<epam>

Containerization. Docker. Lection 2.

Webapps with Docker

Create your image

Now that you have a better understanding of images, it's time to create your own. Our goal here is to create an image that sandboxes a small Flask application.

The goal of this exercise is to create a Docker image which will run a Flask app.

We'll do this by first pulling together the components for a random cat picture generator built with Python Flask, then dockerizing it by writing a Dockerfile. Finally, we'll build the image, and then run it.

Create a Python Flask app that displays random cat pix

- Write a Dockerfile
- Build the image
- Run your image
- Push your image to the Docker registry

Webapps with Docker

Create a Python Flask app that displays random cat pix

For the purposes of this workshop, we've created a fun little Python Flask app that displays a random cat .gif every time it is loaded - because, you know, who doesn't like cats?

Start by creating a directory called ***flask-app*** where we'll create the following files:

```
app.py
requirements.txt
templates/index.html
Dockerfile
```

Make sure to **cd** *flask-app* before you start creating the files, because you don't want to start adding a whole bunch of other random files to your image.

Webapps with Docker

Prepairing all needed files

```
student@localhost//$ cd ~
student@localhost~$ mkdir flask-app
student@localhost~$ cd flask-app/
student@localhost~/flask-app$ nano app.py
student@localhost~/flask-app$ nano requirements.txt
student@localhost~/flask-app$ mkdir templates
student@localhost~/flask-app$ cd templates/
student@localhost~/flask-app/templates$ nano index.html
student@localhost~/flask-app/templates$ nano Dockerfile
student@localhost~/flask-app/templates$ cd ..
student@localhost~/flask-app$ pwd
/home/student/flask-app
```

mv Dockerfile ../

Webapps with Docker. Content of app.py

```
from flask import Flask, render_template
import random
app = Flask(__name__)

# list of cat images
images = [
    "http://ak-hdl.buzzfed.com/static/2013-10/enhanced/webdr05/15/9/anigif_enhanced-buzz-26388-1381844103-11.gif",
    "http://ak-hdl.buzzfed.com/static/2013-10/enhanced/webdr01/15/9/anigif_enhanced-buzz-31540-1381844535-8.gif",
    "http://ak-hdl.buzzfed.com/static/2013-10/enhanced/webdr05/15/9/anigif_enhanced-buzz-26390-1381844163-18.gif",
    "http://ak-hdl.buzzfed.com/static/2013-10/enhanced/webdr06/15/10/anigif_enhanced-buzz-1376-1381846217-0.gif",
    "http://ak-hdl.buzzfed.com/static/2013-10/enhanced/webdr03/15/9/anigif_enhanced-buzz-3391-1381844336-26.gif",
    "http://ak-hdl.buzzfed.com/static/2013-10/enhanced/webdr06/15/10/anigif_enhanced-buzz-29111-1381845968-0.gif",
    "http://ak-hdl.buzzfed.com/static/2013-10/enhanced/webdr03/15/9/anigif_enhanced-buzz-3409-1381844582-13.gif",
    "http://ak-hdl.buzzfed.com/static/2013-10/enhanced/webdr02/15/9/anigif_enhanced-buzz-19667-1381844937-10.gif",
    "http://ak-hdl.buzzfed.com/static/2013-10/enhanced/webdr05/15/9/anigif_enhanced-buzz-26358-1381845043-13.gif",
    "http://ak-hdl.buzzfed.com/static/2013-10/enhanced/webdr06/15/9/anigif_enhanced-buzz-18774-1381844645-6.gif",
    "http://ak-hdl.buzzfed.com/static/2013-10/enhanced/webdr06/15/9/anigif_enhanced-buzz-25158-1381844793-0.gif",
    "http://ak-hdl.buzzfed.com/static/2013-10/enhanced/webdr03/15/10/anigif_enhanced-buzz-11980-1381846269-1.gif"
]
@app.route('/')
def index():
    url = random.choice(images)
    return render_template('index.html', url=url)

if __name__== "__main__":
    app.run(host="0.0.0.0")
```

Webapps with Docker. Content of Dockerfile

```
# this is my base image
FROM alpine:3.5

# Install python and pip
RUN apk add --update py2-pip

# install Python modules needed by the Python app
COPY requirements.txt /usr/src/app/
RUN pip install --no-cache-dir -r /usr/src/app/requirements.txt

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

# tell the port number the container should expose
EXPOSE 5000

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

Webapps with Docker. Content of index.html

