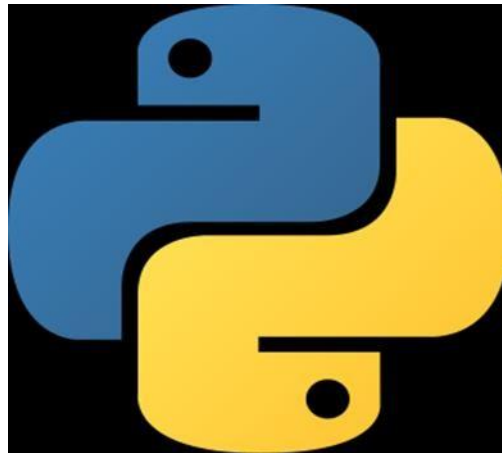


Python y BBDD MySQL



Facilitador: Ing. Henry Duque.

Python y MySQL

- Instalar el controlador MySQL
- Python necesita un controlador MySQL para acceder a la base de datos MySQL.
- En esta presentación usaremos el controlador "Conector MySQL".
- Recomendamos que use PIP para instalar "MySQL Connector".
- Es muy probable que PIP ya esté instalado en su entorno de Python.
- Acceda a través de la línea de comandos a la ubicación de PIP y escriba el siguiente comando, para proceder a descargar e instalar "MySQL Connector":

>python -m pip install mysql-connector

¿Qué es PIP?

PIP es un administrador de paquetes para paquetes Python, o módulos si lo desea.

Nota: si tiene Python versión 3.4 o posterior, PIP está incluido por defecto.

Python y MySQL

```
C:\Python34>cd scripts
```

```
C:\Python34\Scripts>python -m pip install mysql-connector
```

```
'python' is not recognized as an internal or external command,  
operable program or batch file.
```

```
C:\Python34\Scripts>c:\python34\python -m pip install mysql-connector
```

```
Downloading/unpacking mysql-connector
```

```
Running setup.py (path:C:\Users\hduqu\AppData\Local\Temp\pip_build_hduqu\mysql-connector\setup.py) egg_info for package  
mysql-connector
```

```
Installing collected packages: mysql-connector
```

```
Running setup.py install for mysql-connector
```

```
Not Installing C Extension
```

```
Successfully installed mysql-connector
```

```
Cleaning up...
```

```
C:\Python34\Scripts>
```

Python y MySQL

- Editar un programa con el siguiente contenido y ejecutar a fin de verificar si la instalación fue satisfactoria:

Nombre del archivo: demo_prueba_mysql.py

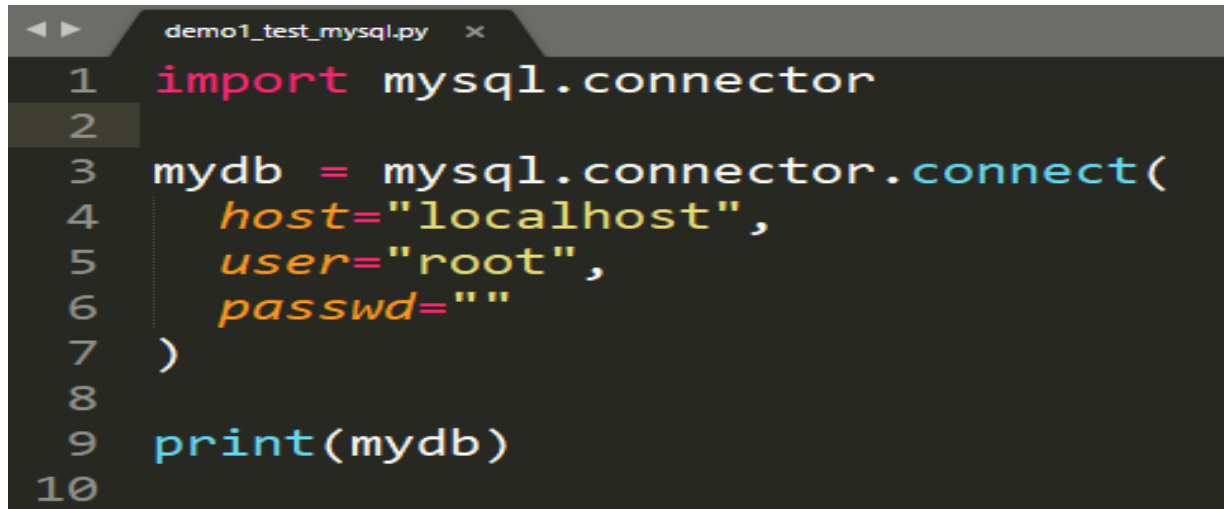
Contenido:

```
import mysql.connector
```

```
#Si este programa se ejecuta sin errores, tiene  
instalado el módulo "mysql.connector".
```

