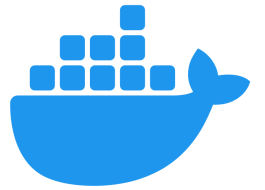


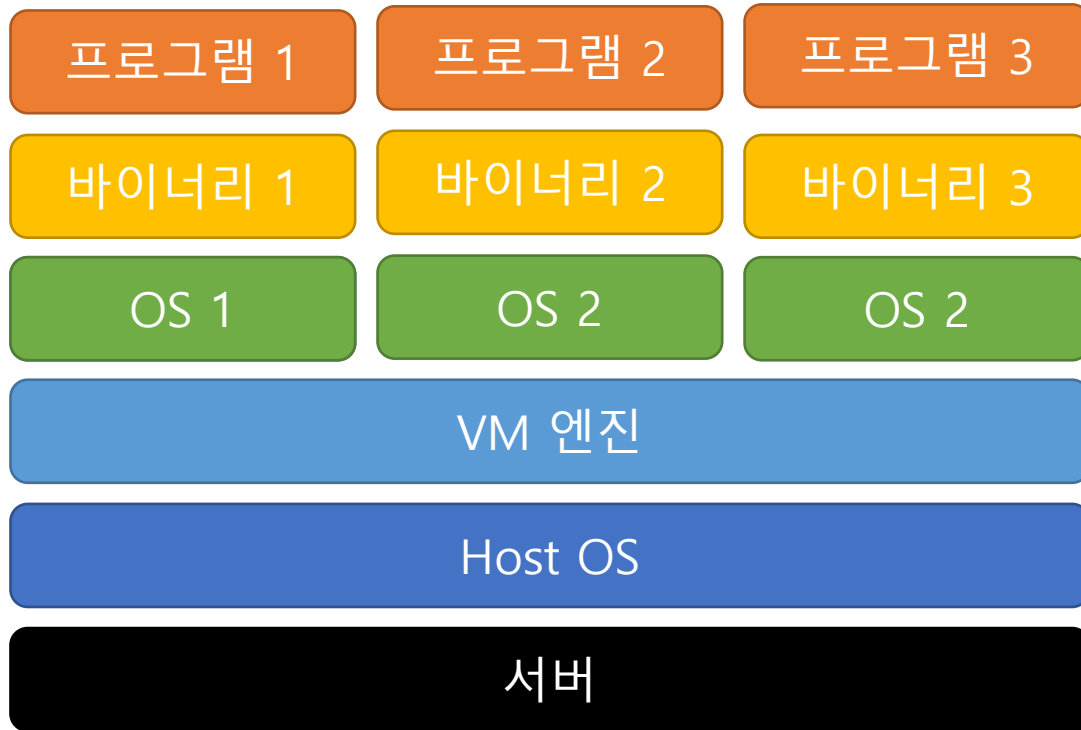
변서형

---

# Docker

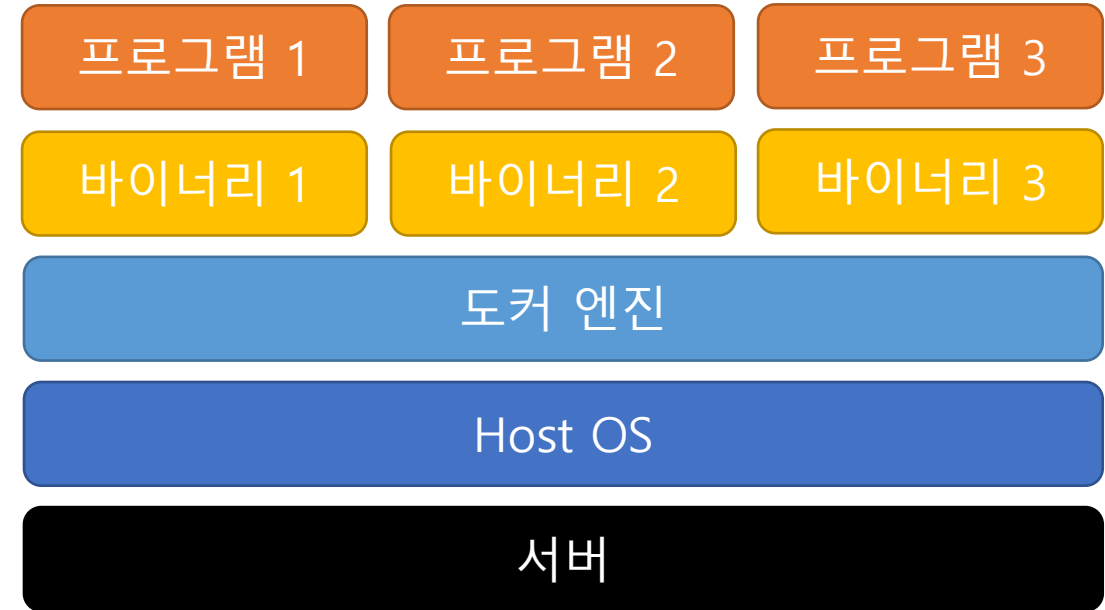


# VM vs Docker



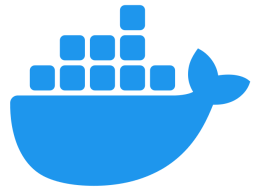
VM (가상머신)

