Python-Django

Table of Contents

Gouse Shaik

1. Install python
2. Install pycharm
3. Install pip (default it will comes with new versions of python)
4. Install Django with pip command.
5. Create a HelloWorld project in Django.

Install python

Install python is a straight forward.

And set the environment variable as python=”your installed python path”

python -m pip install --upgrade pip

```
C:\Users\80053806>python -m pip install --upgrade pip
Collecting pip
  Downloading https://files.pythonhosted.org/packages/54/2e/df11ea7e23e7e761d484ed3740285a34e38548cf2bad2bed3dd5768ec8b9
/pip-20.1-py2.py3-none-any.whl (1.5MB)
     |████████████████████████████████| 1.5MB 198kB/s
Installing collected packages: pip
  Found existing installation: pip 19.3.1
    Uninstalling pip-19.3.1:
      Successfully uninstalled pip-19.3.1
Successfully installed pip-20.1
```

Install Pycharm

Download community edition, and install it.

Install Django module using pip in python

pip install Django

```
C:\Users\80053806>pip install django
Collecting django
  Downloading https://files.pythonhosted.org/packages/9d/04/04abb097c84c770180eeebe7ed920ce42f9917ab5ad4de01ff8ed11bc25b
/Django-3.0.6-py3-none-any.whl (7.5MB)
     |████████████████████████████████| 7.5MB 363kB/s
Collecting pytz
  Downloading https://files.pythonhosted.org/packages/4f/a4/879454d49688e2fad93e59d7d4efda580b783c745fd2ec2a3adf87b0808d
/pytz-2020.1-py2.py3-none-any.whl (510kB)
     |████████████████████████████████| 512kB 63kB/s
Collecting asgiref~=3.2
  Downloading https://files.pythonhosted.org/packages/68/00/25013f7310a56d17e1ab6fd885d5c1f216b7123b550d295c93f8e29d372a
/asgiref-3.2.7-py2.py3-none-any.whl
Collecting sqlparse>=0.2.2
  Downloading https://files.pythonhosted.org/packages/85/ee/6e821932f413a5c4b76be9c5936e313e4fc626b33f16e027866e1d60f588
/sqlparse-0.3.1-py2.py3-none-any.whl (40kB)
     |████████████████████████████████| 40kB 97kB/s
Installing collected packages: pytz, asgiref, sqlparse, django
```

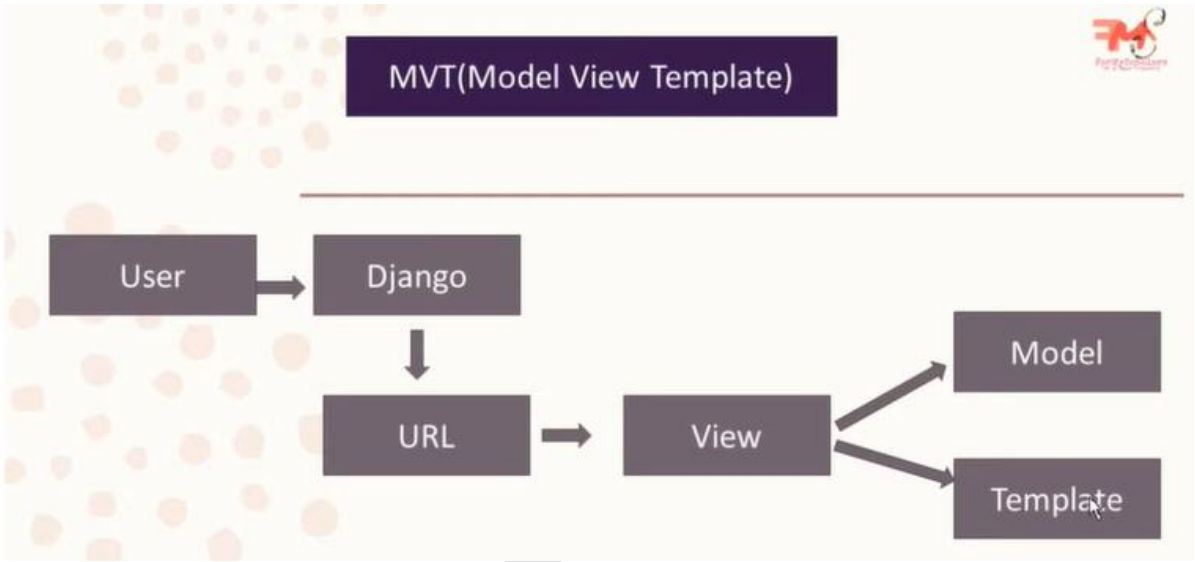
```
/sqlparse-0.3.1-py2.py3-none-any.whl (40kB)
     |████████████████████████████████| 40kB 97kB/s
Installing collected packages: pytz, asgiref, sqlparse, django
Successfully installed asgiref-3.2.7 django-3.0.6 pytz-2020.1 sqlparse-0.3.1
WARNING: You are using pip version 19.3.1; however, version 20.1 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command.
```

