

#013 Docker Containers

Introduction

this is part 13 from the journey it's a long journey(360 day) so go please check previous parts , and if you need to walk in the journey with me please make sure to follow because I may post more than once in 1 Day but surely I will post daily at least one 😊.

And I will cover lot of tools as we move on.

Download app_013

```
(base) in ~
> cd Documents/DevOpsJourney/app_013/
(base) (master)in ~/Documents/DevOpsJourney/app_013
> 1
Dockerfile app.py requirements.txt
(base) (master)in ~/Documents/DevOpsJourney/app_013
>
```

if you follow [part 9](#)

```
cd location/DevOpsJourney/
git pull
cd app_013
```

replace location with where you clone the DevOpsJourney repo

if your new go to [part 9](#) and do same steps will download old lecture files and new one.

Build image

```
docker image build -t app_013 .
```

```

(base) (master)in ~/Documents/DevOpsJourney/app_013
> docker image build -t app_013 .
Sending build context to Docker daemon 3.584kB
Step 1/8 : FROM python:3.9-rc-alpine
--> a206803e6cb1
Step 2/8 : RUN mkdir /app
--> Using cache
--> cccdc7a8e00
Step 3/8 : WORKDIR /app
--> Using cache
--> 1c43ee4da9d4
Step 4/8 : COPY requirements.txt requirements.txt
--> 30fcc80c614c
Step 5/8 : RUN pip install -r requirements.txt
--> Running in 7f7269dfe070
Removing intermediate container 7f7269dfe070
--> 4cb299d53b91
Step 6/8 : COPY . .
--> 9fd865fd7731
Step 7/8 : LABEL maintainer="Omar ElKhatib"
--> Running in 383292a29b6d
Removing intermediate container 383292a29b6d
--> f807b479db51
Step 8/8 : CMD python app.py
--> Running in b2daf33600ad
Removing intermediate container b2daf33600ad
--> db29148a2bf0
Successfully built db29148a2bf0
Successfully tagged app_013:latest
(base) (master)in ~/Documents/DevOpsJourney/app_013
>

```

Containers

first I like to mention that there is no need to docker container you can just put container , docker default option is docker container

```

(base) (master)in ~/Documents/DevOpsJourney/app_013
> docker container ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS              NAMES
(base) (master)in ~/Documents/DevOpsJourney/app_013
> docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS              NAMES
(base) (master)in ~/Documents/DevOpsJourney/app_013
>

```

Let's start

let's run our app

```
Docker containers
(base) (master)in ~/Documents/DevOpsJourney/app_013
> docker run -it app_013
Hello Dev.to Folks! this is Omar
give me an input : omar

/bin/bash 144x55
(base) (master)in ~/Documents/DevOpsJourney/app_013
> docker image ps

Usage: docker image COMMAND

Manage images

Commands:
  build      Build an image from a Dockerfile
  history    Show the history of an image
  import     Import the contents from a tarball to create a filesystem image
  inspect    Display detailed information on one or more images
  load       Load an image from a tar archive or STDIN
  ls         List images
  prune      Remove unused images
  pull       Pull an image or a repository from a registry
  push       Push an image or a repository to a registry
  rm         Remove one or more images
  save       Save one or more images to a tar archive (streamed to STDOUT by default)
  tag        Create a tag TARGET_IMAGE that refers to SOURCE_IMAGE

Run 'docker image COMMAND --help' for more information on a command.
(base) (master)in ~/Documents/DevOpsJourney/app_013
> clear
(base) (master)in ~/Documents/DevOpsJourney/app_013
> docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS   NAMES
3ebe88ec804f   app_013   "/bin/sh -c 'python _"   24 seconds ago Up 22 seconds          silly_mclaren
(base) (master)in ~/Documents/DevOpsJourney/app_013
>
```

```
docker run -it app_013
```

-it for interactive with docker

i am using terminator on linux , he give me ability to split the terminal , in case you have a normal terminal open new tab , and run

```
docker ps
```

you can see our container is running with name silly_mclaren and it's default name given by docker to us.

```
Docker containers
(base) (master)in ~/Documents/DevOpsJourney/app_013
> docker run -it app_013
Hello Dev.to Folks! this is Omar
give me an input : omar
omar
(base) (master)in ~/Documents/DevOpsJourney/app_013
>

/bin/bash 144x55
(base) (master)in ~/Documents/DevOpsJourney/app_013
> docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS   NAMES
3ebe88ec804f   app_013   "/bin/sh -c 'python _"   24 seconds ago Up 22 seconds          silly_mclaren
(base) (master)in ~/Documents/DevOpsJourney/app_013
>
```

when we enter a input the program the container will shutdown.