```
<html>
  <head>
    <style type="text/css">
      body {
        background: black;
        color: green;
      }
      div.container {
        max-width: 500px;
        margin: 100px auto;
        border: 20px solid white;
        padding: 10px;
        text-align: center;
      }
      h4 {
        text-transform: uppercase;
      }
    </style>
  </head>
  <body>
    <div class="container">
      <h4>Welcome to Python App running on K8S cluster in EKS!!</h4>
      <h4>Automating using Jenkins, Kubernetes, Docker, GitHub and Pipeline!!!!!</h4>
      <br>
       <h4>Anything is Possible!!!!!</h4>
    </div>
  </body>
</html>
```

Webapps with Docker

Content of requirements.txt

```
Flask==0.10.1
```

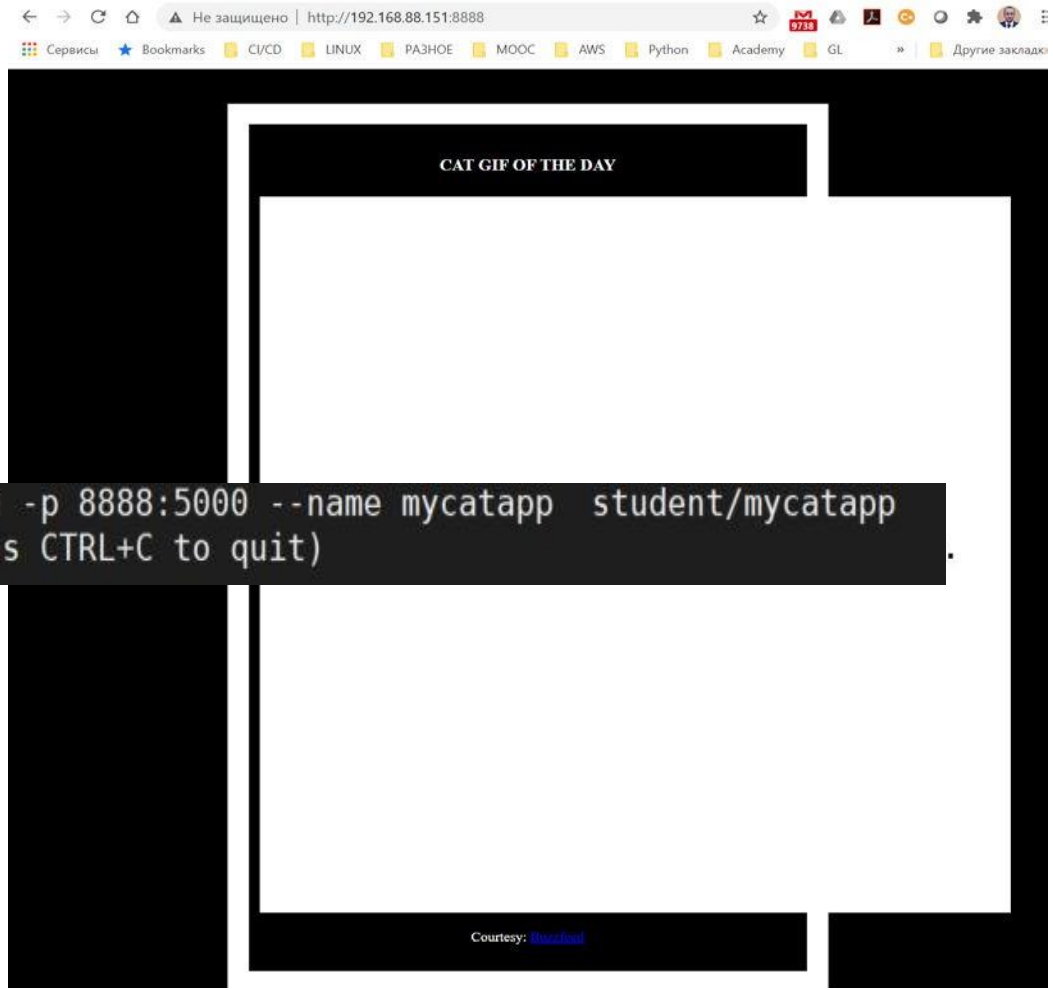
Docker image building

