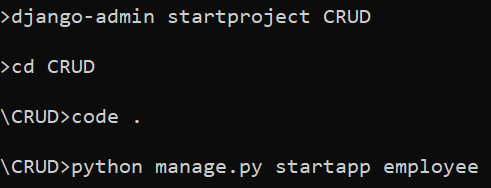
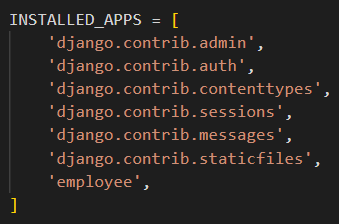
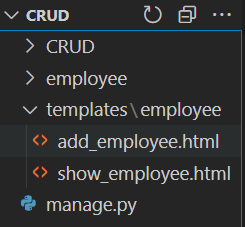
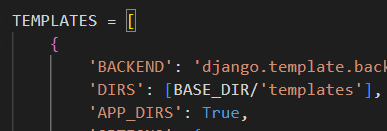
1. Create project and application



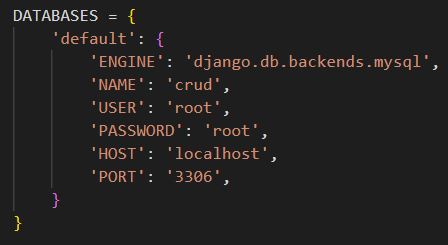
1. Register application in settings.py



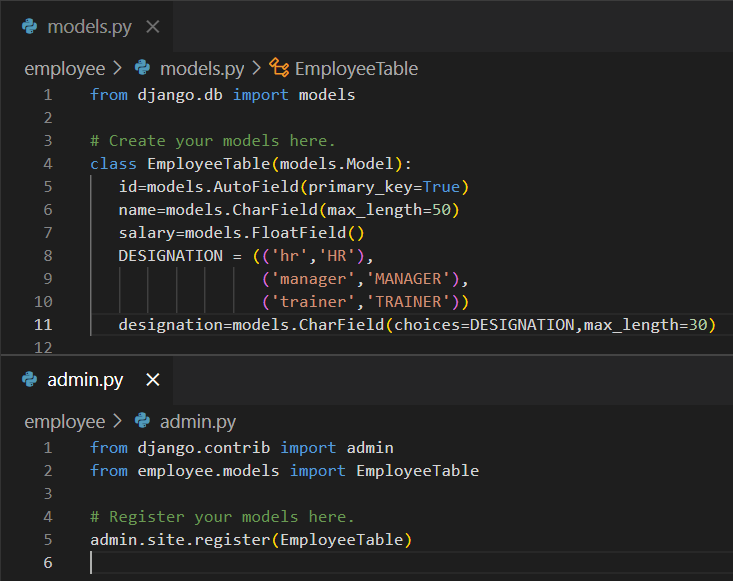
1. Create templates folder and register it in settings.py

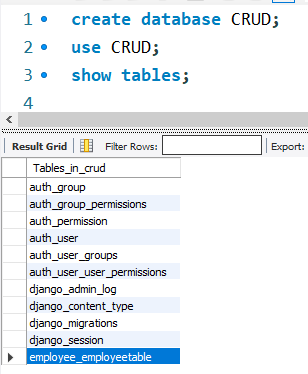
1. Create database “crud” in workbench and register it in settings.py



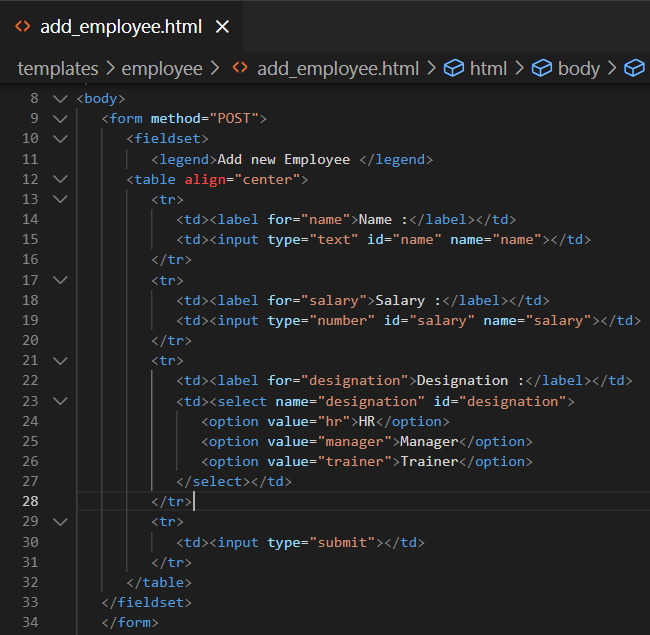
1. Create model EmployeeTable in models.py and register it in admin.py



