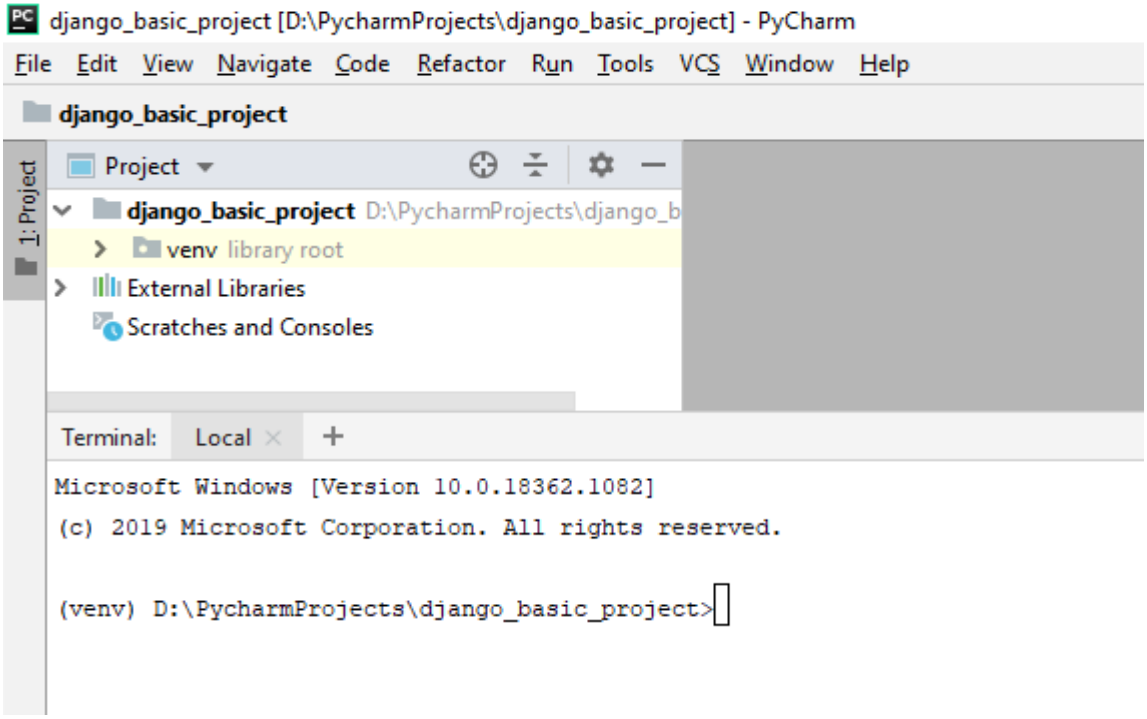
Basic Django project

Django is a web development framework which is used to create web applications in python.

Prerequisites: pycharm community Edition, Python 3.6, 3.7, 3.8, Django 3.1

Creating pycharm project:

- Firstly we create a pycharm project with in virtual environment by name (django_basic_Project) as shown in below fig.
- By clicking on terminal option at the left bottom of page terminal will open.



Install Django package:

- We can install by using below command in the terminal.

Pip install django

```
(venv) D:\PycharmProjects\django_basic_project>pip install django
Collecting django
```

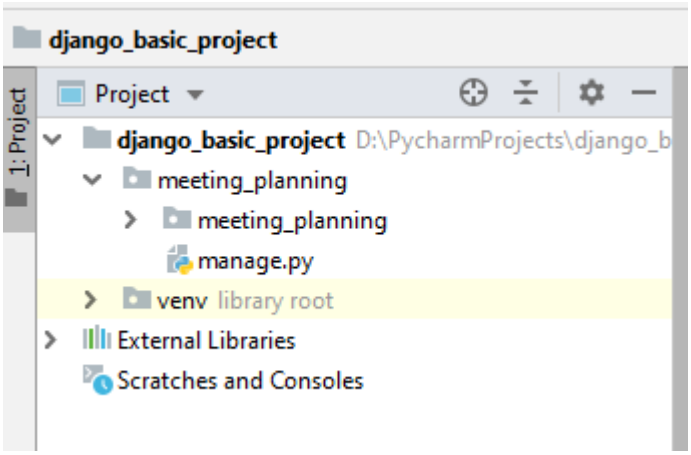
Create Django Project:

- We can create django project (meeting_planning) using below command.

django-admin startproject meeting_planning

```
(venv) D:\PycharmProjects\django_basic_project>django-admin startproject meeting_planning
```

- By entering above command project looks like as shown in below fig.



Run Django server and Launch Browser:

- We can launch development server by using below command. Before that terminal directory should be under django project (meeting_planning).

Python manage.py runserver

```
(venv) D:\PycharmProjects\django_basic_project>cd meeting_planning

(venv) D:\PycharmProjects\django_basic_project\meeting_planning>python manage.py runserver
```

- After executing the above command it will create a server using local host, to launch this go to chrome browser and enter the link as shown in terminal Ex (http://127.0.0.1:8000/) .
- After launching the server in the terminal it shows response 200 with http method GET as shown below fig.

```
Django version 3.1.2, using settings 'meeting_planning.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
[17/Oct/2020 16:16:51] "GET / HTTP/1.1" 200 16351
[17/Oct/2020 16:16:51] "GET /static/admin/css/fonts.css HTTP/1.1" 200 423
[17/Oct/2020 16:16:52] "GET /static/admin/fonts/Roboto-Bold-webfont.woff HTTP/1.1" 200 8
[17/Oct/2020 16:16:52] "GET /static/admin/fonts/Roboto-Light-webfont.woff HTTP/1.1" 200
[17/Oct/2020 16:16:52] "GET /static/admin/fonts/Roboto-Regular-webfont.woff HTTP/1.1" 20
```

- To kill the server process enter CRTL+C.

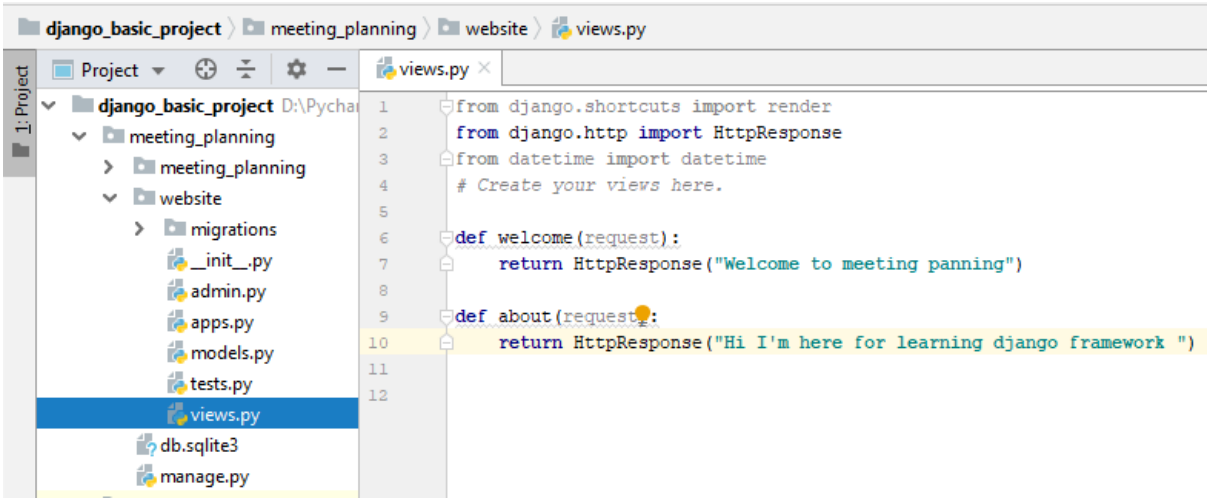
Creating an App (website):

- Every web application has different pages, in django each page is consider as an app.now we going to create ‘website’ app by using below command. Make sure enter this command under meeting_planner.

