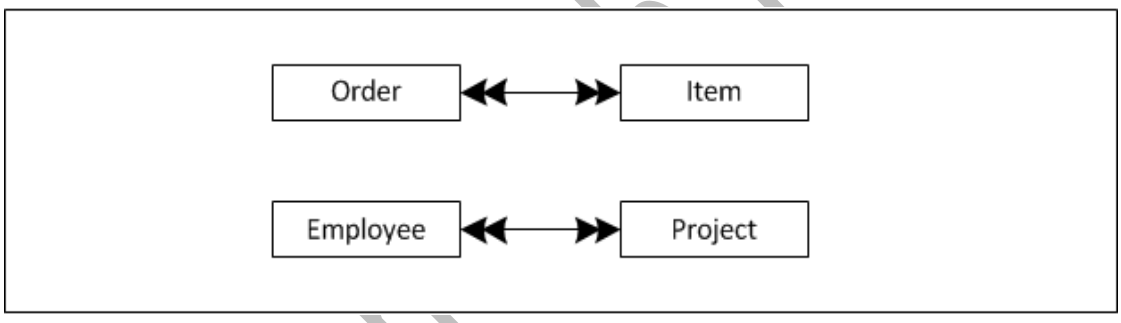
Many-to-many relationship

A *many-to-many* relationship exists when a row in one table has many related rows in a second table. Likewise, those related rows have many rows in the first table. The following figure shows examples of.

An order can contain many items, and an item can appear in many different orders

An employee can work on many projects, and a project can have many employees working on it

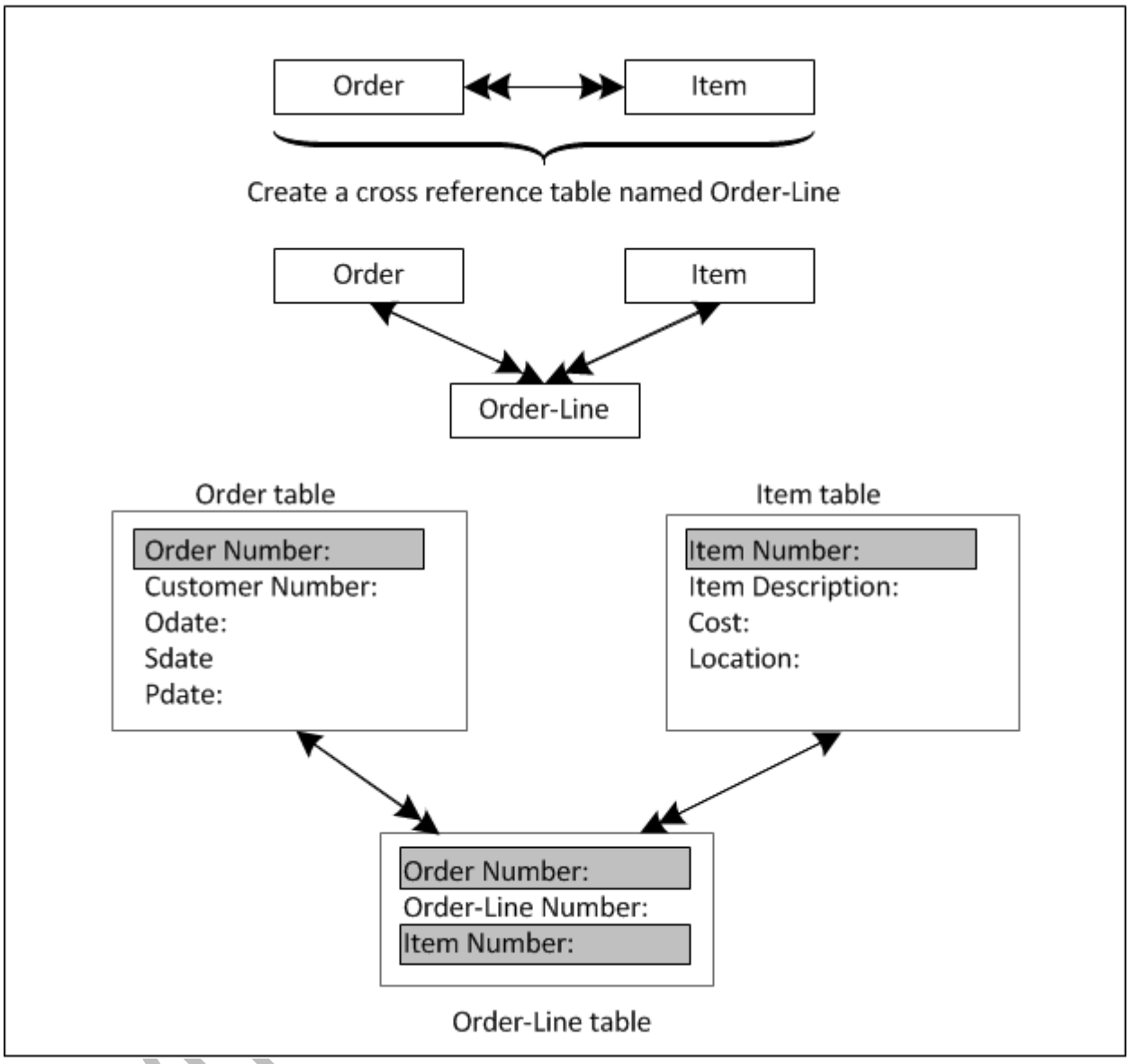
Examples of the many-to-many relationship



