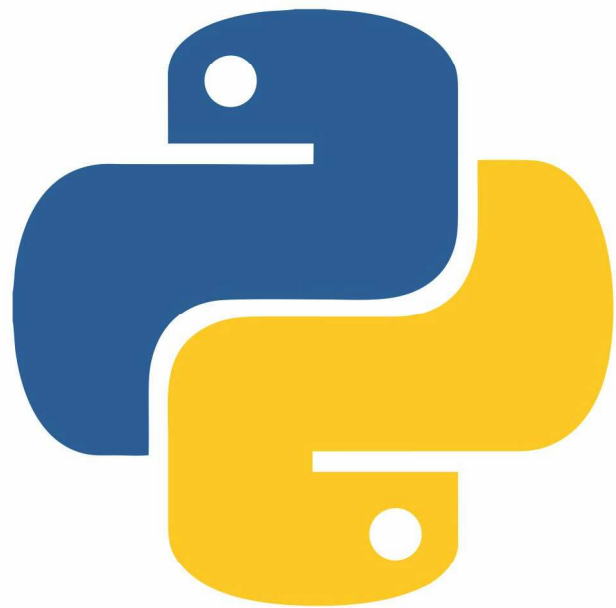


PYTHON

django

web framework

by Kevin Perdana



Integrasi Database MySQL



8.1 Instal Modul mysqlclient

Modul mysqlclient digunakan sebagai interface Python ke Database MySQL. Langkah-langkah instalasi sebagai berikut :

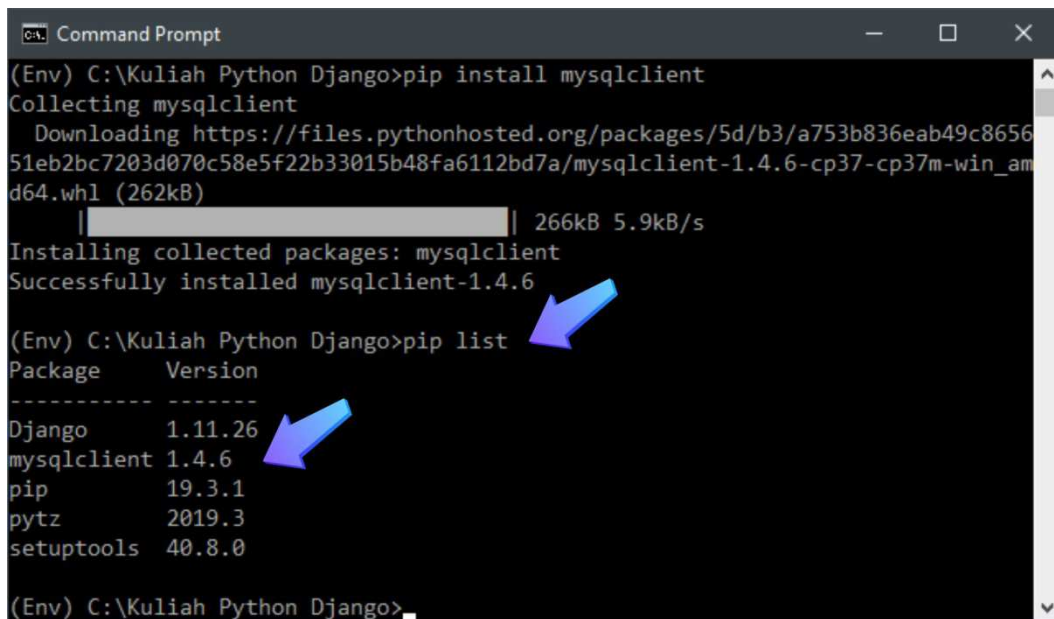
1. Buat **cmd**, kemudian masuk ke dalam **Virtualenv**. Setelah itu ketik **pip list** untuk mengecek daftar modul. Selanjutnya ketik **pip install mysqlclient**, seperti gambar di bawah.

A screenshot of a Windows Command Prompt window. The title bar says 'C:\ Command Prompt'. The command prompt shows the following sequence of commands and output:
C:\Kuliah Python Django>Env\Scripts\activate
(Env) C:\Kuliah Python Django>pip list
Package Version

Django 1.11.26
pip 19.3.1
pytz 2019.3
setuptools 40.8.0
(Env) C:\Kuliah Python Django>pip install mysqlclient
Collecting mysqlclient
Downloading https://files.pythonhosted.org/packages/5d/b3/a753b836eab49c865651eb2bc7203d070c58e5f22b33015b48fa6112bd7a/mysqlclient-1.4.6-cp37-cp37m-win_amd64.whl (262kB)
| 266kB 5.9kB/s
Installing collected packages: mysqlclient
Successfully installed mysqlclient-1.4.6
A blue arrow points to the command 'pip install mysqlclient'.

