

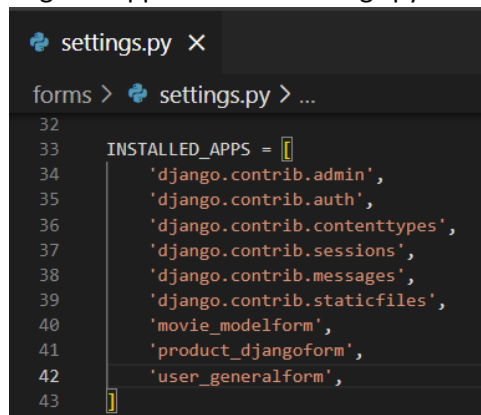
1. Create project

```
\DJANGO T417 203>django-admin startproject forms  
  
\DJANGO T417 203>cd forms  
  
\DJANGO T417 203\forms>code .
```

2. Create 3 applications

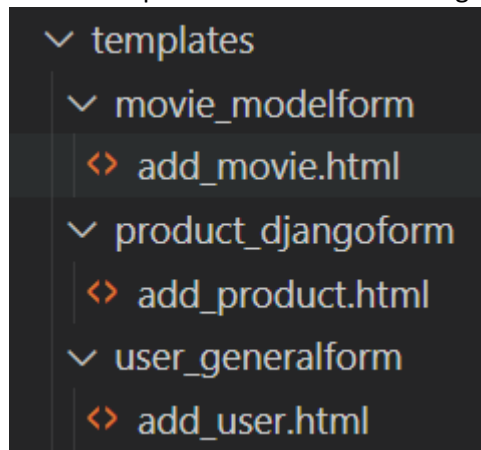
```
\forms>python manage.py startapp product_djangoform  
  
\forms>python manage.py startapp movie_modelform  
  
\forms>python manage.py startapp user_generalform
```

3. Register applications in settings.py



```
settings.py X  
forms > settings.py > ...  
32  
33 INSTALLED_APPS = [  
34     'django.contrib.admin',  
35     'django.contrib.auth',  
36     'django.contrib.contenttypes',  
37     'django.contrib.sessions',  
38     'django.contrib.messages',  
39     'django.contrib.staticfiles',  
40     'movie_modelform',  
41     'product_djangoform',  
42     'user_generalform',  
43 ]
```

4. Create templates folder with following structure

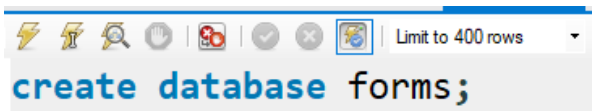


```
▼ templates  
  ▼ movie_modelform  
    <> add_movie.html  
  ▼ product_djangoform  
    <> add_product.html  
  ▼ user_generalform  
    <> add_user.html
```

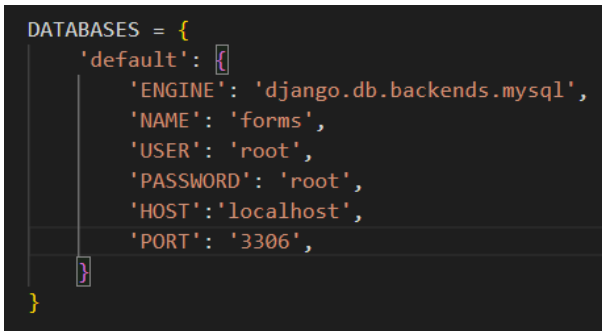
5. Register templates folder in settings.py

```
TEMPLATES = [  
    {  
        'BACKEND': 'django.template.backends.django.DjangoTemplates',  
        'DIRS': [BASE_DIR/'templates'],  
        'APP_DIRS': True,
```