Accessing information in tables with a many-to-many relationship is difficult and time consuming. For efficient processing, you can convert the many-to-many relationship tables into two one-to-many relationships by connecting these two tables with a cross-reference table that contains the related columns.

For example, to establish a one-to-many relationship between Order and Item tables, create a cross-reference table Order-Line, as shown in the following figure. The Order-Line table contains both the Order Number and the Item Number. Without this table, you would have to store repetitive information or create multiple columns in both the Order and Item tables.

Using a cross-reference table to relate Order and Item tables

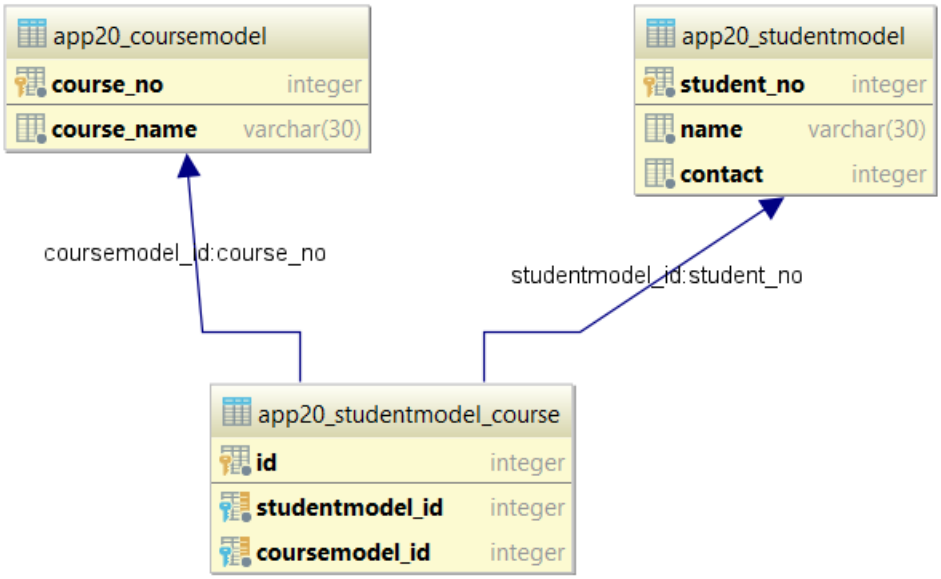


Example Program Using Templates

1) models.py

```
from django.db import models

class CourseModel(models.Model):
    course_no = models.IntegerField(primary_key=True)
    course_name = models.CharField(max_length=30)
```

```
class StudentModel(models.Model):
    student_no = models.IntegerField(primary_key=True)
    name = models.CharField(max_length=30)
    contact = models.IntegerField()
    course = models.ManyToManyField(CourseModel)
```



