









Model

from django.db import models  
  
# Create your models here.  
class stud(models.Model):  
 s\_name=models.CharField(max\_length=100)  
 s\_class=models.CharField(max\_length=100)  
 s\_addr = models.CharField(max\_length=100)  
 s\_school = models.CharField(max\_length=100)  
 s\_email = models.EmailField()  
  
  
# student\_enrolment: db

view

from django.shortcuts import render,redirect  
from school.forms import Studform, SForm  
from student\_app.models import stud  
  
# model - stud  
# Create your views here.  
def home(request):  
 return render(request,**'home.html'**)  
  
def register(request):  
 title=**"New Student Registration"** form=Studform(request.POST)  
 if form.is\_valid(): # Checks if the submitted data in form is valid  
 # Process the valid form data (e.g., save to the database)  
 name=form.cleaned\_data[**'s\_name'**] # get the input values to variables  
 cls=form.cleaned\_data[**'s\_class'**]  
 addr=form.cleaned\_data[**'s\_addr'**]  
 school=form.cleaned\_data[**'s\_school'**]  
 mail=form.cleaned\_data[**'s\_email'**]  
  
 # chk student already registered or not with email-id matching  
 chk\_email=stud.objects.filter(s\_email=mail)  
  
 if len(chk\_email)>0: # to check student is in DB  
 return render(request, **'ack.html'**, {**"title"**: **"Student already exists ... try with other email"**})  
 else:  
 data=stud(s\_name=name,s\_class=cls,s\_addr=addr,s\_school=school,s\_email=mail)  
 data.save()  
 return render(request,**'ack.html'**,{**"title"**:**"Registered succesfuly"**})  
# or not-work need to check ---error  
 # try:  
 # form.save()  
 # return render(request,'ack.html',{"title":"Registered succesfuly"})  
 # # return redirect('/register')  
 # except:  
 # pass  
 context={  
 **"title"**:title,  
 **"form"**:form,  
 }  
 return render(request,**'register.html'**,context)  
  
# to show the registured students  
def existing(request):  
 # print('hai fun exist')  
 title=**"Registered Students List..."** queryset=stud.objects.all()  
 # print(queryset)  
 context={**"title"**:title,  
 **"queryset"**:queryset}  
 print(**"hello"**)  
 return render(request,**'existing.html'**,context)  
  
def search(request):  
 # print("fun search")  
 title=**"Search Student"** form=SForm(request.POST or None)  
  
 if form.is\_valid():  
 name=form.cleaned\_data[**'s\_name'**]  
  
#filter data based on name  
 queryset=stud.objects.filter(s\_name=name)  
  
#Check student is in the db or not  
 if len(queryset)==0:  
 return render(request, **'ack.html'**, {**'title'**:**"Student details not found... Please Enter correct name"**})  
 context = {**"title"**: title,  
 **"queryset"**: queryset}  
 return render(request,**'existing.html'**,context)  
 context={  
 **"title"**:title,  
 **'form'**:form,  
 }  
 return render(request,**'search.html'**, context)  
  
# fun to delete the dropout student from DB  
  
def dropout(request):  
 title=**"Drop Out"** form=SForm(request.POST or None)  
  
 if form.is\_valid():  
 name=form.cleaned\_data[**'s\_name'**]  
 queryset=stud.objects.filter(s\_name=name)  
  
# search student is present or not  
 if len(queryset)==0: # if student not in DB then show a msg  
 return render(request, **'ack.html'**, {**'title'**:**"Student details not found... Please Enter correct name"**})  
  
 else: # if student in DB then delete it  
 queryset=stud.objects.filter(s\_name=name).delete()  
 return render(request, **'ack.html'**, {**'title'**: **"Student Removed From Database"**})  
 context={  
 **"title"**:title,  
 **'form'**:form,  
 }  
 return render(request,**'search.html'**, context)

form

from django import forms  
from student\_app.models import stud  
# class Studform(forms.ModelForm):  
 #  
 # class Meta:  
 # model= stud  
 # fields= "\_\_all\_\_"  
# # # fields=["s\_name"]  
  
# OR  
class Studform(forms.Form):  
 s\_name=forms.CharField(max\_length=30,label=**'Student Name'**)  
 s\_class=forms.CharField(max\_length=30,label=**'Class'**)  
 s\_addr = forms.CharField(max\_length=30,label=**"Address"**)  
 s\_school = forms.CharField(max\_length=30,label=**'School Name'**)  
 s\_email = forms.EmailField(label=**'E-mail'**)  
  
class SForm(forms.Form): # for search  
 s\_name = forms.CharField(max\_length=30, label=**" Enter Student Name"**)

urls

from django.contrib import admin  
from django.urls import path  
from student\_app import views  
  
urlpatterns = [  
 path(**'admin/'**, admin.site.urls),  
 path(**''**, views.home), # Set home page as the root URL  
 path(**'home/'**,views.home),  
 path(**'register/'**,views.register),  
 path(**'existing/'**,views.existing),  
 path(**'search/'**,views.search),  
 path(**'dropout/'**, views.dropout),  
]

templates

base.html

<!DOCTYPE html>  
<html>  
<head>  
<style>  
.form-container {  
 width: 50%;  
 margin: 0 auto;  
 padding: 20px;  
 border: 1px solid #ccc;  
 border-radius: 10px;  
 background-color: #f9f9f9;  
 }  
ul {  
 list-style-type: none;  
 margin: 0;  
 padding: 0;  
 overflow: hidden;  
 background-color: #333;  
}  
  
li {  
 float: left;  
}  
  
li a {  
 display: block;  
 color: white;  
 text-align: center;  
 padding: 14px 16px;  
 text-decoration: none;  
}  
  
li a:hover:not(.active) {  
 background-color: #111;  
}  
  
.active {  
 background-color: #04AA6D;  
}  
</style>  
</head>  
<body>  
<CENTER><h1> STUDENT ENROLLMENT SYSTEM</h1></CENTER>  
<ul>  
 <li><a href="/home">Home</a></li>  
 <li><a href="/register">New Registratoin</a></li>  
 <li><a href="/existing">Registered Students</a></li>  
 <li><a href="/search">Search</a></li>  
 <li><a href="/dropout">Drop Out</a></li>  
  
