



Conexión con MySQL desde python

Ejercicio práctico simple para familiarizarse con el acceso a una base de datos MySQL desde python

Configuración del entorno

Debemos instalar “MySQLConnection” que es el conector que utilizaremos para interactuar con la base de datos.

Agregando A Python3

System-Product-Name:~\$ pip3 install mysql-connector

Probar el funcionamiento desde la consola:

```
>>> import mysql.connector
```

---no debe mostrarte mensaje de error---

Escenario planteado:

contamos con una base de datos en la cual existe una tabla de productos sobre la cual realizaremos las consultas SQL

Detalle de la tabla productos

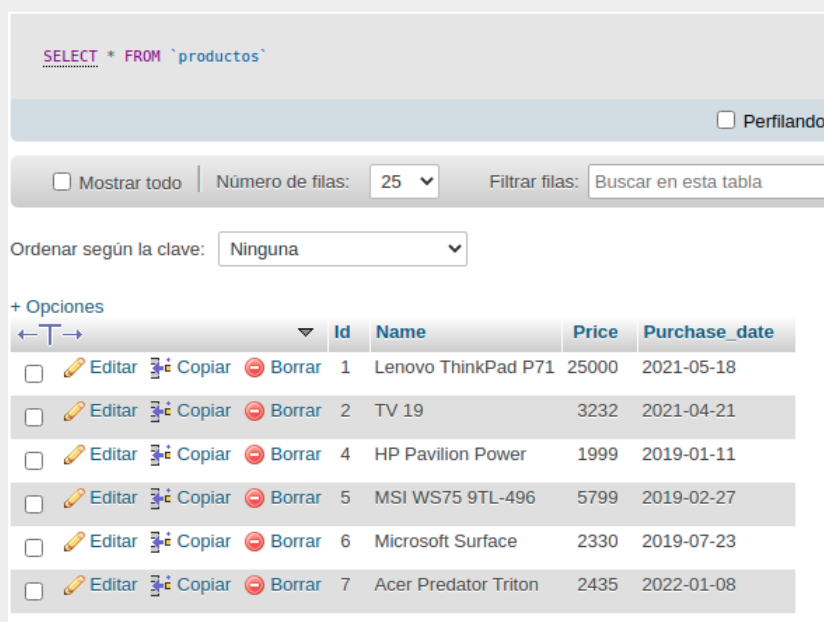
Estructura:



The screenshot shows a database management interface with the following tabs: Examinar, Estructura, SQL, Buscar, Insertar, and Exportar. The 'Estructura de tabla' tab is selected, showing the table structure for 'productos'.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado
1	Id	int(11)			No	Ninguna
2	Name	varchar(250)	latin1_swedish_ci		No	Ninguna
3	Price	float			No	Ninguna
4	Purchase_date	date			No	Ninguna

Detalle del contenido:



The screenshot shows the same database management interface with the 'productos' table selected. The 'Mostrar todo' checkbox is checked, and the 'Número de filas' is set to 25. The 'Filtrar filas' field contains 'Buscar en esta tabla'. The 'Ordenar según la clave' dropdown is set to 'Ninguna'. The table content is displayed below the query editor.

	Id	Name	Price	Purchase_date
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	1	Lenovo ThinkPad P71	25000	2021-05-18
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	2	TV 19	3232	2021-04-21
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	4	HP Pavilion Power	1999	2019-01-11
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	5	MSI WS75 9TL-496	5799	2019-02-27
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	6	Microsoft Surface	2330	2019-07-23
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	7	Acer Predator Triton	2435	2022-01-08

A-Creando una conexión desde python para verificar

Consideraciones:

Todo el código propuesta utiliza la definición y creación de la conexión, en próximos ejemplos vamos a importar el archivo cada vez que se requiera.

Importante:

No olvides de cambiar los parámetros por los correspondientes

```
connection=mysql.connector.connect(  
host='localhost',  
database='nombre de tu base de datos',  
user='usuario de la base de datos',  
password='contraseña')
```

archivo= conexion.py

```
import mysql.connector  
from mysql.connector import Error  
try:  
    connection=mysql.connector.connect(host='localhost',database='python-  
db',user='root',password='mariano')  
    if connection.is_connected():  
        db_Info=connection.get_server_info()  
        print("Connected to MySQL Server version", db_Info)  
        cursor=connection.cursor()  
        cursor.execute("select database();")  
        record=cursor.fetchone()  
        print("Se ha conectado a la base de datos: ", record)  
except Error as e:  
    print ("Error while connecting to MySQL", e)  
finally:  
    if connection.is_connected():  
        cursor.close()  
        connection.close()  
        print("MySQL connection is closed")
```

1- Primero debes crear una base de datos con el nombre “tienda” (o el nombre que desees)

2- En el script de “productos.sql” tienes el código para generar una tabla llamada “productos” en tu base de datos.