1. Do makemogrations and migrate and check table in database

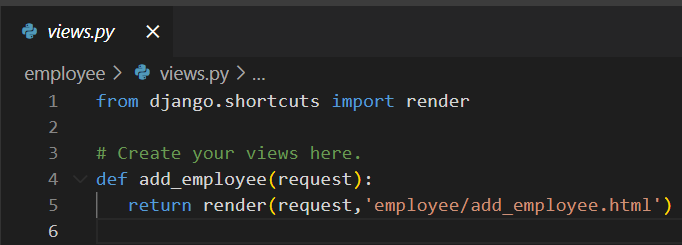


1. Create form and display
   1. Create form in add\_employee.html

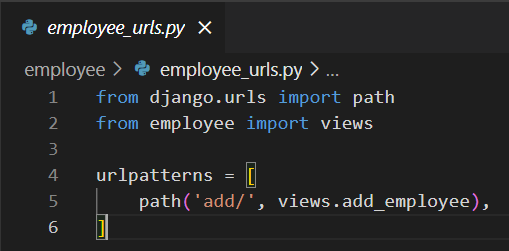


*{% csrf\_token %} add after line 9*

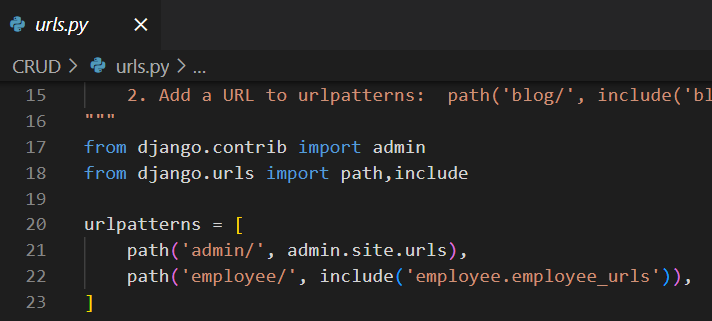
* 1. Create view to show above form



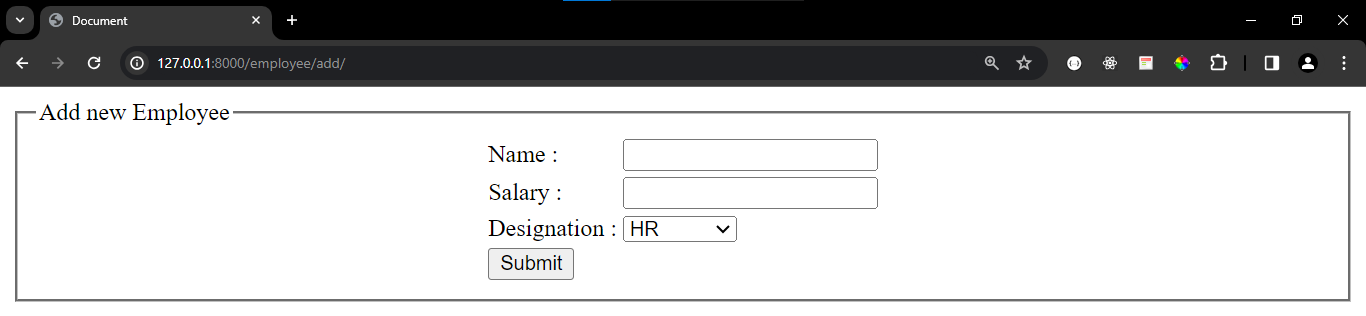
* 1. Create employee\_urls.py file (application level url)



