

Docker demo: Static web page with AWS ECS

This demo shows how to build a [Docker container](#) serving as a webserver. The webserver hosts a [static webpage](#). Afterwards, we will push the Docker container to AWS. Finally, the Docker container used to launch an [Amazon ECS cluster](#). But we can use [Amazon SQS](#) and [Amazon EC2](#) to create a pipeline.

1. First, [\(install and\) activate Docker](#).

2. Generate the [Dockerfile](#).

```
touch Dockerfile
```

Edit the file, e.g. with [nano Dockerfile](#). An example can be found [here](#).

3. Build and tag the Docker container:

```
docker build -t hello-world .
```

4. Test your container on localhost:

```
docker run -p 80:80 hello-world
```

5. Stop your running container:

- [docker ps](#) to retrieve the container-ID
- [docker stop container-ID](#)

6. Create a AWS ECR repository via the [CLI](#):

```
aws ecr create-repository --repository-name hello-repository --region eu-central-1
```

Change [repository-name](#) and [region](#) accordingly.

7. Copy [uri](#) to your clipboard.

8. Tag the Docker container for your repository:

```
docker tag hello-world XXX.dkr.ecr.eu-west-1.amazonaws.com/hello-repository # XXX = your account-ID
```

Change the **uri** accordingly.

9. Get temporary credentials for your repository via the CLI:

```
aws ecr get-login --no-include-email --region eu-west-1
```

Copy the output to your clipboard.

10. Push the Docker container to your repository:

```
docker push XXX.dkr.ecr.eu-west-1.amazonaws.com/hello-repository # XXX  
= your account-ID
```

11. Go the ECS service. Click on **Get started** and choose **Costum container**. Set the port settings to 80.
12. After ECS deployed your application, get the **DNS-Name** of the deployed **ELB** (e.g. **EC2 console**). Paste it in to your browser tab.