



Training & Certification

Lab 5. Multi-stage Docker Build (Solution)

Following is the code which demonstrates a Dockerfile with two stages:

file: Dockerfile.multistage.v1

```
FROM schoolofdevops/maven:spring AS build
WORKDIR /app
COPY . .
RUN mvn spring-javaformat:apply && \

mvn package -DskipTests

FROM openjdk:8-alpine AS run
COPY --from=build /app/target/spring-petclinic-2.3.1.BUILD-SNAPSHOT.jar
/run/petclinic.jar
EXPOSE 8080
CMD java -jar /run/petclinic.jar
```

To build an image with this new Dockerfile, run the following sequence of commands:

```
docker build -f Dockerfile.multistage.v1 -t <docker hub id>/petclinic:v3 .
docker image ls
```

Adding an Opt-in Test Stage

You can use a multi-stage approach to support optional features. The following code demonstrates an opt-in stage to run unit tests:

file: Dockerfile.multistage.v2

```
FROM schoolofdevops/maven:spring AS build
WORKDIR /app
COPY . .
RUN mvn spring-javaformat:apply && \
```

```
mvn package -DskipTests
```

```
FROM build AS test
```

```
CMD mvn clean test
```

```
FROM openjdk:8-alpine AS run
```

```
COPY --from=build /app/target/spring-petclinic-2.3.1.BUILD-SNAPSHOT.jar  
/run/petclinic.jar
```

```
EXPOSE 8080
```

```
CMD java -jar /run/petclinic.jar
```

Now, build using the opt-in stage:

```
docker build -f Dockerfile.multistage.v2 --target test -t <docker hub  
id>/petclinic:test .
```

Please note that the opt-in stage is executed only when you build the image with **--target test** option. In other cases, it is omitted.