Introduction to SQL Package

1

Syed Umaid Ahmed

Install Sublime Text Editor

Create Table

```
import sqlite3

conn = sqlite3.connect('customer.db')
c = conn.cursor()


c.execute("""CREATE TABLE customers(

            first_name text,
            last_name text,
            email text

            )""")

#NULL, INTEGER, REAL, TEXT, BLOB

conn.commit()
conn.close()
```

```
import sqlite3

#conn = sqlite3.connect(':memory:')

conn = sqlite3.connect('customer.db')
c = conn.cursor()


c.execute("INSERT INTO customers VALUES ('Ahmed', 'Young', 'syed@lovefordata.com')")

print("Successful Execution")
#NULL, INTEGER, REAL, TEXT, BLOB

conn.commit()
conn.close()
```


Entry into the Table

```
import sqlite3

#conn = sqlite3.connect(':memory:')
conn = sqlite3.connect('customer.db')
c = conn.cursor()

many_persons = [
            ('Naveed', 'Mid', 'naveed@lfd.com'),
            ('Shahid', 'Old', 'ahmnav@lfd.com'),
            ('Asad', 'Younger', 'asad@lfd.com'),
]

c.executemany("INSERT INTO customers VALUES (?,?,?)", many_persons)
print("Successful Execution")
#NULL, INTEGER, REAL, TEXT, BLOB
conn.commit()
conn.close()
```

Many Entries at Once

Designed by Syed Umaid Ahmed

```
import sqlite3

#conn = sqlite3.connect(':memory:')
conn = sqlite3.connect('customer.db')
c = conn.cursor()


c.execute("SELECT * FROM customers")


#c.fetchone()
#c.fetchmany(3)

print(c.fetchall())

conn.commit()
conn.close()
```


Check all/One by One at Once

Designed by Syed Umaid Ahmed

```
import sqlite3

conn = sqlite3.connect('customer.db')
c = conn.cursor()


c.execute("SELECT * FROM customers")


items = c.fetchall()
for item in items:
        print(item)


conn.commit()
conn.close()
```


See all Entries in a List and with Loops

Designed by Syed Umaid Ahmed

```
import sqlite3

#conn = sqlite3.connect(':memory:')
conn = sqlite3.connect('customer.db')
c = conn.cursor()

#ROW_ID's
c.execute("SELECT rowid, * FROM customers")


items = c.fetchall()
for item in items:
        print(item)


conn.commit()
conn.close()
```

See all Entries with Rowid in Tuples

Designed by Syed Umaid Ahmed

```
import sqlite3

#conn = sqlite3.connect(':memory:')
conn = sqlite3.connect('customer.db')
c = conn.cursor()

#Search Value by Column Name
c.execute("SELECT * FROM customers WHERE first_name LIKE 'Ah%' ")


items = c.fetchall()
for item in items:
        print(item)


conn.commit()
conn.close()
```


Find all Entries having “AH” in 1st name columns

Designed by Syed Umaid Ahmed

```
import sqlite3
conn = sqlite3.connect('customer.db')
c = conn.cursor()

c.execute("""UPDATE customers SET first_name = "Najeeb"
            WHERE last_name='Mid'

            """)

conn.commit()
c.execute("SELECT * FROM customers")

items = c.fetchall()
for item in items:
            print(item)

conn.close()
```

Update the Value on basis of some column

```
import sqlite3
conn = sqlite3.connect('customer.db')
c = conn.cursor()

c.execute("""UPDATE customers SET first_name = "John"
            WHERE rowid=2
            """)

conn.commit()

c.execute("SELECT rowid, * FROM customers")


items = c.fetchall()
for item in items:
            print(item)


conn.close()
```


Update the Value on basis of row-id

```
import sqlite3

conn = sqlite3.connect('customer.db')
c = conn.cursor()

c.execute("DELETE FROM customers WHERE rowid=4")

conn.commit()

c.execute("SELECT rowid, * FROM customers")

items = c.fetchall()
for item in items:
        print(item)

conn.close()
```


Delete complete Row on checking of row-id

```
import sqlite3

#conn = sqlite3.connect(':memory:')
conn = sqlite3.connect('customer.db')
c = conn.cursor()


c.execute("SELECT rowid, * FROM customers ORDER BY rowid DESC")

#ASC = ASCENDING

items = c.fetchall()
for item in items:
        print(item)

conn.close()
```

Check all Entries by Row-ID in Descending Order

```
import sqlite3
conn = sqlite3.connect('customer.db')
c = conn.cursor()

c.execute("DELETE FROM customers WHERE rowid=4")

conn.commit()

# DESC Row ID, Access only last 2 elements

c.execute("SELECT rowid, * FROM customers ORDER BY rowid DESC LIMIT 2")

items = c.fetchall()
for item in items:
        print(item)


conn.close()
```

Check all Entries by Row-ID in Descending Order with Two From Last

Designed by Syed Umaid Ahmed

```
import sqlite3

conn = sqlite3.connect('customer.db')
c = conn.cursor()

c.execute("DELETE TABLE customers")

conn.commit()

c.execute("SELECT rowid, * FROM customers")
items = c.fetchall()
for item in items:
        print(item)



conn.comit()
conn.close()
```

HOW TO DELETE TABLES ?

Homework Assignment

Make an Application of All The Functions You have Studied in SQLite

For guidance of the Application Project Follow this Video From hours **1:09:06**

https://youtu.be/byHcYRpMgI4

Designed by Syed Umaid Ahmed

For all the codes used in the slides, You can visit my Github Account to Download

https://github.com/SyedUmaidAhmed/SQL-Lite-Course-Python-MSc

Thanking You

Designed by Syed Umaid Ahmed