User Management Ex
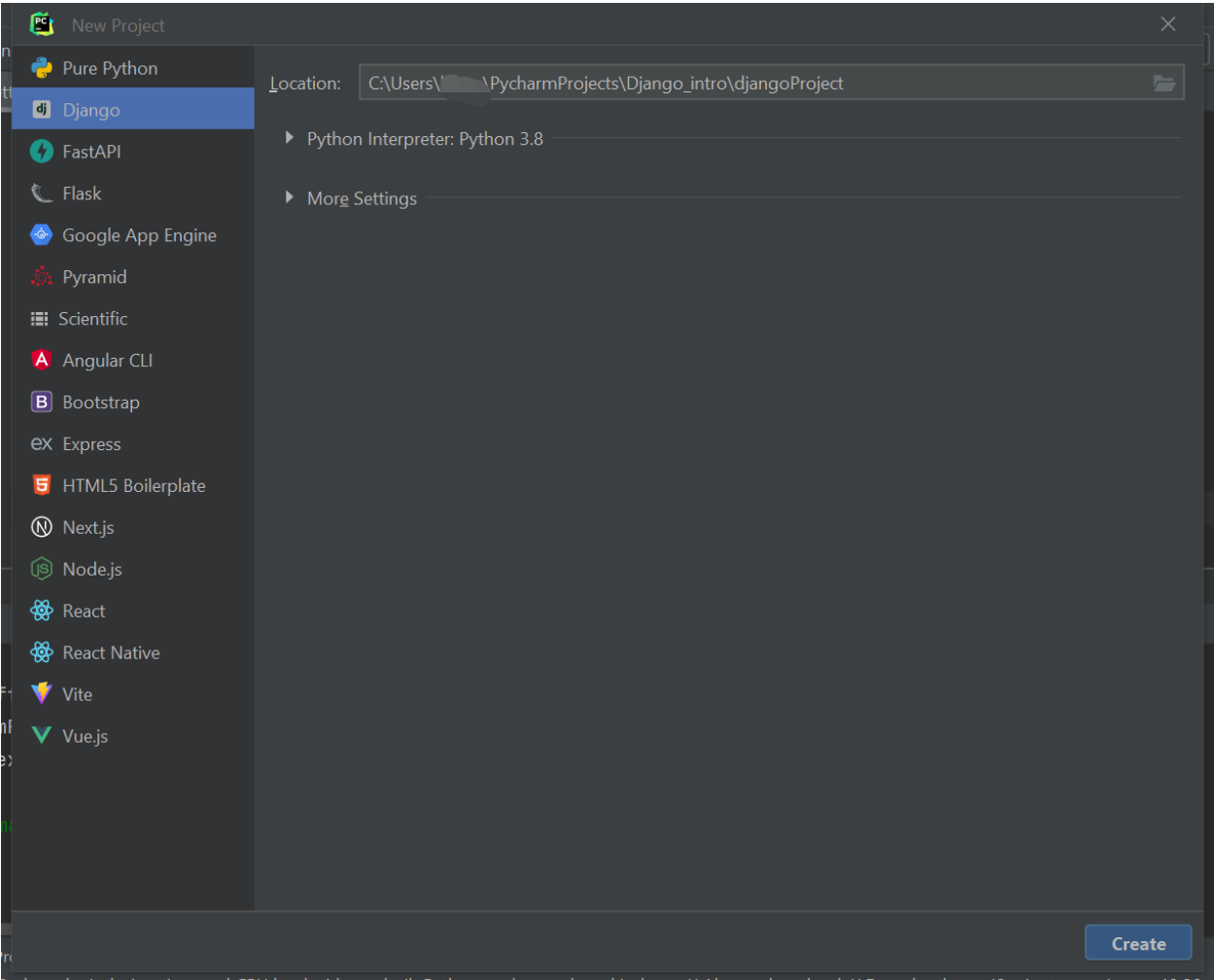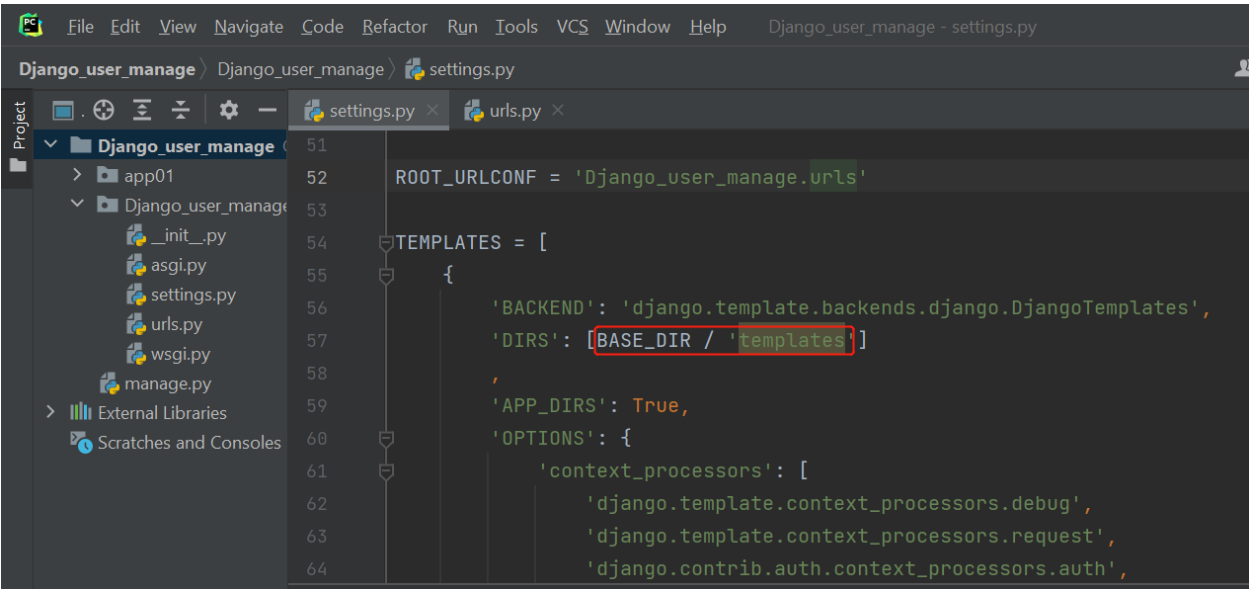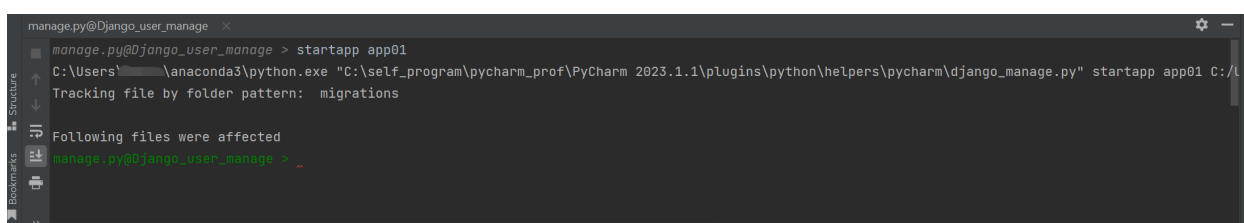
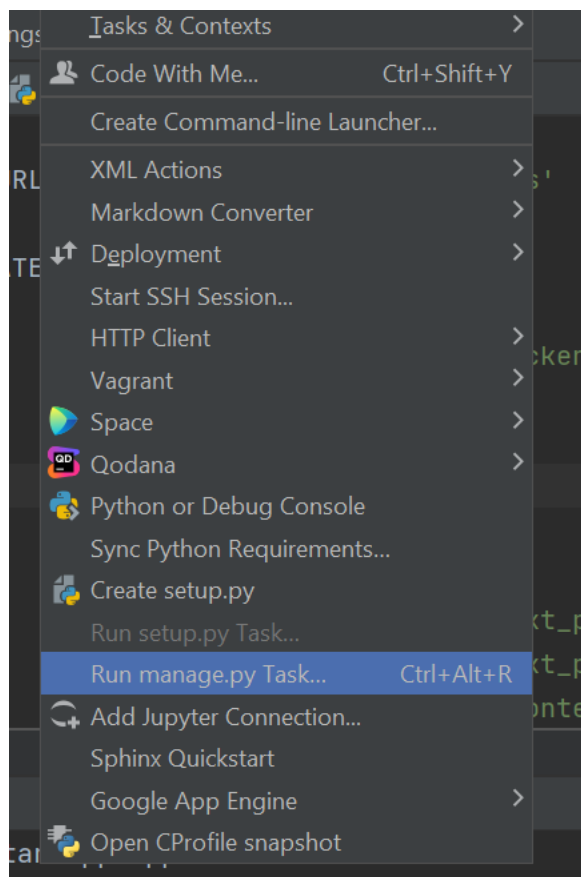## 1. create Django

