



KodeFlux Originals

studio.kodecloud.com/labs/docker/docker_images

Back to Domain Docker Images Understand Docker Images, building custom images, and image management. Intermediate 90-120 min

3 / 15 Next →

We just pulled a new image. What is the tag on the newly pulled `nginx` image?

11cd0b38bc3c

1.14

perl

1.14-alpine

1.10.1

```
~ → docker images nginx
REPOSITORY TAG IMAGE ID CREATED SIZE
nginx alpine c7b4f26a7d93 11 months ago 43.2MB
nginx latest 39286ab8a5e1 11 months ago 188MB
nginx 1.14-alpine 8a2fb25a19f5 6 years ago 16MB
~ →
```

KodeFlux Originals

studio.kodecloud.com/labs/docker/docker_images

Back to Domain Docker Images Understand Docker Images, building custom images, and image management. Intermediate 90-120 min

4 / 15 Next →

We just downloaded the code of an application. What is the base image used in the Dockerfile?

Inspect the Dockerfile in the `webapp-color` directory.

python:3.6

centos:7

ubuntu:latest

rhel:3.6

python

```
~ → cd webapp-color
~/webapp-color via ↵ ls
Dockerfile app.py requirements.txt templates
~/webapp-color via ↵ cat Dockerfile
FROM python:3.6

RUN pip install flask

COPY . /opt/
EXPOSE 8080

WORKDIR /opt

ENTRYPOINT ["python", "app.py"]
~/webapp-color via ↵
~/webapp-color via ↵
```

studio.kodekloud.com/labs/docker/docker_images

Back to Domain Docker Images Understand Docker Images, building custom images, and image management. Intermediate 90-120 min

5 / 15 Next →

To what location within the container is the application code copied to during a Docker build?

Inspect the Dockerfile in the `webapp-color` directory.

`/opt`

`/root`

`/tmp`

`/var`

```
~/webapp-color via ➤ cat Dockerfile
FROM python:3.6

RUN pip install flask

COPY . /opt/

EXPOSE 8080

WORKDIR /opt

ENTRYPOINT ["python", "app.py"]

~/webapp-color via ➤
```

studio.kodekloud.com/labs/docker/docker_images

Back to Domain Docker Images Understand Docker Images, building custom images, and image management. Intermediate 90-120 min

6 / 15 Next →

When a container is created using the image built with this Dockerfile, what is the command used to `RUN` the application inside it?

Inspect the Dockerfile in the `webapp-color` directory.

`docker run app`

`app.exe`

`python app.py`

`pip install flask`

```
~/webapp-color via ➤ ls
Dockerfile  app.py  requirements.txt  templates

~/webapp-color via ➤ cat Dockerfile
FROM python:3.6

RUN pip install flask

COPY . /opt/

EXPOSE 8080

WORKDIR /opt

ENTRYPOINT ["python", "app.py"]

~/webapp-color via ➤
```

7 / 15

What `port` is the web application run within the container?

Inspect the Dockerfile in the `webapp-color` directory.

0.0.0.0

80

8080

5000

```
~/webapp-color via ➤ ls
Dockerfile      app.py          requirements.txt templates

~/webapp-color via ➤ cat Dockerfile
FROM python:3.6

RUN pip install flask

COPY . /opt/

EXPOSE 8080

WORKDIR /opt

ENTRYPOINT ["python", "app.py"]

~/webapp-color via ➤
```

8 / 15

Build a docker image using the Dockerfile and name it `webapp-color`. No tag to be specified.

