NEW ERA PUBLIC SCHOOL

ACADEMIC YEAR :2021-2022

PROJECT REPORT ON SCHOOL MANAGMENT SYSTEM

- **Made By: Aryan Mishra**
  Jasjeev Singh
  Lalit Mohan
- **Class : XII-F**
- **Subject : Computer Science**
- **Subject code : 083**
- **Project guide : Mrs. Gurjeet Kaur**

CERTIFICATE

This is to certify that _____

_____

CBSE Roll no:_____

_____

has successfully completed the project work entitle “SCHOOL MANAGEMENT SYSTEM” in the subject Computer Science (083) laid down in the regulations of CBSE for the purpose of Practical examination in Class XII to be held in New Era Public School on _____

TABLE OF CONTENTS

ACKNOWLEDGEMENT

Apart from the efforts of me, the success of any project depends largely on the encouragement and guidelines of many others. I take this opportunity to express my gratitude to the people who have been instrumental in the successful completion of this project.

I express deep sense of gratitude to almighty God for giving me strength for the successful completion of the project.

I express my heartfelt gratitude to my parents for constant encouragement while carrying out this project.

I gratefully acknowledge the contribution of the individuals who contributed in bringing this project up tothis level, who continues to look after me despite my flaws,

I express my deep sense of gratitude to the luminary Vandana Chawla Principal, New Era Public School who has been continuously motivating and extending their helping hand to us.

My sincere thanks to Mrs. Gurjeet Kaur, Teacher In-charge, A guide, Mentor all the above a friend , who critically reviewed

my project and helped in solving each and every problem, occurred during implementation of the project

The guidance and support received from all the members who contributed and who are contributing to this project, was vital for the success of the project. I am grateful for their constant support and help

OBJECTIVE OF PROJECT

The objective of this project is to let the students apply the programming knowledge into a real- world situation/problem and exposed the students how programming skills helps in developing a good software.

1.Write programs utilizing modern software tools.

2.Apply object-oriented programming principles effectively when developing small to medium sized projects.

3. Write effective procedural code to solve small to Mediumsized problems

4. Students will demonstrate a breadth of knowledge in computer science, as exemplified in the areas of systems, theory and software development.

5. Students will demonstrate ability to conduct a research or applied Computer Science project, requiring writing and presentation skills which exemplify scholarly style

HARDWARE AND SOFTWARE REQUIREMENTS

I.OPERATING SYSTEM : WINDOWS 7 AND ABOVE

II. PROCESSOR : PENTIUM(ANY) OR AMD ATHALON(3800+- 4200+ DUALCORE)

III. MOTHERBOARD : 1.845 OR 915,995 FOR PENTIUM 0R MSI K9MM-V VIAK8M800+8237R PLUS CHIPSET FOR AMD ATHALON

IV. RAM : 512MB+

V. Hard disk : SATA 40 GB OR ABOVE

VI. CD/DVD r/w multi drive combo: (If back up required)

VII. FLOPPY DRIVE 1.44 MB : (If Backup required)

VIII. MONITOR 14.1 or 15 -17 inch

IX. Key board and mouse

X. Printer : (if print is required – [Hard copy])

SOFTWARE REQUIREMENTS:

I. Windows OS
II. Python
III. MySQL

CODE OF PROJECT

```
import sys
import datetime
from tkinter import *
from tkinter import ttk
import tkinter.font as tkFont
from PIL import ImageTk,Image
import time
import mysql.connector
import webbrowser
from tkinter import messagebox
from tkcalendar import DateEntry  # pip install tkcalendar

hereistheuser="root"
hereisthepassword="newera123"


def clicktoviewneps():
    webbrowser.open('https://newerapublicschool.in/')

#########################################3

def loginscreen():
    global login1
    login1 = Toplevel()
    login1.geometry("700x500")
    login1.title("LOGIN")
    login1.config(bg="#2bfcb3")
    global e
    global z
    global rx
    rx=IntVar()
    m = Label(login1, text="LOGIN",font=('Helvetica
bold',40,'bold'),fg="Red",bg="#2bfcb3").place(x=250,y=20)
    e1 = Label(login1,text="Username:",font=('lucida', 20,
'bold'),bg="#2bfcb3").place(x=80,y=150)
    z1 = Label(login1,text="Password:",font=('lucida', 20,
```

```
'bold'),bg="#2bfcb3").place(x=80,y=200)

    e = Entry(login1,font=('lucida', 20, 'bold'),fg="Black")
    e.place(x=245,y=150)

    z = Entry(login1,fg="Black", font=('lucida', 20),show='*')
    z.place(x=245,y=200)

    vft4 = Radiobutton(login1, text="Student", bg="#2bfcb3",font=('molot', 15),
variable=rx, value=1)
    vft4.place(x=245, y=250)
    vft5 = Radiobutton(login1, text="Teacher", bg="#2bfcb3",font=('molot', 15),
variable=rx, value=2)
    vft5.place(x=245, y=280)
    vft6 = Radiobutton(login1, text="Principal", bg="#2bfcb3",font=('molot', 15),
variable=rx, value=3)
    vft6.place(x=245, y=310)
    vft7 = Radiobutton(login1, text="Technical team",
bg="#2bfcb3",font=('molot', 15), variable=rx, value=4)
    vft7.place(x=245, y=340)

battu=Button(login1,text="Submit",padx=1,pady=1,bg="Red",fg="White",font=
('Aerial',20),command=logincheck).place(x=270,y=390)
    login1.mainloop()

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

def popup():
    messagebox.showwarning("WARNING", "The username is incorrect")

def popup1():
    messagebox.showwarning("WARNING", "The password is incorrect")

def popup2():
    messagebox.showwarning("WARNING", "The name is not matching")

def popup3():
```

```
    messagebox.showwarning("ATTENTION", "The record has been added")

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

def logincheck():
    lst = []
    global pl3444
    pl3444 = e.get()
    zl = z.get()
    mydb = mysql.connector.connect(host="localhost", user=hereistheuser,
password=hereisthepassword)
    mycursor = mydb.cursor()
    mycursor.execute("create database if not exists school;")
    mycursor.execute("use school;")
    mycursor.execute("create table if not exists login(Username varchar(30)
NOT NULL,Password varchar(15) NOT NULL,Name varchar(50) NOT
NULL,Class varchar(20) NOT NULL,Admissionnumber varchar(30) ,Logintype
varchar(30) NOT NULL,Gender varchar(1) NOT NULL);")
    mycursor.execute("select Username from login;")
    myresult = mycursor.fetchall()
    mydb.close()
    r = len(myresult)- 1
    for a in range(0, r + 1):
        n = myresult[a][0]
        lst.append(n)
    if pl3444 in lst:
        lst1 = []
        mydb = mysql.connector.connect(host="localhost", user=hereistheuser,
password=hereisthepassword)
        mycursor = mydb.cursor()
        mycursor.execute("use school;")
        mycursor.execute("select password from login where Username='" +
pl3444 + "';")
        myresult1 = mycursor.fetchall()
        mydb.close()
        for ty in myresult1:
```

```
            k = ty[0]
        if k == zl:
            mydb = mysql.connector.connect(host="localhost", user=hereistheuser,
password=hereisthepassword)
            mycursor = mydb.cursor()
            mycursor.execute("use school;")
            mycursor.execute("select Logintype from login where Username='" +
pl3444 + "';")
            myresult2 = mycursor.fetchall()
            mydb.close()
            nam = myresult2[0][0]
            if nam=='principal':
                login1.destroy()
                principaldatamainmenu()
            elif nam=='teacher':
                login1.destroy()
                teacherdatamainmenu()
            elif nam=='student':
                login1.destroy()
                studentdatamainmenu()
            elif nam=='technical team':
                login1.destroy()
                technicalteamdatamainmenu()
            else:
                popup()
        else:
            popup1()

    else:
        popup()

def logincheck2():
    lstw = []
    global yyyyytt
    yyyyytt = e2.get()
    rrtt3e = z2.get()
    mydb = mysql.connector.connect(host="localhost", user=hereistheuser,
```

```
password=hereisthepassword)
    mycursor = mydb.cursor()
    mycursor.execute("create database if not exists school;")
    mycursor.execute("use school;")
    mycursor.execute("create table if not exists login(Username varchar(30)
NOT NULL,Password varchar(15) NOT NULL,Name varchar(50) NOT
NULL,Class varchar(20) NOT NULL,Admissionnumber varchar(30) ,Logintype
varchar(30) NOT NULL,Gender varchar(1) NOT NULL);")
    mycursor.execute("select Username from login;")
    myresultss = mycursor.fetchall()
    mydb.close()
    r1 = len(myresultss) - 1
    for aw in range(0, r1 + 1):
        nz = myresultss[aw][0]
        lstw.append(nz)
    if yyyyytt in lstw:
        lst2 = []
        mydb = mysql.connector.connect(host="localhost", user=hereistheuser,
password=hereisthepassword)
        mycursor = mydb.cursor()
        mycursor.execute("use school;")
        mycursor.execute("select password from login where Username='" +
yyyyytt + "';")
        myresult1ss = mycursor.fetchall()
        mydb.close()
        for tyzz in myresult1ss:
            k34 = tyzz[0]
        if k34 == rrtt3e:
            mydb = mysql.connector.connect(host="localhost", user=hereistheuser,
password=hereisthepassword)
            mycursor = mydb.cursor()
            mycursor.execute("use school;")
            mycursor.execute("select Logintype from login where Username='" +
yyyyytt + "';")
            myre122 = mycursor.fetchall()
            mydb.close()
            if myre122[0][0] != 'technical team':
```

```
            e2.delete(0, END)
            z2.delete(0, END)
            popup1()
        elif myre122[0][0] == 'technical team':
            login2.destroy()
            createaccountfinal()
        else:
            print("Please contact user")
    else:
        popup1()

def createaccount():  # main
    global login2
    login2 = Tk()
    login2.config(bg="#2bfcb3")
    login2.geometry("700x500")
    login2.title("LOGIN")
    global e2
    global z2
    m2 = Label(login2, text="LOGIN TO CREATE ACCOUNT", font=('Helvetica
bold', 30,'bold'), fg="Red",bg="#2bfcb3",anchor=CENTER).place(x=59,y=20)
    e2 = Label(login2, text="Username:", font=('lucida', 20,
'bold'),bg="#2bfcb3").place(x=80, y=150)
    z2 = Label(login2, text="Password:", font=('lucida', 20,
'bold'),bg="#2bfcb3").place(x=80, y=200)
    e2 = Entry(login2, font=('lucida', 20, 'bold'), fg="Black")
    e2.place(x=245, y=150)
    z2 = Entry(login2, font=('lucida', 20, 'bold'), fg="Black", show='*')
    z2.place(x=245, y=200)

    battu2 = Button(login2, text="Submit", padx=1, pady=1, bg="Red",
fg="White", font=('Aerial', 20),command=logincheck2).place(x=270, y=270)

    login2.mainloop()

def createaccountfinal():
    createaccountf = Toplevel()
```

```
    createaccountf.geometry("700x600")
    createaccountf.title("CREATE ACCOUNT")
    createaccountf.config(bg="#ccb678")
    # print(myss2[-1][0])
    weer = Label(createaccountf, text="CREATE ACCOUNT", font=('Aerial',
30),bg="#ccb678").place(x=150, y=10)
    weer2 = Label(createaccountf, text="Name:", font=('lucida', 20,
'bold'),bg="#ccb678").place(x=120, y=145)
    weer3 = Label(createaccountf, text="Username:", font=('lucida', 20,
'bold'),bg="#ccb678").place(x=120, y=195)
    weer4 = Label(createaccountf, text="Password:", font=('lucida', 20,
'bold'),bg="#ccb678").place(x=120, y=245)
    weer5 = Label(createaccountf, text="Login type:", font=('lucida', 20,
'bold'),bg="#ccb678").place(x=120, y=295)

    global uiio1
    global uiio2
    global uiio3
    global rq
    rq = IntVar()
    uiio1 = Entry(createaccountf, font=('lucida', 20, 'bold'), fg="Black")
    uiio1.place(x=270, y=150)
    uiio2 = Entry(createaccountf, font=('lucida', 20, 'bold'), fg="Black")
    uiio2.place(x=270, y=200)
    uiio3 = Entry(createaccountf, font=('lucida', 20, 'bold'), fg="Black",show="*")
    uiio3.place(x=270, y=250)
    uiio4 = Radiobutton(createaccountf, text="Student",font=('molot',
15),bg="#ccb678" ,variable=rq, value=1)
    uiio4.place(x=270, y=300)
    uiio5 = Radiobutton(createaccountf, text="Teacher",font=('molot',
15),bg="#ccb678" ,variable=rq, value=2)
    uiio5.place(x=270, y=325)
    uiio6 = Radiobutton(createaccountf, text="Principal",font=('molot',
15),bg="#ccb678" ,variable=rq, value=3)
    uiio6.place(x=270, y=350)
    uiio7 = Radiobutton(createaccountf, text="Technical team",font=('molot',
15),bg="#ccb678" ,variable=rq, value=4)
```

```
    uiio7.place(x=270, y=375)

    ffddd = Button(createaccountf, text="SUBMIT", font=('Aerial', 15),
bg="Black", fg="White",command=clickcreateaccount).place(x=270, y=430)
    createaccountf.mainloop()

def clickcreateaccount():
    global qwe1
    global qwe4
    qwe1 = uiio1.get()
    qwe2 = uiio2.get()
    qwe3 = uiio3.get()
    qwe4 = rq.get()

    if qwe4 == 1:
        qwe4 = "student"
    elif qwe4 == 2:
        qwe4 = "teacher"
    elif qwe4 == 3:
        qwe4 = "principal"
    elif qwe4 == 4:
        qwe4 = "technical team"
    mydb = mysql.connector.connect(host="localhost", user=hereistheuser,
password=hereisthepassword)
    mycursor = mydb.cursor()
    mycursor.execute("use school;")
    mycursor.execute("insert into login values('" + qwe2 + "','" + qwe3 + "','" +
qwe4 + "');")
    mydb.commit()
    mydb.close()
    if qwe4 == 'student':
        global crtu
        crtu = Toplevel()
        crtu.geometry("400x300")
        crtu.title("ADD STUDENT DETAILS")
        crt1 = Label(crtu, text="Name:", font=('Aerial', 11)).place(x=10, y=45)
        crt2 = Label(crtu, text="Class:", font=('Aerial', 11)).place(x=10, y=95)
```

```
        crt3 = Label(crtu, text="Addmission No.", font=('Aerial', 11)).place(x=10,
y=145)
        crt4 = Label(crtu, text="Gender:", font=('Aerial', 11)).place(x=10, y=195)

        global crty1
        global crty2
        global crty3
        global crty4

        crty1 = Entry(crtu, width=40, borderwidth=2, fg="Black")
        crty1.place(x=120, y=50)
        crty2 = Entry(crtu, width=40, borderwidth=2, fg="Black")
        crty2.place(x=120, y=100)
        crty3 = Entry(crtu, width=40, borderwidth=2, fg="Black")
        crty3.place(x=120, y=150)
        crty4 = Entry(crtu, width=40, borderwidth=2, fg="Black")