To check the Django version

Python -m Django --version

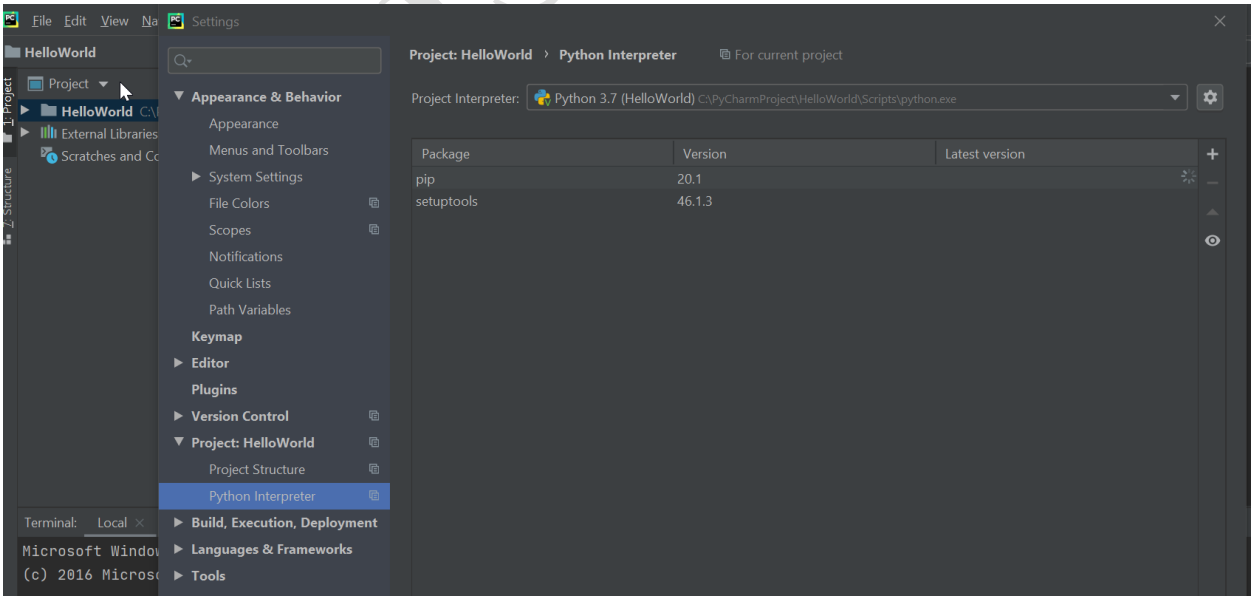
```
C:\Users\80053806>python -m django --version
3.0.6
```

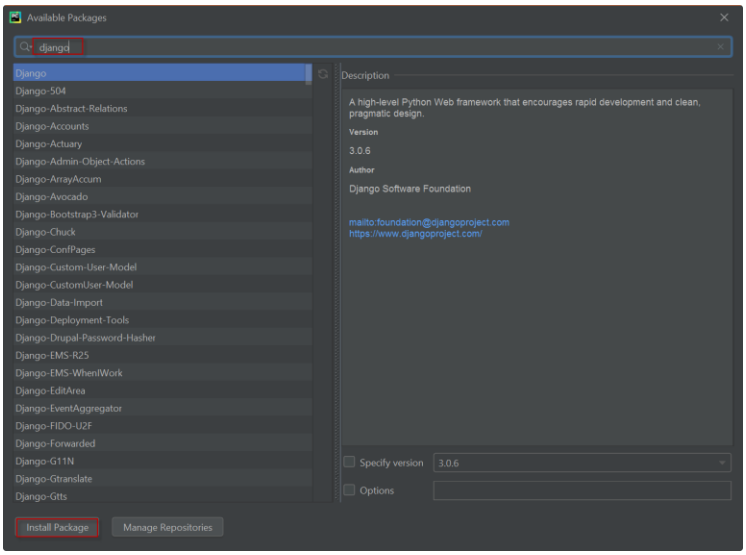
Django is a MVT (Model View Template)



Install Django plugin for your project in pycharm.

Go to file→ settings, click on project and right side click on + button for the installation of Django plugin in pycharm