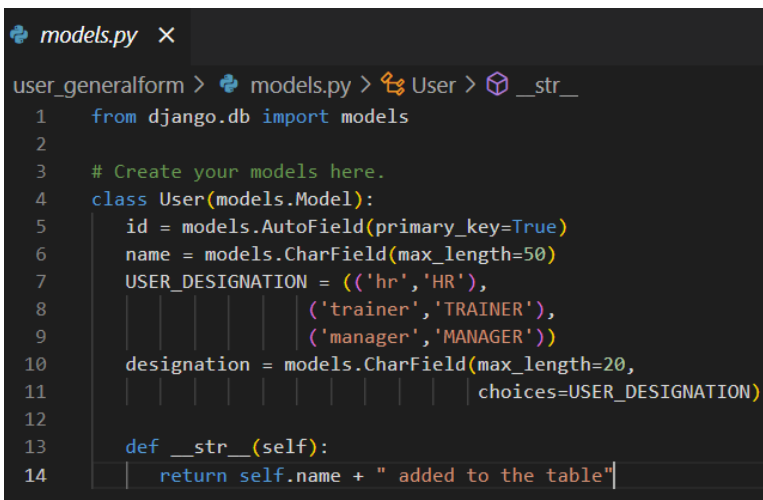
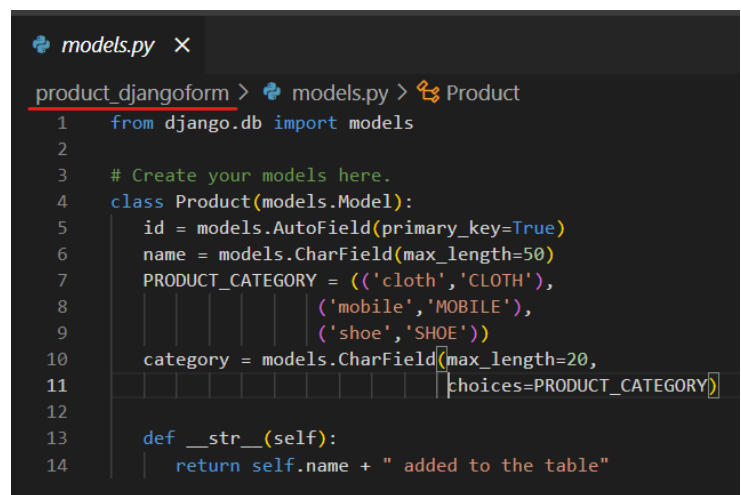
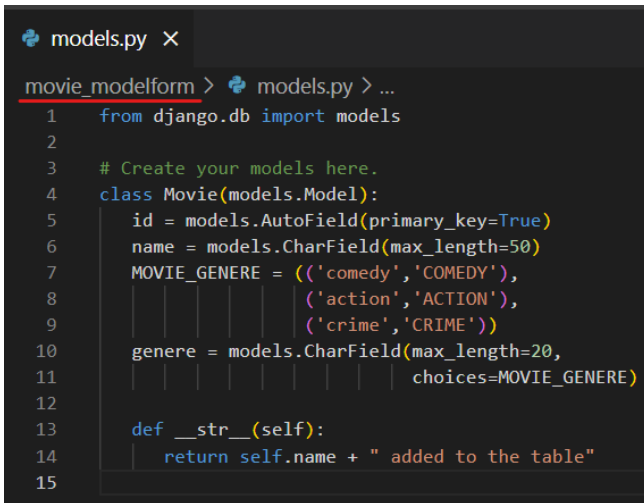
6. Create database forms



7. Register database in settings.py



8. Create model in each application



9. Register model of each application in admin.py

```
admin.py > admin.py > MovieAdmin
1 from django.contrib import admin
2 from movie_modelform.models import Movie
3
4 # Register your models here.
5 class MovieAdmin(admin.ModelAdmin):
6     list_display = ['id','name','genre']
7     admin.site.register(Movie,MovieAdmin)
8

admin.py > admin.py > ProductAdmin
1 from django.contrib import admin
2 from product_djangoform.models import Product
3
4 # Register your models here.
5 class ProductAdmin(admin.ModelAdmin):
6     list_display = ['id','name','category']
7     admin.site.register(Product,ProductAdmin)
8
9

admin.py > admin.py > UserAdmin
1 from django.contrib import admin
2 from user_generalform.models import User
3
4 # Register your models here.
5 class UserAdmin(admin.ModelAdmin):
6     list_display = ['id','name','designation']
7     admin.site.register(User,UserAdmin)
8
```

