**SOFTWARE LABORATORY 1**

**GROUP A – EXPERIMENT 7**

**TITLE:**

Write a program to implement MySQL/Oracle database connectivity with any front end language to implement Database navigation operations (add, delete, edit etc.)

**CODE:**

Step 1: Create a Database and Table

mysql> CREATE DATABASE student\_db;

Query OK, 1 row affected (0.01 sec)

mysql> USE student\_db;

Database changed

mysql> CREATE TABLE students (id INT AUTO\_INCREMENT PRIMARY KEY, name VARCHAR(100), age INT, course VARCHAR(100));

Query OK, 0 rows affected (0.04 sec)

mysql> EXIT;

Step 2: Install Connector (Python ↔ MySQL)

pip install mysql-connector-python

Step 3: Write the Python Code

import mysql.connector

from tkinter import \*

from tkinter import messagebox

# ---------- Connect to MySQL ----------

def connect\_db():

return mysql.connector.connect(

host="localhost",

user="root", # change if you used another username

password="1234", # your MySQL password

database="student\_db"

)

# ---------- Functions ----------

def add\_record():

name = name\_entry.get()

age = age\_entry.get()

course = course\_entry.get()

if name == "" or age == "" or course == "":

messagebox.showwarning("Warning", "Please fill all fields")

return

conn = connect\_db()

cursor = conn.cursor()

cursor.execute("INSERT INTO students (name, age, course) VALUES (%s, %s, %s)", (name, age, course))

conn.commit()

conn.close()

messagebox.showinfo("Success", "Record added successfully")

clear\_fields()

view\_records()

def view\_records():

listbox.delete(0, END)

conn = connect\_db()

cursor = conn.cursor()

cursor.execute("SELECT \* FROM students")

rows = cursor.fetchall()

for row in rows:

listbox.insert(END, row)

conn.close()

def delete\_record():

try:

selected = listbox.get(listbox.curselection())

student\_id = selected[0]

conn = connect\_db()

cursor = conn.cursor()

cursor.execute("DELETE FROM students WHERE id=%s", (student\_id,))

conn.commit()

conn.close()

messagebox.showinfo("Deleted", "Record deleted successfully")

view\_records()

except:

messagebox.showwarning("Warning", "Please select a record to delete")

def edit\_record():

try:

selected = listbox.get(listbox.curselection())

student\_id = selected[0]

name = name\_entry.get()

age = age\_entry.get()

course = course\_entry.get()

conn = connect\_db()

cursor = conn.cursor()

cursor.execute("UPDATE students SET name=%s, age=%s, course=%s WHERE id=%s", (name, age, course, student\_id))

conn.commit()

conn.close()

messagebox.showinfo("Updated", "Record updated successfully")

view\_records()

except:

messagebox.showwarning("Warning", "Please select a record to update")

def clear\_fields():

name\_entry.delete(0, END)

age\_entry.delete(0, END)

course\_entry.delete(0, END)

# ---------- GUI ----------

root = Tk()

root.title("MySQL Database Navigation")

root.geometry("500x500")

Label(root, text="Name").pack()

name\_entry = Entry(root, width=40)

name\_entry.pack()

Label(root, text="Age").pack()

age\_entry = Entry(root, width=40)

age\_entry.pack()

Label(root, text="Course").pack()

course\_entry = Entry(root, width=40)

course\_entry.pack()

Button(root, text="Add Record", command=add\_record).pack(pady=5)

Button(root, text="View Records", command=view\_records).pack(pady=5)

Button(root, text="Edit Record", command=edit\_record).pack(pady=5)

Button(root, text="Delete Record", command=delete\_record).pack(pady=5)

Button(root, text="Clear Fields", command=clear\_fields).pack(pady=5)

listbox = Listbox(root, width=60)

listbox.pack(pady=10)

view\_records()

root.mainloop()

**Output:**