delete:

1. templates folder
2. 'DIRS'
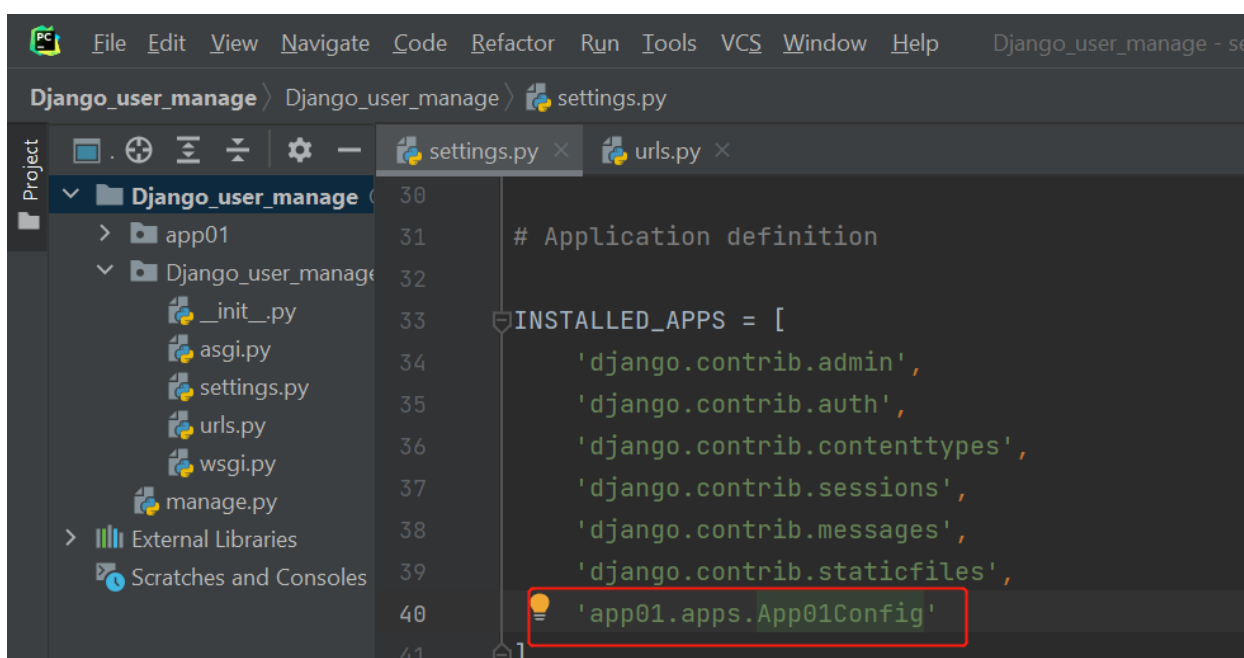

in settings.py

2. create app

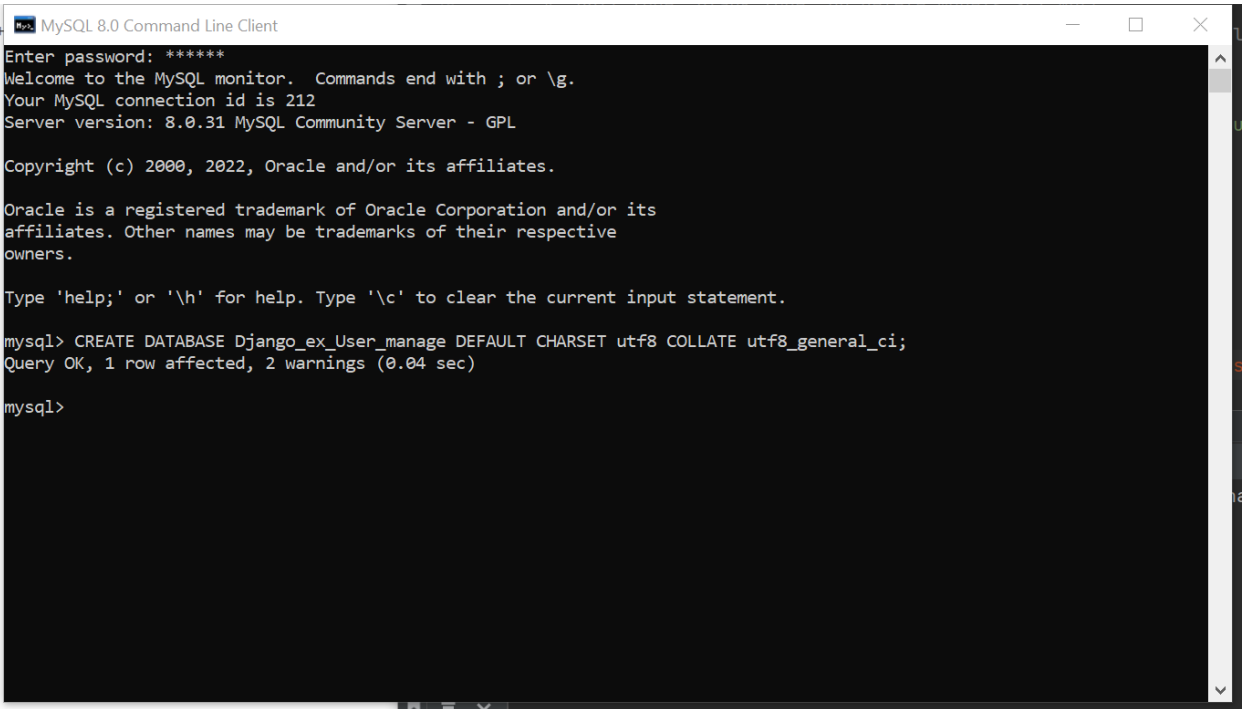Tools→Run manage.py Task

3. install app

in settings.py

4. connect to MySQL

1) create database

```
CREATE DATABASE Django_ex_User_manage DEFAULT CHARSET utf8 COLLATE utf8_general_ci;
```

