

	<p>Pimpri Chinchwad Education Trust's Pimpri Chinchwad College of Engineering (PCCoE) (An Autonomous Institute) Affiliated to Savitribai Phule Pune University (SPPU) ISO 21001:2018 Certified by TUV SUD</p>
<p align="center">Practical Assignment Submission</p>	

Department: MCA

Academic Year: 2024-25

Semester: II

Year: FYMCA

PRN: 124M1H029

Submission Date: 20/02/2025

Name of Student: Aniket Wakte

CRUD Operations in Django

1. Model Creation (student model)

- Defined a **Student** model in `models.py` with required fields.
- Ran migrations to create the database table.

2. Creating Student Form

- Created `StudentForm` in `forms.py` using Django's `ModelForm`.
- Included necessary fields from the **Student** model.

3. Views for CRUD Operations

- **add_student:** Displays a form to add a new student and saves it to the database.
- **update_student:** Fetches an existing student record, populates the form, and updates it.
- **delete_student:** Delete a student record.
- **display_students:** Display the information of a specific student.

4. Defining URLs

- Mapped URLs to corresponding CRUD views in `urls.py`.

5. Creating Templates

- Designed `add_student.html` and `update_student.html` for **adding and updating** students.
- Created `display_student.html` to **display all students**.

6. Running the Application

- Used `python manage.py runserver` to start the server.
- Accessed CRUD functionalities through browser URLs.

1. Code Snippets

Earn_and_Learn_Assist/settings.py

```
INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'home',
    'accounts',
]
```

Earn_and_Learn_Assist/urls.py

```
from django.contrib import admin
from django.urls import path, include

handler404 = 'home.views.handler404'

urlpatterns = [
    path('admin/', admin.site.urls),
    path('', include('home.urls'), name='home'),
    path('accounts/', include('accounts.urls'), name='accounts'),
]
```

accounts/urls.py

```
from django.urls import path
```

```

from . import views
urlpatterns = [
    path('add/', views.add_student, name='add_student'),
    path('delete/<int:id>/', views.delete_student, name='delete_student'),
    path('update/<int:id>/', views.update_student, name='update_student'),
    path('success/', views.success, name='success'),
    path('display/<int:pk>/', views.display_student,
name='display_student'),
]

```

accounts/views.py

```

from django.shortcuts import render
from django.shortcuts import render, redirect
from .forms import StudentForm
from .models import Student
# Create your views here.

def add_student(request):
    if request.method == 'POST':
        form = StudentForm(request.POST)
        if form.is_valid():
            form.save()
            return redirect('success')
    else:
        form = StudentForm()
    return render(request, 'accounts/add_student.html', {'form': form})

def delete_student(request, id):
    student = Student.objects.get(id=id)
    student.delete()
    return redirect('success')

def update_student(request, id):
    student = Student.objects.get(id=id)
    form = StudentForm(request.POST or None, instance=student)
    if form.is_valid():
        form.save()
        return redirect('success')
    return render(request, 'accounts/update_student.html', {'form': form})

def success(request):
    return render(request, 'accounts/success.html')

```

```
def display_student(request, pk):
    student = Student.objects.get(id=pk)
    return render(request, 'accounts/display_student.html', {'student':
student})
```

accounts/templates/accounts/add_student.html

```
{% extends "base_template/base.html" %}
{% block content %}
    <div class='container'>
        <h1>Add Student</h1>
        <form method='POST'>
            {% csrf_token %}
            {{ form.as_p }}
            <button type='submit' class='btn btn-primary'>Add
Student</button>
        </form>
    </div>
{% endblock %}
```

accounts/templates/accounts/update_student.html

```
{% extends "base_template/base.html" %}
{% block content %}
    <div class='container'>
        <h1>Update Student</h1>
        <form method='POST'>
            {% csrf_token %}
            {{ form.as_p }}
            <button type='submit' class='btn btn-primary'>Update</button>
        </form>
    </div>
{% endblock %}
```