2) views.py

```
from django.shortcuts import render,redirect
from app20.models import CourseModel,StudentModel
from django.contrib import messages


def showIndex(request):
    return render(request,"index.html")

def add_course(request):
    return render(request,"add_course.html")
```

```
def save_course(request):
    cid = request.POST.get("c1")
    cname = request.POST.get("c2")
    CourseModel(course_no=cid,course_name=cname).save()
    messages.success(request,"Saved")
    return redirect('add_course')

def view_course(request):
    return
render(request,"view_course.html",{"data":CourseModel.objects.all
()})

def add_student(request):
    cs = CourseModel.objects.all()
    return render(request,"add_student.html",{"data":cs})

def save_student(request):
    sno = request.POST.get("s1")
    sname = request.POST.get("s2")
    scno = request.POST.get("s3")
    subject = request.POST.getlist("s4")
    st = StudentModel(student_no=sno,name=sname,contact=scno)
    st.save()
    st.course.set(subject)
    return redirect('main')

def view_students(request):
    return
```

render(request,"**view_students.html**",{"**data**":StudentModel.objects.all()})

3) Templates

1) index.html

```
<html>
   <a href="{% url 'add_course' %}">
      <h3>Add New Course</h3>
   </a>
   <a href="{% url 'view_course' %}">
      <h3>View all Course's</h3>
   </a>

   <a href="{% url 'add_student' %}">
      <h3>Add New Student</h3>
   </a>

   <a href="{% url 'view_students' %}">
      <h3>View all Student's</h3>
   </a>
</html>
```

2) add_course.html

```
<html>
   <body>
      <form action="{% url 'save_course' %}" method="post">
         {% csrf_token %}
      <table align="center" border="2">
         <tr>
            <th>Add Course Details</th>
```

```
        </tr>
        <tr>
          <th><input type="number" placeholder="Course ID"
name="c1" required></th>
        </tr>
        <tr>
          <th><input type="text" placeholder="Name" name="c2"
required></th>
        </tr>
        <tr>
          <th>
            <button type="submit">Save</button>
          </th>
        </tr>

        {% for x in messages %}
          <tr>
            <th>
              <h3> {{ x }}</h3>
            </th>
          </tr>
        {% endfor %}

      </table>
        </form>
    </body>
</html>
```

3) view_course.html

```
<html>
  <body>
    <table align="center" border="2">
      <tr>
        <th width="100">Course No</th>
        <th width="200">Course Name</th>
      </tr>
      {% for x in data %}
        <tr>
        <th width="100">{{ x.course_no }}</th>
        <th width="200">{{ x.course_name }}</th>
      </tr>
      {% endfor %}
    </table>
  </body>
</html>
```

4) add_student.html

```
<html>
  <body>
  <form action="{% url 'save_student' %}" method="post">
    {% csrf_token %}
    <table align="center" border="2">
      <tr>
        <th>Add Student Details</th>
      </tr>
      <tr>
        <th>
          <input type="number" name="s1" placeholder="Student
No" required>
```

```
            </th>
        </tr>
        <tr>
            <th>
                <input type="text" name="s2" placeholder="Student
Name" required>
            </th>
        </tr>
        <tr>
            <th>
                <input type="number" name="s3" placeholder="Contact
No" required>
            </th>
        </tr>
        <tr>
            <th>
            Course :
                <select name="s4" multiple>
                    {% for x in data %}
                        <option value="{{ x.course_no }}">   {{
x.course_name }}   </option>
                    {% endfor %}
                </select>
            </th>
        </tr>
        <tr>
            <th><button type="submit">Save</button> </th>
        </tr>

    </table>
```

```
  </form>
  </body>
</html>
```