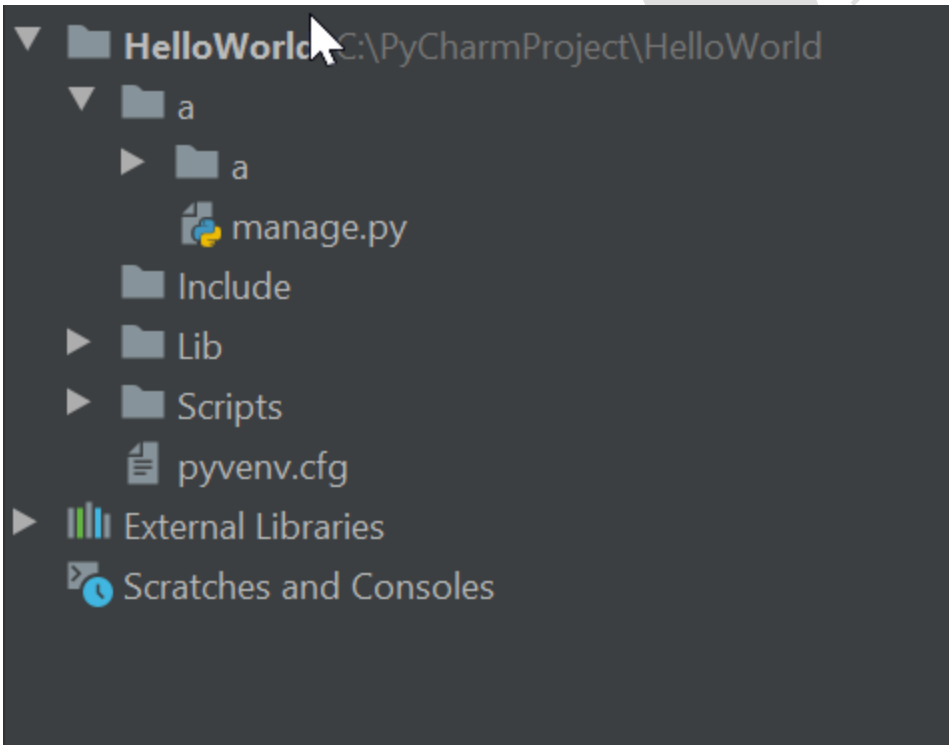




Click on Install Package

#django-admin startproject flipkart

#django-admin startproject a



(HelloWorld) C:\PyCharmProject\HelloWorld>django-admin startproject a

Now change the project directory you have created

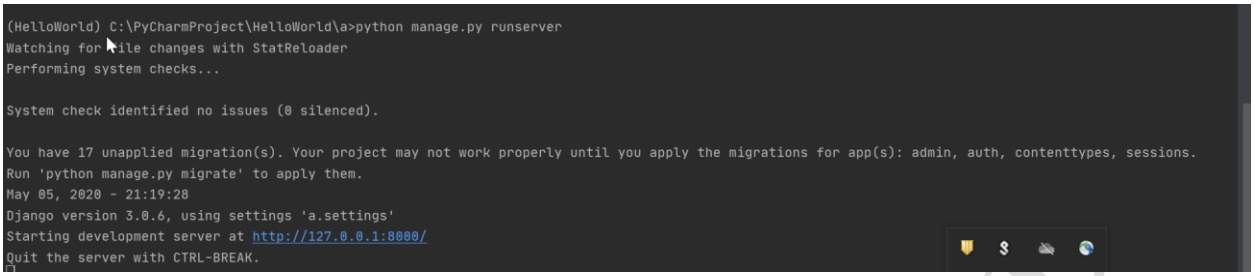
(HelloWorld) C:\PyCharmProject\HelloWorld>cd a

(HelloWorld) C:\PyCharmProject\HelloWorld\a>

Shaik

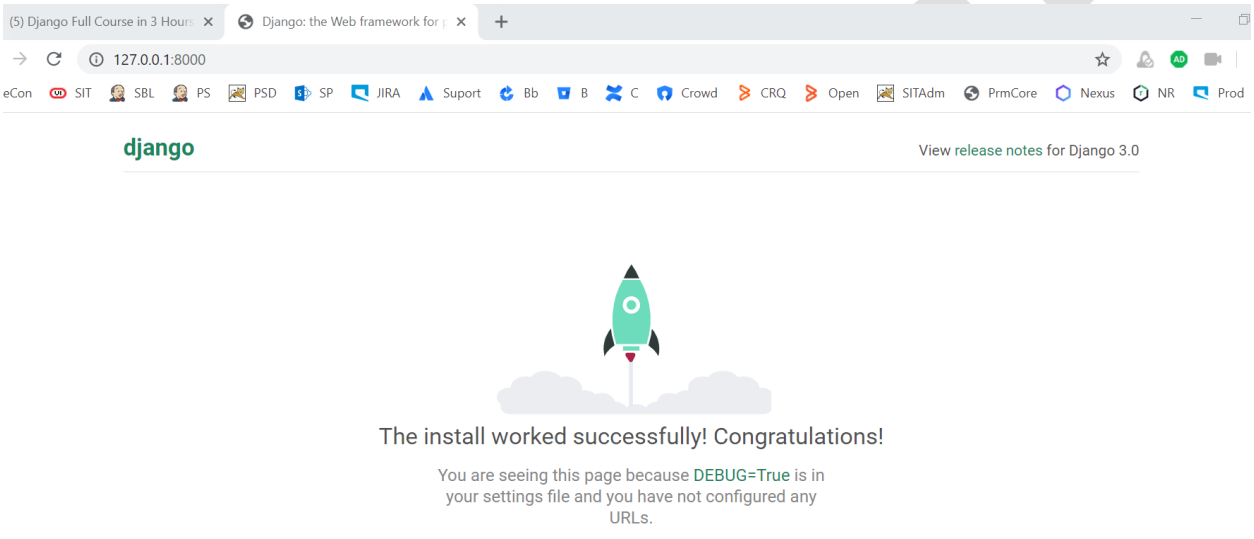
Now run the server http://127.0.0.1:8000/

#python manage.py runserver

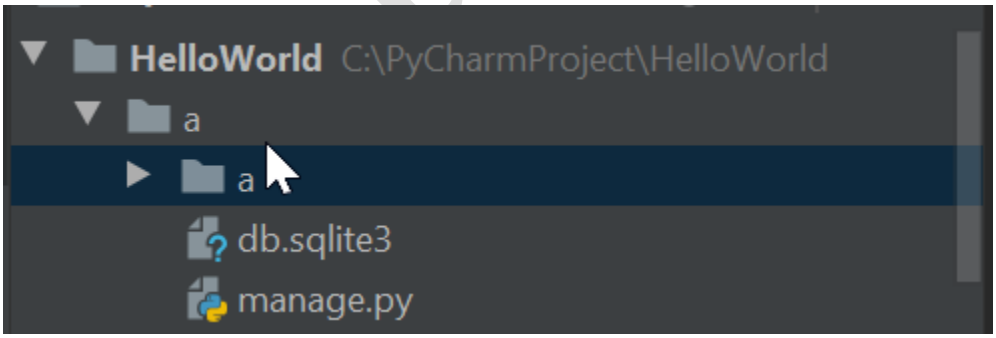


Now you can see the webserver is up and running with the below server url.

