ماي سكول في البايثون

Python Mysql

اعداد الاستاذ: حمزة جميل.

العنوان: العراق- النجف.

للتواصل: https://www.facebook.com/hamza.jameel.568

قناتي على اليوتيوب: https://www.youtube.com/channel/UCz6zmCSR3nZEs-0elgW5O7Q

ماي سكول في البايثون

يمكن استخدام البايثون في تطبيقات قواعد البيانات وتعتبر ماي سكول من اشهر قواعد البيانات واكثرها شيوعاً في العالم

قبل البدء بتوضيح هذه ما سكول يجب تثبتها على حاسوبك الشخصي ولتحميلها مجاناً عليك بزيارة الموقع الاتي: https://www.mysql.com/downloads/.

تثبيت ماي سكول درايفر

ليتمكن البايثون من التعامل مع mysql database يجب تثبيت mysql driver في هذا الكتاب سنستخدم درايفر "mysql connector" مع pycharm هناك عدة طرق لتثبيت mysql-connector منها:

الطريقة الاولى:

File➔Settings ➔project name ➔ project interpreter ➔search box➔mysql-connector➔install package➔ok

الطريقة الثاينة:

من خلال واجهة pycharm اختر terminal ثم local واكتب الامر الاتي:

- pip install mysql-connector-python ثم Enter

وانتظر اكتمال عملية التثبيت ثم تحقق من module من خلال استدعائها في اعلى مشروعك

Import mysql.connector

انشاء اتصال بقاعدة البيانات من خلال الكود الاتي:

```
import mysql.connector

myconn = mysql.connector.connect(
  host="localhost",
  user="yourusername",
  passwd="yourpassword"
)

print(myconn)
```

فيديو يوتيوب يوضح عملية التثبيت mysql-connector

https://www.youtube.com/watch?v=c4PsGr-WUrw&list=PLe_SAAskIE0j1K81FY5UI_ts2w0rD5G2B&index=2&t=26s

انشاء قاعدة البيانات:

لانشاء قاعدة البيانات استخدم التعليمة (CREATE DATABASE)

المثال الاتي نقوم بانشاء قاعدة بيانات باسم (mdb1).

```
import mysql.connector
myconn=mysql.connector.connect(
    host="localhost",
    user=" yourusername ",
```

```
    password="yourpassword"
)
mycursor=myconn.cursor()
mycursor.execute("CREATE DATABASE mdb1")
```

في حال تنفيذ الكود اعلاه دون اخطاء فهذا يعني تمت عملية انشاء قاعدة البيانات بنجاح.

للتحقق من وجود قاعدة البيانات استخدم الكود الاتي لعرض قواعد البيانات في النظام

```
import mysql.connector

myconn = mysql.connector.connect(
  host="localhost",
  user="yourusername",
  password="yourpassword"
)

mycursor = myconn.cursor()

mycursor.execute("SHOW DATABASES")

for i in mycursor:
  print(i)
```

بعد انشاء قاعدة البيانات يتم اضافتها الى عملية الاتصال كما ياتي:

```
import mysql.connector
myconn=mysql.connector.connect(
    host="localhost",
    user=" yourusername ",
    password="password",
    database="mdb1"
)
```

رابط فيديو انشاء قاعدة البيانات

https://www.youtube.com/watch?v=k7cB34_VYKc&list=PLe_SAAskIE0j1K81FY5UI_ts2w0rD5G2B&index=2

انشاء الجدول

يتم انشاء الجدول باستخدام التعليمة الاتية:

```
import mysql.connector
myconn=mysql.connector.connect(
    host="localhost",
    user=" yourusername ",
    password="password",
    database="mdb1"
)
mycursor=myconn.cursor()
#create table in mdb1
mycursor.execute("CREATE TABLE students (id INT AUTO_INCREMENT PRIMARY KEY,
name VARCHAR(100), address VARCHAR(80), depart VARCHAR(60))")
```

تم انشاء جدول باسم (students) يضم اربع حقول (id, name, address, depart) بانواع بيانية مختلفة وبحجوم مختلفة ايضاً.

عرض الجداول في قاعدة البيانات باستخدام (SHOW TABLES) كما في المثال الاتي:

```
import mysql.connector

myconn = mysql.connector.connect(
  host="localhost",
  user="yourusername",
  passwd="yourpassword",
  database="mdb1"
)

mycursor = myconn.cursor()
mycursor.execute("SHOW TABLES")

for i in mycursor:
  print(i)
```

رابط انشاء جدول

https://www.youtube.com/watch?v=a_hrOQdi9_w&list=PLe_SAAskIE0j1K81FY5UI_ts2w0rD5G2B&index=3

اضافة البيانات

يتم اضافة البيانات الى الجدول يتم كما في المثال الاتي:

```
import mysql.connector
myconn=mysql.connector.connect(
    host="localhost",
    user=" yourusername ",
    password="yourpassword",
    database="mdb1"
)
mycursor=myconn.cursor()
#insert data to table in mdb1
name, address, depart=input("Enter your name, address, depart:").split()
sql="INSERT INTO students (name, address, depart) VALUES(%s, %s, %s)"
value=(name, address, depart)
mycursor.execute(sql,value)
myconn.commit()
```

يجب استخدام عبارة myconn.commit هي ضرورية لحفظ التغيرات على البيانات في الجدول.