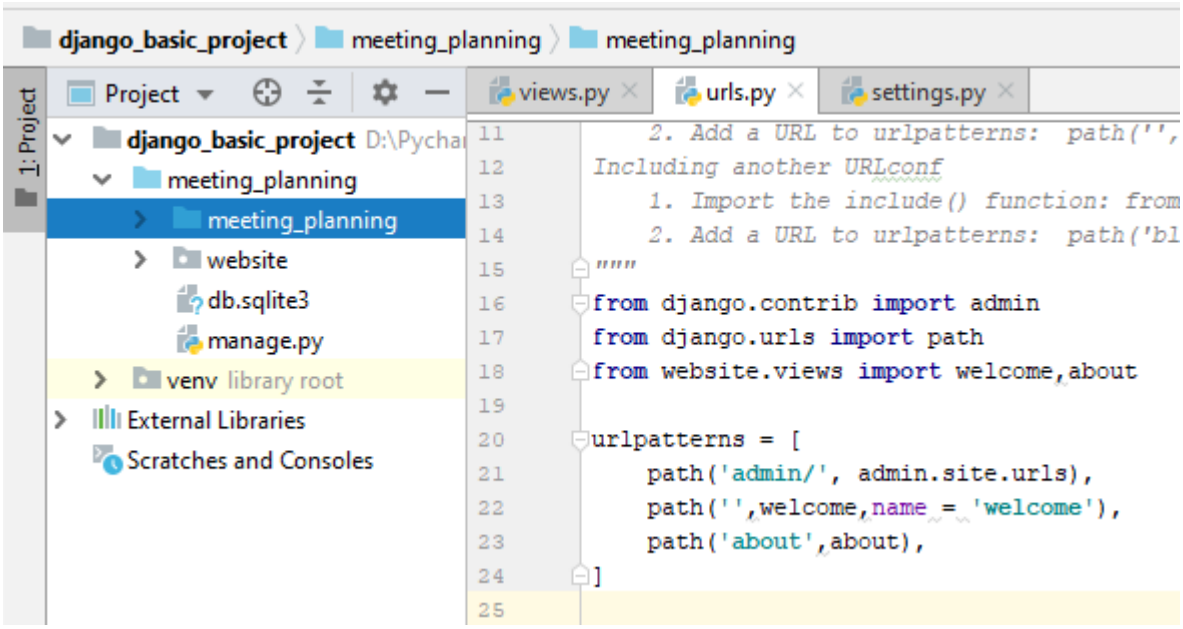
Python manage.py startapp website

```
(venv) D:\PycharmProjects\django_basic_project\meeting_planning>python manage.py startapp website
```

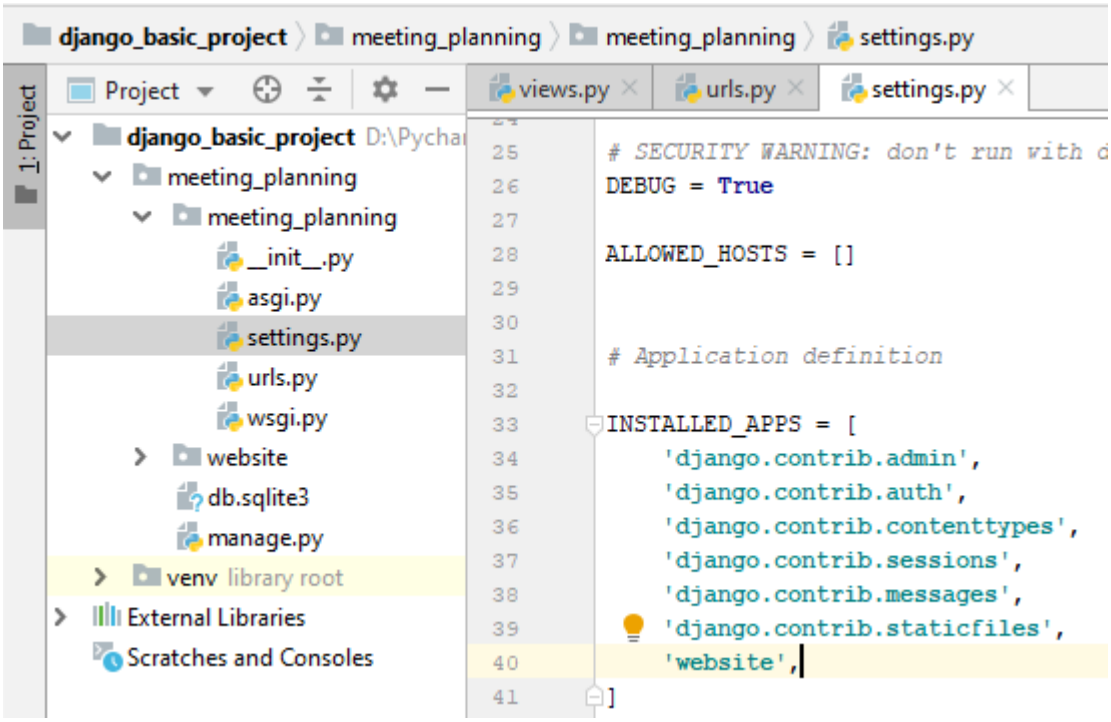
- After Executing the above command website app is created .under website app we can see different python file now let's open the 'views.py'.
- 'views.py' is which contains the web page code.
- In 'views.py' we have define a functions called welcome and about, both the functions get the request and returns response, we also have to import HttpResponse module as shown in below fig.



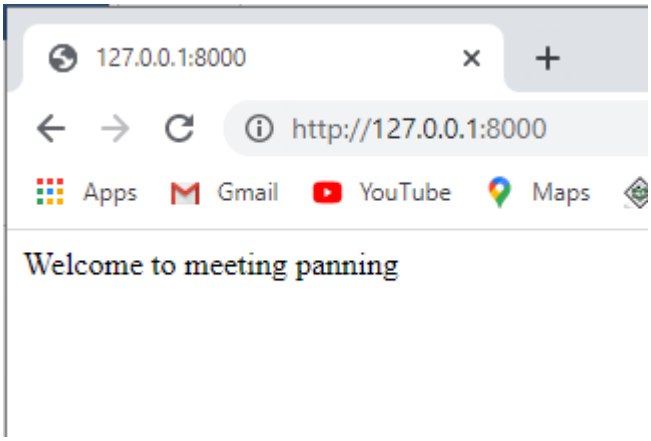
- The responsibility to send response by using urls, under the folder meeting_planning and under again meeting_planning folder this is our core concept folder. We can see python files like _init_.py, asgi.py, wsgi.py, setting.py, urls.py.
- In meeting_planning folder settings.py, urls.py files are backbone of our project, in urls.py file we are going to setup the path for both welcome function and about function and import welcome and about functions as shown below.
- To avoid import package error we have to make meeting_planner as source root.
- We have setup welcome path first argument as empty string it will open by default welcome page in the browser while launching server.