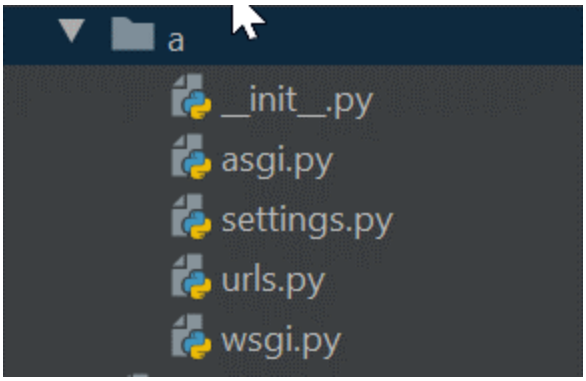
http://127.0.0.1:8000/



You can see the internal database created with sqlite3 in your project a directory



If you expand the a project directory, you can see the 5 files, which are mandatory for your intitial setup of django project.

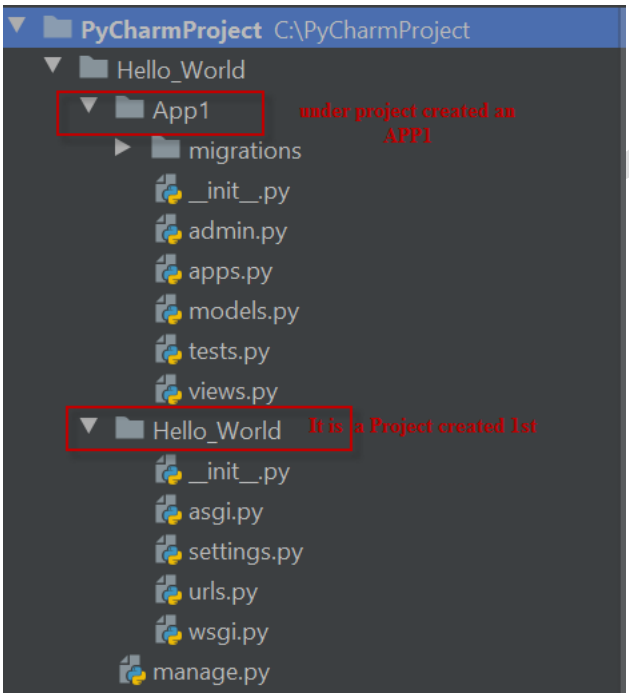


About .py files in project

Manage.py : it will be helpful to interact between differnet apps in your project.
__init__.py : as a package for the project
Settings.py : it contains database, template, password information and settings.
Urls.py : it will be used to map between differnet urls.
Wsgi.py : entry point for all webservers