Python y MySQL

- Conexión a MySQL: Editar el programa demo1_test_mysql.py



```
demo1_test_mysql.py x
1 import mysql.connector
2
3 mydb = mysql.connector.connect(
4     host="localhost",
5     user="root",
6     passwd=""
7 )
8
9 print(mydb)
10
```

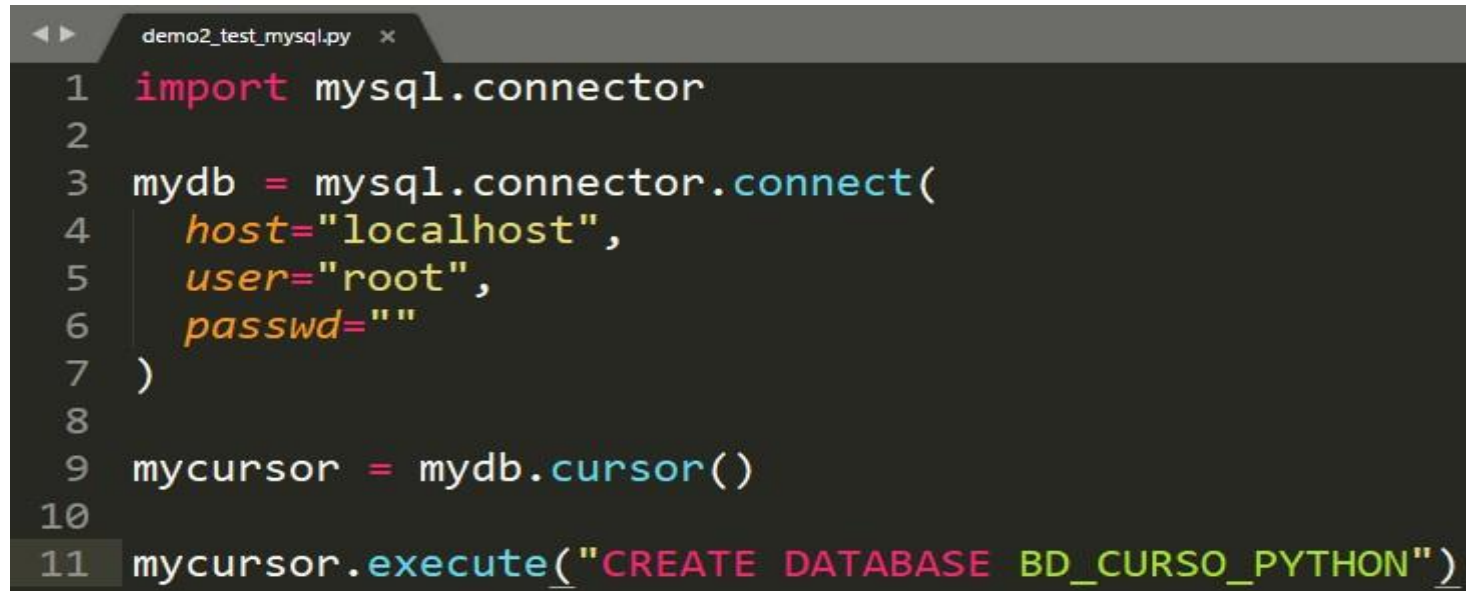
- Ejecutar:

```
C:\Python34>python demo1_test_mysql.py
<mysql.connector.connection.MySQLConnection object at 0x000000002CFA518>

C:\Python34>_
```

