

HARJOITUSTYÖN 1. VAIHE: KEHITYSYMPÄRISTÖN PERUSTAMINEN

YMPÄRISTÖN VALINTA JA ASENNUS

Valitsin kehitysympäristöksi Django kurssin suosituksesta, sillä aikaisempaa kokemusta ei löydy vastaavien sovellusten käytöstä ja ohjelmointiakin on vain yksi johdatuskurssi alla. Käytän itse MacBookia, joten asennuksessa noudatin Echo-videotallenteen ohjeita Django asennukseen macOS käyttöjärjestelmälle.

Ensimmäisenä päivitin netistä Python.org nettisivuilta uusimman Pythonin version koneelleni. Tämän jälkeen asensin Homebrewn macin Terminaalin kautta komennolla `/usr/bin/ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"`. Asennus onnistui heti ensimmäisellä kerralla ja pääsin jatkamaan asennusprosessia.

Seuraavaksi asennusvideon ohjeiden avulla syötin komennon ***pip freeze*** joka tuotti ensimmäisen ristiriidan. Terminaalin mukaan tällaista komentoa ei löytynyt, joten etsin Googlen avulla ratkaisun kyseiseen ongelmaan. Stackoverflow (2012) nettisivun ohjeiden mukaan käytin komentoa ***sudo easy_install pip***, jolloin sain pip:in asennettua. Tämän jälkeen komento ***pip freeze*** toimi kuten pitikin.

Kuva 1: Pip:in asennus.

```
Nea-MacBook:~ neamantere$ pip freeze
-bash: pip: command not found
Nea-MacBook:~ neamantere$ sudo easy_install pip
Password:
Searching for pip
Best match: pip 20.0.2
Adding pip 20.0.2 to easy-install.pth file
Installing pip script to /usr/local/bin
Installing pip3.8 script to /usr/local/bin
Installing pip3 script to /usr/local/bin
```

```
Using /Users/neamantere/Library/Python/2.7/lib/python/site-packages
Processing dependencies for pip
Finished processing dependencies for pip
```

Pip:in asennuksen jälkeen pääsin itse Django asennukseen. Ensin latasin virtualenvin komennolla ***pip3 install virtualenv***. Onnistuneen asennuksen jälkeen loin Terminaalissa kansion nimeltä *Ohjelmallinen sisällönhallinta* (myöhemmin kansion nimeksi muutettu *Ohsiha*) harjoitustyölle

komennolla **mkdir Ohjelmallinen sisällönhallinta**. Tämän jälkeen siirryin *Ohsiha*-kansioon komennolla **cd Ohjelmallinen sisällönhallinta** ja loin kansion sisälle uuden kansion itse harjoitustyölle nimeltä *Harkka* komennolla **mkdir Harkka**. Seuraavaksi siirryin **cd** komennolla *Harkka*-kansion sisälle ja loin virtuaalisen ympäristön.

Kuva 2: Virtuaalisen ympäristön luonti.

```
[Nea-MacBook:ohjelmallinen neamantere$ cd Harkka
Nea-MacBook:Harkka neamantere$ virtualenv venv
created virtual environment in 756ms CPython3Posix(dest=/Users/neamantere/Ohjelmallinen/Harkka/venv, clear=False, global=False) with seeder FromAppData pip=latest setuptools=latest wheel=latest app_data_dir=/Users/neamantere/Library/Application Support/virtualenv/seed-v1 via=copy]
```

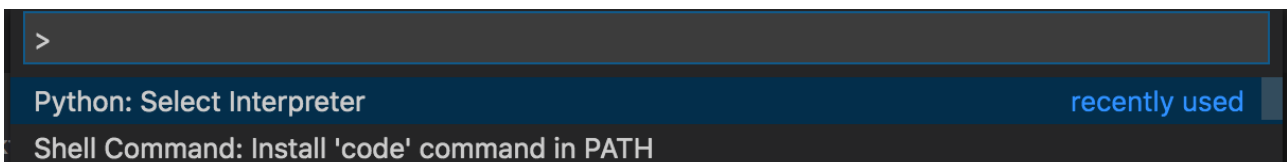
Virtuaalisen ympäristön luonnin jälkeen liikuin terminaalissa komennolla **cd venv venv**-kansioon, jossa aktivoin virtuaalisen ympäristön komennolla **source bin/activate**. Tämän jälkeen itse Django lataus tapahtui komennolla **pip3 install django**.