#python manage.py startapp app2

Project structure of Django

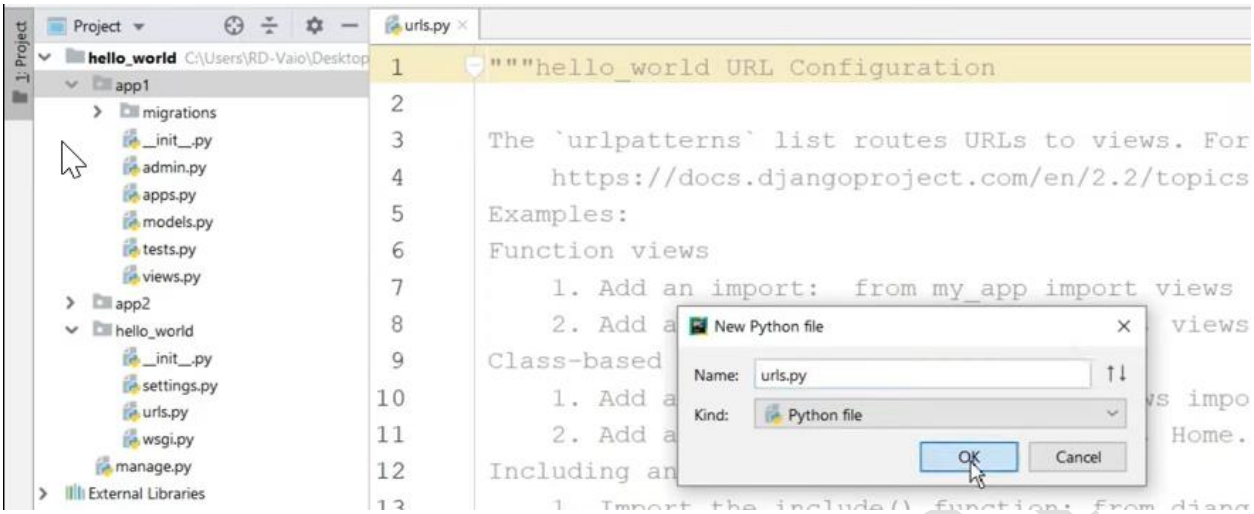


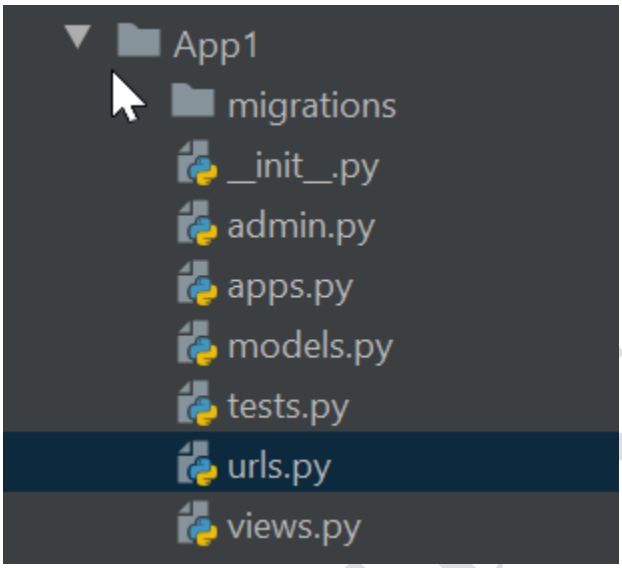
App1 is our application, now we are going to print “Hello World”.

Urls.py

So create Urls.py under the App1

Shaik





Path function contains 3 parameters

1. Endpoint parameter
2. Views object/function
3. Name of the view

And note import view.py as a module in this urls.py


```
from django.contrib import admin
from django.urls import path
from . import views
urlpatterns = [
    path('admin/', admin.site.urls),
    path('', views.home,name='home page'),
]
```

views.py

create a views.py to route the ruquest to httpResponse

```
from django.http import HttpResponse
from django.shortcuts import render


# Create your views here.
def home(request):
    return HttpResponse("$$$~Hello World~$$$")
```

HttpResponse

Import the HttpResponse module before creating any definition

urls.py

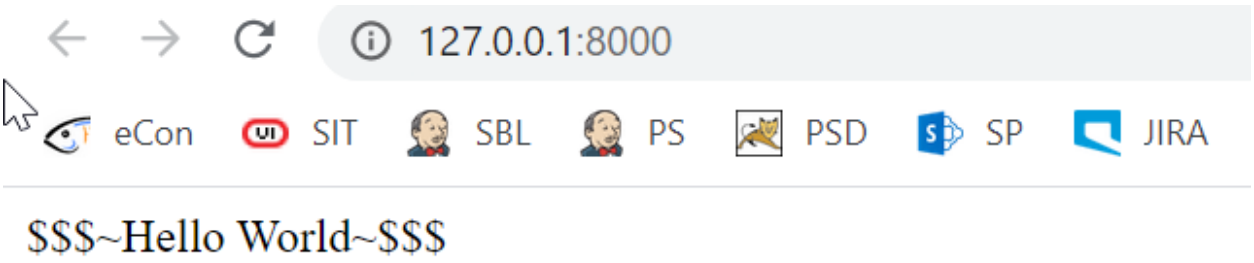
now to the poject urls.py and update the new app1 url to this hello-word project urls.py

```
from django.contrib import admin
from django.urls import path, include

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', include('App1.urls'))
]
```

now run the server

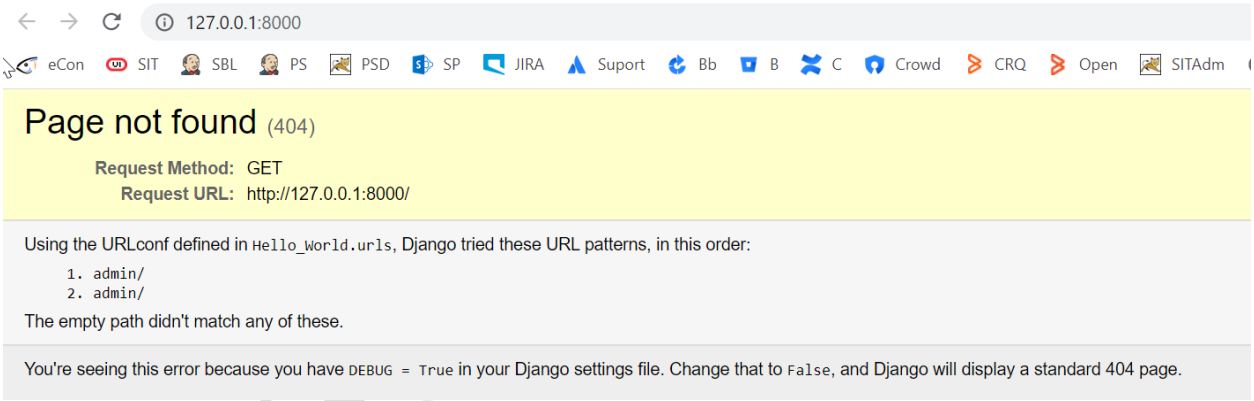
python manage.py runserver



If you can observe, there is no enpoint in this url.

If you update the urls.py with App1.urls, it will show the error. Why because we have configured for App2 not for App1.