2) connect to MySQL in Django project

in settings.py
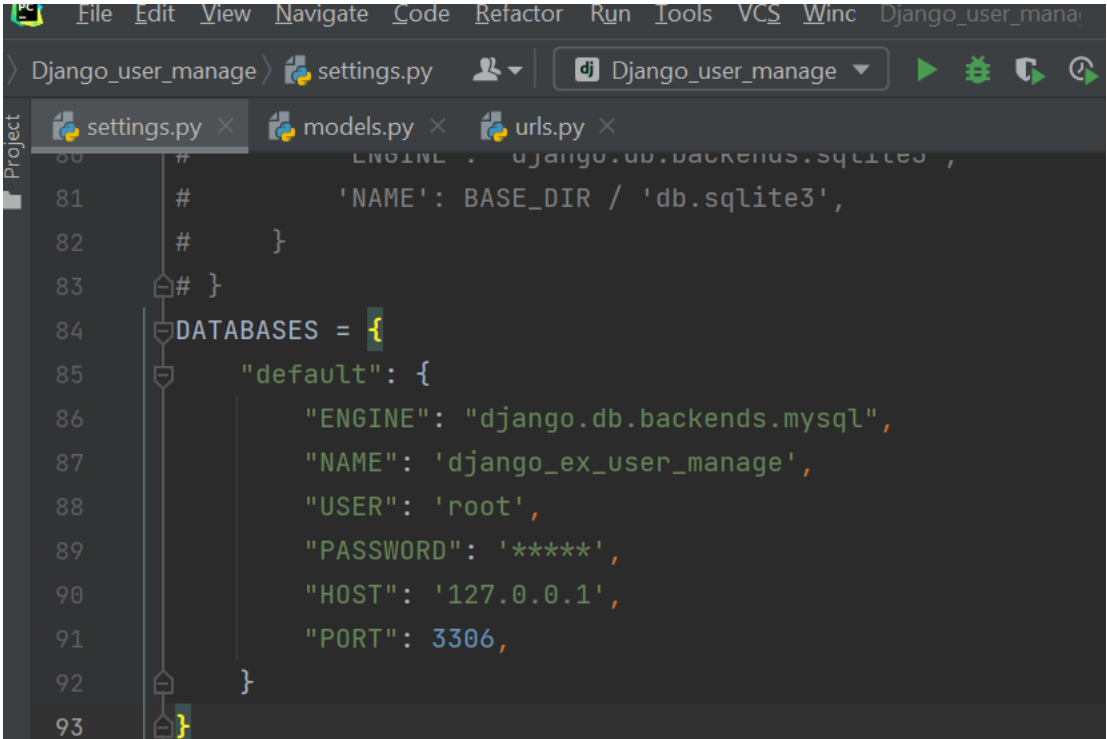

```
DATABASES = {
    "default": {
        "ENGINE": "django.db.backends.mysql",
        "NAME": 'django_ex_user_manage',
        "USER": 'root',
        "PASSWORD": '******',
        "HOST": '127.0.0.1',
        "PORT": 3306,
    }
}
```

3) create tables

```
# models.py
from django.db import models

# Create your models here.
class Department(models.Model):
    """department table"""
    department = models.CharField(verbose_name='Department', max_length=32)

class UserInfo(models.Model):
    """employee info"""
    name = models.CharField(verbose_name='Name', max_length=16)
    password = models.CharField(verbose_name='Password', max_length=64)
    age = models.IntegerField(verbose_name='Age')
    salary = models.DecimalField(verbose_name='Salary', max_digits=10,\
```

```
                                  decimal_places=2, default=2000)
    create_time = models.DateTimeField(verbose_name='EntryDate')
    """
    Join Department
    On id
    """
    # this column will be named as depart_id
    # only id in Department table can appear in this col
    # if delete rows in Department table, also drop related records in this table
    # "on_delete=models.CASCADE"
    depart = models.ForeignKey(to='Department', to_field='id', on_delete=models.CASCADE)
    # if delete rows in Department table, set related records NA in this table
    # "null=True, blank=True, on_delete=models.SET_NULL"
    # depart = models.ForeignKey(to='Department', to_field='id', null=True, blank=True, on_delete=models.SET_NULL)

    """
    also, if we do not want to connect to other tables, and the values are really simple,
    like "gender", only male and female,
    we can also use tuple
    """
    gender_choices = (
        (1, 'male'),
        (2, 'female')
    )
    gender = models.SmallIntegerField(verbose_name='gender', choices=gender_choices)
```

```
manage.py@Django_user_manage   ×
manage.py@Django_user_manage > makemigrations
C:\Users_____\anaconda3\python.exe "C:\self_program\pycharm_prof\PyCharm 2023.1.1\plugins\python\helpers\pycharm\django_manage.py" makemigrations C:\
Tracking file by folder pattern:  migrations
Migrations for 'app01':
  app01\migrations\0001_initial.py
    - Create model Department
    - Create model UserInfo

Following files were affected
 C:\Users_____\PycharmProjects\Django_intro\Django_user_manage\app01\migrations\0001_initial.py
Process finished with exit code 0
manage.py@Django_user_manage > migrate
C:\Users_____\anaconda3\python.exe "C:\self_program\pycharm_prof\PyCharm 2023.1.1\plugins\python\helpers\pycharm\django_manage.py" migrate C:/Users/
Tracking file by folder pattern:  migrations
Operations to perform:
  Apply all migrations: admin, app01, auth, contenttypes, sessions
Running migrations:
  Applying contenttypes.0001_initial... OK
  Applying auth.0001_initial... OK
  Applying admin.0001_initial... OK
manage.py@Django_user_manage > _
```

5. department page

1) build relationship between urls and functions

```python
# urls.py
from django.contrib import admin
from django.urls import path
from app01 import views

urlpatterns = [
    # path('admin/', admin.site.urls),
    path('depart/list/', views.depart_list),
]
```