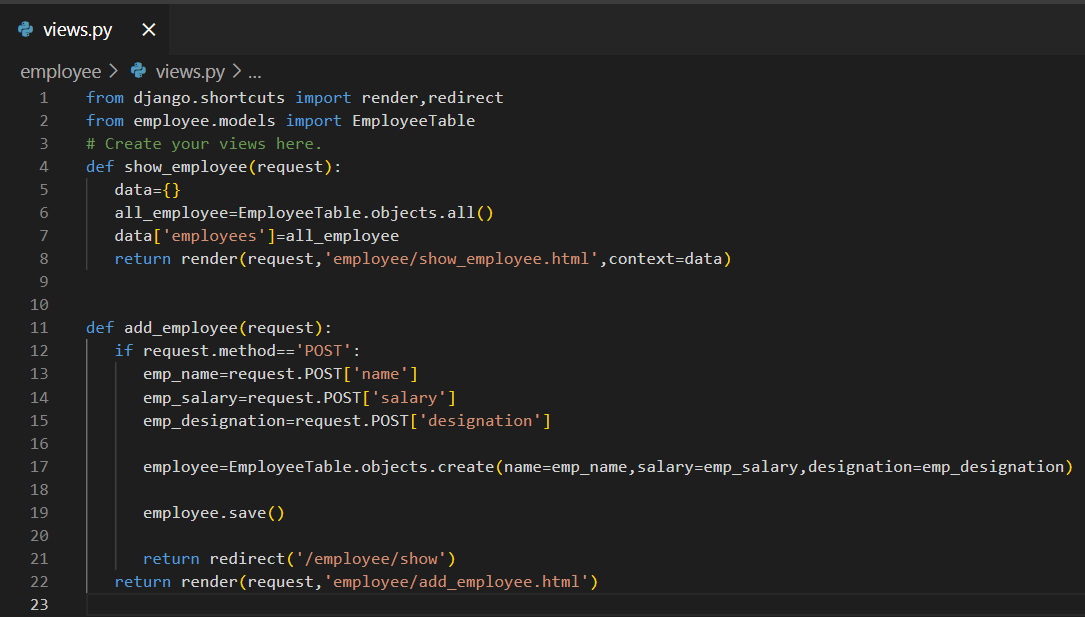
* 1. Register above employee\_urls.py in urls.py



* 1. Output :



1. Collect data from form and add it to the table and after inserting , redirect to show that data
   1. Create fbv(function based view) to insert and get employee data (views.py)



