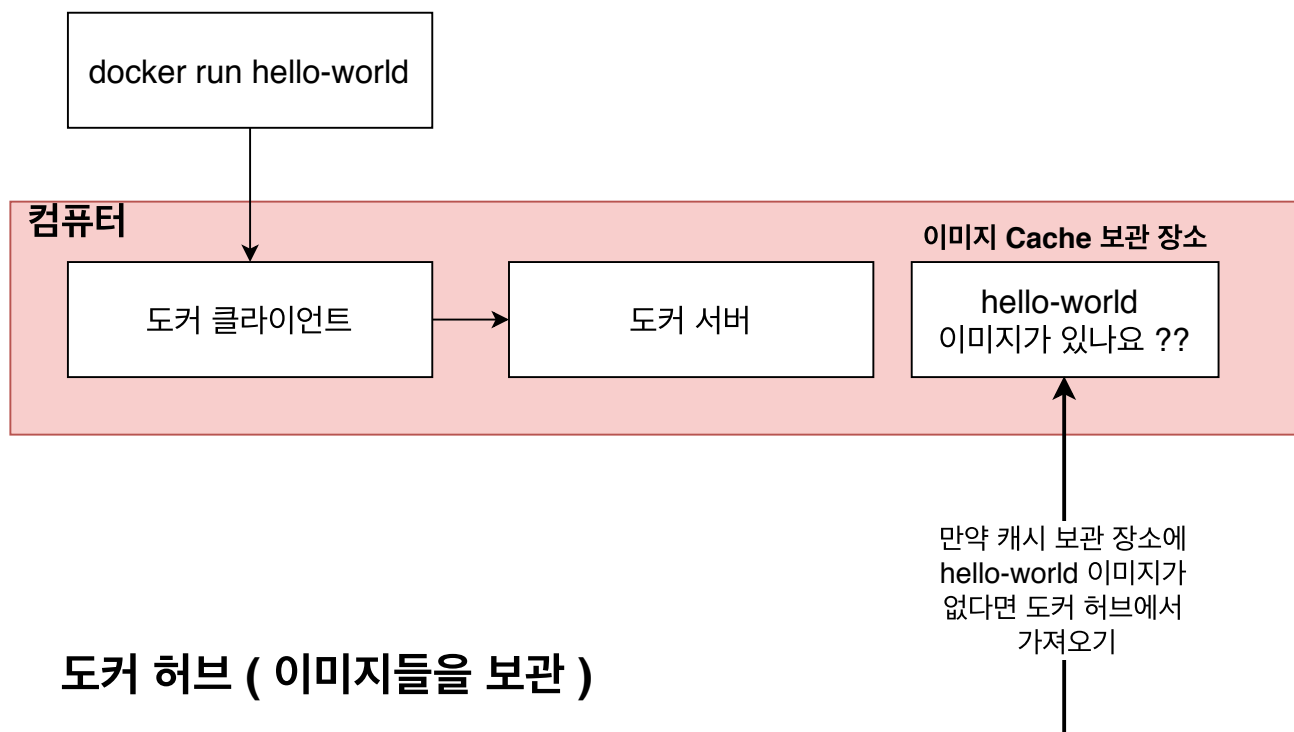
Hello from Docker!
This message shows that your installation appears to be working correctly.