Gambar 85 Instal mysqlclient

2. Apabila berhasil, ketik **pip list** kembali untuk memastikan modul **mysqlclient** telah terinstal.



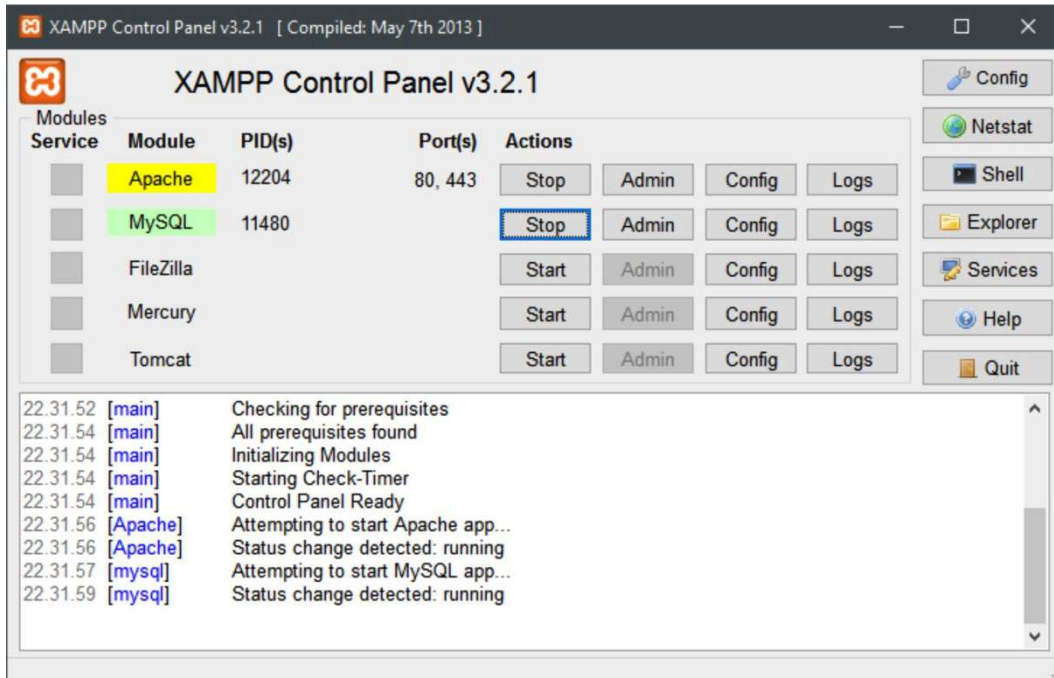
```
(Env) C:\Kuliah Python Django>pip install mysqlclient
Collecting mysqlclient
  Downloading https://files.pythonhosted.org/packages/5d/b3/a753b836eab49c865651eb2bc7203d070c58e5f22b33015b48fa6112bd7a/mysqlclient-1.4.6-cp37-cp37m-win_amd64.whl (262kB)
    |#####| 266kB 5.9kB/s
Installing collected packages: mysqlclient
Successfully installed mysqlclient-1.4.6

(Env) C:\Kuliah Python Django>pip list
Package      Version
-----
Django       1.11.26
mysqlclient  1.4.6
pip          19.3.1
pytz         2019.3
setuptools   40.8.0

(Env) C:\Kuliah Python Django>
```

