**Name:-** Tanishq. Dhokariya

**Technology:-** DJANGO, HTML

**BACKEND**

* **Add Course Entity:**

from django.db import models

class Course(models.Model):

    course\_id = models.AutoField(primary\_key=True)

    course\_name = models.CharField(max\_length=100)

    course\_code = models.CharField(max\_length=20, unique=True)

    course\_duration = models.IntegerField()

    def \_\_str\_\_(self):

        return f"{self.course\_name}"

class Student(models.Model):

    student\_id = models.AutoField(primary\_key=True)

    name = models.CharField(max\_length=100)

    # Add other student fields

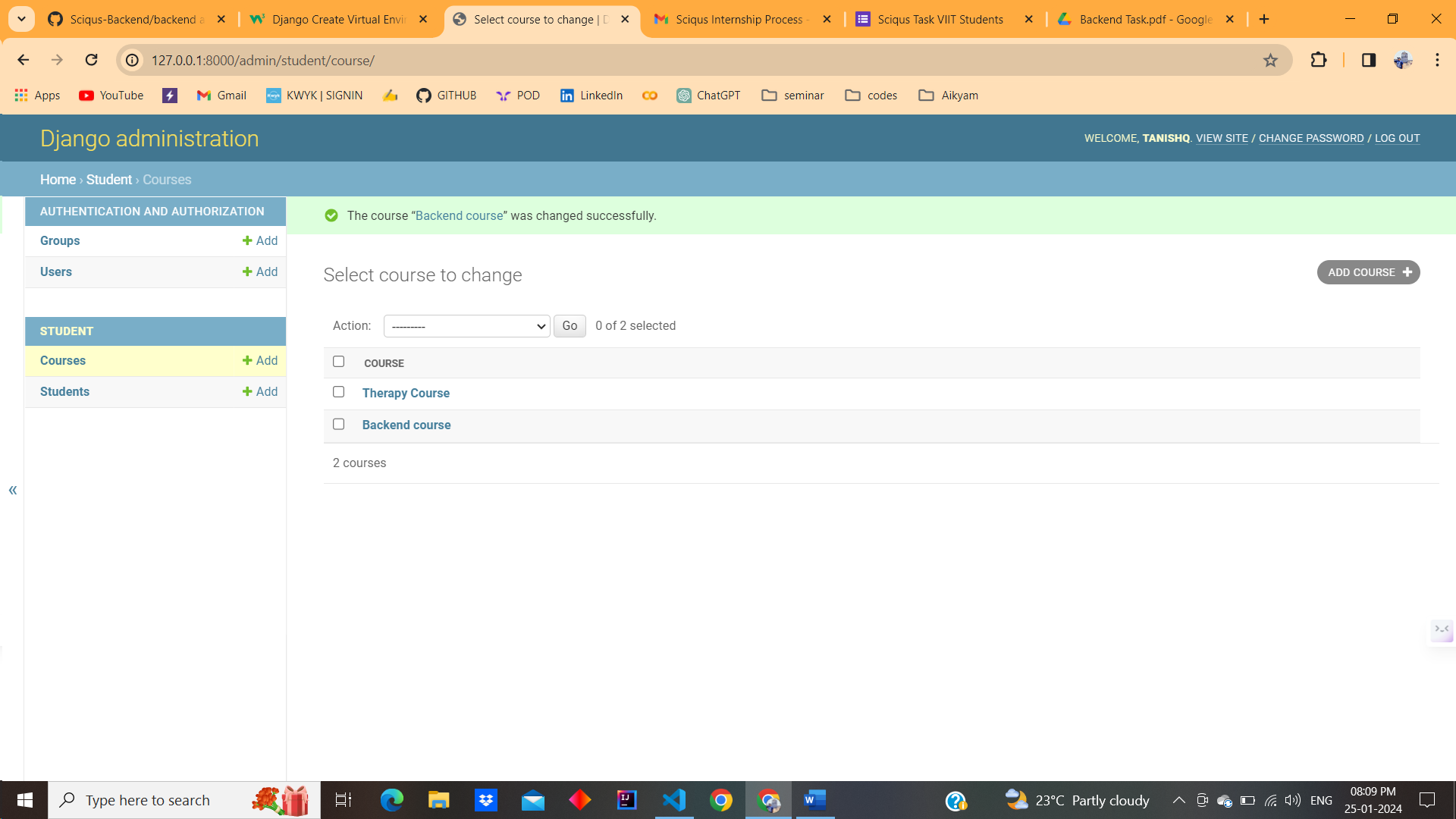
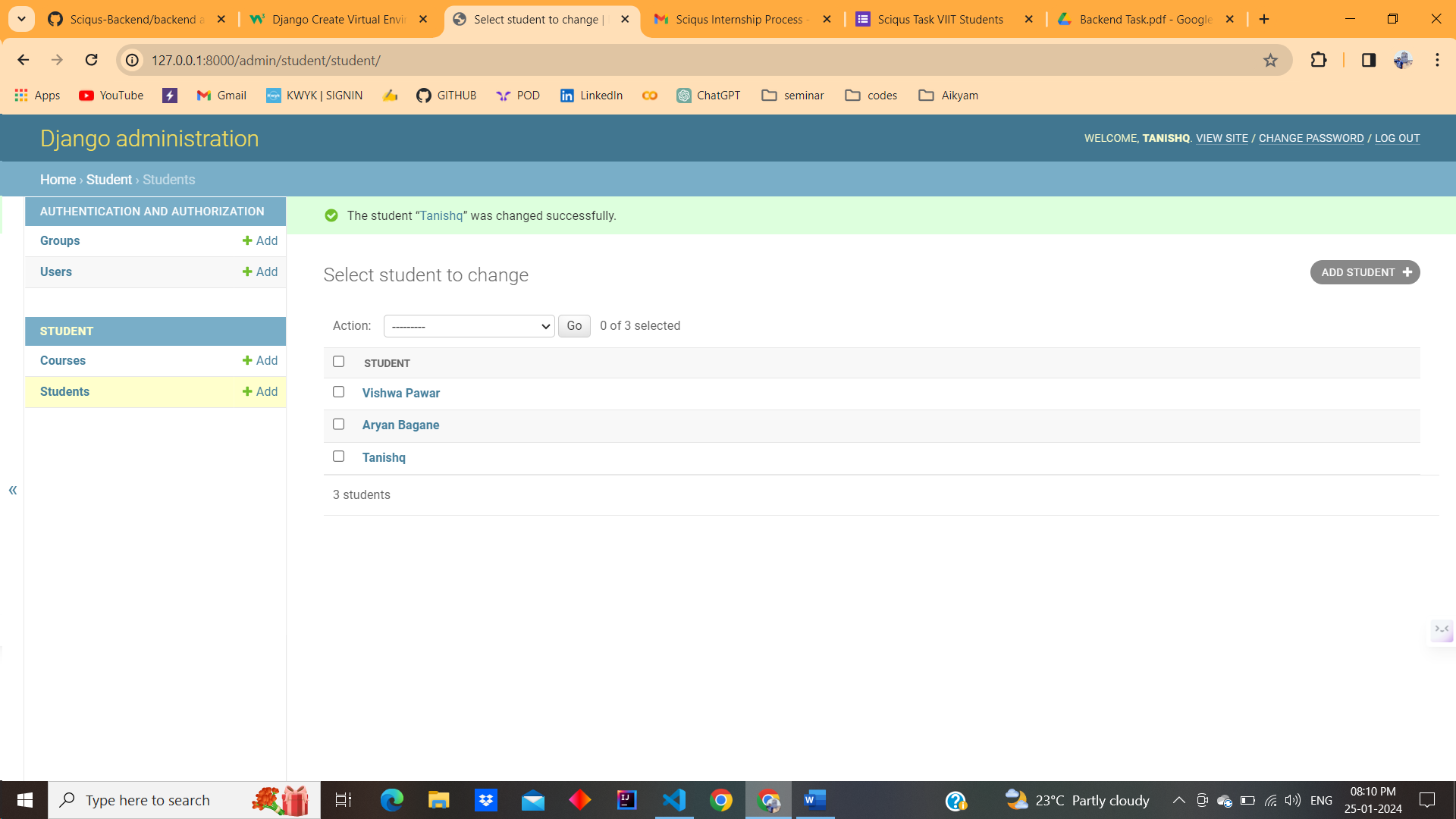
    course = models.ForeignKey(Course, on\_delete=models.CASCADE)

