

EXPERIMENT- 11

NAME: GINI CHACKO

ROLL: 8942

CLASS: SE COMPS B

Aim: To create a web application using Django frame work

Objectives: To learn

- **How to create web application using Django framework in Python**
- **How to connect to the database using Django**

Pre requisite:

- **Basic Knowledge of database management system**
- **Basic knowledge of HTML**

Theory:

Django is a web framework written in [Python](#). It provides a set of tools and functionalities that solves many common problems associated with web development, such as security features, database access, sessions, template processing, URL routing, internationalization, localization, and much more.

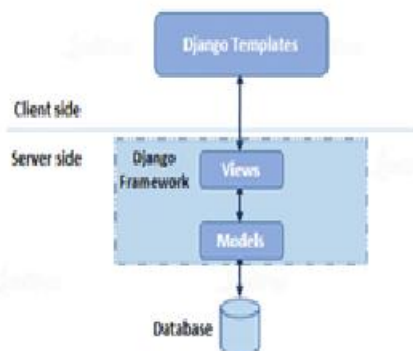
Installing Django

Step 1: Open the command prompt

Step 2: Run the pip command

```
pip install django
```

Django web framework follows the MVT (Model View Template) architecture. This architectural pattern is designed for easy and rapid web development.



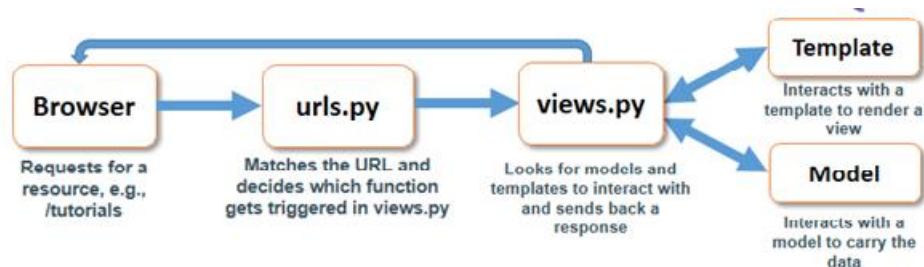
FR. CONCEICAO RODRIGUES COLLEGE OF ENGINEERING.
Fr. Agnel Ashram, Bandstand, Bandra (W) Mumbai 400050

The **model** or the model class is the source of information regarding the data. In other words, the model can be considered as the logical data structure behind the entire web application. Each model is associated with a table in the database.

The **view** or the view function takes up different web requests and returns respective responses. For instance, the request can be POST or GET type, and the response can be an HTML page, a 404 error, a redirect, or an image.

Template layer determines how a user sees the response Django returns. It basically controls the user interaction.

The following diagram shows how these components communicate with each other to make the development in Django easy and rapid



First, a user requests for a resource. Django considers the request as a URL and matches it with the existing URL paths in the `urls.py` file. This process of matching the user-requested URL to the one in `urls.py` is known as URL mapping. Once the URL matches, Django carries out the further process.

Once the URL is mapped, Django jumps to the `views.py` folder and calls a view. The triggered view looks for models and templates to interact with, and then it returns the response back to the user. Here, the model deals with the data associated with the user request. On the other hand, the template deals with the HTML and the static files, such as CSS files, JS files, images, etc., that are required to complete the view.

Alright, now that we know how the architecture works, let us move ahead with our Django tutorial and see how to use Django framework to build a web application.

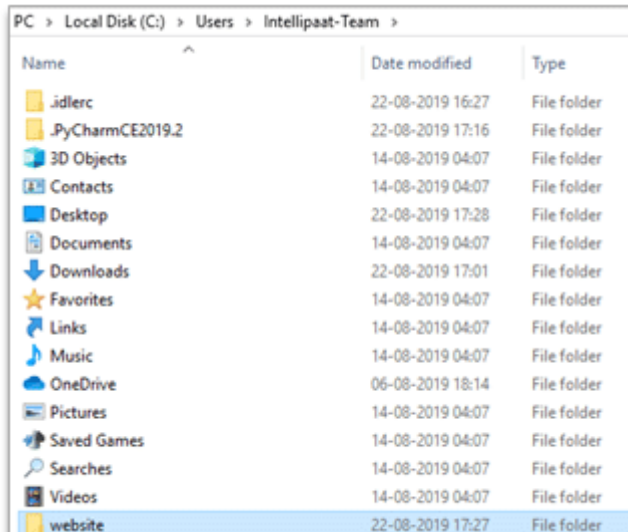
Creating Django Project:

In order to create a new Django project, run the following

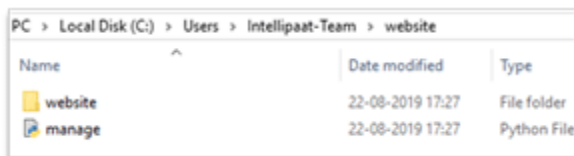
django-admin startproject command in the command prompt

Now, go to the folder where the command prompt was being run from. There you can find a new project directory that looks like this

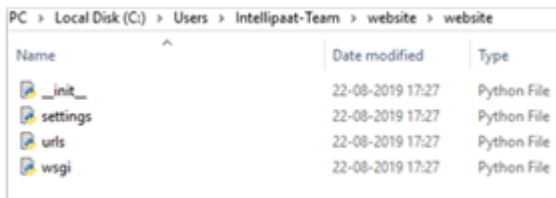
FR. CONCEICAO RODRIGUES COLLEGE OF ENGINEERING.
Fr. Agnel Ashram, Bandstand, Bandra (W) Mumbai 400050



Inside the 'website' directory (root directory), you will see another directory with name same as the root directory.