```
student@localhost~/flask-app$ docker build -t student/mycatapp .
Sending build context to Docker daemon  8.192kB
Step 1/9 : FROM alpine:3.5
 ---> f80194ae2e0c
Step 2/9 : RUN apk add --update py2-pip
 ---> Using cache
 ---> 73ffa428dd8f
Step 3/9 : RUN pip install --upgrade pip
 ---> Running in 4f7a40038a86
Collecting pip
  Downloading https://files.pythonhosted.org/packages/cb/28/91f26bd088ce8e22169032100d4260614fc3da435
025ff389ef1d396a433/pip-20.2.4-py2.py3-none-any.whl (1.5MB)
Installing collected packages: pip
  Found existing installation: pip 9.0.0
    Uninstalling pip-9.0.0:
      Successfully uninstalled pip-9.0.0
Successfully installed pip-20.2.4
```

Webapps with Docker

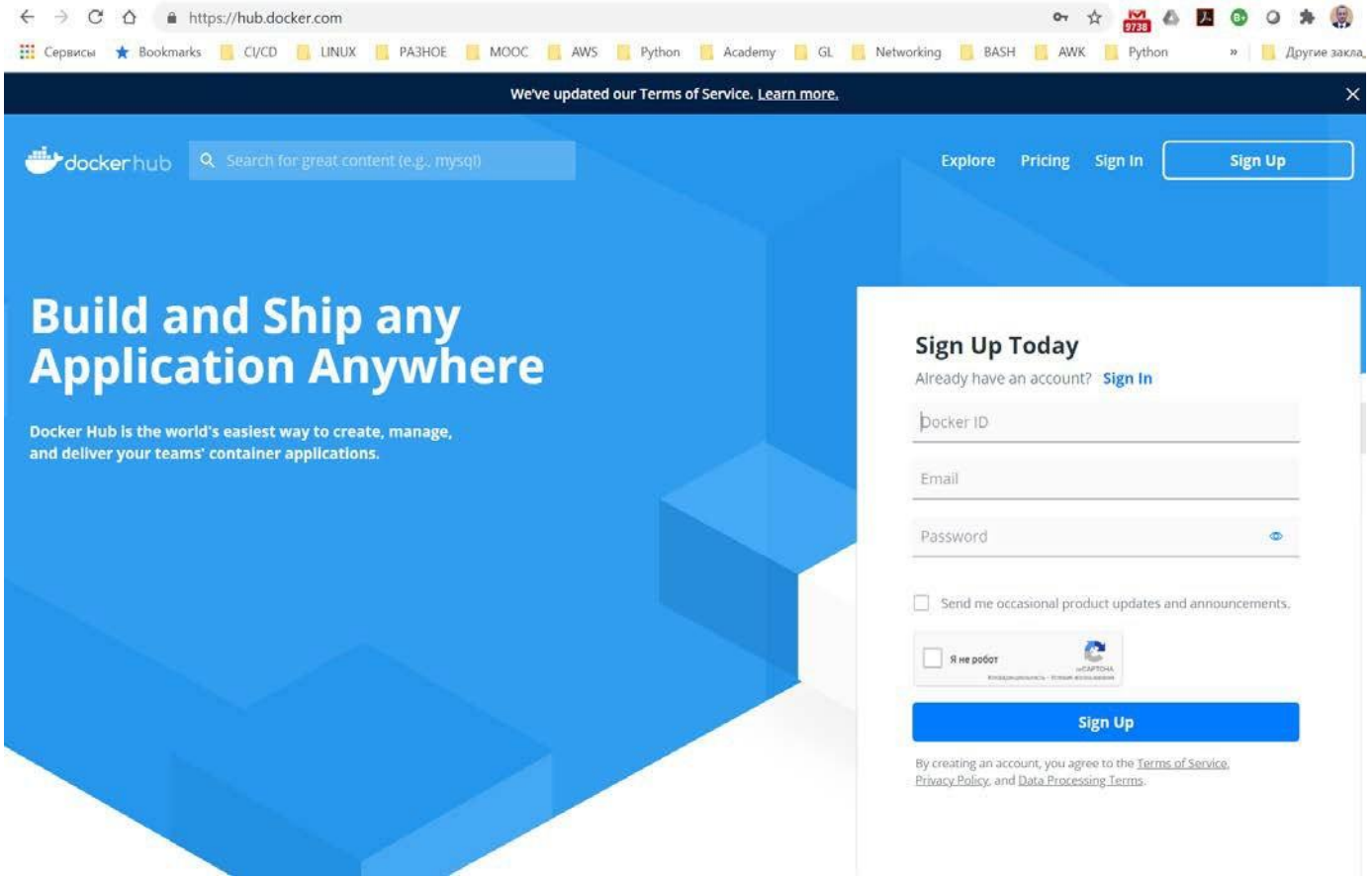
Running and accessing our we application