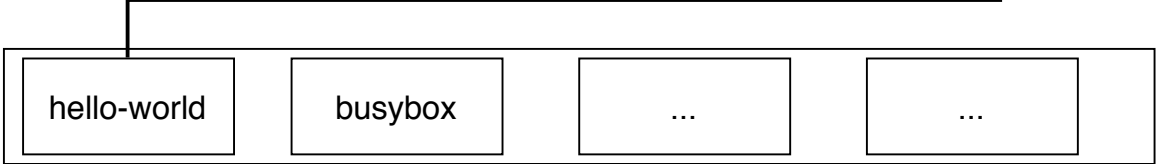
To generate this message, Docker took the following steps:
1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
   (amd64)
3. The Docker daemon created a new container from that image which runs the
   executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it
   to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

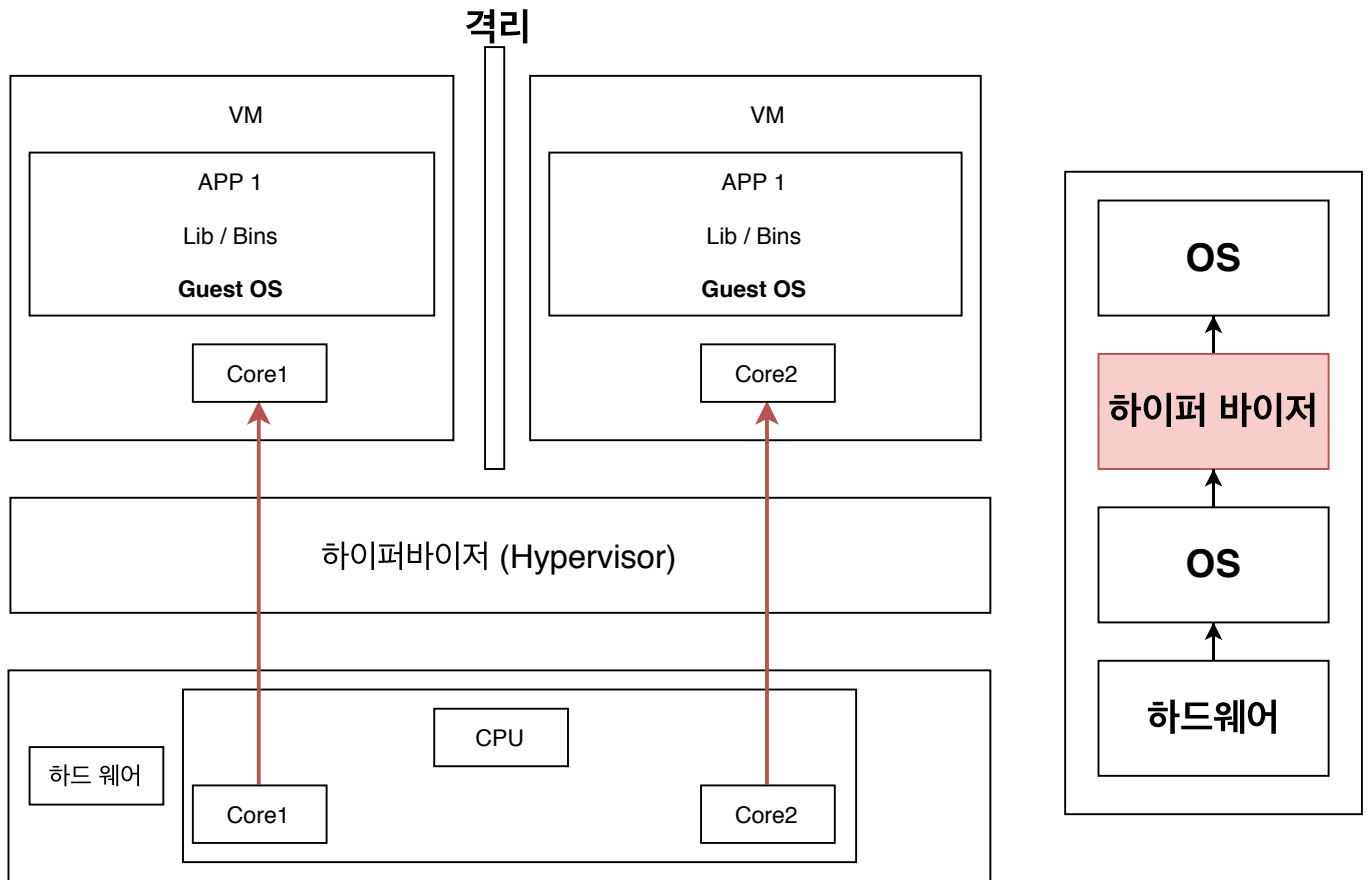
For more examples and ideas, visit:
https://docs.docker.com/get-started/
```





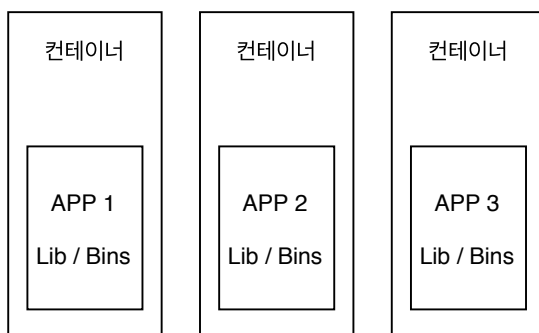
## 도커와 기존의 가상화 기술과의 차이를 통한 컨테이너 이해

### 하이퍼 바이저 기반의 VM 구조

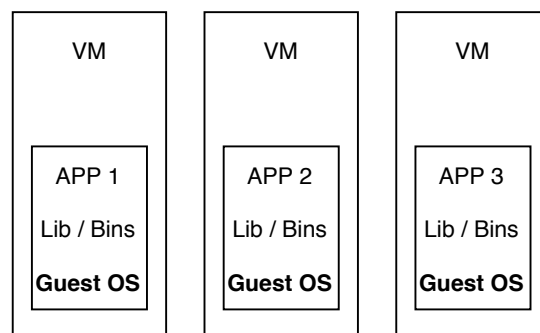


### 이러한 가상화 기술에서 나온 컨테이너 가상화 기술

#### 도커

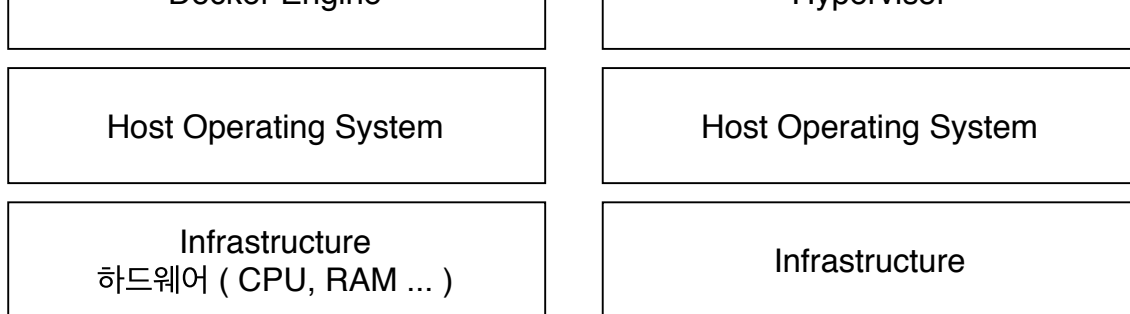