    def \_\_str\_\_(self):

        return f"{self.name}"

[**http://127.0.0.1:8000/admin/**](http://127.0.0.1:8000/admin/)

**Course: Students:**

* **Enhance Student Registration:**

def register\_student(request):

    if request.method == 'POST':

        form = StudentRegistrationForm(request.POST)

        if form.is\_valid():

            form.save()

            return redirect('success\_page')  # Redirect to a success page or any other page

    else:

        form = StudentRegistrationForm()

    return render(request, 'student/registration\_template.html', {'form': form})

**Form**

<form method="post" action="{% url 'register\_student' %}">

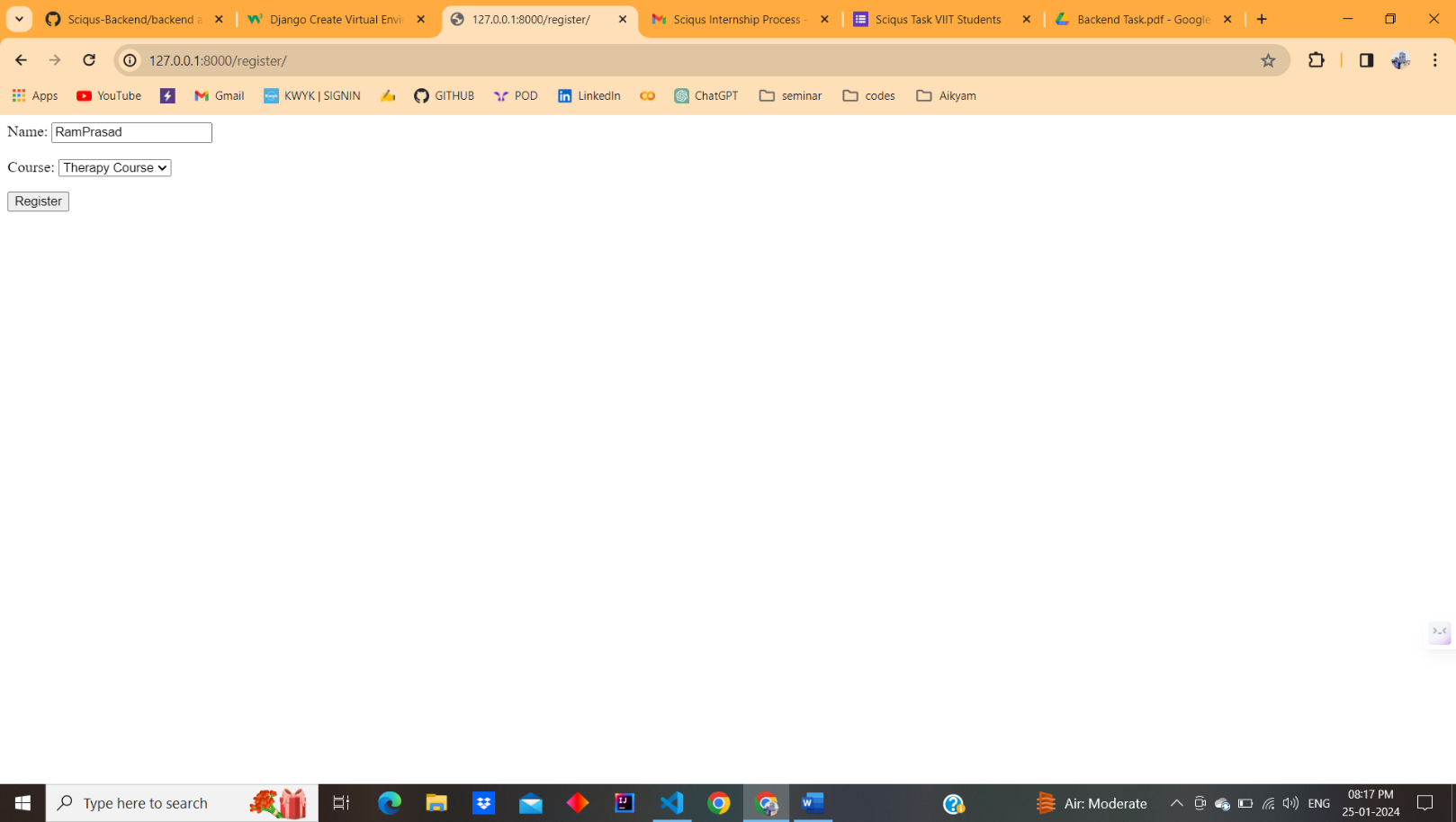
    {% csrf\_token %}

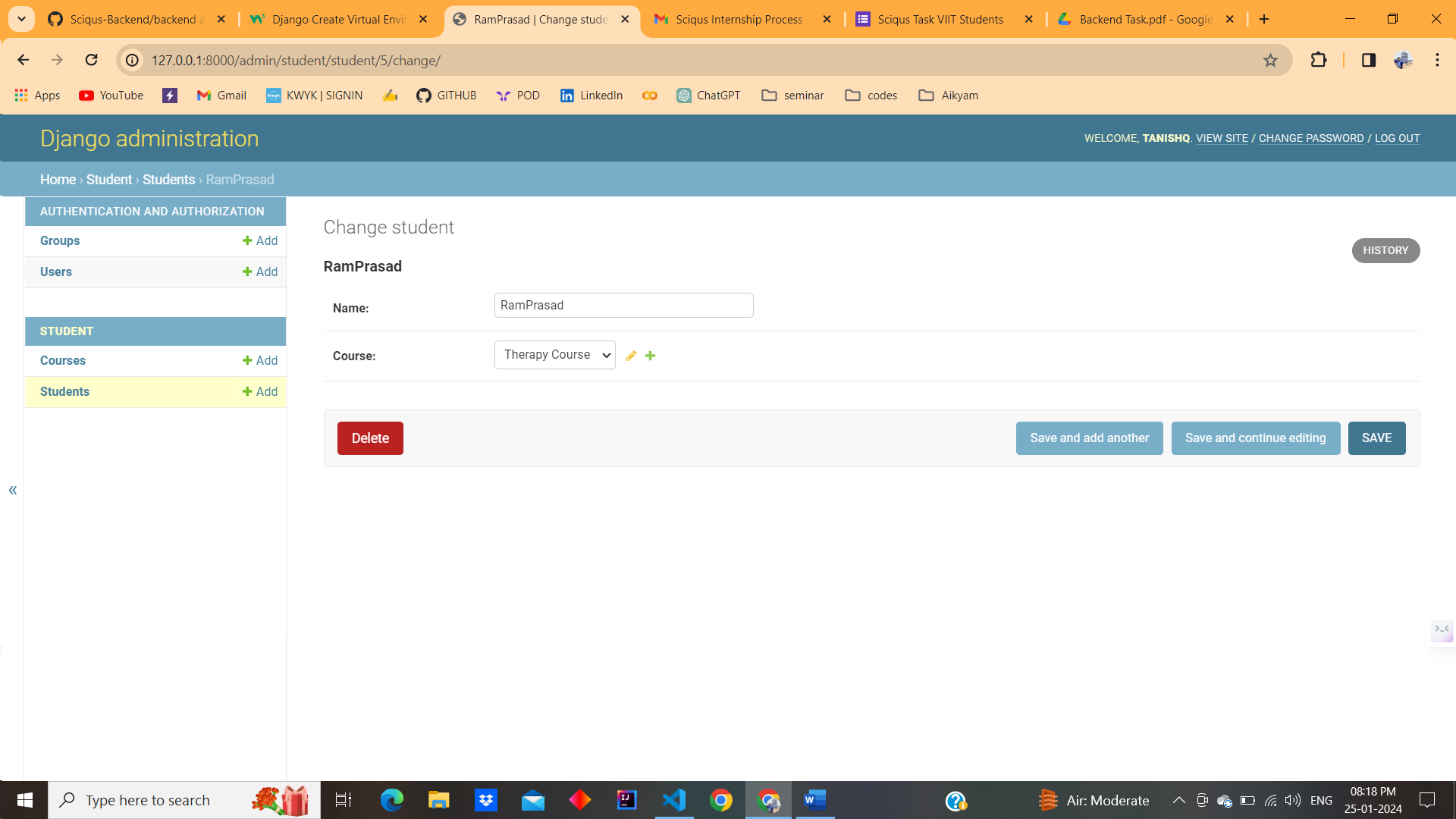
    {{ form.as\_p }}

    <button type="submit">Register</button>

  </form>

[**http://127.0.0.1:8000/register/**](http://127.0.0.1:8000/register/)