10. Makemigrations and migrate

```
C:\TEJAS KASARE (Very imp folder)\my notes\Batch Wise\DJANGO T417 203\forms>python manage.py makemigrations
Migrations for 'movie_modelform':
  movie_modelform\migrations\0001_initial.py
    - Create model Movie
Migrations for 'product_djangoform':
  product_djangoform\migrations\0001_initial.py
    - Create model Product
Migrations for 'user_generalform':
  user_generalform\migrations\0001_initial.py
    - Create model User

C:\TEJAS KASARE (Very imp folder)\my notes\Batch Wise\DJANGO T417 203\forms>python manage.py migrate
Operations to perform:
```

11. Check tables

- 1 • create database forms;
- 2 • use forms;
- 3 • show tables;

Result Grid	Filter Rows:	Export: Wri
Tables_in_forms		
auth_permission		
auth_user		
auth_user_groups		
auth_user_user_permissions		
django_admin_log		
django_content_type		
django_migrations		
django_session		
movie_modelform_movie		
product_djangoform_product		
user_generalform_user		

12. Working with django form

Since we are creating django form in Product application, below step are for application - **product_djangoform**

- a. Create forms.py in product_djangoform application

```
forms.py  X
product_djangoform > forms.py > ProductForm
1  from django import forms
2
3  class ProductForm(forms.Form):
4      name = forms.CharField(max_length=50)
5      PRODUCT_CATEGORY = (('cloth','CLOTH'),
6                          ('mobile','MOBILE'),
7                          ('shoe','SHOE'))
8      category = forms.ChoiceField(choices=PRODUCT_CATEGORY)
```

- b. Create view to display above form

```
views.py  X
product_djangoform > views.py > ...
1  from django.shortcuts import render
2  from product_djangoform.forms import ProductForm
3
4  # Create your views here.
5  def add_product(request):
6      data = {}
7      form=ProductForm()
8      data['product_form']=form
9      return render(request,'product_djangoform/add_product.html',context=data)
10
```

- c. Display form in templates > product_djangoform > add_product.html

```
add_product.html  X
templates > product_djangoform > add_product.html
8  <body>
9      <h1>Add Your Product</h1>
10     <form action="post">
11         {{product_form.as_p}}
12         <br>
13         <input type="submit">
14     </form>
15 </body>
16 </html>
```

IMPORTANT: in above image, line 10, change action='post' to method='method'

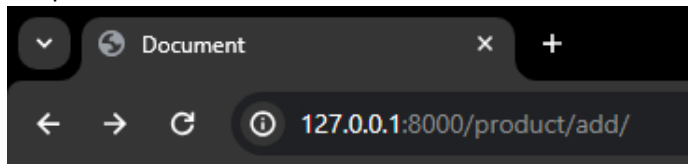
- d. Create application level url
 - i. Create product_urls.py in product_djangoform

```
product_urls.py X
product_djangoform > product_urls.py > ...
1  from django.urls import path
2  from product_djangoform import views
3  urlpatterns = [
4      path('add/', views.add_product),
5  ]
```

- e. Register above application level url in urls.py

```
urls.py X
forms > urls.py > ...
17 from django.contrib import admin
18 from django.urls import path,include
19
20 urlpatterns = [
21     path('admin/', admin.site.urls),
22     path('product/',include('product_djangoform.product_urls'))
23 ]
24
```

- f. Output :



Add Your Product

Name:

Category:

13. Inserting django forms data into table :

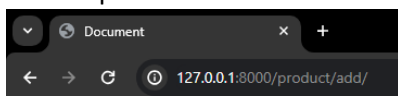
a. Add csrf_token into form

```
add_product.html X
templates > product_djangoform > add_product.html
8 <body>
9 <h1>Add Your Product</h1>
10 <form method="post">
11     {% csrf_token %}
12     {{product_form.as_p}}
13     <br>
14     <input type="submit">
15 </form>
16 </body>
17 </html>
```

b. Collect data from form and insert into table (model) [views.py of product_djangoform]

```
views.py X
product_djangoform > views.py > ...
2 from product_djangoform.forms import ProductForm
3 from django.http import HttpResponse
4 from product_djangoform.models import Product
5
6 # Create your views here.
7 def view_product(request):
8     return HttpResponse("Product added to the table")
9
10 def add_product(request):
11     if request.method == 'POST':
12         product_form=ProductForm(request.POST)
13         #code to check valid data
14         if(product_form.is_valid()):
15             product_model=Product()
16             product_model.name = product_form.cleaned_data['name']
17             product_model.category = product_form.cleaned_data['category']
18             product_model.save()
19             return view_product(request)
20     data = {}
21     form=ProductForm()
22     data['product_form']=form
23     return render(request,'product_djangoform/add_product.html',context=data)
24
```

c. Output



Add Your Product

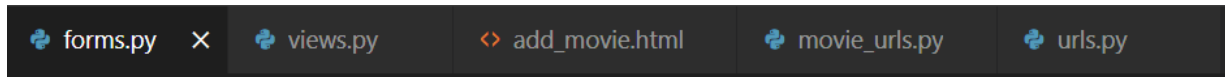
Name:

Category:

```
4 • select * from product_djangoform_product;
```

Result Grid		
Filter Rows:		
	id	name
1	1	iphone
2	2	redmi note 5
3	NULL	NULL

14. Working with model form (here we will use our movie_modelform application)



- a. Create forms.py in movie_modelform. Code –

```
forms.py X
movie_modelform > forms.py > ...
1  from django import forms
2  from movie_modelform.models import Movie
3
4  class MovieForm(forms.ModelForm):
5      class Meta:
6          model=Movie
7          # exclude= ['name']
8          # fields = ('genre',)
9          fields = '__all__'
```

- b. Create view to show above model form (views.py in movie_modelform)

```
views.py X
movie_modelform > views.py > ...
1  from django.shortcuts import render
2  from movie_modelform.forms import MovieForm
3
4  # Create your views here.
5  def add_movie(request):
6      data={}
7      form=MovieForm()
8      data['movie_form']=form
9      return render(request,'movie_modelform/add_movie.html',context=data)
10
```

- c. Display form in templates > movie_modelform > add_movie.html

```
<body>
  <h1>Add your movie </h1>
  <form method="POST">
    {% csrf_token %}
    {{ movie_form.as_p }}
    <input type="submit">
  </form>
</body>
```

- d. Create movie_urls.py in movie_modelform

```
movie_urls.py X
movie_modelform > movie_urls.py > ...
1  from django.urls import path,include
2  from movie_modelform import views
3
4  urlpatterns = [
5      path('add/', views.add_movie)
6  ]
```

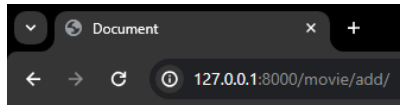
- e. Register above movie_urls.py in urls.py

```

urls.py  X
forms > urls.py > ...
17 from django.contrib import admin
18 from django.urls import path,include
19
20 urlpatterns = [
21     path('admin/', admin.site.urls),
22     path('product/',include('product_djangoform.product_urls')),
23     path('movie/',include('movie_modelform.movie_urls'))
24 ]

```

f. Visit url and check form



Add your movie

Name:

Genre:

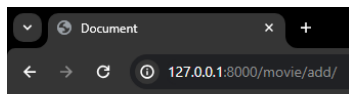
g. Logic to insert data into table

```

views.py  X
movie_modelform > views.py > ...
1  from django.shortcuts import render
2  from movie_modelform.forms import MovieForm
3  from django.http import HttpResponse
4
5  # Create your views here.
6
7  def show_movie(request):
8      return HttpResponse("Movie added to the table")
9
10 def add_movie(request):
11     if request.method=='POST':
12         movie_form = MovieForm(request.POST)
13         if(movie_form.is_valid()):
14             movie_form.save()
15             return show_movie(request)
16     data={}
17     form=MovieForm()
18     data['movie_form']=form
19     return render(request,'movie_modelform/add_movie.html',context=data)
20