2.1- Crear la tabla por medio de código utilizando el archivo creartabla.py y luego insertar valores en ella.

2.2- Puedes generar la tabla e insertarle los valores en forma manual

```
import mysql.connector
```

try:

```
connection = mysql.connector.connect(host='localhost',
                                     database='python-db',
                                     user='root',
                                     password='mariano')
```

```
mysql_Create_Table_Query = """CREATE TABLE productos (  
    Id int(11) NOT NULL,  
    Name varchar(250) NOT NULL,  
    Price float NOT NULL,  
    Purchase_date Date NOT NULL,  
    PRIMARY KEY (Id)) """
```

```
cursor = connection.cursor()
result = cursor.execute(mySql_Create_Table_Query)
print("tabla productos creada con éxito,!! ")
```

```
except mysql.connector.Error as error:
```

```
print("Failed to create table in MySQL: {}".format(error))
```

finally:

```
if connection.is_connected():
    cursor.close()
    connection.close()
    print("MySQL connection is closed")
```

3- Verifica que se haya generado la tabla en tu bd y que puedas acceder a ella

4- Verifica que puedes acceder a Mysql

```
mysql -h localhost -u 'usuario' -p 'password'
```

B-insertando datos en la tabla

--considerar que el campo id debe ser único para evitar errores, entonces reemplazar el valor “X” por un valor entero que no este en la tabla--

archivo= insertar-fila-simple.py

```
import mysql.connector
```

```
try:
```

```
connection = mysql.connector.connect(host='localhost',
                                     database='python-db',
                                     user='root',
                                     password='mariano')
```

```
mysql_insert_query = """INSERT INTO productos (Id, Name, Price, Purchase_date)
VALUES
(X, 'RCL', 49000, '2022-10-20') """
```

```
cursor = connection.cursor()
cursor.execute(mySql_insert_query)
connection.commit()
print(cursor.rowcount, "Record inserted successfully into products table")
cursor.close()
```

```
except mysql.connector.Error as error:
    print("Failed to insert record into product table {}".format(error))
```

```
finally:
    if connection.is_connected():
        connection.close()
        print("MySQL connection is closed")
```

Inserciones múltiples

--inserción de más de un registro a la vez--

archivo= insercionmultiple.py

```
import mysql.connector
```

```
try:
```

```
    connection = mysql.connector.connect(host='localhost',
                                         database='python-db',
                                         user='root',
                                         password='mariano')
```

```
    mySql_insert_query = """INSERT INTO productos (Id, Name, Price, Purchase_date)
                           VALUES (%s, %s, %s, %s) """
```

```
    records_to_insert = [(4, 'HP Pavilion Power', 1999, '2019-01-11'),
                          (5, 'MSI WS75 9TL-496', 5799, '2019-02-27'),
                          (6, 'Microsoft Surface', 2330, '2019-07-23')]
```

```
    cursor = connection.cursor()
    cursor.executemany(mySql_insert_query, records_to_insert)
    connection.commit()
    print(cursor.rowcount, "Record inserted successfully into products table")
```

```
except mysql.connector.Error as error:
    print("Failed to insert record into MySQL table {}".format(error))
```

```
finally:
```

```
    if connection.is_connected():
        cursor.close()
        connection.close()
        print("MySQL connection is closed")
```

Inserción utilizando variables

--inserción de registros pasando valores por parámetro--

archivo= insertarconvariables.py

```
import mysql.connector

def insert_variables_into_table(id, name, price, purchase_date):
    try:
        connection = mysql.connector.connect(host='localhost',
                                             database='python-db',
                                             user='root',
                                             password='mariano')

        cursor = connection.cursor()
        mySql_insert_query = """INSERT INTO productos (Id, Name, Price, Purchase_date)
                                VALUES (%s, %s, %s, %s) """

        record = (id, name, price, purchase_date)
        cursor.execute(mySql_insert_query, record)
        connection.commit()
        print("Record inserted successfully into products table")

    except mysql.connector.Error as error:
        print("Failed to insert into MySQL table {}".format(error))

    finally:
        if connection.is_connected():
            cursor.close()
            connection.close()
            print("MySQL connection is closed")
```

