

Assignment # 3 (Kubernetes)

Group members:

19P-1652 Hifza Majeed

19P-1664 Noman

19P-1672 Ahmad

Step 1:

1. install the docker desktop

```
C:\Users\92325> docker version
error during connect: This error may indicate
r_engine/v1.24/version": open //./pipe/docker_
Client:
Cloud integration: v1.0.29
Version:          20.10.21
API version:      1.41
Go version:       go1.18.7
Git commit:       baeda1f
Built:            Tue Oct 25 18:08:16 2022
OS/Arch:          windows/amd64
Context:          default
Experimental:     true
```

2. install kubectl

```
PS C:\Windows\system32> kubectl virsion
error: unknown command "virsion" for "kubectl"

Did you mean this?
    version
PS C:\Windows\system32> kubectl version
WARNING: This version information is deprecated and will be replaced with the
output from kubectl version --short. Use --output=yaml|json to get the full
version.
Client Version: version.Info{Major:"1", Minor:"25", GitVersion:"v1.25.2", Git
Commit:"5835544ca568b757a8ecae5c153f317e5736700e", GitTreeState:"clean", Buil
dDate:"2022-09-21T14:33:49Z", GoVersion:"go1.19.1", Compiler:"gc", Platform:"
windows/amd64"}
Kustomize Version: v4.5.7
Server Version: version.Info{Major:"1", Minor:"25", GitVersion:"v1.25.3", Git
Commit:"434bfd82814af038ad94d62ebe59b133fcb50506", GitTreeState:"clean", Buil
dDate:"2022-10-12T10:49:09Z", GoVersion:"go1.19.2", Compiler:"gc", Platform:"
linux/amd64"}
```

3. install minikube

```
PS C:\Windows\system32> minikube versin
Error: unknown command "versin" for "minikube"

Did you mean this?
    version

Run 'minikube --help' for usage.
PS C:\Windows\system32> minikube version
minikube version: v1.28.0
commit: 986blebd987211ed16f8cc10aed7d2c42fc8392f
PS C:\Windows\system32>
```

Step 2:

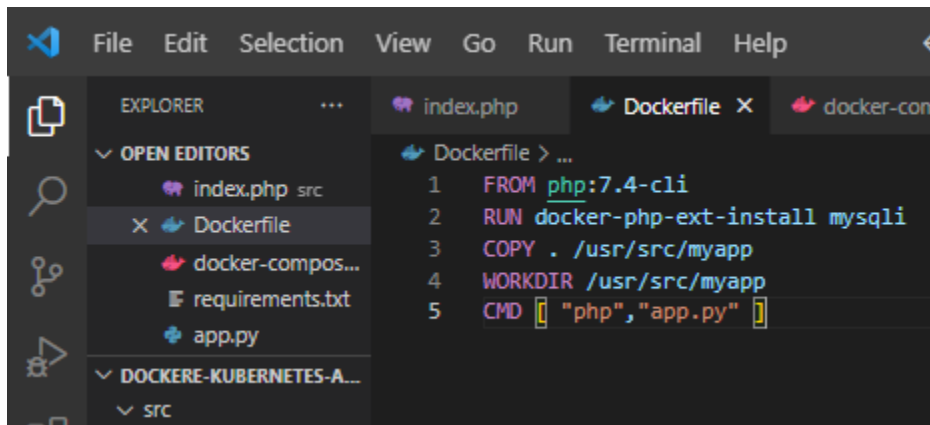
After the docker assignment 2, we start further work to continue in Kubernetes