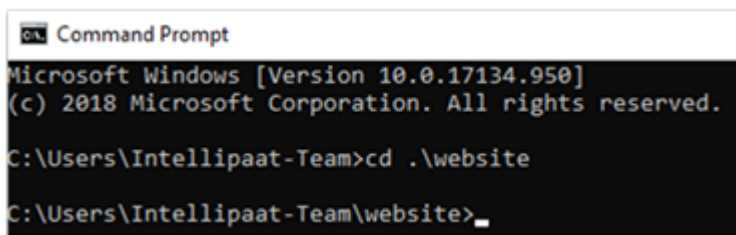


Inside that directory the following default python files are created.



Creating an Application:

- Open the command prompt
- Navigate to the directory where you have created the project



Run the following command to create an app inside the project directory

```
python manage.py startapp tutorial
```

FR. CONCEICAO RODRIGUES COLLEGE OF ENGINEERING.
Fr. Agnel Ashram, Bandstand, Bandra (W) Mumbai 400050

```

Command Prompt
Microsoft Windows [Version 10.0.17134.950]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Intellipaat-Team>cd .\website

C:\Users\Intellipaat-Team\website>python manage.py startapp tutorial
  
```

Here, you will see what difference this command makes inside the project directory. There is a new folder called 'tutorial' inside the 'website' directory.

PC > Local Disk (C:) > Users > Intellipaat-Team > website >

Name	Date modified	Type
.idea	22-08-2019 19:12	File folder
tutorial	23-08-2019 18:50	File folder
website	23-08-2019 17:37	File folder
db.sqlite3	23-08-2019 17:37	SQLITE3 File
manage	22-08-2019 17:27	Python File

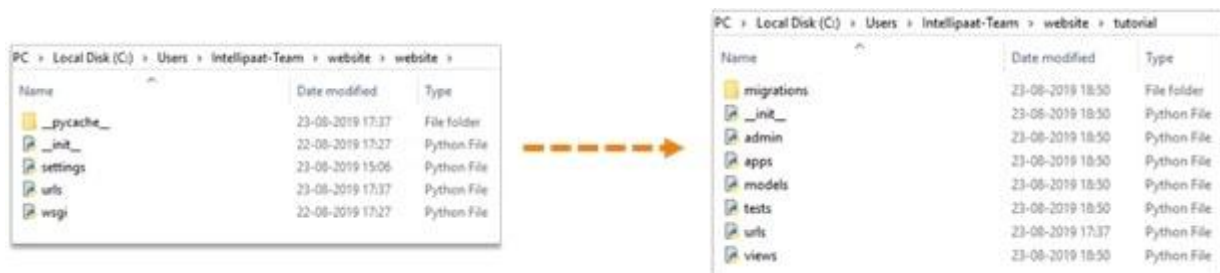
Now, if you go inside the 'tutorial' application folder, you will see the following Python files Django created for you to configure the application.

PC > Local Disk (C:) > Users > Intellipaat-Team > website > tutorial >

Name	Date modified	Type
migrations	23-08-2019 18:50	File folder
init	23-08-2019 18:50	Python File
admin	23-08-2019 18:50	Python File
apps	23-08-2019 18:50	Python File
models	23-08-2019 18:50	Python File
tests	23-08-2019 18:50	Python File
views	23-08-2019 18:50	Python File

Creating a URL and a View for the Web Application

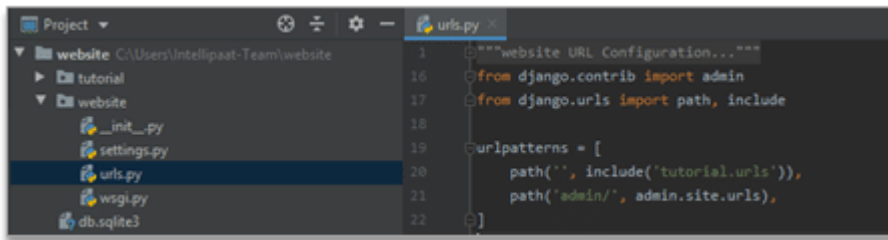
Step 1: First of all, paste the URL file inside the newly created application. You can copy this file from the website directory



Step 2: Now, inside the primary app, which is 'website,' make the following configurations. Open 'urls.py' and add the pattern to point to the 'tutorial' app. Import 'include' and add the path to the 'urls.py' file inside 'tutorial'

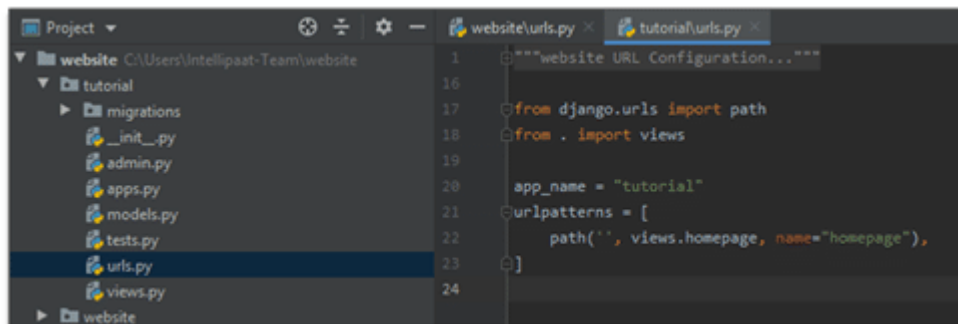
FR. CONCEICAO RODRIGUES COLLEGE OF ENGINEERING.
Fr. Agnel Ashram, Bandstand, Bandra (W) Mumbai 400050

```
Path('',include('tutorial.urls'))
```



