virtualenv create (folder name can be anything)

why we use virtualenv????

Virtual environment will help us to run the same code if the version is changed or upgraded, generally one version code faces some difficulties in another version. Virtual environment helps us to remove the version dependency. So whenever we will create a project we should use virtual env to remove version dependencies.

```
Microsoft Windows [Version 10.0.17763.1]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Mr Hacker>cd Desktop

C:\Users\Mr Hacker\Desktop>virtualenv create
Using base prefix 'c:\\users\\mr hacker\\appdata\\local\\programs\\python\\python37-32'
New python executable in C:\Users\MRHACK~1\Desktop\create\Scripts\python.exe
Installing setuptools, pip, wheel...
done.

C:\Users\Mr Hacker\Desktop>cd
```

Cd create

```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.17763.1]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Mr Hacker>cd Desktop

C:\Users\Mr Hacker\Desktop>virtualenv create
Using base prefix 'c:\\users\\mr hacker\\appdata\\local\\programs\\python\\python37-32'
New python executable in C:\Users\MRHACK~1\Desktop\create\Scripts\python.exe
Installing setuptools, pip, wheel...
done.

C:\Users\Mr Hacker\Desktop>cd create

C:\Users\Mr Hacker\Desktop\create>
```

Cd Script

```
C:\Users\Mr Hacker\Desktop>cd create

C:\Users\Mr Hacker\Desktop\create>cd script
The system cannot find the path specified.

C:\Users\Mr Hacker\Desktop\create>cd ls
The system cannot find the path specified.

C:\Users\Mr Hacker\Desktop\create>cd Scripts

C:\Users\Mr Hacker\Desktop\create\Scripts>
```

activate

```
C:\Users\Mr Hacker\Desktop\create>cd script
The system cannot find the path specified.

C:\Users\Mr Hacker\Desktop\create>cd ls
The system cannot find the path specified.

C:\Users\Mr Hacker\Desktop\create>cd Scripts

C:\Users\Mr Hacker\Desktop\create\Scripts>cd activate
The directory name is invalid.

C:\Users\Mr Hacker\Desktop\create\Scripts>activate

(create) C:\Users\Mr Hacker\Desktop\create\Scripts>
```

pip install django

```
(create) C:\Users\Mr Hacker\Desktop\create\Scripts>pip install django
Collecting django
  Using cached https://files.pythonhosted.org/packages/a0/36/463632a2e9161a7e713488d719a280e8cb0c7e3a66ed32a32e801891ca
e/Django-2.2.7-py3-none-any.whl
Requirement already satisfied: pytz in c:\users\mrhack~1\desktop\create\lib\site-packages (from django) (2019.3)
Requirement already satisfied: sqlparse in c:\users\mrhack~1\desktop\create\lib\site-packages (from django) (0.3.0)
Installing collected packages: django
Successfully installed django-2.2.7
```

cd..

```
(create) C:\Users\Mr Hacker\Desktop\create\Scripts>cd..

(create) C:\Users\Mr Hacker\Desktop\create>
```

django-admin startproject crud

```
(create) C:\Users\Mr Hacker\Desktop\create>django-admin startproject crud

(create) C:\Users\Mr Hacker\Desktop\create>
```

cd crud

```
(create) C:\Users\Mr Hacker\Desktop\create>cd crud

(create) C:\Users\Mr Hacker\Desktop\create\crud>
```
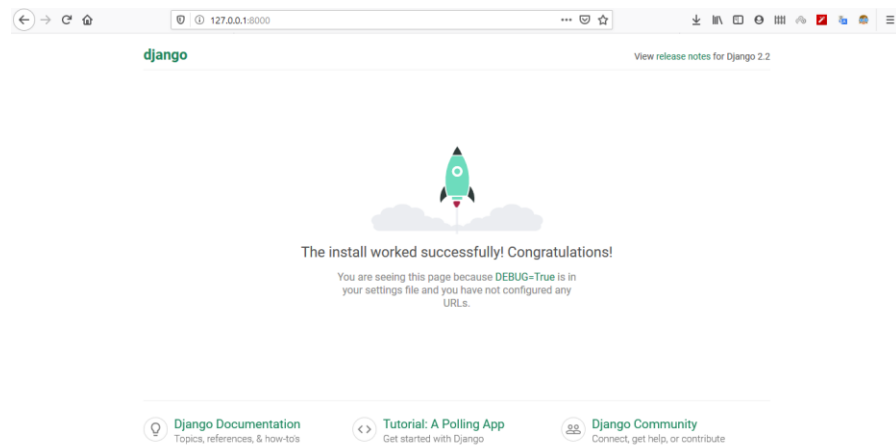
manage.py runserver

```
(create) C:\Users\Mr Hacker\Desktop\create\crud>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 proper
 auth, contenttypes, sessions.
Run 'python manage.py migrate' to apply them.
November 23, 2019 - 01:36:01
Django version 2.2.7, using settings 'crud.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

http://127.0.0.1:8000/

python manage.py startapp crudapp