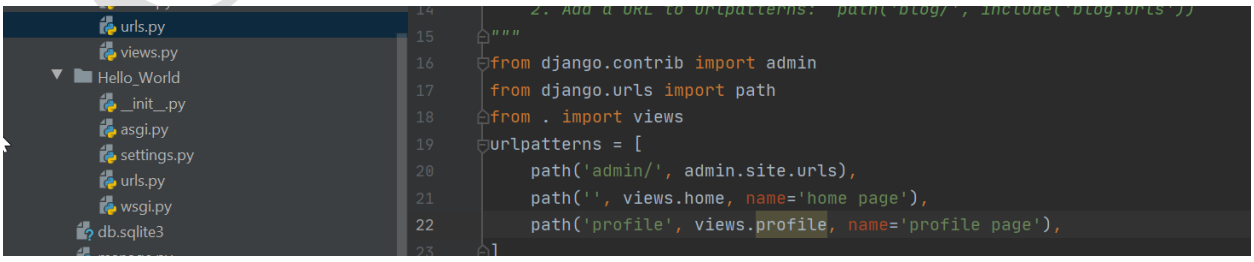
```
urlpatterns = [
    path('admin/', admin.site.urls),
    path('', include('App1.urls'))
]
```



So you will get the 404 page error.

Endpoit configuration

Now we cofigure the endpoint and let check.

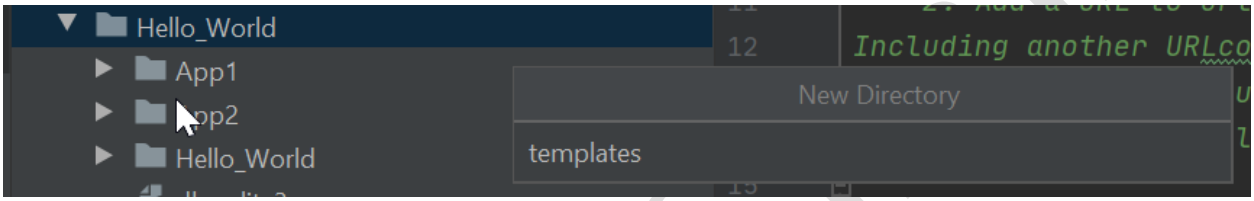


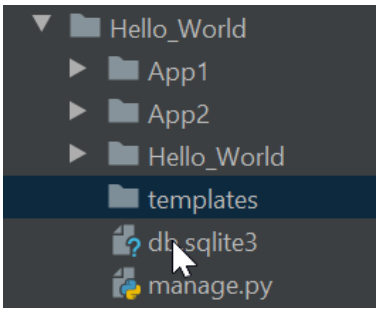
Instead of <tags> in views.py file, create html file seperately and call in your templates file. And calling in views.py file.

```
def home(request):
    return HttpResponse("<h1>Hello World<h1>")
def profile(request):
    return HttpResponse("profile page")
```

Templates

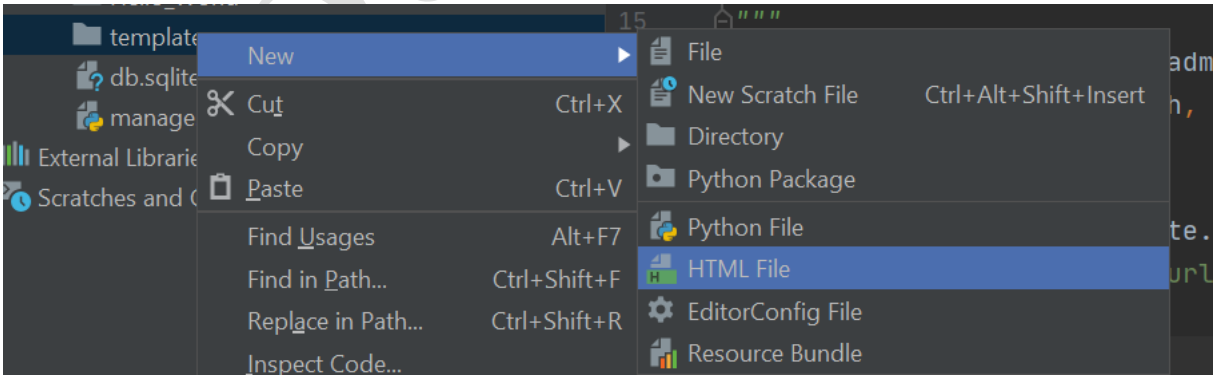
Make sure you have to create a template directory, under project directory(where manage.py available)