        crty4.place(x=120, y=200)

        ffsert = Button(crtu, text="SUBMIT", font=('Aerial', 15), bg="Black",
fg="White", command=studentaddinfo).place(x=140, y=240)
        crtu.mainloop()

    elif qwe4 == 'teacher':
        global trea
        trea = Toplevel()
        trea.geometry("400x300")
        trea.title("ADD TEACHER DETAILS")
        tre1 = Label(trea, text="Name:", font=('Aerial', 11)).place(x=10, y=45)
        tre2 = Label(trea, text="Subject:", font=('Aerial', 11)).place(x=10, y=95)
        tre3 = Label(trea, text="Gender:", font=('Aerial', 11)).place(x=10, y=145)

        global tryl1
        global tryl2
        global tryl3

        tryl1 = Entry(trea, width=40, borderwidth=2, fg="Black")
        tryl1.place(x=120, y=50)
```

```
        tryl2 = Entry(trea, width=40, borderwidth=2, fg="Black")
        tryl2.place(x=120, y=100)
        tryl3 = Entry(trea, width=40, borderwidth=2, fg="Black")
        tryl3.place(x=120, y=150)

        ffsert = Button(trea, text="SUBMIT", font=('Aerial', 15), bg="Black",
fg="White", command=teacheraddinfo).place(x=140, y=240)
        trea.mainloop()

def teacheraddinfo():
    trwq1 = str(tryl1.get())
    trwq2 = str(tryl2.get())
    trwq3 = str(tryl3.get())
    if qwe1 == trwq1:
        mydb = mysql.connector.connect(host="localhost", user=hereistheuser,
password=hereisthepassword)
        mycursor = mydb.cursor()
        mycursor.execute("use school;")
        mycursor.execute("insert into loginteacher values('" + str(uiio2.get()) +
"','" + str(
            uiio3.get()) + "','" + trwq1 + "','" + trwq2 + "','" + qwe4 + "','" + trwq3 +
"');")
        mydb.commit()
        mydb.close()
        popup3()
        trea.destroy()
    else:
        popup2()

def studentaddinfo():
    crti1 = str(crty1.get())
    crti2 = str(crty2.get())
    crti3 = str(crty3.get())
    crti4 = str(crty4.get())

    if qwe1 == crti1:
        mydb = mysql.connector.connect(host="localhost", user=hereistheuser,
```

```
password=hereisthepassword)
        mycursor = mydb.cursor()
        mycursor.execute("use school;")
        mycursor.execute("insert into loginstudent values('" + str(uiio2.get()) +
"','" + str(
            uiio3.get()) + "','" + crti1 + "','" + crti2 + "','" + crti3 + "','" + qwe4 + "','" +
crti4 + "');")
        mydb.commit()
        mydb.close()
        popup3()
        crtu.destroy()
    else:
        popup2()

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

def student_marksshow():
    rootmarsho = Toplevel()
    rootmarsho.geometry("1150x600")
    rootmarsho.title("MySQLProject")
    frame = Frame(rootmarsho, bd=4, bg='cyan')
    frame.place(width=1150, height=600)

    con = mysql.connector.connect(host='localhost', user=hereistheuser,
passwd=hereisthepassword)
    cursor = con.cursor()
    cursor.execute("create database if not exists school;")
    cursor.execute("use school;")
    cursor.execute("Select * from Student_Marksheet where Student_Name='" +
resu1 +"';")
    records = cursor.fetchall()

    label = Label(frame, text="MARKSHEET", font=('Times New Roman', 40,
'bold'), fg='black', bg='white')
    label.pack(side=TOP, fill=X)
```

```
    my_tree = ttk.Treeview(frame)

    style = ttk.Style()
    style.theme_use("default")
    style.configure("Treeview", rowheight=35,
                    foreground='black')

    my_tree['columns'] = (
    "Student's Name", "Physics", "Chemistry", "Mathematics", "English",
"Computer Science", "Total Marks")
    my_tree.column('#0', anchor=W, width=0, stretch=NO)
    my_tree.column("Student's Name", anchor=W, width=100)
    my_tree.column("Physics", anchor=CENTER, width=80)
    my_tree.column("Chemistry", anchor=W, width=80)
    my_tree.column("Mathematics", anchor=W, width=80)
    my_tree.column("English", anchor=W, width=80)
    my_tree.column("Computer Science", anchor=W, width=120)
    my_tree.column("Total Marks", anchor=W, width=120)

    my_tree.heading('#0', text="", anchor=W)
    my_tree.heading("Student's Name", text="Student's Name", anchor=W)
    my_tree.heading("Physics", text="Physics", anchor=CENTER)
    my_tree.heading("Chemistry", text="Chemistry", anchor=W)
    my_tree.heading("Mathematics", text="Mathematics", anchor=W)
    my_tree.heading("English", text="English", anchor=W)
    my_tree.heading("Computer Science", text="Computer Science", anchor=W)
    my_tree.heading("Total Marks", text="Total Marks", anchor=W)

    for i in range(len(records)):
        if i % 2 == 0:
            my_tree.insert(parent='', index='end', iid=i, text='', values=(records[i][0],
                                                    records[i][1], records[i][2],
records[i][3],
                                                    records[i][4]
                                                    , records[i][5], records[i][6]),
                           tags=("evenrow",))
        else:
```

```
        my_tree.insert(parent='', index='end', iid=i, text='', values=(records[i][0],
                                                records[i][1], records[i][2],
records[i][3],
                                                records[i][4]
                                                , records[i][5], records[i][6]),
                       tags=("oddrow",))

    my_tree.pack()
    btn = Button(frame, text="Exit", padx=20,pady=5,font=('lucida', 20,
'bold'),bg="White",fg="Red", command=rootmarsho.destroy)

    btn.pack(side=BOTTOM)
    rootmarsho.mainloop()

def student_homeworkshow():
    rootstuhm = Toplevel()
    rootstuhm.geometry("1150x600")
    rootstuhm.title("MySQLProject")

    con = mysql.connector.connect(host='localhost', user=hereistheuser,
passwd=hereisthepassword)
    cursor = con.cursor()
    cursor.execute("create database if not exists school;")
    cursor.execute("use school;")
    cursor.execute("Select * from Student_HW")
    records = cursor.fetchall()

    frame = Frame(rootstuhm, bd=4, bg="cyan")

    frame.place(width=1150, height=600)
    label = Label(frame, text="Student's Homework\n", font=('Times New
Roman', 40, 'bold'), fg='black', bg='white')
    label.pack(side=TOP, fill=X)
    my_tree = ttk.Treeview(frame)

    style = ttk.Style()
    style.theme_use("default")
```

```
    style.configure("Treeview", rowheight=35,
                    foreground='black')

    my_tree.tag_configure('oddrow', background='white')
    my_tree.tag_configure('evenrow', background='lightblue')

    # TreeColumns
    my_tree['columns'] = ("Subject Code", "Subject", "Date", 'Homework')

    my_tree.column('#0', anchor=W, width=0, stretch=NO)
    my_tree.column("Subject Code", anchor=W, width=80)
    my_tree.column("Subject", anchor=W, width=80)
    my_tree.column("Date", anchor=CENTER, width=80)
    my_tree.column("Homework", anchor=W, width=580)

    my_tree.heading('#0', text="", anchor=W)
    my_tree.heading("Subject Code", text="Subject_Code", anchor=W)
    my_tree.heading("Subject", text="Subject", anchor=W)
    my_tree.heading("Date", text="Date", anchor=CENTER)
    my_tree.heading("Homework", text="Homework", anchor=W)

    for i in range(len(records)):
        if i % 2 == 0:
            my_tree.insert(parent='', index='end', iid=i, text='', values=(records[i][0],
                                                    records[i][1], records[i][2],
records[i][3]),
                           tags=("evenrow",))
        else:
            my_tree.insert(parent='', index='end', iid=i, text='', values=(records[i][0],
                                                    records[i][1], records[i][2],
records[i][3]),
                           tags=("oddrow",))

    my_tree.pack()
    btn = Button(frame, text="Exit",  padx=20,pady=5,font=('lucida', 20,
'bold'),bg="White",fg="Red", command=rootstuhm.destroy)
```

```
    btn.pack(side=BOTTOM)
    rootstuhm.mainloop()

def student_attendanceshow():
    rootwert = Tk()
    rootwert.geometry("1150x600")
    rootwert.title("MySQLProject")
    frame = Frame(rootwert, bd=4, bg='cyan')
    frame.place(width=1150, height=600)

    con = mysql.connector.connect(host='localhost', user=hereistheuser,
passwd=hereisthepassword)
    cursor = con.cursor()
    cursor.execute("create database if not exists school;")
    cursor.execute("use school;")
    cursor.execute('''Create table if not exists
Student_Attendance(Admission_No varchar(10),
                    Student_Name varchar(50),
                    Date_of_Attendance varchar(40),Attendance char(1))''')
    cursor.execute("Select * from Student_Attendance where Student_Name='" +
resu1 + "';")
    records = cursor.fetchall()

    label = Label(frame, text="ATTENDANCE", font=('Times New Roman', 40,
'bold'), fg='black', bg='white')
    label.pack(side=TOP, fill=X)

    my_tree = ttk.Treeview(frame)

    style = ttk.Style()
    style.theme_use("default")
    style.configure("Treeview", rowheight=35,
                    foreground='black')

    my_tree['columns'] = ("Admission No.", "Student's Name", "Date",
'Attendance(P/A)')
```

```
    my_tree.column('#0', anchor=W, width=0, stretch=NO)
    my_tree.column("Admission No.", anchor=W, width=80)
    my_tree.column("Student's Name", anchor=W, width=80)
    my_tree.column("Date", anchor=CENTER, width=80)
    my_tree.column("Attendance(P/A)", anchor=W, width=580)

    my_tree.heading('#0', text="", anchor=W)
    my_tree.heading("Admission No.", text="Admission No.", anchor=W)
    my_tree.heading("Student's Name", text="Student's Name", anchor=W)
    my_tree.heading("Date", text="Date", anchor=CENTER)
    my_tree.heading("Attendance(P/A)", text="Attendance(P/A)", anchor=W)

    for i in range(len(records)):
        if i % 2 == 0:
            my_tree.insert(parent='', index='end', iid=i, text='', values=(records[i][0],
                                                    records[i][1], records[i][2],
records[i][3]),
                       tags=("evenrow",))
        else:
            my_tree.insert(parent='', index='end', iid=i, text='', values=(records[i][0],
                                                    records[i][1], records[i][2],
records[i][3]),
                       tags=("oddrow",))

    my_tree.pack()
    btn = Button(frame, text="Exit", padx=20,pady=5,font=('lucida', 20,
'bold'),bg="White",fg="Red", command=rootwert.destroy)

    btn.pack(side=BOTTOM)
    rootwert.mainloop()

def student_noticeshow():
    ws = Tk()
    ws.title('Notice')
    ws.geometry('725x510')

    f = open("notice.txt", "rt")
```

```
    message = ""
    for x in f:
        message += x
    f.close()
    text_box = Text(ws, height=24, width=80)
    text_box.pack(expand=True)
    text_box.insert('end', message)
    text_box.config(state='disabled')

    ws.mainloop()

def student_competitionshow():
    root = Toplevel()
    root.geometry("1150x600")
    root.title("MySQLProject")

    con = mysql.connector.connect(host='localhost', user=hereistheuser,
