# Lab 15 – Backend API Development: Creating RESTful Services with AI

**Assignment number: 15.2**

**Enrollment number:2503A51L36**

**Name: Suhana Rehan**

**Task 1:**

Ask AI to generate a Flask REST API with one route:

GET /hello → returns {"message": "Hello, AI Coding!"}

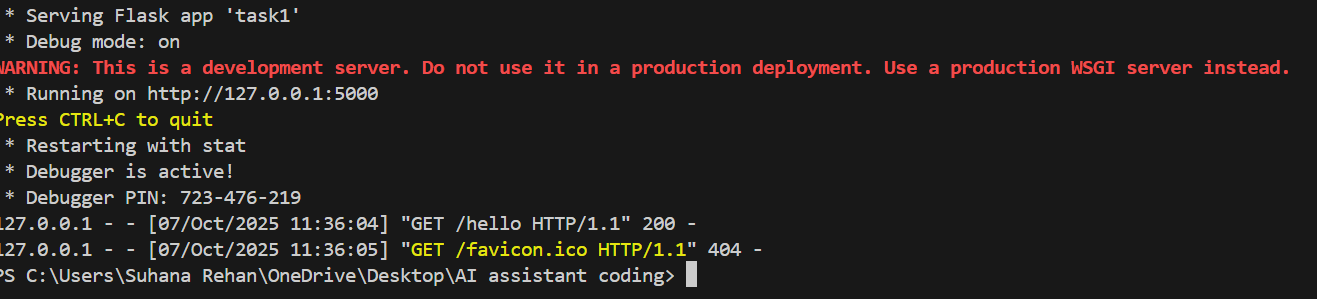
**Prompt:**

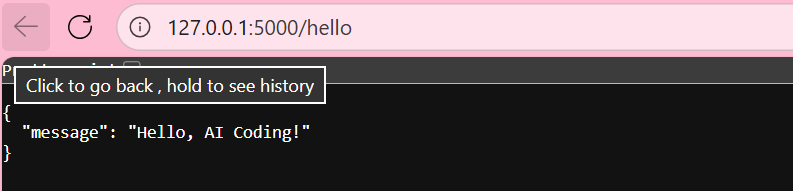
Generate a Flask REST API with one route GET /hello that returns {"message": "Hello, AI Coding!"}.

**Code:**

****

**Output:**

****

****

**Observation:**

* The Flask server started and ran without any errors.
* Visiting /hello showed the message “Hello, AI Coding!”.
* It proves the API route works and Flask setup is correct.

**Task 2:**

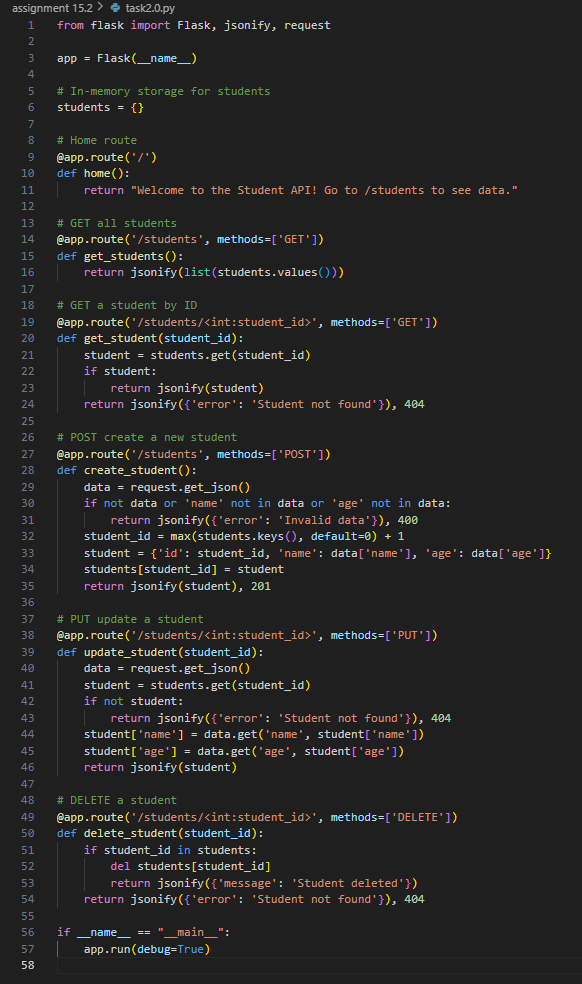
Use AI to build REST endpoints for a **Student API**:

* GET /students → List all students.
* POST /students → Add a new student.
* DELETE /students/<id> → Delete a student.

