SQL Alchemy

Installation

→ pip install sqlalchemy | apt install python3-sqlachemy | apt install python3-venv
→ pip install psycopg2 | | python3 -m venv venv
| | source venv/bin/activate
| | pip install sqlalchemy
| | pip install psycopg2-binary

Create Table

```
from sqlalchemy import create_engine, Column, Integer, String
from sqlalchemy.orm import sessionmaker
from sqlalchemy.ext.declarative import declarative_base
engine = create_engine('postgresql://jerinjose:jerinjose@localhost:5432/alchemy', echo=True)

Session = sessionmaker(bind=engine)
session = Session()

Base = declarative_base()

class Student(Base):
    __tablename__ = 'student'
    id = Column(Integer, primary_key=True)
    name = Column(String(50))
    age = Column(Integer)
    grade = Column(String(50))

Base.metadata.create_all(engine)
```

→ metadata

← help for migrate

Dpt. No. _____ Date _____

Page No. _____

Insert Data

```
from sqlalchemy import create_engine, Column, Integer, String
from sqlalchemy.orm import sessionmaker
from sqlalchemy.ext.declarative import declarative_base
engine = create_engine('postgresql://jerinjose:jerinjose@localhost:5432/alchemy',
        echo = False)

Session = sessionmaker(bind = engine)
session = Session()

Base = declarative_base()

class Student(Base):
    __tablename__ = 'student'

    id = Column(Integer, primary_key = True)
    name = Column(String(50))
    age = Column(Integer)
    grade = Column(String(50))

Base.metadata.create_all(engine)   #If database table is created then comment this line

student1 = Student(name="test1", age = 30, grade = "A")
student2 = Student(name="test2", age = 20, grade = "B")
student3 = Student(name="test3", age = 10, grade = "C")
session.add(student1)   # for inserting one table
session.add_all([student2, student3])
Session.commit()
```

Teacher's Signature _____

Get the Data

```
students = session.query(Student)
for student in students:
    print(student.name, student.age, student.grade)

#Get data in order
students = session.query(Student).order_by(Student.name)
for student in students:
    print(student.name)

#Get data by filtering
student = session.query(Student).filter(Student.name == "Devin").first()
print(student.name, student.age)

students = session.query(Student).filter(or_(Student.name == "test1", Student.name == "test2"))
for student in students:
    print(student.name, student.age)

#Count of the result
student_count = session.query(Student).filter(or_(Student.name == "test1", student.name
                                                  == "test2"))
print(student_count)
```

Expt. No. _____ Date _____

Page No. _____

Update Data

```
student = session.query(Student).filter(Student.name == 'test').filter()
student.name = "Lala"
session.commit()
print(student.name)  # get the error, change the user name
```

Delete Data

```
student = session.query(Student).filter(Student.name == "Lala").first()
session.delete(student)
session.commit()
print(student.name)  # Data deleted showing error
```

Basic of postgresql

Installation

pip install sqlalchemy psycopg2-binary

Check

sudo systemctl start postgresql

sudo systemctl status postgresql

=> psql -U root -d db2

=> psql -U postgres -d db2

=> sudo -u postgres psql

List database

\l

\list

Connect specific database

\c db2

Expt. No. _____ Date _____

Page No. _____

List all tables

\dt

View detailed information about a specific table

\d student

Show all rules

\dy

Exit

\q