For Kubernetes, we create a deployment.yaml file

These files create two potions one is services and the other is the deployment

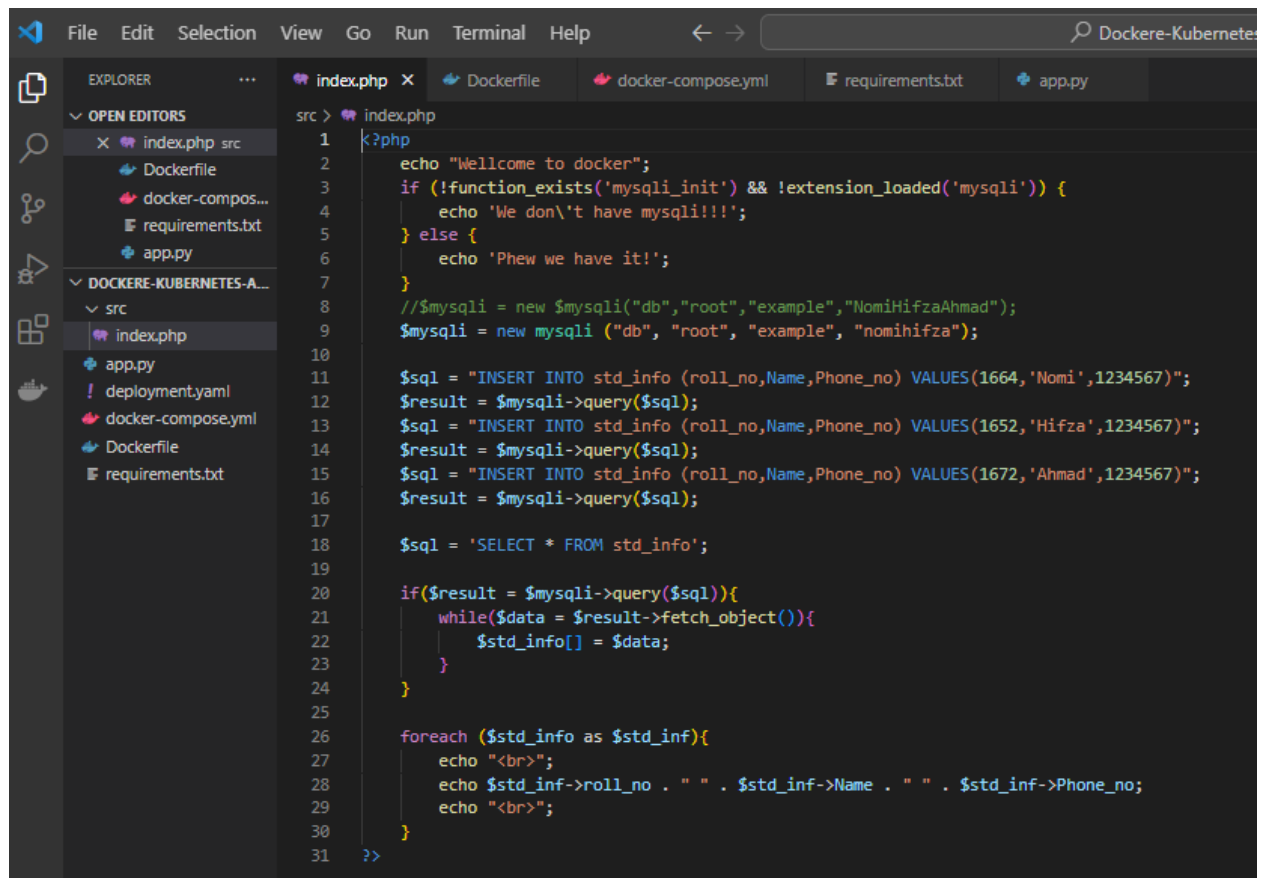
Code of all files:

Dockerfile



```
File Edit Selection View Go Run Terminal Help
EXPLORER
OPEN EDITORS
  index.php src
  Dockerfile
  docker-compos...
  requirements.txt
  app.py
DOCKERE-KUBERNETES-A...
  src
    index.php
    Dockerfile
    docker-compos...
    requirements.txt
    app.py
Dockerfile > ...
1 FROM php:7.4-cli
2 RUN docker-php-ext-install mysqli
3 COPY . /usr/src/myapp
4 WORKDIR /usr/src/myapp
5 CMD [ "php", "app.py" ]
```