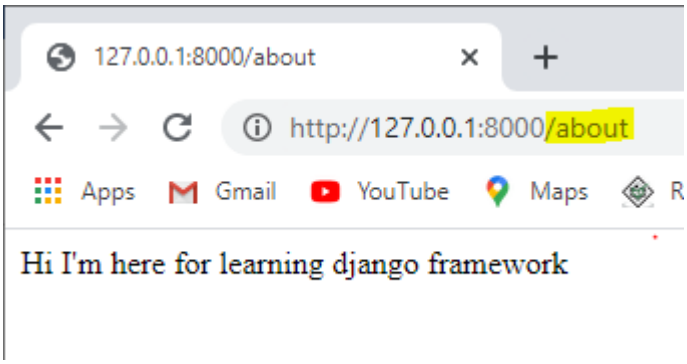
- Before we running the server it is mandatory to mention app name(website) in the settings.py under the list of INSTALLED APPS as shown below fig. If we skip this step the website app is not going to execute.



- Now we are going to run server and launch the browser, we can see the response message in the browser which we have mention in welcome function as shown below.



- If we have enter '/about' in URL it will call about function and get the response as shown below.



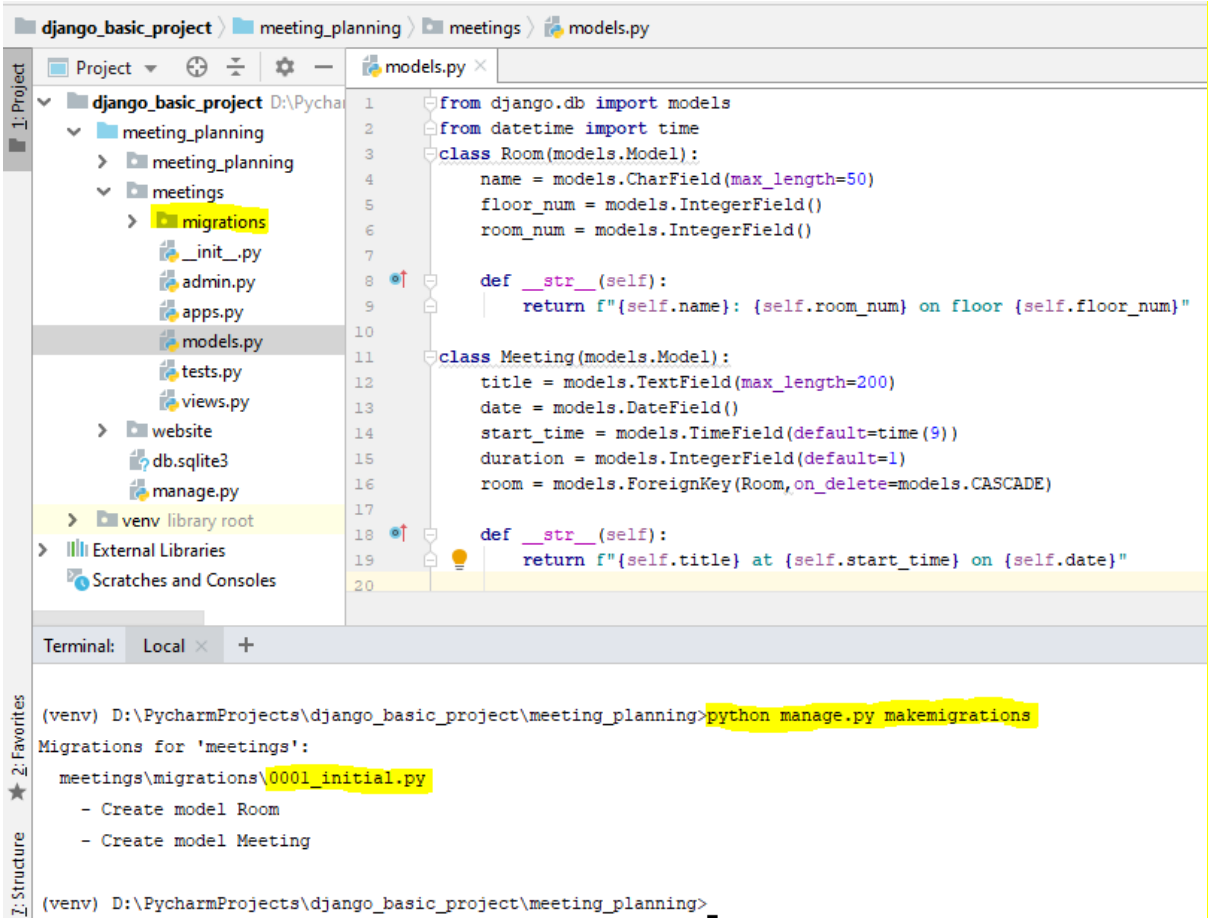
Create meeting app:

- Now we are going to create meeting app which contains all meeting and rooms for those by using below command.

Python manage.py startapp meetings

- We have to mention meeting app in list of INSTALLED APPS of setting.py under meeting_planning folder.
- Under the meetings app it contains migration folder and different files _init_.py, admin.py, models.py, tests.py, urls.py, views.py.
- Django follows the MTV pattern (Model, Template and View).
- Models.py consists of python classes that represent data to save data in database table.
- Migrations are the python script which helps us to keep database structure up-to-date with our model classes.
- Here we are going to write a code in two model classes which are Room and Meeting then we have make migrate them to database by using command in the terminal as shown below fig.

Python manage.py makmigrations



- We can see '0001_initial.py' in the migration folder which we want to migarate to the data base shell to migrate we are going to below command.

Python manage.py migrate

```
(venv) D:\PycharmProjects\django_basic_project\meeting_planning>python manage.py migrate
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, meetings, sessions
Running migrations:
  Applying contenttypes.0001_initial... OK
  Applying auth.0001_initial... OK
```

Connect db.sqlite3:

- To connect to sqlite3 and see Meeting table and Room table by using below commands as shown in below fig.

Python manage.py dbshell
.tables (to see tables in db.)

