MAD 1 LA 5

SQLAlchemy is a popular SQL toolkit and Object Relational Mapper.**ORM**?ORM, or Object-Relational Mapping, is a technique that allows developers to interact with a database using the programming language they're coding in, without having to write SQL queries directly.

Basic info:

Table 1: student

Column Name	Column Type	Constraints
student_id	Integer	Primary Key, Auto Increment
roll_number	String	Unique, Not Null
first_name	String	Not Null
last_name	String	

Installation

```
pip install sqlalchemy
```

or

Activate venv:

```
python -m venv myenv
myenv\Scripts\activate
```

Then install

Import

```
from flask_sqlalchemy import SQLAlchemy
```

Config:

```
app.config['SQLALCHEMY_DATABASE_URI'] =
'sqlite:///api_database.sqlite3'
```

Initialise:

```
db=SQLAlchemy()
```

Define: for example:

```
class Student(db.Model):
    __tablename__='student'
    student_id=db.Column(db.Integer,primary_key=True,autoincrement=True)
```

Create

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

Then login routes

```
from flask_sqlalchemy import SQLAlchemy
app=Flask(__name__)
app.config['SQLALCHEMY_DATABASE_URI'] = 'sqlite:///api_database.sqlite3'
db=SQLAlchemy()
class Student(db.Model):
```

```
def myapp(someparam):
#Does something

@app.route('/',methods=['GET','POST']
def index():
#Do something
```

stu=Example.query.all()

Html

{{%for }}
1
2
{{%end for %}}

add/create
@app.route('/st…
Def

assign=student.

db.session.commit()

Assign is not null:
Render exist

stu=student.query.filter_by().delete

```python
class Example(db.Model):
    id=db.Column(db.Integer,primary_key=True,autoincrement=True)
    name=db.Column(db.String(200),nullable=False)
```

id	name
1	ABC

```sql
Select * from id,name
exam=Example.query.all()
```