5) view_student.html

```
<table align="center" border="2">
  <tr>
    <th>Student IDNO</th>
    <th>Student NAME</th>
    <th>Student CONTACT NO</th>
    <th>Student SUBJECTS</th>
  </tr>

  {% for x in data %}
    <tr>
      <th>{{ x.student_no }}</th>
      <th>{{ x.name }}</th>
      <th>{{ x.contact }}</th>
      <th>
        {% for y in x.course.all %}
          <h3 style="color: blue">{{ y.course_name }}</h3>
        {% endfor %}
      </th>
    </tr>
  {% endfor %}

</table>
```

4) urls.py

```
from django.contrib import admin
from django.urls import path
from app20 import views
urlpatterns = [
    path('admin/', admin.site.urls),
    path('',views.showIndex,name="main"),
    path('add_course/',views.add_course,name="add_course"),
    path('save_course/',views.save_course,name="save_course"),
    path('view_course/',views.view_course,name="view_course"),
    path('add_student/',views.add_student,name="add_student"),
    path('save_student/',views.save_student,name="save_student"),
path('view_students/',views.view_students,name="view_students")
]
```

After Inserting The Records Data in Tables look like

main.app20_coursemodel [db]

	course_no	course_name
1	1	Python
2	2	Django
3	3	Rest API
4	4	Real Time Project
5	5	UI

main.app20_studentmodel [db]

	student_no	name	contact
1	101	Ravi	9876543210
2	102	Kumar	9876543102
3	103	Mohan	9876543210
4	104	Krishna	9876543211
5	105	Naveen	9052492329

main.app20_studentmodel_course [db]

12 rows | Tx: Auto

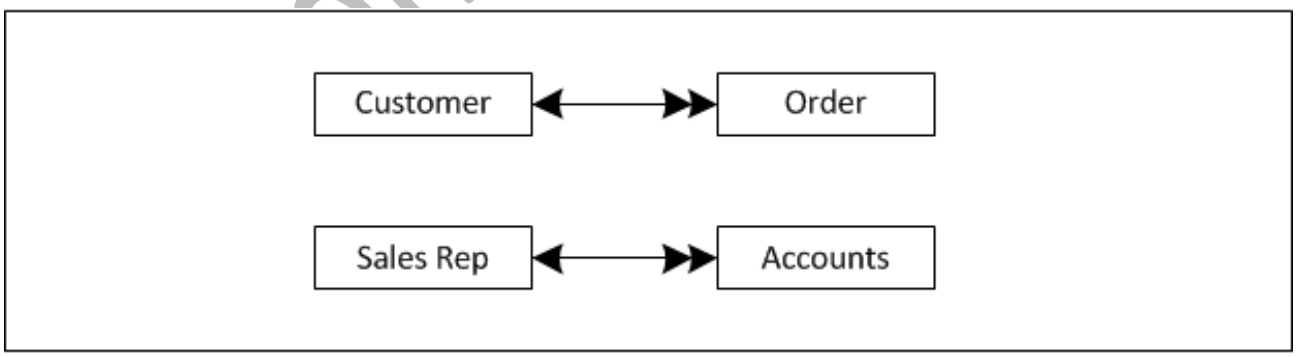
<Filter criteria>

	id	studentmodel_id	coursemodel_id
1	9	101	1
2	10	101	2
3	11	102	1
4	12	102	2
5	13	102	5
6	14	102	3
7	15	102	4
8	16	103	4
9	17	104	5
10	18	104	1
11	19	105	2
12	20	105	3

One-to-many relationship

A *one-to-many* relationship exists when each row in one table has one or many related rows in a second table. The following figure shows examples: one customer can place many orders, or a sales representative can have many customer accounts.