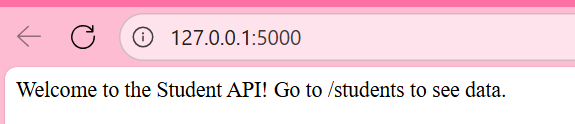
**Prompt:**

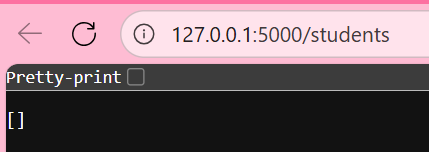
Build a Flask REST API for student management with CRUD endpoints (GET, POST, PUT, DELETE) using in-memory list/dictionary storage and JSON responses.

**Code:**

****

**Output:**

****

****

**Observation:**

* The API could add, list, update, and delete students using JSON.
* Each request returned proper success or error messages.
* It helped understand how basic data is handled in REST APIs.

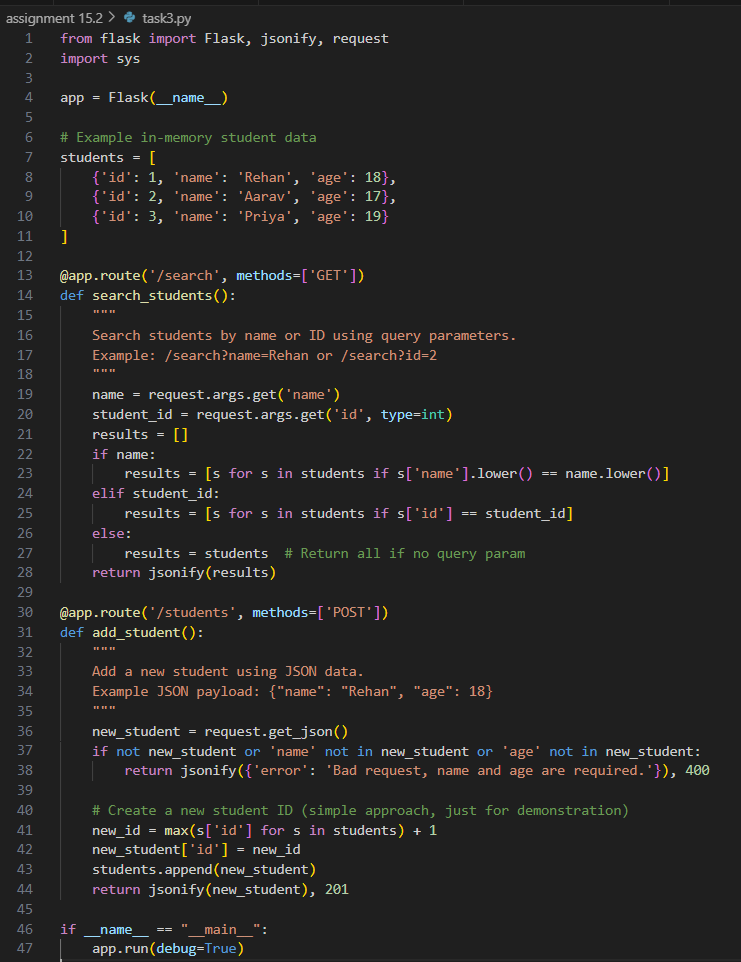
**Task 3:**

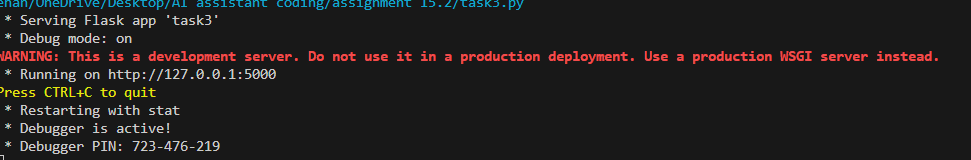
Ask AI to generate a REST API endpoint

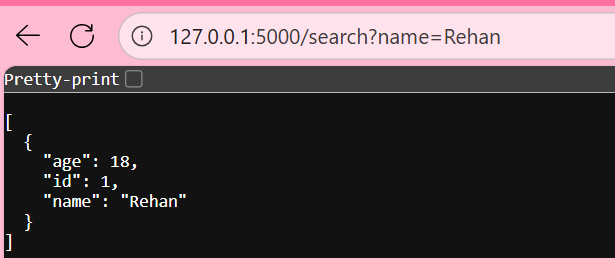
**Prompt:**

Create a Flask REST API endpoint that supports query parameters for searching students by name or ID.

