A basic webserver

Running arbitrary Linux commands inside a Docker container is fun, but let's do something more useful.

Pull down the `nginx` Docker image from the Docker Hub. This Docker image uses the Nginx webserver to serve a static HTML website.

Start a new container from the `nginx` image that exposes port 80 from the container to port 8000 on your host. You will need to use the `-p` flag with the docker container run command.

> NOTE: Mapping ports between your host machine and your containers can get confusing. Here is the syntax you will use:: `docker container run -p 8000:80 nginx` The trick is to remember that **the host port always goes to the left**, and **the container port always goes to the right.** Remember it as traffic coming *from* the host, *to* the container.

Open a web browser and go to port 8000 on your host. The exact address will depend on how you're running Docker today:

- **Native Linux** - http://localhost:8000
- **Cloud server** - Use the hostname (or IP) for your server. Ex: http://ec2-54-69-126-146.us-west-2.compute.amazonaws.com:8000

If you see a webpage saying "Welcome to nginx!" then you're done!

If you look at the console output from docker, you see nginx producing a line of text for each time a browser hits the webpage:

```
sofus@Praq-Sof:~/git/docker-exercises$ docker container run -p 8000:80 nginx
172.17.0.1 - - [31/May/2017:11:52:48 +0000] "GET / HTTP/1.1" 200 612 "-"
"Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:53.0) Gecko/20100101 Firefox/53.0" "-
```

Press **control + c** in your terminal window to stop your container.

Working with your docker container

When running a webserver like nginx, it's pretty useful that you do not have to have an open session into the server at all times to run it. We need to make it run in the background, freeing up our terminal for other things. Docker enables this with the `-d` parameter for run. `docker container run -p 8000:80 -d nginx`

```
sofus@Praq-Sof:~/git/docker-exercises$ docker container run -p 8000:80 -d nginx
78c943461b49584ebdf841f36d113567540ae460387bbd7b2f885343e7ad7554
```

Docker prints out the container ID and returns to the terminal.