

## H5

Aloitin tehtävän lataamalla virtualenv-ohjelmiston

Virtualenv on työkalu, jolla voidaan luoda eristetty ympäristö pythonia varten. Ympäristössä voidaan asentaa projektin kannalta oleellisia riippuvuuksia ilman että ne vaikuttavat koko järjestelmän toimintaan.

Asensin työkalun komennolla: `sudo apt-get install virtualenv`

Seuraavaksi loin uuden kansion virtuaaliympäristöä varten. Tämä kansio sisältää pythonin kannalta oleelliset riippuvuudet.

Loin kansion komennolla `virtualenv -system-site-packages -p python3 env/`

Kokeillaan uutta virtuaaliympäristöä komennolla `source env/bin/activate`

Nyt komentorivin vasemmassa reunassa lukee (env) josta tiedetään, että työskennellään virtuaaliympäristössä.

Takaisin "host" järjestelmään pääsee komennolla `deactivate`

```
(env) walter@waldonboksi:~/Desktop$
```

Pythoniin liittyvät asennukset tulee tehdä virtualenv:n sisällä.

pip on pythonin paketinhallinta työkalu, jolla voidaan asentaa paketteja, jotka eivät ole sisällytetty vakio asennukseen. Toisin kuin apt-get komennolla, pip komennolla on mahdollista asentaa haittaohjelmia, joten on syytä olla tarkkana.

Komennolla `which pip` nähdään missä komennon pip tiedostot sijaitsevat tiedostorakenteessa.

```
(env) walter@waldonboksi:~/Desktop$ which pip
/home/walter/Desktop/env/bin/pip
```

Seuraavaksi tein requirements.txt tiedoston, johon listasin asennettavat ohjelmat, tässä tapauksessa pelkästään django

Seuraavaksi asensin requirements.txt tiedoston komennolla

`pip install -r requirements.txt`

requirements.txt tiedostoon voi listata asennukseen tarvittavia tiedostoja ja asentaa kaikki näppärästi yhdellä komennolla.

```
(env) walter@waldonboksi:~/Desktop$ pip install -r requirements.txt
Collecting django
  Downloading Django-4.1.1-py3-none-any.whl (8.1 MB)
  ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 8.1/8.1 MB 5.0 MB/s eta 0:00:00
Collecting sqlparse>=0.2.2
  Downloading sqlparse-0.4.3-py3-none-any.whl (42 kB)
  ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 42.8/42.8 KB 3.3 MB/s eta 0:00:00
Collecting asgiref<4,>=3.5.2
  Downloading asgiref-3.5.2-py3-none-any.whl (22 kB)
Installing collected packages: sqlparse, asgiref, django
Successfully installed asgiref-3.5.2 django-4.1.1 sqlparse-0.4.3
```

Seuraavaksi tarkistin vielä djangon version komennolla `django-admin --version`

```
(env) walter@waldonboksi:~/Desktop$ django-admin --version
4.1.1
```