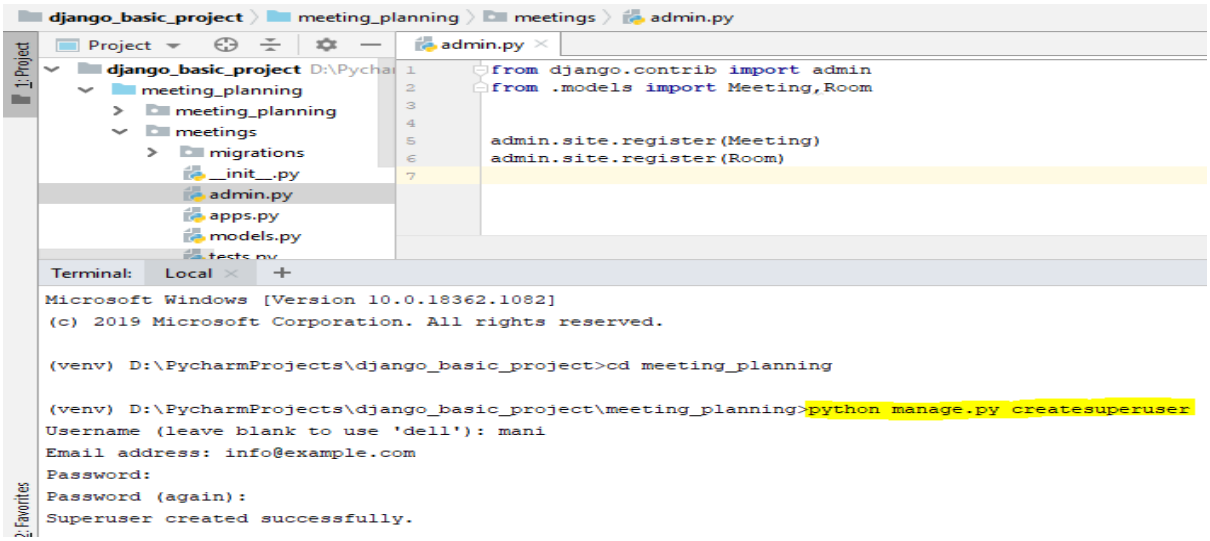
```
(venv) D:\PycharmProjects\django_basic_project\meeting_planning>python manage.py dbshell
SQLite version 3.32.2 2020-06-04 12:58:43
Enter ".help" for usage hints.
sqlite> .tables
auth_group                  django_admin_log
auth_group_permissions      django_content_type
auth_permission             django_migrations
auth_user                   django_session
auth_user_groups            meetings_meeting
auth_user_user_permissions  meetings_room
sqlite>
```

- Here we going to see two tables which has prefix of app name meetings_meeting and meetings_room.
- To exit from database enter ‘ .exit ’.

Create an Admin Interface:

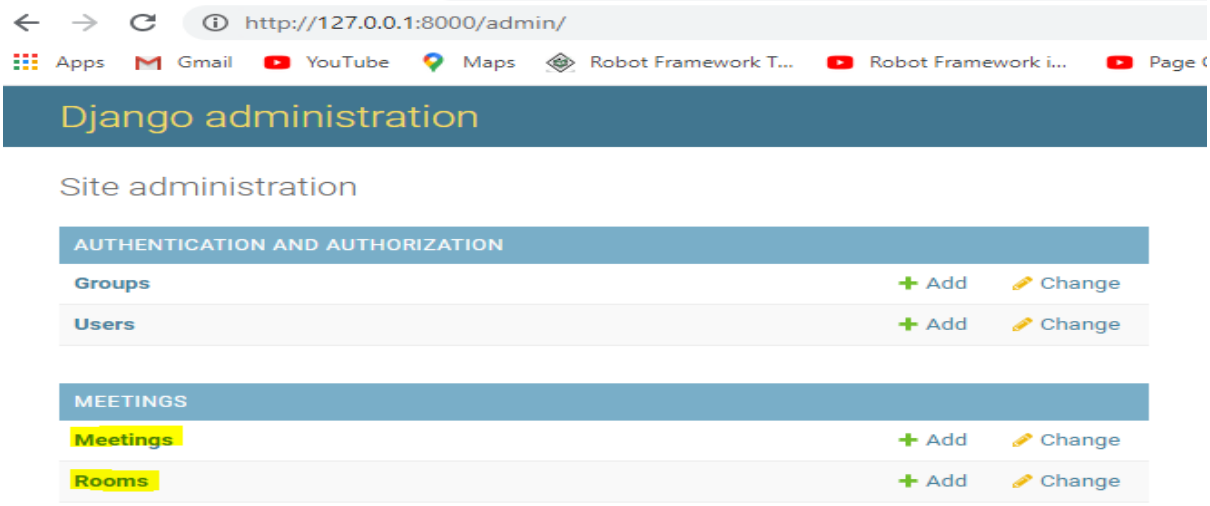
- Admin interface is very convenient and powerful, it has nice way to manage data in site meant to use only administrators**.**
- To create an admin lets open admin.py file in meetings app write the code as shown and enter the command and give the credentials as shown in below fig.

Python manage.py createsuperuser

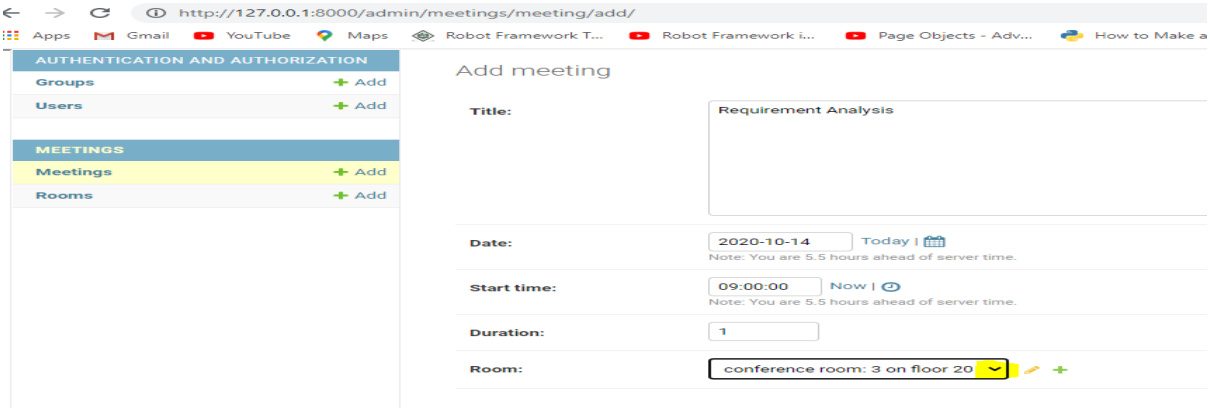


Create Meetings Through Admin Interface:

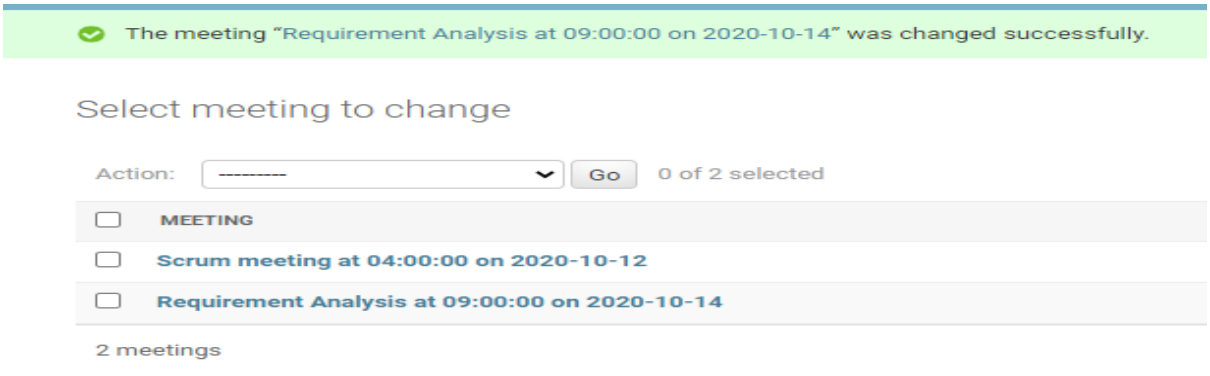
- Now are going to create two meetings in the meeting using admin credentials.
- Firstly run the development server and enter the following URL (http://127.0.0.1:8000/admin) in browser give your credentials .we can see administrator page as shown below.



