

SQL trong Python

I. Một số lệnh cơ bản trong Python để tạo cơ sở dữ liệu:

1. Kết nối với 1 cơ sở dữ liệu:

```
dbase=sqlite3.connect('mydata.db')
```

Mở cơ sở dữ liệu có tên mydata.db. Nếu nó không tồn tại nó sẽ tự động tạo ra.

2. `dbase.close()`: đóng kết nối cơ sở dữ liệu
3. `cursor = dbase.cursor()`: tạo một đối tượng cursor để thao tác với cơ sở dữ liệu
4. Tạo bảng trong cơ sở dữ liệu:

Cú pháp:

```
cursor.execute("""CREATE TABLE [IF NOT EXISTS] table_name(  
    col1 datatypes,  
    col2 datatypes,  
    col3 datatypes,  
    -----  
    )""")
```

Ví dụ:

```
cursor.execute("""CREATE TABLE IF NOT EXISTS employees (  
    id INTEGER PRIMARY KEY,  
    name TEXT,  
    age INTEGER  
    )""")
```

5. Chèn dữ liệu vào bảng:

Cú pháp:

```
cursor.execute(""" INSERT INTO ten_bang (cot1, cot2, cot3, ...) VALUES (?, ?, ?, ...) """,  
(gia_tri1, gia_tri2, gia_tri3, ...))
```

Trong đó:

`INSERT INTO ten_bang (cot1, cot2, cot3, ...)` là câu lệnh SQL để chèn dữ liệu vào bảng `ten_bang`.

`("?, ?, ?, ...")` là chuỗi placeholder để đại diện cho giá trị cần chèn.

`(gia_tri1, gia_tri2, gia_tri3, ...)` là một tuple chứa giá trị thực sự tương ứng với các placeholder.

Ví dụ:

```
cursor.execute("INSERT INTO employees (name, age) VALUES (?, ?) ", ("John Doe", 30))
```

6. Lệnh lưu thay đổi vào cơ sở dữ liệu:

```
conn.commit()
```

7. Lệnh thực hiện trong Python:

```
cursor.execute(query)
```

Trong đó: query là câu lệnh truy vấn trong ngôn ngữ SQL

8. Lấy các dòng dữ liệu sau khi đã được truy vấn:

```
rows = cursor.fetchall()
```

9. Lấy dữ liệu từ bảng với điều kiện:

Cú pháp:

```
cursor.execute(" SELECT cot1, cot2, cot3, ...
```

```
FROM ten_bang
```

```
WHERE dieu_kien ")
```

Ví dụ:

```
data = cursor.execute("SELECT * FROM employee_records WHERE ID=1")
```

10. Cập nhật dữ liệu từ bảng:

Cú pháp:

```
cursor.execute("UPDATE ten_bang SET cot1 = gia_tri1, cot2 = gia_tri2 WHERE dieu_kien")
```

Ví dụ:

```
cursor.execute(" UPDATE employee_records set STARS=3 WHERE ID=5 ")
```

11. Xóa dữ liệu từ bảng:

Cú pháp:

```
cursor.execute("DELETE FROM ten_bang WHERE dieu_kien")
```

Ví dụ:

```
cursor.execute(" DELETE from employee_records WHERE ID = 1 ")
```

12. Đọc dữ liệu từ cơ sở dữ liệu vào một DataFrame:

```
df = pd.read_sql_query(query,conn)
```

Ví dụ:

```
df = pd.read_sql_query("SELECT * FROM danh_ba", conn)
```

13. Xuất dữ liệu từ DataFrame ra file *.csv:

```
df.to_csv("filename.csv",index=False)
```

II. Một số đoạn code mẫu về database trong Python:

1. Tạo 1 cơ sở dữ liệu:

```
#create_database.py
```

```
import sqlite3
```

```
dbase = sqlite3.connect('Our_data.db') # Open a database File
```

```
print ('Database opened')
dbase.close()
print (' Database Closed')
```

2. Tạo 1 bảng:

```
#create_table.py
import sqlite3
dbase = sqlite3.connect('Our_data.db') # Open a database File
print ('Database opened')
dbase.execute(""" CREATE TABLE IF NOT EXISTS employee_records(
    ID INT PRIMARY KEY NOT NULL,
    NAME TEXT NOT NULL,
    DIVISION TEXT NOT NULL,
    STARS INT NOT NULL) """)
print ('Table created')
dbase.close()
print (' Database Closed')
```