Starttasin projektin komennolla `django-admin startproject walter`

Menin `cd walter`

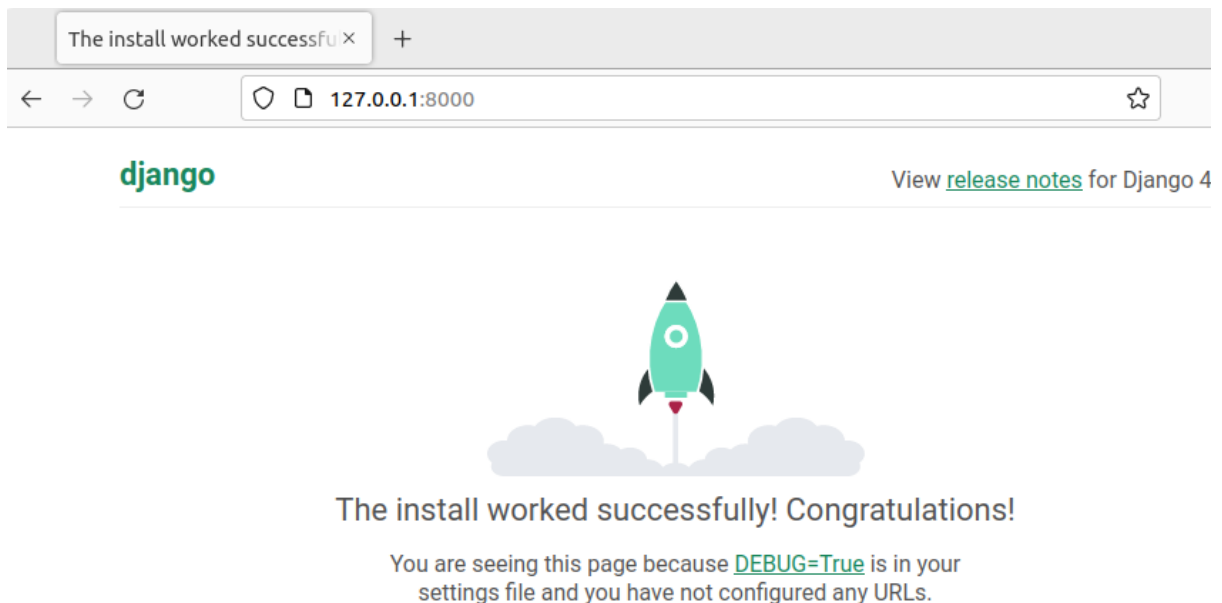
Käynnistetään palvelin komennolla `./manage.py runserver`

```
(env) walter@waldonboksi:~/Desktop$ django-admin startproject walter
(env) walter@waldonboksi:~/Desktop$ cd walter
(env) walter@waldonboksi:~/Desktop/walter$ ./manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).

You have 18 unapplied migration(s). Your project may not work properly until you apply the migrations for app(s): admin, auth, contenttypes, sessions.
Run 'python manage.py migrate' to apply them.
September 27, 2022 - 08:24:31
Django version 4.1.1, using settings 'walter.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CONTROL-C.
```

Kokeilin selaimella 127.0.0.1:8000 ja sivu aukeaa



Django ja monet muut python frameworkit vastaavat porttiin 8000

Aikaisemmin tehdyssä komennossa `./manage.py runserver` tuli herjaa puuttuvista migraatioista

Seuraavaksi ajoin komennon `python manage.py migrate`

```
(env) walter@waldonboksi:~/Desktop/walter$ python manage.py migrate
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, sessions
Running migrations:
  Applying contenttypes.0001_initial... OK
  Applying auth.0001_initial... OK
  Applying admin.0001_initial... OK
  Applying admin.0002_logentry_remove_auto_add... OK
  Applying admin.0003_logentry_add_action_flag_choices... OK
  Applying contenttypes.0002_remove_content_type_name... OK
  Applying auth.0002_alter_permission_name_max_length... OK
  Applying auth.0003_alter_user_email_max_length... OK
  Applying auth.0004_alter_user_username_opts... OK
  Applying auth.0005_alter_user_last_login_null... OK
  Applying auth.0006_require_contenttypes_0002... OK
  Applying auth.0007_alter_validators_add_error_messages... OK
  Applying auth.0008_alter_user_username_max_length... OK
  Applying auth.0009_alter_user_last_name_max_length... OK
  Applying auth.0010_alter_group_name_max_length... OK
  Applying auth.0011_update_proxy_permissions... OK
  Applying auth.0012_alter_user_first_name_max_length... OK
  Applying sessions.0001_initial... OK
```

## Admin interface

Lisätään käyttäjä Admin-interfacea varten

Asensin ensiksi ohjelman pwgen, jolla voi generoida vahvoja salasanoja


Asensin ohjelman komennolla `sudo apt-get install pwgen`

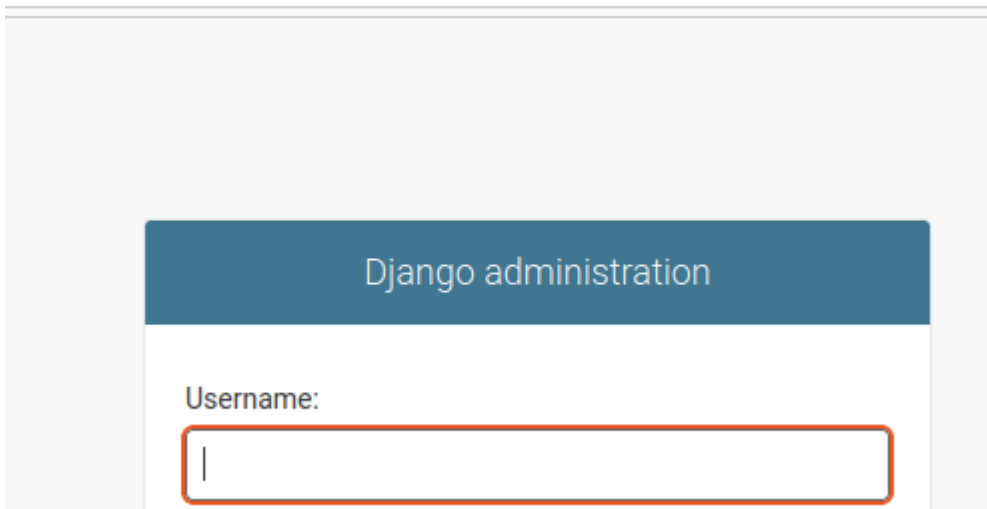
Generoin vahvan salasanan komennolla `pwgen -s 20 1 # randomize a password`

