So to give us a platform to somewhat build off of I went ahead and put together something for us to build off of. I have tried to include directions that make sense. I didnt go into setting up mySQL, there are tutorials online. I use DBeaver, its pretty straightforward. I tried to follow our statement in the docs where sql for database, python for back end, html/CSS for front. This shows basic integration so we can expand from here.

Setting up test environment:

Using a database management software; set up mySQL
Once connected to your localhost, use this script to make the database and fill it with some basic values:

```
-- Step 1: Create Database
CREATE DATABASE soft_project;
USE soft_project;

-- Step 2: Create Users Table (Student Registry)
CREATE TABLE student_information (
  id INT AUTO_INCREMENT PRIMARY KEY,
  name VARCHAR(255) NOT NULL,
  email VARCHAR(255) UNIQUE NOT NULL,
  password_hash VARCHAR(255) NOT NULL, -- Store hashed passwords
  created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP
);
-- Step 3: Create Subjects Table (List of Available Subjects)
CREATE TABLE available_subjects (
  id INT AUTO_INCREMENT PRIMARY KEY,
  subject_name VARCHAR(255) UNIQUE NOT NULL
);
-- Step 4: Create Student-Subjects Table (Many-to-Many Relationship)
CREATE TABLE student_subjects (
  student_id INT,
  subject_id INT,
  PRIMARY KEY (student_id, subject_id),
  FOREIGN KEY (student_id) REFERENCES student_information(id) ON DELETE CASCADE,
  FOREIGN KEY (subject_id) REFERENCES available_subjects(id) ON DELETE CASCADE
);
-- Step 5: Create Availability Table (Stores Student Available Times)
CREATE TABLE student_availability (
  id INT AUTO_INCREMENT PRIMARY KEY,
  student_id INT NOT NULL,
  day_of_week ENUM('Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday', 'Saturday', 'Sunday') NOT
NULL,
  start_time TIME NOT NULL,
  end_time TIME NOT NULL,
  timezone VARCHAR(50) DEFAULT 'UTC',
```

```
  FOREIGN KEY (student_id) REFERENCES student_information(id) ON DELETE CASCADE
);
-- Step 6: Insert Sample Data (Optional)
INSERT INTO student_information (name, email, password_hash)
VALUES
('Alice Johnson', 'alice@example.com', 'hashedpassword1'),
('Bob Smith', 'bob@example.com', 'hashedpassword2');
INSERT INTO available_subjects (subject_name)
VALUES
('Mathematics'),
('Physics'),
('Computer Science');
INSERT INTO student_subjects (student_id, subject_id)
VALUES
(1, 1), -- Alice studies Mathematics
(1, 2), -- Alice studies Physics
(2, 3); -- Bob studies Computer Science
INSERT INTO student_availability (student_id, day_of_week, start_time, end_time, timezone)
VALUES
(1, 'Monday', '14:00:00', '16:00:00', 'UTC'), -- Alice available Monday 2-4 PM
(2, 'Wednesday', '10:00:00', '12:00:00', 'UTC'); -- Bob available Wednesday 10AM-12PM
```

Hosting API Functions locally:

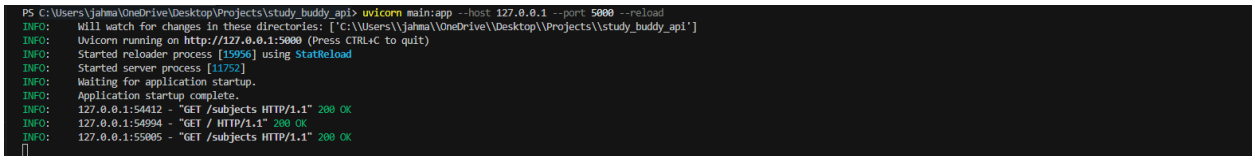
Install fastapi, uvicorn, mysql connector, sqlalchemy

```
pip install fastapi uvicorn mysql-connector-python sqlalchemy
```

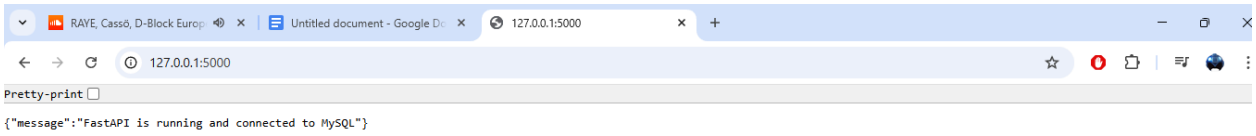
Inside your project folder use the main.py. This main file will contain all the api calls to the database

You need to open a python terminal and run (make sure you are trying to run this from the same folder main.py is in)

```
uvicorn main:app --host 127.0.0.1 --port 5000 --reload
```



To locally host the api. After that you should be able to type in that IP(127.0.0.1) into your browser and see the connection message.



From there you should be able to open the index.html in a browser and see the available subjects.

Sqlalchemy commands are being used in lieu of direct sql calls.

https://docs.sqlalchemy.org/en/20/orm/quickstart.html