Insertar con Timestamp

– aquí se estan pasando como parametro los valores a insertar--

– se le da un formato a la fecha--

archivo= insertarcontimestamp.py

```
from datetime import datetime
```

```
import mysql.connector
```

```
try:
```

```
    connection = mysql.connector.connect(host='localhost',
                                         database='python-db',
                                         user='root',
                                         password='mariano')
```

```
    mySql_insert_query = """INSERT INTO productos (Id, Name, Price, Purchase_date)
                            VALUES (%s, %s, %s, %s) """
```

```
    cursor = connection.cursor()
```

```

current_Date = datetime.now()
# convertir la fecha en el formato que desee
formatted_date = current_Date.strftime('%Y-%m-%d %H:%M:%S')
insert_tuple = (7, 'Acer Predator Triton', 2435, current_Date)

result = cursor.execute(mySql_insert_query, insert_tuple)
connection.commit()
print("Date Record inserted successfully")

except mysql.connector.Error as error:
    connection.rollback()
    print("Failed to insert into MySQL table {}".format(error))

finally:
    if connection.is_connected():
        cursor.close()
        connection.close()
        print("MySQL connection is closed")

```

C- Seleccionando datos

--descomenta la línea que contiene la sentencia que quieras ejecutar o añade una propia para probar--

Prueba las sentencias Sql realizando consultas :

// **describe la tabla productos**

mySql_query = "describe productos"

// **selección completa de la tabla**

mySql_query = "select * from productos"

// **productos que contengan “av en su nombre”**

mySql_query = "select * from productos where Name like '%av%'"

// **cuenta los productos**

mySql_query = "select count(*) from productos"

archivo= select.py

import mysql.connector

try:

```

connection = mysql.connector.connect(host='localhost',
                                     database='python-db',
                                     user='root',
                                     password='mariano')

```

mySql_query = "describe productos"

mySql_query = "select * from productos"

mySql_query = "select * from productos where Name like '%av%'"

mySql_query = "select count(*) from productos"

```

cursor = connection.cursor()

```

```

cursor.execute(mySql_query)

rows=cursor.fetchall()
for row in rows:
    print(row)

cursor.close()

except mysql.connector.Error as error:
    print("Failed to insert record into MySQL table {}".format(error))

finally:
    if connection.is_connected():
        connection.close()
        print("MySQL connection is closed")

```

D- Borrando una fila

– aquí deberás tener en cuenta los criterios para eliminar registros--

archivo= borrar.py

```

import mysql.connector

import mysql.connector

try:
    connection = mysql.connector.connect(host='localhost',
                                         database='python-db',
                                         user='root',
                                         password='mariano')

    mySql_insert_query = " DELETE FROM productos WHERE Id = '8';"

    cursor = connection.cursor()
    cursor.execute(mySql_insert_query)
    connection.commit()
    print(cursor.rowcount, "registro(s) borrado")

except mysql.connector.Error as error:
    print("Failed to delete record into MySQL table {}".format(error))

finally:
    if connection.is_connected():
        cursor.close()
        connection.close()
        print("MySQL connection is closed")

```


E- Actualizando una fila

– aquí deberás tener en cuenta los criterios para actualizar registros--

archivo= update.py

```
import mysql.connector

import mysql.connector

try:
    connection = mysql.connector.connect(host='localhost',
                                         database='python-db',
                                         user='root',
                                         password='mariano')

    mySql_insert_query = " UPDATE productos SET price='0' WHERE Id='8';"

    cursor = connection.cursor()
    cursor.execute(mySql_insert_query)
    connection.commit()
    print(cursor.rowcount, "registro(s) actualizado")

except mysql.connector.Error as error:
    print("Failed to delete record into MySQL table {}".format(error))

finally:
    if connection.is_connected():
        cursor.close()
        connection.close()
        print("MySQL connection is closed")
```

NOTA:

se adjuntan los archivos correspondientes para todo el código

Orden de ejecución posible:

- 1- crear la tabla**
- 2- seleccionar registros**
- 3- insertar registros**
- 4- verificar cambios**
- 5- borrar registros**
- 6- verificar cambios**
- 7- actualizar registros**
- 8- verificar cambios**