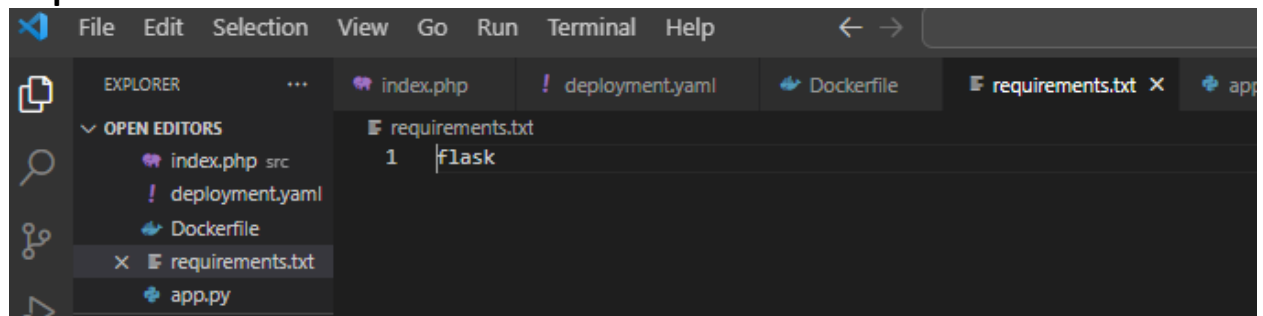
Index.php



The screenshot shows the Visual Studio Code editor with the 'index.php' file open. The Explorer sidebar on the left shows the project structure with folders 'src' and 'DOCKERE-KUBERNETES-A...' and files 'app.py', 'deployment.yaml', 'docker-compose.yml', 'Dockerfile', and 'requirements.txt'. The main editor area displays PHP code for a web application that connects to a MySQL database and inserts data into a table named 'std_info'.

```
1 <?php
2 echo "Wellcome to docker";
3 if (!function_exists('mysqli_init') && !extension_loaded('mysqli')) {
4     echo 'We don\'t have mysqli!!!';
5 } else {
6     echo 'Phew we have it!';
7 }
8 // $mysqli = new mysqli("db","root","example","NomiHifzaAhmad");
9 $mysqli = new mysqli ("db", "root", "example", "nomihifza");
10
11 $sql = "INSERT INTO std_info (roll_no,Name,Phone_no) VALUES(1664,'Nomi',1234567)";
12 $result = $mysqli->query($sql);
13 $sql = "INSERT INTO std_info (roll_no,Name,Phone_no) VALUES(1652,'Hifza',1234567)";
14 $result = $mysqli->query($sql);
15 $sql = "INSERT INTO std_info (roll_no,Name,Phone_no) VALUES(1672,'Ahmad',1234567)";
16 $result = $mysqli->query($sql);
17
18 $sql = 'SELECT * FROM std_info';
19
20 if($result = $mysqli->query($sql)){
21     while($data = $result->fetch_object()){
22         $std_info[] = $data;
23     }
24 }
25
26 foreach ($std_info as $std_inf){
27     echo "<br>";
28     echo $std_inf->roll_no . " " . $std_inf->Name . " " . $std_inf->Phone_no;
29     echo "<br>";
30 }
31 >>
```