* 1. Code for show\_employee.html to show employees

<!DOCTYPE html>

<html lang="en">

<head>

   <meta charset="UTF-8">

   <meta name="viewport" content="width=device-width, initial-scale=1.0">

   <title>Document</title>

</head>

<body>

   <center>

      <h1>Employee Data</h1>

      <button>Add new Employee</button>

   </center>

   <br>

   <table border="1" align="center", cellpadding="10" cellspacing="0">

      <tr>

         <th>ID</th>

         <th>NAME</th>

         <th>SALARY</th>

         <th>DESIGNATION</th>

         <th colspan="2">ACTION</th>

      </tr>

      {% for employee in employees %}

      <tr>

         <td>{{employee.id}}</td>

         <td>{{employee.name}}</td>

         <td>{{employee.salary}}</td>

         <td>{{employee.designation}}</td>

         <td><button>Update</button></td>

         <td><button>Delete</button></td>

      </tr>

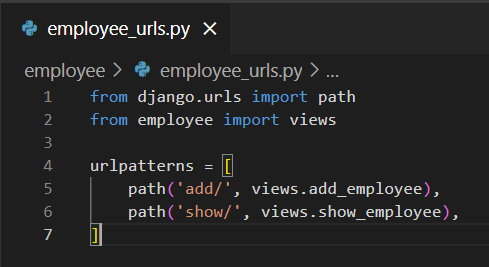
      {% endfor %}

   </table>

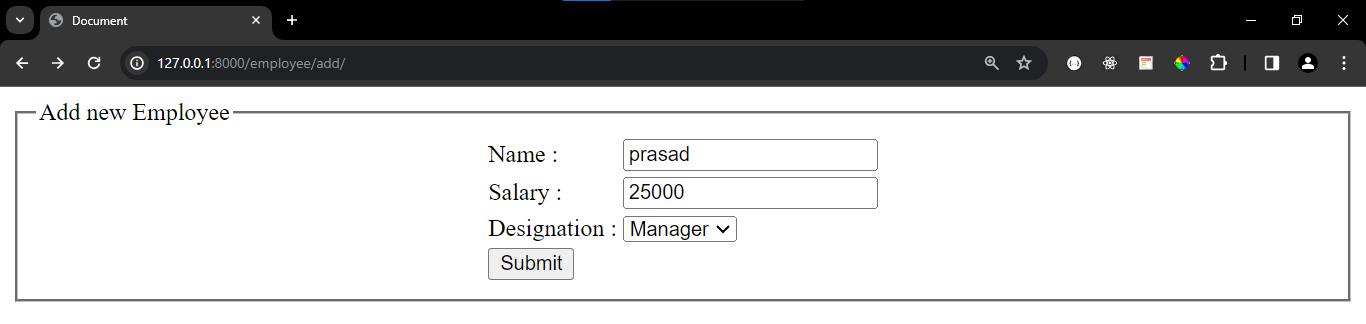
</body>

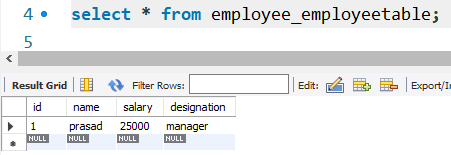
</html>

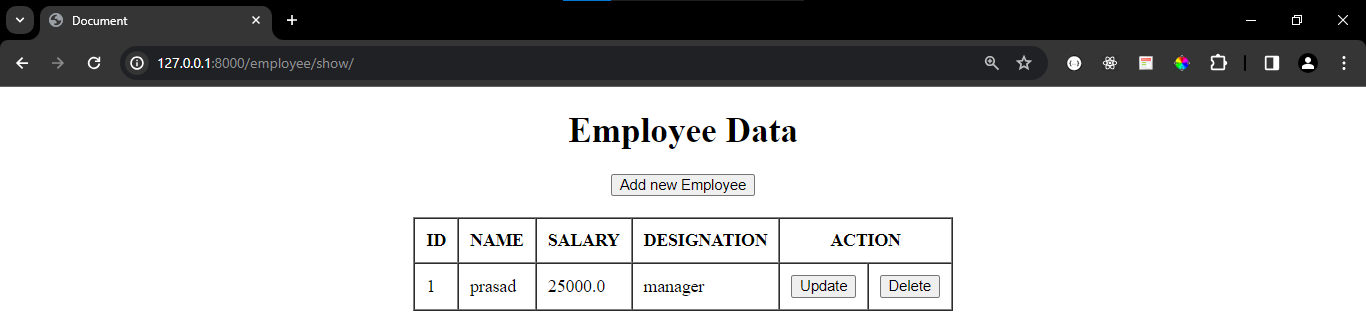
* 1. Create URL in employee\_urls.py



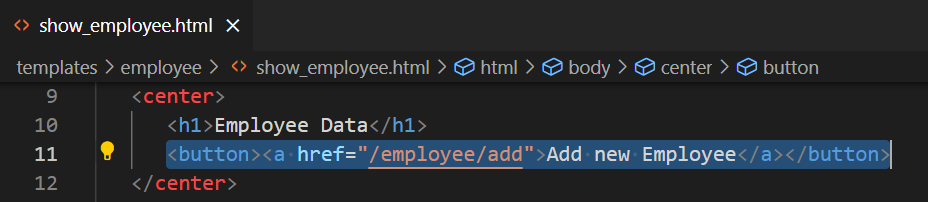
* 1. OUTPUT :



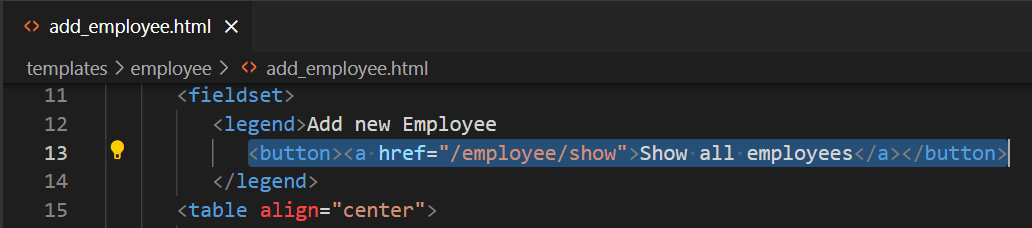




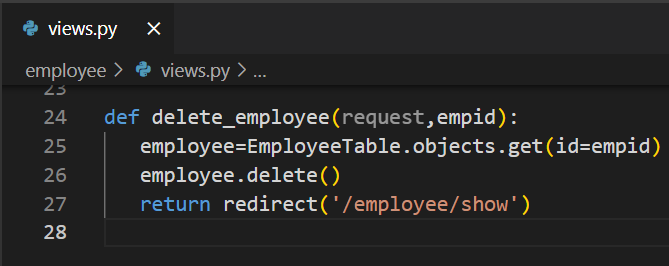
1. Provide link on add new employee button to open add\_employee.html



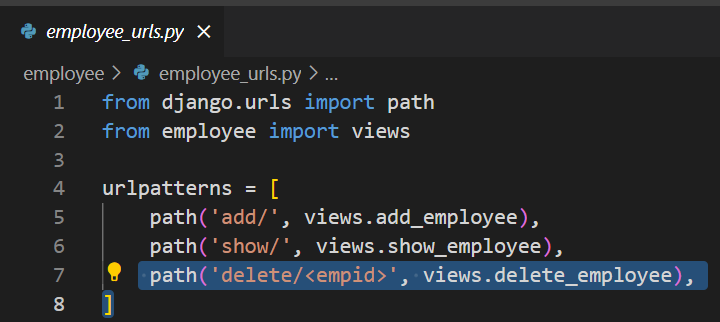
1. Provide button in add\_employee.html to show all employees