passwd=hereisthepassword)
    cursor = con.cursor()
    cursor.execute("Create database if not exists school")
    cursor.execute("use school")
    cursor.execute("Select * from Student_competition")
    records = cursor.fetchall()

    frame = Frame(root, bd=4, bg="cyan")

    frame.place(width=1150, height=600)
    label = Label(frame, text="Student's Competition\n", font=('Times New
Roman', 40, 'bold'), fg='black', bg='white')
    label.pack(side=TOP, fill=X)
    my_tree = ttk.Treeview(frame)

    style = ttk.Style()
    style.theme_use("default")
    style.configure("Treeview", rowheight=35,
                    foreground='black')
```

```
    my_tree.tag_configure('oddrow', background='white')
    my_tree.tag_configure('evenrow', background='lightblue')

    # TreeColumns
    my_tree['columns'] = ("Hosted By", "Name of Comp", "Activity", 'Details')

    my_tree.column('#0', anchor=W, width=0, stretch=NO)
    my_tree.column("Hosted By", anchor=CENTER, width=80)
    my_tree.column("Name of Comp", anchor=CENTER, width=80)
    my_tree.column("Activity", anchor=CENTER, width=80)
    my_tree.column("Details", anchor=CENTER, width=580)

    my_tree.heading('#0', text="", anchor=W)
    my_tree.heading("Hosted By", text="Hosted By", anchor=CENTER)
    my_tree.heading("Name of Comp", text="Name of Comp", anchor=CENTER)
    my_tree.heading("Activity", text="Activity", anchor=CENTER)
    my_tree.heading("Details", text="Details", anchor=CENTER)

    for i in range(len(records)):
        if i % 2 == 0:
            my_tree.insert(parent='', index='end', iid=i, text='', values=(records[i][0],
                                                    records[i][1], records[i][2],
records[i][3]),
                       tags=("evenrow",))
        else:
            my_tree.insert(parent='', index='end', iid=i, text='', values=(records[i][0],
                                                    records[i][1], records[i][2],
records[i][3]),
                       tags=("oddrow",))

    my_tree.pack()
    btn = Button(frame, text="Exit",  padx=20,pady=5,font=('lucida', 20,
'bold'),bg="White",fg="Red", command=root.destroy)

    btn.pack(side=BOTTOM)
    root.mainloop()
```

```
def student_reportshow():
    rootmarsho = Toplevel()
    rootmarsho.geometry("1350x700")
    rootmarsho.title("MySQLProject")
    frame = Frame(rootmarsho, bd=4, bg='cyan')
    frame.place(width=1350, height=700)

    con = mysql.connector.connect(host='localhost', user=hereistheuser,
passwd=hereisthepassword)
    cursor = con.cursor()
    cursor.execute("create database if not exists school;")
    cursor.execute("use school;")
    cursor.execute('''Create table if not exists student_report(Student_Name
varchar(20),Admission_No int(10),
                    Physics_marks int(3),
                    Chemistry_marks int(3),Mathematics_marks
int(3),English_marks int(3),Cs_marks int(3),
                    TotalMarksObtained_Outof500 int(3))''')
    cursor.execute("Select * from student_report where Student_name='" +
resu1 + "';")
    records = cursor.fetchall()
    con.close()
    for callq in records:
        name=callq[0]
        adm=callq[1]
        phy=callq[2]
        chem=callq[3]
        Math=callq[4]
        Eng=callq[5]
        Cs=callq[6]
        Total=callq[7]


    label = Label(frame, text="REPORT CARD", font=('Times New Roman', 40,
'bold'), fg='black', bg='Red')
    label.pack(side=TOP, fill=X)
    label1=Label(frame,text="NAME:", font=('Times New Roman', 30, 'bold'),
```

```
fg='black',bg="cyan").place(x=100,y=100)
   label2=Label(frame,text=name , font=('Times New Roman', 30, 'bold'),
fg='Red',bg="cyan").place(x=260,y=100)
   label3=Label(frame,text="ADMISSION NO:", font=('Times New Roman', 30,
'bold'), fg='black',bg="cyan").place(x=760,y=100)
   label4=Label(frame,text=adm , font=('Times New Roman', 30, 'bold'),
fg='Red',bg="cyan").place(x=1090,y=100)
   label5=Label(frame,text='-'*700,font=('Times New Roman', 10,
'bold'),bg="cyan").place(x=-1,y=150)
   label6=Label(frame,text="PHYSICS MARKS:", font=('Times New Roman', 30,
'bold'), fg='black',bg="cyan").place(x=100,y=200)
   label7=Label(frame,text="CHEMISTRY MARKS:", font=('Times New Roman',
30, 'bold'), fg='black',bg="cyan").place(x=100,y=270)
   label8=Label(frame,text="MATHS MARKS:", font=('Times New Roman', 30,
'bold'), fg='black',bg="cyan").place(x=100,y=340)
   label9=Label(frame,text="ENGLISH MARKS:", font=('Times New Roman', 30,
'bold'), fg='black',bg="cyan").place(x=100,y=410)
   label10=Label(frame,text="COMPUTERS MARKS:", font=('Times New
Roman', 30, 'bold'), fg='black',bg="cyan").place(x=100,y=480)
   label11=Label(frame,text="TOTAL MARKS(OUT OF 500):", font=('Times
New Roman', 30, 'bold'), fg='black',bg="cyan").place(x=100,y=550)
   label12=Label(frame,text=phy , font=('Times New Roman', 30, 'bold'),
fg='Red',bg="cyan").place(x=690,y=200)
   label13=Label(frame,text=chem , font=('Times New Roman', 30, 'bold'),
fg='Red',bg="cyan").place(x=690,y=270)
   label14=Label(frame,text=Math , font=('Times New Roman', 30, 'bold'),
fg='Red',bg="cyan").place(x=690,y=340)
   label15=Label(frame,text=Eng , font=('Times New Roman', 30, 'bold'),
fg='Red',bg="cyan").place(x=690,y=410)
   label16=Label(frame,text=Cs , font=('Times New Roman', 30, 'bold'),
fg='Red',bg="cyan").place(x=690,y=480)
   label17=Label(frame,text=Total , font=('Times New Roman', 30, 'bold'),
fg='Red',bg="cyan").place(x=690,y=550)
```

```
    btn = Button(frame, text="Exit",  padx=20,pady=5,font=('lucida', 20,
'bold'),bg="White",fg="Red", command=rootmarsho.destroy)

    btn.pack(side=BOTTOM)
    rootmarsho.mainloop()

def teacher_reportwrite():
    root = Tk()
    root.geometry("1150x600")
    root.title("MySQLProject")
    frame = Frame(root, bd=4, bg='cyan')
    frame.place(width=1150, height=600)

    label = Label(frame, text="REPORT CARD RECORDS", font=('Times New
Roman', 40, 'bold'), fg='black', bg='white')
    label.pack(side=TOP, fill=X)

    con = mysql.connector.connect(host='localhost', user=hereistheuser,
passwd=hereisthepassword)
    cursor = con.cursor()
    cursor.execute("create database if not exists school;")
    cursor.execute("use school;")
    cursor.execute('''Create table if not exists student_report(Student_Name
varchar(20),Admission_No int(10),
                    Physics_marks int(3),
                    Chemistry_marks int(3),Mathematics_marks int(3),English_marks
int(3),Cs_marks int(3),
                    TotalMarksObtained_Outof500 int(3))''')

    st_label = Label(frame, text="Student's Name", font=('lucida', 20, 'bold'),
fg="black", bg="cyan")
    marks_label = Label(frame, text='MARKSHEET', font=('lucida', 20, 'bold'),
fg="black", bg="yellow")
    m0_label = Label(frame, text='Admission No.', font=('lucida', 20, 'bold'),
fg="black", bg="cyan")
    m1_label = Label(frame, text='Physics', font=('lucida', 20, 'bold'), fg="black",
bg="cyan")
```

```
    m2_label = Label(frame, text='Chemistry', font=('lucida', 20, 'bold'),
fg="black", bg="cyan")
    m3_label = Label(frame, text='Mathematics', font=('lucida', 20, 'bold'),
fg="black", bg="cyan")
    m4_label = Label(frame, text='English', font=('lucida', 20, 'bold'), fg="black",
bg="cyan")
    m5_label = Label(frame, text='Computer Science', font=('lucida', 20, 'bold'),
fg="black", bg="cyan")

    st_entry = Entry(frame, width=30, font=('lucida', 20, 'bold'), fg="black")
    m0_entry = Entry(frame, width=30, font=('lucida', 20, 'bold'), fg="black")
    m1_entry = Entry(frame, width=30, font=('lucida', 20, 'bold'), fg="black")
    m2_entry = Entry(frame, width=30, font=('lucida', 20, 'bold'), fg="black")
    m3_entry = Entry(frame, width=30, font=('lucida', 20, 'bold'), fg="black")
    m4_entry = Entry(frame, width=30, font=('lucida', 20, 'bold'), fg="black")
    m5_entry = Entry(frame, width=30, font=('lucida', 20, 'bold'), fg="black")

    st_label.place(x=50, y=100, anchor=W)
    marks_label.place(x=50, y=150, anchor=W)
    m0_label.place(x=50, y=200, anchor=W)
    m1_label.place(x=50, y=250, anchor=W)
    m2_label.place(x=50, y=300, anchor=W)
    m3_label.place(x=50, y=350, anchor=W)
    m4_label.place(x=50, y=400, anchor=W)
    m5_label.place(x=50, y=450, anchor=W)

    st_entry.place(x=300, y=100, anchor=W)
    m0_entry.place(x=300, y=200, anchor=W)
    m1_entry.place(x=300, y=250, anchor=W)
    m2_entry.place(x=300, y=300, anchor=W)
    m3_entry.place(x=300, y=350, anchor=W)
    m4_entry.place(x=300, y=400, anchor=W)
    m5_entry.place(x=300, y=450, anchor=W)

    def submit():
        cursor.execute("insert into student_report values('{}','{}',{},{},{},{},{},{})"
                       .format(st_entry.get(),m0_entry.get(), m1_entry.get(),
```

```
m2_entry.get(),
                        m3_entry.get(), m4_entry.get(), m5_entry.get(),
int(m1_entry.get()) + int(m2_entry.get())
                        + int(m3_entry.get()) +
                        int(m4_entry.get()) + int(m5_entry.get())))

        con.commit()

        st_entry.delete(0, END)
        m0_entry.delete(0, END)
        m1_entry.delete(0, END)
        m2_entry.delete(0, END)
        m3_entry.delete(0, END)
        m4_entry.delete(0, END)
        m5_entry.delete(0, END)

        messagebox.showinfo('Records Inserted:', 'Records have been succesfully
inserted in the Table')

    # Button to enter records
    btn = Button(frame, text="Submit",padx=20,pady=5,font=('lucida', 20,
'bold'),bg="White",fg="Red", command=submit)
    btn2 = Button(frame, text="Exit", padx=20,pady=5,font=('lucida', 20,
'bold'),bg="White",fg="Red", command=root.destroy)

    btn.place(x=50, y=520, anchor=W)
    btn2.place(x=290, y=520, anchor=W)

    root.mainloop()

def teacher_competitionwrite():
    win = Toplevel()
    win.geometry("1150x600")
    win.title("MYSQLProject")
    frame = Frame(win, bd=4, bg='cyan')
    frame.place(width=1150, height=600)
```

```
    mlabel = Label(frame, text="Records", font=('lucida', 40, 'bold'), fg='black',
bg='white')
    mlabel.pack(side=TOP, fill=X)

    con = mysql.connector.connect(host='localhost', user=hereistheuser,
passwd=hereisthepassword)
    cursor = con.cursor()
    cursor.execute("Create database if not exists school")
    cursor.execute("use school")
    cursor.execute('''Create table if not exists Student_competition(hosted_by
varchar(20) Primary Key,
                Name_of_competition varchar(20),Activity varchar(20),Details
varchar(50) )''')

    host_label = Label(frame, text='Hosted By:', font=('lucida', 20, 'bold'),
fg="black")
    comp_label = Label(frame, text='Name of Comp.', font=('lucida', 20, 'bold'),
fg="black")
    acti_label = Label(frame, text='Activity', font=('lucida', 20, 'bold'), fg="black")
    det_label = Label(frame, text='Details', font=('lucida', 20, 'bold'), fg="black")

    host_entry = Entry(frame, width=30, font=('lucida', 20, 'bold'), fg="black")
