Chapter-6

Working with Django Model Forms

Model Forms (Forms based on Model):

☕ Sometimes we can create form based on Model, such type of forms are called model based forms or model forms.

☕ The main advantage of model forms is we can grab end user input and we can save that input data very easily to the database.

☕ Django provides inbuilt support to develop model based forms very easily.

How to develop Model based Forms:

1) **While develop FormClass instead of inheriting forms.Form class,we have to inherit forms.ModelForm class.**

   class RegisterForm(forms.ModelForm):
   ....

2) **We have to write one nested class (Meta class) to specify Model information and required fields.**

class RegisterForm(forms.ModelForm):
  # field declarations if we are performing any custom validations.If we are not defining any custom validations then here we are not required to specify any field.

  class Meta:
    # we have to specify Model class name and requied fields
    model=Student
    fields='__all__'

Case-1: Instead of all fields if we want only selected fields, then we have to specify as follows

class Meta:
  model=Student
  fileds=('field1','field2','field3')

In the form only 3 fields will be considered.
If Model class contains huge number of fields and we required to consider very less number of fields in the form then we should use this approach.

Case-2:
Instead of all fields if we want to exclude certain fields,then we have to specify as follows

class Meta:
  model=Student
  exclude=['field1','field2']
 In the form all fields will be considered except field1 and field2.

If the Model class contains huge number of fields and if we want to exclude very few fields then we have to use this approach.

Q) In Model based Forms, how many ways are there to specify Fields Information

Ans: 3 ways
1) All fields
2) Include certain fields
3) Exclude certain fields

Note: The most commonly used approach is to include all fields.

How to Save User's Input Data to Database in Model based Forms:

We have to use save() method.

```
def student_view(request):
    ...
    if request.method=='POST':
        form=RegisterForm(request.POST)
        if form.is_valid():
            form.save(commit=True)
        ..
```

Demo project-1 (modelformproject):

models.py:

```
1) from django.db import models
2)
3) # Create your models here.
4) class Student(models.Model):
5)     name=models.CharField(max_length=30)
6)     marks=models.IntegerField()
```

forms.py:

```
1) from django import forms
2) from testapp.models import Student
3) class StudentForm(forms.ModelForm):
4)     #fields with validations
5)     class Meta:
6)         model=Student
7)         fields='__all__'
```

views.py:

```
1) from django.shortcuts import render
2) from . import forms
3)
```

```
4)    # Create your views here.
5)    def student_view(request):
6)      form=forms.StudentForm
7)      if request.method=='POST':
8)        form=forms.StudentForm(request.POST)
9)        if form.is_valid():
10)         form.save(commit=True)
11)   return render(request,'testapp/studentform.html',{'form':form})
```

Demo Project-2: movieproject

models.py

```
1)   from django.db import models
2)
3)   # Create your models here.
4)   class Movie(models.Model):
5)     rdate=models.DateField()
6)     moviename=models.CharField(max_length=30)
7)     hero=models.CharField(max_length=30)
8)     heroine=models.CharField(max_length=30)
9)     rating=models.IntegerField()
```

admin.py

```
1)   from django.contrib import admin
2)   from testapp.models import Movie
3)   # Register your models here.
4)   class MovieAdmin(admin.ModelAdmin):
5)     list_display=['rdate','moviename','hero','heroine','rating']
6)   admin.site.register(Movie,MovieAdmin)
```

forms.py

```
1)   from django import forms
2)   from testapp.models import Movie
3)   class MovieForm(forms.ModelForm):
4)     class Meta:
5)       model=Movie
6)       fields='__all__'
```

views.py

```python
1) from django.shortcuts import render
2) from testapp.forms import MovieForm
3) from testapp.models import Movie
4)
5) # Create your views here.
6) def index_view(request):
7)    return render(request,'testapp/index.html')
8)
9) def add_movie_view(request):
10)   form=MovieForm()
11)   if request.method=='POST':
12)     form=MovieForm(request.POST)
13)     if form.is_valid():
14)       form.save()
15)     return index_view(request)
16)   return render(request,'testapp/addmovie.html',{'form':form})
17)
18) def list_movie_view(request):
19)   movies_list=Movie.objects.all().order_by('-rating')
20)   return render(request,'testapp/listmovie.html',{'movies_list':movies_list})
```

index.html

```html
1) <!DOCTYPE html>
2) {%load staticfiles%}
3) <html lang="en" dir="ltr">
4)   <head>
5)     <meta charset="utf-8">
6)     <title></title>
7)     <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css" integrity="sha384-BVYiiSIFeK1dGmJRAkycuHAHRg32OmUcww7on3RYdg4Va+PmSTsz/K68vbdEjh4u" crossorigin="anonymous">
8)     <link rel="stylesheet" href="{% static "css/movie.css"%}">
9)     <link href="https://fonts.googleapis.com/css?family=Merriweather" rel="stylesheet">
10)   </head>
11)   <body>
12)   <div class="container" align="center">
13)     <div class="jumbotron">
14)       <h1>DURGA Movies...</h1>
15)       <p>up to date movies information</p>
```

```html
16)        <a class="btn btn-primary btn-lg"
    href="/addmovie" role="button">Add Movie</a>
17)        <a class="btn btn-primary btn-lg"
    href="/listmovies" role="button">List Movies</a>
18)    </div>
19)    </div>
20)  </body>
21) </html>
```