2) create page

```html
<!-- depart_list.html -->
{% load static %}

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Title</title>
    <link rel="stylesheet" href="{% static "plugins/bootstrap-3.4.1-dist/css/bootstrap.css" %}">

</head>
<body>
<nav class="navbar navbar-default">
    <div class="container">
        <!-- Brand and toggle get grouped for better mobile display -->
        <div class="navbar-header">
            <button type="button" class="navbar-toggle collapsed" data-toggle="collapse"
                    data-target="#bs-example-navbar-collapse-1" aria-expanded="false">
                <span class="sr-only">Toggle navigation</span>
                <span class="icon-bar"></span>
                <span class="icon-bar"></span>
                <span class="icon-bar"></span>
            </button>
            <a class="navbar-brand" href="#">Employee Department Management</a>
        </div>

        <!-- Collect the nav links, forms, and other content for toggling -->
        <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
            <ul class="nav navbar-nav">
                {#              <li class="active"><a href="#">Link <span class="sr-only">(current)</span></a></li>#}
```
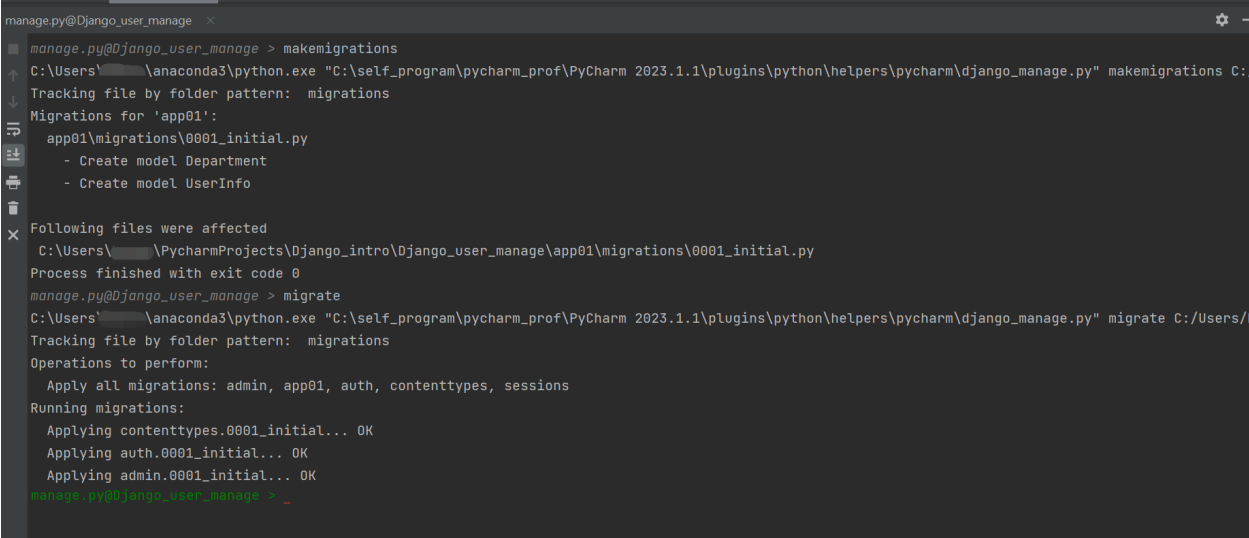
```
                    <li><a href="/depart/list/">Department Management</a></li>
                    <li><a href="#">User Management</a></li>

                </ul>

                <ul class="nav navbar-nav navbar-right">
                    <li><a href="#">Login</a></li>
                    <li class="dropdown">
                        <a href="#" class="dropdown-toggle" data-toggle="dropdown" role="button" aria-haspopup="true"
                            aria-expanded="false">Dropdown <span class="caret"></span></a>
                        <ul class="dropdown-menu">
                            <li><a href="#">Action</a></li>
                            <li><a href="#">Another action</a></li>
                            <li><a href="#">Something else here</a></li>
                            <li role="separator" class="divider"></li>
                            <li><a href="#">Separated link</a></li>
                        </ul>
                    </li>
                </ul>
            </div><!-- /.navbar-collapse -->
        </div><!-- /.container-fluid -->
    </nav>

    <div>
        <div class="container">
            <div>
                <a href="####" class="btn btn-success" style="margin-bottom: 10px">
                    <span class="glyphicon glyphicon-plus" aria-hidden="true"></span>
                    Create New Department
                </a>
            </div>

            <div class="bs-example" data-example-id="panel-without-body-with-table">
                <div class="panel panel-default">
                    <!-- Default panel contents -->
                    <div class="panel-heading">Panel heading</div>

                    <!-- Table -->
                    <table class="table">
                        <thead>
                        <tr>
                            <th>ID</th>
                            <th>Department Name</th>
                            <th>Operations</th>
                        </tr>
                        </thead>
                        <tbody>
                        <tr>
                            <th scope="row">1</th>
                            <td>Department 1</td>
                            <td>
                                <a class="btn btn-primary btn-sm" href="##">Edit</a>
                                <a class="btn btn-danger btn-sm" href="##">Delete</a>
                            </td>
                        </tr>

                        </tbody>
                    </table>
                </div>
            </div>

        </div>
    </div>

    <script src="{% static "js/jquery-3.6.4.min.js" %}"></script>
    <script src="{% static "plugins/bootstrap-3.4.1-dist/js/bootstrap.min.js" %}"></script>
</body>
</html>
```
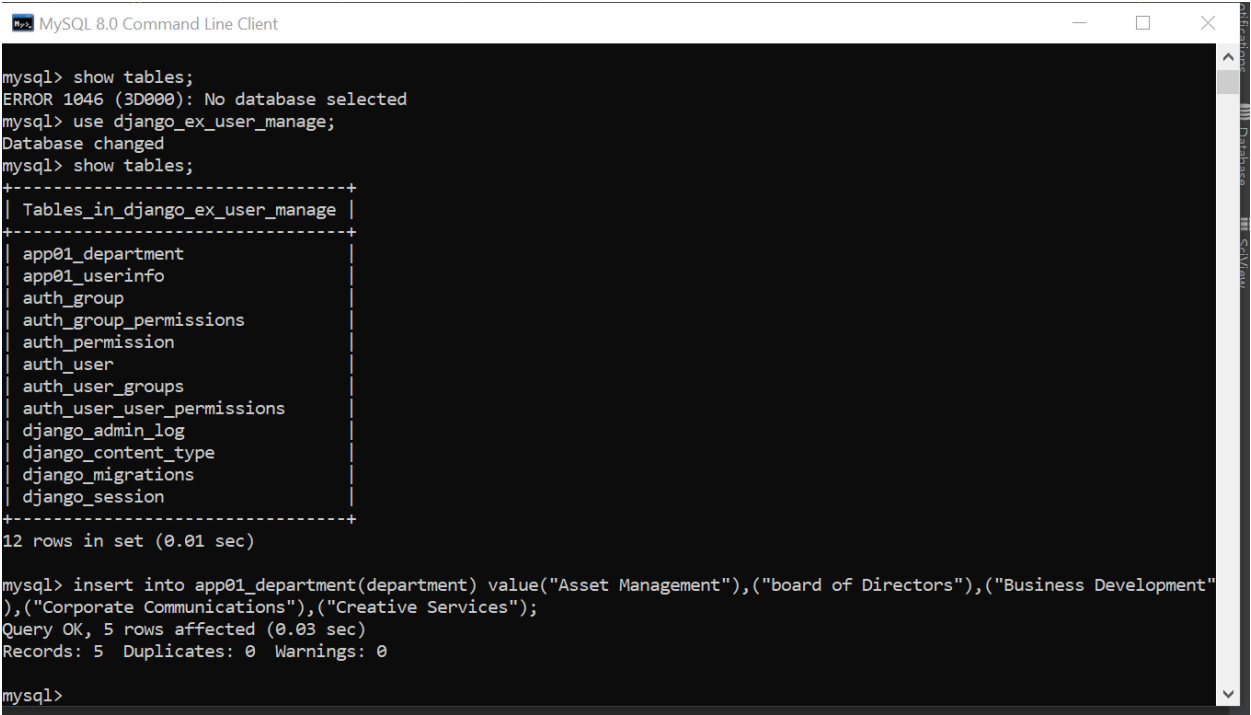
```python
# views.py
from django.shortcuts import render

# Create your views here.
def depart_list(request):


    return render(request, 'depart_list.html')
```