Kuva 3: Django asennus virtuaaliseen ympäristöön.

```
[(venv) Nea-MacBook:venv neamantere$ pip3 install django
Collecting django
  Using cached Django-3.0.3-py3-none-any.whl (7.5 MB)
Collecting asgiref~=3.2
  Using cached asgiref-3.2.3-py2.py3-none-any.whl (18 kB)
Collecting sqlparse>=0.2.2
  Using cached sqlparse-0.3.0-py2.py3-none-any.whl (39 kB)
Collecting pytz
  Using cached pytz-2019.3-py2.py3-none-any.whl (509 kB)
Installing collected packages: asgiref, sqlparse, pytz, django
Successfully installed asgiref-3.2.3 django-3.0.3 pytz-2019.3 sqlparse-0.3.0]
```

Tämän jälkeen asensin virtuaaliseen ympäristöön *pylint*-lisäosan Pythonille komennolla **pip3 install pylint** sekä siirryin yhden kansiorakenteen taaksepäin komennolla **cd ..**. Seuraavaksi käynnistin Django-projektin komennolla **django-admin startproject projekti**. Projektin aloituksen jälkeen latsin Visual Studio Code:in ohjelman nettisivuilta sekä asensin *code*-komennon Terminaaliin.

Kuva 4: Code-komennon asentaminen.



Code-komennon asentamisen jälkeen latsin Python-lisäosan Visual Studio Code:en ja liikuin **cd** komennolla *projekti*-kansioon. Seuraavaksi annoin komennon **python3 manage.py runserver**, joka ei kuitenkaan toiminut oletetulla tavalla.

Kuva 5: *manage.py*:n käynnistämisen virhe.

```
(venv) Nea-MacBook:projekti neamantere$ python3 manage.py runserver
Watching for file changes with StatReloader
Performing system checks...
```

```
System check identified no issues (0 silenced).
```

```
You have 17 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.
```

```
February 16, 2020 - 10:46:46
Django version 3.0.3, using settings 'Projekti.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CONTROL-C.
```

Ongelma ratkesi loppu viimein komennolla ***python3 manage.py migrate***. Tämän jälkeen käynnistäminen onnistui komennolla ***python3 manage.py runserver***, jonka jälkeen laitoin selaimeen serverin osoitteen ***http://127.0.0.1:8000/*** nähdäkseni, toimiiko serveri oikein.

TIETOKANNAN TESTAAMINEN JA TESTISOVELLUKSEN LUOMINEN

Tietokannan testaaminen ja testisovelluksen luominen onnistui koodiklinikoiden sekä Django:n dokumentaatio-ohjeiden avulla. Aivan ensiksi loin applikaation Terminaaliin komennolla ***python3 manage.py startapp applikaatio***. Tämän jälkeen avasin komennolla ***code*** . Visual Studio Code:n ja loin *settings.py* tiedostoon tiedon uudesta applikaatiosta.

Kuva 6: Applikaation lisääminen *settings.py* –tiedostoon.

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

Tämän jälkeen lisäsin Projektin *urls.py* –tiedostoon sovelluksen polun.

Kuva 7: Projektin *urls.py* –tiedosto.

```
from django.contrib import admin
from django.urls import path, include

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', include('applikaatio.urls'))
]
```

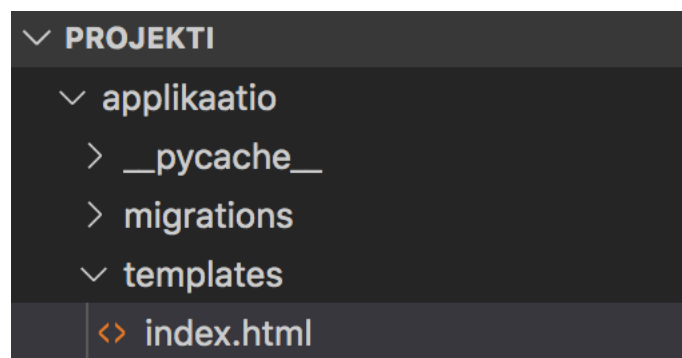
Seuraavaksi applikaation *urls.py* –tiedostoon haettiin *views* ja *path* kirjastot ja luotiin vastaavanlainen tietorakenne kuin Projektin *urls.py* –tiedostossa.