****



* **Show Student Details with Course Information:**

from .models import Student

def student\_list(request):

    students = Student.objects.select\_related('course').all()

    # Use select\_related to fetch related course information in a single query

    context = {'students': students}

    return render(request, 'student/student\_list.html', context)

{% for student in students %}

  <div>

    <h3>{{ student.name }}</h3>

    <p>Course: {{ student.course.course\_name }}</p>

    <!-- Edit Button -->

    <a href="{% url 'edit\_student' student.pk %}">Edit</a>

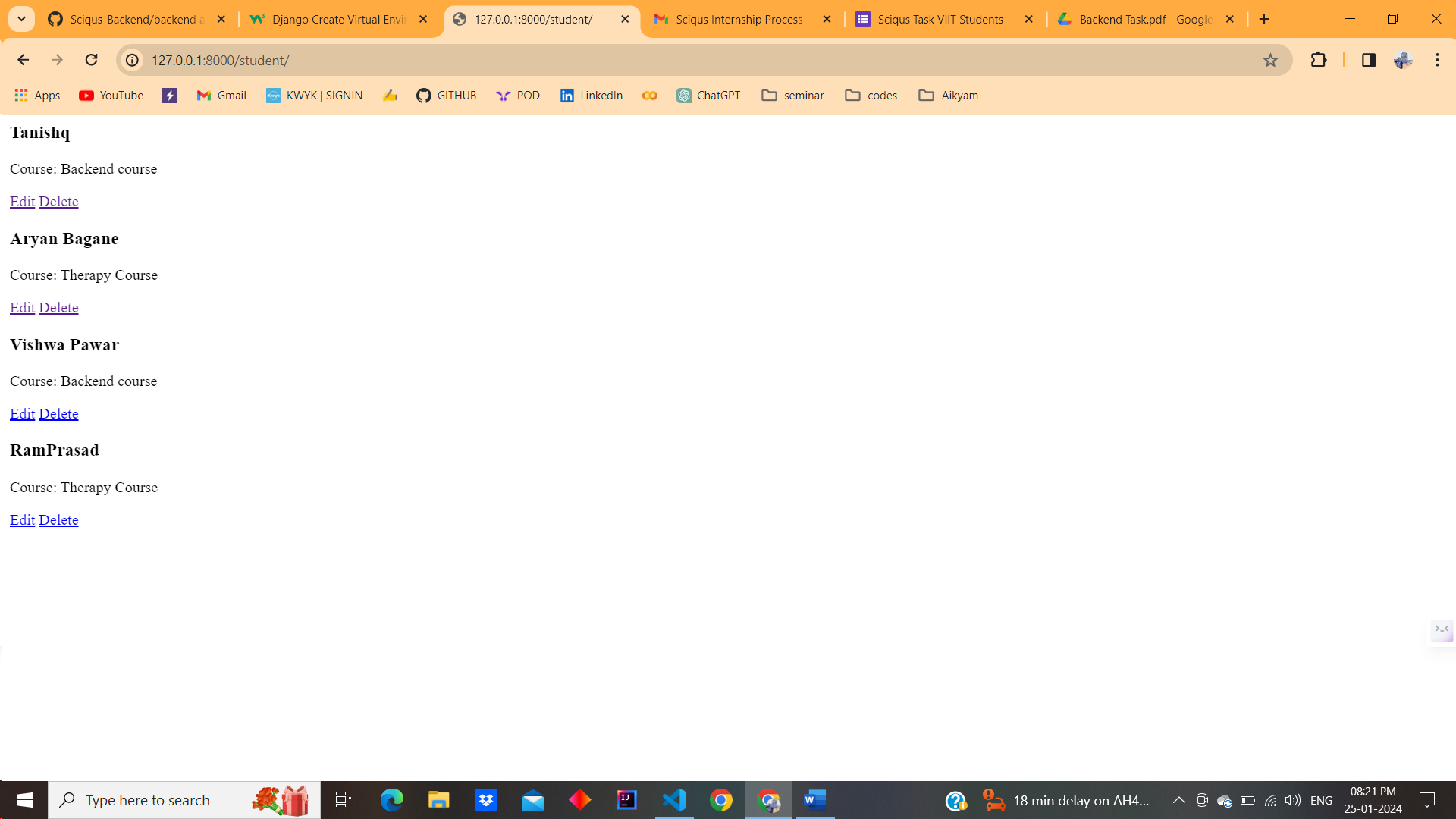
    <!-- Delete Button -->

    <a href="{% url 'delete\_student' student.pk %}">Delete</a>

  </div>

{% endfor %}

[**http://127.0.0.1:8000/student/**](http://127.0.0.1:8000/student/)