3. Thêm dữ liệu vào bảng:

```
#insert_record.py
import sqlite3

dbase = sqlite3.connect('Our_data.db') # Open a database File
print ('Database opened')

dbase.execute(""" CREATE TABLE IF NOT EXISTS employee_records(
    ID INT PRIMARY KEY NOT NULL,
    NAME TEXT NOT NULL,
    DIVISION TEXT NOT NULL,
    STARS INT NOT NULL) """)

print ('Table created')

dbase.execute(""" INSERT INTO employee_records(ID,NAME,DIVISION,STARS)
    VALUES(5,'James','Maintenance',4)
""")

dbase.commit()
print ('REcord inserted')

dbase.close()
print (' Database Closed')
```

4. Sử dụng hàm để thêm dữ liệu vào bảng:

```
# using_func.py
import sqlite3

dbase = sqlite3.connect('Our_data.db') # Open a database File
```

```
print ('Database opened')
```

```
dbase.execute(""" CREATE TABLE IF NOT EXISTS employee_records(  
    ID INT PRIMARY KEY NOT NULL,  
    NAME TEXT NOT NULL,  
    DIVISION TEXT NOT NULL,  
    STARS INT NOT NULL) """)
```

```
print ('Table created')
```

```
def insert_record(ID,NAME,DIVISION,STARS):  
    dbase.execute(""" INSERT INTO  
employee_records(ID,NAME,DIVISION,STARS)  
    VALUES(?,?,?,?)""",(ID,NAME,DIVISION,STARS))
```

```
dbase.commit()  
print ('REcord inserted')
```

```
insert_record(6,'Bob','Hardware',4)
```

```
dbase.close()  
print (' Database Closed')
```

5. Đọc dữ liệu từ database:

```
#reading_data.py  
import sqlite3
```

```
dbase = sqlite3.connect('Our_data.db') # Open a database File  
print ('Database opened')
```

```
dbase.execute(""" CREATE TABLE IF NOT EXISTS employee_records(  
    ID INT PRIMARY KEY NOT NULL,  
    NAME TEXT NOT NULL,  
    DIVISION TEXT NOT NULL,  
    STARS INT NOT NULL) """)
```

```
print ('Table created')
```

```
def insert_record(ID,NAME,DIVISION,STARS):  
    dbase.execute(""" INSERT INTO  
employee_records(ID,NAME,DIVISION,STARS)  
    VALUES(?,?,?,?)""",(ID,NAME,DIVISION,STARS))
```

```
dbase.commit()  
print ('REcord inserted')
```

```

##insert_record(6,'Bob','Hardware',4)

def read_Data():
    #from math import *
    data = dbase.execute( SELECT * FROM employee_records ORDER BY
NAME")
    for record in data:
        print ('ID : '+str(record[0]))
        print ('NAME : '+str(record[1]))
        print ('DIVISION : '+str(record[2]))
        print ('STARS : '+str(record[3])+'\n')
    read_Data()

dbase.close()
print (' Database Closed')

```

6. Cập nhật 1 record trong database:

```

#update_record.py
import sqlite3

dbase = sqlite3.connect('Our_data.db') # Open a database File
print ('Database opened')

dbase.execute(" CREATE TABLE IF NOT EXISTS employee_records(
    ID INT PRIMARY KEY NOT NULL,
    NAME TEXT NOT NULL,
    DIVISION TEXT NOT NULL,
    STARS INT NOT NULL) ")

print ('Table created')

def insert_record(ID,NAME,DIVISION,STARS):
    dbase.execute(" INSERT INTO
employee_records(ID,NAME,DIVISION,STARS)
VALUES(?, ?, ?, ?)",(ID,NAME,DIVISION,STARS))

    dbase.commit()
    print ('REcord inserted')

##insert_record(6,'Bob','Hardware',4)

def read_Data():
    #from math import *
    data = dbase.execute(" SELECT * FROM employee_records ORDER BY
NAME")