Kuva 8: Applikaation *urls.py* –tiedosto.

```
from django.urls import path
from . import views

urlpatterns = [
    path('', views.HomePageView, name='kotisivu')
]
```

Tämän jälkeen loin applikaation kansion alle *templates* –kansion, jonka sisälle loin *index* nimisen html-tiedoston.

Kuva 9: *Index.html* –tiedoston sijainti.

Html-tiedoston luomisen jälkeen muokattiin *views.py* –tiedostoa, jotta se pääsee käsiksi *index.html* –tiedostoon.

Kuva 10: *views.py* –tiedosto.

```
from django.shortcuts import render

# Create your views here.

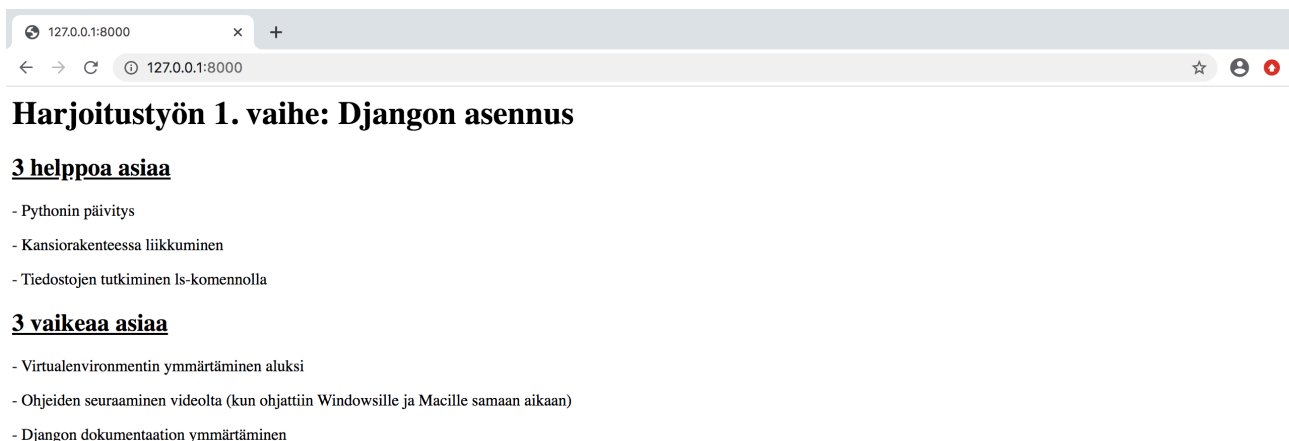
def HomePageView(request):
    return render(request, 'index.html', {})
```

Seuraavaksi loin *index.html* –tiedoston sisällön. Kirjoitin tiedostoon testimielessä tehtävänannossa vaaditut 3 helppoa ja 3 vaikeaa asiaa. Osoite <http://127.0.0.1:8000/>.

Kuva 11: Html-tiedoston sisältö.

```
<html>
  <body>
    <h1>Harjoitustyön 1. vaihe: Django asennus</h1>
    <h2><u>3 helppoa asiaa </u></h2>
    <p>- Pythonin päivitys</p>
    <p>- Kansiorakenteessa liikkuminen</p>
    <p>- Tiedostojen tutkiminen ls-komennolla</p>
    <h2><u>3 vaikeaa asiaa </u></h2>
    <p>- Virtualenvin ymmärtäminen aluksi</p>
    <p>- Ohjeiden seuraaminen videolta (kun ohjattiin Windowsille ja Macille samaan aikaan)</p>
    <p>- Django dokumentaation ymmärtäminen</p>
  </body>
</html>
```

Kuva 12: Selainnäky.