- 높은 보안성
- 커널 공유 -> 멀티 OS 가능
- 무거운 프로그램



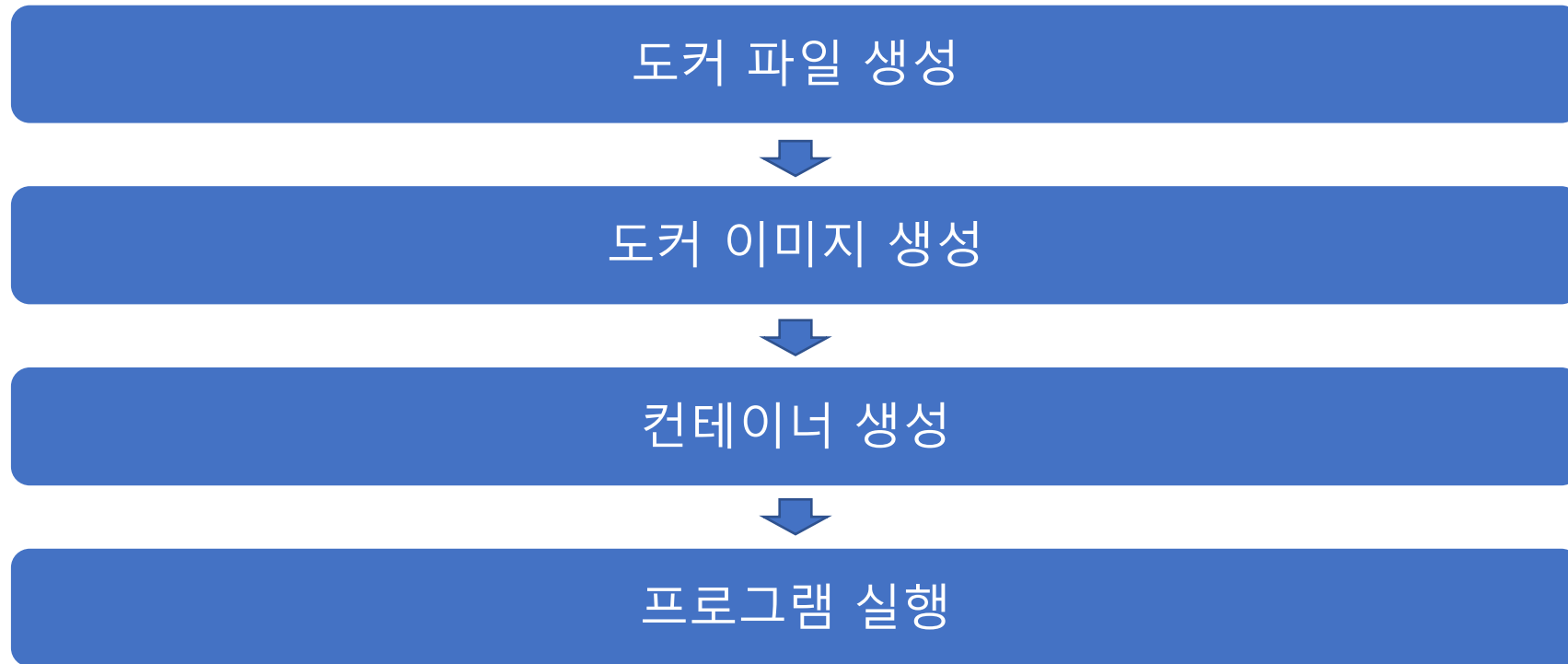
Docker

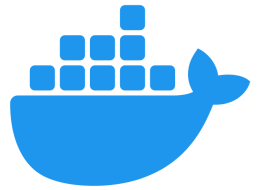
- 효율적
- 뛰어난 이식성
- 낮은 격리 기능



# Docker를 통한 프로그램의 실행 순서

---





# Dockerfile 필수 요소

---

수정  
빈도가  
높을수록  
아래로

**FROM** 기본 이미지 - 파이썬 버전이랑 OS 지정

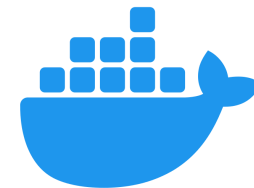
**WORKDIR** 작업 폴더 생성/지정

**COPY** ~복사 - 가져올 코드 선택

**RUN** 터미널에 작성할 명령어 사전 입력 - requir~.txt 다운로드 명령어 등

**CMD** 컨테이너 실행 시 명령 지정

# 도커 실행 실습



## Python 코드

```
1 import os
2 import platform
3 import socket
4
5 def get_container_id():
6     try:
7         with open('/proc/self/cgroup', 'r') as f:
8             for line in f:
9                 if 'docker' in line:
10                     return line.split('/')[-1].strip()
11     except FileNotFoundError:
12         return "Not in a Docker container"
```