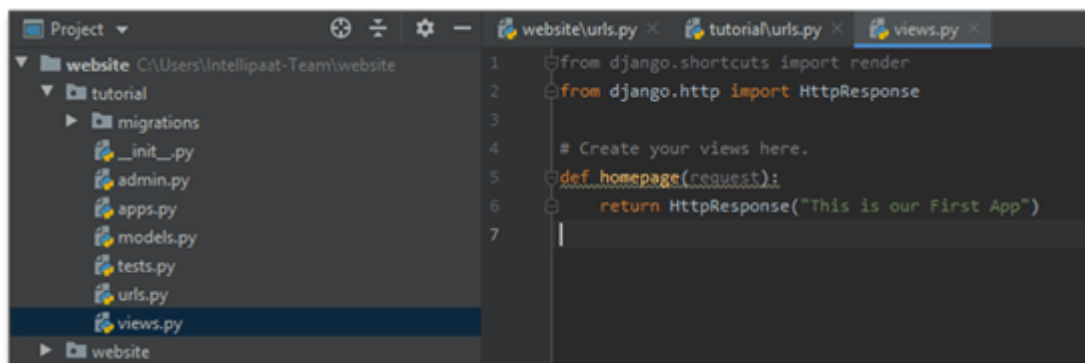
Step 3: Go to 'tutorial.urls.py' and import 'views' as shown below. Add the app name for future reference and add the path to point to a homepage view for the tutorial app

```
Path('',views.homepage, name="homepage")
```



Step 4: But so far we don't have any view called homepage. So, add a simple HttpResponse view called homepage. Don't forget to import HttpResponse from django.http as shown below:

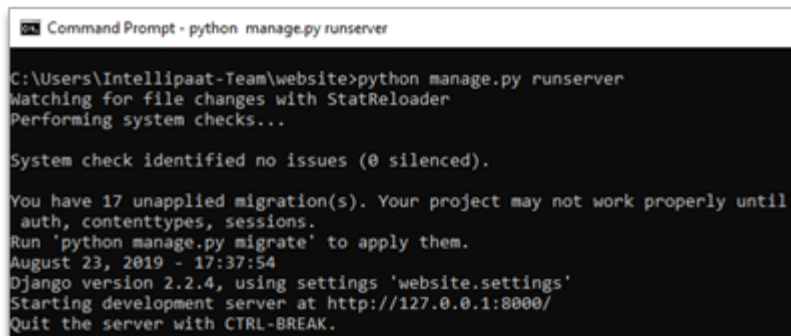
```
def homepage(request):  
    return HttpResponse("First App")
```



Step 5: Now, boot up the web server and run the following command

```
python manage.py runserver
```

FR. CONCEICAO RODRIGUES COLLEGE OF ENGINEERING.
Fr. Agnel Ashram, Bandstand, Bandra (W) Mumbai 400050



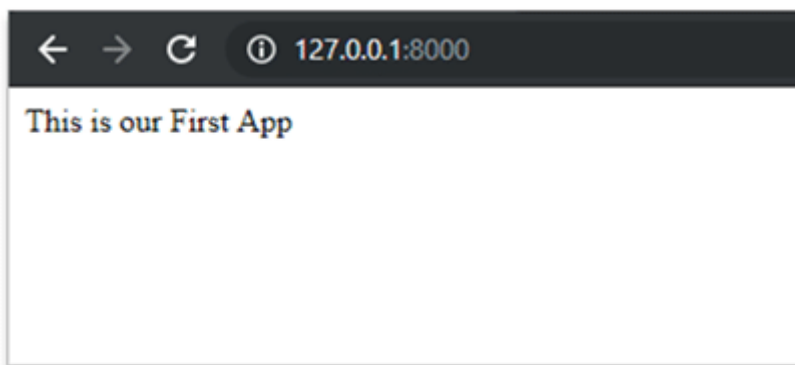
```
Command Prompt - python manage.py runserver

C:\Users\Intellipaat-Team\website>python 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
auth, contenttypes, sessions.
Run 'python manage.py migrate' to apply them.
August 23, 2019 - 17:37:54
Django version 2.2.4, using settings 'website.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

Step 6: Open your browser location and go to the following location



You will see the string that was passed as an `HttpResponse` being rendered over there. This means, we have successfully created our very first Django application.

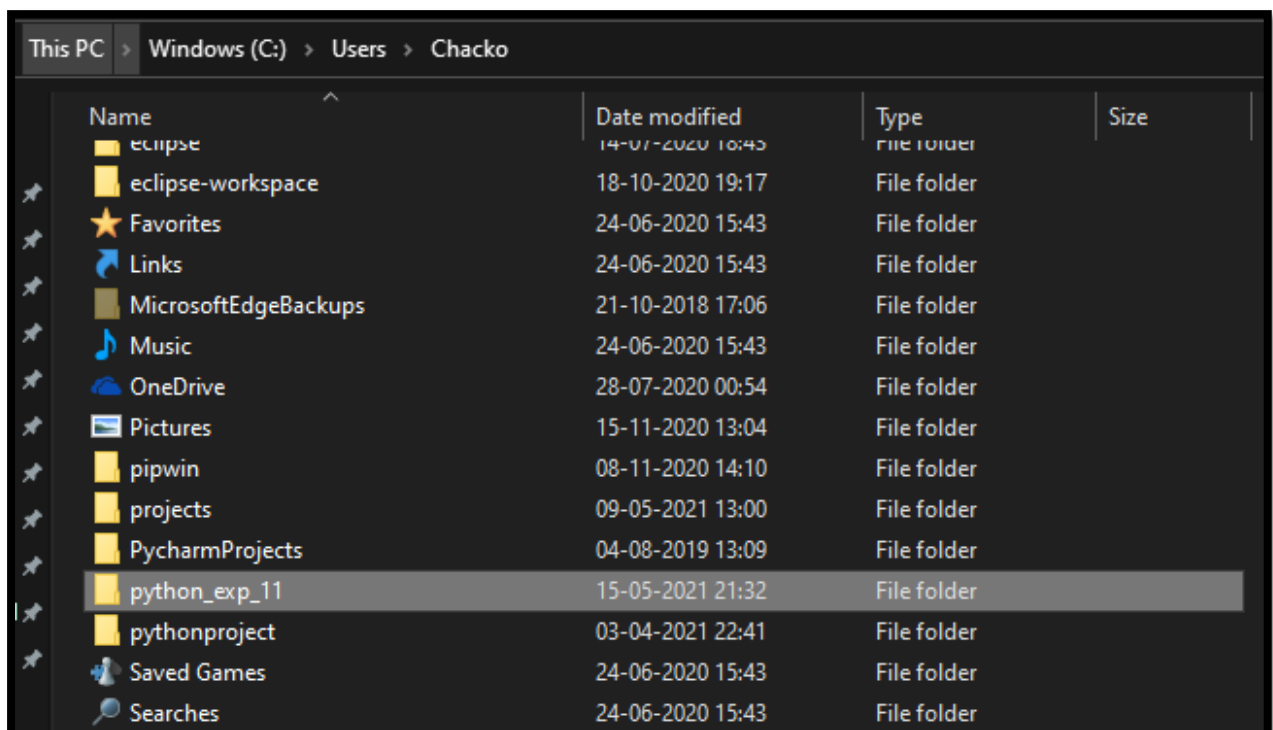
FR. CONCEICAO RODRIGUES COLLEGE OF ENGINEERING.
Fr. Agnel Ashram, Bandstand, Bandra (W) Mumbai 400050

IMPLEMENTATION:

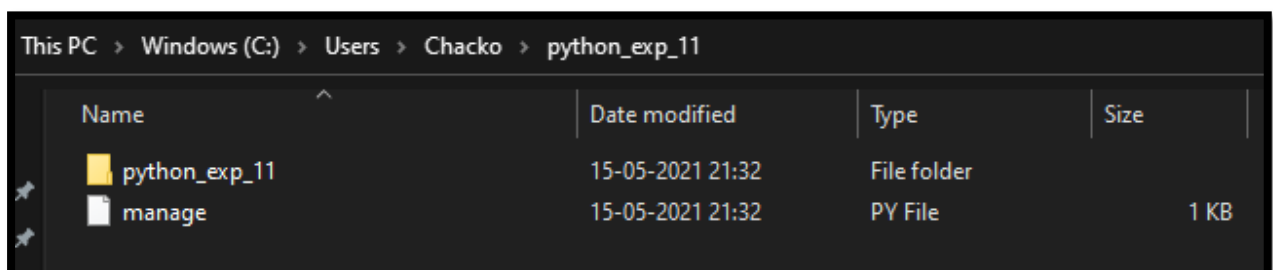
- **django-admin startproject** command in the command prompt

