

## HW\_24

### Структура:

- python-random.py
- Dockerfile
- deployment.yaml
- service.yaml

### python-random.py

```
python-random.py
1 #!/usr/bin/python3
2
3 from http.server import BaseHTTPRequestHandler, HTTPServer
4 import random
5
6 class SimpleHTTPRequestHandler(BaseHTTPRequestHandler):
7     def do_GET(self):
8         if self.path == '/':
9             self.send_response(200)
10            self.send_header('Content-Type', 'text/plain; charset=utf-8')
11            self.end_headers()
12            self.wfile.write(bytes(random_name, "UTF-8"))
13        else:
14            self.send_response(404)
15            self.end_headers()
16
17 def run(server_class=HTTPServer, handler_class=SimpleHTTPRequestHandler, port=8082):
18     server_address = ('0.0.0.0', port)
19     httpd = server_class(server_address, handler_class)
20     print(f'Starting httpd server on port {port}...')
21     httpd.serve_forever()
22
23 if __name__ == '__main__':
24     random_name = f'name-{random.randint(0, 255)}'
25     run()
26
```

### Dockerfile

```
*Dockerfile
1 # Використовуємо офіційний Python-образ
2 FROM python:3.9-slim
3
4 # Копіюємо скрипт у контейнер
5 COPY python-random.py /app/python-random.py
6
7 # Переходимо в робочу директорію
8 WORKDIR /app
9
10 # Встановлюємо необхідні залежності
11 RUN chmod +x python-random.py
12
13 # Відкриваємо порт 8082
14 EXPOSE 8082
15
16 # Запускаємо додаток
17 CMD ["python3", "python-random.py"]
18
```

### deployment.yaml

```
deployment.yaml
1 apiVersion: apps/v1
2 kind: Deployment
3 metadata:
4   name: python-random-deployment
5   labels:
6     app: python-random
7 spec:
8   replicas: 3
9   selector:
10     matchLabels:
11       app: python-random
12   template:
13     metadata:
14       labels:
15         app: python-random
16     spec:
17       containers:
18       - name: python-random
19         image: dmytrovaschuk/python-random:latest
20         ports:
21         - containerPort: 8082
22
```

### service.yaml

```
service.yaml
1 apiVersion: v1
2 kind: Service
3 metadata:
4   name: python-random-service
5 spec:
6   type: NodePort
7   selector:
8     app: python-random
9   ports:
10   - protocol: TCP
11     port: 8082
12     targetPort: 8082
13     nodePort: 30082
14
```

## Перевіримо статус Minikube

```
dmytro@ubuntu22:~/dan_it_homeworks/Homework_24/src$ minikube status
minikube
type: Control Plane
host: Running
kubelet: Running
apiserver: Running
kubeconfig: Configured
```