```

h. Output



Add your movie

Name:

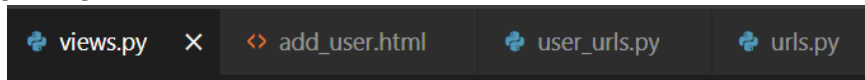
Genre:

5 • `select * from movie_modelform_movie;`

Result Grid			
Filter Rows:			
Edit: Export/Import: Wrap Cell			
id	name	genre	
1	golmaal	comedy	
2	drishyam	crime	
NULL	NULL	NULL	

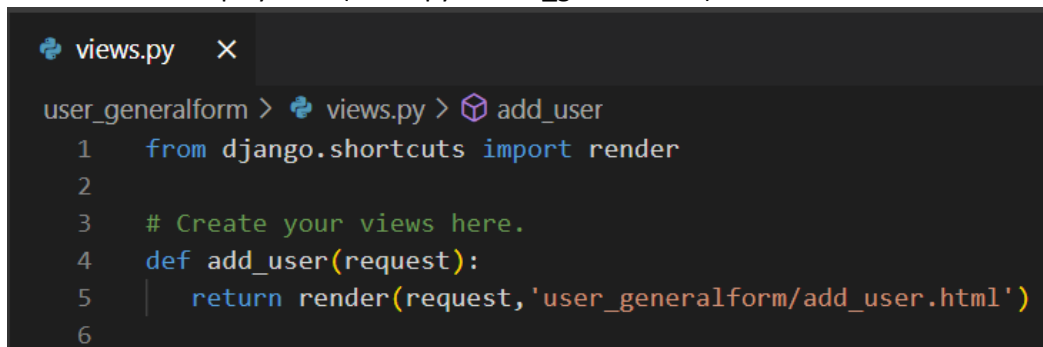
i. DONE!

15. Working with general form

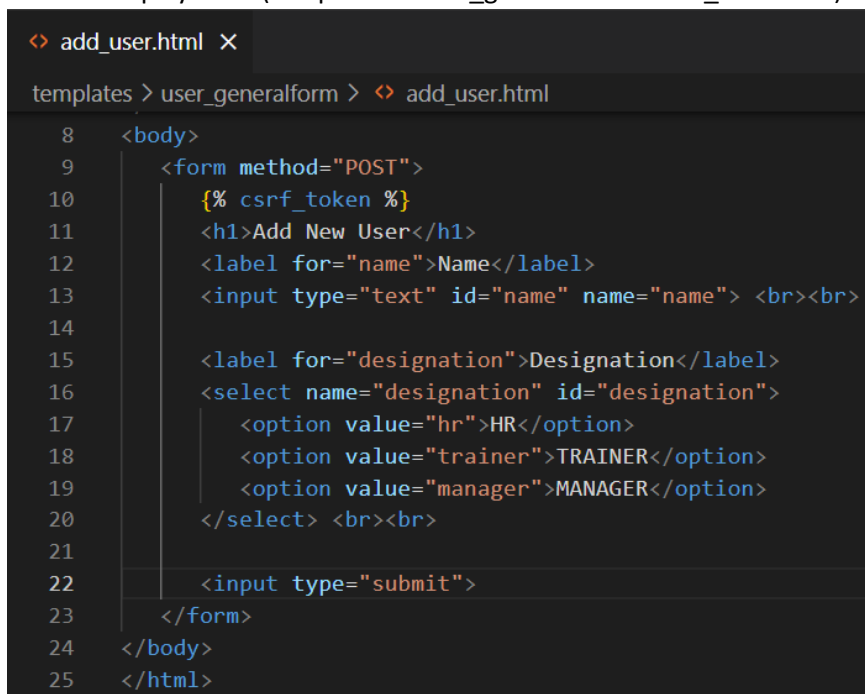


No need to create forms.py since we are creating general form

a. Create view to display form (views.py of user_generalform)



b. Code to display form (templates > user_gerenalform > add_user.html)