```
C:\Users\Chacko>django-admin startproject python_exp_11
```

- Now, go to the folder where the command prompt was being run from. There you can find a new project directory that looks like this

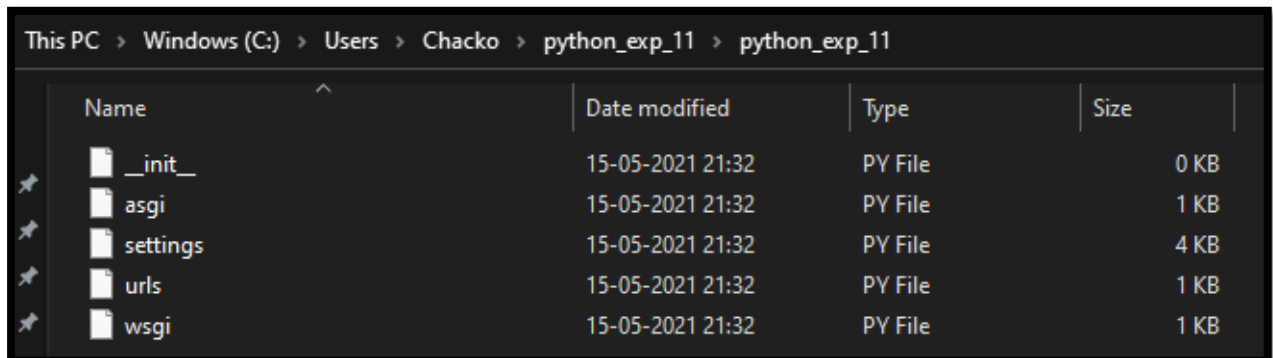


- Inside the 'python_exp_11' directory (root directory), you will see another directory with name same as the root directory.



FR. CONCEICAO RODRIGUES COLLEGE OF ENGINEERING.
Fr. Agnel Ashram, Bandstand, Bandra (W) Mumbai 400050

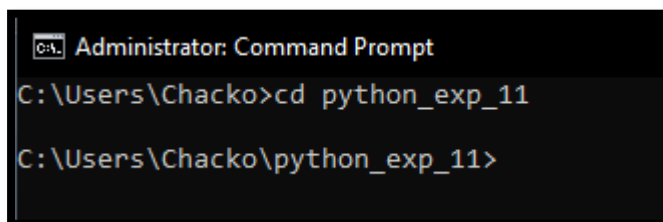
- Inside that directory the following default python files are created.



This PC > Windows (C:) > Users > Chacko > python_exp_11 > python_exp_11				
Name	Date modified	Type	Size	
__init__	15-05-2021 21:32	PY File	0 KB	
asgi	15-05-2021 21:32	PY File	1 KB	
settings	15-05-2021 21:32	PY File	4 KB	
urls	15-05-2021 21:32	PY File	1 KB	
wsgi	15-05-2021 21:32	PY File	1 KB	

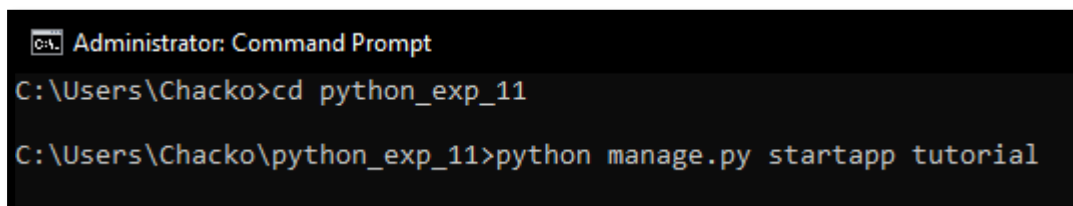
- **Creating an Application:**

- Open the command prompt
- Navigate to the directory where you have created the project



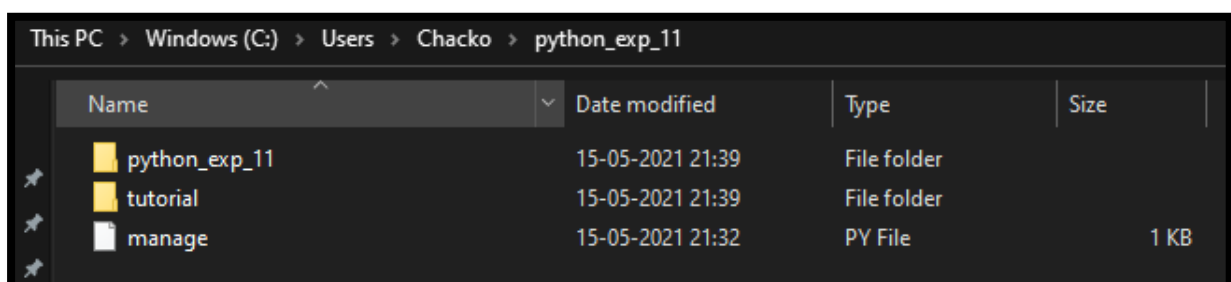
```
Administrator: Command Prompt
C:\Users\Chacko>cd python_exp_11
C:\Users\Chacko\python_exp_11>
```

- Run the following command to create an app inside the project directory
 - python manage.py startapp tutorial