 <li style="float:right"><a class="active" href="#about">About</a></li> <!-- not connected-->  
</ul>  
  
</body>  
</html>

Home

<html >  
<head>  
 {% include "base.html"%}  
 {% load static %}  
 <title>Home Page</title>  
</head>  
<body><br>  
<br>  
 <!-- Display the image -->  
<img src="{% static 'image/AAA.jpg' %}" alt="Home Image" width="1000px" height="300px">  
  
<h4> Student Enrollment Management System Software</h4>  
<p>  
The Student enrollment system is designed to manage all the activities involved during the student enrollment  
 process and combines them into a cloud-based system. The main objective of the enrollment system is to  
 help staff members to enroll students and maintain their records.  
</p>  
</body>  
</html>

Register

<html>  
{% include "base.html" %} <!-- Include your base template correctly -->  
  
<body>  
  
<h1 style="text-align: center;"> {{title}}</h1><!-- Place the heading in the body -->  
<div style="display: flex; justify-content: center; align-items: center; height: 100vh;">  
 <form method='post'>  
 {% csrf\_token%}  
 {{form.as\_p}} <!-- Render the form fields as <p> tags --><!-- form is the dictionary variable given in function register-->  
 <button type="submit">Register </button>  
<h6> Enter the details then click on Register button</h6>  
 </form>  
</div>  
  
</body>  
</html>

Existing

<html >  
<head>  
 {% include "base.html"%}  
 <h1 style="text-align: center;"> {{title}}</h1>  
</head>  
<body>  
  
<div style="display: flex; justify-content: center; align-items: center; height: 100vh;">  
<table border=1>  
 <thead>  
 <tr>  
 <td>Student ID</td>  
 <td>Name</td>  
 <td>class</td>  
 <td>Address</td>  
 <td>School</td>  
 <td> E-mail</td>  
 </tr>  
 </thead>  
 {% for i in queryset %}  
 <tbody>  
 <tr>  
 <td>{{forloop.counter}}</td>  
 <td>{{i.s\_name}}</td>  
 <td>{{i.s\_class}}</td>  
 <td>{{i.s\_addr}}</td>  
 <td>{{i.s\_school}}</td>  
 <td>{{i.s\_email}}</td>  
 </tr>  
 </tbody>  
 {% endfor %}  
</table></div>  
</body>  
</html>

Search

<html>  
{% include "base.html"%} <!-- Include your base template correctly -->  
  
<body>  
<h1 style="text-align: center;"> {{title}}</h1><!-- Place the heading in the body -->  
<div style="display: flex; justify-content: center; align-items: center; height: 100vh;">  
<form method='post'>  
 {% csrf\_token%}  
 {{form.as\_p}} <!-- Render the form fields as <p> tags --><!-- form is the dictionary variable given in function register-->  
 <button type="submit">Search </button>  
  
</form></div>  
</body>  
</html>

Dropout

<html>  
{% include "base.html"%} <!-- Include your base template correctly -->  
  
<body>  
<div style="display: flex; justify-content: center; align-items: center; height: 100vh;">  
<h1 style="text-align: center;"> {{title}}</h1><!-- Place the heading in the body -->  
<form method='post'>  
 {% csrf\_token%}  
 {{form.as\_p}} <!-- Render the form fields as <p> tags --><!-- form is the dictionary variable given in function register-->  
 <button type="submit">Drop out </button>  
  
</form></div>  
</body>  
</html>

Db --- **student\_enrolment**

Settings

DATABASES = {  
**'default'**: {  
**'ENGINE'**: **'django.db.backends.mysql'**,  
**'NAME'**: **'student\_enrolment'**, # +++ db name in workbench  
**'USER'**: **'root'**,  
**'PASSWORD'**: **'3636'**,  
**'HOST'**: **''**,  
**'PORT'**: **''**,  
**'OPTIONS'**: {  
**'init\_command'**: **"SET sql\_mode='STRICT\_TRANS\_TABLES'"**}  
}  
}

TEMPLATES = [  
 {

**'DIRS'**: [**'templates'**],} ]

STATIC\_URL = **'/static/'**STATICFILES\_DIRS = [  
 BASE\_DIR / **"static"**,  
]

Ack.html acknledgement /message

<html >  
<head>  
 {% include "base.html"%}  
 <h1 style="text-align: center;"> {{title}}</h1>  
 <title>Document</title>  
</head>  
<body>  
  
</body>  
</html>  
<!-- to show messages to user-->

Notes

ack.html --> this page created to display acknoledgement / messages to user  
  
 if len(queryset)==0:  
 return render(request, 'ack.html', {'title':"Student details not found... Please Enter correct name"})  
-----------------------------------------------------------------------------------------------------  
context={  
 "title":title,  
 'form':form,  
 }  
 return render(request,'search.html', context)  
  
 -->  
 A dictionary named 'context' is created to hold data that will be passed to the template (search.html).  
 -->  
render(request, 'search.html', context):  
The render function is used to combine a template with a context dictionary and return an HttpResponse object.  
request: The HTTP request object from the view.  
'search.html': This is the name of the template that will be rendered.  
 It should be an HTML file located in one of the directories listed in the TEMPLATES setting in your Django project.  
context: The data passed to the template.  
 In this case, the template will have access to the title and form variables  
 ------------------------------------------