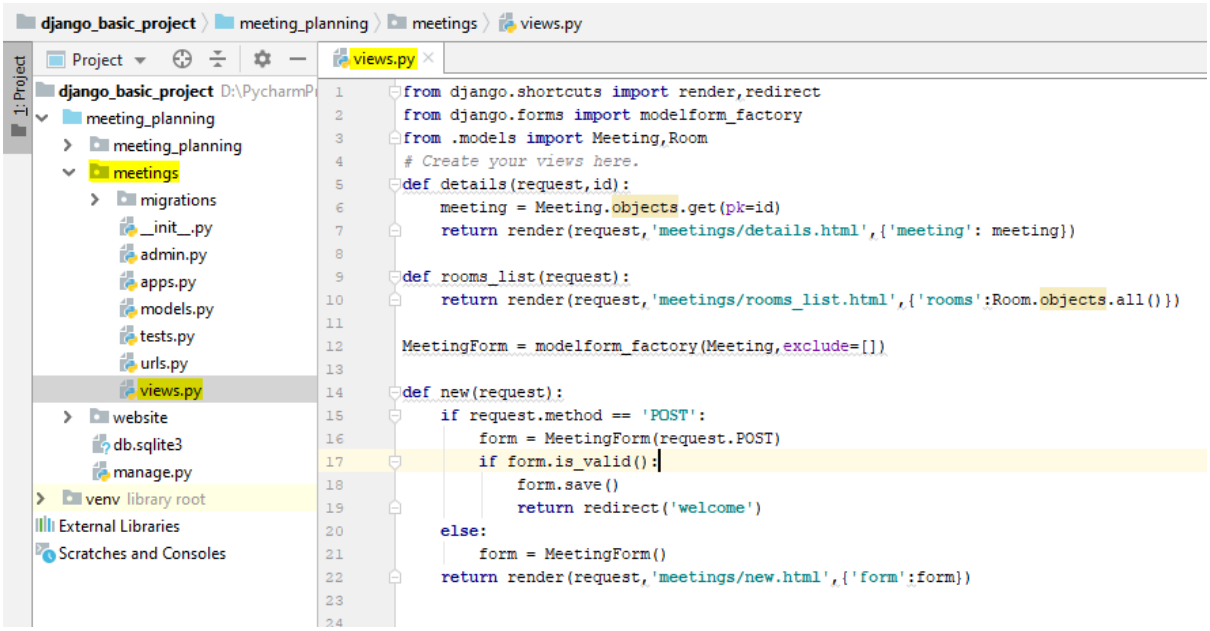
- By clicking on +Add Option of Rooms menu and we have created two room (conference room and auditorium room)
- By clicking on +Add Option of Meetings menu and we have created two meetings(Requirement analysis meeting and Scrum meeting) under the above created rooms by clicking on dropdown option of room and click on save option as shown below fig.



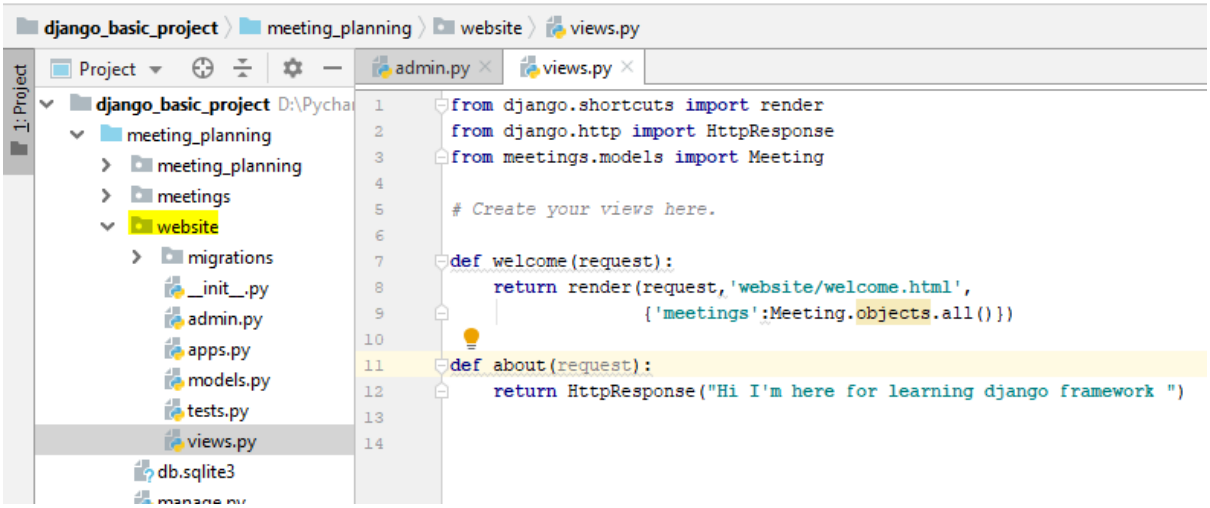
- After click save you are going to see below page.



- Now we are creating views function under meetings/views.py file, here we have define three view functions like details, rooms_list and new ,also we have to import the model classes , modelform_factory and redirect modules as shown in below fig.

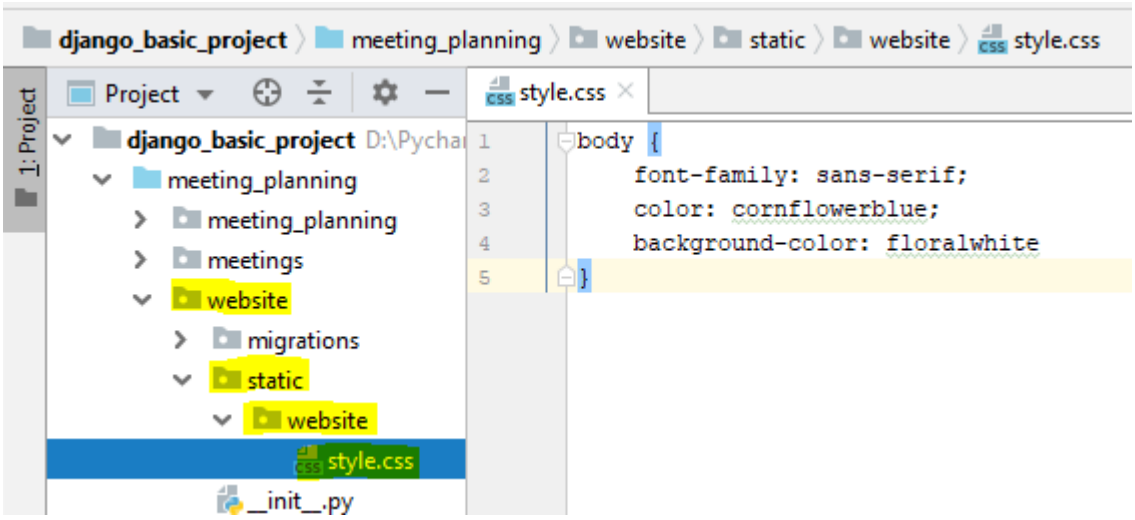


- Now we are modifying the welcome view function under website app by changing HttpRequests to render because for better code to call from browser and to give path for templates as shown in below fig.

