

Creating docker containers

1. Create a tomcat docker container.

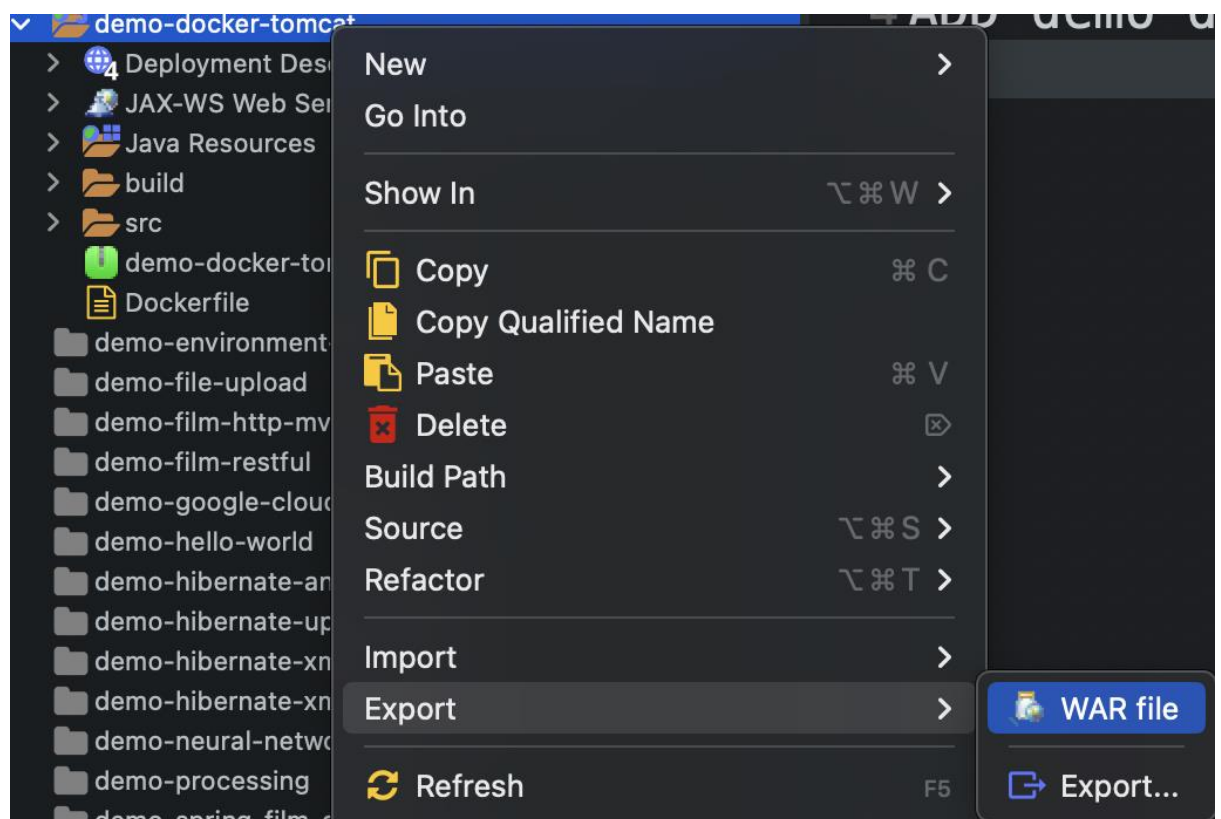
```
docker run -it --rm -p 8080:8080 tomcat:9.0
```

2. Deploy some web apps to the container, use the WAR files on moodle.

```
docker cp "local_file" container_id:"container_path"
```

Tomcat webapp folder "usr/local/tomcat/webapps"

3. Test apps locally (localhost:8080).
4. Import the HelloWorld project into eclipse.
5. Export the application as a WAR file. Right click the project->export->war file



6. Create a Dockerfile in your eclipse project. *Remember to copy over the WAR file to the “webapps” folder in the container using the “ADD” instruction. See Slide 18 for an example Dockerfile.*

7. Build a docker image using the Dockerfile.

```
docker image build -t image-name .
```

8. Using the docker image created in previous step a docker container.

```
docker container run --name container-name -it --publish 8080:8080 image-name
```

9. Test the container locally (localhost:8080/app-name)

Deploying docker containers to the cloud

1. If all is working locally, then create a new ubuntu EC2 instance and SSH into it.
2. Open port 80 on the ec2 instance for inbound connections.
3. Run the following commands to setup Docker on the ec2 instance.

Update the linux install:

```
sudo apt update
```

Download docker install bash script:

```
curl -fsSL https://get.docker.com -o get-docker.sh
```

Run script:

```
sh get-docker.sh
```

4. Copy/upload the local WAR file and Docker file to the EC2 server. Run this command

in a new terminal window:

```
scp -i pem_file.pem local_file_path ubuntu@[IP_ADDRESS]:/remote_file_path
```

5. On the server (SSH) build a new image and container using the following commands:

Build image:

```
image build -t image-name .
```

Build Container (notice we are exposing port 80 on the EC2 VM to 8080 on the docker container):

```
container run --name container_name -it --publish 80:8080 image-name
```

6. Test the to see if the app is running. ***ec2_IP/name_of_app***