Gambar 86 mysqlclient Berhasil Diinstal

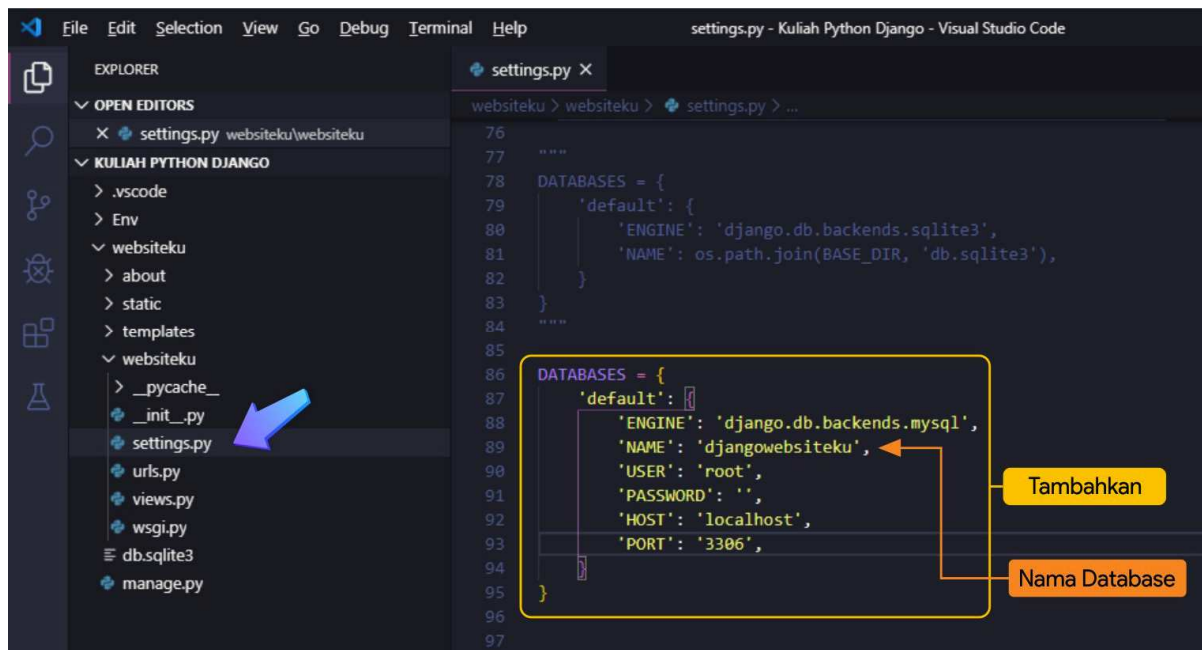
3. Buka **xampp** Anda, aktifkan **Apache** dan **MySQL**.



Gambar 87 Aktifkan Apache dan MySQL

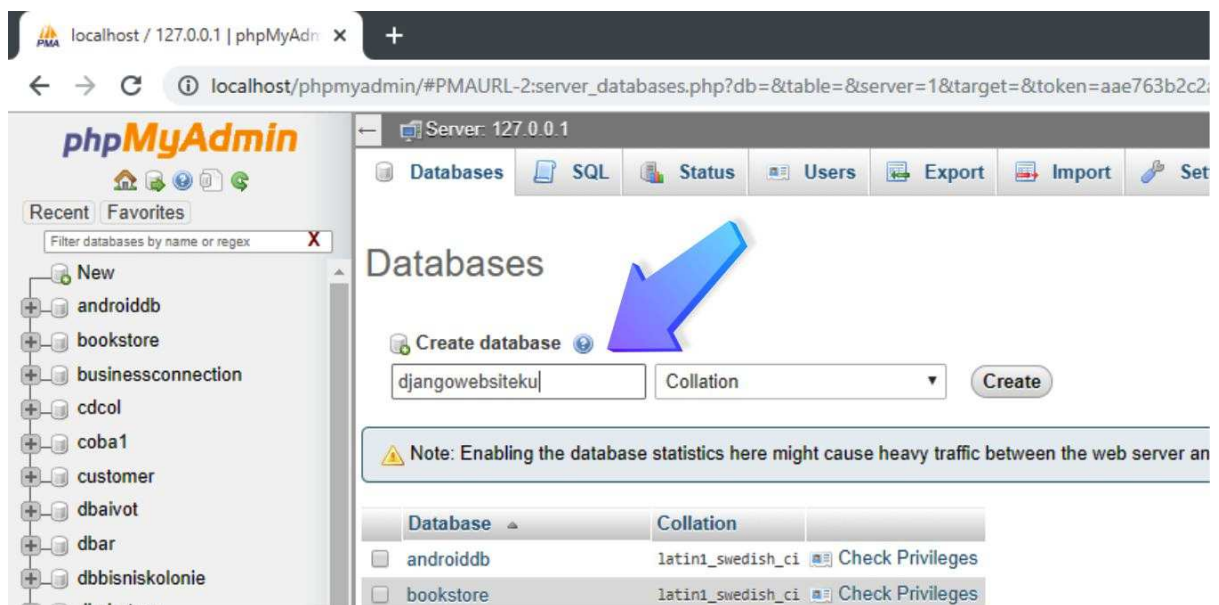
PYTHON **django** web framework

4. Buka project Anda di **Visual Studio Code**, buka file **settings.py** di dalam folder project **websiteku**. Kemudian tambahkan kode seperti gambar dibawah.



