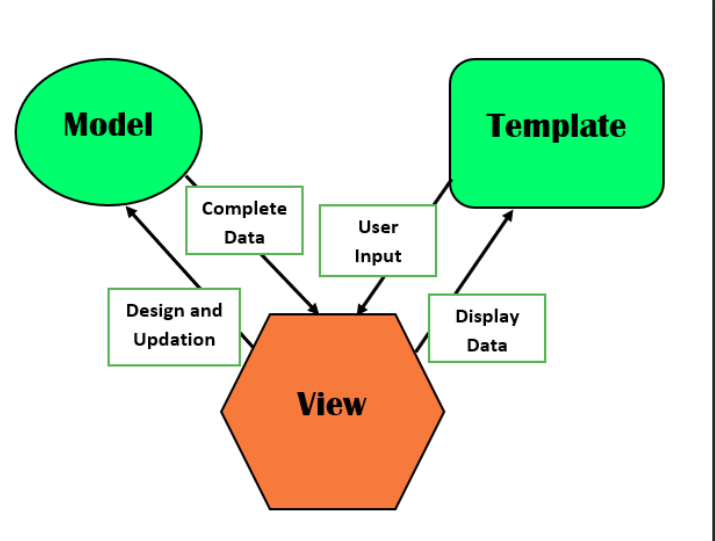
Django for web development

Model View Template (MVT) Architecture.

MVT is a software design pattern for developing a web application.

MVT Structure has the following three parts –

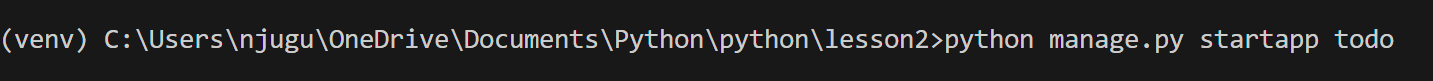
1. Model: The model is going to act as the interface of your data. It is responsible for maintaining data. It is the logical data structure behind the entire application and is represented by a database (generally relational databases such as MySql, Postgres)
2. View: The View is the user interface — what you see in your browser when you render a website. It is represented by HTML/CSS/Javascript and Jinja files.
3. Template: A template consists of static parts of the desired HTML output as well as some special syntax describing how dynamic content will be inserted.



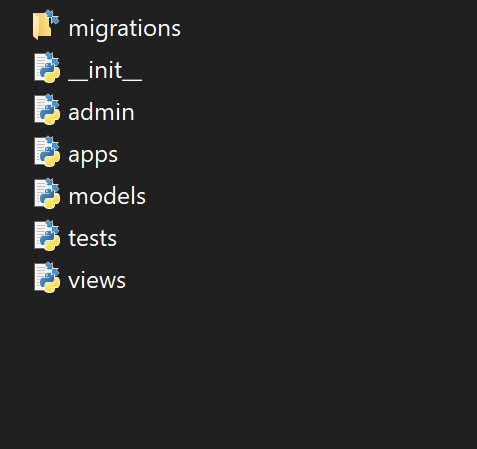
1. Create a django app

First of all, make sure the virtual environment is activated and you are in the same directory as the manage.py file.

The command to create a django app is: “python manage.py startapp <name>”

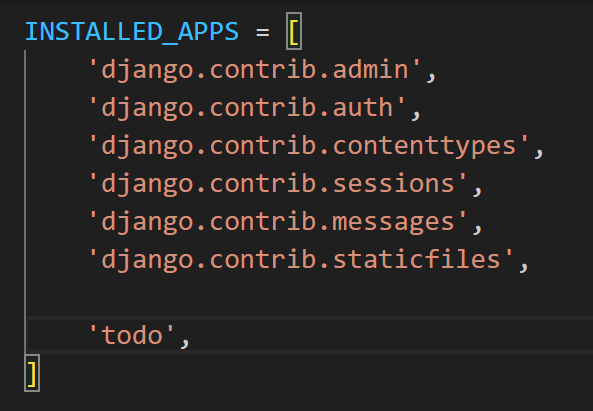


A folder called todo will be created and this are the contents of the folder:



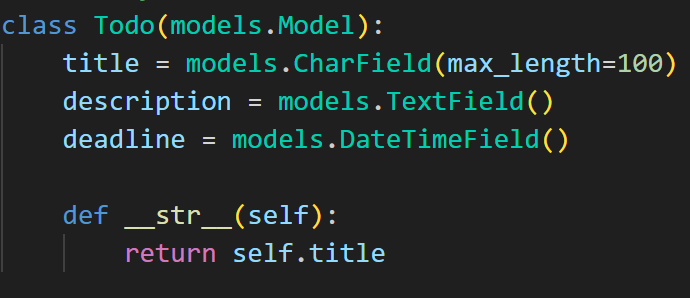
1. Register the app

In the settings.py file which is in the saturdayproject folder, add todo to the INSTALLED\_APPS:



1. Create a model:

In the todo/models.py file:



This model has four fields: title, description, deadline, and id. The id field is automatically created by Django as the primary key of the table.

The \_\_str\_\_ method returns the title of the task as the string representation of the model object.

1. Register the model:

In the todo/admin.py file:

1. Create the table:

In the command line make sure you are in the root directory (where manage.py file is located). Run the following commands:

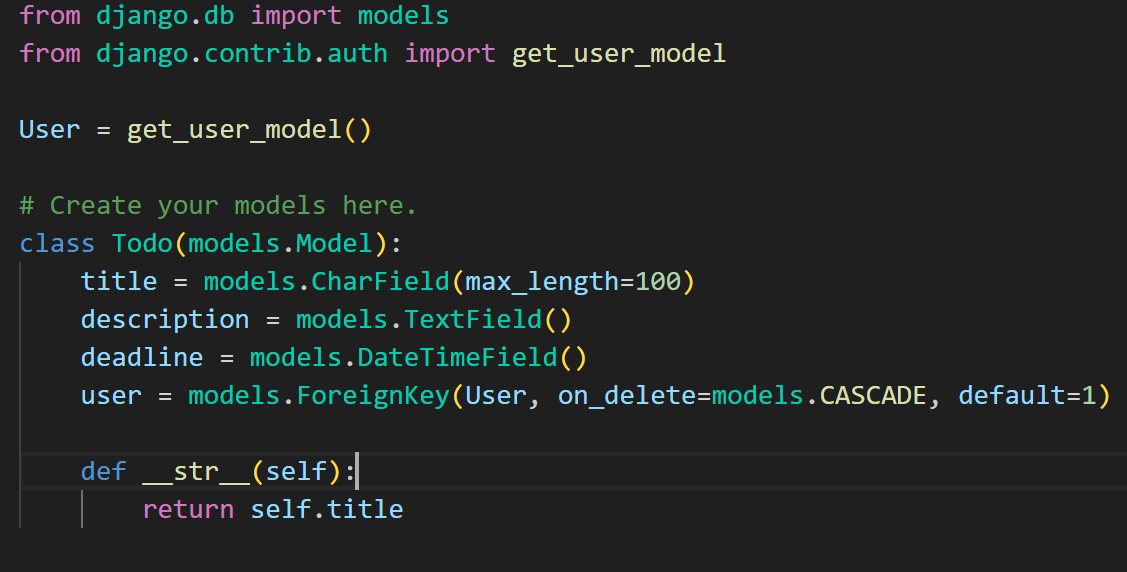
* + - “python manage.py makemigrations”
    - “python manage.py migrate”

1. Start the development server and go to the admin page
2. How to make changes to the model:

Lets say we want User to make many todos the relationship will be;

One to many.

In the todo/model.py:



Make changes to the database by running the following commands:

“python manage.py makemigrations”

“python manage.py migrate”

Note: you may be asked to provide a default hence the reason in the code I have set default to 1. After finishing the migrations remove the parameter.