Examples of one-to-many relationships

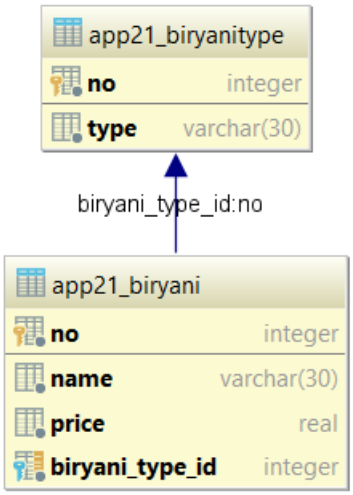


Example Program Using Templates

1) models.py

```
from django.db import models
```

```
class BiryaniType(models.Model):
    no = models.IntegerField(primary_key=True)
    type = models.CharField(max_length=30)

class Biryani(models.Model):
    no = models.IntegerField(primary_key=True)
    name =models.CharField(max_length=30)
    price = models.FloatField()
    biryani_type =
models.ForeignKey(BiryaniType,on_delete=models.CASCADE)
```



2) views.py

```
from django.shortcuts import render,redirect
from app21.models import BiryaniType,Biryani
from django.contrib import messages

def showIndex(request):
    return render(request,"index.html")
```

```
def add_bt(request):
    return render(request,"add_bt.html")

def save_bt(request):
    n = request.POST.get("bt1")
    na = request.POST.get("bt2")
    BiryaniType(no=n,type=na).save()
    messages.success(request,"Biryani Type is Saved")
    return redirect('main')



def view_bt(request):
    return
render(request,"view_bt.html",{"data":BiryaniType.objects.all()})

def add_b(request):
    return
render(request,"add_b.html",{"data":BiryaniType.objects.all()})

def save_b(request):
    a = request.POST.get("bt1")
    b = request.POST.get("bt2")
    c = request.POST.get("bt3")
    d = request.POST.get("bt4")
    Biryani(no=a,name=b,price=c,biryani_type_id=d).save()
    messages.success(request, "Biryani is Saved")
    return redirect('main')

def view_b(request):
```

```
    return render(request, "view_b.html", {"data":
Biryani.objects.all()})

def del_biryani(request):
    no = request.GET.get("no")
    Biryani.objects.filter(no=no).delete()
    return redirect('view_b')

def del_btype(request):
    no = request.GET.get("no")
    BiryaniType.objects.filter(no=no).delete()
    return redirect('view_bt')
```

3) Templates

1) index.html

```
<html>
  <body>
    <a href="{% url 'add_bt' %}">
      <h3>Add a Biryani Type</h3>
    </a>
    <a href="{% url 'view_bt' %}">
      <h3>View all Biryani Type's</h3>
    </a>
     <a href="{% url 'add_b' %}">
       <h3>Add a Biryani</h3>   </a>

     <a href="{% url 'view_b' %}">
       <h3>View all Biryani's</h3>
     </a>
```

```
    <br><br>
    {% for x in messages %}
      <h1 style="font-family: Chiller;size: 30px">{{ x }}</h1>
    {% endfor %}
  </body>
</html>
```

2) add_bt.html

```
<html>
  <body>
    <form action="{% url 'save_bt' %}" method="post">
      {% csrf_token %}
      <table align="center" border="2">
        <tr><th>Biryani Type Details</th></tr>
        <tr><th><input type="number" placeholder="No"
name="bt1"></th></tr>
        <tr><th><input type="text" placeholder="Name"
name="bt2"></th></tr>
        <tr><th><button type="submit">Save</button> </th></tr>
      </table>
    </form>
  </body>
</html>
```

3) view_bt.html

```
{% load static %}
<html>
  <body>
    <table align="center" border="2">
      <tr bgcolor="yellow">
```

```
            <th width="100">No</th>
            <th width="200">Type</th>
            <th width="150">Delete</th>
          </tr>

          {% for x in data %}
            <tr>
              <th width="100">{{ x.no }}</th>
              <th width="200">{{ x.type }}</th>
              <th >
                <a href="{% url 'del_btype' %}?no={{ x.no }}">
                  <img src="{% static 'images/trash.png' %}">
                </a>
              </th>
            </tr>
          {% endfor %}

        </table>
      </body>
</html>
```