도커 컨테이너 존재 확인

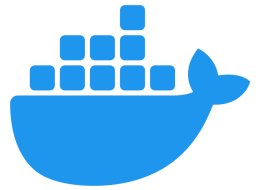
```
14 print("=== System Information ===")
15 print(f"Operating System: {platform.system()}")
16 print(f"OS Version: {platform.version()}")
17 print(f"Architecture: {platform.machine()}")
18 print(f"Python Version: {platform.python_version()}")
19 print(f"Hostname: {socket.gethostname()}")
20 print(f"Container ID: {get_container_id()}")
21 print(f"Current Working Directory: {os.getcwd()}")
```

실행 호스트 정보 확인

```
22 print("\n=== Environment Variables ===")
23
24 for key, value in os.environ.items():
25     print(f"{key}: {value}")
26
```

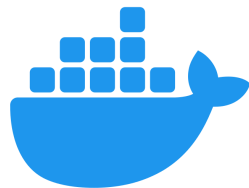
```
27 # 파일 생성 테스트
28 test_file = '/tmp/docker_test.txt'
29 try:
30     with open(test_file, 'w') as f:
31         f.write("Docker test successful!")
32         print(f"\nSuccessfully created file: {test_file}")
33 except Exception as e:
34     print(f"\nFailed to create file: {e}")
35
36 # 추가: Docker 환경 확인
37 print("\n=== Docker Environment Check ===")
38 if os.path.exists('/.dockerenv'):
39     print("This is running inside a Docker container!")
40 else:
41     print("This is not running in a Docker container.")
42
43 # 추가: Docker-specific 환경 변수 확인
44 docker_env_var = os.environ.get('DOCKER_ENV_TEST')
45 if docker_env_var:
46     print(f"Docker-specific environment variable found: DOCKER_ENV_TEST = {docker_env_var}")
47 else:
48     print("No Docker-specific environment variable found.")
```

# 도커 실행 실습



## Dockerfile

```
test_docker.py × Dockerfile ×  
B2_project > Dockerfile > ...  
1 FROM python:3.11-slim  
2  
3 WORKDIR /app  
4  
5 COPY test_docker.py .  
6  
7 ENV DOCKER_ENV_TEST="This is a Docker container"  
8  
9 CMD ["python", "test_docker.py"]
```



# 도커 실습 결과 – Raspi 5

---

```
seju@seju:~/Documents $ /bin/python /home/seju/Documents/B2_project/test_docker.py
```

```
=== System Information ===
```

```
Operating System: Linux
```

```
OS Version: #1 SMP PREEMPT Debian 1:6.1.63-1+rpt1 (2023-11-24)
```

```
Architecture: aarch64
```

```
Python Version: 3.11.2
```

```
Hostname: seju
```

```
Container ID: None
```

```
Current Working Directory: /home/seju/Documents
```

```
(중략)
```

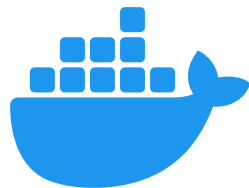
```
Successfully created file: /tmp/docker_test.txt
```

```
=== Docker Environment Check ===
```

```
This is not running in a Docker container.
```

```
No Docker-specific environment variable found.
```

```
seju@seju:~/Documents $
```



# 도커 실습 결과 - 데스크탑

## 도커 이미지 빌드 전

```
PS C:\Users\Wa6690> & C:/Users/a6690/AppData/Local/Microsoft/WindowsApps/python3.11.exe  
"c:/Users/a6690/OneDrive/바탕 화면/app/test_docker.py"
```

=== System Information ===

Operating System: Windows

OS Version: 10.0.22631

Architecture: AMD64

Python Version: 3.11.9

Hostname: seaju\_co

Container ID: Not in a Docker container

Current Working Directory: C:\Users\Wa6690

(중략)

Failed to create file: [Errno 2] No such file or directory: '/tmp/docker\_test.txt'

=== Docker Environment Check ===

This is not running in a Docker container.

No Docker-specific environment variable found.

```
PS C:\Users\Wa6690> docker build -t docker-test-app .
```

```
[+] Building 0.0s (1/1) FINISHED
```

```
docker:desktop-linux
```

```
=> [internal] load build definition from Dockerfile 0.0s
```

```
=> => transferring dockerfile: 2B 0.0s
```

```
ERROR: failed to solve: failed to read dockerfile: open Dockerfile: no such file or directory
```

## 컨테이너 생성 후

```
PS C:\Users\Wa6690\OneDrive\바탕 화면\app> docker run --name docker-  
test-container docker-test-app
```

=== System Information ===

Operating System: Linux

OS Version: #1 SMP Fri Mar 29 23:14:13 UTC 2024

Architecture: x86\_64

Python Version: 3.11.9

Hostname: 63a868976bf5

Container ID:

63a868976bf559e0a52a580a51e33cc97b140ef88e18d5b7bc3b03a3bf04935d

Current Working Directory: /app

(중략)

Successfully created file: /tmp/docker\_test.txt

=== Docker Environment Check ===

This is running inside a Docker container!

Docker-specific environment variable found: DOCKER\_ENV\_TEST = This is a  
Docker container

```
PS C:\Users\Wa6690\OneDrive\바탕 화면\app>
```