3) add primary data into the table

```
MySQL 8.0 Command Line Client                                                    —    □    ×

mysql> show tables;
ERROR 1046 (3D000): No database selected
mysql> use django_ex_user_manage;
Database changed
mysql> show tables;
+-------------------------------+
| Tables_in_django_ex_user_manage |
+-------------------------------+
| app01_department              |
| app01_userinfo                |
| auth_group                    |
| auth_group_permissions        |
| auth_permission               |
| auth_user                     |
| auth_user_groups              |
| auth_user_user_permissions    |
| django_admin_log              |
| django_content_type           |
| django_migrations            |
| django_session                |
+-------------------------------+
12 rows in set (0.01 sec)

mysql> insert into app01_department(department) value("Asset Management"),("board of Directors"),("Business Development"
),("Corporate Communications"),("Creative Services");
Query OK, 5 rows affected (0.03 sec)
Records: 5  Duplicates: 0  Warnings: 0

mysql>
```

4) get data from dataset

```python
# views.py
def depart_list(request):
    # get data from dataset
    queryset = models.Department.objects.all()

    return render(request, 'depart_list.html', {'queryset': queryset})
```

```html
<!-- depart_list.html -->
<tbody>
{% for obj in queryset %}
    <tr>
        <th scope="row">{{ obj.id }}</th>
        <td>{{ obj.department }}</td>
        <td>
            <a class="btn btn-primary btn-sm" href="##">Edit</a>
            <a class="btn btn-danger btn-sm" href="##">Delete</a>
        </td>
    </tr>
{% endfor %}
</tbody>
```

| Employee Department Management | Department Management | User Management | | Login | Dropdown ▾ |

+ Create New Department

Panel heading		
ID	**Department Name**	**Operations**
1	Asset Management	Edit Delete
2	board of Directors	Edit Delete
3	Business Development	Edit Delete
4	Corporate Communications	Edit Delete
5	Creative Services	Edit Delete

5) add department

```html
<!-- depart_list.html -->
<a href="/depart/add/" class="btn btn-success" style="margin-bottom: 10px">
    <span class="glyphicon glyphicon-plus" aria-hidden="true"></span>
    Create New Department
</a>
```

```html
<!-- department_add.html -->
{% load static %}

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Title</title>
    <link rel="stylesheet" href="{% static "plugins/bootstrap-3.4.1-dist/css/bootstrap.css" %}">

</head>
<body>
<nav class="navbar navbar-default">
    <div class="container">
        <!-- Brand and toggle get grouped for better mobile display -->
        <div class="navbar-header">
            <button type="button" class="navbar-toggle collapsed" data-toggle="collapse"
                    data-target="#bs-example-navbar-collapse-1" aria-expanded="false">
                <span class="sr-only">Toggle navigation</span>
                <span class="icon-bar"></span>
                <span class="icon-bar"></span>
                <span class="icon-bar"></span>
            </button>
            <a class="navbar-brand" href="#">Employee Department Management</a>
        </div>

        <!-- Collect the nav links, forms, and other content for toggling -->
        <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
            <ul class="nav navbar-nav">
                {#                    <li class="active"><a href="#">Link <span class="sr-only">(current)</span></a></li>#}
                <li><a href="/depart/list/">Department Management</a></li>
                <li><a href="#">User Management</a></li>

            </ul>

            <ul class="nav navbar-nav navbar-right">
                <li><a href="#">Login</a></li>
                <li class="dropdown">
                    <a href="#" class="dropdown-toggle" data-toggle="dropdown" role="button" aria-haspopup="true"
                       aria-expanded="false">Dropdown <span class="caret"></span></a>
                    <ul class="dropdown-menu">
                        <li><a href="#">Action</a></li>
                        <li><a href="#">Another action</a></li>
                        <li><a href="#">Something else here</a></li>
                        <li role="separator" class="divider"></li>
                        <li><a href="#">Separated link</a></li>
                    </ul>
                </li>
            </ul>
        </div><!-- /.navbar-collapse -->
    </div><!-- /.container-fluid -->
</nav>

<div>
    <div class="container">
        <div class="panel panel-default">
            <div class="panel-heading"> New Apartment</div>
            <div class="panel-body">
                <form method="post">
                    {% csrf_token %}
                    <div class="form-group">
                        <label for="exampleInputEmail1">Department Name</label>
                        <input type="text" class="form-control" name='Depart' placeholder="Department name">
                    </div>

                    <div class="form-group">
                            <button type="submit" class="btn btn-primary">Save</button>
                    </div>
                </form>
            </div>
        </div>

    </div>
</div>

<script src="{% static "js/jquery-3.6.4.min.js" %}"></script>
<script src="{% static "plugins/bootstrap-3.4.1-dist/js/bootstrap.min.js" %}"></script>
</body>
</html>
```

Employee Department Management	Department Management	User Management		Login	Dropdown ▾

+ Create New Department

Panel heading			
ID	**Department Name**	**Operations**	
1	Asset Management	Edit Delete	
2	board of Directors	Edit Delete	
3	Business Development	Edit Delete	
4	Corporate Communications	Edit Delete	
5	Creative Services	Edit Delete	
6	Customer Service	Edit Delete	

add Customer Service for later Edit function test

```
# views.py
def depart_add(request):
    if request.method == 'GET':
        return render(request, 'depart_add.html')

    depart_title = request.POST.get('Depart')
    models.Department.objects.create(department=depart_title)
    return redirect('/depart/list/')
```

later also add one for later delete function test

6) delete department

```
<!-- depart_list.html -->
<td>
    <a class="btn btn-primary btn-sm" href="##">Edit</a>
    <a class="btn btn-danger btn-sm" href="/depart/delete/?nid={{ obj.id }}">Delete</a>
</td>
```

```
# views.py
def depart_delete(request):
    nid = request.GET.get('nid')
    models.Department.objects.filter(id=nid).delete()

    return redirect('/depart/list/')
```