```
Microsoft Windows [Version 10.0.17763.1]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Mr Hacker\Desktop\create\crud>python manage.py startapp crudapp

C:\Users\Mr Hacker\Desktop\create\crud>_
```

Name	Date modified	Type	Size
crud	11/23/2019 1:35 AM	File folder	
crudapp	11/23/2019 1:55 AM	File folder	
db.sqlite3	11/23/2019 1:36 AM	SQLITE3 File	0 KB
manage.py	11/23/2019 1:28 AM	Python File	1 KB

******output*****

Now go to settings.py file and add created app name in the INSTALLED_APPS list (Line 33)

```
INSTALLED_APPS = [
        'django.contrib.admin',
        'django.contrib.auth',
        'django.contrib.contenttypes',
        'django.contrib.sessions',
        'django.contrib.messages',
        'django.contrib.staticfiles',
        'crudapp',
```

manage.py runserver

```
C:\Users\Mr Hacker\Desktop\create\crud>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
 auth, contenttypes, sessions.
Run 'python manage.py migrate' to apply them.
November 23, 2019 - 01:58:58
Django version 2.2.5, using settings 'crud.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

Now go to views.py file to create a view. And then add

from django.shortcuts import render

from django.http import HttpResponse

 # Create your views here.

def index(request):

   return HttpResponse("Hello World!!!!!!")

```
                    settings.py                    views.py
1    from django.shortcuts import render
2    from django.http import HttpResponse
3
4    # Create your views here.
5
6    def index(request):
7        return HttpResponse("Hello World!!!!!!")
8    |
```

We just create a view in views.py and now go to urls.py and set thr URL to see the views in running server. Also need to import views from existing app

```
from django.contrib import admin
from django.urls import path
from crudapp import views

urlpatterns = [

    path('admin/', admin.site.urls),
    path('home/', views.index, name="index"),
]
```
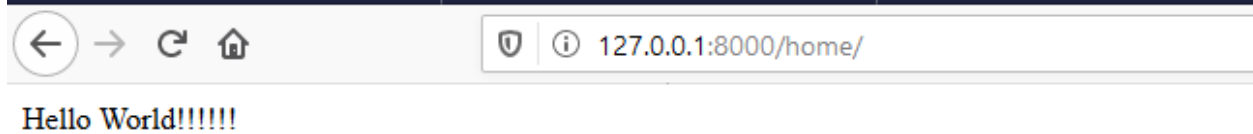
manage.py runserver

```
C:\Users\Mr Hacker\Desktop\create\crud>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
 auth, contenttypes, sessions.
Run 'python manage.py migrate' to apply them.
November 23, 2019 - 01:58:58
Django version 2.2.5, using settings 'crud.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

View is working now

Hello World!!!!!!

Now we are going to map all URL for our project. We will create a new URL file inside the app folder and then add the file location to urls.py file which inside the project folder.

```python
from django.contrib import admin
from django.urls import path
from django.conf.urls import include
from crudapp import views


urlpatterns = [
    path('admin/', admin.site.urls),
    path('crud/', include('crudapp.urls')),
]
```
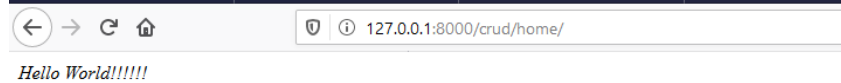
New urls.py file inside the app folder

```python
from django.urls import path
from django.conf.urls import include
from crudapp import views


urlpatterns = [

    path('home/', views.index, name="index"),
]
```

And the view

Create a new variable for template folder inside the settings.py file.

TEMPLATE_DIR = os.path.join(BASE_DIR,"templates")

```
import os

# Build paths inside the project like this: os.path.join(BASE_DIR, ...)
BASE_DIR = os.path.dirname(os.path.dirname(os.path.abspath(__file__)))
TEMPLATE_DIR = os.path.join(BASE_DIR,"templates")
```

And then add this variable inside below of the settings.py file