Seuraavaksi tein muutokset *models.py* –tiedostoon testejä varten.

Kuva 13: *models.py* –tiedosto.

```
from django.db import models

class Question(models.Model):
    question_text = models.CharField(max_length=200)
    pub_date = models.DateTimeField('date published')
    def __str__(self):
        return self.question_text

class Choice(models.Model):
    question = models.ForeignKey(Question, on_delete=models.CASCADE)
    choice_text = models.CharField(max_length=200)
    votes = models.IntegerField(default=0)
    def __str__(self):
        return self.choice_text
```

Lopuksi tallensin muutokset Terminaalin kautta tietokantaan komennoilla ***python3 manage.py makemigrations*** ja ***python3 manage.py migrate***. Viimeisenä testasin Django:n dokumentaatioiden avulla SQLite-tietokantaa Terminaalin kautta.

Kuva 14: Terminaalitestit.

```

^C(venv) Nea-MacBook:Projekti neamantere$ python3 manage.py makemigrations
Migrations for 'applikaatio':
  applikaatio/migrations/0001_initial.py
    - Create model Question
    - Create model Choice
(venv) Nea-MacBook:Projekti neamantere$ python3 manage.py migrate
Operations to perform:
  Apply all migrations: admin, applikaatio, auth, contenttypes, sessions
Running migrations:
  Applying applikaatio.0001_initial... OK
(venv) Nea-MacBook:Projekti neamantere$ python3 manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
February 21, 2020 - 18:03:40
Django version 3.0.3, using settings 'Projekti.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CONTROL-C.
[21/Feb/2020 18:03:44] "GET / HTTP/1.1" 200 88
^C(venv) Nea-MacBook:Projekti neamantere$ python3 manage.py shell
Python 3.8.1 (v3.8.1:1b293b6006, Dec 18 2019, 14:08:53)
[Clang 6.0 (clang-600.0.57)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
(InteractiveConsole)
>>> from applikaatio.models import Choice, Question
>>> Question.objects.all()
<QuerySet []>
>>> from django.utils import timezone
>>> q = Question(question_text="Miten menee?", pub_date=timezone.now())
>>> q.save()
>>> q.id
1
>>> q.question_text
'Miten menee?'
>>> q.pub_date
datetime.datetime(2020, 2, 21, 18, 10, 36, 232034, tzinfo=<UTC>)
>>> q.question_text = "Onko leuka lattiassa?"
>>> q.save()
>>> Question.objects.all()
<QuerySet [<Question: Question object (1)>]>
>>> q.question_text
'Onko leuka lattiassa?'
>>> q.id
1
>>> exit()
(venv) Nea-MacBook:Projekti neamantere$ █

```

LÄHTEET

Django. *Writing your first Django app, part 2*. Saatavilla [www-osoitteessa](https://docs.djangoproject.com/en/3.0/intro/tutorial02/):
<<https://docs.djangoproject.com/en/3.0/intro/tutorial02/>> (Luettu 21.2.2020)

Echon luentotallenteet: *Djangon asentaminen macOS ja Django*
(*KATSO KUVAUS ENNEN KATSOMISTA*), saatavilla [www-osoitteessa](https://echo360.org.uk/section/b8122fc9-4500-40aa-b302-d3ccddff1377/home) (rajattu pääsy):
<<https://echo360.org.uk/section/b8122fc9-4500-40aa-b302-d3ccddff1377/home>>

Python.org. *Download the latest version of Python*. Saatavilla [www-osoitteessa](https://www.python.org/downloads/):
<<https://www.python.org/downloads/>> (Luettu 16.2.2020)

Stackoverflow, 2012. *Bas: pip: command not found*. Saatavilla [www-osoitteessa](https://stackoverflow.com/questions/9780717/bash-pip-command-not-found):
<<https://stackoverflow.com/questions/9780717/bash-pip-command-not-found>> (Luettu 16.2.2020)

Visual Studio Code. *Code editing. Redefined*. Saatavilla [www-osoitteessa](https://code.visualstudio.com):
<<https://code.visualstudio.com>> (Luettu 16.2.2020)