* **Show all Students Details Enroll in the Course:**

from .models import Student, Course

def students\_in\_course(request, course\_id):

    course = get\_object\_or\_404(Course, pk=course\_id)

    students = Student.objects.filter(course=course)

    context = {'course': course, 'students': students}

    return render(request, 'student/students\_in\_course.html', context)

<h2>Students enrolled in {{ course.course\_name }}</h2>

{% for student in students %}

  <div>

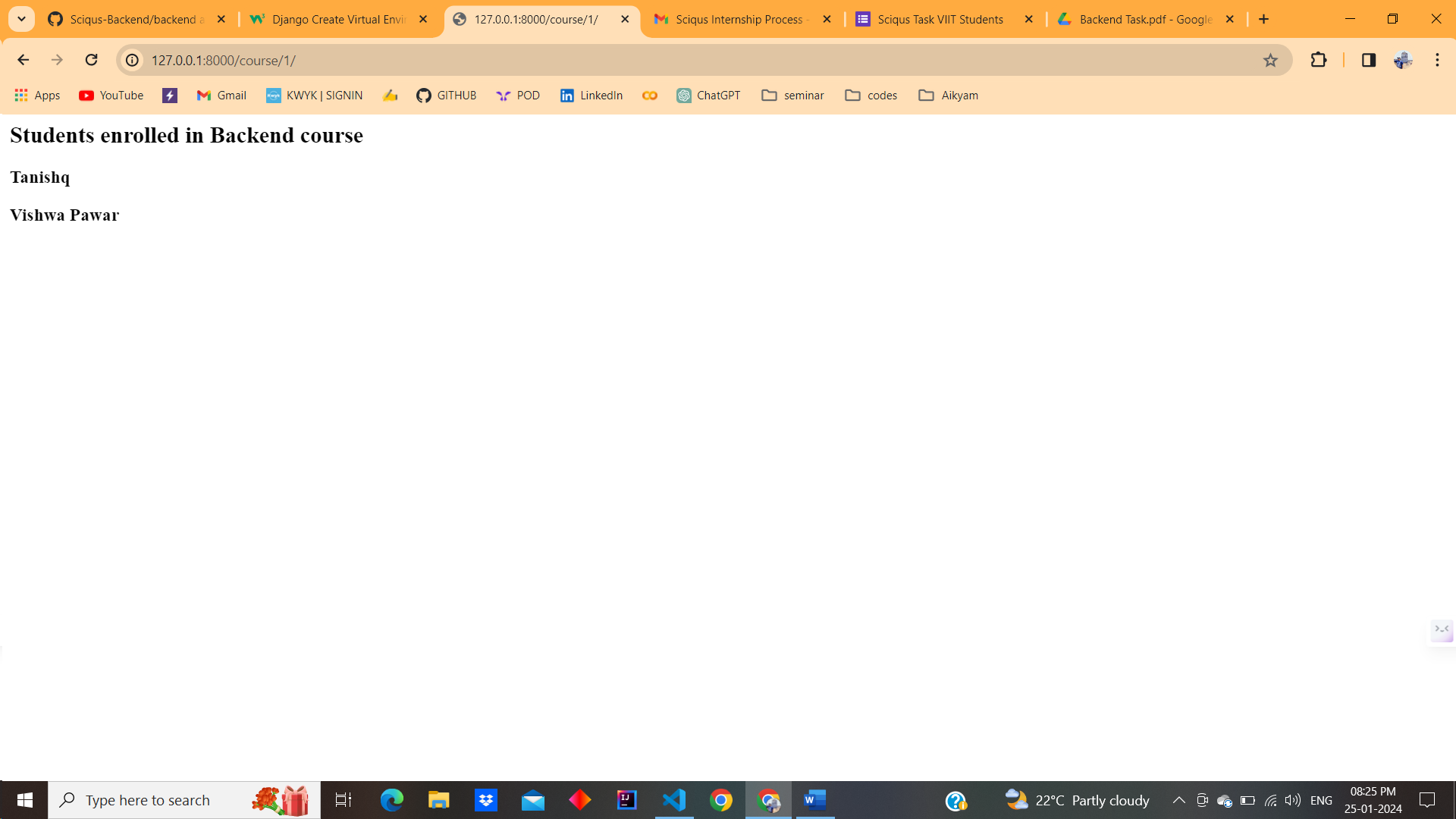
    <h3>{{ student.name }}</h3>

    <!-- Add other student information as needed -->

  </div>

{% endfor %}

[**http://127.0.0.1:8000/course/1/**](http://127.0.0.1:8000/course/1/)



* **Edit Student Details with Course Modification:**

def edit\_student(request, student\_id):

    student = get\_object\_or\_404(Student, pk=student\_id)

    if request.method == 'POST':

        form = StudentUpdateForm(request.POST, instance=student)

        if form.is\_valid():

            form.save()

            return redirect('student\_list')

    else:

        form = StudentUpdateForm(instance=student)

    return render(request, 'student/edit\_student.html', {'form': form, 'student': student})