```
58    TEMPLATES = [
59        {
60            'BACKEND': 'django.template.backends.django.DjangoTemplates',
61            'DIRS': [TEMPLATE_DIR,],
62            'APP_DIRS': True,
63            'OPTIONS': {
```

Now in the views.py file create a view and return render the view to the index.html file

```
def index(request):
    my_dict = {'insert_me':"Hello i am from views.py!!"}
    return render(request,'index.html',context=my_dict)
```

```
1    from django.shortcuts import render
2    from django.http import HttpResponse
3
4    # Create your views here.
5
6    def index(request):
7        my_dict = {'insert_me':"Hello i am from views.py!!"}
8        return render(request,'index.html',context=my_dict)
9    |
```

Now create a index.html file inside the template folder. And then call the variable inside the html file.

{{insert_me}}

```
1    <!DOCTYPE html>
2    <html lang="en" dir="ltr">
3      <head>
4        <meta charset="utf-8">
5        <title>Python CRUD</title>
6      </head>
7      <body>
8        <h1>Welcoime to python CRUD System</h1>
9        {{insert_me}}
10
11      </body>
12   </html>
```

Now check the website. Boom!!!

127.0.0.1:8000/crud/home/

Welcoime to python CRUD System

Hello i am from views.py!!

Go to project folder and then go to urls.py >> remove path first parameter or value then go to app folder and remove as same.

```
from django.contrib import admin
from django.urls import path
from django.conf.urls import include
from crudapp import views

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

```
1   from django.urls import path
2   from django.conf.urls import include
3   from crudapp import views
4
5   urlpatterns = [
6
7       path('', views.index, name="index"),
8   ]
9
```

Now hit to the IP

← → C ⌂   🛡 ⓘ 127.0.0.1:8000

Welcoime to python CRUD System

Hello i am from crudapp/index.html!!

Now go again to views.py file and create a new view help

```
def help(request):
    my_help = {'help':"Help Page!!"}
    return render(request,'help.html',context=my_help)
```

Create a help.html file inside the template folder.

```
1   <!DOCTYPE html>
2   <html lang="en" dir="ltr">
3     <head>
4       <meta charset="utf-8">
5       <title>This is a Help Page</title>
6     </head>
7     <body>
8       <h1>{{help}} </h1>
9

10

11     </body>
12   </html>
13
```

Now go to the app folder and change the URL mapping into like this

```
1   from django.urls import path
2   from django.conf.urls import include
3   from crudapp import views
4
5   urlpatterns = [
6
7       path('', views.index, name="index"),
8       path('help', views.help, name="help"),
9   ]
```

Boom hit the ip/help

127.0.0.1:8000/help

Help Page!!

Create a New folder (static) inside the project folder and create a (images) folder inside the static folder and then go to settings file to add the line below the base directory.

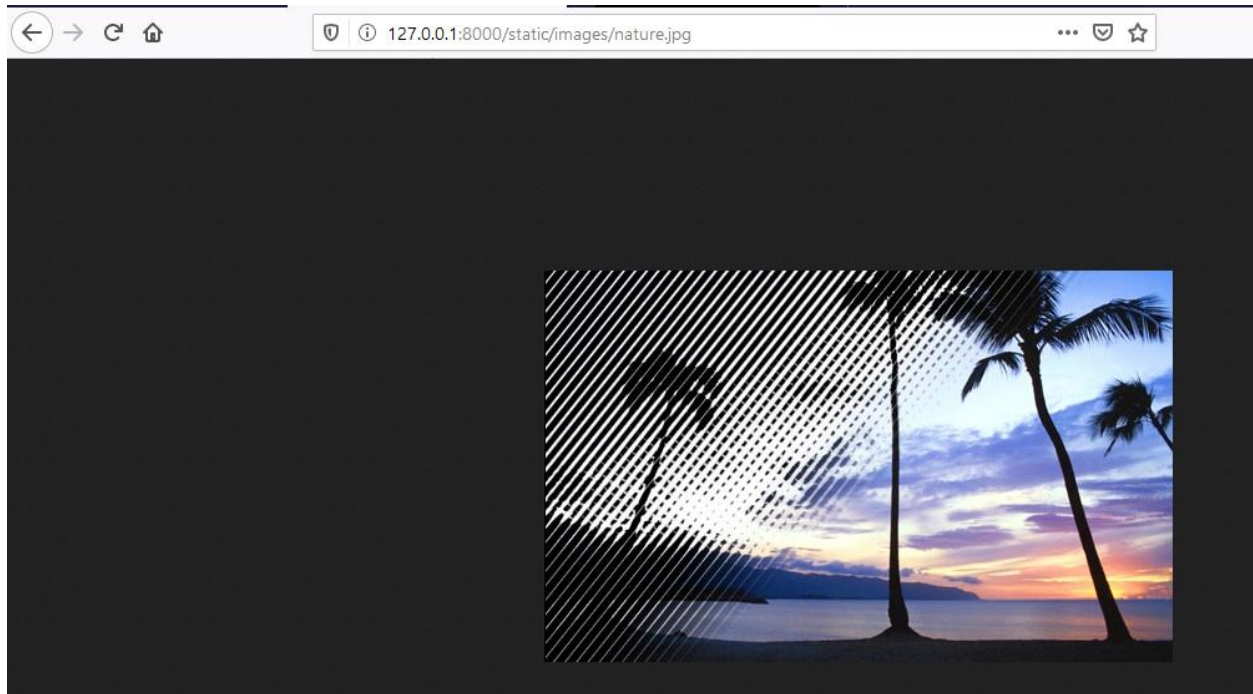STATIC_DIR = os.path.join(BASE_DIR,"static")

```
12
13    import os
14
15    # Build paths inside the project like this: os.path.join(BASE_DIR, ...)
16    BASE_DIR = os.path.dirname(os.path.dirname(os.path.abspath(__file__)))
17    TEMPLATE_DIR = os.path.join(BASE_DIR,"templates")
18    STATIC_DIR = os.path.join(BASE_DIR,"static")
19
20
```

