

# Simple Container Creation

Sanjana Sudarshan  
Jetstream - Indiana University

PEARC20 (Virtual Edition) - July 26-30, 2020



# Docker

```
$ docker --version  
Docker version 1.13.1, build 64e9980/1.13.1
```

```
$ docker run hello-world  
Unable to find image 'hello-world:latest' locally  
Trying to pull repository docker.io/library/hello-world ...  
latest: Pulling from docker.io/library/hello-world  
0e03bdcc26d7: Pull complete  
Digest:  
sha256:6a65f928fb91fcfbc963f7aa6d57c8eeb426ad9a20c7ee045538ef34847f44f1  
Status: Downloaded newer image for docker.io/hello-world:latest
```

.....



# Running a container from prebuilt image

Run a container from Alpine Linux

```
$ docker run alpine ls -l
Unable to find image 'alpine:latest' locally
Trying to pull repository docker.io/library/alpine ...
latest: Pulling from docker.io/library/alpine
df20fa9351a1: Pull complete
Digest: sha256:185518070891758909c9f839cf4ca393ee977ac378609f700f60a771a2dfe321
Status: Downloaded newer image for docker.io/alpine:latest
total 8
drwxr-xr-x    2 root    root           4096 May 29 14:20 bin
drwxr-xr-x    5 root    root           340 Jun  2 15:11 dev
drwxr-xr-x    1 root    root            66 Jun  2 15:11 etc
.....
```



# Running a container from prebuilt image

```
$ docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
docker.io/alpine	latest	a24bb4013296	3 days ago	5.57 MB
docker.io/hello-world	latest	bf756fb1ae65	5 months ago	13.3 kB

```
$ docker run alpine echo "Hello world"
```

```
Hello world
```

```
$ docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS
--------------	-------	---------	---------	--------

```
$ docker ps --all
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS
b61702e8cf09	alpine	"echo 'Hello world'"	32 seconds ago	Exited (0) 31 seconds ago



# Build a Docker Image

## Odd / Even

- app.py
- Dockerfile

```
$ cd ~ && mkdir simple-script && cd simple-script
```

Create a file named `app.py` with the following content

```
# Python program to check if the input number is odd or even
```

```
num = int(input("Enter a number: "))
```

```
if (num % 2) == 0:
```

```
    print("{} is Even".format(num))
```

```
else:
```

```
    print("{} is Odd".format(num))
```

# Build a Docker Image

Create a file called Dockerfile in the simple-script directory

```
# our base image
FROM alpine:3.9

# install python and pip
RUN apk add --update py3-pip

# copy files required for the app to run
COPY app.py /usr/src/app/

# run the application
CMD ["python3" , "/usr/src/app/app.py"]
```



```
$ docker build -t $YOUR_DOCKERHUB_USERNAME/simple-script .
```

```
sudo docker build -t sanjanasudarshan/simple-script .
```

```
Sending build context to Docker daemon 3.072kB
```

```
Step 1/4 : FROM alpine:3.9
```

```
3.9: Pulling from library/alpine
```

```
.
```

```
Step 2/4 : RUN apk add --update py3-pip
```

```
---> Running in ead201b4a5a9
```

```
fetch http://dl-cdn.alpinelinux.org/alpine/v3.9/main/x86_64/APKINDEX.tar.gz
```

```
fetch http://dl-cdn.alpinelinux.org/alpine/v3.9/community/x86_64/APKINDEX.tar.gz
```

```
(1/11) Installing libbz2 (1.0.6-r7)
```

```
.
```

```
(11/11) Installing python3 (3.6.9-r2)
```

```
Step 3/4 : COPY my_script.py /usr/src/app/
```

```
---> 3c12a3940c4d
```

```
Step 4/4 : CMD ["python3" "/usr/src/app/my_script.py"]
```

```
---> Running in f84bfd09474a
```

```
Removing intermediate container f84bfd09474a
```

```
---> 514dbb79d853
```

```
Successfully built 514dbb79d853
```

```
Successfully tagged sanjanasudarshan/simple-script:latest
```

```
$ docker run -i $YOUR_DOCKERHUB_USERNAME/simple-script
```

```
Enter a number: 5
```

```
5 is Odd
```



# Build a Docker Image

## Dice Roll

- diceroll.py
- Dockerfile

```
$ cd ~ && mkdir dice-script && cd dice-script
```

Create a file named `diceroll.py` with the following content

