**Lab Exercise 2- MySQL with Python**

Here's a lab exercise that demonstrates how to connect to a MySQL database using Python:

**Step 1: Install the necessary packages**

Make sure you have the mysql-connector-python package installed. You can install it using pip:

pip install mysql-connector-python

**Step 2: Connect to the MySQL database**

import mysql.connector

# Connect to the MySQL server

mydb = mysql.connector.connect(

host="localhost",

user="root",

password="admin@123",

database="mydatabase"

)

**Step 3: Create a table and insert data**

# Create a table and insert data

mycursor = mydb.cursor()

mycursor.execute("CREATE TABLE customers (id INT AUTO\_INCREMENT PRIMARY KEY, name VARCHAR(255), address VARCHAR(255))")

sql = "INSERT INTO customers (name, address) VALUES (%s, %s)"

val = ("John Doe", "Highway 21")

mycursor.execute(sql, val)

mydb.commit()

print(mycursor.rowcount, "record inserted.")

**Step 4: Query the database**

# Query the database

mycursor.execute("SELECT \* FROM customers")

myresult = mycursor.fetchall()

for x in myresult:

print(x)

**Step 5: Update data in the table**

# Update data in the table

sql = "UPDATE customers SET address = 'Canyon 123' WHERE address = 'Highway 21'"

mycursor.execute(sql)

mydb.commit()

print(mycursor.rowcount, "record(s) affected")

**Step 6: Delete data from the table**

# Delete data from the table

sql = "DELETE FROM customers WHERE address = 'Canyon 123'"

mycursor.execute(sql)

mydb.commit()

print(mycursor.rowcount, "record(s) deleted")

**Step 7: Close the connection**

# Close the connection

mydb.close()

**Conclusion**

This exercise provides a basic demonstration of how to connect to a MySQL database, create a table, and perform CRUD operations using Python. Experiment further with different SQL commands and error handling to deepen your understanding of MySQL database operations with Python.