

PYTHON django- install Django

1. Pengertian PIP

pip singkatan dari Pip Installs Python atau Pip Installs Package. Merupakan program untuk manajemen paket di Python. Tugasnya untuk menginstal, menghapus, upgrade paket Python, dll.

Mengecek Modul dengan pip

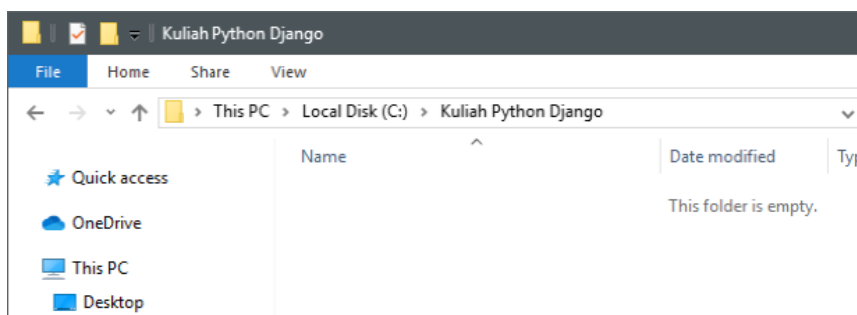
1. Buka command prompt Anda. Dengan mengetik **cmd** pada kotak searching Windows.
2. Setelah Command Prompt terbuka ketik **pip list**. Akan terlihat **modul-modul global**.
3. Apabila muncul keterangan DEPRECATION. Contoh seperti gambar di bawah. Artinya pip harus di-update.
4. Untuk meng-update pip. Pada command prompt, silahkan ketik : **python -m pip install --upgrade pip**
5. Contoh pip yang berhasil diupdate.

2. Virtual Environment

Adalah tools untuk membuat lingkungan python virtual yang terisolasi yang artinya tertutup dan tidak bias diakses dari luar.

Instalasi Virtualenv dan Django

1. Buat folder baru (**Kuliah Python Django**) untuk lokasi Virtualenv. Contoh seperti gambar di bawah.



2. Buka Command Prompt. Ini adalah perintah cmd yang perlu diingat. Untuk instalasi **Virtualenv** dan menggunakan **Django Framework** kita selalu gunakan **command prompt**.
 - **dir** untuk mengecek directory yang ada pada folder tersebut.
 - **cd** untuk berpindah ke suatu directory. **cd..** untuk kembali ke folder sebelumnya. **cd nama folder** untuk masuk ke dalam directory
3. Posisi sudah di dalam directory **Kuliah Python Django**.
4. Ketik **python -m venv Env**
5. Cek Windows Explorer. Sudah terbuat folder **Env**.
6. Kemudian kita akan mengakses file **activate** di dalam folder **Scripts**. Untuk masuk ke **Virtual Environment**.
7. Kembali ke cmd, tambahkan **Env\Scripts\activate.bat** maka kita sudah masuk ke dalam **Virtualenv**
8. Cek **modul** kembali dengan mengetik **pip list**. Terlihat bahwa **modul-modul di dalam Virtualenv** berbeda dengan **modul-modul global**.
9. Update kembali **pip** di dalam **Virtualenv**. Ketik kembali **python -m pip install --upgrade pip**. Gambar di bawah menunjukkan **pip** berhasil di-update.

10. Cek **modul** kembali dengan mengetik **pip list**.

11. Selanjutnya instal **Django** di dalam **Virtualenv**. Ketik **pip install Django=**

12. Cek **modul** kembali dengan mengetik **pip list**. Paket/modul **Django** sudah terinstal di dalam **Virtualenv**.

13. Untuk keluar dari **Virtualenv**, ketik **Env\Scripts\deactivate.bat**

MODUL 1

Membuat project

1. Langkah-Langkah

Membuat project di **Django** masih dan selalu gunakan **cmd**. Ikuti perintah di bawah ini untuk membuat project dan menjalankan server website.