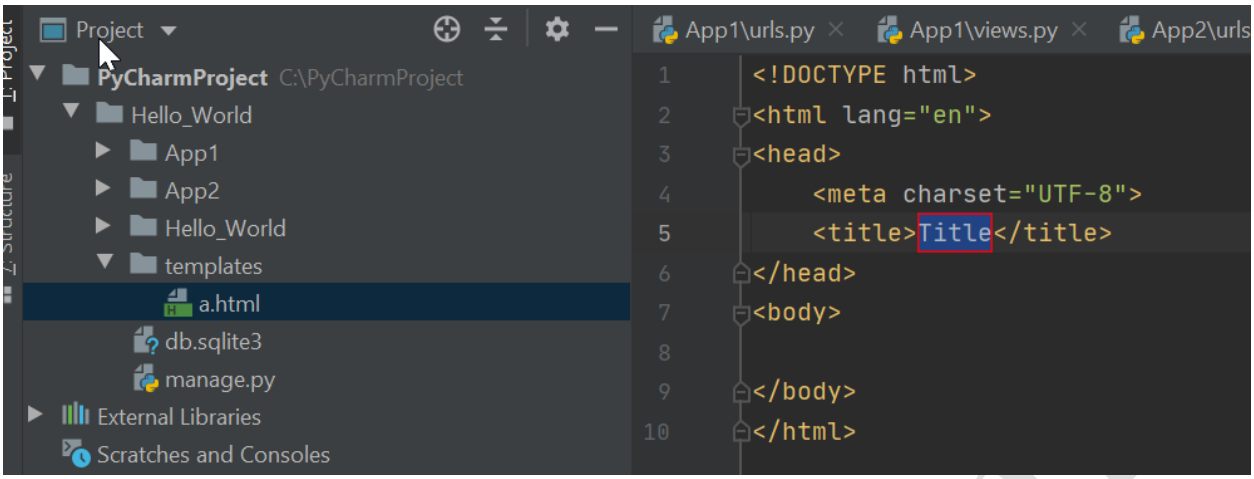




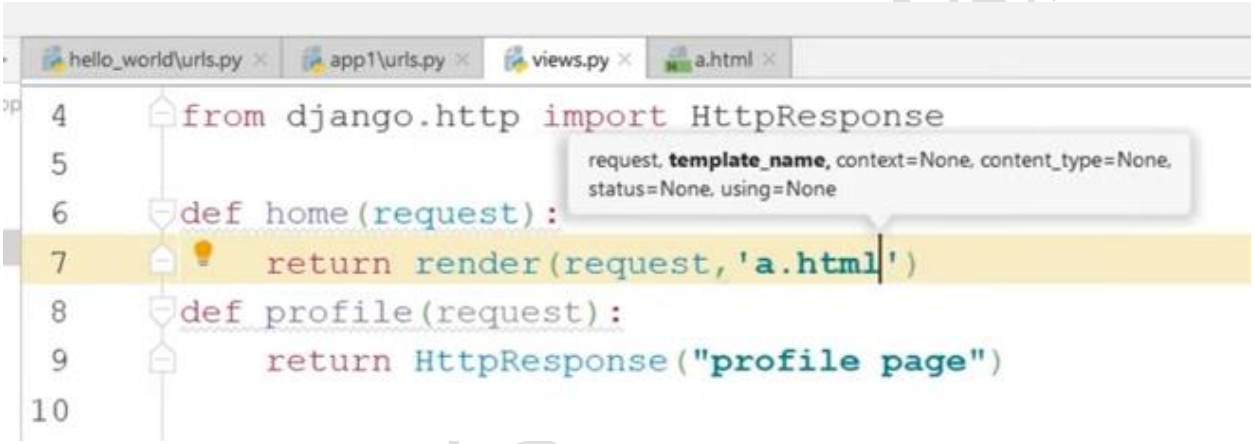
Html file

Now create an HTML file under templates directory





Now go to views.py, we have to remove the HttpResponse, and add the render for the request of a.html



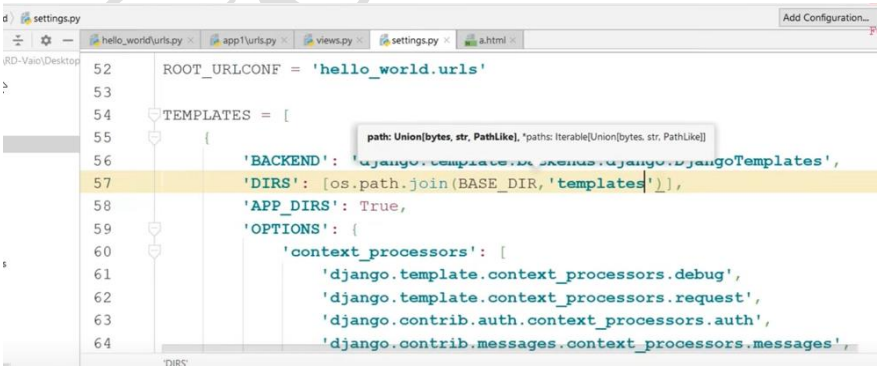
Settings.py

Important to update settings.py

Join the Project BASE_DIR and template directory with in settings.py @TEMPLATES section :

'DIRS': [os.path.join(BASE_DIR,'templates')],

Search for Templates: in settings.py



Defining dynamic values

Get the value from views.py

Call with render() function

```
def home(request):
    return render(request,'a.html',{'titles':'Django'})
def profile(request):
    return HttpResponse("profile page")
```

{'titles': 'Django'}

Goto html file call the {{titles}} variable in html file

```
<h1>hello {{titles}}</h1>
```

{{titles}} in html file

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Hello {{titles}}</title>
    <h1>Hello {{titles}}</h1>
</head>
<body>

<body>
</html>
```

GET Method : Not Secure

It is used to get info from the server

views.py | output.html | urls.py | a.html

```
7   def home(request):
8       return render(request,'a.html',{'titles':'Djang o','link':'http:/
9   def profile(request):
10      return render(request,'a.html',{'titles':'profile page','link':'h
11  def expression(request):
12      a=int(request.GET['text1'])
13      b=int(request.GET['text2'])
14      c=a+2*b
15      return render(request,'output.html',{'result':c})
16
```

App1\views.py | App2\urls.py | App2\views.py | output.html | settings.py | Hello_World\urls.py | a.html