<form method="post" action="">

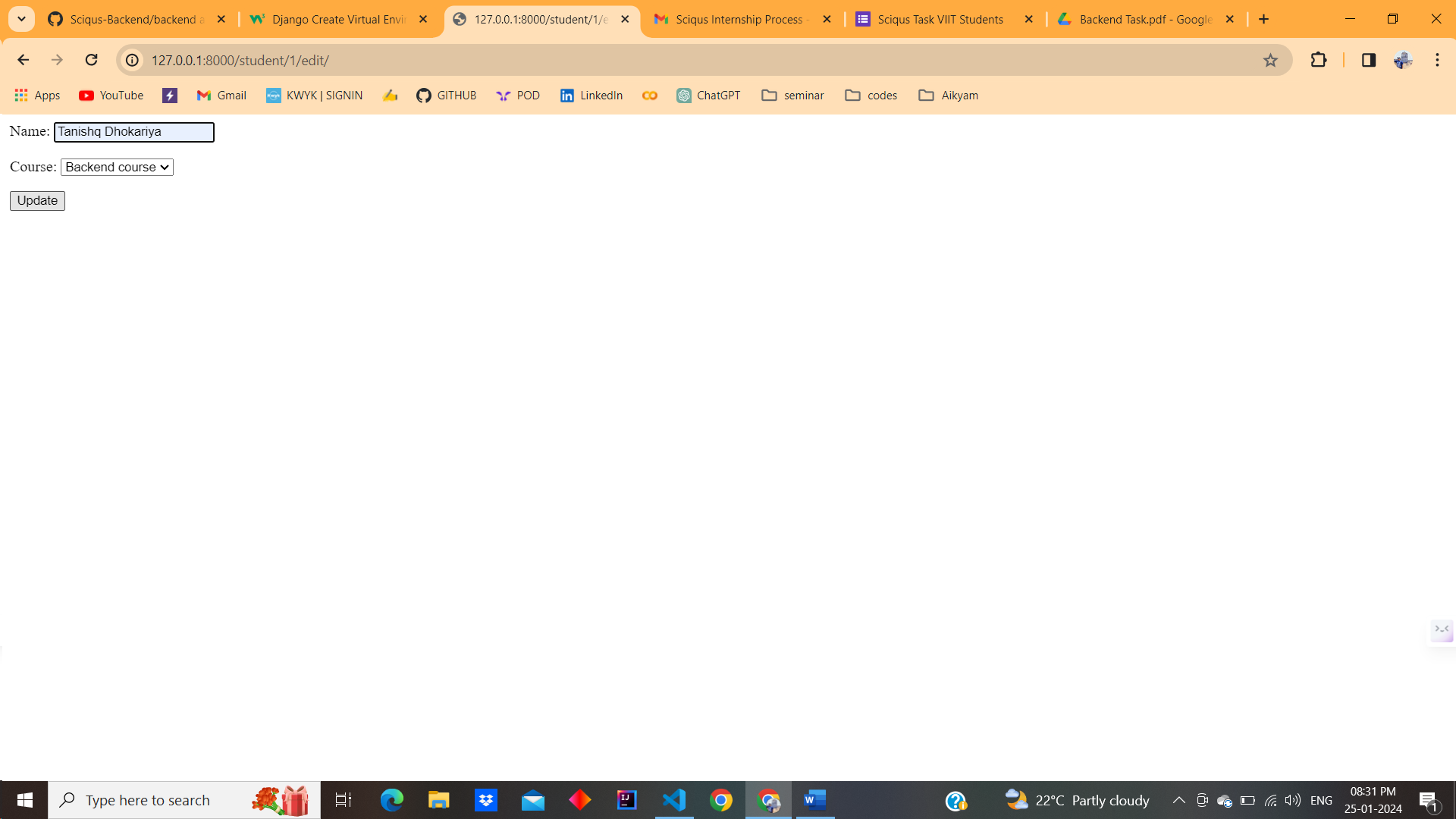
    {% csrf\_token %}

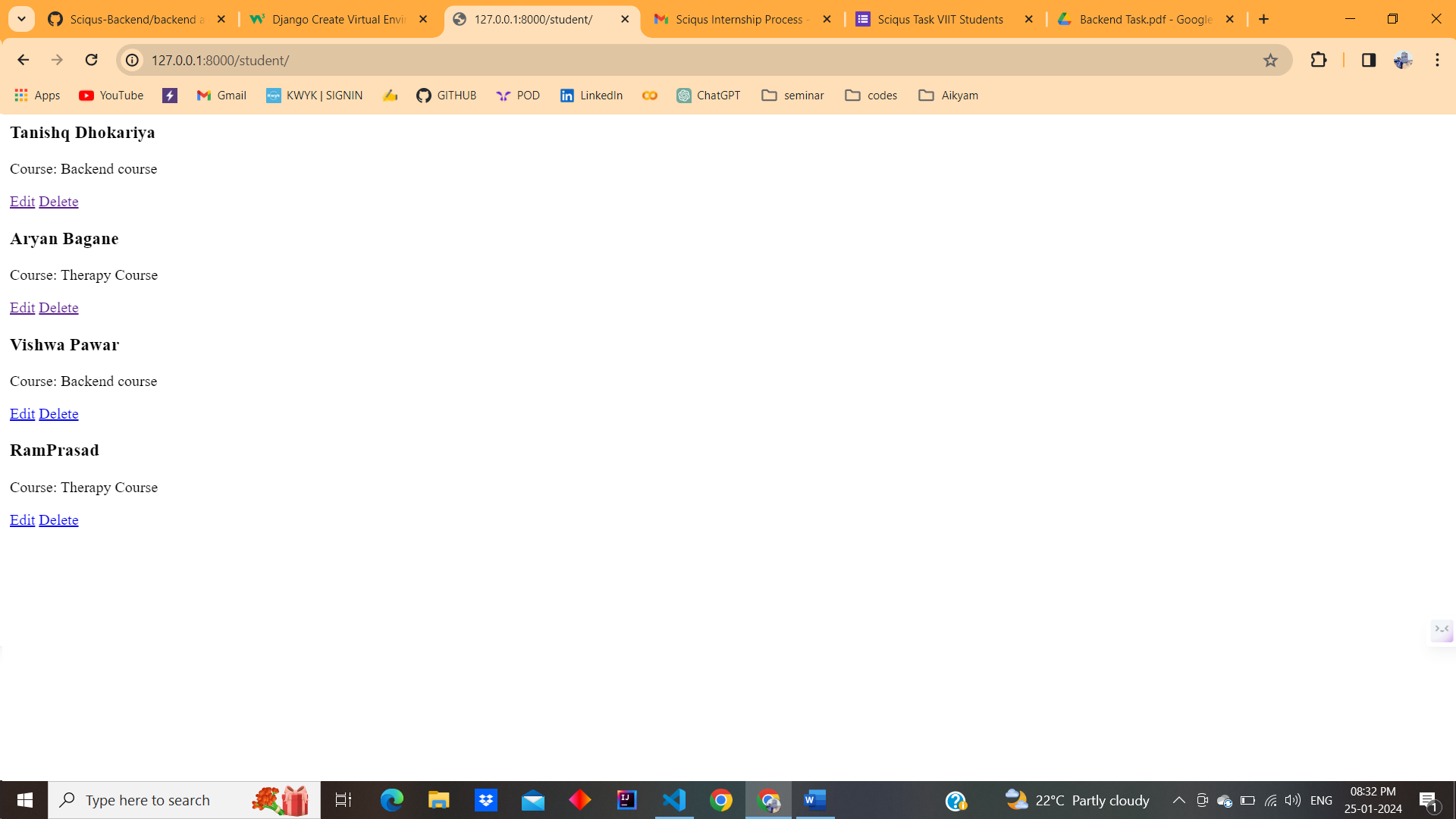
    {{ form.as\_p }}

    <button type="submit">Update</button>

  </form>

[**http://127.0.0.1:8000/student/1/edit/**](http://127.0.0.1:8000/student/1/edit/)





* **Delete Students with Course Deletion:**

def delete\_student(request, student\_id):

    student = get\_object\_or\_404(Student, pk=student\_id)

    if request.method == 'POST':

        # Delete the student

        student.delete()

        # Redirect to the list of students or any other page

        return redirect('student\_list')

    return render(request, 'student/delete\_student.html', {'student': student})