Python y MySQL

- Creación de una Base de Datos: Editar el programa demo2_test_mysql.py

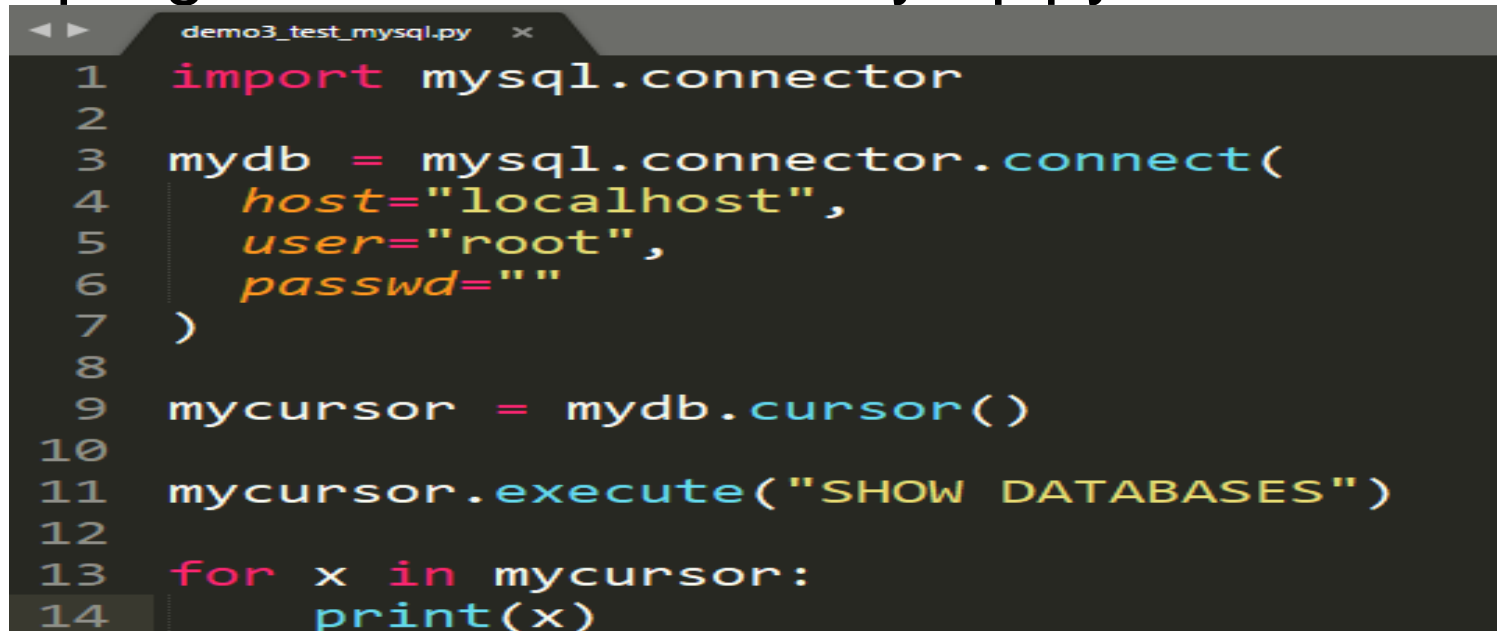
A screenshot of a code editor window titled 'demo2_test_mysql.py'. The code is written in Python and uses the mysql.connector library. It connects to a MySQL database on localhost with the user 'root' and no password. Then, it creates a new database named 'BD_CURSO_PYTHON'.

```
1 import mysql.connector
2
3 mydb = mysql.connector.connect(
4     host="localhost",
5     user="root",
6     passwd=""
7 )
8
9 mycursor = mydb.cursor()
10
11 mycursor.execute("CREATE DATABASE BD_CURSO_PYTHON")
```