Employee Department Management	Department Management	User Management		Login	Dropdown ▾

+ Create New Department

Panel heading			
ID	**Department Name**	**Operations**	
1	Asset Management	Edit Delete	
2	board of Directors	Edit Delete	
3	Business Development	Edit Delete	
4	Corporate Communications	Edit Delete	
5	Creative Services	Edit Delete	
6	Customer Service	Edit Delete	
7	Customer Service	Edit Delete	

Employee Department Management	Department Management	User Management		Login	Dropdown ▾

+ Create New Department

Panel heading		
ID	**Department Name**	**Operations**
1	Asset Management	Edit Delete
2	board of Directors	Edit Delete
3	Business Development	Edit Delete
4	Corporate Communications	Edit Delete
5	Creative Services	Edit Delete
7	Customer Service	Edit Delete

7) edit department

```
# urls.py
path('depart/<int:nid>/edit/', views.depart_edit),
# id will appeared at <int:nid>
```

```
# views.py
def depart_edit(request, nid):
    if request.method == 'GET':
        row_obj = models.Department.objects.filter(id=nid).first()
        return render(request, 'depart_edit.html', {"row_obj": row_obj})

    depart = request.POST.get('Depart')
    models.Department.objects.filter(id=nid).update(department=depart)
    return redirect('/depart/list/')
```

```html
<!-- depart_list.html -->
<td>
    <a class="btn btn-primary btn-sm" href="/depart/{{ obj.id }}/edit/">Edit</a>
    <a class="btn btn-danger btn-sm" href="/depart/delete/?nid={{ obj.id }}">Delete</a>
</td>
```

```html
<!-- depart_edit.html -->
{% load static %}

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Title</title>
    <link rel="stylesheet" href="{% static "plugins/bootstrap-3.4.1-dist/css/bootstrap.css" %}">

</head>
<body>
<nav class="navbar navbar-default">
    <div class="container">
        <!-- Brand and toggle get grouped for better mobile display -->
        <div class="navbar-header">
            <button type="button" class="navbar-toggle collapsed" data-toggle="collapse"
                    data-target="#bs-example-navbar-collapse-1" aria-expanded="false">
                <span class="sr-only">Toggle navigation</span>
                <span class="icon-bar"></span>
                <span class="icon-bar"></span>
                <span class="icon-bar"></span>
            </button>
            <a class="navbar-brand" href="#">Employee Department Management</a>
        </div>

        <!-- Collect the nav links, forms, and other content for toggling -->
        <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
            <ul class="nav navbar-nav">
                {#                  <li class="active"><a href="#">Link <span class="sr-only">(current)</span></a></li>#}
                <li><a href="/depart/list/">Department Management</a></li>
                <li><a href="#">User Management</a></li>

            </ul>

            <ul class="nav navbar-nav navbar-right">
                <li><a href="#">Login</a></li>
                <li class="dropdown">
                    <a href="#" class="dropdown-toggle" data-toggle="dropdown" role="button" aria-haspopup="true"
                       aria-expanded="false">Dropdown <span class="caret"></span></a>
                    <ul class="dropdown-menu">
                        <li><a href="#">Action</a></li>
                        <li><a href="#">Another action</a></li>
                        <li><a href="#">Something else here</a></li>
                        <li role="separator" class="divider"></li>
                        <li><a href="#">Separated link</a></li>
```

```
                    </ul>
                </li>
            </ul>
        </div><!-- /.navbar-collapse -->
    </div><!-- /.container-fluid -->
</nav>

<div>
    <div class="container">
        <div class="panel panel-default">
            <div class="panel-heading"> Edit Apartment Name </div>
            <div class="panel-body">
                <form method="post">
                    {% csrf_token %}
                    <div class="form-group">
                        <label>Department Name</label>
                        <input type="text" class="form-control" name='Depart' value="{{ row_obj.department }}" placeholder="Department name
                    </div>

                    <div class="form-group">
                        <button type="submit" class="btn btn-primary">Update</button>
                    </div>
                </form>
            </div>
        </div>

    </div>
</div>

<script src="{% static "js/jquery-3.6.4.min.js" %}"></script>
<script src="{% static "plugins/bootstrap-3.4.1-dist/js/bootstrap.min.js" %}"></script>
</body>
</html>
```

Employee Department Management   Department Management   User Management   Login   Dropdown ▾

+ Create New Department

Panel heading		
ID	**Department Name**	**Operations**
1	Asset Management	Edit Delete
2	board of Directors	Edit Delete
3	Business Development	Edit Delete
4	Corporate Communications	Edit Delete
5	Creative Services	Edit Delete
7	Customer Service	Edit Delete

Employee Department Management   Department Management   User Management   Login   Dropdown ▾

Edit Apartment Name

Department Name

Customer Service

Update

Employee Department Management   Department Management   User Management   Login   Dropdown ▾

Edit Apartment Name

Department Name

Customer Experience

Update

Employee Department Management   Department Management   User Management   Login   Dropdown ▾

+ Create New Department

Panel heading

ID	Department Name	Operations
1	Asset Management	Edit Delete
2	board of Directors	Edit Delete
3	Business Development	Edit Delete
4	Corporate Communications	Edit Delete
5	Creative Services	Edit Delete
7	Customer Experience	Edit Delete

5. Template inheritance

1) create template

```
{% load static %}

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Title</title>
    <link rel="stylesheet" href="{% static "plugins/bootstrap-3.4.1-dist/css/bootstrap.css" %}">

</head>
<body>
<nav class="navbar navbar-default">
    <div class="container">
        <!-- Brand and toggle get grouped for better mobile display -->
        <div class="navbar-header">
            <button type="button" class="navbar-toggle collapsed" data-toggle="collapse"
                    data-target="#bs-example-navbar-collapse-1" aria-expanded="false">
                <span class="sr-only">Toggle navigation</span>
                <span class="icon-bar"></span>
                <span class="icon-bar"></span>
                <span class="icon-bar"></span>
            </button>
            <a class="navbar-brand" href="#">Employee Department Management</a>
        </div>

        <!-- Collect the nav links, forms, and other content for toggling -->
        <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
            <ul class="nav navbar-nav">
                {#              <li class="active"><a href="#">Link <span class="sr-only">(current)</span></a></li>#}
                <li><a href="/depart/list/">Department Management</a></li>
                <li><a href="#">User Management</a></li>

            </ul>

            <ul class="nav navbar-nav navbar-right">
                <li><a href="#">Login</a></li>
                <li class="dropdown">
                    <a href="#" class="dropdown-toggle" data-toggle="dropdown" role="button" aria-haspopup="true"
                       aria-expanded="false">Dropdown <span class="caret"></span></a>
```

```
                        <ul class="dropdown-menu">
                            <li><a href="#">Action</a></li>
                            <li><a href="#">Another action</a></li>
                            <li><a href="#">Something else here</a></li>
                            <li role="separator" class="divider"></li>
                            <li><a href="#">Separated link</a></li>
                        </ul>
                    </li>
                </ul>
            </div><!-- /.navbar-collapse -->
        </div><!-- /.container-fluid -->
    </nav>

    <div>
        {% block content %}{% endblock %}
    </div>

    <script src="{% static "js/jquery-3.6.4.min.js" %}"></script>
    <script src="{% static "plugins/bootstrap-3.4.1-dist/js/bootstrap.min.js" %}"></script>
    </body>
    </html>
```

