**Requirements:**

Install required packages:

pip install flask mysql-connector-python

Make sure you have:

* MySQL 8 running
* A database named studentdb
* A table students as below:

CREATE DATABASE studentdb;

USE studentdb;

CREATE TABLE students (

id INT AUTO\_INCREMENT PRIMARY KEY,

name VARCHAR(100),

email VARCHAR(100),

age INT

);

**Flask REST API Code (CRUD using cursor):**

from flask import Flask, request, jsonify

import mysql.connector

app = Flask(\_\_name\_\_)

# Database connection

def get\_db\_connection():

return mysql.connector.connect(

host='localhost',

user='root',

password=’root’,

database='studentdb'

)

# CREATE

@app.route('/students', methods=['POST'])

def add\_student():

data = request.json

conn = get\_db\_connection()

cursor = conn.cursor()

cursor.execute("INSERT INTO students (name, email, age) VALUES (%s, %s, %s)",

(data['name'], data['email'], data['age']))

conn.commit()

cursor.close()

conn.close()

return jsonify({'message': 'Student added successfully'}), 201

# READ ALL

@app.route('/students', methods=['GET'])

def get\_students():

conn = get\_db\_connection()

cursor = conn.cursor(dictionary=True)

cursor.execute("SELECT \* FROM students")

students = cursor.fetchall()

cursor.close()

conn.close()

return jsonify(students)

# READ ONE

@app.route('/students/<int:id>', methods=['GET'])

def get\_student(id):

conn = get\_db\_connection()

cursor = conn.cursor(dictionary=True)

cursor.execute("SELECT \* FROM students WHERE id = %s", (id,))

student = cursor.fetchone()

cursor.close()

conn.close()

if student:

return jsonify(student)

else:

return jsonify({'message': 'Student not found'}), 404

# UPDATE

@app.route('/students/<int:id>', methods=['PUT'])

def update\_student(id):

data = request.json

conn = get\_db\_connection()

cursor = conn.cursor()

cursor.execute("UPDATE students SET name=%s, email=%s, age=%s WHERE id=%s",

(data['name'], data['email'], data['age'], id))

conn.commit()

cursor.close()

conn.close()

return jsonify({'message': 'Student updated successfully'})

# DELETE

@app.route('/students/<int:id>', methods=['DELETE'])

def delete\_student(id):

conn = get\_db\_connection()

cursor = conn.cursor()

cursor.execute("DELETE FROM students WHERE id = %s", (id,))

conn.commit()

cursor.close()

conn.close()

return jsonify({'message': 'Student deleted successfully'})

if \_\_name\_\_ == '\_\_main\_\_':

app.run(debug=True)

**💡 Notes:**

* Make sure to replace 'your\_mysql\_password' with your actual MySQL root password.
* Use **Postman** or **curl** to test endpoints like /students.