2. Apabila cmd ditutup / dibersihkan/ di-clear, maka server website akan berhenti.

```
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

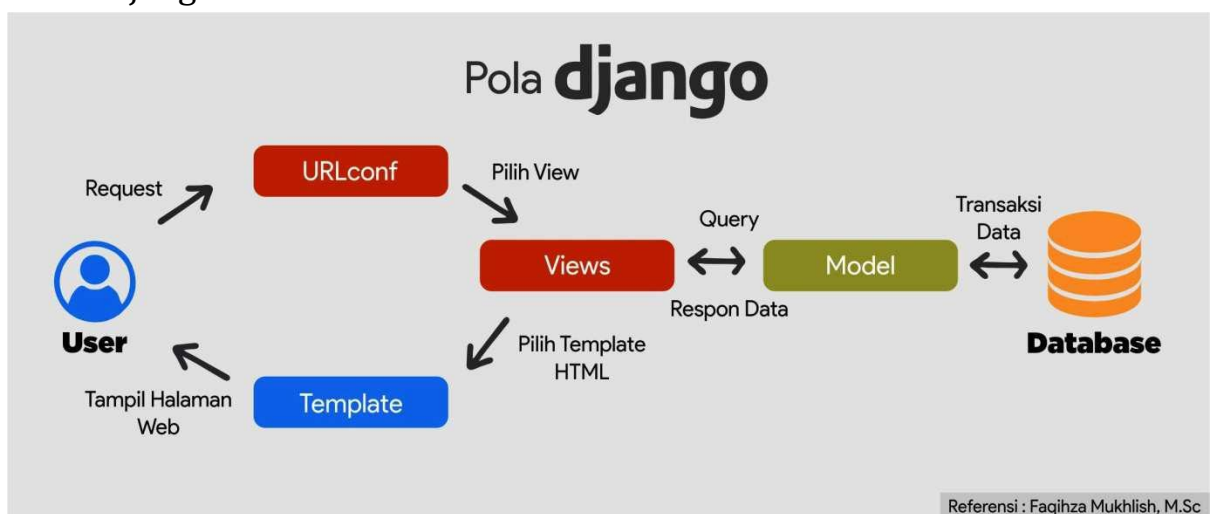
C:\Users\kevin>
```