```
Administrator: Command Prompt
C:\Users\Chacko>cd python_exp_11
C:\Users\Chacko\python_exp_11>python manage.py startapp tutorial
```

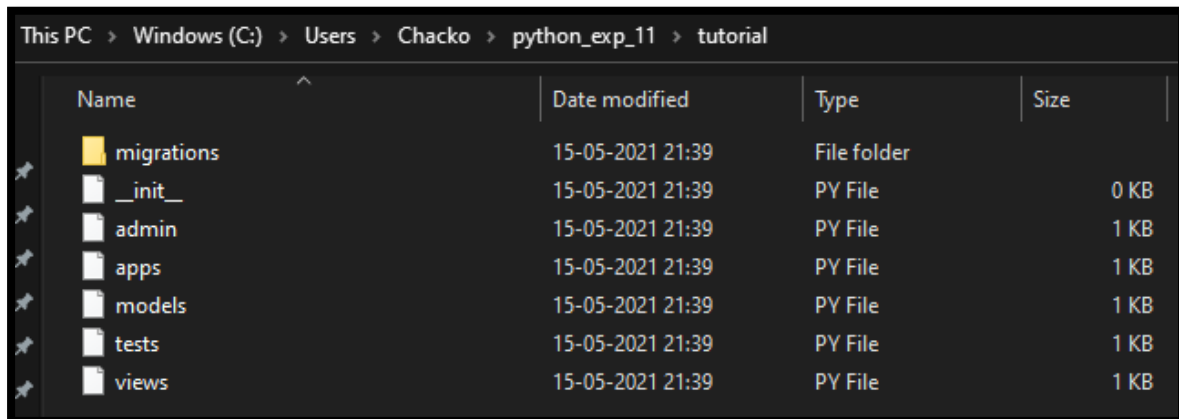
- Here, you will see what difference this command makes inside the project directory. There is a new folder called ‘tutorial’ inside the ‘python_exp_11’ directory.



This PC > Windows (C:) > Users > Chacko > python_exp_11				
Name	Date modified	Type	Size	
python_exp_11	15-05-2021 21:39	File folder		
tutorial	15-05-2021 21:39	File folder		
manage	15-05-2021 21:32	PY File	1 KB	

FR. CONCEICAO RODRIGUES COLLEGE OF ENGINEERING.
Fr. Agnel Ashram, Bandstand, Bandra (W) Mumbai 400050

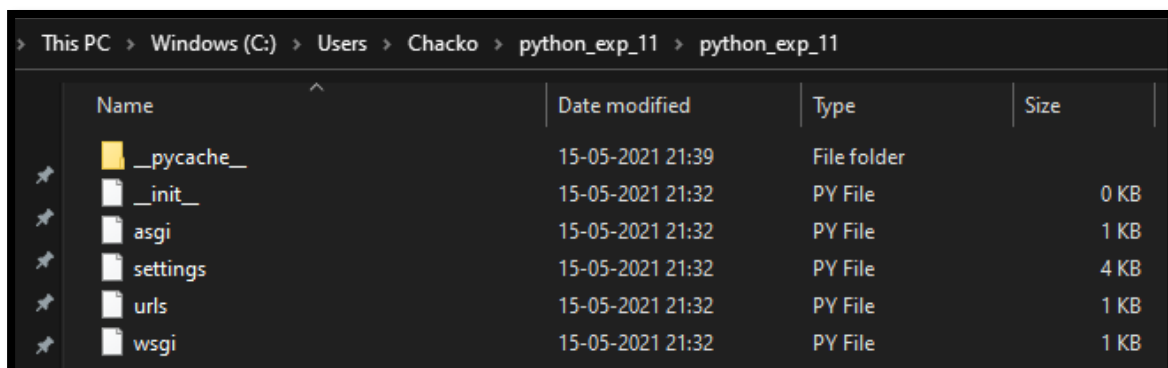
- Now, if you go inside the ‘tutorial’ application folder, you will see the following Python files Django created for you to configure the application.



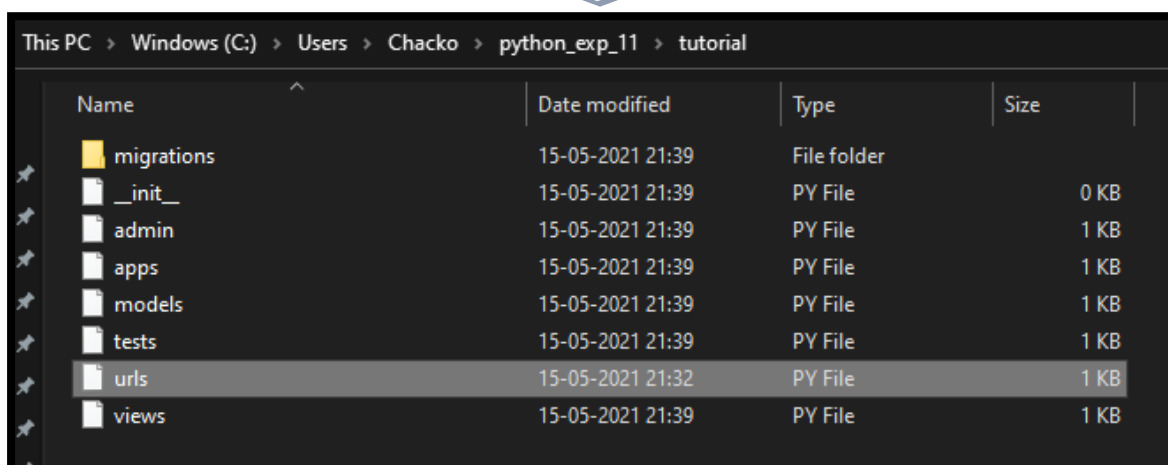
Name	Date modified	Type	Size
migrations	15-05-2021 21:39	File folder	
__init__	15-05-2021 21:39	PY File	0 KB
admin	15-05-2021 21:39	PY File	1 KB
apps	15-05-2021 21:39	PY File	1 KB
models	15-05-2021 21:39	PY File	1 KB
tests	15-05-2021 21:39	PY File	1 KB
views	15-05-2021 21:39	PY File	1 KB