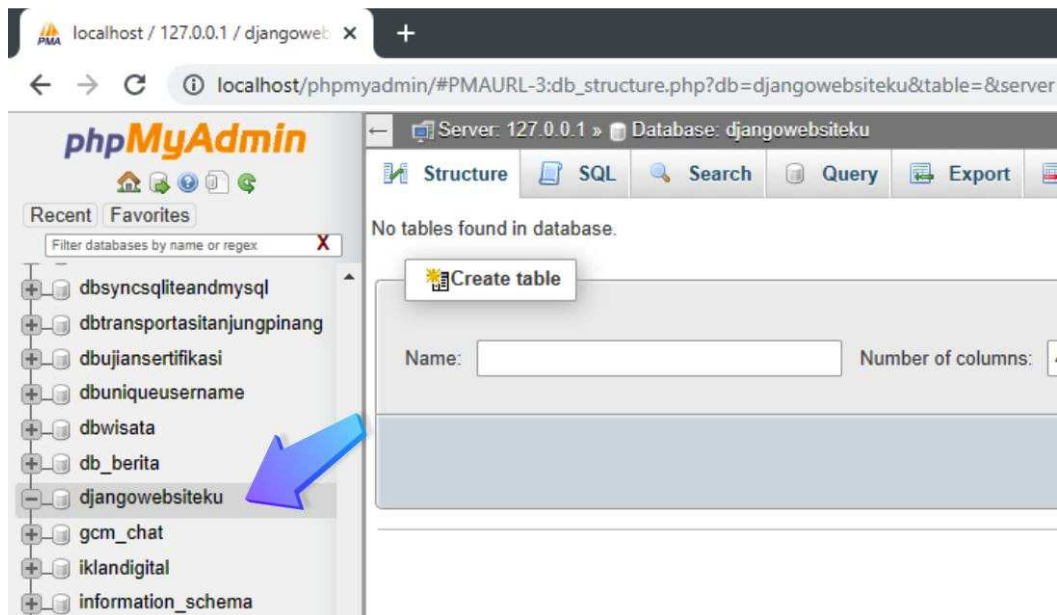
Gambar 88 Setting Database

5. Buka **phpmyadmin** pada browser. Buatlah database dengan nama **djangowebsiteku**.



Gambar 89 Buat Database

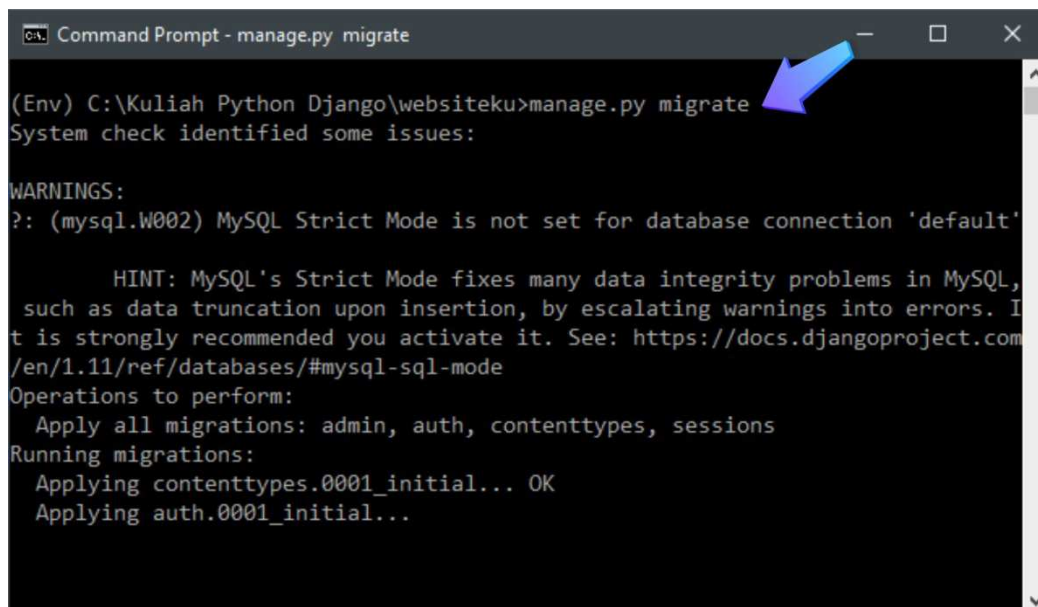
6. Database kosong telah dibuat.



