App:

D:\Django_20MAR_7PM>django-admin startproject modelproject2

D:\Django_20MAR_7PM>cd modelproject2

D:\Django_20MAR_7PM\modelproject2>py manage.py startapp testapp

-->Add app in settings.py

- **models.py**

class Employee(models.Model):

   eno = models.IntegerField()

   ename = models.CharField(max_length=30)

   esal = models.FloatField()

   eaddr = models.CharField(max_length=30)

-->Makemigrations and migrate

-->Register model class and ModelAdmin class in admin.py

- **admin.py**

from testapp.models import Employee

class EmployeeAdmin(admin.ModelAdmin):

   list_display = ['eno','ename','esal','eaddr']

admin.site.register(Employee,EmployeeAdmin)

-->**create super user:**

     py manage.py createsuperuser

-->Login to admin interface and add some data manually.

Create a function in views.py

- **views.py**

```python
from testapp.models import Employee

def empdata_view(request):
    emp_list = Employee.objects.all()
    my_dict = {'emp_list':emp_list}
    return render(request,'testapp/emp.html',my_dict)
```

- **emp.html**

```html
<link rel="stylesheet" href="{% static 'css/emp1.css' %}">
<body>
  <h1>Employee List</h1>
  {% if emp_list %}
  <table border="3">
   <thead>
     <th>ENO</th>
     <th>ENAME</th>
     <th>ESAL</th>
     <th>EADDR</th>
   </thead>
   {% for emp in emp_list %}
   <tr>
     <td>{{emp.eno}}</td>
     <td>{{emp.ename}}</td>
     <td>{{emp.esal}}</td>
     <td>{{emp.eaddr}}</td>
   </tr>
```

```
    {% endfor %}

   </table>

   {% else %}

   <p>No Records Found!!!!</p>

   {% endif %}

  </body>
```

- **urls.py**
- path('emp/', views.empdata_view),

- **emp1.css**

```css
body{

  background: yellow;

  color: red;

}

h1{

  text-align: center;

}

table{

  margin: auto;

}
```

1).End user sending request:

http://127.0.0.1:8000/emp/

2).View function will get request.

3).view asking model to connect with database and provide required data.

Employee.objects.all()

4).Model will communicate with database and provide required data to the view function.

emp_list

5).View will send that data to template file

return render(request,'testapp/emp.html',{'emp_list':emp_list})

Questions related to models and templates?

1).How to configure database inside settings.py?

2).How to check the connections?

3).How to define model class inside models.py?

4).How to perform makemigrations?

5).How to perform migrate?

6).What is the diff between makemigrations and migrate?

7).What is the advantage of creating table by using migrate command instead of creating manually in the database?

8).How to add modle to admin interface inside admin.py?

9).To display total data, how to write model admin class inside admin.py?

10).How to create super user?

11).How to see generated sql code as result of makemigrations?

How to generate fake data for application:

```
>>>pip install faker

from faker import Faker

fake = Faker()

name = fake.name()

print(name)

fname = fake.first_name()

print(fname)

lname = fake.last_name()

print(lname)

date1 = fake.date()

print(date1)

number = fake.random_number(5)

print(number)

email1 = fake.email()

print(email1)

city = fake.city()

print(city)

print(fake.random_int(min=0,max=9999))

print(fake.random_element(elements=('sunny','katrina','kareena','deepika')))
```

Phone Number Generation:

```
from random import *

def phonenumbergen():

        d1 = randint(6,9)

        num = '' + str(d1)
```

```python
    for i in range(9):
        num += str(randint(0,9))
    return int(num)
for i in range(10):
    print(phonenumbergen())
```