to able to see shutdown containers we need

```
(base) (master)in ~/Documents/DevOpsJourney/app_013
> docker run -it app_013
Hello Dev.to Folks! this is Omar
give me an input : omar
omar
(base) (master)in ~/Documents/DevOpsJourney/app_013
> docker ps -a
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS   NAMES
3ebe88ec804f   app_013   "/bin/sh -c 'python ..." 3 minutes ago Exited (0) About a minute ago          silly_mclaren
aren
```

```
docker ps -a
```

as we see it's status is exited

to automatically delete the container after it ends do

```
(base) (master)in ~/Documents/DevOpsJourney/app_013
> docker run --rm -it app_013
Hello Dev.to Folks! this is Omar
give me an input : omar
omar
(base) (master)in ~/Documents/DevOpsJourney/app_013
> docker ps -a
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS   NAMES
```

```
docker run --rm -it app_013
docker ps -a
```

we can see that he auto remove our container

running in background

to run in background we add -d

```
(base) (master)in ~/Documents/DevOpsJourney/app_013
> docker run -d -it app_013
23b85060fe31050dc8e7ccb22d4794d8d7b215ecfac7b133a7b16d5f60f2abdb
(base) (master)in ~/Documents/DevOpsJourney/app_013
> docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS              NAMES
23b85060fe31        app_013            "/bin/sh -c 'python ..." 5 seconds ago       Up 3 seconds              amazing_wu
(base) (master)in ~/Documents/DevOpsJourney/app_013
>
```

```
docker run -d -it app_013
docker ps
```

we can see it's running but in background with name amazing_wu

```
(base) (master)in ~/Documents/DevOpsJourney/app_013
> docker logs amazing_wu
Hello Dev.to Folks! this is Omar
(base) (master)in ~/Documents/DevOpsJourney/app_013
> docker logs -f amazing_wu
Hello Dev.to Folks! this is Omar
omar
asd
asdfsdfdjshf
asdasdasdasd
```

to see our app output (in case of -d) docker logs name

```
docker logs amazing_wu
```

to interact with the container that is running in background

```
docker logs -f amazing_wu
```

we use -f to do this

Run another one

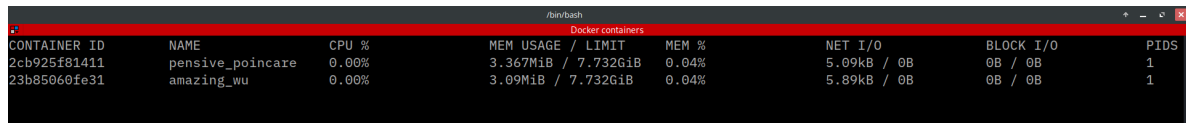
let's run another container in background also

```
Docker containers
(base) (master)in ~/Documents/DevOpsJourney/app_013
> docker run -d -it app_013
2cb925f8141158cb7866bc237558c5426d818ddaa84270060692702a68307c19
(base) (master)in ~/Documents/DevOpsJourney/app_013
> docker stats
```

```
docker run -d -it app_013
```

to see the resources used by our containers run

```
docker logs
```



The screenshot shows a terminal window with a red title bar labeled 'docker containers'. Inside, a table displays the status of two containers: 'pensive_poincare' and 'amazing_wu'. The table has columns for Container ID, Name, CPU %, Memory Usage / Limit, Memory %, Network I/O, Block I/O, and PIDs.

CONTAINER ID	NAME	CPU %	MEM USAGE / LIMIT	MEM %	NET I/O	BLOCK I/O	PIDS
2cb925f81411	pensive_poincare	0.00%	3.367MiB / 7.732GiB	0.04%	5.09kB / 0B	0B / 0B	1
23b85060fe31	amazing_wu	0.00%	3.09MiB / 7.732GiB	0.04%	5.89kB / 0B	0B / 0B	1

to stop a container

```
docker stop amazing_wu
```

more about containers

you can do a lot with containers , example we can give a name , not a random one , this is your challenge

```
docker run --name app_013 -it app_013
```

more about containers run

```
docker container --help
```

or see the [docs!](#)