

## CLASS-XII

### COMPUTER SCIENCE-PYTHON (083)

#### UNIT-III DATA MANAGEMENT-2 (Django)

By: Vikash Kumar Yadav  
PGT-Computer Science  
K.V.No.-IV ONGC Vadodara

#### 3.1 Django:

- Django is a free and open source web application framework written in Python.
- It is based on Model-View-Controller (MVC) framework.
- It is used for developing the web application.

#### 3.2 Installation of Django:

- Install Python 3.5 or upper version
- Connect your computer with internet and open Command Prompt & type the following command to install Django:  
**pip install django**
- For verification of installation of Django. Open Command prompt and type the following command:  
**django-admin --version**  
It will display the installed version in computer.

#### 3.3 Prepare the machine to develop the web Application:

- After installation of Python and Django in computer, next step is install the IDE (Integrated Development Environment) to write the code for web application. Here, we shall use PyCharm as IDE.
- PyCharm : PyCharm is an IDE used in computer programming, specifically for the Python language.
- To download PyCharm, go to the following link:  
**<https://www.jetbrains.com/pycharm/download/>**
- Install the setup.
- Now your system is ready to develop a web application in PyCharm using Django.

### 3.4 Create a Project and web application:

- After Installation of PyCharm IDE, open command prompt window.
- Go to the directory where you want to create the project. Here, we shall create the project in D:\ drive.
- To create the project type the following command in command prompt.

**D:\>django-admin startproject SPORTSMEET**

Here SPORTSMEET is the name of Project. A folder with the name SPORTSMEET will be created in D:\ directory.

- Now start PyCharm, Click on File →Open→SPORTSMEET folder in D:\ drive →OK
- SPORTSMEET project will be opened in PyCharm. It will have SPORTSMEET subfolder and manage.py file.
- SPORTSMEET subfolder has the following files:
  - \_\_init\_\_.py
  - settings.py
  - urls.py
  - wsgi.py
- Now open the Terminal window in PyCharm. Go to the directory D:\SPORTSMEET>
- Create the application under SPORTSMEET Project. Here the application name is NATIONALS. To create this application, type the following command in terminal window of PyCharm.

**D:\SPORTSMEET>python manage.py startapp NATIONALS**

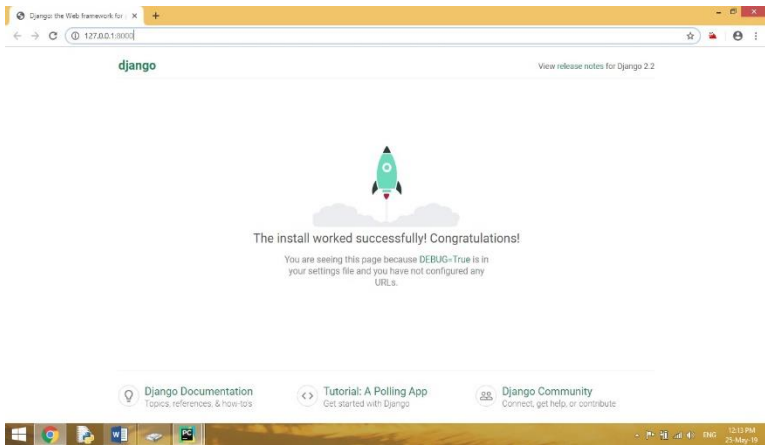
- Now to start the Django server, Type following command in Terminal window of PyCharm.

**D:\SPORTSMEET>python manage.py runserver**

- After successfully execution of above command, it will provide an address, which is given below:

<http://127.0.0.1:8000/>

Copy the above link and paste it in web browser. The following screen will appear in your web browser. It means Django server is working properly.



So, we have created a project SPORTSMEET and a web application NATIONALS under this project.

- In this web application we will perform the CRUD (Create, Read, Update, Delete) operations.
- We shall store the data of those students who will participate in National Sports Meet and perform the CRUD operations with the data stored in database.
- To Store the data, we will use **sqlite3** database system which is in-built in PyCharm. So no need to install it separately. You can use MySQL database system, which we will discuss later.

### 3.5 Now start the coding:

- Open PyCharm.
- Open settings.py file under SPORTSMEET subfolder to add our application in installed apps for migration.