Seuraavaksi tein superuserin komennolla `./manage.py createsuperuser`

Käynnistin palvelimen komennolla `./manage.py runserver`

Menin selaimella osoitteeseen `127.0.0.1:8000/admin`

 `127.0.0.1:8000/admin/login/?next=/admin/`



Seuraavaksi kirjauduin sisään luomallani admin tunnuksilla ja käytin admin interfacea luodakseni uuden käyttäjän. Lisäsin TaunoTestaajalle oikeuksia sisällön muokkaamiseen. Lisäsin myös katselu oikeudet ryhmiin ja lokeihin.

<input type="checkbox"/>	USERNAME	EMAIL ADDRESS	FIRST NAME	LAST NAME	STAFF STATUS
<input type="checkbox"/>	TaunoTestaaja				✓
<input type="checkbox"/>	wadmin				✓

Tämän jälkeen kirjauduin TaunoTestaajalla sisään, kokeilin että pääsin kirjautumaan ja oikeudet toimivat.

Tämän jälkeen loin tietokannan komennolla `./manage.py startapp crm`

Tämä komento loi uuden kansion nimellä crm

Tämän jälkeen lisäsin settings.py tiedostoon INSTALLED\_APPS kohtaan juuri luomani crm tiedoston.

```
INSTALLED_APPS = [  
    'django.contrib.admin',  
    'django.contrib.auth',  
    'django.contrib.contenttypes',  
    'django.contrib.sessions',  
    'django.contrib.messages',  
    'django.contrib.staticfiles',  
    'crm',  
]
```

Seuraavaksi lisäsin malleja tiedostoon crm/models.py

```
1 from django.db import models  
2 class Customer(models.Model):  
3     name = models.CharField(max_length=300)  
4 # Create your models here.  
5
```

tämän jälkeen vedin komennon `./manage.py makemigrations` ja `./manage.py migrate`

Seuraavaksi rekisteröin tietokannan muokkaamalla tiedostoa crm/admin.py

```
1 from django.contrib import admin  
2 from . import models  
3 admin.site.register(models.Customer)  
4 # Register your models here.  
5
```

Seuraavaksi laitoin palvelimen päälle ja kokeilin osoitteesta 127.0.0.1:8000/admin

Nyt admin sivulla näkyy CRM customer tietokanta

CRM	
Customers	<a href="#">+ Add</a>

Action: 

<input type="checkbox"/>	CUSTOMER
<input type="checkbox"/>	Customer object (2)
<input type="checkbox"/>	Customer object (1)

Seuraavaksi muokkasin vielä hieman crm/models.py tiedostoa, jotta asiakkaiden nimet näkyvät niminä, eikä customer object (1) muodossa.

Lisätään muutama rivi models.py tiedostoon.

```
from django.db import models
class Customer(models.Model):
    name = models.CharField(max_length=300)
    def __str__(self):
        return self.name
# Create your models here.
```

Tämän jälkeen kokeilin vielä admin interfacea ja nyt asiakkaiden nimet näkyvät kunnolla.

Users	<a href="#">+ Add</a>
CRM	
Customers	<a href="#">+ Add</a>

<input type="checkbox"/>	CUSTOMER
<input type="checkbox"/>	testi2
<input type="checkbox"/>	testi1

2 customers

Tehtävä valmis

Lähteet:

<https://terokarvinen.com/2022/django-instant-crm-tutorial/>

<https://www.freecodecamp.org/news/python-virtual-environments-explained-with-examples/>

<https://learnpython.com/blog/python-requirements-file/>

<https://www.dabapps.com/blog/introduction-to-pip-and-virtualenv-python/>