    comp_entry = Entry(frame, width=30, font=('lucida', 20, 'bold'), fg="black")
    acti_entry = Entry(frame, width=30, font=('lucida', 20, 'bold'), fg="black")
    det_entry = Text(frame, width=35, height=5)

    host_label.place(x=80, y=100, anchor=W)
    comp_label.place(x=80, y=150, anchor=W)
    acti_label.place(x=80, y=200, anchor=W)
    det_label.place(x=80, y=270, anchor=W)

    host_entry.place(x=300, y=100, anchor=W)
    comp_entry.place(x=300, y=150, anchor=W)
    acti_entry.place(x=300, y=200, anchor=W)
    det_entry.place(x=300, y=280, anchor=W)

    def submit1():
```

```
    if not host_entry.get() or not comp_entry.get() or not acti_entry.get() or not
det_entry.get(1.0, END):
        messagebox.showerror('Error!', "Please fill all the missing fields!!")

    else:
        try:
            cursor.execute("insert into Student_competition
values('{}','{}','{}','{}')"
                           .format(host_entry.get(), comp_entry.get(), acti_entry.get(),
                                   det_entry.get(1.0, END)))

            con.commit()

            host_entry.delete(0, END)
            comp_entry.delete(0, END)
            acti_entry.delete(0, END)
            det_entry.delete(1.0, END)

            messagebox.showinfo('Records Inserted:', 'Records have been
succesfully inserted in the Table')

            # Button to enter records

        except:
            messagebox.showerror('Wrong type',
                                 'The type of the values entered is not accurate.')

  btn = Button(frame, text="Submit", padx=20, pady=5, font=('lucida', 20,
'bold'), bg="White", fg="Red",
               command=submit1)
  btn2 = Button(frame, text="Exit", padx=20, pady=5, font=('lucida', 20, 'bold'),
bg="White", fg="Red",
               command=win.destroy)

  btn.place(x=50, y=520, anchor=W)
  btn2.place(x=290, y=520, anchor=W)
```

```
    win.mainloop()

def teacher_thoughtwrite():
    if __name__ == '__main__':
        # Basic tkinter setup
        root = Tk()
        root.title("Thought")
        # root.wm_iconbitmap("1.ico")
        root.geometry("525x310")
        root.maxsize(525, 310)
        root.minsize(525, 310)

        def saveFile():
            global file
            try:
                f = open("thought.txt", "x")
            except:
                pass
            with open("thought.txt", "w+") as f:
                # Move read cursor to the start of file.
                f.seek(0)
                # If file is not empty then append '\n'
                data = f.read(100)
                if len(data) > 0:
                    f.write("\n")
                # Append text at the end of file
                f.write("~")
                f.write(TextArea.get(1.0, END))

                messagebox.askokcancel("Thought", "Thought Saved.")
                root.destroy()
            f.close()

        TextArea = Text(root, font="lucida 13", height=13, width=58)
        file = None
        TextArea.pack()
        TextArea.place(x=1, y=1)
```

```
    # button to upload notice
    my_button1 = Button(root, text="Save", font="lucida", padx=30, pady=3,
bg="white", fg="blue",
                        command=saveFile).place(x=200, y=265)
    # buuton ends here
    root.mainloop()

def teacher_markswrite():
   rootopio = Toplevel()
   rootopio.geometry("1150x600")
   rootopio.title("MySQLProject")
   frame = Frame(rootopio, bd=4, bg='cyan')
   frame.place(width=1150, height=600)

   label = Label(frame, text="MARKSHEET RECORDS", font=('Times New
Roman', 40, 'bold'), fg='black', bg='white')
   label.pack(side=TOP, fill=X)

   con = mysql.connector.connect(host='localhost', user=hereistheuser,
passwd=hereisthepassword)
   cursor = con.cursor()
   cursor.execute("create database if not exists school;")
   cursor.execute("use school;")
   cursor.execute('''Create table if not exists Student_Marksheet(Student_Name
varchar(50),
                Physics_marks int(3),
                Chemistry_marks int(3),Mathematics_marks int(3),English_marks
int(3),Cs_marks int(3),
                TotalMarksObtained_Outof500 int(3))''')

   st_label = Label(frame, text="Student's Name", font=('lucida', 20, 'bold'),
fg="black", bg="cyan")
   marks_label = Label(frame, text='MARKSHEET', font=('lucida', 20, 'bold'),
fg="black", bg="yellow")
   m1_label = Label(frame, text='Physics', font=('lucida', 20, 'bold'), fg="black",
bg="cyan")
   m2_label = Label(frame, text='Chemistry', font=('lucida', 20, 'bold'),
```

```
fg="black", bg="cyan")
   m3_label = Label(frame, text='Mathematics', font=('lucida', 20, 'bold'),
fg="black", bg="cyan")
   m4_label = Label(frame, text='English', font=('lucida', 20, 'bold'), fg="black",
bg="cyan")
   m5_label = Label(frame, text='Computer Science', font=('lucida', 20, 'bold'),
fg="black", bg="cyan")

   st_entry = Entry(frame, width=30, font=('lucida', 20, 'bold'), fg="black")
   m1_entry = Entry(frame, width=30, font=('lucida', 20, 'bold'), fg="black")
   m2_entry = Entry(frame, width=30, font=('lucida', 20, 'bold'), fg="black")
   m3_entry = Entry(frame, width=30, font=('lucida', 20, 'bold'), fg="black")
   m4_entry = Entry(frame, width=30, font=('lucida', 20, 'bold'), fg="black")
   m5_entry = Entry(frame, width=30, font=('lucida', 20, 'bold'), fg="black")

   st_label.place(x=50, y=100, anchor=W)
   marks_label.place(x=50, y=150, anchor=W)
   m1_label.place(x=50, y=200, anchor=W)
   m2_label.place(x=50, y=250, anchor=W)
   m3_label.place(x=50, y=300, anchor=W)
   m4_label.place(x=50, y=350, anchor=W)
   m5_label.place(x=50, y=400, anchor=W)

   st_entry.place(x=300, y=100, anchor=W)
   m1_entry.place(x=300, y=200, anchor=W)
   m2_entry.place(x=300, y=250, anchor=W)
   m3_entry.place(x=300, y=300, anchor=W)
   m4_entry.place(x=300, y=350, anchor=W)
   m5_entry.place(x=300, y=400, anchor=W)

   def submit():
      if not st_entry.get() or not m1_entry.get() or not m2_entry.get() or not
m3_entry.get() or not m4_entry.get() or not m5_entry.get():
         messagebox.showerror('Error!', "Please fill all the missing fields!!")
      else:
         try:
            cursor.execute("insert into Student_Marksheet
```

```
values('{}',{},{},{},{},{},{})"
                        .format(st_entry.get(), m1_entry.get(), m2_entry.get(),
                                m3_entry.get(), m4_entry.get(), m5_entry.get(),
int(m1_entry.get()) + int(m2_entry.get())
                                + int(m3_entry.get()) +
                                int(m4_entry.get()) + int(m5_entry.get())))

            con.commit()

            st_entry.delete(0, END)
            m1_entry.delete(0, END)
            m2_entry.delete(0, END)
            m3_entry.delete(0, END)
            m4_entry.delete(0, END)
            m5_entry.delete(0, END)

            messagebox.showinfo('Records Inserted:', 'Records have been
succesfully inserted in the Table')
        except:
            messagebox.showerror('Wrong type',
                    'The type of the values entered is not accurate. Pls note that the
contact field can only contain numbers')

    # Button to enter records
    btn = Button(frame, text="Submit", padx=20, pady=5, font=('lucida', 20,
'bold'), bg="White", fg="Red",
                 command=submit)
    btn2 = Button(frame, text="Exit", padx=20, pady=5, font=('lucida', 20, 'bold'),
bg="White", fg="Red",
                  command=rootopio.destroy)

    btn.place(x=50, y=520, anchor=W)
    btn2.place(x=290, y=520, anchor=W)

    rootopio.mainloop()

def teacher_attendancewrite():
```

```
    roottyu = Toplevel()
    roottyu.geometry("1150x600")
    roottyu.title("MySQLProject")
    frame = Frame(roottyu, bd=4, bg='cyan')
    frame.place(width=1150, height=600)

    label = Label(frame, text="ATTENDANCE", font=('Times New Roman', 40,
'bold'), fg='black', bg='white')
    label.pack(side=TOP, fill=X)

    con = mysql.connector.connect(host='localhost', user=hereistheuser,
passwd=hereisthepassword)
    cursor = con.cursor()
    cursor.execute("create database if not exists school;")
    cursor.execute("use school;")
    cursor.execute('''Create table if not exists
Student_Attendance(Admission_No varchar(10),
                    Student_Name varchar(50),
                    Date_of_Attendance varchar(40),Attendance char(1))''')

    stid_label = Label(frame, text='Admission No: ', font=('lucida', 20, 'bold'),
fg="black", bg="cyan")
    st_label = Label(frame, text="Student's Name", font=('lucida', 20, 'bold'),
fg="black", bg="cyan")
    dt_label = Label(frame, text='Date', font=('lucida', 20, 'bold'), fg="black",
bg="cyan")
    at_label = Label(frame, text='Attendance(P/A)', font=('lucida', 20, 'bold'),
fg="black", bg="cyan")

    var = StringVar()
    var.set("P")

    stid_entry = Entry(frame, width=30, font=('lucida', 20, 'bold'), fg="black")
    st_entry = Entry(frame, width=30, font=('lucida', 20, 'bold'), fg="black")
    dt_entry = DateEntry(frame, font=("Arial", 12), width=15)
    at1_entry = Radiobutton(frame, text="Present", variable=var, value='P')
    at2_entry = Radiobutton(frame, text="Absent", variable=var, value='A')
```

```
    stid_label.place(x=50, y=100, anchor=W)
    st_label.place(x=50, y=150, anchor=W)
    dt_label.place(x=50, y=200, anchor=W)
    at_label.place(x=50, y=250, anchor=W)

    stid_entry.place(x=270, y=100, anchor=W)
    st_entry.place(x=270, y=150, anchor=W)
    dt_entry.place(x=270, y=200, anchor=W)
    at1_entry.place(x=270, y=260, anchor=W)
    at2_entry.place(x=270, y=300, anchor=W)

    def submit():
        if not stid_entry.get() or not st_entry.get() or not dt_entry.get_date() or not
var.get():
            messagebox.showerror('Error!', "Please fill all the missing fields!!")

        else:
            try:
                cursor.execute("insert into Student_Attendance values('{}','{}','{}','{}')"
                               .format(stid_entry.get(), st_entry.get(), dt_entry.get_date(),
                                       var.get()))

                con.commit()

                stid_entry.delete(0, END)
                st_entry.delete(0, END)
                dt_entry.set_date(datetime.datetime.now().date())

                messagebox.showinfo('Records Inserted:', 'Records have been
succesfully inserted in the Table')
            except:
                messagebox.showerror('Wrong type','The type of the values entered is
not accurate.')

    # Button to enter records
    btn = Button(frame, text="Submit", padx=20, pady=5, font=('lucida', 20,
```

```
'bold'), bg="White", fg="Red",
              command=submit)
   btn2 = Button(frame, text="Exit", padx=20, pady=5, font=('lucida', 20, 'bold'),
bg="White", fg="Red",
               command=roottyu.destroy)

   btn.place(x=50, y=520, anchor=W)
   btn2.place(x=290, y=520, anchor=W)

   roottyu.mainloop()

def teacher_homeworkwrite():
