INF105 gruppetime 12

Logg in på git.app.uib.no, vi snakker med hver prosjektgruppe i løpet av timen

Combine Gunicorn + Flask + SQLite

We'll use the sqlite version of the movie app from the lectures as a test case
https://www.inf105.org/movieapp.tgz

You'll need gunicorn, flask and sqlite on your web server
`$ doas apk add py-gunicorn py-flask sqlite w3m`

We don't want to run the gunicorn service as the main "alpine" user, it has access to too many things. Let's make a new regular user that doesn't have admin privileges:
`$ doas adduser movies`

Switch to the *movies* user with `su - movies` , then
double-check with `id` and `pwd` who you are
As the movies user, download the app, and unpack

```
$ wcurl https://www.inf105.org/movieapp.tgz
$ tar xzf movieapp.tgz
$ cd movieapp
```

Check that
`$ gunicorn mymovies:app`
starts up without errors, then use ctrl-C to stop it again. We'll instead start it in the background with

```
$ gunicorn mymovies:app --daemon
```

Now log out from the movies user with `$ exit` or `ctrl-D`.

Check id and pwd, you should be **back as the initial *alpine* user** now.

```
$ doas netstat -tulpn
```

shows that we now have a python process listening on port 8000. This is the gunicorn interface, that can be used directly for debugging on the local machine:
`$ w3m http://127.0.0.1:8000/` # navigate with tab, arrows and enter, exit with q

Now we need to make this available via our nginx web server. Here we use the one from a few weeks ago that serves the h26-...inf105.org address. Edit its config file in /etc/nginx/http.d/ to look something like below. We remove the root directive and replace it with a proxy_pass so that everything under the top URL / is forwarded to the gunicorn port at 127.0.0.1:8000. The proxy_set_header instructions make sure that gunicorn sees the external ip address that the requests come from, rather than our own nginx address:

```
server {
  listen 80;
  listen [::]:80;
  server_name h26-NNN.inf105.org;

  # root /var/www/my_second_site;

  location / {
        proxy_pass http://127.0.0.1:8000;
        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header X-Forwarded-Proto $scheme;
  }

  # Everything starting with . is a 404
  location ~ /\. {
    return 404;
  }

}
```

(this is just the most minimal way of connecting nginx to gunicorn, the gunicorn website explains many more options about caching etc.)

Now reload nginx,

```
$ doas rc-service nginx reload
```

and browse your website http://h26-....inf105.org, it should now show the movies app.

To stop / reload gunicorn, you can do

```
$ doas killall gunicorn  # stops it

$ doas killall -HUP gunicorn  # reloads config
```