Gambar 90 Database Masih Kosong

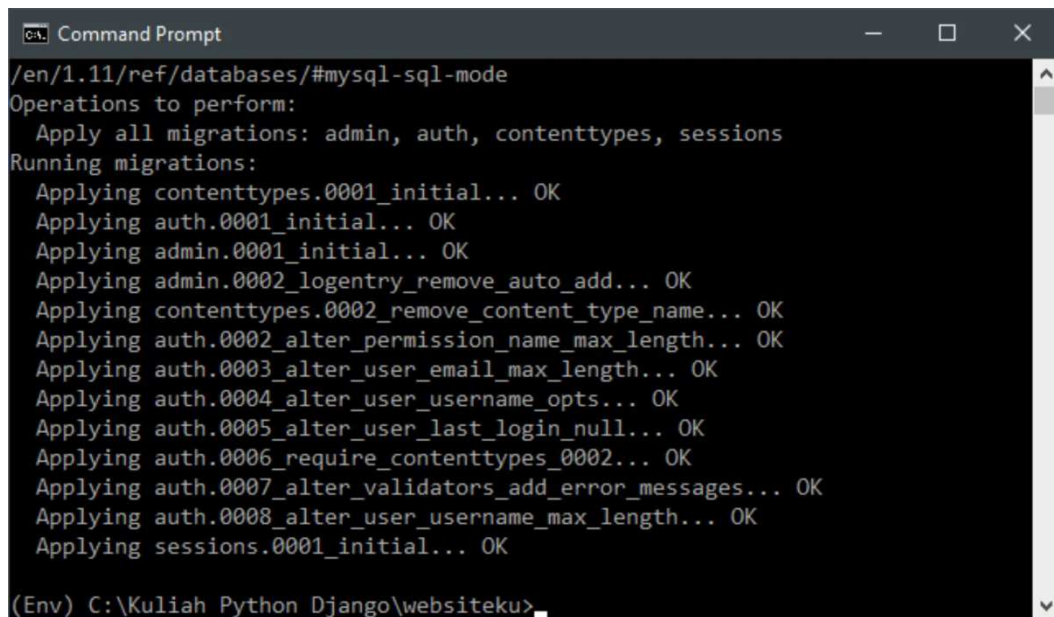
7. Buka kembali cmd, ketik **manage.py migrate**

Migration adalah sebuah fasilitas yang digunakan untuk mempermudah kita ketika ada perubahan dalam database. Pada langkah ini kita membuat tabel, field, dan record untuk aplikasi yang sudah didefinisikan di dalam list **INSTALLED_APPS** pada **settings.py** ke database **djangowebseiteku**.



Gambar 91 Proses Migrasi

8. Proses migrasi selesai.



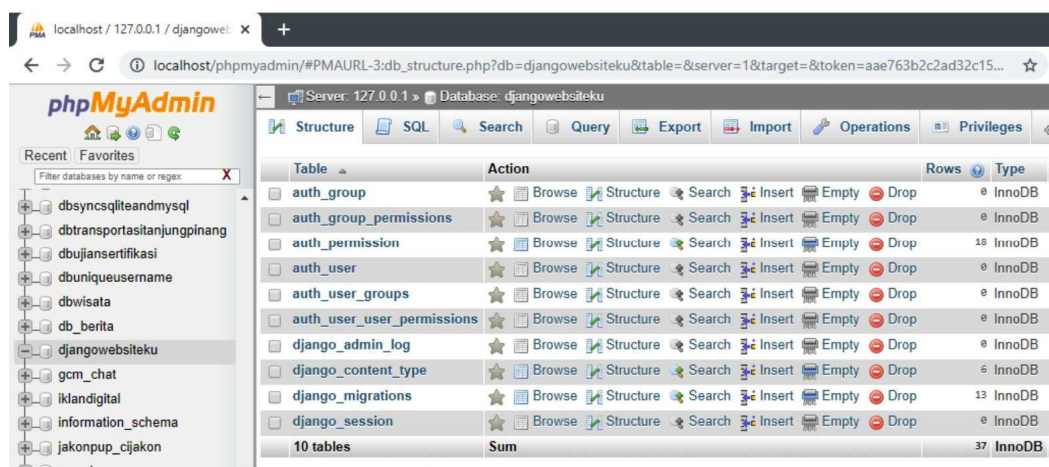
```
C:\> Command Prompt

/en/1.11/ref/databases/#mysql-sql-mode
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 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 sessions.0001_initial... OK

(Env) C:\Kuliah Python Django\websiteku>
```

