**Multiplication API with Validation**

**Objective:**

Create a FastAPI application that exposes a POST endpoint to multiply two numbers and return the result. Ensure that the input is validated and both a and b should be integers.

**Instructions:**

1. **Create a FastAPI application**:
   * Define a POST endpoint at /multiply to accept two numbers, a and b.
   * Validate that both a and b are integers using **Pydantic models**.
2. **Test the API** using:
   * **Postman** (send JSON body with a and b).
   * **curl** to test the endpoint with JSON data.
   * **Python requests** for programmatic testing.
3. **Bonus**: Return an error message if the input values are not integers.

**Simple Calculator API**

**Objective:**

Create a FastAPI application that acts as a simple calculator and exposes endpoints to handle addition, subtraction, multiplication, and division.

**Instructions:**

1. **Create a FastAPI application**:
   * Expose a POST method at /calculator to handle four operations: addition, subtraction, multiplication, and division.
   * Accept two numbers and an operation type (add, subtract, multiply, divide) as JSON input.
   * Return the result of the operation.
2. **Test the API** using:
   * **Postman** to test the calculator API.
   * **curl** to test each operation (addition, subtraction, multiplication, and division).
   * **Python requests** to send requests and receive responses.
3. **Bonus**: Handle division by zero gracefully and return a custom error message.

**User Registration API**

**Objective:**

Create a FastAPI application that exposes a POST endpoint to register a user. The API should accept a **JSON body** with a username, email, and password. The application should validate the input, ensuring the email is in the correct format and the password has at least 8 characters.

**Instructions:**

1. **Create a FastAPI application**:
   * Define a POST method at /register to accept the user's username, email, and password in JSON format.
   * Use **Pydantic validation** to ensure the email is a valid email format and the password has a minimum length of 8 characters.
   * Return a success message if registration is successful.
2. **Test the API** using:
   * **Postman** to register a new user.
   * **curl** to test the registration endpoint with valid and invalid data.
   * **Python requests** to test user registration programmatically.
3. **Bonus**: Store the registered users in memory (for the sake of the assignment) and return the list of registered users when requested.