```

```

for record in data:
    print ('ID : '+str(record[0]))
    print ('NAME : '+str(record[1]))
    print ('DIVISION : '+str(record[2]))
    print ('STARS : '+str(record[3])+'\n')

```

```

read_Data()

```

```

def update_record():
    dbase.execute(" UPDATE employee_records set STARS=3 WHERE ID=5 ")
    dbase.commit()
    print ('Updated')

```

```

update_record()
print ('-----')
read_Data()

```

```

dbase.close()
print (' Database Closed')

```

7. Xóa 1 record trong database:

```

#delete_record.py
import sqlite3

```

```

dbase = sqlite3.connect('Our_data.db') # Open a database File
print ('Database opened')

```

```

dbase.execute(" CREATE TABLE IF NOT EXISTS employee_records(
    ID INT PRIMARY KEY NOT NULL,
    NAME TEXT NOT NULL,
    DIVISION TEXT NOT NULL,
    STARS INT NOT NULL) ")

```

```

print ('Table created')

```

```

def insert_record(ID,NAME,DIVISION,STARS):
    dbase.execute(" INSERT INTO
employee_records(ID,NAME,DIVISION,STARS)
VALUES(?,?,?,?)",(ID,NAME,DIVISION,STARS))

```

```

dbase.commit()
print ('REcord inserted')

```

```

##insert_record(6,'Bob','Hardware',4)

```

```

def read_Data():
    # from math import *
    data = dbase.execute(" SELECT * FROM employee_records ORDER BY
NAME")
    for record in data:
        print ('ID : '+str(record[0]))
        print ('NAME : '+str(record[1]))
        print ('DIVISION : '+str(record[2]))
        print ('STARS : '+str(record[3])+'\n')

read_Data()

def update_record():
    dbase.execute(" UPDATE employee_records set STARS=3 WHERE ID=2 ")
    dbase.commit()
    print ('Updated')

##update_record()
##print '-----'
##read_Data()

def delete_record():
    dbase.execute(" DELETE from employee_records WHERE ID = 1 ")
    dbase.commit()
    print ('Deleted')

delete_record()
print ('-----')
read_Data()

dbase.close()
print (' Database Closed')

```

III. Bài tập về cơ sở dữ liệu:

Bài 1:

- Tạo 1 CSDL mydatabase.db và 1 bảng employees gồm có các cột: ID(khóa chính), name, age.
- Thêm 2 record vào bảng
- Đọc dữ liệu từ bảng và in thông tin lên màn hình.

Bài 2:

- Tạo 1 CSDL contacts.db và 1 bảng contacts gồm có các cột: ID(khóa chính), name, email, phone.
- Viết các hàm thêm dữ liệu vào CSDL, cập nhật thông tin, và tìm kiếm, xóa theo tên trong bảng.
- Thêm 1 số record vào cơ sở dữ liệu.
- Tìm kiếm và in ra thông tin.
- Xóa dữ liệu theo tên.

Bài 3: Cho câu truy vấn sau:

```
SELECT employee_id, first_name, hire_date, strftime('%Y', 'now') - strftime('%Y',
hire_date) AS tham_nien
FROM employees
ORDER BY tham_nien DESC;
```

Dùng Python để tạo cơ sở dữ liệu có thêm dữ liệu vào. Sau đó truy vấn và kết quả sẽ in ra màn hình để kiểm chứng.

Bài 4: Cho câu truy vấn sau:

```
SELECT *
FROM employees
ORDER BY salary DESC
LIMIT 1;
```

Dùng Python để tạo cơ sở dữ liệu có thêm dữ liệu vào. Sau đó truy vấn và kết quả sẽ in ra màn hình để kiểm chứng.

Bài 5:

Tạo cơ sở dữ liệu SQLite3 với một bảng để lưu trữ thông tin về người liên hệ. Danh bạ này sẽ lưu trữ thông tin về các người liên hệ, bao gồm tên, số điện thoại, địa chỉ email và địa chỉ.

Chương trình sẽ tạo menu:

- Thêm dữ liệu vào danh bạ
- Cập nhật dữ liệu danh bạ theo số điện thoại.
- Xóa danh bạ theo số điện thoại
- Xem toàn bộ danh bạ
- Xuất danh bạ ra file *.csv