```
import random
min = 1
max = 6

roll_again = "yes"

while roll_again == "yes" or roll_again == "y":
    print "Rolling the dices..."
    print "The values are...."
    print random.randint(min, max)
    print random.randint(min, max)

    roll_again = raw_input("Roll the dices again?")
```

- Dockerfile

```
# our base image
FROM alpine:3.9

# install python and pip
RUN apk add --update py3-pip

# copy files required for the app to run
COPY diceroll.py /usr/src/app/

# run the application
CMD ["python3" , "/usr/src/app/diceroll.py"]
```



# Build a Docker Image

## Jupyter Notebook

```
$ docker search jupyter
```

INDEX	NAME	DESCRIPTION
docker.io	docker.io/jupyter/datascience-notebook	Jupyter Notebook Data Science Stack from h...
docker.io	docker.io/jupyter/all-spark-notebook	Jupyter Notebook Python, Scala, R, Spark, ...
docker.io	docker.io/jupyterhub/jupyterhub	JupyterHub: multi-user Jupyter notebook se...
docker.io	docker.io/jupyter/scipy-notebook	Jupyter Notebook Scientific Python Stack f...

```
$ cd ~ && mkdir mynotebook && cd mynotebook
```

- model.py

```
def introduce(name):  
    return 'Hello ' + name
```

- Dockerfile

```
# our base image  
FROM jupyter/minimal-notebook  
  
# copy files required for the model to work  
COPY model.py /home/jovyan/work/  
  
# tell the port number the container should expose  
EXPOSE 8888
```



```
$ docker build -t $YOUR_DOCKERHUB_USERNAME /mynotebook .
```

```
Step 1/3 : FROM jupyter/minimal-notebook
```

```
Trying to pull repository docker.io/jupyter/minimal-notebook ...
```

```
latest: Pulling from docker.io/jupyter/minimal-notebook
```

```
.  
. .
```

```
Status: Downloaded newer image for docker.io/jupyter/minimal-notebook:latest
```

```
---> b61382e30c1d
```

```
Step 2/3 : COPY model.py /home/jovyan/work/
```

```
---> 961a469fb881
```

```
Removing intermediate container 7a2ba5ef7f8c
```

```
Step 3/3 : EXPOSE 8888
```

```
---> Running in a4cd0615b004
```

```
---> f1c18e7b1fac
```

```
Removing intermediate container a4cd0615b004
```

```
Successfully built f1c18e7b1fac
```



```
$ docker images
```

REPOSITORY	TAG	IMAGE ID	..
sanjanasudarshan/mynotebook	latest	f1c18e7b1fac	..
sanjanasudarshan/simple-script	latest	ea8a273af483	..

```
$ docker run -p 8888:8888 $YOUR_DOCKERHUB_USERNAME/mynotebook
```

```
docker run -p 8888:8888 sanjanasudarshan/mynotebook
```

```
Executing the command: jupyter notebook
```

```
[I 16:22:57.132 NotebookApp] Writing notebook server cookie secret to /home/jovyan/. . .
```

```
[I 16:22:57.961 NotebookApp] JupyterLab extension loaded from /opt/conda/lib/python3.7/.
```

```
.
```

```
.
```

```
.
```

```
To access the notebook, open this file in a browser:
```

```
file:///home/jovyan/.local/share/jupyter/runtime/nbserver-7-open.html
```

```
Or copy and paste one of these URLs:
```

```
http://577b35de6162:8888/?token=575733d74407ad1aefc7bdae50dba08aa97811675234bfb8
```

```
or http://127.0.0.1:8888/?token=575733d74407ad1aefc7bdae50dba08aa97811675234bfb8
```

---

# Dockerizing Samtools

- Create a Dockerfile for Samtools and build it using docker build
- Tag it as “my\_samtools”

```
sudo docker tag sanjanasudarshan/samtools:latest  
sanjanasudarshan/my_samtools
```

## Dockerfile

```
# our base image  
FROM ubuntu  
  
# install samtools  
RUN apt-get update  
RUN apt-get install -y wget  
RUN apt-get install -y apt-utils  
RUN apt-get install -y gcc  
RUN apt-get install -y make  
RUN apt-get install -y libbz2-dev  
RUN apt-get install -y zlib1g-dev  
RUN apt-get install -y libncurses5-dev  
RUN apt-get install -y libncursesw5-dev  
RUN apt-get install -y liblzma-dev  
RUN apt-get install -y libcurl4-openssl-dev:amd64
```



```
# Pulling HTSLIB from its repository, unpacking the archive and installing  
RUN wget https://github.com/samtools/htslib/releases/download/1.9/htslib-1.9.tar.bz2 \  
&& tar -vxjf htslib-1.9.tar.bz2 \  
&& cd htslib-1.9 \  
&& make \  
&& make install
```

```
# Pulling SAMTools from its repository, unpacking the archive and installing  
RUN wget https://github.com/samtools/samtools/releases/download/1.10/samtools-  
1.10.tar.bz2 \  
&& tar jxf samtools-1.10.tar.bz2 \  
&& cd samtools-1.10 \  
&& make \  
&& make install
```

```
# Run the main script  
CMD ["samtools"]
```