```
student@localhost~/flask-app$ docker run -p 8888:5000 --name mycatapp  student/mycatapp
 * Running on http://0.0.0.0:5000/ (Press CTRL+C to quit)
```



Webapps with Docker

Create an account in Docker registry



Webapps with Docker

Login to docker registry

```
student@localhost~/flask-app$ docker login
Login with your Docker ID to push and pull images from Docker Hub. If you don't have a Docker ID, hea
d over to https://hub.docker.com to create one.
Username: dimdimuzun
Password:
WARNING! Your password will be stored unencrypted in /home/student/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
```

Webapps with Docker

Pushing to docker registry

```
student@localhost~/flask-app$ docker tag student/mycatapp dimdimuzun/mycatapp
student@localhost~/flask-app$ docker push dimdimuzun/mycatapp
The push refers to repository [docker.io/dimdimuzun/mycatapp]
b21b498be7d6: Pushed
f454e8d7487a: Pushed
3a3e05191589: Pushed
2a12741f47f9: Pushed
686a8eb6a374: Pushed
ab85763677da: Pushed
f566c57e6f2d: Pushed
latest: digest: sha256:5a120d861e21201a3717d566d35bd0692d47b5aa3d754a2a101852820603b185 size: 1782
student@localhost~/flask-app$
```

Webapps with Docker. What's left behind the scenes (1)

```
Step 4/8 : RUN pip install --no-cahe-dir -r /usr/src/app/requirements.txt
 ---> Running in 99feede32af5

Usage:
  pip install [options] <requirement specifier> [package-index-options] ...
  pip install [options] -r <requirements file> [package-index-options] ...
  pip install [options] [-e] <vcs project url> ...
  pip install [options] [-e] <local project path> ...
  pip install [options] <archive url/path> ...

no such option: --no-cahe-dir
The command '/bin/sh -c pip install --no-cahe-dir -r /usr/src/app/requirements.txt' returned a non-ze
ro code: 2
student@localhost~/flask-app$ nano Dockerfile
```

Webapps with Docker. What's left behind the scenes (2)

```
student@localhost~/flask-app$ docker build -t student/mycatapp .
Sending build context to Docker daemon   7.68kB
Step 1/8 : FROM alpine:3.5
 ---> f80194ae2e0c
Step 2/8 : RUN apk add --update py2-pip
 ---> Using cache
 ---> 73ffa428dd8f
Step 3/8 : COPY requirements.txt /usr/src/app
 ---> Using cache
 ---> 56f8186797a8
Step 4/8 : RUN pip install -r /usr/src/app/requirements.txt
 ---> Running in f2a1322d4846
Could not open requirements file: [Errno 20] Not a directory: '/usr/src/app/requirements.txt'
You are using pip version 9.0.0, however version 20.2.4 is available.
You should consider upgrading via the 'pip install --upgrade pip' command.
The command '/bin/sh -c pip install -r /usr/src/app/requirements.txt' returned a non-zero code: 1
student@localhost~/flask-app$ nano Dockerfile
student@localhost~/flask-app$ docker build -t student/mycatapp .
Sending build context to Docker daemon   7.68kB
Step 1/8 : FROM alpine:3.5
 ---> f80194ae2e0c
Step 2/8 : RUN apk add --update py2-pip
 ---> Using cache
 ---> 73ffa428dd8f
Step 3/8 : COPY requirements.txt /usr/src/app
 ---> Using cache
 ---> 56f8186797a8
Step 4/8 : RUN pip install -r /usr/src/app/
 ---> Running in 161000a7437c
Could not open requirements file: [Errno 20] Not a directory: '/usr/src/app/'
You are using pip version 9.0.0, however version 20.2.4 is available.
You should consider upgrading via the 'pip install --upgrade pip' command.
The command '/bin/sh -c pip install -r /usr/src/app/' returned a non-zero code: 1
student@localhost~/flask-app$ nano Dockerfile
```

Webapps with Docker. What's left behind the scenes (3)

