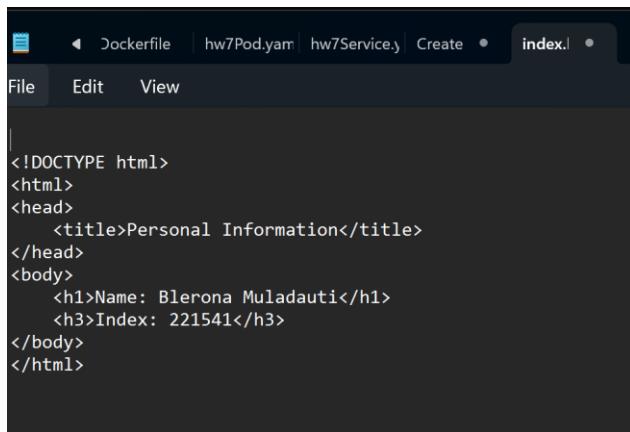


# Homework 8 – DevOps

Blerona Muladauti 221541

## 1. Create a static index.html with your info in it ○ name, surname, index

Index.html:

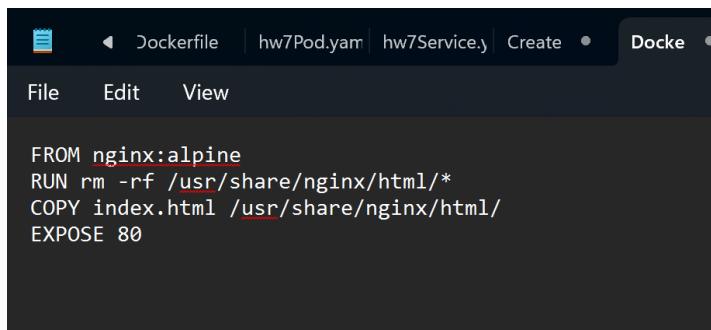


```
<!DOCTYPE html>
<html>
<head>
    <title>Personal Information</title>
</head>
<body>
    <h1>Name: Blerona Muladauti</h1>
    <h3>Index: 221541</h3>
</body>
</html>
```

## 2. Dockerize it with nginx

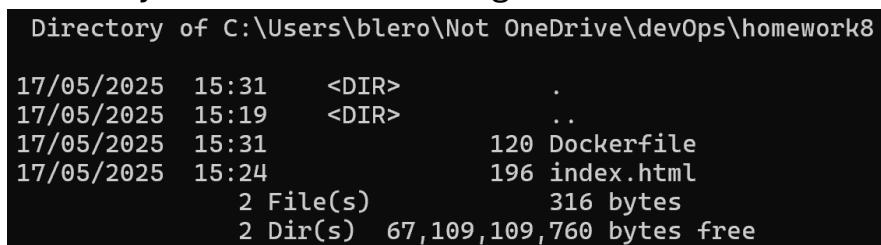
I'm using the official nginx:alpine, it removes the default content, and copies my index.html into the web server's root directory. This allows to serve my custom index.html page.

Dockerfile:



```
FROM nginx:alpine
RUN rm -rf /usr/share/nginx/html/*
COPY index.html /usr/share/nginx/html/
EXPOSE 80
```

Directory content after creating both files:



```
Directory of C:\Users\blero\Not OneDrive\devOps\homework8

17/05/2025  15:31    <DIR>      .
17/05/2025  15:19    <DIR>      ..
17/05/2025  15:31                  120 Dockerfile
17/05/2025  15:24                  196 index.html
              2 File(s)            316 bytes
              2 Dir(s)  67,109,109,760 bytes free
```