3. Apabila stop, jalankan kembali. **Biarkan cmd tetap terbuka** selama proses development.

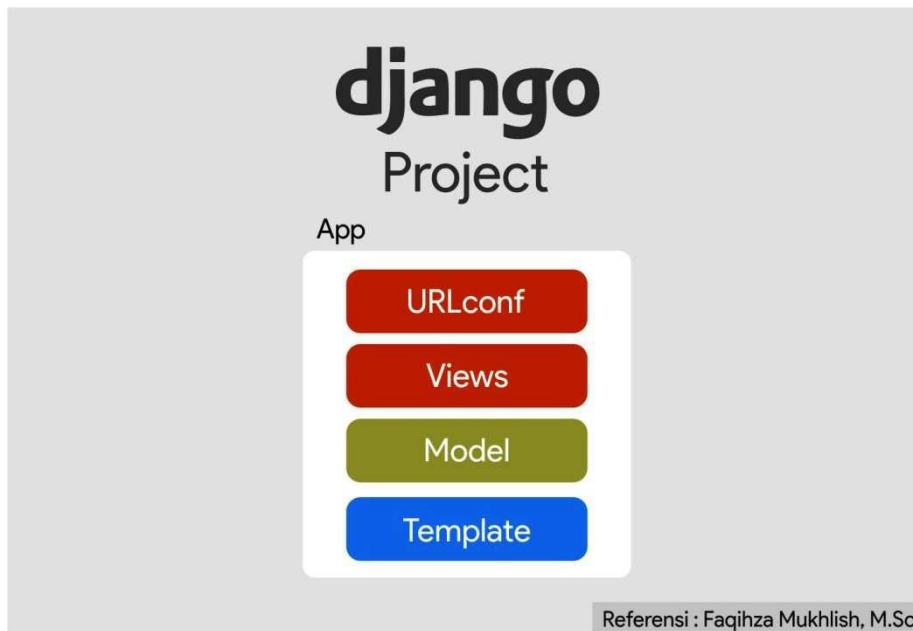
MODUL 2

Pola Django Framework

1. POLA Django



2. Struktur App Django



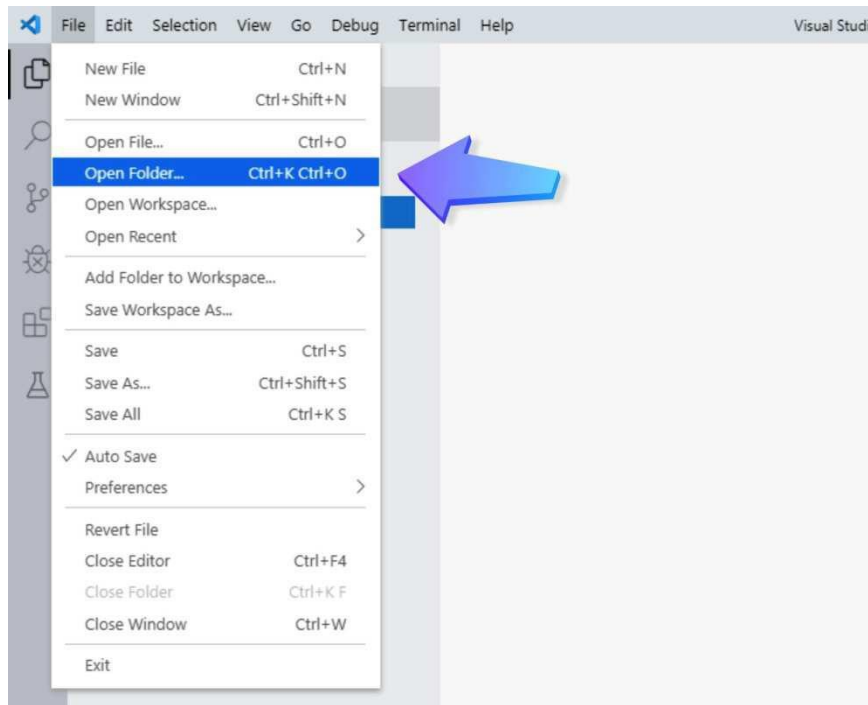
URL, Views, dan Template

1. Jalankan Server

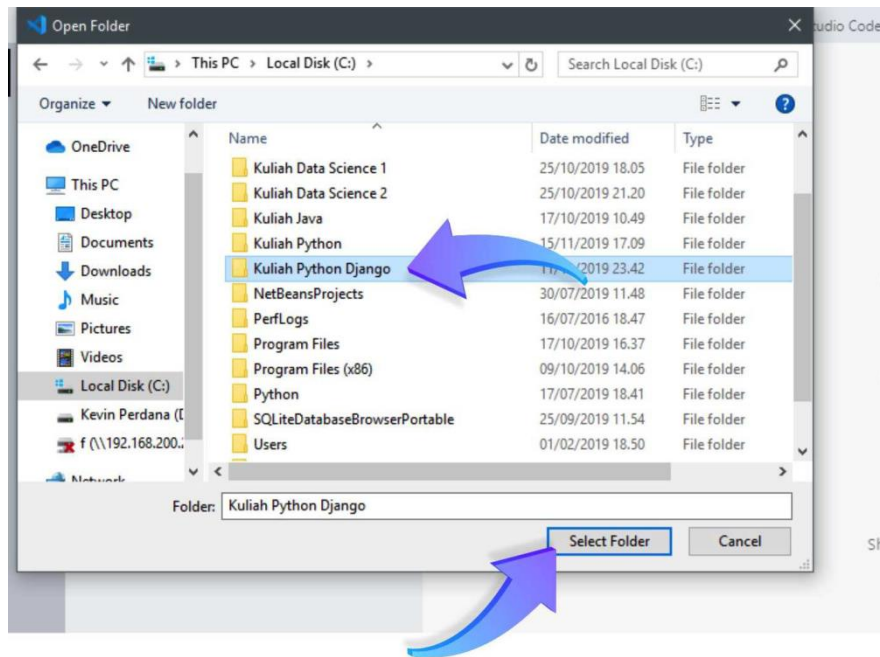
Jalankan terlebih dahulu server website menggunakan **cmd** seperti di tutorial sebelumnya.

2. Buka Project ke dalam VS Code

1. Pada Menu, klik **File Open Folder**.



2. Cari Folder **Virtual Env** yang sudah dibuat sebelumnya, dan klik **Select Folder**.



URL, Views, dan Template untuk Halaman Utama

1. Pada project (websiteku) klik file **urls** kemudian ketik kode seperti gambar dibawah.
2. Klik kanan **project directory**, klik **New File**
3. Ketik **views.py**
4. Klik kanan pada **Base Directory**, kemudian **New Folder**.
5. Buat folder baru dengan nama **templates**.
6. Klik file **settings.py** pada project. Dan ketik kode seperti gambar di bawah.
7. Klik file **settings.py** pada project. Dan ketik kode seperti gambar di bawah.
8. Buat file baru dengan nama **index.html** di dalam folder **templates project**.
9. Ketik kode **index.html** seperti gambar di bawah.
10. Halaman Utama selesai. Cek di browser Anda.