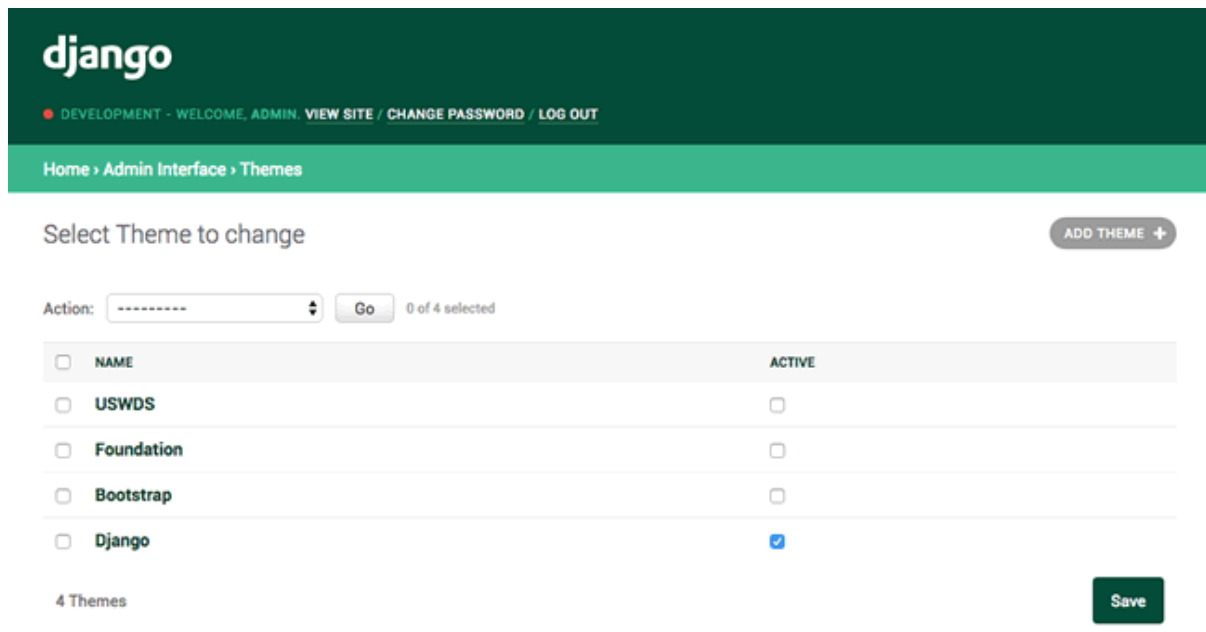
Gambar 92 Proses Migrasi Selesai

9. Cek database untuk melihat data yang berhasil dimigrasi.



Gambar 93 Data yang Berhasil Dimigrasi

Django Admin



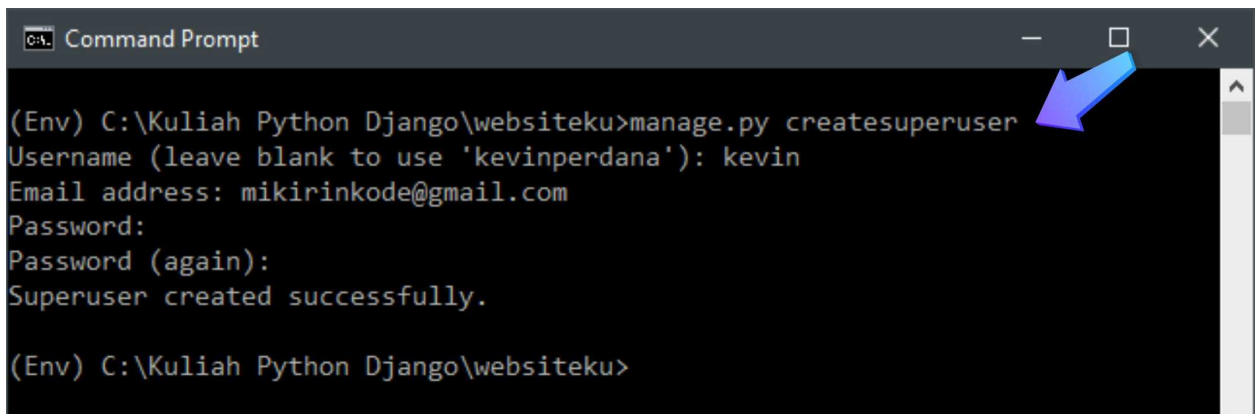
Salah satu bagian dari Django adalah antarmuka admin otomatis. Membaca data dari model untuk menyediakan antarmuka yang dapat mengelola konten di website Anda seperti **insert, read, update, dan delete**. Penggunaan Django Admin disarankan pada alat manajemen internal organisasi. Tidak dimaksudkan untuk digunakan seluruh user sebagai Front-End website, dikarenakan menampilkan terlalu banyak elemen dari model yang tidak perlu.

9.1 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**. Anda dapat membuat akun “**superuser**” yang memiliki akses penuh ke website dan semua izin yang diperlukan menggunakan **manage.py**.

PYTHON **django** web framework

1. Buat **cmd**, kemudian ketik **manage.py createsuperuser**
Isikan **username**, **email**, dan **password** seperti gambar dibawah.

