

How Does One Install Pip in a Docker Container using a Dockerfile?

[Go Programming](#)[Server Side Programming](#)[Programming](#)

In this tutorial, we are going to learn how we can install pip in a Docker container using a Dockerfile. This tutorial will cover the creation of the Dockerfile and then we will see the docker command to build a docker image from the Dockerfile, and in last we will see the docker command to run a container based on that image and see whether PIP is installed properly or not.

Prerequisites

There are some prerequisites to creating and building the Dockerfile as mentioned below.

- The stable version of Docker should be installed.
- Create a file in any folder named **Dockerfile**.

Creating a Dockerfile

Dockerfile is a collection of commands or we can say a recipe of an image in the form of a text document. Now step by step we will create a Dockerfile that will contain the commands to install PIP.

Step 1

In the very first line of the Dockerfile, we will mention the base OS image our image will create for that we use FROM command in Dockerfile.

```
1. FROM ubuntu:latest
```

Using the above command we have mentioned that we want to use ubuntu as the base image with the latest tag.

Step 2

In the next step, as we are using the ubuntu as base OS then we need to run **apt-get update** command to update all the ubuntu packages by using the RUN command in the Dockerfile that will run the command in front of it. After adding this our Dockerfile will look like the below.

```
1. FROM ubuntu:latest  
2. RUN apt-get update
```

Step 3

In the last step of the Dockerfile we will install the pip package using the same RUN command in the Dockerfile mentioned above. The name of the PIP package is **python3-pip** and the command we need to run as per Ubuntu OS is **apt-get install -y python3-pip**. After adding this the complete Dockerfile will look like this.

```
1. FROM ubuntu:latest  
2. RUN apt-get update  
3. RUN apt-get install -y python3-pip
```

Now we are done with the Dockerfile and proceed with building an image from it.

Explore our **latest online courses** and learn new skills at your own pace. Enroll and become a certified expert to boost your career.

Building Image from Dockerfile

To build the image we use the **docker build** command to learn about it you can run **docker build -help**.

Syntax

```
docker build [OPTIONS] PATH | URL | -
```

Command

In our case, we will use the **-t** flag to add a tag to the image and we will run the command in the same directory where Dockerfile is present so the path will be **.** in our command stated below.

```
docker build -t pip .
```

Once the above command is completed successfully by running the below command you can see your image with the tag you mentioned earlier.

```
% docker image ls
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
pip	latest	e03f400cf705	5 hours ago	467MB

Running a Container from the Image

As our image has been built successfully now we will run a container from the above image and for that, we will run **docker container run** command.

Syntax

```
docker container run [OPTIONS] IMAGE [COMMAND] [ARG...]
```

Command

In options, we will use **-it** flags that will run the container in an interactive way and we will be able to run the command inside the container in place of IMAGE. We will use the Image ID as shown in **docker image ls** output.

```
docker container run -it e03f4
```

Now we have entered into the container we will run the pip command to check whether it is installed properly or not. For that, we will run the **pip --version** command.

```
pip --version
pip 22.0.2 from /usr/lib/python3/dist-packages/pip (python 3.10)
```

Conclusion

This is the way to install pip in a Docker container using a Dockerfile. Above image, you can push it to the docker hub and you and others can use this image as the base image in other Dockerfile. To learn more about Docker you can follow these [tutorials](#).