   wintoiis = Toplevel()
   wintoiis.geometry("1150x600")
   wintoiis.title("MYSQLProject")
   frame = Frame(wintoiis, bd=4, bg='cyan')
   frame.place(width=1150, height=600)

   mlabel = Label(frame, text="Records", font=('lucida', 40, 'bold'), fg='black',
bg='white')
   mlabel.pack(side=TOP, fill=X)

   con = mysql.connector.connect(host='localhost', user=hereistheuser,
passwd=hereisthepassword)
   cursor = con.cursor()
   cursor.execute("create database if not exists school;")
   cursor.execute("use school;")
   cursor.execute('''Create table if not exists Student_HW(Subject_Code
varchar(3) Primary Key,
               Subject varchar(100), Date date,Homework varchar(100) )''')

   subc_label = Label(frame, text='Subject Code: ', font=('lucida', 20, 'bold'),
fg="black")
   sub_label = Label(frame, text='Subject', font=('lucida', 20, 'bold'), fg="black")
   dt_label = Label(frame, text='Date', font=('lucida', 20, 'bold'), fg="black")
   hw_label = Label(frame, text='Homework', font=('lucida', 20, 'bold'),
fg="black")
```

```
    subc_entry = Entry(frame, width=30, font=('lucida', 20, 'bold'), fg="black")
    sub_entry = Entry(frame, width=30, font=('lucida', 20, 'bold'), fg="black")
    dt_entry = DateEntry(frame, font=("Arial", 12), width=15)
    hw_entry = Text(frame, width=30, height=5)

    subc_label.place(x=50, y=100, anchor=W)
    sub_label.place(x=50, y=150, anchor=W)
    dt_label.place(x=50, y=200, anchor=W)
    hw_label.place(x=50, y=250, anchor=W)

    subc_entry.place(x=270, y=100, anchor=W)
    sub_entry.place(x=270, y=150, anchor=W)
    dt_entry.place(x=270, y=200, anchor=W)
    hw_entry.place(x=270, y=260, anchor=W)

    def submit():
        if not subc_entry.get() or not sub_entry.get() or not dt_entry.get_date() or
not hw_entry.get:
            messagebox.showerror('Error!', "Please fill all the missing fields!!")

        else:
            try:
                cursor.execute("insert into Student_HW values('{}','{}','{}','{}')"
                               .format(subc_entry.get(), sub_entry.get(), dt_entry.get_date(),
                                       hw_entry.get(1.0, END)))

                con.commit()

                subc_entry.delete(0, END)
                sub_entry.delete(0, END)
                dt_entry.set_date(datetime.datetime.now().date())
                hw_entry.delete(1.0, END)

                messagebox.showinfo('Records Inserted:', 'Records have been
succesfully inserted in the Table')
```

```
        except:
            messagebox.showerror('Wrong type','The type of the values entered is
not accurate.')

    # Button to enter records
    btn = Button(frame, text="Submit", padx=20, pady=5, font=('lucida', 20,
'bold'), bg="White", fg="Red",
                 command=submit)
    btn2 = Button(frame, text="Exit", padx=20, pady=5, font=('lucida', 20, 'bold'),
bg="White", fg="Red",
                  command=wintoiis.destroy)

    btn.place(x=50, y=520, anchor=W)
    btn2.place(x=290, y=520, anchor=W)
    wintoiis.mainloop()

def teacher_noticewrite():
    if __name__ == '__main__':
        # Basic tkinter setup
        rootnotla = Tk()
        rootnotla.title("Notice")
        # root.wm_iconbitmap("1.ico")
        rootnotla.geometry("725x510")
        rootnotla.maxsize(725, 510)
        rootnotla.minsize(725, 510)

        def saveFile():
            global file
            try:
                f = open("notice.txt", "x")
            except:
                pass
            with open("notice.txt", "w+") as f:
                # Move read cursor to the start of file.
                f.seek(0)
                # If file is not empty then append '\n'
                data = f.read(100)
```

```
            if len(data) > 0:
                f.write("\n")
            # Append text at the end of file
            f.write("~")
            f.write(TextArea.get(1.0, END))

            messagebox.askokcancel("Notice", "Notice Saved.")
            rootnotla.destroy()
        f.close()

    TextArea = Text(rootnotla, font="lucida 13",height=24,width=80)
    file = None
    TextArea.pack(expand=True)
    TextArea.place(x=1, y=1)
    # button to upload notice
    my_button1 = Button(rootnotla, text="Save", font="lucida", padx=30,
pady=3, bg="white", fg="blue",
                        command=saveFile).place(x=300, y=465)
    # buuton ends here
    rootnotla.mainloop()

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

def teacherdatamainmenu():
    teacher = Toplevel()
    teacher.geometry("1200x700")
    teacher.title("SCHOOL MANAGMENT SYSTEM")
    My_L8 = Label(teacher, text="YOUR DETAILS", font=('Aerial', 50)).place(x=0,
y=0)

    my_image1q =
ImageTk.PhotoImage(Image.open("Photos/entermarks.png"))
    my_image2q =
ImageTk.PhotoImage(Image.open("Photos/enternotice.png"))
    my_image3q =
ImageTk.PhotoImage(Image.open("Photos/enterreport.png"))
```

```
    my_image4q =
ImageTk.PhotoImage(Image.open("Photos/enterthought.png"))
    my_image5q =
ImageTk.PhotoImage(Image.open("Photos/enterattendance.png"))
    my_image6q =
ImageTk.PhotoImage(Image.open("Photos/entercompetiton.png"))
    my_image7q =
ImageTk.PhotoImage(Image.open("Photos/enterhomework.png"))

    f = open("thought.txt", "r")
    message = ""
    for x in f:
        message += x
    f.close()

    stud_butt1 = Button(teacher, image=my_image1q, bg="#000000",
width=240, height=180,command=teacher_markswrite).place(x=0, y=180)
    stud_butt2 = Button(teacher, image=my_image2q, bg="#96ff00", width=240,
height=180,command=teacher_noticewrite).place(x=0, y=0)
    stud_butt3 = Button(teacher, image=my_image3q, bg="#847b75",
width=240, height=180,command=teacher_reportwrite).place(x=240, y=0)
    stud_butt4 = Button(teacher, image=my_image4q, bg="#f07c2e", width=240,
height=180,command=teacher_thoughtwrite).place(x=480, y=0)
    stud_butt5 = Button(teacher, image=my_image5q, bg="#847b75",
width=240, height=180,command=teacher_attendancewrite).place(x=720,
y=0)
    stud_butt7 = Button(teacher, image=my_image7q, bg="#000000",
width=240, height=180,command=teacher_homeworkwrite).place(x=960,
y=180)
    stud_butt6 = Button(teacher, image=my_image6q, bg="#e4e1b0",
width=240, height=180,command=teacher_competitionwrite).place(x=960,
y=0)
    temesadf="Thought of the day:\n\t\t\t"+message
    stud_butt8 = Label(teacher,text=temesadf,font=('lucida', 30,
'bold'),fg="#5d6af0").place(x=0, y=480)

    mydb = mysql.connector.connect(host="localhost", user=hereistheuser,
```

```
password=hereisthepassword)
    mycursor = mydb.cursor()
    mycursor.execute("use school;")
    mycursor.execute("select Name,Subject,Gender from loginteacher where
Username='" + pl3444 + "';")
    aaaa44455 = mycursor.fetchall()
    aaaa = aaaa44455[0][0]
    aaaa1 = aaaa44455[0][1]
    aaaa2 = aaaa44455[0][2]
    if aaaa2 == 'M':
        my_image3ww = ImageTk.PhotoImage(Image.open("Photos/male
photo.jpg"))
        myLabel9900 = Label(teacher, image=my_image3ww).place(x=241,
y=186)
        mlaywwwl = Label(teacher, text=aaaa, font=('Aerial', 45)).place(x=455,
y=215)
        mlac222l = Label(teacher, text="Subject:", font=('Aerial', 45)).place(x=455,
y=290)
        mlaclasel = Label(teacher, text=aaaa1, font=('Aerial', 45)).place(x=691,
y=290)

    elif aaaa2 == 'F':
        my_image3ww = ImageTk.PhotoImage(Image.open("Photos/female
photo.jpg"))
        myLabel9900 = Label(teacher, image=my_image3ww).place(x=241,
y=186)
        mlaywwwl = Label(teacher, text=aaaa, font=('Aerial', 45)).place(x=455,
y=215)
        mlac222l = Label(teacher, text="Subject:", font=('Aerial', 45)).place(x=455,
y=290)
        mlaclasel = Label(teacher, text=aaaa1, font=('Aerial', 45)).place(x=691,
y=290)
    else:
        print("PLEASE CONTACT DEVELOPERS")
    mydb.close()
    teacher.mainloop()
```

```
def studentdatamainmenu():
    global resu1
    student=Toplevel()
    student.geometry("1200x706")
    My_Ladel898=Label(student,text="YOUR
DETAILS",font=('Aerial',50)).place(x=300,y=-5)

    my_image1q = ImageTk.PhotoImage(Image.open("Photos/home.png"))
    my_image2q = ImageTk.PhotoImage(Image.open("Photos/thought.png"))
    my_image3q =
ImageTk.PhotoImage(Image.open("Photos/competition.png"))
    my_image4q = ImageTk.PhotoImage(Image.open("Photos/attendance.png"))
    my_image5q = ImageTk.PhotoImage(Image.open("Photos/marks.jpg"))
    my_image6q = ImageTk.PhotoImage(Image.open("Photos/notice.png"))
    my_image7q = ImageTk.PhotoImage(Image.open("Photos/report card.jpg"))

    mydb = mysql.connector.connect(host="localhost", user=hereistheuser,
password=hereisthepassword)
    mycursor = mydb.cursor()
    mycursor.execute("use school;")
    mycursor.execute("select Gender,Name,Class,Admissionnumber from
loginstudent where Username='" + pl3444 + "';")
    resu44455 = mycursor.fetchall()
    resu = resu44455[0][0]
    resu1 = resu44455[0][1]
    resu2 = resu44455[0][2]
    resu3 = resu44455[0][3]
    if resu == 'M':
        my_image33444 = ImageTk.PhotoImage(Image.open("Photos/male
photo.jpg"))
        myLabel9900 = Label(student, image=my_image33444).place(x=149,
y=75)
        mlayel = Label(student, text=resu1, font=('Aerial', 45)).place(x=369, y=95)
        mlaclasel = Label(student, text=resu2, font=('Aerial', 45)).place(x=369,
y=165)
        mlrewqqqsel = Label(student, text=resu3, font=('Aerial', 45)).place(x=360,
y=228)
```

```
        strbg = '_' * 1000
        mlrewqqqsel = Label(student, text=strbg, width=121,
height=1).place(x=147, y=290)
    elif resu == 'F':
        my_image300044 = ImageTk.PhotoImage(Image.open("Photos/female
photo.jpg"))
        myLabel5656 = Label(student, image=my_image300044).place(x=149,
y=75)
        mlayel = Label(student, text=resu1, font=('Aerial', 45)).place(x=369, y=95)
        mlaclasel = Label(student, text=resu2, font=('Aerial', 45)).place(x=369,
y=165)
        mlrewqqqsel = Label(student, text=resu3, font=('Aerial', 45)).place(x=360,
y=228)
        strbg = '_' * 1000
        mlrewqqqsel = Label(student, text=strbg, width=121,
height=1).place(x=147, y=290)
    else:
        print("PLEASE CONTACT DEVELOPERS")
    mydb.close()

    f = open("thought.txt", "r")
    message = ""
    for x in f:
        message += x
    f.close()

    temesadf = "Thought of the day:\n\t" + message
    stud_butt8 = Label(student, text=temesadf, font=('lucida', 20),
fg="#5d6af0").place(x=10, y=450)


stud_butt1=Button(student,image=my_image1q,bg="#bc8585",width=140,hei
ght=100).place(x=0,y=0)

stud_butt2=Button(student,image=my_image2q,bg="#000000",width=140,hei
ght=100).place(x=0,y=100)
```

```
stud_butt3=Button(student,image=my_image3q,bg="#b13030",width=140,hei
ght=100,command=student_competitionshow).place(x=0,y=200)

stud_butt4=Button(student,image=my_image4q,bg="#a61bff",width=140,heig
ht=100,command=student_attendanceshow).place(x=0,y=300)

stud_butt5=Button(student,image=my_image5q,bg="#86f601",width=140,heig