URL, Views, dan Template untuk App

1. Buka **cmd baru** atau **terminal** (Klik View>Terminal) yang ada di **VS Code** Anda, kemudian ketik **manage.py startapp about** seperti gambar di bawah. Jangan lupa masuk ke dalam **Virtual Environment** dan masuk ke **Base Directory** terlebih dahulu
2. Apabila pembuatan App berhasil, hasilnya seperti gambar di bawah.
3. Tambahkan kode pada file **urls** yang ada di dalam **Project Directory** seperti gambar di bawah.
4. Klik kanan pada App **about**. Kemudian **New File**.
5. Tambahkan file **urls.py** pada **about**
6. Ketik kode pada file **urls.py** seperti gambar di bawah.
7. Buka **settings.py** pada Project Directory dan sesuaikan dengan nama **App yang telah dibuat** seperti gambar dibawah.
8. Klik kanan pada **about**, kemudian **New Folder**.
9. Buat folder **templates** di dalam App **about**.

- 10.. Tambahkan folder lagi di dalam **templates** dengan nama **about** (sesuai dengan nama App).
11. Klik kanan pada folder **about** kemudian **New File**.
12. Buat file dengan nama **index.html**
13. Ketik kode **index.html** seperti gambar di bawah.
14. Ketik kode pada file **views.py** di dalam folder App **about**.
15. Tambahkan satu baris kode pada **index.html** di dalam folder **templates** utama Ikuti gambar di bawah.
16. Tampilan Halaman Utama website.
17. Tampilan Halaman App about.

MODUL 3

Mengatur Static Files (Gambar dan CSS)

1. Melanjutkan project yang telah dijelaskan di modul sebelumnya. Pada materi ini dibahas cara memasukkan gambar dan file CSS, pada Django ini disebut dengan "Static Files".
 - Buat folder baru dengan nama **static** di dalam **Base Directory**.
 - Buka file **settings.py** dan tambahkan kode seperti gambar di bawah.
 - Buat folder baru dengan nama **img** di dalam folder **static**.
 - Copy gambar yang akan dimasukkan ke Halaman Utama website.
 - Buat folder **static, about, dan img** di dalam app **about**.
 - Copy gambar yang akan dimasukkan ke Halaman About.
 - Jalankan server website.
 - Tes URL gambar.
 - Sesuaikan kode **HTML Halaman Utama** seperti gambar di bawah.
 - Sesuaikan kode **HTML Halaman About** seperti gambar di bawah.
 - Buat folder baru dengan nama **css** di dalam folder **static utama**.
 - Buat file baru dengan nama **styles.css** di dalam folder **css** yang telah dibuat.
 - Tulis kode **styles.css** seperti gambar di bawah.
 - Sesuaikan kode **index.html Halaman Utama** seperti gambar di bawah.

MODUL 4

Integrasi Bootstrap

1. Melanjutkan project yang telah dijelaskan di modul sebelumnya. Pada materi ini dibahas cara integrasi Bootstrap ke dalam Django.
 - Buat folder baru dengan nama **vendor** dan **bootstrap** di dalam vendor.
 - Buka folder **bootstrap-4.3.1-dist** dan di dalam folder **css**, copy **bootstrap.css**. Dan paste ke dalam folder **bootstrap** yang telah dibuat. Ikuti gambar di bawah.
 - Buka folder **bootstrap-4.3.1-dist** dan di dalam folder **js**, copy beberapa file javascript (ikuti gambar dibawah). Dan paste ke dalam folder **bootstrap** yang telah dibuat.
 - Cek di dalam Visual Studio Code.
 - Buka file **index.html** untuk Halaman Utama.

- Sesuaikan kode **index.html**. Copy dan paste kode dibawah

MODUL 5

Django Admin

1. Membuat Superuser

Untuk masuk ke situs admin, diperlukan akun pengguna dengan status **Staf diaktifkan**, untuk melihat dan membuat record. Pengguna ini memiliki **izin untuk mengelola semua objek**.

- Buat **cmd**, kemudian ketik **manage.py createsuperuser** Isikan **username**, **email**, dan **password** seperti gambar dibawah.
- Cek pada **database** dan **tabel auth_user** untuk memastikan data user telah masuk.jalankan server
- Buka halaman **Django Admin** di url **http://127.0.0.1:8000/admin**.
- Isi **username** dan **password** superuser yang telah dibuat tadi.
- Tampilan Django Admin.
-