## Створимо Docker-образ

```
dmytro@ubuntu22:~/dan_it_homeworks/Homework_24/src$ docker build -t dmytrovashchuk/python-random:latest .
[+] Building 22.1s (9/9) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 610B
=> [internal] load metadata for docker.io/library/python:3.9-slim
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load build context
=> => transferring context: 876B
=> [1/4] FROM docker.io/library/python:3.9-slim@sha256:bb8009c87ab89e751a1dd2c6c7f8abaae3d9fce8e072802d4a23c95594d16d84
=> => resolve docker.io/library/python:3.9-slim@sha256:bb8009c87ab89e751a1dd2c6c7f8abaae3d9fce8e072802d4a23c95594d16d84
=> => sha256:bb8009c87ab89e751a1dd2c6c7f8abaae3d9fce8e072802d4a23c95594d16d84 10.41kB / 10.41kB
=> => sha256:ddb56f2e39ec00c79c1207f182c60e03ddf417525b56ef467e1519706792cd 1.75kB / 1.75kB
=> => sha256:453d3342b002fa5f904ba0cd72a07accb5121641d20776f3e64339842f275d38 5.28kB / 5.28kB
=> => sha256:af302e5c37e9dc1dbe2eadc8f5059d82a914066b541bd1a6daa91d0cc55057d 28.21MB / 28.21MB
=> => sha256:1da0723265ec311debef6bec17d4fae5f1e5f7809fca4378aac265cdef238f1c 3.51MB / 3.51MB
=> => sha256:4f4cb1a24c66f1a92f204ba0bbdd2a7c941a853c83161ffa38bbfa121448861 14.93MB / 14.93MB
=> => sha256:c876ae22765e4a125855eb121718c3f8f07bd8b0daead4e68e716571961f37 249B / 249B
=> => extracting sha256:af302e5c37e9dc1dbe2eadc8f5059d82a914066b541bd1a6daa91d0cc55057d
=> => extracting sha256:1da0723265ec311debef6bec17d4fae5f1e5f7809fca4378aac265cdef238f1c
=> => extracting sha256:4f4cb1a24c66f1a92f204ba0bbdd2a7c941a853c83161ffa38bbfa121448861
=> => extracting sha256:c876ae22765e4a125855eb121718c3f8f07bd8b0daead4e68e716571961f37
=> [2/4] COPY python-random.py /app/python-random.py
=> [3/4] WORKDIR /app
=> [4/4] RUN chmod +x python-random.py
=> => exporting to image
=> => exporting layers
=> => writing image sha256:8088e9c756040e670de3a41a7e24769f7bdfec389d10b92cf5058c2f27bd5c3
=> => naming to docker.io/dmytrovashchuk/python-random:latest
dmytro@ubuntu22:~/dan_it_homeworks/Homework_24/src$ docker login -u dmytrovashchuk
Password:
WARNING! Your password will be stored unencrypted in /home/dmytro/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credential-stores
Login Succeeded
```

## Публікація образу у Docker Hub ( попередньо залогінувшись )

```
dmytro@ubuntu22:~/dan_it_homeworks/Homework_24/src$ docker push dmytrovashchuk/python-random:latest
The push refers to repository [docker.io/dmytrovashchuk/python-random]
4de926f84ca2: Pushed
5f70bf18a086: Pushed
d30c84b607e7: Pushed
9ab7333053f8: Layer already exists
c0ecdab1104b: Layer already exists
6f208d2e47a8: Layer already exists
f5fe472da253: Layer already exists
latest: digest: sha256:43e9024871c9cf18d5ab3753c20dfad5b9917d0747ef6fa6f75c7a2b65e77057 size: 1779
```

dockerhub

Explore

Repositories

Organizations

Usage

Search Docker Hub

Ctrl+K

dmytrovashchuk

Search by repository name

All content

Create a repository

Name	Last Pushed	Contains	Visibility	Scout
dmytrovashchuk/python-random	31 minutes ago	IMAGE	Public	Inactive

## Деплоїмо у Kubernetes

```
dmytro@ubuntu22:~/dan_it_homeworks/Homework_24/src$ kubectl apply -f deployment.yaml
deployment.apps/python-random-deployment created
dmytro@ubuntu22:~/dan_it_homeworks/Homework_24/src$ kubectl apply -f service.yaml
service/python-random-service created
```

## Перевірка

```
dmytro@ubuntu22:~/dan_it_homeworks/Homework_24/src$ kubectl get svc
NAME                TYPE          CLUSTER-IP      EXTERNAL-IP      PORT(S)          AGE
kubernetes          ClusterIP     10.96.0.1       <none>           443/TCP          22d
nginx-service       LoadBalancer 10.110.153.187   <pending>        80:30683/TCP     22d
python-random-service NodePort      10.102.51.109   <none>           8082:30082/TCP   2m23s
dmytro@ubuntu22:~/dan_it_homeworks/Homework_24/src$ minikube ip
192.168.49.2
dmytro@ubuntu22:~/dan_it_homeworks/Homework_24/src$ curl http://192.168.49.2:30082
name-61dmytro@ubuntu22:~/dan_it_homeworks/Homework_24/src$
```