ht=100,command=student_marksshow).place(x=0,y=400)

stud_butt6=Button(student,image=my_image6q,bg="#165d33",width=140,hei
ght=100,command=student_noticeshow).place(x=0,y=500)

stud_butt7=Button(student,image=my_image7q,bg="#8ffaba",width=140,heig
ht=100,command=student_reportshow).place(x=0,y=600)

stud_butt8=Button(student,text="P/A",bg="#a8a325",width=4,height=1,font=(
'Aerial',50),padx=18,pady=23,command=student_attendanceshow).place(x=10
00,y=0)

stud_butt9=Button(student,text="FEES",bg="#208749",width=4,height=1,font
=('Aerial',50),padx=18,pady=23).place(x=1000,y=177)

stud_butt10=Button(student,text="HW",bg="#2d6c7c",width=4,height=1,font=
('Aerial',50),padx=18,pady=23,command=student_homeworkshow).place(x=1
000,y=353)

stud_butt11=Button(student,text="INFO",bg="#847b75",width=4,height=1,fon
t=('Aerial',50),padx=18,pady=23).place(x=1000,y=529)
   my_button12 =
Button(student,text="BACK",command=student.destroy).place(x=146, y=680)


   student.mainloop()


##########################################################
######
```

```
def updaterecord():
    updt=Toplevel()
    updt.title("SELECT RECORD")
    updt.geometry("300x200")
    global gq
    gq = IntVar()
    upq1 = Label(updt, text="Q.Which record do you want to:-").place(x=10,
y=10)
    upq2 = Radiobutton(updt, text="Username", variable=gq,
value=1).place(x=110, y=40)
    upq2 = Radiobutton(updt, text="Password", variable=gq,
value=2).place(x=110, y=60)
    upq2 = Radiobutton(updt, text="Name", variable=gq, value=3).place(x=110,
y=80)
    upq2 = Radiobutton(updt, text="Class", variable=gq, value=4).place(x=110,
y=100)
    Button(updt, text="SUBMIT", padx=1, pady=1, bg="Red",
fg="Black",command=updaterecord1).place(x=120, y=140)
    updt.mainloop()

def updaterecord1():
    updf = gq.get()
    if updf==1:
        updf="username"
        upod=Toplevel()
        upod.title("CHANGING USERNAME")
        upod.geometry("400x250")
        global uped4
        global uped5
        global uped6
        uped = Label(upod, text="OLD USERNAME").place(x=20, y=30)
        uped2 = Label(upod, text="PASSWORD").place(x=20, y=90)
        uped3 = Label(upod, text="NEW USERNAME").place(x=20, y=150)
        uped4 = Entry(upod, width=30, borderwidth=2, fg="Black")
        uped4.place(x=130, y=30)
        uped5 = Entry(upod, width=30, borderwidth=2, fg="Black",show="*")
```

```
        uped5.place(x=130, y=90)
        uped6 = Entry(upod, width=30, borderwidth=2, fg="Black")
        uped6.place(x=130, y=150)
        uped7 = Button(upod, text="SUBMIT", bg="Black",
fg="White",command=usernamechange).place(x=160, y=200)
        upod.mainloop()

    elif updf==2:
        updf="password"
        ufef = Toplevel()
        ufef.title("CHANGING PASSWORD")
        ufef.geometry("400x250")
        global uftq4
        global uftq5
        global uftq6
        ufqd1 = Label(ufef, text="USERNAME").place(x=20, y=30)
        uftq2 = Label(ufef, text="OLD PASSWORD").place(x=20, y=90)
        uftq3 = Label(ufef, text="NEW PASSWORD").place(x=20, y=150)
        uftq4 = Entry(ufef, width=30, borderwidth=2, fg="Black")
        uftq4.place(x=130, y=30)
        uftq5 = Entry(ufef, width=30, borderwidth=2, fg="Black",show="*")
        uftq5.place(x=130, y=90)
        uftq6 = Entry(ufef, width=30, borderwidth=2, fg="Black",show="*")
        uftq6.place(x=130, y=150)
        uftq7 = Button(ufef, text="SUBMIT", bg="Black", fg="White",
command=passwordchange).place(x=160, y=200)
        ufef.mainloop()

    elif updf==3:
        updf="name"
        upna = Toplevel()
        upna.title("CHANGING NAME")
        upna.geometry("400x250")
        global upnv4
        global upnv5
        global upnv6
        upnv1 = Label(upna, text="USERNAME").place(x=20, y=30)
```

```
        upnv2 = Label(upna, text="OLD NAME").place(x=20, y=90)
        upnv3 = Label(upna, text="NEW NAME").place(x=20, y=150)
        upnv4 = Entry(upna, width=30, borderwidth=2, fg="Black")
        upnv4.place(x=130, y=30)
        upnv5 = Entry(upna, width=30, borderwidth=2, fg="Black")
        upnv5.place(x=130, y=90)
        upnv6 = Entry(upna, width=30, borderwidth=2, fg="Black")
        upnv6.place(x=130, y=150)
        upnv7 = Button(upna, text="SUBMIT", bg="Black", fg="White",
command=namechange).place(x=160, y=200)
        upna.mainloop()

    elif updf==4:
        updf="class"
        upoi = Toplevel()
        upoi.title("CHANGING NAME")
        upoi.geometry("400x250")
        global upgy4
        global upgy6
        upgy1 = Label(upoi, text="USERNAME").place(x=20, y=50)
        upgy3 = Label(upoi, text="NEW CLASS").place(x=20, y=120)
        upgy4 = Entry(upoi, width=30, borderwidth=2, fg="Black")
        upgy4.place(x=130, y=50)
        upgy6 = Entry(upoi, width=30, borderwidth=2, fg="Black")
        upgy6.place(x=130, y=120)
        upgy7 = Button(upoi, text="SUBMIT", bg="Black",
fg="White",command=classchange).place(x=160, y=200)
        upoi.mainloop()

    else:
        pass

def usernamechange():
    mydb = mysql.connector.connect(host="localhost", user=hereistheuser,
password=hereisthepassword)
    mycursor = mydb.cursor()
    mycursor.execute("create database if not exists school;")
```

```
    mycursor.execute("use school;")
    mycursor.execute("select username,password from login;")
    myupdto = mycursor.fetchall()
    mydb.close()
    for upo in myupdto:
        if uped4.get() == upo[0]:
            if uped5.get() == upo[1]:
                mydb = mysql.connector.connect(host="localhost",
user=hereistheuser, password=hereisthepassword)
                mycursor = mydb.cursor()
                mycursor.execute("create database if not exists school;")
                mycursor.execute("use school;")
                mycursor.execute('update login set Username="' + uped6.get() + '"
where Username="' + uped4.get() + '";')
                mycursor.execute('update loginstudent set Username="' + uped6.get()
+ '" where Username="' + uped4.get() + '";')
                mycursor.execute('update loginteacher set Username="' + uped6.get()
+ '" where Username="' + uped4.get() + '";')
                mydb.commit()
                mydb.close()

            else:
                popup1()
        else:
            pass

def passwordchange():
    mydb = mysql.connector.connect(host="localhost", user=hereistheuser,
password=hereisthepassword)
    mycursor = mydb.cursor()
    mycursor.execute("create database if not exists school;")
    mycursor.execute("use school;")
    mycursor.execute("select username,password from login;")
    myupdte = mycursor.fetchall()
    mydb.close()
    for upoc in myupdte:
        if uftq4.get() == upoc[0]:
```

```
        if uftq5.get() == upoc[1]:
            mydb = mysql.connector.connect(host="localhost",
user=hereistheuser, password=hereisthepassword)
            mycursor = mydb.cursor()
            mycursor.execute("create database if not exists school;")
            mycursor.execute("use school;")
            mycursor.execute('update login set password="' + uftq6.get() + '"
where Username="' + uftq4.get() + '";')
            mycursor.execute('update loginstudent set password="' + uftq6.get() +
'" where Username="' + uftq4.get() + '";')
            mycursor.execute('update loginteacher set password="' + uftq6.get() +
'" where Username="' + uftq4.get() + '";')
            mydb.commit()
            mydb.close()
        else:
            popup1()
    else:
        pass

def namechange():
    mydb = mysql.connector.connect(host="localhost", user=hereistheuser,
password=hereisthepassword)
    mycursor = mydb.cursor()
    mycursor.execute("create database if not exists school;")
    mycursor.execute("use school;")
    mycursor.execute("select username from login;")
    myunamo = mycursor.fetchall()
    mydb.close()
    for upona in myunamo:
        if upnv4.get() == upona[0]:
            mydb = mysql.connector.connect(host="localhost",
user=hereistheuser, password=hereisthepassword)
            mycursor = mydb.cursor()
            mycursor.execute("create database if not exists school;")
            mycursor.execute("use school;")
            mycursor.execute('select Logintype from login where
Username="'+upnv4.get()+'";')
```

```
            mylomo=mycursor.fetchall()
            mydb.close()
            for yu in mylomo:
                loce=yu[0]

            if loce =="student":
                mydb = mysql.connector.connect(host="localhost",
user=hereistheuser, password=hereisthepassword)
                mycursor = mydb.cursor()
                mycursor.execute("create database if not exists school;")
                mycursor.execute("use school;")
                mycursor.execute('update loginstudent set Name="' + upnv6.get() +
'" where Username="' + upnv4.get() + '";')
                mydb.commit()
                mydb.close()
            elif loce == "teacher":
                mydb = mysql.connector.connect(host="localhost",
user=hereistheuser, password=hereisthepassword)
                mycursor = mydb.cursor()
                mycursor.execute("create database if not exists school;")
                mycursor.execute("use school;")
                mycursor.execute('update loginteacher set Name="' + upnv6.get() +
'" where Username="' + upnv4.get() + '";')
                mydb.commit()
                mydb.close()
            else:
                break



        else:
            pass

def classchange():
    mydb = mysql.connector.connect(host="localhost", user=hereistheuser,
password=hereisthepassword)
    mycursor = mydb.cursor()
```

```
    mycursor.execute("create database if not exists school;")
    mycursor.execute("use school;")
    mycursor.execute('update loginstudent set Class="' + upgy6.get() + '" where
Username="' + upgy4.get() + '";')
    mydb.commit()
    mydb.close()

def deleterecord():
    dkc = Toplevel()
    dkc.title("Delete records")
    dkc.geometry("400x200")
    global dlk2
    dlk1 = Label(dkc, text="Username(record to delete)").place(x=10, y=70)
    dlk2 = Entry(dkc, width=30, borderwidth=2, fg="Black")
    dlk2.place(x=180, y=70)
    dlk3 = Button(dkc, text="DELETE
RECORD",command=deleterecord1).place(x=140,y=120)
    dkc.mainloop()

def deleterecord1():
    mydb = mysql.connector.connect(host="localhost", user=hereistheuser,
password=hereisthepassword)
    mycursor = mydb.cursor()
    mycursor.execute("create database if not exists school;")
    mycursor.execute("use school;")
    mycursor.execute('delete from login where Username="' + dlk2.get() + '";')
    mycursor.execute('delete from loginstudent where Username="' + dlk2.get()
+ '";')
    mycursor.execute('delete from loginteacher where Username="' + dlk2.get()
+ '";')
    mydb.commit()
    mydb.close()

def technicalteamdatamainmenu():
    toc = Tk()
    toc.title("Update records")
    toc.geometry("400x300")
```