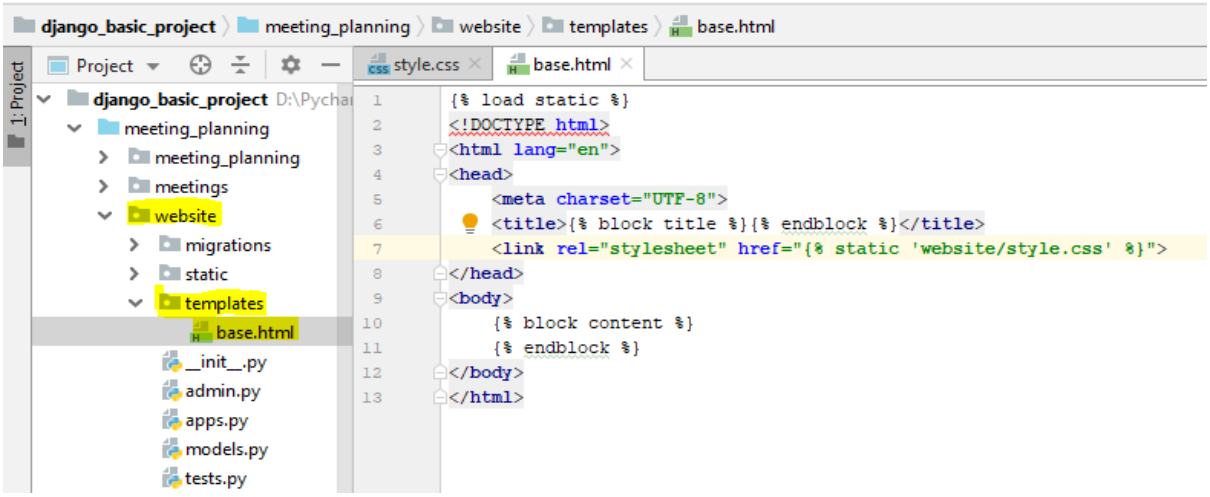


Create templates and static context:

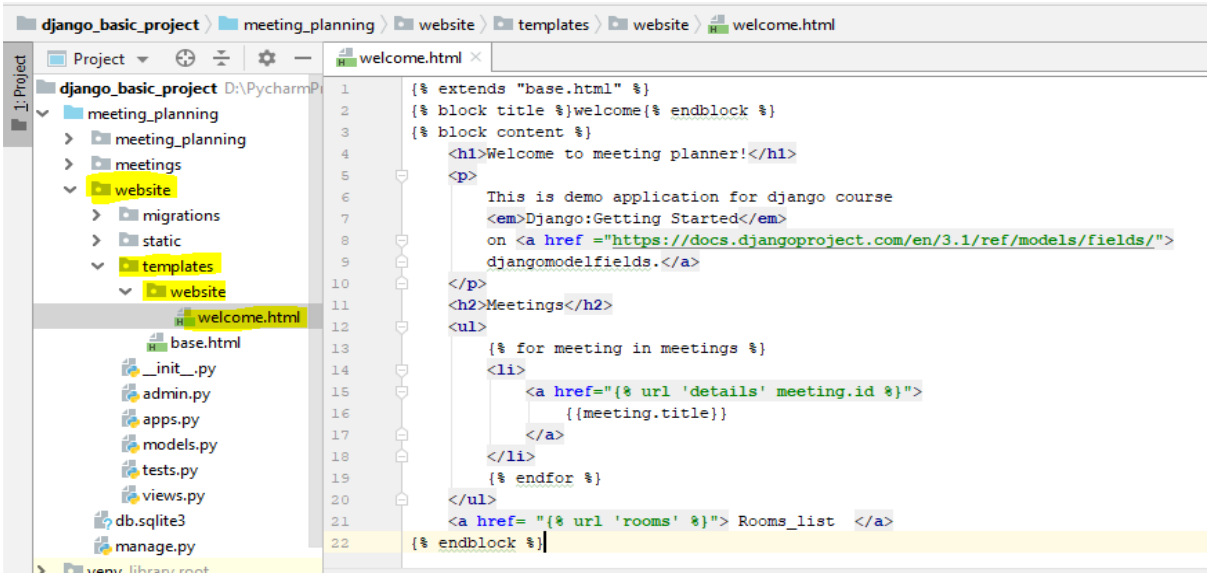
- Templates consists html files in django project.
- Static context consists css styles for web pages. For that we are creating style.css file
- Crate a folder under website app and named it as ‘static’, under the ‘static ‘folder create an another folder named it as ‘website’ and again under ‘website’ folder create ‘style.css’ file and write the code as shown in below fig.



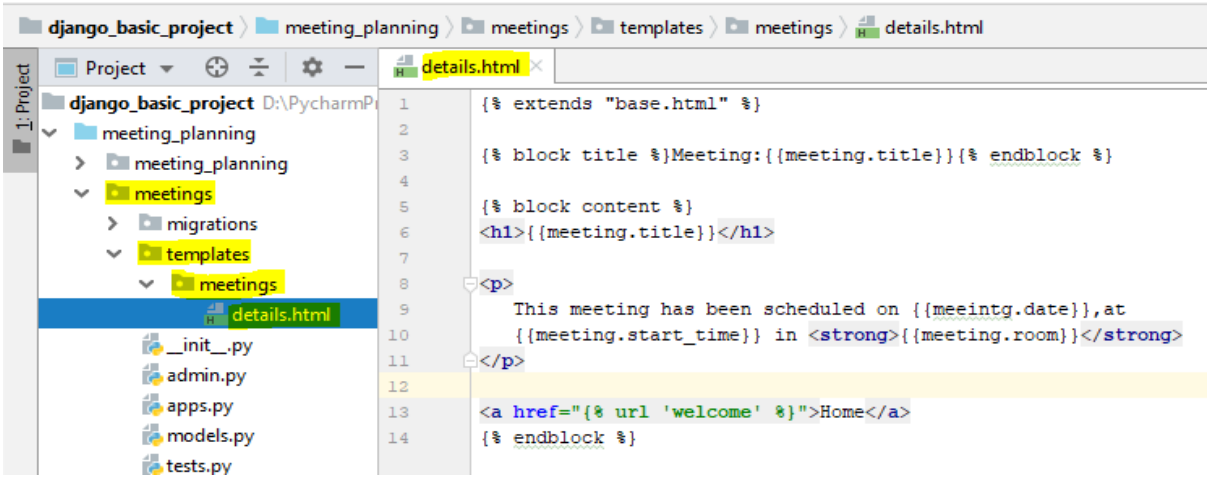
- Firstly we are creating base template and also we can inherit this template properties to all our apps templates in django project.
- Create a folder under the website app and named it as 'templates' ,under the 'templates' folder create 'base.html' file and write the code as shown below fig.