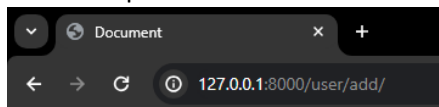
- c. Create user_urls.py (user_generalform > user_urls.py) to create application level url

```
user_urls.py X
user_generalform > user_urls.py > ...
1 from django.urls import path
2 from user_generalform import views
3
4 urlpatterns = [
5     path('add/', views.add_user),
6 ]
```

- d. Register above user_urls.py in urls.py

```
urls.py X
forms > urls.py > ...
15 2. Add a URL to urlpatterns: path('blog/', include('blog.urls
16 """
17 from django.contrib import admin
18 from django.urls import path,include
19
20 urlpatterns = [
21     path('admin/', admin.site.urls),
22     path('product/',include('product_djangoform.product_urls')),
23     path('movie/',include('movie_modelform.movie_urls')),
24     path('user/',include('user_generalform.user_urls'))
25 ]
26
```

- e. Check output



Add New User

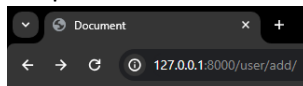
Name

Designation

- f. Code to insert data into table (update views.py)

```
views.py X
user_generalform > views.py > ...
1 from django.shortcuts import render
2 from django.http import HttpResponse
3 from user_generalform.models import User
4
5 # Create your views here.
6 def show_user(request):
7     return HttpResponse("User added to the table")
8
9 def add_user(request):
10     if request.method == 'POST':
11         user_name=request.POST['name']
12         user_designation=request.POST['designation']
13         user=User.objects.create(name=user_name,designation=user_designation)
14         user.save()
15         return show_user(request)
16     return render(request,'user_generalform/add_user.html')
17
```

g. Output



Add New User

Name

Designation

6 • `select * from user_generalform_user;`

Result Grid			
	id	name	designation
1	1	tejas	trainer
2	2	raj	manager
3	3	rajat	manager
	NULL	NULL	NULL

16. FORM DONE !!!!!!!!!!!!!!!!!!!!!!!

17. Displaying added data into table.

Here I am going to show how to display usres data after adding new user. You hve to do same for movie and product

a. Create view to fetch all users

```
views.py
user_generalform > views.py > add_user
1 from django.shortcuts import render
2 from django.http import HttpResponse
3 from user_generalform.models import User
4
5 # Create your views here.
6 def show_user(request):
7     #return HttpResponse("User added to the table")
8     data={}
9     all_users = User.objects.all()
10    data['users']=all_users
11    return render(request,'user_generalform/show_user.html',context=data)
```

b. Pass fetched users data to show_user.html (create show_user.html in) templates > user

```
show_user.html
templates > user_generalform > show_user.html
7 </head>
8 <body>
9     <table align="center" border="1" cellspacing="0" cellpadding="10">
10         <tr>
11             <th>ID</th>
12             <th>NAME</th>
13             <th>DESIGNATION</th>
14         </tr>
15         {% for user in users %}
16         <tr>
17             <td>{{user.id}}</td>
18             <td>{{user.name}}</td>
19             <td>{{user.designation}}</td>
20         </tr>
21         {% endfor %}
22     </table>
23 </body>
24 </html>
```

c. Create url for view

```
user_urls.py
user_generalform > user_urls.py > ...
1 from django.urls import path
2 from user_generalform import views
3
4 urlpatterns = [
5     path('add/', views.add_user),
6     path('show/', views.show_user),
7 ]
```

d. Output



ID	NAME	DESIGNATION
1	tejas	trainer
2	raj	manager
3	rajat	manager
4	abhijeet	manager

18. DONE ALL !!!!!!!!!!!!!!!