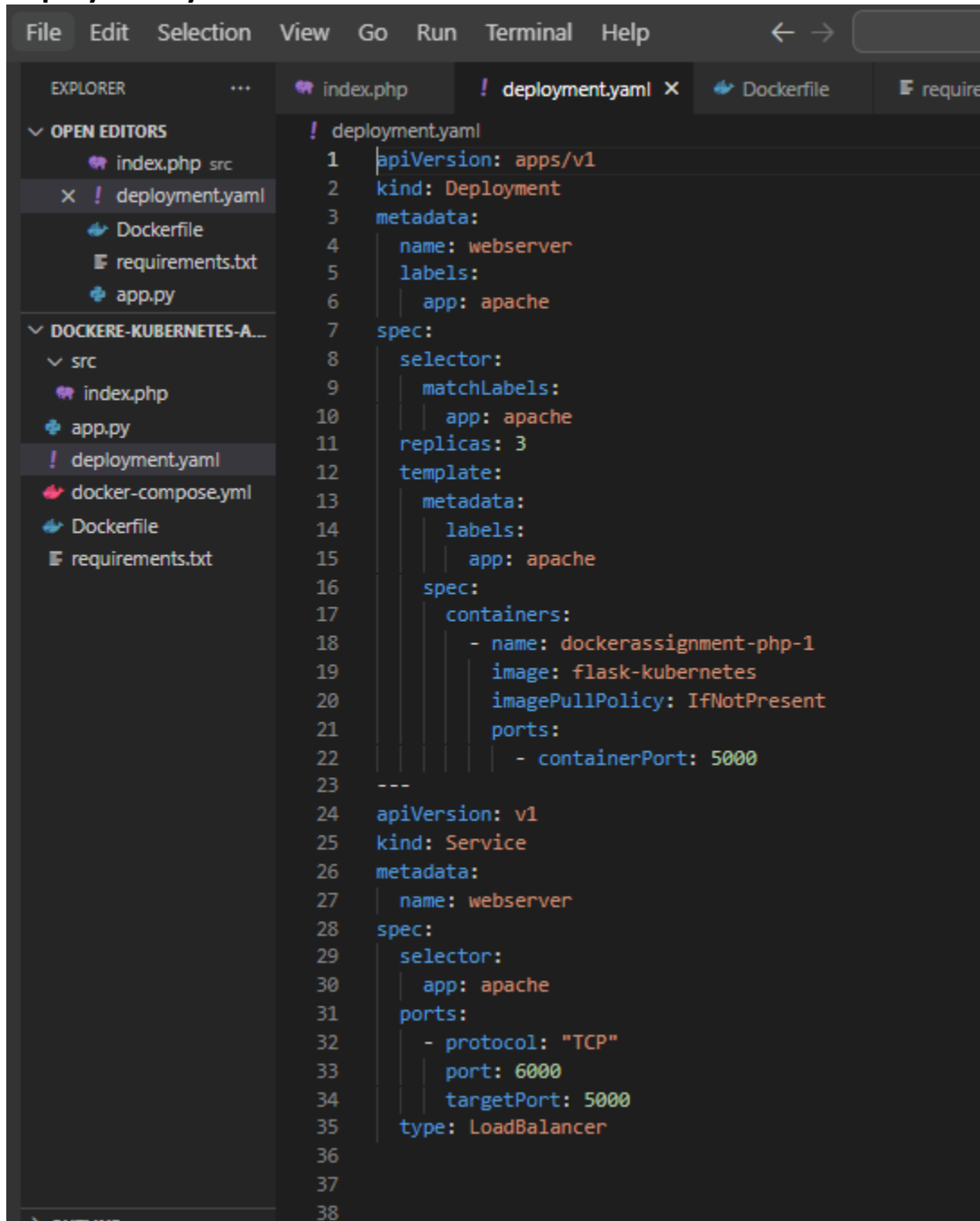
requirements.txt



The screenshot shows the Visual Studio Code editor with the 'requirements.txt' file open. The Explorer sidebar on the left shows the project structure with folders 'src' and 'DOCKERE-KUBERNETES-A...' and files 'app.py', 'deployment.yaml', 'docker-compose.yml', 'Dockerfile', and 'requirements.txt'. The main editor area displays the content of the requirements.txt file, which is a single line: 'flask'.

```
1 flask
```

deployment.yaml



```
File Edit Selection View Go Run Terminal Help
EXPLORER
  OPEN EDITORS
    index.php src
    ! deployment.yaml
    Dockerfile
    requirements.txt
    app.py
  DOCKERE-KUBERNETES-A...
    src
      index.php
      app.py
      ! deployment.yaml
      docker-compose.yml
      Dockerfile
      requirements.txt
! deployment.yaml
1  apiVersion: apps/v1
2  kind: Deployment
3  metadata:
4    name: webserver
5    labels:
6      app: apache
7  spec:
8    selector:
9      matchLabels:
10       app: apache
11    replicas: 3
12    template:
13      metadata:
14        labels:
15          app: apache
16      spec:
17        containers:
18          - name: dockerassignment-php-1
19            image: flask-kubernetes
20            imagePullPolicy: IfNotPresent
21            ports:
22              - containerPort: 5000
23    ---
24    apiVersion: v1
25    kind: Service
26    metadata:
27      name: webserver
28    spec:
29      selector:
30        app: apache
31      ports:
32        - protocol: "TCP"
33          port: 6000
34          targetPort: 5000
35      type: LoadBalancer
36
37
38
```