- Creating a URL and a View for the Web Application

Step 1: First of all, paste the URL file inside the newly created application.
You can copy this file from the website directory



Name	Date modified	Type	Size
__pycache__	15-05-2021 21:39	File folder	
__init__	15-05-2021 21:32	PY File	0 KB
asgi	15-05-2021 21:32	PY File	1 KB
settings	15-05-2021 21:32	PY File	4 KB
urls	15-05-2021 21:32	PY File	1 KB
wsgi	15-05-2021 21:32	PY File	1 KB

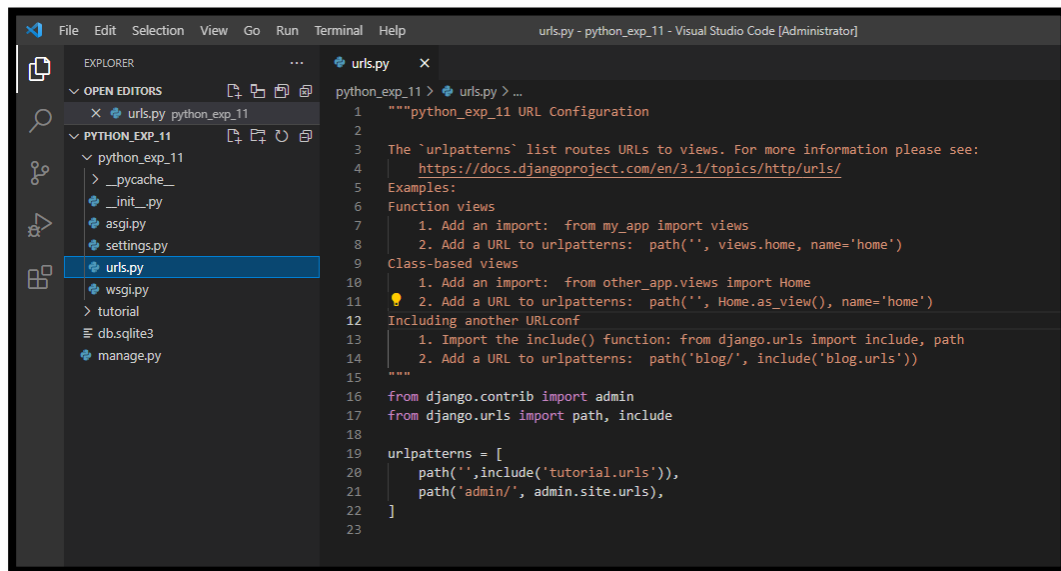


Name	Date modified	Type	Size
migrations	15-05-2021 21:39	File folder	
__init__	15-05-2021 21:39	PY File	0 KB
admin	15-05-2021 21:39	PY File	1 KB
apps	15-05-2021 21:39	PY File	1 KB
models	15-05-2021 21:39	PY File	1 KB
tests	15-05-2021 21:39	PY File	1 KB
urls	15-05-2021 21:32	PY File	1 KB
views	15-05-2021 21:39	PY File	1 KB

FR. CONCEICAO RODRIGUES COLLEGE OF ENGINEERING.
Fr. Agnel Ashram, Bandstand, Bandra (W) Mumbai 400050

- **Step 2:** Now, inside the primary app, which is 'python_exp_11,' make the following configurations. Open 'urls.py' and add the pattern to point to the 'tutorial' app. Import 'include' and add the path to the 'urls.py' file inside 'tutorial'

Path('',include('tutorial.urls'))



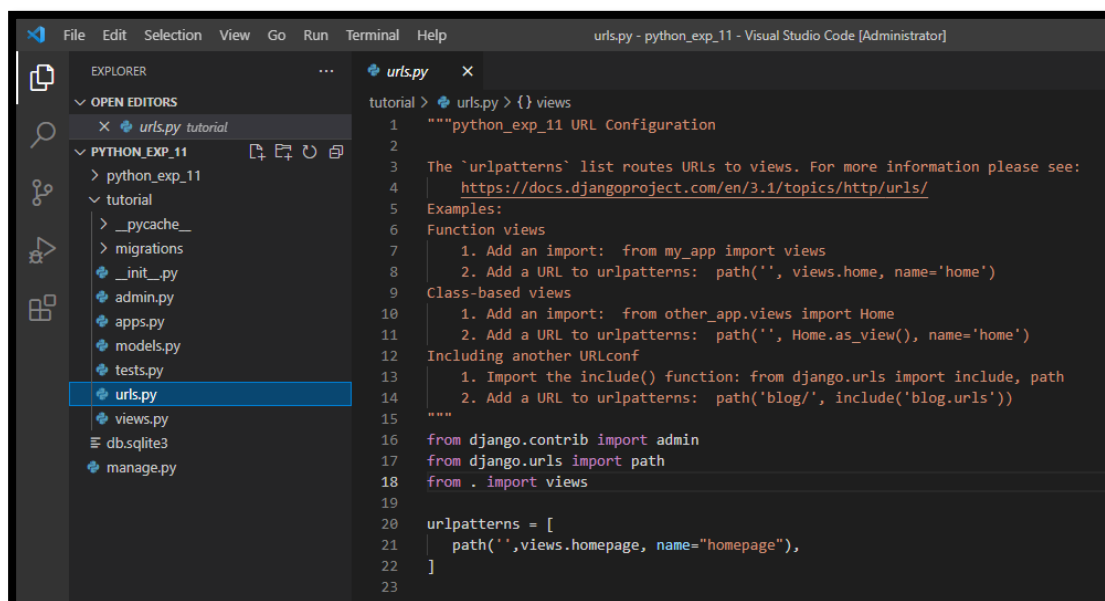
```

