Python SQLite Select Data Example

To select data from SQLite database using Python, follow the below steps-

1. Connect to the database by passing a filename to `connect()` function.
2. After this, call `cursor()` function to obtain `Cursor` object.
3. Execute `Select` query using `execute()` function.
4. Next, you are having three methods to fetch the data. Use any one of them-
    i. Use `fetchall()` function to get all the rows.
    ii. Call `fetchone()` to obtain a single row and check if it equal to None or not.
    iii. Assign the output of execute() function to a variable and loop through it to obtain the data.
5. Call `close()` function to close the connection.

Method 1: Using fetchall() to query data in Python SQLite

```
import sqlite3

conn = sqlite3.connect('mobiledevices.db')
print('Connected to database successfully.')

cur = conn.cursor()
cur.execute("select * from androidphones")

rows = cur.fetchall()
for row in rows:
   print("ID: "+str(row[0]))
   print("Brand: "+row[1])
   print("Model: "+row[2])
   print("O.S.: "+row[3])
   print("C.P.U.: "+row[4])
   print("")

conn.close()
```

Output of the above program

```
Connected to database successfully.
ID: 1
Brand: Samsung
Model: Galaxy A7 2018
O.S.: Android v8.0 Oreo
C.P.U.: Octa core 2.2 GHz

ID: 2
Brand: LG
Model: G7 Fit
O.S.: Android v8.1 Oreo
C.P.U.: Quad core 2.15 GHz

ID: 3
Brand: Motorola
Model: G6 Plus
O.S.: Android v8.0 Oreo
C.P.U.: Octa core 2.2 GHz

ID: 4
Brand: Samsung
Model: Galaxy J4 Core
O.S.: Android v8.1 Oreo
C.P.U.: Quad core 1.4 GHz
```

Method 2: Using fetchone() to query data in Python SQLite

```
import sqlite3

conn = sqlite3.connect('mobiledevices.db')
print('Connected to database successfully.')

cur = conn.cursor()
cur.execute("select * from androidphones")

row = cur.fetchone()
while(row != None):
   print("ID: "+str(row[0]))
   print("Brand: "+row[1])
   print("Model: "+row[2])
   print("O.S.: "+row[3])
   print("C.P.U.: "+row[4])
   print("")
   row = cur.fetchone()

conn.close()
```

Output will be same as shown in Method 1.

Method 3: Assigning the output of execute() and iterating through it

```
import sqlite3

conn = sqlite3.connect('mobiledevices.db')
print('Connected to database successfully.')

cur = conn.cursor()
rows = cur.execute("select * from androidphones")
for row in rows:
   print("ID: "+str(row[0]))
   print("Brand: "+row[1])
   print("Model: "+row[2])
   print("O.S.: "+row[3])
   print("C.P.U.: "+row[4])
   print("")

conn.close()
```

Output will be same as shown in Method 1.