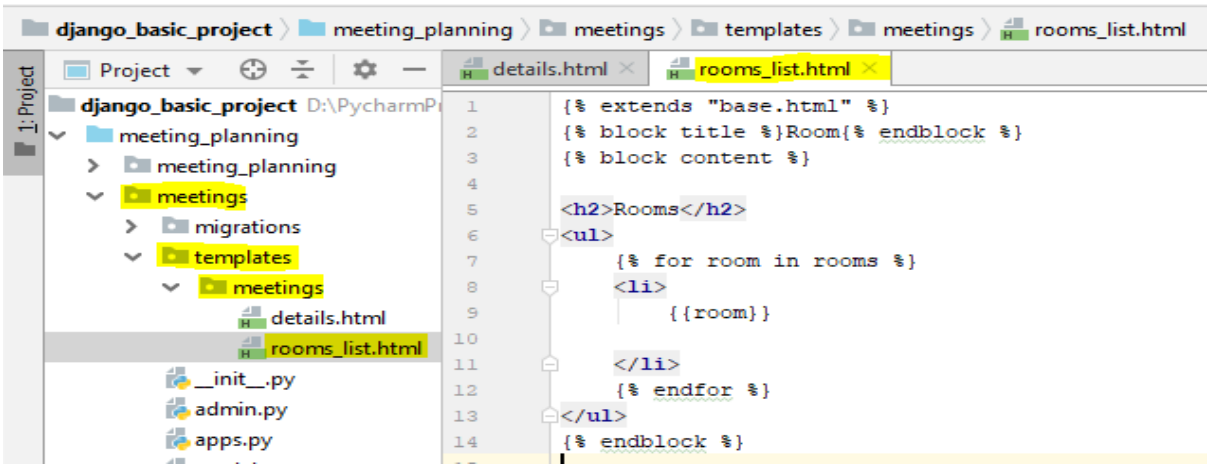


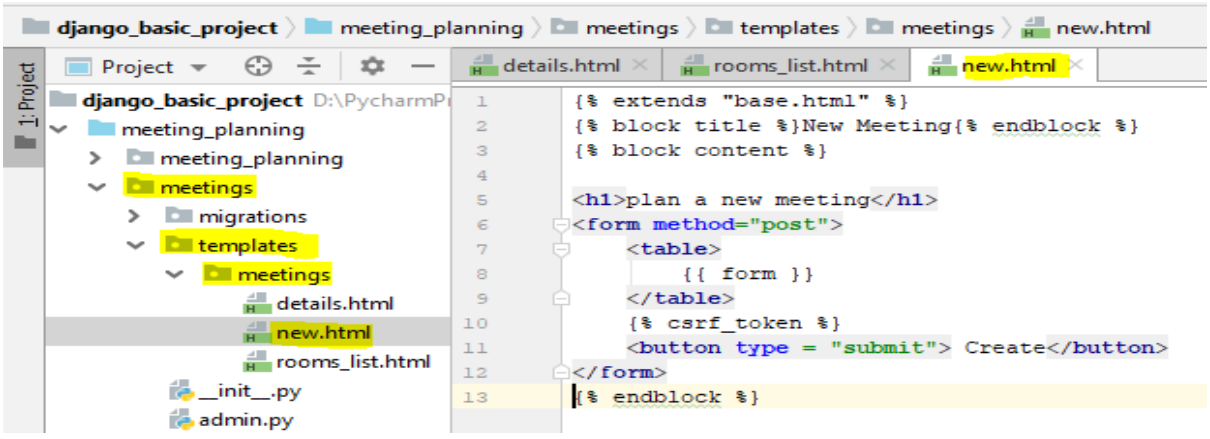
- Now create a folder under templates folder and named it as 'website' and again under website folder create an 'welcome.html' file and write the code as shown below fig.



- Now we are going to create templates for meetings app of every views function.
- Create a folder under the meetings folder and named it as ‘templates’ and again create a folder under the ‘templates’ folder and named it as ‘meetings’ and again under ‘meetings’ folder create three html files like ‘details.html’,’rooms_list.html’ and ‘new.html’. Write the code in respective html files as shown below figs.