Example:

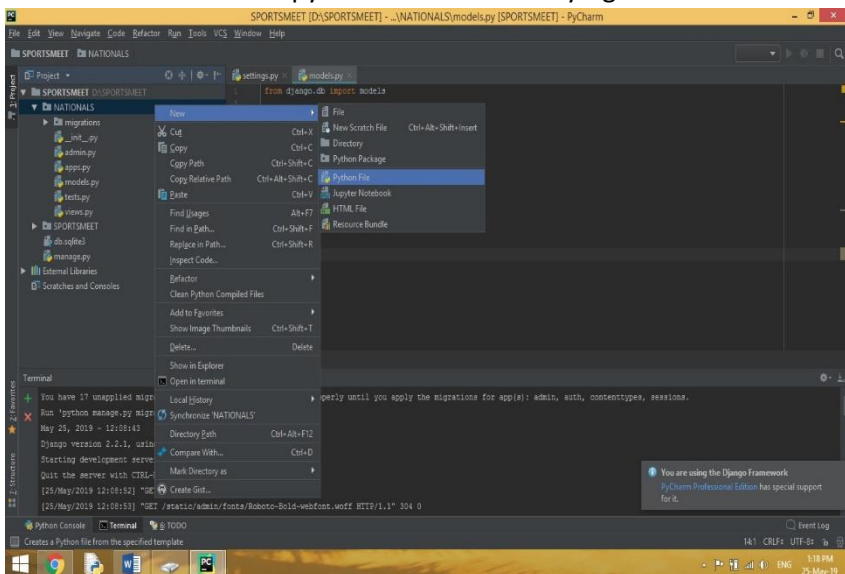
```
INSTALLED_APPS = [  
    'django.contrib.admin',  
    'django.contrib.auth',  
    'django.contrib.contenttypes',  
    'django.contrib.sessions',  
    'django.contrib.messages',  
    'django.contrib.staticfiles',  
    'NATIONALS'  
]
```

- Open the file **models.py** under NATIONALS web application to create a model and its fields. Models are considered as tables in MVC framework. An application may have multiple models.

```
from django.db import models

class PLAYER(models.Model):
    pid = models.CharField(max_length=10)
    pname = models.CharField(max_length=50)
    dob = models.CharField(max_length=20)
    gender = models.CharField(max_length=10)
    game = models.CharField(max_length=30)
    region = models.CharField(max_length=35)
    class Meta:
        db_table = 'player'
```

- Here, class Meta is as an instance of class PLAYER, that executes with Meta class.
- Now create a file forms.py under NATIONALS by right click on NATIONALS.



Django provides the facility of forms. It means we need not create any HTML form for table fields. Django denotes the table fields as a form. For that we have to create a **forms.py** and write the following code in forms.py file.

```

from django import forms
from NATIONALS.models import PLAYER
class PLAYERFORM (forms.ModelForm):
    class Meta:
        model = PLAYER
        fields = "__all__"

```

- Now migrate the models in database. For this, we have to run the following commands in Terminal window of PyCharm:

**D:\SPORTSMET>python manage.py makemigrations**

This command will create the model and make the changes if any. If the command successfully runs, the following information will be displayed in Terminal window.

Migrations for 'NATIONALS':

NATIONALS\migrations\0001\_initial.py

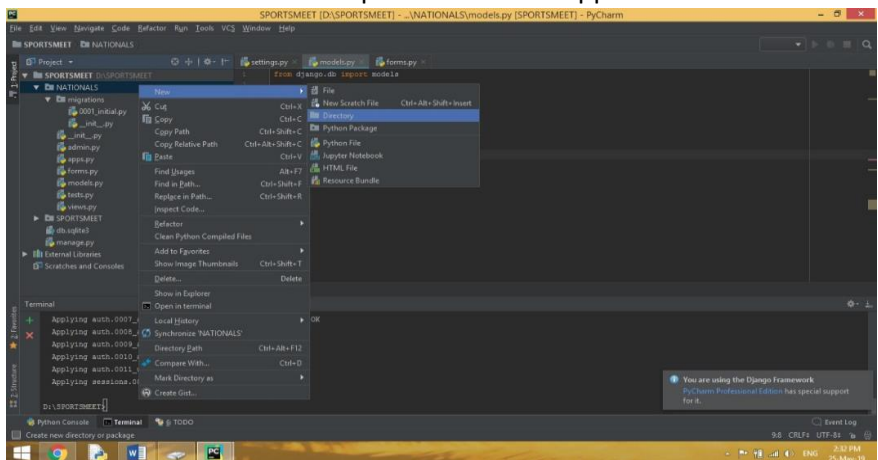
- Create model PLAYER

- Now run another command to migrate the created model. In Terminal window, write the following command:

**D:\SPORTSMET>python manage.py migrate**

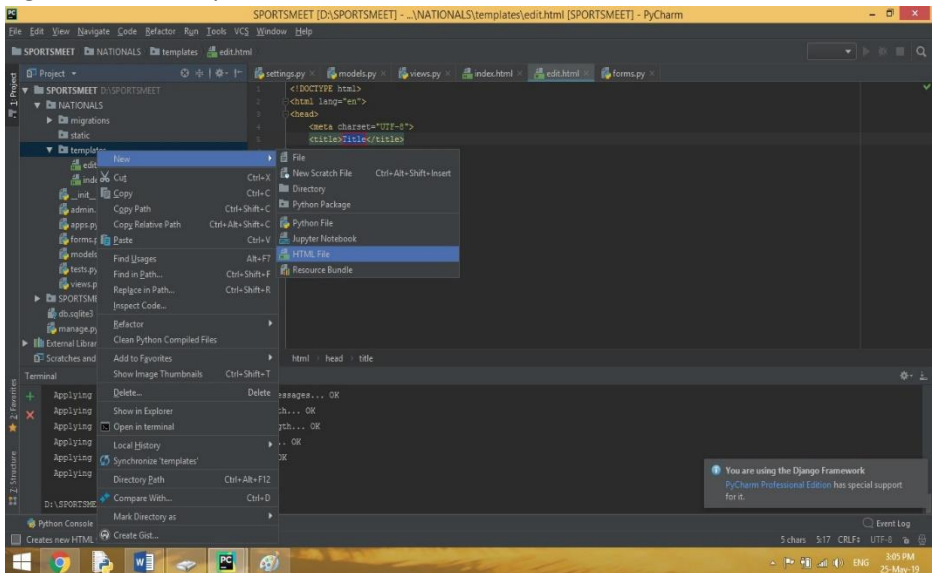
Now the created model is stored in the file **0001\_initial.py** under the folder migration in NATIONALS application.

- A table player has been created in database. It has following fields: (id, pid, pname, DOB, gender, game, region). id field is automatically created in Django, which has its own system generated values.
- Create a folder named as templates under the application NATIONALS.



templates folder will have the .html files. In our application, we will have index.html, edit.html and show.html files under templates folder.

- Create the html files with the name index.html, edit.html and show.html, by right click on templates folder under NATIONALS.



- Create another folder with the name static, under NATIONALS. It will have **style.css** file.

Right click on static folder → New → File → write the name style.css → OK  
Write following code in style.css file:

```
table, th, td {
    border: 1px solid black;
}

div {
    font-weight: bold;
}
```

- Now, open **views.py** under the application NATIONALS.
- **views.py** file will work as a controller. It is not the view in MVC. In this file we would call the view according to their actions. In this file, we will also work with models. So, it is an important file.
- Write the following code in **views.py** file:

```
from django.shortcuts import render, redirect
from NATIONALS.forms import PLAYERFORM
from NATIONALS.models import PLAYER
```

```

# Create your views here.
def ply(request):
    if request.method == "POST":
        form = PLAYERFORM(request.POST)
        if form.is_valid():
            try:
                form.save()
                return redirect('/show')
            except:
                pass
        else:
            form = PLAYERFORM()
    return render(request,"index.html",{ 'form':form})

def show(request):
    players = PLAYER.objects.all()
    return render(request,"show.html",{ 'players':players})

def edit(request, id):
    player = PLAYER.objects.get(id=id)
    return render(request, "edit.html", { 'player': player})

def update(request, id):
    player = PLAYER.objects.get(id=id)
    form = PLAYERFORM(request.POST, instance= player)
    if form.is_valid():
        form.save()
        return redirect('/show')
    return render(request, "edit.html", { 'player': player})

def delete (request, id):
    player = PLAYER.objects.get(id=id)
    player.delete()
    return redirect('/show')

