MAD TA SESSION ABOUT SQLALCHEMY & JINJA

Basic Flask app:

```python
from flask import Flask, render_template

app = Flask(__name__)

@app.route("/")
def home():
    return render_template("home.html")

if __name__ == "__main__":
    app.run(debug=True)
```

Basic jinja:

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Home</title>
</head>
<body>
    <h1>Welcome to Flask</h1>
    <p>Hello, {{ name }}! Welcome to our first Flask app with Jinja.</p>
</body>
</html>
```

How to pass name?

```python

@app.route("/")
```

```
def home():
    return render_template("home.html", name="Student")
```

Loops in jinja:

```
<h2>Your Favorite Fruits</h2>
<ul>
    {% for fruit in fruits %}
        <li>{{ fruit }}</li>
    {% end for %}
</ul>
```

flask:

```
@app.route("/")
def home():
    fruits = ["Apple", "Banana", "Cherry"]
    return render_template("home.html", name="Student", fruits=fruits)
```

Conditional in jinja:

```
{% if name == "Student" %}
    <p>Welcome back, {{ name }}!</p>
{% else %}
    <p>Who are you?</p>
{% endif %}
```

Template inheritance:

Create base.html:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>{% block title %}My Flask App{% endblock %}</title>
</head>
<body>
    <header>
        <h1>My Flask App</h1>
    </header>

    <main>
        {% block content %}{% endblock %}
    </main>
</body>
</html>
```

Home.html:

```
{% extends "base.html" %}

{% block title %}Home{% endblock %}

{% block content %}
    <h2>Welcome to the Home Page</h2>
    <p>Hello, {{ name }}!</p>
{% endblock %}
```

SQLAlchemy

Basic flask:

```
from flask import Flask, render_template
```

```python
from flask_sqlalchemy import SQLAlchemy

app = Flask(__name__)
app.config['SQLALCHEMY_DATABASE_URI'] = 'sqlite:///students.db'
db = SQLAlchemy(app)

@app.route('/')
def home():
    return render_template('home.html')

if __name__ == '__main__':
    app.run(debug=True)
```

Create model:

```python
class Student(db.Model):
 id = db.Column(db.Integer, primary_key=True)
 name = db.Column(db.String(50), nullable=False)
 grade = db.Column(db.Integer, nullable=False)
```

DONT FORGET:

```python
with app.app_context():
   db.create_all()
```

Query data:

In app.py

```python
@app.route('/students')
def students():
    students = Student.query.all()
    return render_template('students.html', students=students)
```

Populate the db:

```
# Run in Python shell
from app import db, Student
student1 = Student(name="Alice", grade=90)
db.session.add(student1)
db.session.commit()
```

Jinja to display:

<!-- templates/students.html -->

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Students</title>
</head>
<body>
    <h1>Student List</h1>
    <ul>
        {% for student in students %}
            <li>{{ student.name }} - Grade: {{ student.grade }}</li>
        {% endfor %}
    </ul>
</body>
</html>
```

Filter:

```
top_students = Student.query.filter(Student.grade > 80).all()
```

Update in main.py(diff file):

```
from app import db,Student,app
with app.app_context():
```

```
    student1 = Student(name="hgvu", grade=100)
    db.session.add(student1)
    db.session.commit()
```

Deleting

```
@app.route('/delete/<int:id>',methods=['GET','POST'])
def delete(id):
   student=Student.query.get(id)
   db.session.delete(student)
   db.session.commit()
   return render_template('home.html')
```

Html:

```
<ul>
        {% for student in student %}
        <li>{{ student.id }}</li>
       <li>{{ student.name }}</li>
       <li>{{ student.grade }}</li>

    <a href="{{url_for('delete',id=student.id)}}">Delete</a>
       {% endfor %}

   </ul>
```

JINJA DOCS: https://jinja.palletsprojects.com/en/stable/
SQL DATABASE VIEWER APP: https://sqliteviewer.app/
SQLALCHEMY DOCS: https://docs.sqlalchemy.org/en/20/