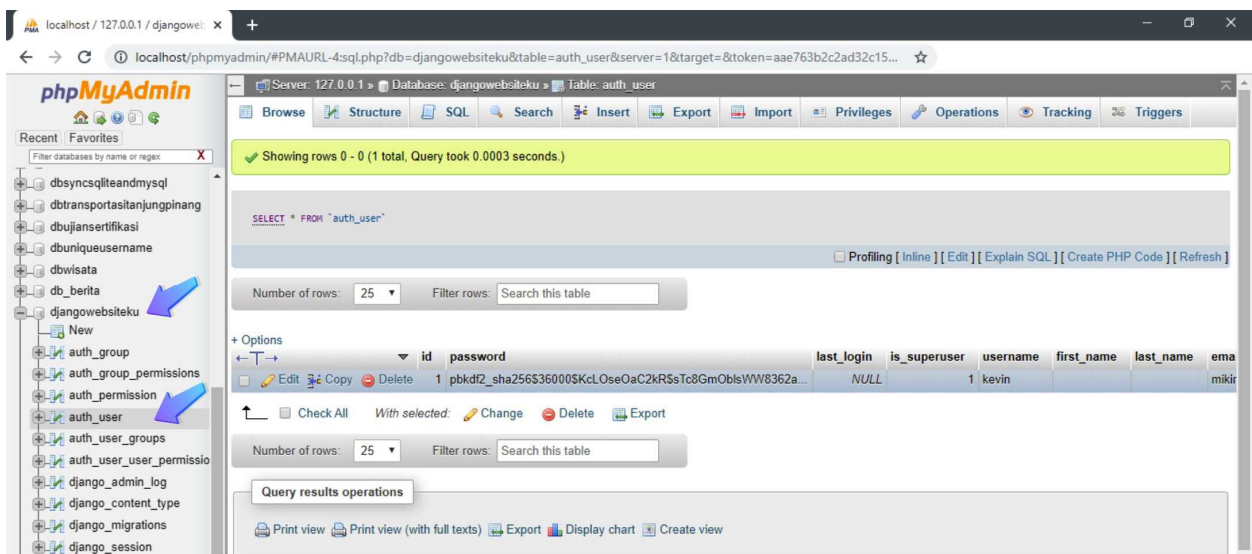


```
(Env) C:\Kuliah Python Django\websiteku>manage.py createsuperuser
Username (leave blank to use 'kevinperdana'): kevin
Email address: mikirinkode@gmail.com
Password:
Password (again):
Superuser created successfully.

(Env) C:\Kuliah Python Django\websiteku>
```