Check

Next →

Image Name: webapp-color

```
~/webapp-color via ➤ docker build -t webapp-color .
DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
BuildKit is currently disabled; enable it by removing the DOCKER_BUILDKIT=
0
environment-variable.

Sending build context to Docker daemon  8.704kB
Step 1/6 : FROM python:3.6
3.6: Pulling from library/python
0e29546d541c: Pull complete
9b829c73b52b: Pull complete
cb5b7ae36172: Pull complete
6494e4811622: Pull complete
6f9f74896dfa: Pull complete
5a3b1213efc5: Pull complete
9fd9fde56234: Pull complete
404f62844bac: Pull complete
c4f42be2be53: Pull complete
Digest: sha256:f8652afaf88c25f0d22354d547d892591867aa4026a7fa9a6819df9f308af6fc
Status: Downloaded newer image for python:3.6
--> 54260638d07c
Step 2/6 : RUN pip install flask
--> Running in 18da6a8b047c
Collecting flask
  Downloading Flask-2.0.3-py3-none-any.whl (95 kB)
Collecting Jinja2>=3.0
  Downloading Jinja2-3.0.3-py3-none-any.whl (133 kB)
Collecting click>=7.1.2
  Downloading click-8.0.4-py3-none-any.whl (97 kB)
Collecting Werkzeug>=2.0
  Downloading Werkzeug-2.0.3-py3-none-any.whl (289 kB)
```

studio.kodekloud.com/labs/docker/docker_images

Docker Images

Understand Docker images, building custom images, and image management. Intermediate 90-120 min

9 / 15

Run an instance of the image `webapp-color` and publish port `8080` on the container to `8282` on the host.

☒

☒

- ✓ Container with image 'webapp-color'
- ✓ Container Port: 8080
- ✓ Host Port: 8282

```
~/webapp-color via  docker run -d --name webb -p 8282:8080 webapp-color c5147f0a186e2f24cd3b2593f29b8fcd29a5c0eea283785665f743cd477250ca
```

```
~/webapp-color via  
```

studio.kodekloud.com/labs/docker/docker_images

Docker Images

Understand Docker images, building custom images, and image management. Intermediate 90-120 min

Task AI Assistant (Beta) 39:31

10 / 15

Access the application by clicking on the tab named `HOST:8282` above your terminal.

After you are done, you may stop the running container by hitting `CTRL + C` if you wish to.

Terminal 1

```
~/webapp-color via  docker run -d --name webb -p 8282:8080 webapp-color c5147f0a186e2f24cd3b2593f29b8fcd29a5c0eea283785665f743cd477250ca
```

```
~/webapp-color via  docker stop webb
```

```
~/webapp-color via  
```

studio.kodekloud.com/labs/docker/docker_images

Back to Domain Docker Images Understand Docker images, building custom images, and image management. Intermediate 90-120 min

11 / 15

What is the base Operating System used by the `python:3.6` image?
If required, run an instance of the image to figure it out.

ubuntu
centos
rhel
debian

Next →

```
~/webapp-color via ➤ docker run -it python:3.6 cat /etc/os-release
PRETTY_NAME="Debian GNU/Linux 11 (bullseye)"
NAME="Debian GNU/Linux"
VERSION_ID="11"
VERSION="11 (bullseye)"
VERSION_CODENAME=bullseye
ID=debian
HOME_URL="https://www.debian.org/"
SUPPORT_URL="https://www.debian.org/support"
BUG_REPORT_URL="https://bugs.debian.org/"

~/webapp-color via ➤
```

studio.kodekloud.com/labs/docker/docker_images

Back to Domain Docker Images Understand Docker images, building custom images, and image management. Intermediate 90-120 min

12 / 15

What is the approximate size of the `webapp-color` image?

2GB
80MB
920MB
500MB

Next →

```
~/webapp-color via ➤ docker images webapp-color
REPOSITORY TAG IMAGE ID CREATED SIZE
webapp-color latest 54b8614d78d9 11 minutes ago 913MB

~/webapp-color via ➤
```

KodeFox Originals X +

studio.kodecloud.com/labs/docker/docker_images

Back to Domain Docker Images Understand Docker Images, building custom images, and image management. Intermediate 90-120 min

13 / 15

Next →

That's really BIG for a Docker Image. Docker images are supposed to be small and light weight. Let us try to trim it down.