Now add a variable STATICFILES_DIRS

STATICFILES_DIRS = [

   STATIC_DIR,

]

```
126
127    STATICFILES_DIRS = [
128        STATIC_DIR,
129    ]
130
```

Now save a image inside the images folder and run the server and visit the URL
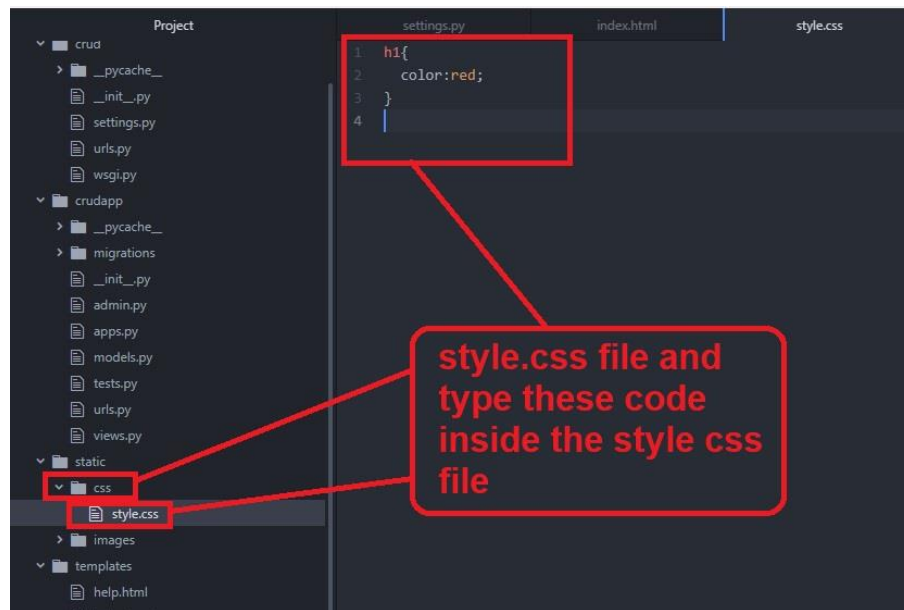
Now go to the index.html file and link the image to the index page.

```html
<!DOCTYPE html>
{%load staticfiles%}
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title>Python CRUD</title>

  </head>
  <body>
    <h1>Welcoime to python CRUD System</h1>
    <img src="{%static "images/nature.jpg"%}" alt="ulala">



  </body>
</html>
```

Output of the interface

127.0.0.1:8000

Welcoime to python CRUD System

How to add CSS in django

style.css file and type these code inside the style css file

And then add the CSS file inside the index.html file.


```html
<!DOCTYPE html>
{%load staticfiles%}
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title>Python CRUD</title>
    <link rel="stylesheet" href="{%static "css/style.css" %}"/>
  </head>
  <body>
    <h1>Welcoime to python CRUD System</h1>
    <img src="{%static "images/nature.jpg"%}" alt="ulala">


  </body>
</html>
```

carefull about marked section when add a css file

CSS output

127.0.0.1:8000

Welcoime to python CRUD System