

1. 「Ubuntu+C 言語コンパイラ&VIM」の資料を参考に compose.yaml と Dockerfile を使用して Debian(タグ 11.3)+OpenJDK11+vim コンテナを構築してください。

コンテナは Windows と AWS Ubuntu 上2つのプラットフォームで構築してください。

Hello.java をコンテナの /usr/local/src に配置してください。

コンテナ構築後、Hello.java をコンパイルして実行してください。

📎 コンテナ構築過程、コンパイルと実行の画面をキャプチャして貼り付けてください。

compose.yaml、Dockerfile を提出してください。

```
ubuntu@ip-172-31-3-122:~/docker/openjdk_debian$ ls
Dockerfile Hello.java compose.yaml
ubuntu@ip-172-31-3-122:~/docker/openjdk_debian$
ubuntu@ip-172-31-3-122:~/docker/openjdk_debian$
ubuntu@ip-172-31-3-122:~/docker/openjdk_debian$
ubuntu@ip-172-31-3-122:~/docker/openjdk_debian$ docker compose up -d
[+] Building 67.0s (8/8) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 301B
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load metadata for docker.io/library/debian:11.3
=> [internal] load build context
=> => transferring context: 32B
=> [1/3] FROM docker.io/library/debian:11.3@sha256:859ea45db307402ee024b153c7a63ad4888eb4751921abbef68679fc73c4c739
=> CACHED [2/3] COPY Hello.java /usr/local/src
=> [3/3] RUN apt update && apt -y install openjdk-11-jdk && apt-get -y install vim
=> exporting to image
=> => exporting layers
=> => writing image sha256:bd6b70ce825f4bdc3b473fadae706484752207832e88d079f2f08db877796aa3
=> => naming to docker.io/library/openjdk_debian-debian
[+] Running 2/2
? Network openjdk_debian_default Created
? Container openjdk_debian-debian-1 Started
ubuntu@ip-172-31-3-122:~/docker/openjdk_debian$
```

```
root@ec9d33ab71a6:/usr/local/src# ls
Hello.java
root@ec9d33ab71a6:/usr/local/src#
root@ec9d33ab71a6:/usr/local/src#
root@ec9d33ab71a6:/usr/local/src#
root@ec9d33ab71a6:/usr/local/src# javac Hello.java
root@ec9d33ab71a6:/usr/local/src#
root@ec9d33ab71a6:/usr/local/src#
root@ec9d33ab71a6:/usr/local/src# java Hello
No.1 : Hello Java
No.2 : Hello Java
No.3 : Hello Java
No.4 : Hello Java
No.5 : Hello Java
No.6 : Hello Java
No.7 : Hello Java
No.8 : Hello Java
No.9 : Hello Java
No.10 : Hello Java
No.11 : Hello Java
No.12 : Hello Java
No.13 : Hello Java
No.14 : Hello Java
No.15 : Hello Java
No.16 : Hello Java
No.17 : Hello Java
No.18 : Hello Java
No.19 : Hello Java
No.20 : Hello Java
root@ec9d33ab71a6:/usr/local/src#
```

Docker 実習 13

```
C:\Users\ >docker compose up -d
[+] Building 85.3s (8/8) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 32B
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load metadata for docker.io/library/debian:11.3
=> [internal] load build context
=> => transferring context: 32B
=> [1/3] FROM docker.io/library/debian:11.3
=> CACHED [2/3] COPY Hello.java /usr/local/src
=> [3/3] RUN apt update && apt-get -y install openjdk-11-jdk && apt-get -y install vim
=> exporting to image
=> => exporting layers
=> => writing image sha256:e581c560e35b1d3ad53a3079c343ba3ec9e76254c0d142fe0a2a07536e129b2e
=> => naming to docker.io/library/openjdk_debian-debian
[+] Running 2/2
 - Network openjdk_default Created
 - Container openjdk_debian-debian-1 Started
```

```
root@0b66ad163ea9:/usr/local/src# javac Hello.java
root@0b66ad163ea9:/usr/local/src#
root@0b66ad163ea9:/usr/local/src#
root@0b66ad163ea9:/usr/local/src# java Hello
No. 1 : Hello Java
No. 2 : Hello Java
No. 3 : Hello Java
No. 4 : Hello Java
No. 5 : Hello Java
No. 6 : Hello Java
No. 7 : Hello Java
No. 8 : Hello Java
No. 9 : Hello Java
No. 10 : Hello Java
No. 11 : Hello Java
No. 12 : Hello Java
No. 13 : Hello Java
No. 14 : Hello Java
No. 15 : Hello Java
No. 16 : Hello Java
No. 17 : Hello Java
No. 18 : Hello Java
No. 19 : Hello Java
No. 20 : Hello Java
root@0b66ad163ea9:/usr/local/src# _
```