# Building containers

### Building the App's Container Image

In order to build the application, we need to use a Dockerfile. A Dockerfile is simply a text-based script of instructions that is used to create a container image. If you've created Dockerfiles before, you might see a few flaws in the Dockerfile below. But, don't worry! We'll go over them.

Create a file named Dockerfile nano Dockerfile with the following contents:

FROM ubuntu  
RUN apt-get update  
RUN apt-get install -y nginx  
EXPOSE 80  
CMD nginx -g 'daemon off;'

Build the container image using the docker build command.

docker build -t windesheim .

This command used the Dockerfile to build a new container image. You might have noticed that a lot of "layers" were downloaded. This is because we instructed the builder that we wanted to start from the ubuntu image.

### Starting an App Container

Now that we have an image, let's run the application! To do so, we will use the docker run command (remember that from earlier?).

Start your container using the docker run command:

docker run -d -P windesheim

Remember the -d and -P flags? We're running the new container in "detached" mode (in the background) and creating a mapping between the host's port to the container's port.

### Pushing our Image

We need to "tag" our image, which basically means give it another name.

Login to the Docker Hub using the command docker login -u YOUR-USER-NAME.

Use the docker tag command to give the WINDESHEIM image a new name. Be sure to swap out YOUR-USER-NAME with your Docker ID.

docker tag windesheim YOUR-USER-NAME/windesheim

Now try your push command again. If you're copying the value from Docker Hub, you can drop the tagname portion, as we didn't add a tag to the image name.

docker push YOUR-USER-NAME/windesheim

### Running our Image on a New Instance

Now that our image has been built and pushed into a registry, let's try running your app on a server that has never seen this container (the server of the person next to you)!

docker run -d -P YOUR-USER-NAME/windesheim

You should see the image get pulled down and eventually start up!

You now created a new image and pushed it to the public Dockerhub. This image is available for everybody.