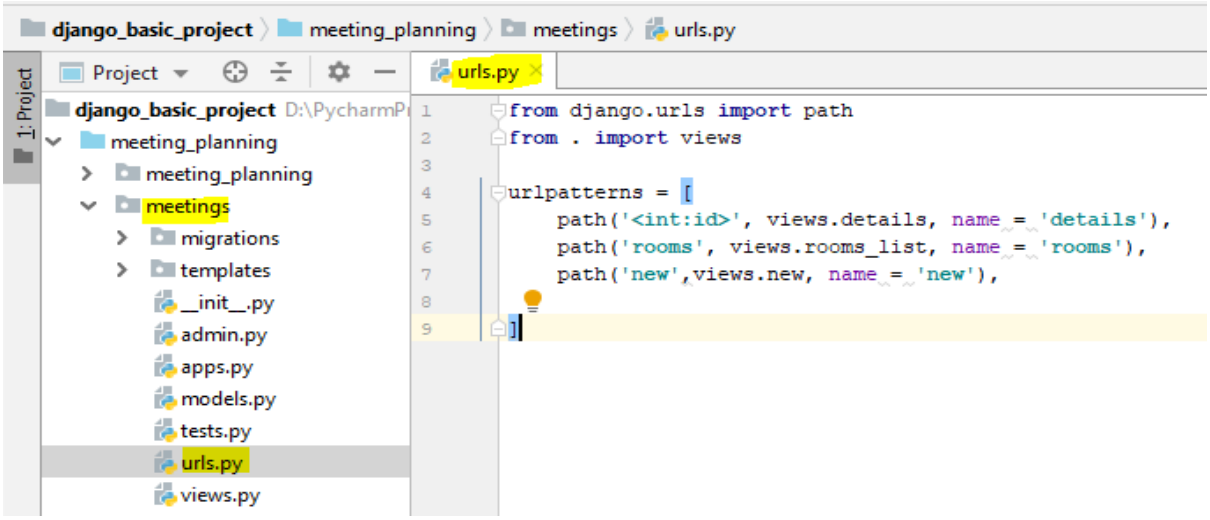




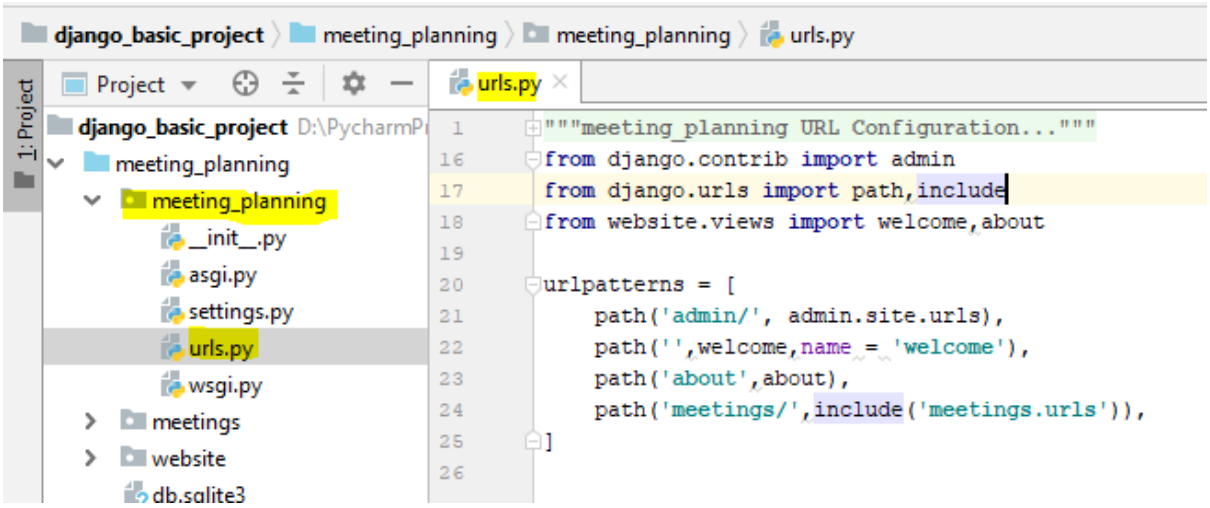


URL's and link bindings:

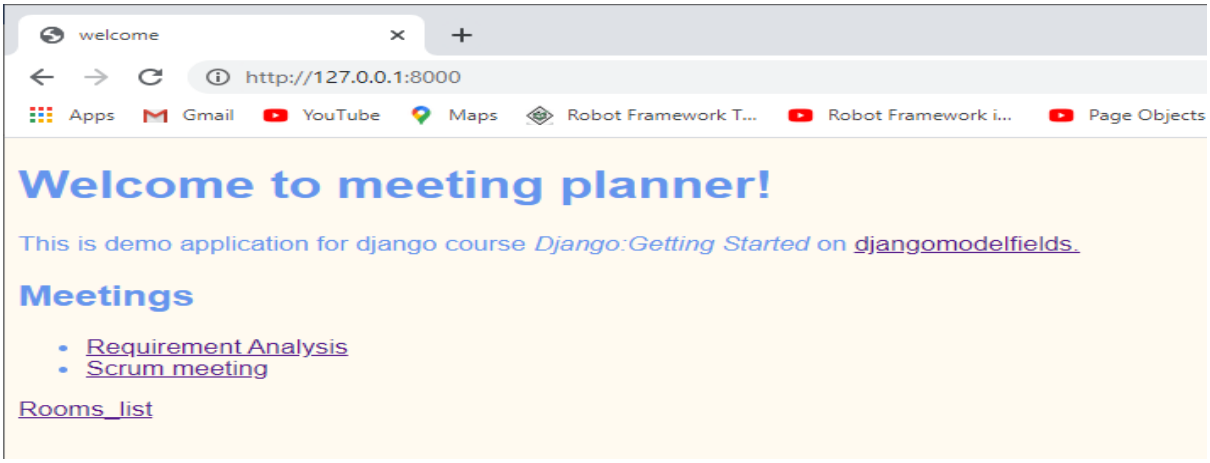
- Finally we are going to give path for view functions from core folder meeting_planning and meetings app folder.
- Before that we have to create 'urls.py' file in 'meetings' folder.
- Open the urls.py file under meetings folder and write the code as show below fig.



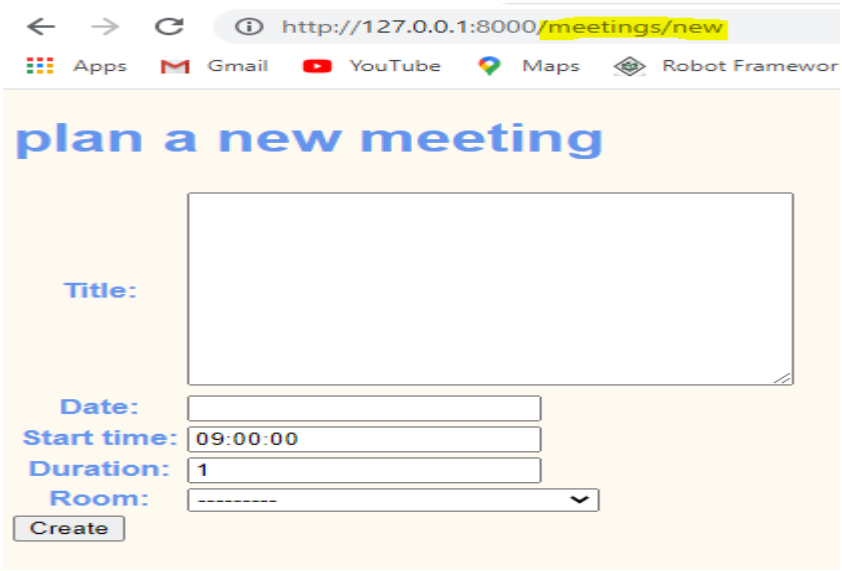
- Open the urls.py file under meeting_planning folder and write the code as shown below fig.



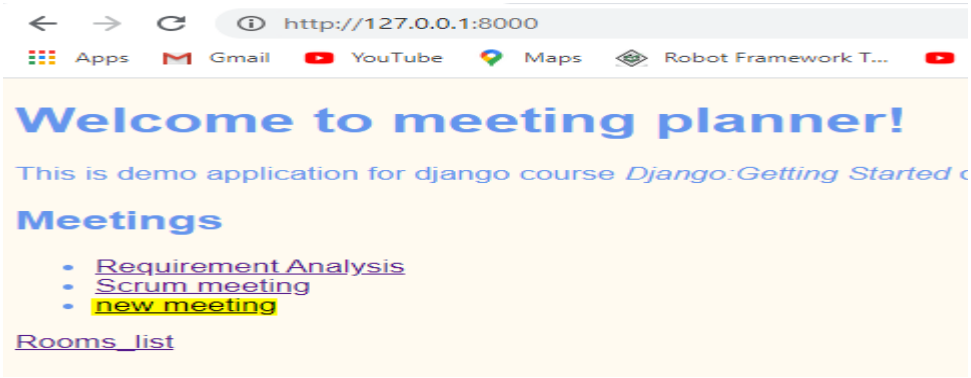
- Now run the server and launch the browser we can see as show in below fig.



- Click on 'Requirement Analysis' to see the detail about meeting.
- Click on 'Rooms_list' to see number of room are available.
- If you want add new meeting enter /meetings/new in URL and you can see page open as shown in below fig.



- Enter the respective valid details and click on create button.
- Now we can see added meeting in the welcome page as shown in below.



I hope this document will help you to understand basics about django python framework web development.