2) inherit ex

```
{% extends 'layout.html' %}

{% block content %}
    <h1>Front page</h1>
{% endblock %}
```

3) inherit in previous code

```
<!-- depart_list.html -->
{% extends 'layout.html' %}

{% block content %}
<div class="container">
        <div>
            <a href="/depart/add/" class="btn btn-success" style="margin-bottom: 10px">
                <span class="glyphicon glyphicon-plus" aria-hidden="true"></span>
                Create New Department
            </a>
        </div>

        <div class="bs-example" data-example-id="panel-without-body-with-table">
            <div class="panel panel-default">
                <!-- Default panel contents -->
                <div class="panel-heading">Panel heading</div>

                <!-- Table -->
                <table class="table">
                    <thead>
                    <tr>
                        <th>ID</th>
                        <th>Department Name</th>
                        <th>Operations</th>
                    </tr>
                    </thead>
                    <tbody>
                    {% for obj in queryset %}
                    <tr>
                        <th scope="row">{{ obj.id }}</th>
                        <td>{{ obj.department }}</td>
                        <td>
                            <a class="btn btn-primary btn-sm" href="/depart/{{ obj.id }}/edit/">Edit</a>
                            <a class="btn btn-danger btn-sm" href="/depart/delete/?nid={{ obj.id }}">Delete</a>
                        </td>
                    </tr>
                    {% endfor %}

                    </tbody>
                </table>
            </div>
        </div>

    </div>
{% endblock %}
```

also we have

```
{% block css %}{% endblock %}
```

```
{% block js %}{% endblock %}
```

for special css and js we defined

6. User management

<1> insert new data

```
insert into app01_userinfo(name, password, age, salary, create_time, gender, depart_id) values("Joseph", "9j0o2s3elkjh", 35, 7659.65, "2018-
insert into app01_userinfo(name, password, age, salary, create_time, gender, depart_id) values("Nancy", "y8c29n1auhb", 27, 6521.25, "2020-0
insert into app01_userinfo(name, password, age, salary, create_time, gender, depart_id) values("Jenny", "j4n3n9edcijn", 24, 6241.25, "2022-
insert into app01_userinfo(name, password, age, salary, create_time, gender, depart_id) values("Andy", "19376y5d2a6n", 24, 6241.25, "2022-0
```


<2> Foreign key

```
<td>{{ obj.depart_id }}</td>
```



```
<td>{{ obj.depart.department }}</td>
```

ID	Name	Password	age	salary	Enter time	Gender	Department	Operations
1	Joseph	9j0o2s3elkjh	35	7659.65	Feb. 6, 2018, midnight	1	Business Development	Edit Delete
2	Nancy	y8c29n1auhb	27	6521.25	July 15, 2020, midnight	2	Asset Management	Edit Delete
3	Jenny	j4n3n9edcijn	24	6241.25	May 20, 2022, midnight	2	board of Directors	Edit Delete
4	Andy	19376y5d2a6n	24	6241.25	May 20, 2022, midnight	1	Corporate Communications	Edit Delete

Employee Department Management · Department Management · User Management · Login · Dropdown ▾

+ Enter new User

Panel heading

Because when we create table, we use:

```
depart = models.ForeignKey(to='Department', to_field='id', on_delete=models.CASCADE)
```

in the table this col will be named as depart_id, if we want to get the exact name of the department, just use the original name of the col.

```
def user_list(request):
    queryset = models.UserInfo.objects.all()
    for obj in queryset:
        print(obj.id, obj.name, obj.salary, obj.age)
        print(obj.create_time, obj.create_time.strftime('%Y-%m-%d'))
        print(obj.gender, obj.get_gender_display())
        print(obj.depart_id, obj.depart.department)

    return render(request, 'user_list.html', {'queryset': queryset})
```

```
May 12, 2023 - 18:22:45
Django version 4.2.1, using settings 'Django_user_manage.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.

1 Joseph 7659.65 35
2018-02-06 00:00:00+00:00 2018-02-06
1 male
3 Business Development
2 Nancy 6521.25 27
2020-07-15 00:00:00+00:00 2020-07-15
2 female
1 Asset Management
3 Jenny 6241.25 24
2022-05-20 00:00:00+00:00 2022-05-20
2 female
2 board of Directors
4 Andy 6241.25 24
2022-05-20 00:00:00+00:00 2022-05-20
1 male
4 Corporate Communications
[12/May/2023 18:22:45] "GET /user/list/ HTTP/1.1" 200 6187
```

```
<td>{{ obj.create_time|date:"Y-m-d" }}</td>
```

+ Enter new User

Panel heading

ID	Name	Password	age	salary	Enter time	Gender	Department	Operations
1	Joseph	9j0o2s3elkjh	35	7659.65	2018-02-06	1	Business Development	Edit Delete
2	Nancy	y8c29n1auhb	27	6521.25	2020-07-15	2	Asset Management	Edit Delete
3	Jenny	j4n3n9edcijn	24	6241.25	2022-05-20	2	board of Directors	Edit Delete
4	Andy	19376y5d2a6n	24	6241.25	2022-05-20	1	Corporate Communications	Edit Delete

```
<td>{{ obj.get_gender_display }}</td>
```

+ Enter new User

Panel heading

ID	Name	Password	age	salary	Enter time	Gender	Department	Operations
1	Joseph	9j0o2s3elkjh	35	7659.65	2018-02-06	male	Business Development	Edit Delete
2	Nancy	y8c29n1auhb	27	6521.25	2020-07-15	female	Asset Management	Edit Delete
3	Jenny	j4n3n9edcijn	24	6241.25	2022-05-20	female	board of Directors	Edit Delete
4	Andy	19376y5d2a6n	24	6241.25	2022-05-20	male	Corporate Communications	Edit Delete

<3> Form

```python
class MyForm(Form):
    user = forms.CharField(widget=forms.TextInput)
    pwd = forms.CharField(widget=forms.TextInput)
    email = forms.CharField(widget=forms.TextInput)

def user_add(request):
    if request.method == 'GET':
        form = MyForm()
        return render(request, 'user_add.html', {'form': form})
```

```html
<form method="post">
    {{ form.user }}
    {{ form.pwd }}
    {{ form.email }}
</form>
```