- Ejecutar y verificar

Python y MySQL

- Mostrar las Bases de Datos existentes: Editar el programa demo3_test_mysql.py



```
demo3_test_mysql.py x
1  import mysql.connector
2
3  mydb = mysql.connector.connect(
4      host="localhost",
5      user="root",
6      passwd=""
7  )
8
9  mycursor = mydb.cursor()
10
11 mycursor.execute("SHOW DATABASES")
12
13 for x in mycursor:
14     print(x)
```

- Ejecutar y verificar

Python y MySQL

- Conectarse a una Base de Datos existente:
Editar el programa demo4_test_mysql.py

```
demo4_test_mysql.py
1 import mysql.connector
2
3 mydb = mysql.connector.connect(
4     host="localhost",
5     user="root",
6     passwd="",
7     database="BD_CURSO_PYTHON"
8 )
9 # Si este programa se ejecuta sin errores,
10 # la base de datos "BD_CURSO_PYTHON"
11 # existe en su sistema
```

- Ejecutar y verificar

Python y MySQL

- Crear una tabla: Editar el programa demo5_test_mysql.py

```
1 import mysql.connector
2
3 mydb = mysql.connector.connect(
4     host="localhost",
5     user="root",
6     passwd="",
7     database="BD_CURSO_PYTHON"
8 )
9
10 mycursor = mydb.cursor()
11
12 mycursor.execute("CREATE TABLE PERSONAS (CEDULA VARCHAR(10),\
13                                     NOMBRE VARCHAR(30),\
14                                     APELLIDO VARCHAR(30),\
15                                     DIRECCION VARCHAR(30),\
16                                     FECHANAC DATE)")
```

- Ejecutar y verificar

Python y MySQL

- Modificar una tabla: Editar el programa demo6_test_mysql.py

```
demo6_test_mysql.py x
1 import mysql.connector
2
3 mydb = mysql.connector.connect(
4     host="localhost",
5     user="root",
6     passwd="",
7     database="BD_CURSO_PYTHON"
8 )
9
10 mycursor = mydb.cursor()
11
12 mycursor.execute("ALTER TABLE PERSONAS \
13     ADD COLUMN ID INT AUTO_INCREMENT PRIMARY KEY")
```

- Ejecutar y verificar

Python y MySQL

- Insertar un registro en una tabla: Editar el programa demo7_test_mysql.py

```
1 import mysql.connector
2
3 mydb = mysql.connector.connect(
4     host="localhost",
5     user="root",
6     passwd="",
7     database="BD_CURSO_PYTHON"
8 )
9 mycursor = mydb.cursor()
10 sql = "INSERT INTO PERSONAS(CEDULA,NOMBRE,APELLIDO,\
11     DIRECCION,FECHANAC) \
12     VALUES (%s,%s,%s,%s,%s)"
13 valores = ("1234","JOSE","PEREZ","CHACAITO","2000/01/01")
14 mycursor.execute(sql,valores)
15
16 mydb.commit()
17 print("1 registro insertado, ID:", mycursor.lastrowid)
```

- Ejecutar y verificar

Python y MySQL

- Insertar varios registros en una tabla: Editar el programa demo8_test_mysql.py

```
1  import mysql.connector
2  mydb = mysql.connector.connect(
3      host="localhost",
4      user="root",
5      passwd="",
6      database="BD_CURSO_PYTHON"
7  )
8  mycursor = mydb.cursor()
9  sql = "INSERT INTO PERSONAS(CEDULA,NOMBRE,APELLIDO,\
10      DIRECCION,FECHANAC) \
11      VALUES (%s,%s,%s,%s,%s)"
12  valores = [
13      ("1234","JOSE","PEREZ","CHACAITO","2000/01/01"),
14      ("5678","LAURA","SUAREZ","SBNA.GRANDE","1998/06/15"),
15      ("9012","YAMILE","CACERES","PZA.VZLA","1990/07/25")
16  ]
17  mycursor.executemany(sql,valores) #insercción de varios registros
18  mydb.commit()
19  print(mycursor.rowcount, "fueron insertados")
20  print("Ultimo registro insertado, ID:", mycursor.lastrowid)
```