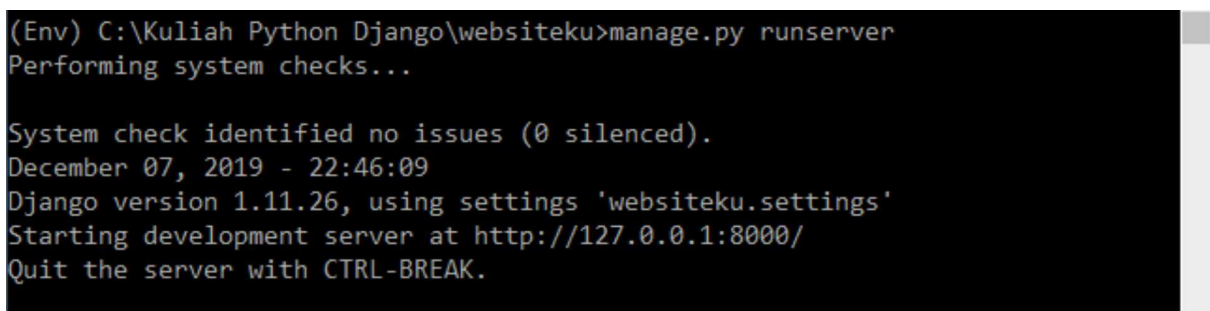
Gambar 94 Membuat Superuser

2. Cek pada **database** dan **tabel auth_user** untuk memastikan data user telah masuk.



Gambar 95 Record Superuser

3. Jalankan server.

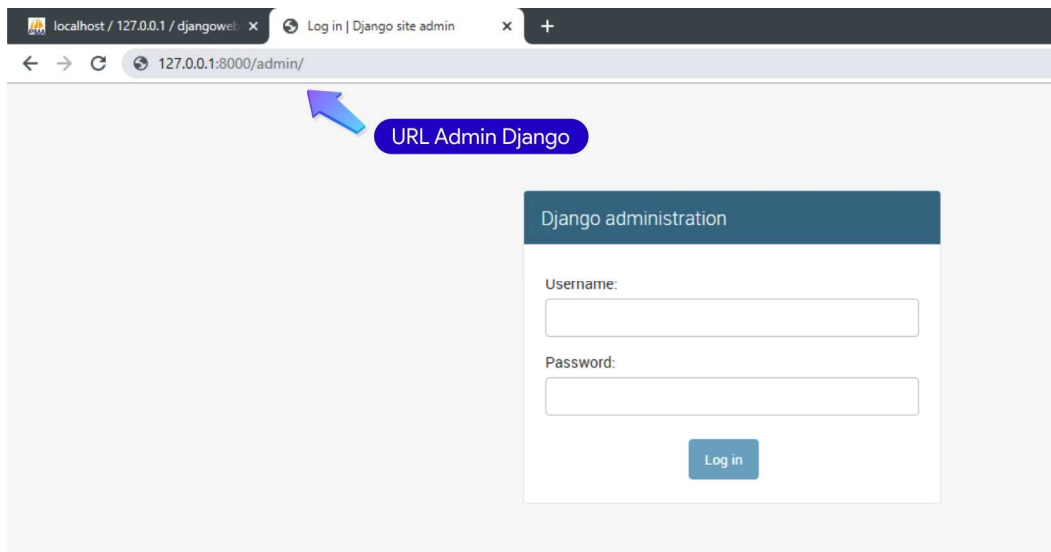


```
(Env) C:\Kuliah Python Django\websiteku>manage.py runserver
Performing system checks...

System check identified no issues (0 silenced).
December 07, 2019 - 22:46:09
Django version 1.11.26, using settings 'websiteku.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

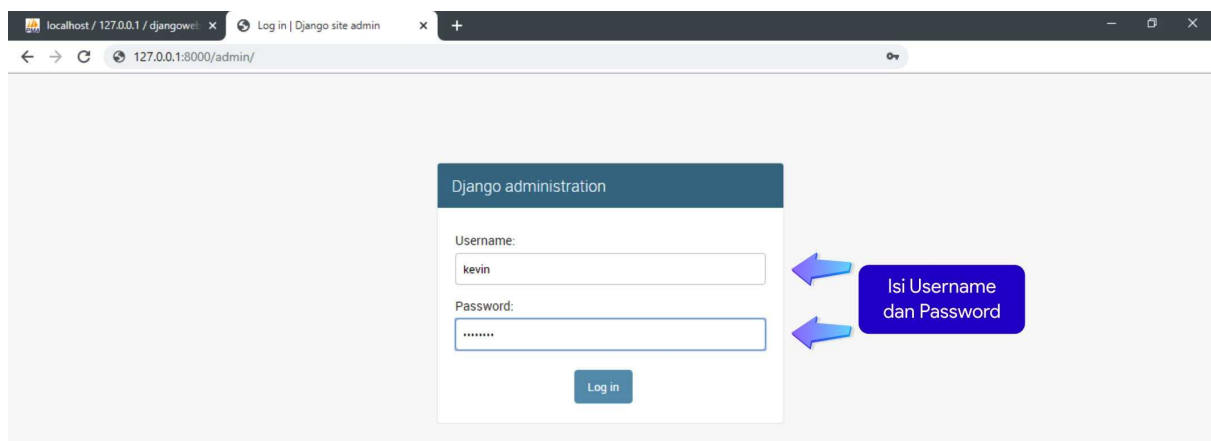
Gambar 96 Jalankan Server

4. Buka halaman **Django Admin** di url **http://127.0.0.1:8000/admin**.



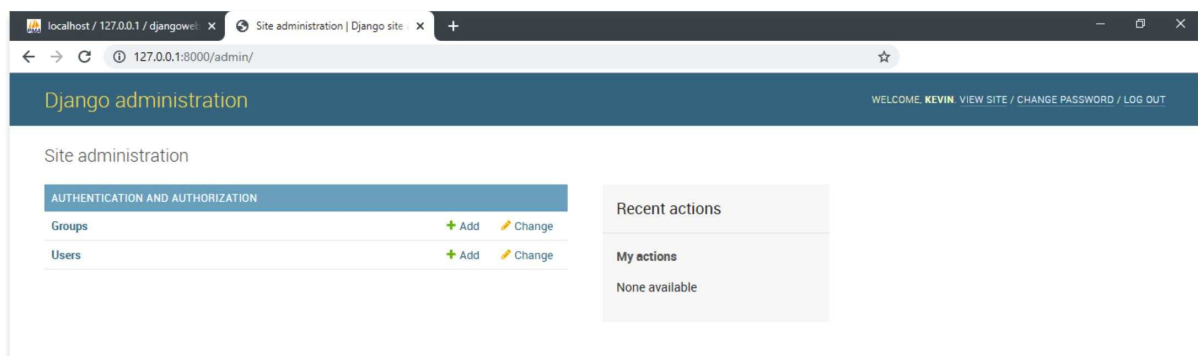
Gambar 97 Halaman Django Admin

5. Isi **username** dan **password** superuser yang telah dibuat tadi.



Gambar 98 Isi Username dan Password

6. Tampilan Django Admin.



Gambar 99 Tampilan Django Admin