```

- Open **index.html** file under templates folder and write the following code:

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Index Page</title>
    {% load staticfiles %}
    <link rel="stylesheet" href="{% static '/style.css' %}" />
</head>
<body>
<center><strong><u><h1>PLAYER DATA</h1></u></strong></center>
<center>

<form method = "POST" class = "post-form" action="/ply">
    {% csrf_token %}
    <div class="container">

```





```

        <button type="submit" class="btn btn-
primary">Submit</button>
    </div>
</div>
</div>

</form></center>
</body>
</html>

```

- Open **show.html** file under templates folder and write the following code:

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Show Player Details</title>
    {% load staticfiles %}
    <link rel = "stylesheet " href="{% static '/style.css' %}"
/>
</head>
<body>
<center><strong><u><h1>PLAYER
DETAILS</h1></u></strong></center>
<center>
<table class="table table-striped table-bordered table=80%">
    <thead class= "thead-dark">
        <tr>
            <th>Player ID</th>
            <th>Player Name</th>
            <th>Date of Birth</th>
            <th>Gender</th>
            <th>Game</th>
            <th>Region</th>
            <th colspan="2">Action</th>
        </tr>
    </thead>
    <tbody>
        {% for player in players %}
        <tr>
            <td>{{ player.pid }}</td>
            <td>{{ player.pname }}</td>
            <td>{{ player.dob }}</td>
            <td>{{ player.gender }}</td>
            <td>{{ player.game }}</td>
            <td>{{ player.region }}</td>
            <td>
                <a href = "/edit/{{player.id}}"><span
class="glyphicon glyphicon-pencil">Edit</span> </a>
            </td>
            <td>
                <a href="/delete/{{player.id}}">Delete</a>
            </td>

```





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

urlpatterns = [
    path('admin/', admin.site.urls),
    path('ply', views.ply),
    path('show', views.show),
    path('edit/<int:id>', views.edit),
    path('update/<int:id>', views.update),
    path('delete/<int:id>', views.delete),
```

- After that save the project. Now open Terminal window in PyCharm, Write the following command:

**D:\SPORTSMEET>Python manage.py runserver**

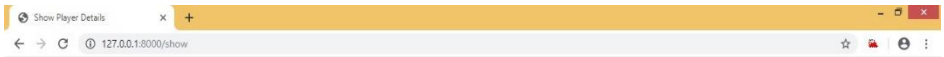
After execution of above command, the following address will appear in Terminal window.

<http://127.0.0.1:8000/>

Copy it and paste it into web browser. Press Enter. It will display the error page. Now write the address of the show page. Means, write the address given below:

<http://127.0.0.1:8000/show>

## OUTPUTS:



## PLAYER DETAILS

Player ID	Player Name	Date of Birth	Gender	Game	Region	Action
-----------	-------------	---------------	--------	------	--------	--------

[Add New Record](#)



### OUTPUT-1 Empty Table, No Records



## PLAYER DATA

Enter Details

Player ID:

Player Name:

Date of Birth:

Gender:

Game:

Region:



### OUTPUT-2 Adding the New Record

## PLAYER DETAILS

Player ID	Player Name	Date of Birth	Gender	Game	Region	Action
142	Anil	10/10/2005	M	Badminton	Ahmedabad	<a href="#">Edit</a> <a href="#">Delete</a>
245	Harsh	10/10/2005	M	TT	Mumbai	<a href="#">Edit</a> <a href="#">Delete</a>
423	Diya	13/02/2002	F	Kho-Kho	Patna	<a href="#">Edit</a> <a href="#">Delete</a>
782	Vidhmahey	12/01/2002	M	Chess	Ahmedabad	<a href="#">Edit</a> <a href="#">Delete</a>
102	Rupali	02/05/2003	F	Cricket	Chandigarh	<a href="#">Edit</a> <a href="#">Delete</a>

[Add New Record](#)

### OUTPUT-3 Records in the table after addition

#### Update Details

Player ID :

Player Name :

DOB :

Gender :

Game :

Region :

### OUTPUT-4 Update the Record