- Ejecutar y verificar

Python y MySQL

- Consultar registros de una tabla: Editar el programa demo9_test_mysql.py

```
1  import mysql.connector
2
3  mydb = mysql.connector.connect(
4      host="localhost",
5      user="root",
6      passwd="",
7      database="BD_CURSO_PYTHON"
8  )
9  mycursor = mydb.cursor()
10
11  sql = "SELECT * FROM PERSONAS"
12
13  mycursor.execute(sql)
14
15  myresult = mycursor.fetchall()
16
17  for x in myresult:
18      print(x)
```

- Ejecutar y verificar

Python y MySQL

- Consultar algunos campos de los registros en una tabla: Editar el programa `demo10_test_mysql.py`


```
1  import mysql.connector
2
3  mydb = mysql.connector.connect(
4      host="localhost",
5      user="root",
6      passwd="",
7      database="BD_CURSO_PYTHON"
8  )
9  mycursor = mydb.cursor()
10
11  sql = "SELECT CEDULA, NOMBRE, APELLIDO FROM PERSONAS"
12
13  mycursor.execute(sql)
14
15  myresult = mycursor.fetchall() # devuelve todos
16  #myresult = mycursor.fetchone() # devuelve sólo el primero
17
18  for x in myresult:
19      print(x)
```

- Ejecutar y verificar

Python y MySQL

- Consulta condicionada: Editar el programa demo11_test_mysql.py

```
1  import mysql.connector
2  ▼ mydb = mysql.connector.connect(
3      host="localhost",
4      user="root",
5      passwd="",
6      database="BD_CURSO_PYTHON"
7  )
8  mycursor = mydb.cursor()
9  # Modo 1 de consulta
10 # sql = "SELECT * FROM PERSONAS WHERE CEDULA = '1234'"
11 # mycursor.execute(sql)
12
13 # Modo 2 de consulta
14 sql = "SELECT * FROM PERSONAS WHERE CEDULA = %s"
15 variable = ('5678',)
16 mycursor.execute(sql,variable)
17
18 myresult = mycursor.fetchall() # devuelve todos
19 #myresult = mycursor.fetchone() # devuelve sólo el primero
20
21 for x in myresult:
22     print(x)
```

- Ejecutar y verificar

Python y MySQL

- Eliminación de registros: Editar el programa demo12_test_mysql.py

```
1 import mysql.connector
2 mydb = mysql.connector.connect(
3     host="localhost",
4     user="root",
5     passwd="",
6     database="BD_CURSO_PYTHON"
7 )
8 mycursor = mydb.cursor()
9
10 sql = "DELETE FROM PERSONAS WHERE CEDULA = '1234'"
11 mycursor.execute(sql)
12
13 mydb.commit() # toda actualización de la tabla debe llevar esta instrucción
14 print(mycursor.rowcount, "registro(s) eliminado(s)")
```

- Ejecutar y verificar

Python y MySQL

- Actualización de registros: Editar el programa demo13_test_mysql.py

```
1 import mysql.connector
2 mydb = mysql.connector.connect(
3     host="localhost",
4     user="root",
5     passwd="",
6     database="BD_CURSO_PYTHON"
7 )
8 mycursor = mydb.cursor()
9
10 sql = "UPDATE PERSONAS SET NOMBRE = 'SUSANA', APELLIDO = 'MENDEZ' WHERE CEDULA = '5678'"
11 mycursor.execute(sql)
12
13 mydb.commit() # toda actualización de la tabla debe llevar esta instrucción
14 print(mycursor.rowcount, "registro(s) actualizado(s)")
```

- Ejecutar y verificar

FIN