accounts/templates/accounts/display_student.html

```
{% extends "base_template/base.html" %}
```

```

{% block content %}

    <div class='container'>
        <h1>Student Details</h1>
        <table class='table'>
            <tr>
                <th>First Name</th>
                <td>{{ student.first_name }}</td>
            </tr>
            <tr>
                <th>Last Name</th>
                <td>{{ student.last_name }}</td>
            </tr>
            <tr>
                <th>Age</th>
                <td>{{ student.age }}</td>
            </tr>
            <tr>
                <th>Address</th>
                <td>{{ student.address }}</td>
            </tr>
            <tr>
                <th>City</th>
                <td>{{ student.city }}</td>
            </tr>
            <tr>
                <th>State</th>
                <td>{{ student.state }}</td>
            </tr>
            <tr>
                <th>Zip Code</th>
                <td>{{ student.zip_code }}</td>
            </tr>
        </table>
    </div>
{% endblock %}

```

accounts/templates/accounts/success.html

```

{% extends "base_template/base.html" %}

{% block content %}

```

```
<div class='container'>
    <h1>Success</h1>
</div>

{% endblock %}
```

accounts/models.py

```
from django.db import models

# Create your models here.
class Student(models.Model):
    prn_no = models.CharField(max_length=10)
    first_name = models.CharField(max_length=50)
    last_name = models.CharField(max_length=50)
    email = models.EmailField(max_length=50)
    phone_number = models.CharField(max_length=15)
    address = models.CharField(max_length=100)
    city = models.CharField(max_length=50)
    state = models.CharField(max_length=50)
    zip_code = models.CharField(max_length=10)
    date_of_birth = models.DateField
    def __str__(self) -> str:
        return self.first_name + " " + self.last_name
```

accounts/forms.py

```
from django import forms
from .models import Student

class StudentForm(forms.ModelForm):
    class Meta:
        model = Student
        fields = '__all__'
```

2. Screenshots

Home Page Assignment 2 Django CRUD Operations 127.0.0.1:8000/accounts/add/ Select student to change | Django site: +

127.0.0.1:8000/accounts/add/ You are signed in as 1... Django Documentation YouTube Home - Chess.com Django Forms - HonKit WhatsApp Web All Bookmarks

E & L Assist Home About the Scheme Eligibility & Benefits Registration

Add Student

Prn no: 124M1H029

First name: Aniket

Last name: Wakte

Email: aniketwakte42@gmail.com

Phone number: 09021948843

Address: Wadali

City: Nandura

State: Maharashtra

Zip code: 443404

Add Student

DN: 0.0 B
UP: 10.8 B

23:43
20-02-2025

Home Page Assignment 2 127.0.0.1:8000/accounts/update/1/ Select student to change | Django site: +

127.0.0.1:8000/accounts/update/1/ You are signed in as 1... Django Documentation YouTube Home - Chess.com Django Forms - HonKit WhatsApp Web All Bookmarks

E & L Assist Home About the Scheme Eligibility & Benefits Registration

Update Student

Prn no: 124M1H029

First name: Aniket

Last name: Wakte

Email: aniketwakte42@gmail.com

Phone number: 9021948843

Address: Wadali

City: Nandura

State: Maharashtra

Zip code: 443404

Update

DN: 26.9 kB
UP: 33.2 kB

23:44
20-02-2025

The screenshot shows a web browser window with the address bar displaying '127.0.0.1:8000/accounts/display/1/'. The page title is 'E & L Assist'. The navigation bar includes links for Home, About the Scheme, Eligibility & Benefits, and Registration. The main content area is titled 'Select student to change | Django site'.

Student Details

First Name	Aniket
Last Name	Wakte
Age	
Address	Wadali
City	Nandura
State	Maharashtra
Zip Code	443404

DN: 10.5 kB
UP: 1.3 kB

Success

DN: 16.2 B
UP: 10.8 B