Apr 04

- **views.py**

```
from testapp.models import HydJobs
def hydjobs_view(request):
   jobs_list = HydJobs.objects.all()
   return render(request,'testapp/hydjobs.html',{'jobs_list':jobs_list})
```

- **hydjobs.html**

```
<body>
   <h1>Hyderabad Jobs Information</h1>
   {% if jobs_list %}
   <table border="3">
     <thead>
       <th>Date</th>
       <th>Company</th>
       <th>Title</th>
       <th>Eligibility</th>
       <th>Address</th>
       <th>Email</th>
       <th>Phone Number</th>
     </thead>
     {% for job in jobs_list%}
     <tr>
       <td>{{job.date}}</td>
       <td>{{job.company}}</td>
       <td>{{job.title}}</td>
       <td>{{job.eligibility}}</td>
       <td>{{job.address}}</td>
       <td>{{job.email}}</td>
       <td>{{job.phonenumber}}</td>
     </tr>
     {% endfor %}
   </table>
   {% else %}
   <p>No Jobs In Hyderabad......</p>
   {% endif %}
  </body>
```

- **populate.py**

```
import os
```

```
os.environ.setdefault('DJANGO_SETTINGS_MODULE', 'sunnyjobs.settings')
import django
django.setup()

from testapp.models import HydJobs
from faker import Faker
from random import *
fake = Faker()
def phonenumbergen():
        d1 = randint(6,9)
        num = '' + str(d1)
        for i in range(9):
                num += str(randint(0,9))
        return int(num)
def populate(n):
  for i in range(n):
    fdate = fake.date()
    fcompany = fake.company()
    ftitle = fake.random_element(elements=('Project Manager','Team
Lead','Software Engineer','Associate Engineer'))
    feligibility =
fake.random_element(elements=('B.Tech','M.Tech','MCA','Phd','Msc','Mahesh
Sir Student'))
    faddress = fake.address()
    femail = fake.email()
    fphonenumber = phonenumbergen()
    hyd_jobs_record = HydJobs.objects.get_or_create(
    date = fdate,
    company = fcompany,
    title = ftitle,
    eligibility = feligibility,
    address = faddress,
    email = femail,
    phonenumber = fphonenumber
    )
n = int(input('Enter number of records:'))
populate(n)
print(f'{n} Records inserted successfully.....')
```

Chapter-4

Django Forms:

-->The main purpose of the forms is to take user input.

Ex:

login form, registration form, enquiry form.......

-->From the forms we can read end user provided input data and we can use that data based on requirement. We may store in the database for future purpose. We may use just for validation/authentication purpose.

-->Here we have to use Django specific forms not HTML forms.

Advantages of Django Forms over HTML forms:

1).We can develop forms very easily with python code.

2).We can generate HTML Form widget/componets(like text area,email, pwd etc)

very quickly.

3).Validating data will become very easy.

4).Processing data into python data structures like list, set etc will become easy.

5).Creation of Models based forms will be come very wasy.

model class--->Converted into database table.

form class-->Converted into html form

Process to generate Django Forms:

Step-1: Creation of forms.py file in our application folder with our required fields.

- **forms.py**

```
from django import forms
class StudentForm(forms.Form):
   name = forms.CharField()
   marks = forms.IntegerField()
```

Note:

name and marks are the field names which will be available in HTML form forms.py=====>views.py=====>Template file(HTML)

Step-2: Usage of forms.py inside views.py file
views.py file is responsible to send this form to the template file.

- **views.py**

```
from testapp.forms import StudentForm
def studentinput_view(request):
  form = StudentForm()
  return render(request,'testapp/input.html',{'form':form})
```

Step-3: Creation of html file to hold form
Inside template file we have to use template tag to inject form {{form}}

- **input.html**

```
<!DOCTYPE html>
{% load static %}
<html lang="en" dir="ltr">
 <head>
  <meta charset="utf-8">
  <title>Student Form</title>
  <link rel="stylesheet" href="{% static 'css/form1.css' %}">
 </head>
 <body>
  <h1>Student Input Form</h1>
  <div class="container" align='center'>
   <form method="post">
    {{form.as_p}}
    <input type="submit" name="" value="Submit">
   </form>
  </div>
 </body>
</html>
```

- **form1.css**

```
h1{
 text-align: center;
}
body{
 background: yellow;
 color: red;
}
```

- **urls.py**

```
from testapp import views
urlpatterns = [
    path('admin/', admin.site.urls),
    path('std/', views.studentinput_view),
]
```