

## **Creating Django Templates:**

It is not recommended to write an Html code inside Python Scripts (views.py file).

1. It reduces the readability because python code mixed with Html code.
2. No separation of roles python developer has to concentrate on both python code and html code.
3. It doesn't promote reusability of code. So, we can overcome these problems by separating html code into the separate html file.
4. This html file is nothing but template from the python file (views.py file). We can use these templates based on our requirement. So, we have to write templates at project level at only once and we can use this in multiple applications.

## **Creating a Project:**

Here we are developing some School Project

1. Creating a project SchoolProject in the current directory.

```
django-admin startproject SchoolProject
```

2. Move to that current directory

```
cd SchoolProject
```

3. To run the server if server is working

```
python manage.py runserver
```

4. Go to browser window and type this url <http://127.0.0.1:8000>

After checking the servers is worked and drag and drop the SchoolProject into the atom IDE.

## **Creating an First Application**

Here we are creating the first application name as admissions

1. Creating an application admissions  
django-admin startapp admissions

### **Settings.py**

- In this settings.py file and go to the Installed\_Apps and add the application 'admissions' and put the comma and save it.

## **Creating an Second Application**

Here we are creating the first application name as finance

1. Creating an application finance  
django-admin startapp finance

### **Settings.py**

- In this settings.py file and go to the Installed\_Apps and add the application 'finance' and put the comma and save it.

1. Go to main folder of SchoolProject and create new folder name templates.
2. Click templates and create new folder and type admissions.
3. Click templates and create new folder and type finance.
4. Click an admissions folder and create new file add addadmission.html

5. Click an admissions folder and create new file add admissionreport.html
6. Click an finance folder and create new file add feecollection.html
7. Click an finance folder and create new file add feeduesreport.html
8. Click an finance folder and create new file add feecollectionreport.html

### **addadmission.html**

```
<html>

<head>

<title>add admission page</title>

</head>

<body>

<h1>Hello this is add admission page</h1>

<h3>Please fill below form</h3>

</body>

</html>
```

### **admissionreport.html**

```
<html>

<head>

<title>admission report page</title>

</head>

<body>
```

```
<h1>Hello this is admission report page</h1>
```

```
<h3>Please fill below form</h3>
```

```
</body>
```

```
</html>
```

### **feecollection.html**

```
<html>
```

```
<head>
```

```
<title>fee collection page</title>
```

```
</head>
```

```
<body>
```

```
<h1>Hello this is fee collection page</h1>
```

```
<h3>Please fill below form</h3>
```

```
</body>
```

```
</html>
```

### **feeduesreport.html**

```
<html>
```

```
<head>
```

```
<title>fee dues report page</title>
```

```
</head>
```

```
<body>
```

```
<h1>Hello this is fee dues report page</h1>
```

```
<h3>Please fill below form</h3>
```

```
</body>
```

```
</html>
```

### **feecollectionreport.html**

```
<html>
```

```
<head>
```

```
<title>fee collection report page</title>
```

```
</head>
```

```
<body>
```

```
<h1>Hello this is fee collection report page</h1>
```

```
<h3>Please fill below form</h3>
```

```
</body>
```

```
</html>
```

### **Go to admissions app and click views.py file**

render ( ):

- This function uses the render ( ) function to create the HttpResponse that is sent back to the browser.
- This function is a shortcut; it creates an HTML file by combining a specified HTML template and some data to insert in the template (provided in the variable named "context").

## **views.py**

```
from django.shortcuts import render

def addadmission(request):

    return render(request, 'admissions/addadmission.html')

def admissionreport(request):

    return render(request, 'admissions/admissionreport.html')
```

## **Creating urls.py file in admission application**

Right click on admissions app and click new file and type urls.py file

Open urls.py in the admissions app

## **urls.py**

```
from django.urls import path
from admissions.views import addadmission
from admissions.views import admissionreport

urlpatterns = [
    path('newadm/', addadmission),
    path('admreport/', admissionreport),
]
```

## **Go to finance app and click views.py file**

## **views.py**

```
from django.shortcuts import render

def feecollection(request):

    return render(request, 'finance/feecollection.html')
```

```
def feeduesreport(request):  
    return render(request, 'finance/feeduesreport.html')  
  
def feecollectionreport(request):  
    return render(request, 'finance/feecollectionreport.html')
```

## **Creating urls.py file in finance application**

Right click on finance app and click new file and type urls.py file

Open urls.py in the finance app

### **urls.py**

```
from django.urls import path  
from finance.views import feecollection  
from finance.views import feeduesreport  
from finance.views import feecollectionreport  
  
urlpatterns = [  
    path('feecoll/', feecollection),  
    path('duesreport/', feeduesreport),  
    path('collectionreport/', feecollectionreport),  
  
]
```

## **Open urls.py in the project folder**

### **urls.py**

```
from django.contrib import admin  
from django.urls import path,include  
  
urlpatterns = [  

```

```
path('admin/', admin.site.urls),
path('ad/', include('admissions.urls')),
path('fin/', include('finance.urls')),

]
```

### **To know the current python filename**

```
print (__file__) # for example: template.py
```

### **To know the absolute path of current python file name**

```
print(os.path.abspath(__file__))
```

for example:d:\djangoclass\python template.py

### **To know base directory name of the current file**

```
print(os.path.dirname(os.path.abspath(__file__)))
```

Open the schoolproject and click the settings.py file and go to the templates and write the below command.

### **settings.py file**

```
import os
```

```
TEMPLATES=[
```

```
{
```

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

```
}
```

```
]
```



## To run the server application

`python manage.py runserver`

- Go to browser window and type this url `http://127.0.0.1:8000`
- It will shows the view action and choose the view particular action

`http://127.0.0.1:8000/ad/newadm`

## Steps to develop Template based applications:

1. Create a Project
2. Creation of Applications
3. Add these applications to the project in settings.py file. So, that django aware of applications.
4. Create a template folder inside main project in that templates older create a separate folder named with appname (admissions, finance) to hold that particular application specific templates.
5. Add templates folder path to settings.py file. So, that django can aware of our templates.

```
TEMPLATES=[  
  
    {  
  
        'DIRS':['D:\djangoclass\SchoolProject']  
  
    }  
]
```

It is not recommended to hardcore system specific locations in settings.py file to overcome this problem we can generate templates directory path programmatically as follows.

```
import os
```

```
BASE_DIR=os.path.dirname(os.path.dirname(os.path.abspath(__file__)))
```

```
TEMPLATES=[
```

```
{
```

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

```
}
```

```
]
```