**Data Base Program using Python**

**Single Table**

**Reading data from an already created table in SQL Server**

import pyodbc

con = pyodbc.connect('Driver={SQL Server};'

'Server= LAPTOP-PQS0U4VH\SQLEXPRESS;'

'Database=a;'

'Trusted\_Connection=yes;')

cur = con.cursor()

cur.execute('select \* from Departments')

for row in cur:

print(row)

conn.commit()

conn.close() # close the connection established in the beginning using connect()

#################################

**Creating a table in SQL Server through Python Code**

import pyodbc

conn = pyodbc.connect('Driver={SQL Server};'

'Server=DESKTOP-KU0CRG6\SQLEXPRESS;'

'Database=a;'

'Trusted\_Connection=yes;')

cursor = conn.cursor()

cursor.execute('create table Books (Id INT PRIMARY KEY IDENTITY(10,3),Name VARCHAR (50) NOT NULL,Price INT)')

##In the example above, the starting value for IDENTITY is 1, and it will increment by 1 for each new record.

##To specify that the "Id" column should start at value 10 and increment by 5, change it to IDENTITY(10,5).

conn.commit()

conn.close()

#######################################

**Adding rows to the table in SQL Server through Python Code**

import pyodbc

conn = pyodbc.connect('Driver={SQL Server};'

'Server=DESKTOP-KU0CRG6\SQLEXPRESS;'

'Database=a;'

'Trusted\_Connection=yes;')

cur = conn.cursor()

# below there are 4 list

record\_1= ["Let Us C", 500]

record\_2= ["C# The Complete Reference", 800]

record\_3=["C++ OOPC", 600]

record\_4=["Java OOPC", 700]

record\_list = []

record\_list.append(record\_1)

record\_list.append(record\_2)

record\_list.append(record\_3)

record\_list.append(record\_4)

print(record\_list)

insert\_records = 'INSERT INTO Books(Name, Price) VALUES(?,?) '

cur.executemany(insert\_records, record\_list)

cur.execute('select \* from Books')

for row in cur:

print(row)

conn.commit()

conn.close()

##########################

* 'INSERT INTO Books(Name, Price) VALUES(?,?) '

The ? symbols are **placeholders** used in **parameterized queries**. They stand for **values that will be supplied later**, typically using a programming language like Python.

Here 2 question marks is for 2 columns in a row that is Name and Price

* execute() is used for **executing a single SQL statement**.
* executemany() is used for **executing the same SQL statement multiple times** with different parameters (usually for bulk inserts or updates), In our case for 4 rows of data stored in a list the **Insert** SQL statement will be executed 4 times for 4 records.

###################################

**Updating rows of the table in SQL Server through Python Code**

import pyodbc

conn = pyodbc.connect('Driver={SQL Server};'

'Server=DESKTOP-KU0CRG6\SQLEXPRESS;'

'Database=a;'

'Trusted\_Connection=yes;')

cursor = conn.cursor()

update\_query = 'UPDATE Books SET Price = 500 WHERE Id= 2'

cursor.execute(update\_query)

cursor.execute('select \* from Books')

for row in cursor:

print(row)

conn.commit()

conn.close()

###################################

**Deleting rows from the table in SQL Server through Python Code**

import pyodbc

conn = pyodbc.connect('Driver={SQL Server};'

'Server=DESKTOP-KU0CRG6\SQLEXPRESS;'

'Database=a;'

'Trusted\_Connection=yes;')

cursor = conn.cursor()

delete\_query = 'DELETE FROM Books WHERE Id= 3'

cursor.execute(delete\_query )

cursor.execute('select \* from Books')

for row in cursor:

print(row)

conn.commit()

conn.close()

###############################################