#### 기존 가상화 기술( VM )

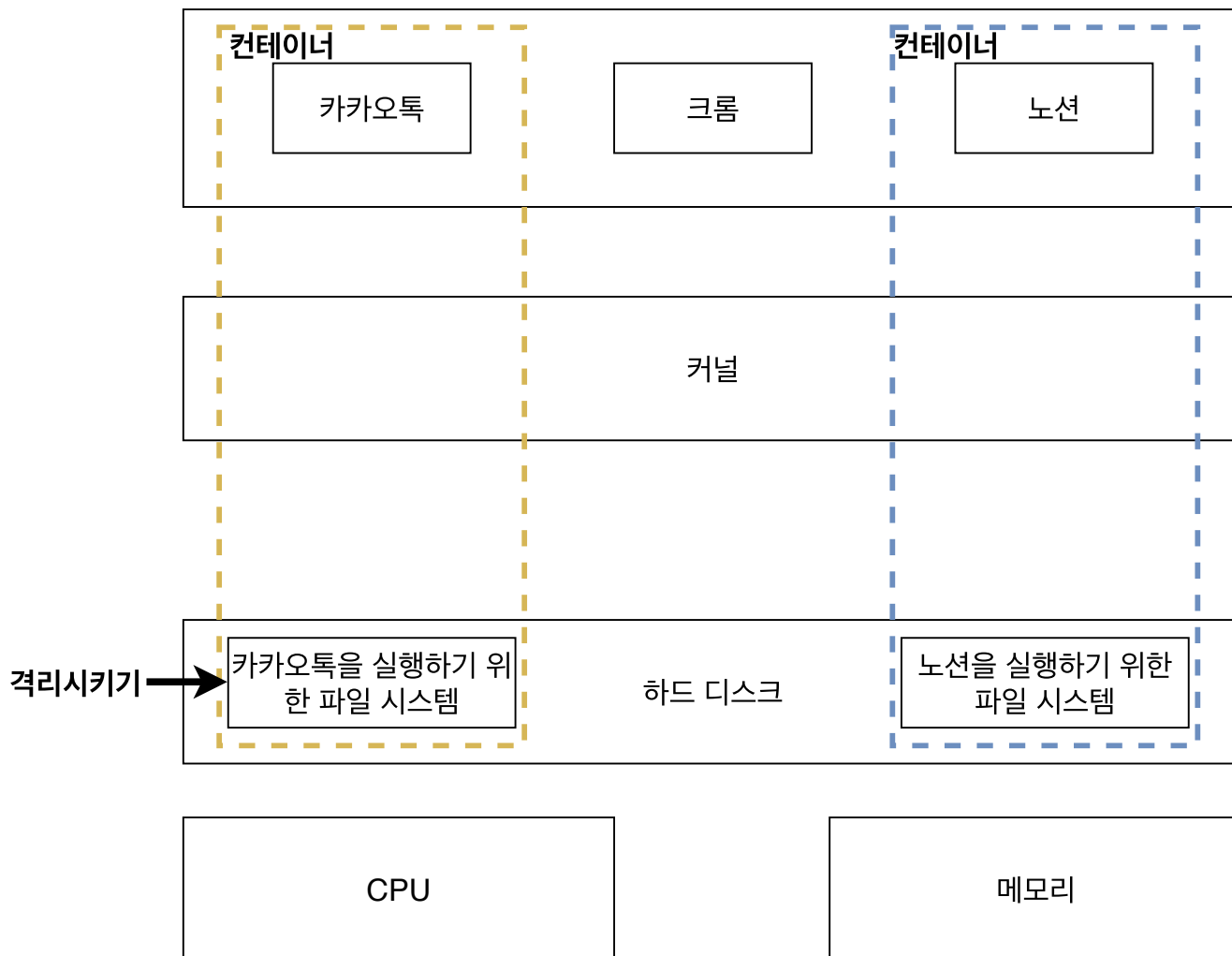


Docker Engine

Hypervisor



컴퓨터에 실행되고 있는 프로세스들



## 이미지로 컨테이너를 만들기

### 이미지

#### 1. 시작시 실행 될 명령어

run kakaotalk

#### 2. 파일 스냅샷

카카오톡 파일

### 이미지

#### 시작시 실행 될 명령어

run kakaotalk

#### 파일 스냅샷

카카오톡 설치에  
필요한 파일

프로세스를  
작동시키는데  
필요한 양