<h2>Are you sure you want to delete {{ student.name }}?</h2>

<form method="post" action="">

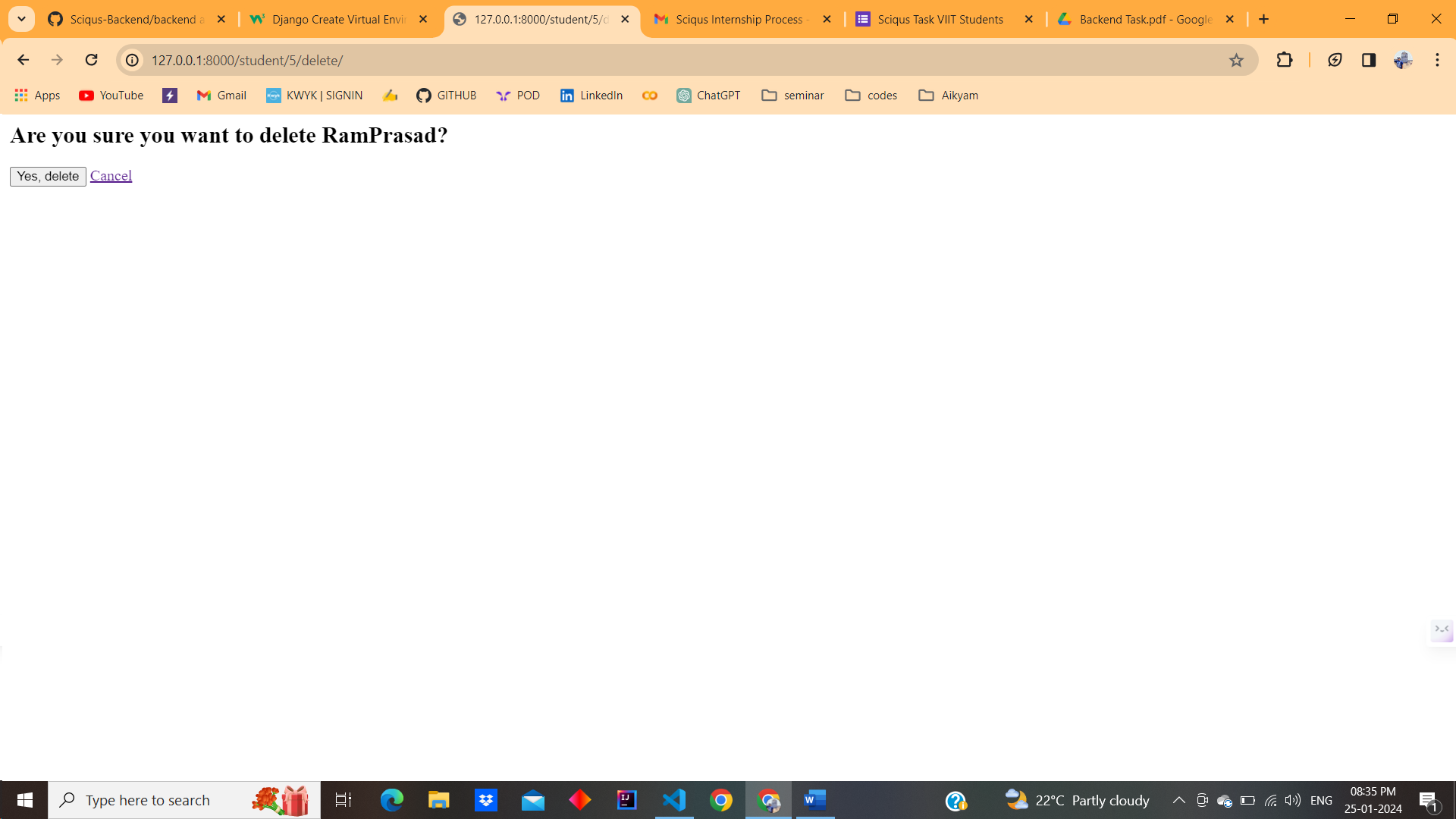
  {% csrf\_token %}

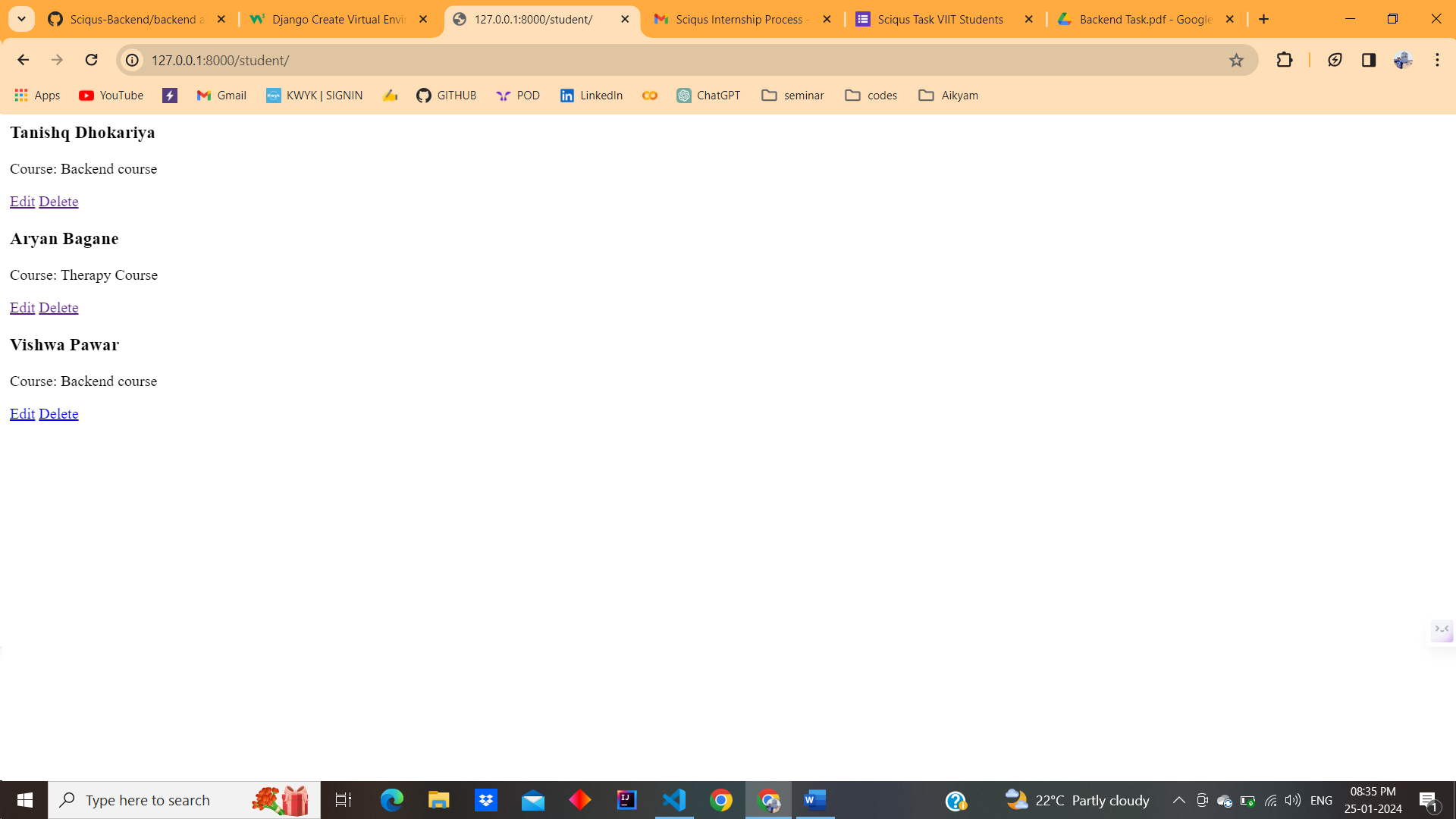
  <button type="submit">Yes, delete</button>

  <a href="{% url 'student\_list' %}">Cancel</a>

</form>

[**http://127.0.0.1:8000/student/5/delete/**](http://127.0.0.1:8000/student/5/delete/)





* **Include Transaction Management:**

@transaction.atomic

def edit\_student(request, student\_id):

    student = get\_object\_or\_404(Student, pk=student\_id)

    if request.method == 'POST':

        form = StudentUpdateForm(request.POST, instance=student)

        if form.is\_valid():

            form.save()

            return redirect('student\_list')

    else:

        form = StudentUpdateForm(instance=student)

    return render(request, 'student/edit\_student.html', {'form': form, 'student': student})