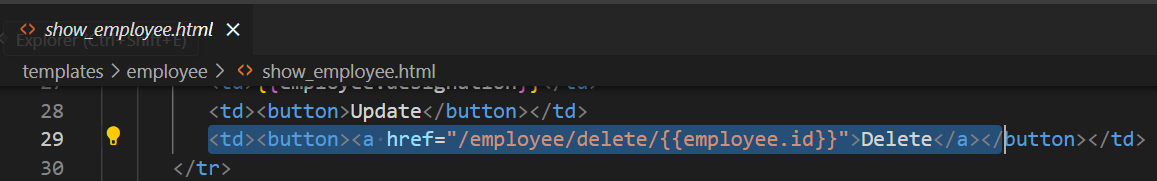
1. Delete functionality
   1. Create fbv and write logic to delete



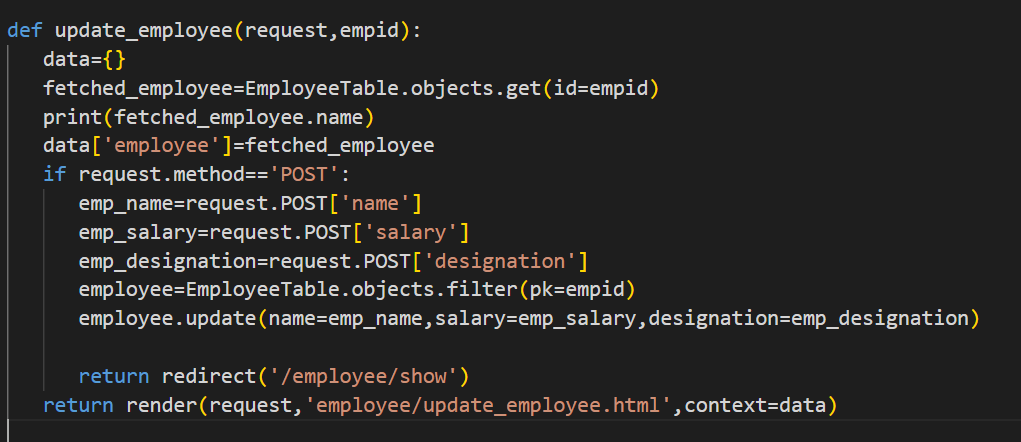
* 1. Create URL for view



* 1. Use above url on delete button in show\_employees.html



1. Update Logic
   1. Create view



* 1. Create update\_employee.html

<!DOCTYPE html>

<html lang="en">

<head>

   <meta charset="UTF-8">

   <meta name="viewport" content="width=device-width, initial-scale=1.0">

   <title>Document</title>

</head>

<body>

   <form method="POST">

      {% csrf\_token %}

      <fieldset>

         <legend>Update Employee</legend>

      <table align="center">

         <tr>

            <td><label for="name">Name :</label></td>

            <td><input type="text" id="name" name="name" value={{employee.name}}></td>

         </tr>

         <tr>

            <td><label for="salary">Salary :</label></td>

            <td><input type="number" id="salary" name="salary" value={{employee.salary}}></td>

         </tr>

         <tr>

            <td><label for="designation">Designation :</label></td>

            <td><select name="designation" id="designation">

               <option value="hr">HR</option>

               <option value="manager">Manager</option>

               <option value="trainer">Trainer</option>

            </select></td>

         </tr>

         <tr>

            <td><input type="submit"></td>

         </tr>

      </table>

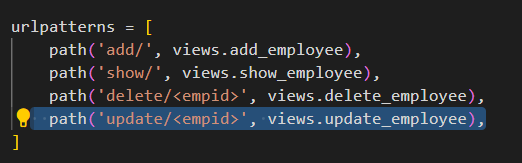
   </fieldset>

   </form>

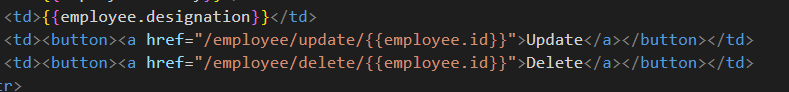
</body>

</html>

* 1. Create application level url



* 1. Use above url on button



1. DONE !!!!!!!!!!!!!!!!!!!!

Done