app.py

```
File Edit Selection View Go Run Terminal Help
EXPLORER index.php deployment.yaml Dockerfile requirements.txt app.py
OPEN EDITORS
index.php src
deployment.yaml
Dockerfile
requirements.txt
X app.py
DOCKERC...
src
index.php
app.py
deployment.yaml
docker-compose.yml
Dockerfile
requirements.txt
```

```
1 from flask import Flask, jsonify, render_template
2 import time
3 app = Flask(__name__)
4
5 @app.route("/")
6 def myfunction():
7     return render_template('index.php')
8
9 if __name__ == "__main__":
10     app.run(host='0.0.0.0', debug=True)
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
-> [internal] load build definition from Dockerfile
-> [internal] load .dockerignore
-> [internal] load metadata for docker.io/library/python:3.7
-> [internal] load build context
-> transferring context: 1.66kB
-> [1/5] FROM docker.io/library/python:3.7@sha256:520bd41d196f37b077c3e586009785cc698a7afd11291209324cc267780c8657
-> resolve docker.io/library/python:3.7@sha256:520bd41d196f37b077c3e586009785cc698a7afd11291209324cc267780c8657
-> sha256:520bd41d196f37b077c3e586009785cc698a7afd11291209324cc267780c8657 1.86kB / 1.86kB
-> sha256:19675f873997771853c774be3f9c29a2aed309bb6614408c04514996163aac7b 2.22kB / 2.22kB
-> sha256:5cda39795abb5d66354c474fdd58f880141a593de54e292f2610e7757ae98aef 9.20kB / 9.20kB
-> sha256:5c8cfbf51e6e6869f1af2a1e7067b07fd6733036a33c9d29f743b0285e26037 5.16kB / 5.16kB
-> sha256:aa3a609d15798d35c1484072876b7d22a316e98de6a097de33b9daded6a689cd1 10.88kB / 10.88kB
-> sha256:f2f58072e9ed1aa1b0143341c5ee83815c00ce4754b309fa240155067ab0e698 55.04kB / 55.04kB
-> sha256:094e7d9bb04ebf214ea8dc5a488995449684104ae8ad9603bf061cac0d18eb54 54.59kB / 54.59kB
-> sha256:2cbfd734f3824a4390fe4be45f6a11a5543bca1023e4814d72460aeabc2eeff89 196.87kB / 196.87kB
-> sha256:a88ac293d0fa66515f0a670445969ba98dd8d6a114a7f6ae934aaad44084d0 6.29kB / 6.29kB
-> sha256:e8aea10b41aba7606151f26f7590f539bf99da2d82c6d1cfe13623a8dbb453e3 15.49kB / 15.49kB
-> extracting sha256:f2f58072e9ed1aa1b0143341c5ee83815c00ce4754b309fa240155067ab0e698 3.0kB
-> extracting sha256:5c8cfbf51e6e6869f1af2a1e7067b07fd6733036a33c9d29f743b0285e26037 0.44kB
-> extracting sha256:aa3a609d15798d35c1484072876b7d22a316e98de6a097de33b9daded6a689cd1 0.44kB
-> extracting sha256:094e7d9bb04ebf214ea8dc5a488995449684104ae8ad9603bf061cac0d18eb54 2.8kB
-> sha256:3f1c4dc605dc72fd0baec6d66733fee7732cd6b82e96a04279e0907cb97fa15ed 2330 / 2330
-> sha256:ee7d24a3adf39850d5fed64525837db7ef1c36681e74e1057bf6a8bcafa5ce842 2.89kB / 2.89kB
-> extracting sha256:2cbfd734f3824a4390fe4be45f6a11a5543bca1023e4814d72460aeabc2eeff89 12.3kB
-> extracting sha256:a88ac293d0fa66515f0a670445969ba98dd8d6a114a7f6ae934aaad44084d0 0.44kB
-> extracting sha256:e8aea10b41aba7606151f26f7590f539bf99da2d82c6d1cfe13623a8dbb453e3 1.1kB
-> extracting sha256:3f1c4dc605dc72fd0baec6d66733fee7732cd6b82e96a04279e0907cb97fa15ed 0.0kB
-> extracting sha256:ee7d24a3adf39850d5fed64525837db7ef1c36681e74e1057bf6a8bcafa5ce842 0.55kB
-> [2/5] RUN mkdir /app
-> [3/5] WORKDIR /app/
-> [4/5] ADD . /app/
-> [5/5] RUN pip install -r requirements.txt
-> exporting to image
-> exporting layers
-> writing image sha256:e0785271457aedc694fc718dd94b9720e140e621653d74bf23e2dbc3301926a
-> naming to docker.io/library/flask-kubernetes
Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them
PS G:\Semester_7\Dockere-Kubernetes-Assignment> docker build -t flask-kubernetes .
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
PS G:\Semester_7\Dockere-Kubernetes-Assignment> docker images
REPOSITORY TAG IMAGE ID CREATED SIZE
flask-kubernetes latest e0785271457a 30 minutes ago 918MB
docker-assignment-php latest fcbdd81dc176 5 days ago 453MB
my-php-app latest b14180fb6783 5 days ago 474MB
<none> <none> 9d2ddebca4e91 5 days ago 474MB
mysql latest 7484689f290f 6 days ago 538MB
php 7.4-apache 20a3732f422b 4 weeks ago 453MB
adminer latest ef63a68bb1a5 4 weeks ago 91.3MB
php 7.2-apache c61d277263e1 2 years ago 410MB
PS G:\Semester_7\Dockere-Kubernetes-Assignment>
```