### 3. Deploy it to dockerhub with tag 1.0

```
C:\Users\blero\Not OneDrive\devOps\homework8>docker build -t rona03/homework8image:1.0
2025/05/17 15:56:51 http2: server: error reading preface from client //./pipe/docker_engine: file has already been closed
[*] Building 0.0s (0/0) docker:default
2025/05/17 15:56:51 http2: server: error reading preface from client //./pipe/docker_engine: file has already been closed
[*] Building 1.2s (9/9) FINISHED docker:default
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 143B 0.0s
=> [internal] load metadata for docker.io/library/nginx:alpine 1.0s
=> [auth] library/nginx:pull token for registry-1.docker.io 0.0s
=> [internal] load .dockerrcignore 0.0s
=> => transferring context: 2B 0.0s
=> [internal] load build context 0.0s
=> => transferring context: 32B 0.0s
=> CACHED [2/3] RUN rm -rf /usr/share/nginx/html/* 0.0s
=> CACHED [3/3] COPY index.html /usr/share/nginx/html/ 0.0s
=> exporting to image 0.0s
=> => exporting layers 0.0s
=> => writing image sha256:0cb14decfa7a1a5f254acef184cceff67af9f6ffd0b17df5f89dd2fb6e65713f3 0.0s
=> => naming to docker.io/rona03/homework8image:1.0 0.0s
View build details: docker-desktop://dashboard/build/default/zf4my5ugdjvhxsxhi72qqb9u9
```

```
C:\Users\blero\Not OneDrive\devOps\homework8>docker images
REPOSITORY                                TAG
IMAGE ID         CREATED        SIZE
rona03/homework8image                     1.0
9cb14decfa7a    7 minutes ago  48.2MB
```

```
C:\Users\blero\Not OneDrive\devOps\homework8>docker push rona03/homework8image:1.0
The push refers to repository [docker.io/rona03/homework8image]
```

```
906a5df39341: Pushed
708939822676: Pushed
0d853d50b128: Mounted from rona03/nginx-static
947e805a4ac7: Mounted from rona03/nginx-static
811a4dbbf4a5: Mounted from rona03/nginx-static
b8d7d1d22634: Mounted from rona03/nginx-static
e244aa659f61: Mounted from rona03/nginx-static
c56f134d3805: Mounted from rona03/nginx-static
d71cae0084c1: Mounted from rona03/nginx-static
08000c18d16d: Mounted from rona03/nginx-static
1.0: digest: sha256:97f619efef43c1b52347cd09ccb8a6cef361290247d73c3a189db767baa8e8e43 size: 2403
```

**Repositories**

All repositories within the rona03 namespace.

Name	Last Pushed	Contains	Visibility	Scopes
rona03/homework8image	1 minute ago	IMAGE	Public	In

### 4. Write a deployment manifest, 5 replicas, 2 max surge, 1 max unavailable

Creating the .yaml file for the deployment:

```
File Edit View
File Edit View
apiVersion: apps/v1
kind: Deployment
metadata:
  name: h8-deployment
spec:
  replicas: 5
  strategy:
    rollingUpdate:
      maxSurge: 2
      maxUnavailable: 1
  selector:
    matchLabels:
      app: h8-app
  template:
    metadata:
      labels:
        app: h8-app
    spec:
      containers:
        - name: h8-container
          image: rona03/homework8image:1.0
          ports:
            - containerPort: 80
```

## 5. Deploy it to your cluster

Deploying and checking the pods:

```
C:\Users\blero\Not OneDrive\devOps\homework8>notepad h8-deploy.yaml  
C:\Users\blero\Not OneDrive\devOps\homework8>kubectl apply -f h8-deploy.yaml  
deployment.apps/h8-deployment created  
  
C:\Users\blero\Not OneDrive\devOps\homework8>kubectl get deployments  
NAME           READY   UP-TO-DATE   AVAILABLE   AGE  
h8-deployment  5/5     5            5           16s  
  
C:\Users\blero\Not OneDrive\devOps\homework8>kubectl get pods  
NAME           READY   STATUS    RESTARTS   AGE  
h8-deployment-6f4f47b5d6-29j9n  1/1     Running   0          22s  
h8-deployment-6f4f47b5d6-6knrf   1/1     Running   0          22s  
h8-deployment-6f4f47b5d6-djz76  1/1     Running   0          22s  
h8-deployment-6f4f47b5d6-h7dnt  1/1     Running   0          22s  
h8-deployment-6f4f47b5d6-xm52d  1/1     Running   0          22s  
  
C:\Users\blero\Not OneDrive\devOps\homework8>
```

## 6. Refactor index.html with current year in it - tag 2.0

```
File Edit View  
  
<!DOCTYPE html>  
<html>  
<head>  
    <title>Personal Information</title>  
</head>  
<body>  
    <h1>Name: Blerona Muladauti</h1>  
    <h3>Index: 221541</h3>  
    <h3>Current Year: 2025</h3> +  
</body>  
</html>
```

```
C:\Users\blero\Not OneDrive\devOps\homework8>docker build -t rona03/homework8image:2.0 .  
[+] Building 0.0s (0/0) docker:default  
2025/05/17 16:11:01 http2: server: error reading preface from client //./pipe/docker_engine: file has already been close  
[+] Building 1.5s (9/9) FINISHED  
=> [internal] load build definition from Dockerfile  
=> => transferring dockerfile: 143B  
=> [internal] load metadata for docker.io/library/nginx:alpine  
=> [auth] library/nginx:pull token for registry-1.docker.io  
=> [internal] load .dockerrignore  
=> => transferring context: 2B  
=> [1/3] FROM docker.io/library/nginx:alpine@sha256:65645c7bb6a0661892a8b03b89d0743208a18dd2f3f17a54ef4b76fb8e2f  
=> [internal] load build context  
=> => transferring context: 249B  
=> CACHED [2/3] RUN rm -rf /usr/share/nginx/html/*  
=> [3/3] COPY index.html /usr/share/nginx/html/  
=> exporting to image  
=> => exporting layers  
=> => writing image sha256:d91f30455f5bc6180c644f81693f12f54bbea62085abff52891b1b20bc04bd0  
=> => naming to docker.io/rona03/homework8image:2.0
```