### 컨테이너

커널

하드 디스크

네트워크

RAM

CPU

### 이미지

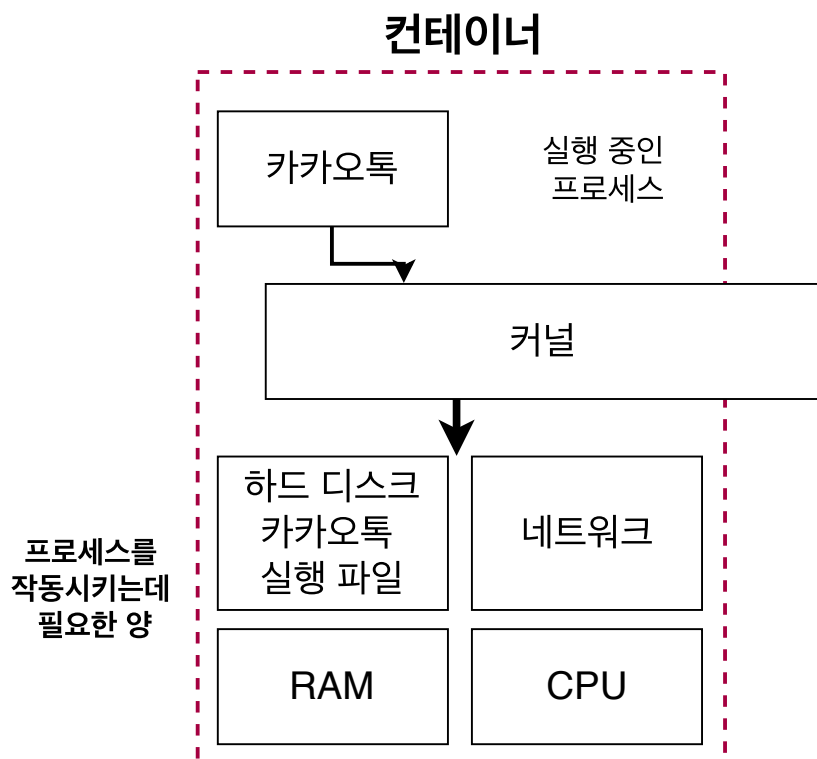
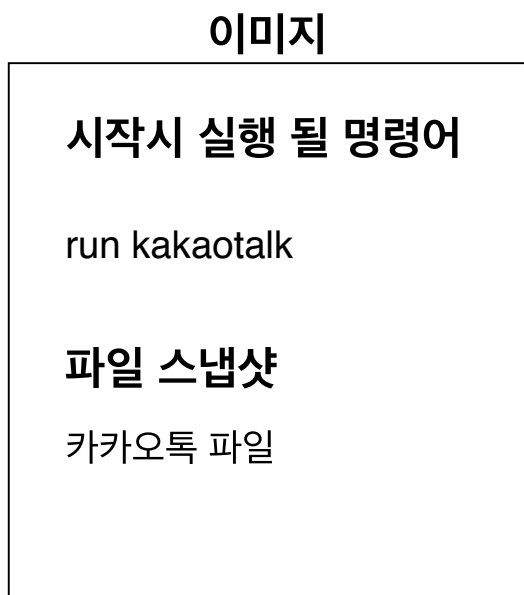
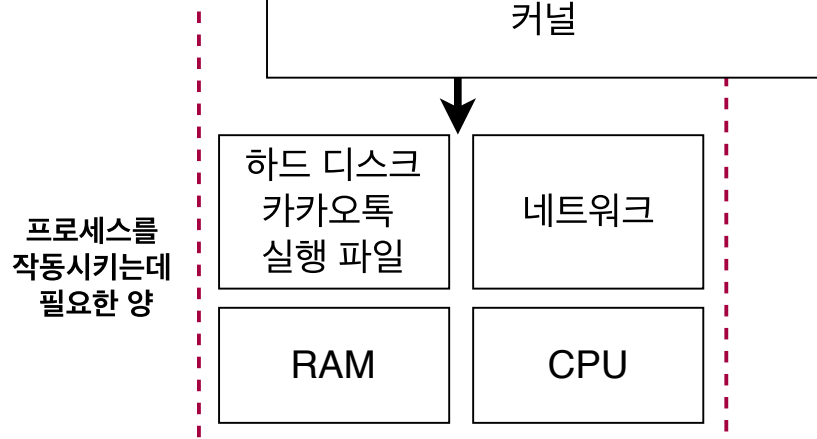
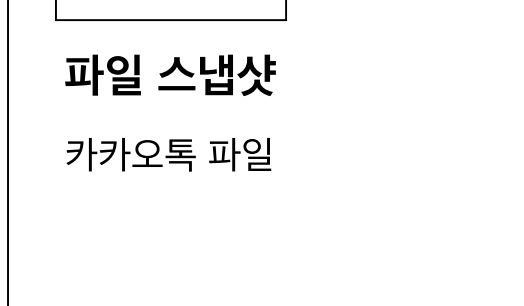
#### 시작시 실행 될 명령어

run kakaotalk

### 컨테이너

run  
kakaotalk

실행 중인  
프로세스



## Cgroup, 네임스페이스를 쓸수 있는 이유

컴퓨터에 실행되고 있는 프로세스들

