Database with Python

Handling DB in Python

Database Introduction

- Database systems support SQL, the Structured Query Language, which is used to create, access and manipulate the data. SQL is used to access data, and also to create and exploit the relationships between the stored data. Additionally, these databases support database normalization rules for avoiding redundancy of data.
- The Python programming language has powerful features for database programming. Python supports various databases like MySQL, Oracle, Sybase, PostgreSQL, etc. Python also supports Data Definition Language (DDL), Data Manipulation Language (DML) and Data Query Statements.

Introduction

- Creating SQL file or Establish connection between python and SQL by using SQL.connect(“SQLfile_name”)
- Execute→ used to execute sql query in python
- Commit-> save

Script to connect with SQL

- import sqlite3
- connection = sqlite3.connect('database.db')
- connection.execute('create table car(carID text,CarMake text, CarModel text,CarYear int)')
- print("table is created")

Insert Table in SQL

- connection=sqlite3.connect('database.db')
- connection.execute('insert into Car(CarID,CarMake,CarModel,CarYear) values("c1","A","B",2014)')
- connection.execute('insert into Car(CarID,CarMake,CarModel,CarYear) values("c2","c","d",2004)')
- connection.execute('insert into Car(CarID,CarMake,CarModel,CarYear) values("c3","E","F",1994)')
- connection.execute('insert into Car(CarID,CarMake,CarModel,CarYear) values("c4","G","H",1984)')
- connection.commit()

Retrieve Data

```
import sqlite3

connection=sqlite3.connect('database.db')
result=connection.execute('select * from Car')

for line in result:
        print(line)
```