4) add_b.html

```
<html>
  <body>
    <form action="{% url 'save_b' %}" method="post">
      {% csrf_token %}
      <table align="center" border="2">
        <tr><th>Biryani Details</th></tr>
        <tr><th><input type="number" placeholder="No"
name="bt1"></th></tr>
```

```
            <tr><th><input type="text" placeholder="Biryani Name"
name="bt2"></th></tr>
            <tr><th><input type="number" placeholder="Price"
name="bt3"></th></tr>
            <tr>
              <th>
                <select name="bt4">
                  {% for x in data %}
                    <option value="{{ x.no }}">  {{ x.type }}  </option>
                  {% endfor %}
                </select>
              </th>
            </tr>
            <tr><th><button type="submit">Save</button> </th></tr>
          </table>
        </form>
    </body>
</html>
```

5) view_b.html

```
{% load static %}
<html>
  <body>
    <table align="center" border="2">
      <tr bgcolor="#adff2f">
        <th width="100">NO</th>
        <th width="200">Name</th>
        <th width="150">Price</th>
        <th width="150">Type</th>
        <th width="150">Delete</th>
```

```
        </tr>

        {% for x in data %}
          <tr>
          <th width="100">{{ x.no }}</th>
          <th width="200">{{ x.name }}</th>
          <th width="150">{{ x.price }}</th>
          <th width="150">{{ x.biryani_type.type }}</th>
          <th width="150">
            <a href="{% url 'del_biryani' %}?no={{ x.no }}">
              <img src="{% static 'images/trash.png' %}">
            </a>
          </th>
        </tr>
        {% endfor %}
      </table>   </body>
</html>
```

4) urls.py

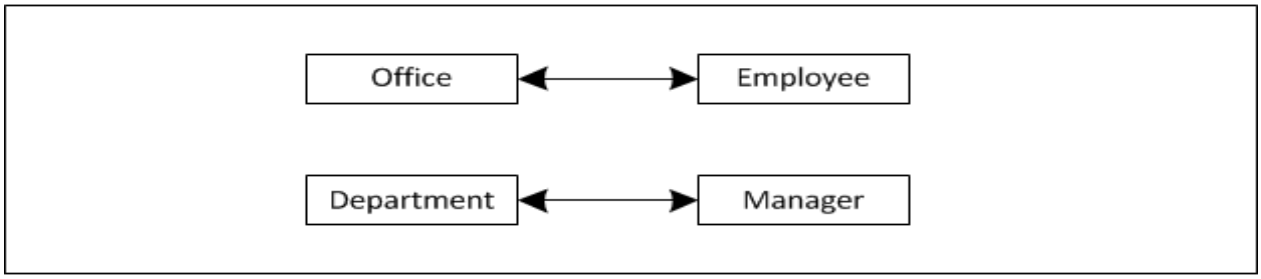
```
from app21 import views

urlpatterns = [
   path('admin/', admin.site.urls),
   path('',views.showIndex,name="main"),
   path('add_bt/',views.add_bt,name="add_bt"),
   path('save_bt/',views.save_bt,name="save_bt"),
   path('view_bt/',views.view_bt,name="view_bt"),
   path('del_btype/',views.del_btype,name="del_btype"),
   path('add_b/',views.add_b,name="add_b"),
   path('save_b/',views.save_b,name="save_b"),
```

```
    path('view_b/',views.view_b,name="view_b"),
    path('del_biryani/',views.del_biryani,name="del_biryani"),
]
```


One-to-one relationship

A *one-to-one* relationship exists when each row in one table has only one related row in a second table. For example, a business might decide to assign one office to exactly one employee. Thus, one employee can have only one office. The same business might also decide that a department can have only one manager. Thus, one manager can manage only one department. The following figure shows these one-to-one relationships.