Ok

```
comment instead: https://pip.pypa.io/warnings/venv
WARNING: You are using pip version 21.2.4; however, version 21.3.1 is available.
You should consider upgrading via the '/usr/local/bin/python -m pip install --upgrade pip' command.
--> Removed intermediate container 752a9e4c94a4
--> 286719a43ed1
Step 3/6 : COPY . /opt/
--> f37bd0bc9064
Step 4/6 : EXPOSE 8080
--> Running in 83113de92b83
--> Removed intermediate container 83113de92b83
--> e04856553010
Step 5/6 : WORKDIR /opt
--> Running in dc7b3201aded
--> Removed intermediate container dc7b3201aded
--> 8fee856b7277
Step 6/6 : ENTRYPOINT ["python", "app.py"]
--> Running in 73106606c7ee
--> Removed intermediate container 73106606c7ee
--> 04cf7b22498c
Successfully built 04cf7b22498c
Successfully tagged webapp-color-slim:latest

~/webapp-color via ➡ docker images webapp-color
REPOSITORY TAG IMAGE ID CREATED SIZE
webapp-color latest 54b8614d78d9 17 minutes ago 913MB

~/webapp-color via ➡ docker images webapp-color-slim
REPOSITORY TAG IMAGE ID CREATED SIZE
webapp-color-slim latest 04cf7b22498c 59 seconds ago 130MB

~/webapp-color via ➡
```

KodeFox Originals X +

studio.kodecloud.com/labs/docker/docker_images

Back to Domain Docker Images Understand Docker Images, building custom images, and image management. Intermediate 90-120 min

14 / 15

Next →

Build a new smaller docker image by modifying the same Dockerfile and name it `webapp-color` and tag it `lite`.

Hint: Find a smaller base image for `python:3.6`. Make sure the final image is less than `150MB`.

Check

Next →

- ✓ Name: `webapp-color:lite`
- ✓ Image size less than 150MB.

```
~/webapp-color via ➡ docker build -t webapp-color:lite .
DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
BuildKit is currently disabled; enable it by removing the DOCKER_BUILDKIT=
0
environment-variable.

Sending build context to Docker daemon 8.704kB
Step 1/6 : FROM python:3.6-slim
--> c1e40b69532f
Step 2/6 : RUN pip install flask
--> Using cache
--> 286719a43ed1
Step 3/6 : COPY . /opt/
--> Using cache
--> f37bd0bc9064
Step 4/6 : EXPOSE 8080
--> Using cache
--> e04856553010
Step 5/6 : WORKDIR /opt
--> Using cache
--> 8fee856b7277
Step 6/6 : ENTRYPOINT ["python", "app.py"]
--> Using cache
--> 04cf7b22498c
Successfully built 04cf7b22498c
Successfully tagged webapp-color:lite

~/webapp-color via ➡
```

KodeFlix Originals

studio.kodekloud.com/labs/docker/docker_images

Back to Domain Docker Images Understand Docker Images, building custom images, and image management. Intermediate 90-120 min

15 / 15

Next →

Run an instance of the new image `webapp-color:lite` and publish port `8080` on the container to `8383` on the host.

Check

Next →

- ✓ Container with image 'webapp-color:lite'
- ✓ Container to publish port 8080 to 8383

```
~/webapp-color via ➤ docker run -d -p 8383:8080 webapp-color:lite
04b04de07d0b460d5e9ae0bd8eef36f560aac748f2a04dc8039b213d2fe7a528

~/webapp-color via ➤
```

KodeFlix Originals

studio.kodekloud.com/labs/docker/docker_images

Back to Domain Docker Images Understand Docker Images, building custom images, and image management. Intermediate 90-120 min

CONGRATULATIONS!!!!
You have successfully completed the challenge.

End lab session

```
~/webapp-color via ➤ docker run -d -p 8383:8080 webapp-color:lite
04b04de07d0b460d5e9ae0bd8eef36f560aac748f2a04dc8039b213d2fe7a528

~/webapp-color via ➤
```