```
Successfully installed Flask-0.10.1 Jinja2-2.11.2 MarkupSafe-1.1.1 Werkzeug-1.0.1 itsdangerous-1.1.0
You are using pip version 9.0.0, however version 20.2.4 is available.
You should consider upgrading via the 'pip install --upgrade pip' command.
Removing intermediate container c83896d65f7d
 ---> ca5d987d265b
Step 5/8 : COPY app.py /usr/src/app/
failed to copy files: failed to create new directory: mkdir /var/lib/docker/overlay2/67b2570765f9b83e
1a8e82bd9241a13e7a37c4d4865e52f21b888fcc8fa1aff8/merged/usr/src/app: not a directory
student@localhost~/flask-app$ docker images
]REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
<none>              <none>              ca5d987d265b        59 seconds ago      57.9MB
test                v1                  452cf07c51e3        21 hours ago        419MB
test                v2                  452cf07c51e3        21 hours ago        419MB
centos              7                   7e6257c9f8d8        2 months ago        203MB
hello-world         latest              bf756fb1ae65        9 months ago        13.3kB
alpine              3.5                 f80194ae2e0c        20 months ago       4MB
student@localhost~/flask-app$ nano Dockerfile
student@localhost~/flask-app$ docker build -t student/mycatapp .
Sending build context to Docker daemon   7.68kB
Step 1/8 : FROM alpine:3.5
 ---> f80194ae2e0c
Step 2/8 : RUN apk add --update py3-pip
 ---> Running in 70fe84e0a5fc
fetch http://dl-cdn.alpinelinux.org/alpine/v3.5/main/x86_64/APKINDEX.tar.gz
fetch http://dl-cdn.alpinelinux.org/alpine/v3.5/community/x86_64/APKINDEX.tar.gz
ERROR: unsatisfiable constraints:
  py3-pip (virtual):
    provided by: python3
    required by: world[py3-pip]
The command '/bin/sh -c apk add --update py3-pip' returned a non-zero code: 1
student@localhost~/flask-app$ nano Dockerfile
```

Webapps with Docker. What's left behind the scenes (4)

```
Step 7/9 : COPY templates/index.html /usr/src/app/templates/
 ---> 7c61517099bb
Step 8/9 : EXPOSE 5000
 ---> Running in afb17d3b0a37
Removing intermediate container afb17d3b0a37
 ---> a1806762bf2f
Step 9/9 : CMD ["python", "/usr/src/app/app.py"]
 ---> Running in 38cdb64de624
Removing intermediate container 38cdb64de624
 ---> 96f4332e6f7d
Successfully built 96f4332e6f7d
Successfully tagged student/mycatapp:latest
student@localhost~/flask-app$ docker run -p 8888:5000 --name myfirstapp student/mycatapp
 * Running on http://0.0.0.0:5000/ (Press CTRL+C to quit)
192.168.88.152 - - [21/Oct/2020 15:21:32] "GET / HTTP/1.1" 200 -
192.168.88.152 - - [21/Oct/2020 15:21:33] "GET /favicon.ico HTTP/1.1" 404 -
192.168.88.152 - - [21/Oct/2020 15:21:45] "GET / HTTP/1.1" 200 -
192.168.88.152 - - [21/Oct/2020 15:21:46] "GET / HTTP/1.1" 200 -
192.168.88.152 - - [21/Oct/2020 15:21:47] "GET / HTTP/1.1" 200 -
192.168.88.152 - - [21/Oct/2020 15:21:47] "GET / HTTP/1.1" 200 -
192.168.88.152 - - [21/Oct/2020 15:21:48] "GET / HTTP/1.1" 200 -
192.168.88.152 - - [21/Oct/2020 15:21:48] "GET / HTTP/1.1" 200 -
192.168.88.152 - - [21/Oct/2020 15:21:48] "GET / HTTP/1.1" 200 -
192.168.88.152 - - [21/Oct/2020 15:21:49] "GET / HTTP/1.1" 200 -
```

https://github.com/docker/labs/tree/master/beginner

Webapps with Docker. What's left behind the scenes (5)

```
student@localhost~/flask-app$ docker login
Authenticating with existing credentials...
WARNING! Your password will be stored unencrypted in /home/student/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
student@localhost~/flask-app$ docker push student/mycatapp
The push refers to repository [docker.io/student/mycatapp]
b21b498be7d6: Preparing
f454e8d7487a: Preparing
3a3e05191589: Preparing
2a12741f47f9: Preparing
686a8eb6a374: Preparing
ab85763677da: Waiting
f566c57e6f2d: Waiting
denied: requested access to the resource is denied
```

Frequently used Docker commands

```
$ docker ps [-a]                          #list
$ docker stop $(docker ps -a -q)          #stop all containers [you need stop before delete]
$ docker rm 0fd99ee0cb61                  #remove a single container
$ docker images -a                        # list
$ docker rmi $(docker images -a -q)       # remove all images

$ docker search tomcat
$ docker pull tomcat
$ docker search nginx
$ docker pull nginx
$ docker run -it -p 8889:8080 tomcat
$ docker run -it -p 8888:80 nginx
$ docker run -d -p 8890:80 nginx
```

Q&A


Thank you!