<4> ModelForm

```python
class MyForm(ModelForm):
    class Meta:
        model = UserInfo
        fields = ['name', 'password', 'age']
```
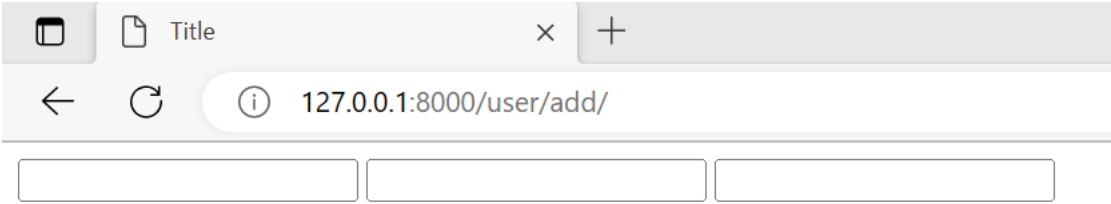
fields:

```
# all cols
fields = "__all__"
# except this cols
exclude = ['col name']
```

```
class UserModelForm(forms.ModelForm):
    class Meta:
        model = models.UserInfo
        fields = ['name', 'password', 'age']
def user_add(request):

    form = UserModelForm()

    return render(request, 'user_add.html', {'form': form})
```

```
<!-- user_add.html -->
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Title</title>
</head>
<body>
    <form method="post">
        {% csrf_token %}
        {{ form.name }}
        {{ form.password }}
        {{ form.age }}
    </form>
</body>
</html>
```


💡 {{ form.name.label }}: verbose_name

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Title</title>
</head>
<body>
    <form method="post">
        {% csrf_token %}
        {{ form.name.label }}:{{ form.name }}
        {{ form.password.label }}:{{ form.password }}
        {{ form.age.label }}:{{ form.age }}
    </form>
</body>
</html>
```


for loop:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Title</title>
</head>
<body>
    <form method="post">
        {% csrf_token %}
        {% for field in form %}
            {{ field.label }}: {{ field }}
        {% endfor %}
```

```
        </form>
    </body>
</html>
```
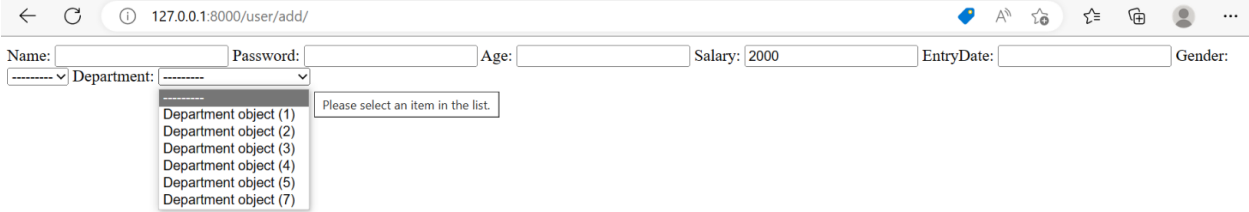
add more input:

```
class UserModelForm(forms.ModelForm):
    class Meta:
        model = models.UserInfo
        fields = ['name', 'password', 'age', 'salary', 'create_time', 'gender', 'depart']
def user_add(request):

    form = UserModelForm()

    return render(request, 'user_add.html', {'form': form})
```

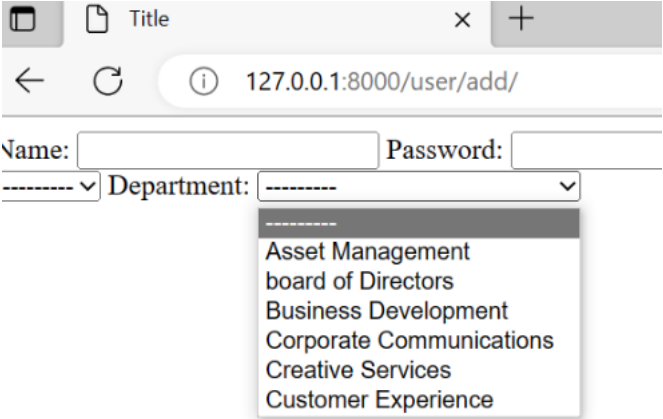This will cause trouble in department because what we get is an object, not values.


```
# models.py
class Department(models.Model):
    """department table"""
    department = models.CharField(verbose_name='Department', max_length=32)

    def __str__(self):
        return self.department
```



```
{% extends 'layout.html' %}

{% block content %}
    <div class="container">
        <div class="panel panel-default">
            <div class="panel-heading"> New User </div>
            <div class="panel-body">
                <form method="post">
                    {% csrf_token %}
                    {% for field in form %}
                        <label>{{ field.label }}</label>
                        {{ field }}
                    {% endfor %}
                    <div class="form-group">
                            <button type="submit" class="btn btn-primary">Save</button>
                    </div>
                </form>
            </div>
        </div>

    </div>
{% endblock %}
```

New User

Name

Password

Age

Salary 2000

EntryDate

Gender --------- ▾

Department ---------- ▾

Save

add style

```
class UserModelForm(forms.ModelForm):
    class Meta:
        model = models.UserInfo
        fields = ['name', 'password', 'age', 'salary', 'create_time', 'gender', 'depart']
        widgets = {
            "name": forms.TextInput(attrs={"class": "form-control"}),
            "password": forms.TextInput(attrs={"class": "form-control"}),
            "age": forms.TextInput(attrs={"class": "form-control"}),
            "salary": forms.TextInput(attrs={"class": "form-control"}),
            "create_time": forms.TextInput(attrs={"class": "form-control"}),
            "gender": forms.TextInput(attrs={"class": "form-control"}),
            "depart": forms.TextInput(attrs={"class": "form-control"})
        }
```

New User

Name

Password

Age

Salary

2000

EntryDate

Gender

Department

shorter way:

```
class UserModelForm(forms.ModelForm):
    class Meta:
        model = models.UserInfo
        fields = ['name', 'password', 'age', 'salary', 'create_time', 'gender', 'depart']

    def __init__(self, *args, **kwargs):
        super().__init__(*args, **kwargs)

        for name, field in self.fields.items():
            field.widget.attrs = {"class": "form-control", "placeholder": field.label}
```

error message:

```python
class UserModelForm(forms.ModelForm):
    class Meta:
        model = models.UserInfo
        fields = ['name', 'password', 'age', 'salary', 'create_time', 'gender', 'depart']

    def __init__(self, *args, **kwargs):
        super().__init__(*args, **kwargs)

        for name, field in self.fields.items():
            field.widget.attrs = {"class": "form-control", "placeholder": field.label}


def user_add(request):
    if request.method == 'GET':
        form = UserModelForm()

        return render(request, 'user_add.html', {'form': form})

    form = UserModelForm(data=request.POST)
    if form.is_valid():
        form.save()
        return redirect('/user/list/')

    return render(request, 'user_add.html', {'form': form})
```

```html
<div class="form-group">
        <label>{{ field.label }}</label>
        {{ field }}
        <span>{{ field.errors.0 }}</span>
</div>
```


password length check:

```python
password = forms.CharField(min_length=8, label='password')
```