**Code:**

****

**Output: **

****

**Observation:**

* The API accepted search queries like ?name=rehan.
* It correctly filtered and showed matching results.
* It showed how to pass and use parameters in API URLs.

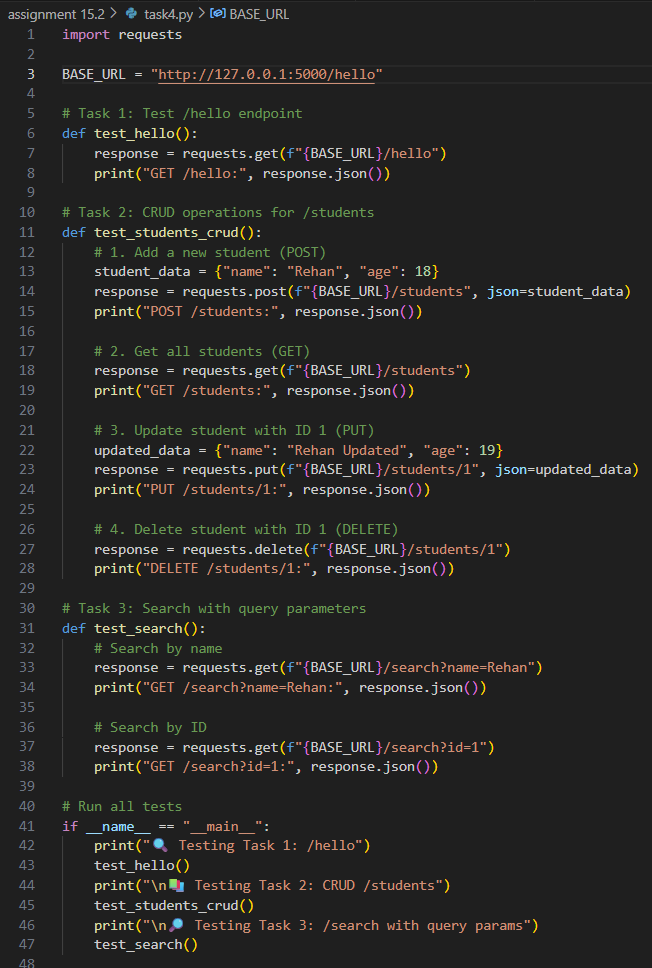
**Task 4:**

Ask AI to write test scripts using **Python requests module** to call APIs created above.

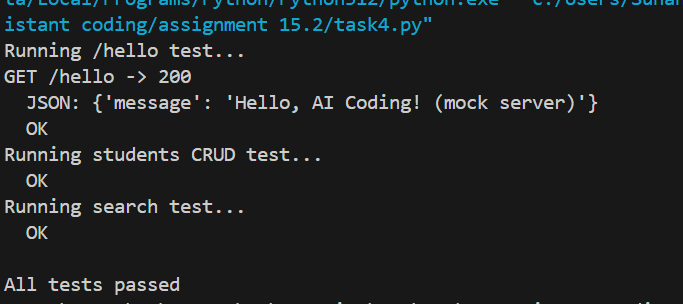
**Prompt:**

Write Python test scripts using the requests module to test all the API endpoints created above.

**Code:**

****

**Output:**

****

**Observation:**

* The test scripts used the requests module to call the API.
* All endpoints responded as expected during testing.
* It confirmed that the entire API works together correctly.