كما يمكن اضافة عدة سجلات لبيانات الجدول من خلال استخدام قائمة من الصفوف

```
import mysql.connector
myconn=mysql.connector.connect(
    host="localhost",
    user=" yourusername ",
    password="yourpassword",
    database="mdb1"
)
mycursor=myconn.cursor()
```

```
#insert data to table in mdb1
sql="INSERT INTO students (name, address, depart) VALUES(%s, %s, %s)"
#to insert mutiple data to table we use list of tuple
value=[
    ("Hani", "Mousel", "developer"),
    ("Noori", "diyala", "designer"),
    ("Nada", "Erbil", "security")
]
mycursor.executemany(sql, value)
#make change data on table
myconn.commit()
print(mycursor.lastrowid," ID", " Insert done")
```

رابط الفيديو

https://www.youtube.com/watch?v=AtiQstjf_EA&list=PLe_SAAsklE0j1K81FY5UI_ts2w0rD5G2B&index=4

عرض بيانات الجدول:

يمكن عرض بيانات الجدول في mysql من خلال عبارة select كما في المثال الاتي:

```
import mysql.connector

myconn = mysql.connector.connect(
  host="localhost",
  user="yourusername",
  password="yourpassword",
  database="mdb1"
)
mycursor = myconn.cursor()
mycursor.execute("SELECT * FROM students")
result = mycursor.fetchall()

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

دالة fetchall(): تعيد هذه الدالة كل السجلات من جدول قاعدة البيانات.

يمكن اختيار حقول معينة من الجدول

```
import mysql.connector

myconn = mysql.connector.connect(
  host="localhost",
  user="yourusername",
  password="yourpassword",
  database="mdb1"
)
mycursor = myconn.cursor()
mycursor.execute("SELECT name, address FROM sutdents")
result = mycursor.fetchall()

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

يمكن عرض بيانات السجل الاول من الجدول باستخدام دالة fetchone() كما في المثال الاتي:

```
mycursor.execute("SELECT * FROM students")
result=mycursor.fetchone()
print(result)
```

رابط الفيديو

https://www.youtube.com/watch?v=zTDdTAh-j1o&list=PLe_SAAskIE0j1K81FY5UI_ts2w0rD5G2B&index=5

البحث عن بيانات الجدول

عندما يتم اختيار سجل معين من بين عدد كبير من سجلات الجدول فهذا يسمى فلترة البيانات حسب شرط معين، في المثال الاتي يتم اختيار الطالب الذي عنوان سكنه بغداد.

```
import mysql.connector
myconn=mysql.connector.connect(
    host="localhost",
    user="yourusername",
    password="yourpassword",
    database="mdb1"
)
mycursor=myconn.cursor()
sql="SELECT * FROM students WHERE address='Baghdad'"
mycursor.execute(sql)
result=mycursor.fetchall()
for i in result:
    print(i)
```

يمكن استرجاع اكثر من سجل حسب الشرط دالة عبارة sql حيث في المثال الاتي يتم البحث عن كل العناوين التي تبدأ بالحرفين "Ba".

```
sql="SELECT * FROM students WHERE address LIKE '%Ba%'"
mycursor.execute(sql)
result=mycursor.fetchall()
for i in result:
    print(i)
```

كما يمكن السماح بادخال البيانات للبحث عنها في الجدول كما في المثال الاتي:

```
address=input("Enter your address: ")
sql="SELECT * FROM students WHERE address=%s"
value= (address,)
mycursor.execute(sql, value)
result=mycursor.fetchall()
for i in result:
    print(i)
```

رابط الفيديو

https://www.youtube.com/watch?v=ol6mrNCW-ZE&list=PLe_SAAskIE0j1K81FY5UI_ts2w0rD5G2B&index=6&pbjreload=10

ترتيب بيانات الجدول:

لترتيب بيانات الجدول نستخدم عبارة "ORDER BY" حيث يمكن ترتيبها بطريقة تصاعدية او تنازلية حسب المطلوب لاحظ المثال الاتي حيص يتم ترتيب بيانات الجدول حسب حقل القسم.

```
import mysql.connector
myconn=mysql.connector.connect(
    host="localhost",
    user="yourusername",
    password="yourpassword",
    database="mdb1"
)
mycursor=myconn.cursor()
#order data of the table by depart field
sql="SELECT * FROM students ORDER BY depart"
mycursor.execute(sql)
result=mycursor.fetchall()
for i in result:
    print(i)
```