listmovie.html

```html
1)  <!DOCTYPE html>
2)  {% load cust_filters%}
3)  {%load staticfiles%}
4)  <html lang="en" dir="ltr">
5)    <head>
6)      <meta charset="utf-8">
7)      <title></title>
8)      <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7
    /css/bootstrap.min.css" integrity="sha384-
    BVYiiSIFeK1dGmJRAkycuHAHRg32OmUcww7on3RYdg4Va+PmSTsz/K68vbdEjh4u" c
    rossorigin="anonymous">
9)      <link rel="stylesheet" href="{% static "css/movie.css"%}">
10)     <link href="https://fonts.googleapis.com/css?family=Merriweather"
    rel="stylesheet">
11)   </head>
12)   <body>
13)     <h1>Movies Information</h1>
14)     {% if movies_list %}
15)      <table>
16)       <thead>
17)         <th>Released Date</th>
18)         <th>Movie Name</th>
19)         <th>Hero</th>
20)         <th>Heroine</th>
21)         <th>Rating</th>
22)       </thead>
23)       {% for movie in movies_list%}
24)        <tr>
25)          <td>{{movie.rdate}}</td>
26)          <td>{{movie.moviename}}</td>
27)          <td>{{movie.hero}}</td>
28)          <td>{{movie.heroine}}</td>
29)          <td>{{movie.rating}}</td>
30)        </tr>
```

```
31)      {% endfor %}
32)    </table>
33)  {% else %}
34)   <p>No Movies Found</p>
35)  {%endif%}
36) </body>
37) </html>
```

addmovie.html

```
1)  <!DOCTYPE html>
2)  {%load staticfiles%}
3)  <html lang="en" dir="ltr">
4)   <head>
5)    <meta charset="utf-8">
6)    <title></title>
7)    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7
        /css/bootstrap.min.css" integrity="sha384-
        BVYiiSIFeK1dGmJRAkycuHAHRg32OmUcww7on3RYdg4Va+PmSTsz/K68vbdEjh4u"
        crossorigin="anonymous">
8)    <link rel="stylesheet" href="{% static "css/movie.css"%}">
9)    <link href="https://fonts.googleapis.com/css?family=Merriweather"
        rel="stylesheet">
10)  </head>
11)  <body>
12)  <div class="container" align='center'>
13)   <h1>Add Movie Information</h1><hr>
14)   <form  method="post">
15)    {{form.as_p}}
16)    {%csrf_token%}
17)    <input type="submit" class='btn btn-primary' value="Add Movie">
18)   </form>
19)  </div>
20)  </body>
21) </html>
```

movie.css

```
1)  body{
2)    background: url(https://images.unsplash.com/photo-1513366208864-
      87536b8bd7b4?ixlib=rb-
      0.3.5&ixid=eyJhcHBfaWQiOjEyMDd9&s=b04402d9c23edcc4b321db9fd9be60a7&au
      to=format&fit=crop&w=334&q=80);
3)    /* background-repeat: no-repeat;
```

```css
4)    background-size: cover; */
5)  }
6)  .container .jumbotron{
7)    margin-top:200px;
8)    background: orange;
9)    color:white;
10)   font-family: 'Merriweather', serif;
11) }
12) table,tr,td,thead,th{
13)   border:2px solid white;
14)   font-size: 30px;
15)   color:white;
16)   background:red;
17)   margin:auto;
18) }
19) thead,th{
20)   border:2px solid red;
21)   font-size: 30px;
22)   color:blue;
23)   margin:auto;
24)   background:yellow;
25) }
26) h1{
27)   text-align: center;
28) }
```

urls.py

```python
1)  from django.conf.urls import url
2)  from django.contrib import admin
3)  from testapp import views
4)
5)  urlpatterns = [
6)    url(r'^admin/', admin.site.urls),
7)    url(r'^$', views.index_view),
8)    url(r'^addmovie/', views.add_movie_view),
9)    url(r'^listmovies/', views.list_movie_view),
10) ]
```

How to Add Date Widget:

write a DateInput class in forms.py as:

```python
class DateInput(forms.DateInput):
  input_type = 'date'
```

--> Now define ModelForm class as:

```
class MoviesForm(forms.ModelForm):
    movie_name=forms.CharField(widget=forms.TextInput(attrs={'class':'form-control'}))
    release_date=forms.DateField(widget=DateInput(attrs={'class':'form-control'}))
    hero_name=forms.CharField(widget=forms.TextInput(attrs={'class':'form-control'}))
    heroine_name=forms.CharField(widget=forms.TextInput(attrs={'class':'form-control'}))
    director_name=forms.CharField(widget=forms.TextInput(attrs={'class':'form-control'}))
    movie_ratings=forms.FloatField(widget=forms.TextInput(attrs={'class':'form-control'}))

    class Meta:
        model=MoviesModel

    fields=['movie_name','release_date','hero_name','heroine_name','director_name','movie
_ratings']
```

--> Here 'class':'form-control' is a bootstrap class used to format the date field

--> link for DateInput class https://docs.djangoproject.com/en/2.1/ref/forms/widgets/#dateinp