1 """python_exp_11 URL Configuration
2
3 The 'urlpatterns' list routes URLs to views. For more information please see:
4     https://docs.djangoproject.com/en/3.1/topics/http/urls/
5 Examples:
6 Function views
7     1. Add an import: from my_app import views
8     2. Add a URL to urlpatterns: path('', views.home, name='home')
9 Class-based views
10    1. Add an import: from other_app.views import Home
11    2. Add a URL to urlpatterns: path('', Home.as_view(), name='home')
12 Including another URLconf
13    1. Import the include() function: from django.urls import include, path
14    2. Add a URL to urlpatterns: path('blog/', include('blog.urls'))
15 """
16 from django.contrib import admin
17 from django.urls import path, include
18
19 urlpatterns = [
20     path('',include('tutorial.urls')),
21     path('admin/', admin.site.urls),
22 ]
23

```

- **Step 3:** Go to 'tutorial.urls.py' and import 'views' as shown below. Add the app name for future reference and add the path to point to a homepage view for the tutorial app

Path('',views.homepage, name="homepage")



```

tutorial > urls.py > {} views
1 """python_exp_11 URL Configuration
2
3 The 'urlpatterns' list routes URLs to views. For more information please see:
4     https://docs.djangoproject.com/en/3.1/topics/http/urls/
5 Examples:
6 Function views
7     1. Add an import: from my_app import views
8     2. Add a URL to urlpatterns: path('', views.home, name='home')
9 Class-based views
10    1. Add an import: from other_app.views import Home
11    2. Add a URL to urlpatterns: path('', Home.as_view(), name='home')
12 Including another URLconf
13    1. Import the include() function: from django.urls import include, path
14    2. Add a URL to urlpatterns: path('blog/', include('blog.urls'))
15 """
16 from django.contrib import admin
17 from django.urls import path
18 from . import views
19
20 urlpatterns = [
21     path('',views.homepage, name="homepage"),
22 ]
23