كما يمكن ترتيب البيانات حسب حقل الاسم فتصبح عبارة sql كالاتي:

```
#order table by name field
sql="SELECT * FROM students ORDER BY name"
```

الطريقة الافتراضية هي الترتيب التصاعدي ولترتيب البيانات بشكل تنازلي نستخدم "DESC" كما ياتي:

```
#order by depart desc
sql="SELECT * FROM students ORDER BY depart DESC"
```

رابط الفيديو

https://www.youtube.com/watch?v=MPb0e-CjIlE&list=PLe_SAAskIE0j1K81FY5UI_ts2w0rD5G2B&index=7

حذف السجلات:

يمكن حذف السجلات من الجدول باستخدام عبارة "DELETE FROM"

المثال الاتي يتم حذف سجل حسب الشرط الاتي:

```
import mysql.connector
myconn=mysql.connector.connect(
    host="localhost",
    user="yourusername",
    password="yourpassword",
    database="mdb1"
)
mycursor=myconn.cursor()
#delete record that id =7
sql="DELETE FROM students WHERE id=7"
mycursor.execute(sql)
myconn.commit()
#display all records in table
mycursor.execute("SELECT * FROM students")
result=mycursor.fetchall()
for i in result:
    print(i)
```

لا تنسى استخدام myconn.commit() لحفظ التغيرات في جدول قاعدة البيانات

في المثال الاتي يتم حذف سجل بعنوان "Kut"

```
sql="DELETE FROM students WHERE address=%s"
val=("Kut",)
mycursor.execute(sql, val)
```

رابط الفيديو

https://www.youtube.com/watch?v=obQdORKCpFU&list=PLe_SAAskIE0j1K81FY5UI_ts2w0rD5G2B&index=8

حذف الجدول:

يمكن حذف الجدول من قاعدة البيانات باستخدام عبارة "DROP TABLE" في هذا المثال سنقوم بعرض جداول قاعدة البيانات ومن ثم حذف احدهما كما في هذا المثال:

```
import mysql.connector
myconn=mysql.connector.connect(
    host="localhost",
    user="yourusername",
    password="yourpassword",
    database="mdb1"
)
mycursor=myconn.cursor()
#display tables
mycursor.execute("SHOW TABLES")
for i in mycursor:
    print(i)
#delete table
mycursor.execute("DROP TABLE employees")
#display tables after delete
mycursor.execute("SHOW TABLES")
for i in mycursor:
    print(i)
```

في المثال اعلاه اذا كان الجدول غير موجود فهذا سيولد خطأ ولتجنب هذا الخطأ نستخدم العبارة الاتية:

```
#delete table if exist in database
mycursor.execute("DROP TABLE IF EXISTS employees")
```

رابط الفيديو

https://www.youtube.com/watch?v=nmq03pbAnZ0&list=PLe_SAAskIE0j1K81FY5UI_ts2w0rD5G2B&index=9

تحديث البيانات:

يمكن تحديث بيانات سجلات الجدول لقاعدة البيانات باستخدام عبارة "UPDATE".

```
import mysql.connector
myconn=mysql.connector.connect(
    host="localhost",
    user="yourusername",
    password="yourpassword",
    database="mdb1"
)
mycursor=myconn.cursor()
#display data before update
print("Data before
update\n_____")
mycursor.execute("select * from students")
for i in mycursor:
    print(i)
sql="update students set address='Kut' where name='Hamza'"
```

```
mycursor.execute(sql)
myconn.commit()
mycursor.execute("select * from students")
print("Data after update\n_____________________________________")
for i in mycursor:
    print(i)
```

يمكن التعديل على sql بالطريقة التي تتلائم مع متطلباتك.

```
sql="update students set address=%s where address=%s"
value=("Kut", "Basra")
mycursor.execute(sql, value)
```

رابط الفيديو

https://www.youtube.com/watch?v=tG0w46lLzdQ&list=PLe_SAAskIE0j1K81FY5UI_ts2w0rD5G2B&index=10

تحديد عرض البيانات:

يمكن تحديد عرض معين من السجلات لجدول قاعدة البيانات باستخدام عبارة "LIMIT".

في المثال الاتي سنختار ٥ سجلات من الجدول ابتداءاً من السجل الاول.

```
import mysql.connector
myconn=mysql.connector.connect(
    host="localhost",
    user="yourusername",
    password="yourpassword",
    database="mdb1"
)
mycursor=myconn.cursor()
#display 5 records from first
sql="select * from students limit 5"
mycursor.execute(sql)
result=mycursor.fetchall()
for i in result:
    print(i)
```

في الكود الاتي سنقوم بعرض اربع سجلات ابتداءاً من السجل الثالث:

```
#display 4 records start from 3rd record
sql="select * from students limit 4 offset 3"
```

رابط الفيديو

https://www.youtube.com/watch?v=m8T4y7eXSbQ&list=PLe_SAAskIE0j1K81FY5UI_ts2w0rD5G2B&index=11

تم بحمد الله