Examples of one-to-one relationships



Example Program Using Templates

1) models.py

```
from django.db import models

class Employee(models.Model):
    idno = models.IntegerField(primary_key=True)
    name = models.CharField(max_length=30)


class AccountInfo(models.Model):
    acno = models.IntegerField(primary_key=True)
    brach = models.CharField(max_length=30)
    ifsc = models.CharField(max_length=30)
    emp =
models.OneToOneField(Employee,on_delete=models.CASCADE)
```

2) views.py

```
from django.shortcuts import render,redirect
from app22.models import Employee,AccountInfo
from django.contrib import messages

def showIndex(request):
    return render(request,"index.html")


def add_emp(request):
```

```
    return render(request,"add_emp.html")

def save_emp(request):
    eno = request.POST.get("e1")
    ena = request.POST.get("e2")
    Employee(idno=eno,name=ena).save()
    messages.success(request,"Employee Is Saved")
    return redirect('main')

def view_emp(request):
    return
render(request,"view_emp.html",{"data":Employee.objects.all()})

def add_acc(request):
    return
render(request,"add_acc.html",{"data":Employee.objects.all()})

def save_account(request):
    acno = request.POST.get("t1")
    bn = request.POST.get("t2")
    code = request.POST.get("t3")
    eid = request.POST.get("t4")
    AccountInfo(acno=acno,brach=bn,ifsc=code,emp_id=eid).save()
    messages.success(request, "Account Is Saved")
    return redirect('main')

def view_acc(request):
    return
render(request,"view_acc.html",{"data":AccountInfo.objects.all()})
```

3) Templates

1) index.html

```
<html>
  <body>
    <a href="{% url 'add_emp' %}">
      <h1>Add Employee</h1>
    </a>
    <a href="{% url 'view_emp' %}">
      <h1>View All Employee's</h1>
    </a>
     <a href="{% url 'add_acc' %}">
      <h1>Add Account To A Employee</h1>
    </a>
    <a href="{% url 'view_acc' %}">
      <h1>View All Accounts</h1>
    </a>
    {% for x in messages %}
      <h1 style="font-family: Chiller;color: red">
        {{ x }}
      </h1>
    {% endfor %}   </body>
</html>
```

2) add_emp.html

```
<html>
  <body>
  <form action="{% url 'save_emp' %}" method="post">
    {% csrf_token %}
    <input type="number" required name="e1"
```

```
placeholder="Employee No"><br><br>
      <input type="text" required name="e2"
placeholder="Employee Name"><br><br>
      <button type="submit">Save</button>
   </form>
   </body>
</html>
```

3) view_emp.html

```
<html>
   <body>
      <table align="center" border="2">
         <tr>
            <th width="100">IDNO</th>
            <th width="100">NAME</th>
         </tr>

         {% for x in data %}
            <tr>
            <th width="100">{{ x.idno }}</th>
            <th width="100">{{ x.name }}</th>
         </tr>
         {% endfor %}
      </table>  </body></html>
```

4) add_acc.html

```
<form action="{% url 'save_account' %}" method="post">
   {% csrf_token %}
   <input type="number" placeholder="Account No" name="t1"
required> <br><br>
```

```
  <input type="text" placeholder="Branch Name" name="t2"
required> <br><br>
  <input type="text" placeholder="IFSC" name="t3" required>
<br><br>
  <select name="t4">
    {% for x in data %}
      <option>{{ x.idno }}</option>
    {% endfor %}
  </select> <br><br>
  <button type="submit">Save</button>
</form>
```

5) view_acc.html

```
<html>
  <body>
    <table align="center" border="2">
        <tr>
          <th>Account No</th>
          <th>Branch</th>
          <th>IFSC</th>
          <th>Employee ID</th>
        </tr>

        {% for x in data %}
          <tr>
           <th>{{ x.acno }}</th>
           <th>{{ x.brach }}</th>
           <th>{{ x.ifsc }}</th>
           <th>{{ x.emp_id }}</th>
           <th>{{ x.emp.name }}</th>
```

```
            </tr>
        {% endfor %}
    </table>
  </body>
</html>
```

4) urls.py

```
from app22 import views
urlpatterns = [
   path('admin/', admin.site.urls),
   path('',views.showIndex,name="main"),
   path('add_emp/',views.add_emp,name="add_emp"),
   path('save_emp/',views.save_emp,name="save_emp"),
   path('view_emp/',views.view_emp,name="view_emp"),
   path('add_acc/',views.add_acc,name="add_acc"),
   path('save_account/',views.save_account,name="save_account"),
   path('view_acc/',views.view_acc,name="view_acc"),
]
```