SQL to Jupyter Notebook Integration

```
conda install -c anaconda mysql-connector-python
mydb = mysql.connector.connect(host="localhost", user="root",passwd="password",
database= “classicmodels”)
print(mydb)
```

- The above example shows how to connect to a mysql server using python.
- The initial line of code will install the mysql python connection package which will help python connect to the sql workbench to access the database and make the queries.
- The connect() constructor creates a connection to the MySQL server and returns a MySQLConnection object.

Making queries to the SQL SERVER using python:

```
mycursor = mydb.cursor()

##If you had to create a new database: the below statement can be used
#mycursor.execute("CREATE DATABASE mydatabase")

mycursor = mydb.cursor()
sql = "SELECT * FROM customers;"
mycursor.execute(sql)

myresult = mycursor.fetchall()

for x in myresult:
  print(x)

print(type(x))
```

Code explanation for the example above:

- The cursor object helps perform functions like execute(), fetchall(), search(), close()
- The cursor.execute() executes a sql statement and retrieves data from the SQL Database -- which is stored in the variable mycursor.
- The variable "sql" has the sql query made to the database which is selecting all rows from the customer table.

- The fetchall() method fetches all (or all remaining) rows of the query result and returns a list of tuples which is stored in the variable "myresult".
- The for loop iterates over the tuple and the print statement prints the data.
- The print(type(x)) returns the data type of x which is TUPLE.