REPOSITORY	TAG
rona03/homework8image	2.0
d91f30455f5b	33 seconds ago
rona03/homework8image	1.0
0cb14decfa7a	21 minutes ago
	48.2MB

```
C:\Users\blero\Not OneDrive\devOps\homework8>docker push rona03/homework8image:2.0
The push refers to repository [docker.io/rona03/homework8image]
8d5326a7cd30: Pushed
70893982267e: Layer already exists
0d853d50b128: Layer already exists
947e805a4ac7: Layer already exists
811a4dbbf4a5: Layer already exists
b8d7d1d22634: Layer already exists
e244aa659f61: Layer already exists
c56f134d3805: Layer already exists
d71eae0084c1: Layer already exists
08000c18d16d: Layer already exists
2.0: digest: sha256:286d3ad8ac28ab89d881eeb10479110da57b63222dc0af0f26df006994bc2f3e size: 2403
```

## 7. Make a rollout with 2.0 image tag

to rollout the update I changed the image tag in the .yaml file from 1.0 to 2.0

```
spec:
  containers:
    - name: h8-container
      image: rona03/homework8image:2.0
    ports:
      - containerPort: 80
```

Reapplied changes:

```
C:\Users\blero\Not OneDrive\devOps\homework8>kubectl apply -f h8-deploy.yaml
deployment.apps/h8-deployment configured
```

Checking the rollout status

```
C:\Users\blero\Not OneDrive\devOps\homework8>kubectl rollout status deployment h8-deployment
deployment "h8-deployment" successfully rolled out
```

These are the running pods after the rollout changes

NAME	READY	STATUS	RESTARTS	AGE	IP	NODE	NOMINATED NODE	READIN
h8-deployment-5c5b54cffc-74x8f	1/1	Running	0	63s	10.1.0.20	docker-desktop	<none>	<none>
h8-deployment-5c5b54cffc-gffcl	1/1	Running	0	65s	10.1.0.16	docker-desktop	<none>	<none>
h8-deployment-5c5b54cffc-gzqfb	1/1	Running	0	65s	10.1.0.17	docker-desktop	<none>	<none>
h8-deployment-5c5b54cffc-hpml4	1/1	Running	0	63s	10.1.0.19	docker-desktop	<none>	<none>
h8-deployment-5c5b54cffc-vbb52	1/1	Running	0	65s	10.1.0.18	docker-desktop	<none>	<none>

## 8. Revert to 1.0 version

The change, rollout is supposed to be in history. I used the kubectl annotate command to note the change-cause so its visible for me here.

```
C:\Users\blero\Not OneDrive\devOps\homework8>kubectl rollout history deployment h8-deployment  
deployment.apps/h8-deployment  
REVISION  CHANGE-CAUSE  
1        <none>  
2        Rolled out version 2.0
```

**Revision 1** = the first deployment using image 1.0

**Revision 2** = The rollout where you updated the image to 2.0

**Change-Cause (I manually named it) – Rolled out version 2.0**

As for going back to v1.0 rollback, since I have only one rollout I think its safe to use just the undo command without specifying the Revision

```
C:\Users\blero\Not OneDrive\devOps\homework8>kubectl rollout undo deployment h8-deployment  
deployment.apps/h8-deployment rolled back
```

```
C:\Users\blero\Not OneDrive\devOps\homework8>kubectl get pods  
NAME                  READY   STATUS    RESTARTS   AGE  
h8-deployment-6f4f47b5d6-42mg8   1/1     Running   0          17s  
h8-deployment-6f4f47b5d6-82prj   1/1     Running   0          16s  
h8-deployment-6f4f47b5d6-m9r78   1/1     Running   0          17s  
h8-deployment-6f4f47b5d6-sqxc7   1/1     Running   0          17s  
h8-deployment-6f4f47b5d6-vxcz4   1/1     Running   0          16s
```

```
C:\Users\blero\Not OneDrive\devOps\homework8>kubectl describe deployment h8-deployment  
Name:                   h8-deployment  
Namespace:              default  
CreationTimestamp:      Sat, 17 May 2025 16:04:36 +0200  
Labels:                 <none>  
Annotations:            deployment.kubernetes.io/revision: 3  
Selector:               app=h8-app  
Replicas:               5 desired | 5 updated | 5 total | 5 available | 0 unavailable  
StrategyType:           RollingUpdate  
MinReadySeconds:        0  
RollingUpdateStrategy:  1 max unavailable, 2 max surge  
Pod Template:  
  Labels:  app=h8-app  
  Containers:  
    h8-container:  
      Image:  rона03/homework8image:1.0  
      Port:   80/TCP  
      Host Port: 0/TCP  
      Environment: <none>  
      Mounts:  <none>  
      Volumes:  
        ...
```

Done 😊