```
<body>
    <body bgcolor="Yellow">
    <h1>Hello {{titles}}</h1>
    <a href="{{link}}">Home</a>

    <form action="adding">
        First Value:<input type="text" name="FirstValue">
        Second Value:<input type="text" name="SecondValue">
        <input type="submit">
    </form>

</body>
</html>
```

App1\views.py | App2\urls.py | App2\views.py | output.html | se

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Title</title>
</head>
<body>
output: {{result}}
</body>
</html>
```

App1\views.py | App2\urls.py | App2\views.py | output.html | settings.py | Hello_World\urls.py

```python
def profile(request):
    # return HttpResponse("$$$~Profile Page~$$$")
    return render(request, 'a.html', {'titles': 'Search profile', 'link': 'http://google.com/'})


def adding(request):
    a = int(request.GET['FirstValue'])
    b = int(request.GET['SecondValue'])
    c = a + 2 * b
    return render(request, 'output.html', {'result': c})
```

App1\views.py | App2\urls.py | App2\views.py | output.html | settings.py

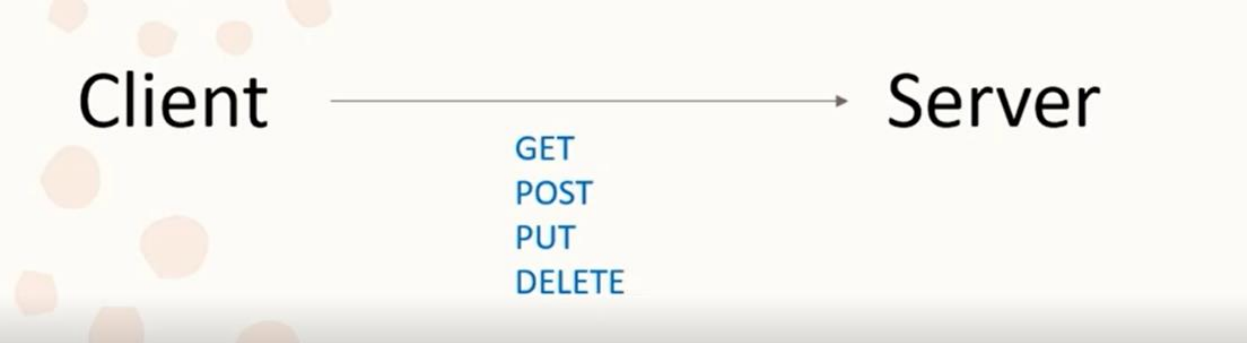
```python
    2. Add a URL to urlpatterns:  path('blog/', include('blog.urls'))
"""
from django.contrib import admin
from django.urls import path
from . import views
urlpatterns = [
    path('admin/', admin.site.urls),
    path('', views.home, name='home page'),
    path('profile', views.profile, name='profile page'),
    path('adding', views.adding, name='expression value')
]
```

Http Methods: used for request and response

Used in order to fetch the data and in order response the client.

1. Request Method [GET]
2. Response Method [HTTP]

GET Method: use for to fetch the data from the client to server



HTTP Methods:

- GET : fetch the data from the server to client
- POST : send the data
- PUT : put the data on some server
- DELETE : delete the data on server from the client side

POST Method : More Secure

Now Working with POST Method.

```
def expression(request):
    a=int(request.post['text1'])
    b=int(request.post['text2'])
    c=a+2*b
    return render(request,'output.html',{'result':c})
```

CSRF Error : is nothing an attach

- The view function passes a request to the template's render method.
- In the template, there is a {% csrf_token %} template tag inside each POST form that targets an internal URL.

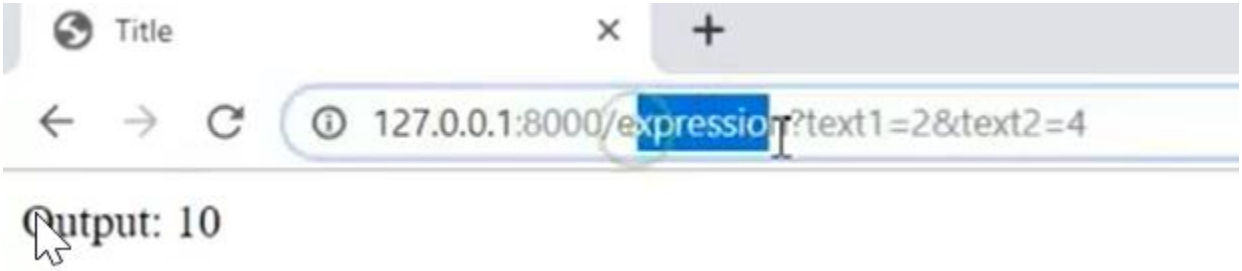
Add the {% csrf_token %} in your template as follows

```
<body bgcolor="yellow">
<form action="expression" method="post">
    {% csrf_token %}
    1st Value:<input type="text" name="text1">
    2nd Value:<input type="text" name="text2">
    <input type="submit">
</form>
```

```
def expression(request):
    a=int(request.POST['text1'])
    b=int(request.POST['text2'])
    c=a+2*b
```

Note:

Benefit of POST method is to avoid to show the values in address bar with expression values as follows in GET method



1:24:56

Gouse Shaik