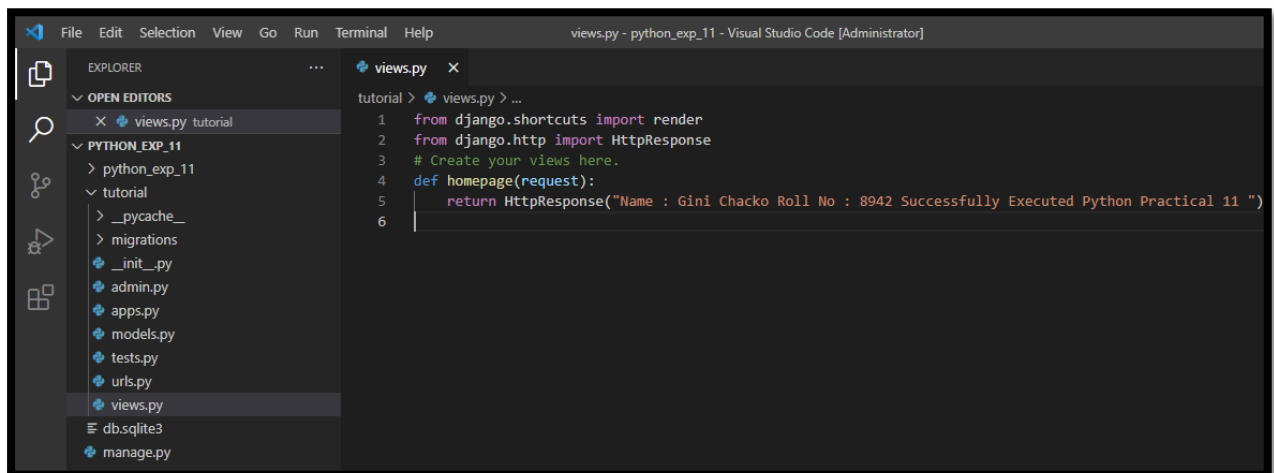
```

FR. CONCEICAO RODRIGUES COLLEGE OF ENGINEERING.
Fr. Agnel Ashram, Bandstand, Bandra (W) Mumbai 400050

- **Step 4:** But so far we don't have any view called homepage. So, add a simple HttpResponse view called homepage. Don't forget to import HttpResponse from django.http as shown below:

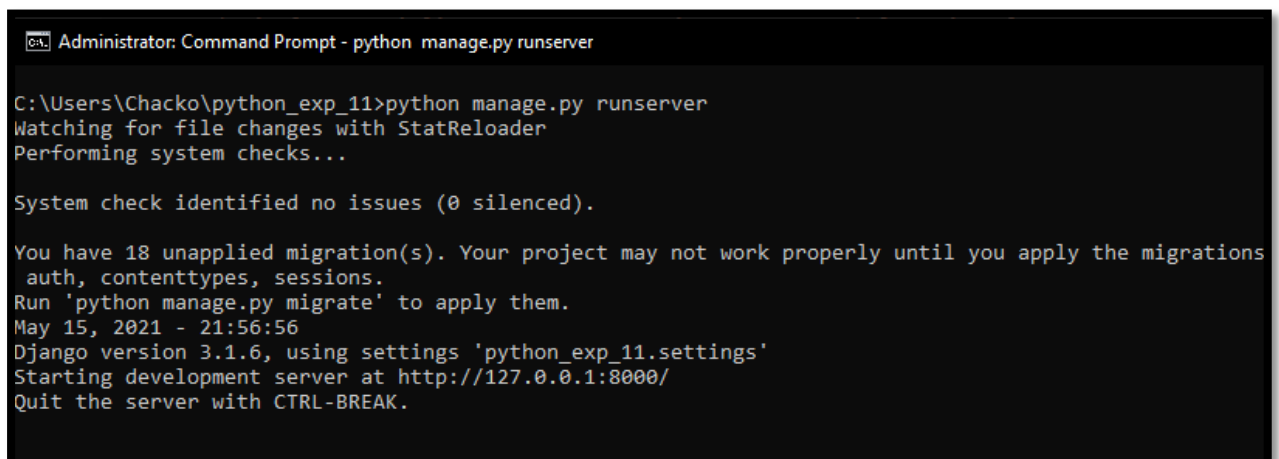
def homepage(request):

**return HttpResponse("Name : Gini Chacko Roll No : 8942 Successfully Executed
Python Practical 11")**



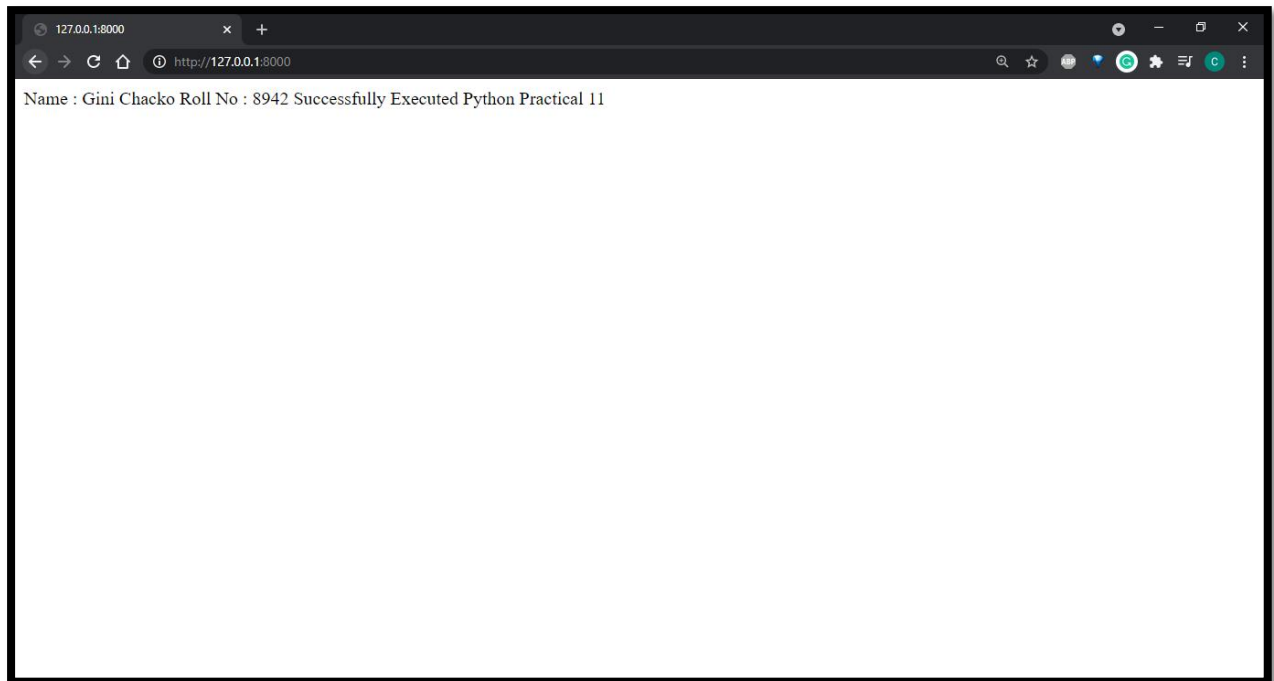
- **Step 5:** Now, boot up the web server and run the following command

python manage.py runserver



FR. CONCEICAO RODRIGUES COLLEGE OF ENGINEERING.
Fr. Agnel Ashram, Bandstand, Bandra (W) Mumbai 400050

- **Step 6:** Open your browser location and go to the following location



You will see the string that was passed as an HttpResponse being rendered over there. This means, we have successfully created our very first Django application.

POSTLAB QUESTION:

- 1) Kindly attempt the following quiz based on Django and database connectivity <https://forms.gle/gggC6JT2HMBq8xFK7> and upload screenshot of the score of quiz.

A screenshot of a Google Form titled 'SEC_B_PYTHON QUIZ 5 - Django and Database connectivity'. The form is displayed on a light purple background. It includes a header section with the title and a 'Total points' indicator showing '10/10'. Below the header, there is a description: 'CSL405.4: Develop python based web application using Django framework, database and networking concepts. (Apply)'. The form contains three input fields: 'Email' with the value 'croe.8942.ce@gmail.com', 'Roll No' with the value '8942', and 'Name' with the value 'Gini Chacko'.

GINI CHACKO 8942