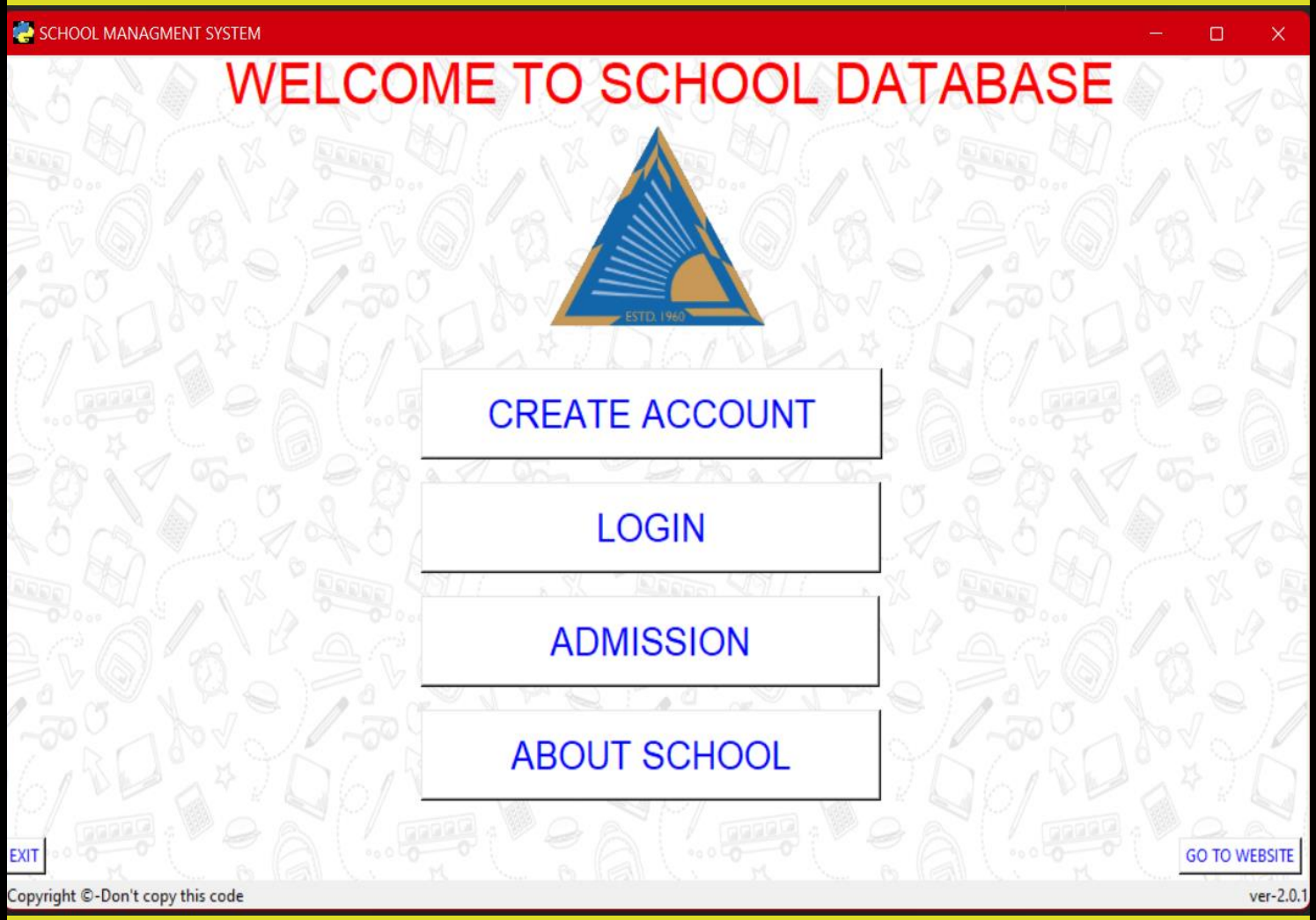
```
    toc1 = Button(toc, text=" UPDATE RECORD", padx=1, pady=1, bg="Red",
fg="Black", font=('Aerial', 20),command=updaterecord).place(x=65, y=50)
    toc4 = Button(toc, text=" DELETE RECORD", padx=1, pady=1, bg="Red",
fg="Black", font=('Aerial', 20),command=deleterecord).place(x=65, y=160)
    toc.mainloop()

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

def admissionshow():
    win = Toplevel()
    win.geometry("900x850")
    win.title("MYSQLProject")
    win.config(bg='#2bfcb3')
    # frame = Frame(win, bd=4, bg='blue')
    # frame.place(width=1150, height=600)

    label = Label(win, text='ADMISSION NOTICE', font=('lucida', 40, 'bold'),
fg='black',bg='red')
    label.pack(side=TOP, fill=Y)

    photo = ImageTk.PhotoImage(Image.open("Photos/Untitled.png"))
    label2 = Label(win, image=photo)
    label2.place(x=145, y=60)

    win.mainloop()

##########################################################
#3

def student_aboutschoolshow():
    ws = Toplevel()
    ws.title('PythonGuides')
    ws.geometry('1025x500')
    ws.config(bg='#2bfcb3')
    photo = PhotoImage(file="Photos/neps.png")
    label = Label(ws,image=photo,bg="#2bfcb3")
```

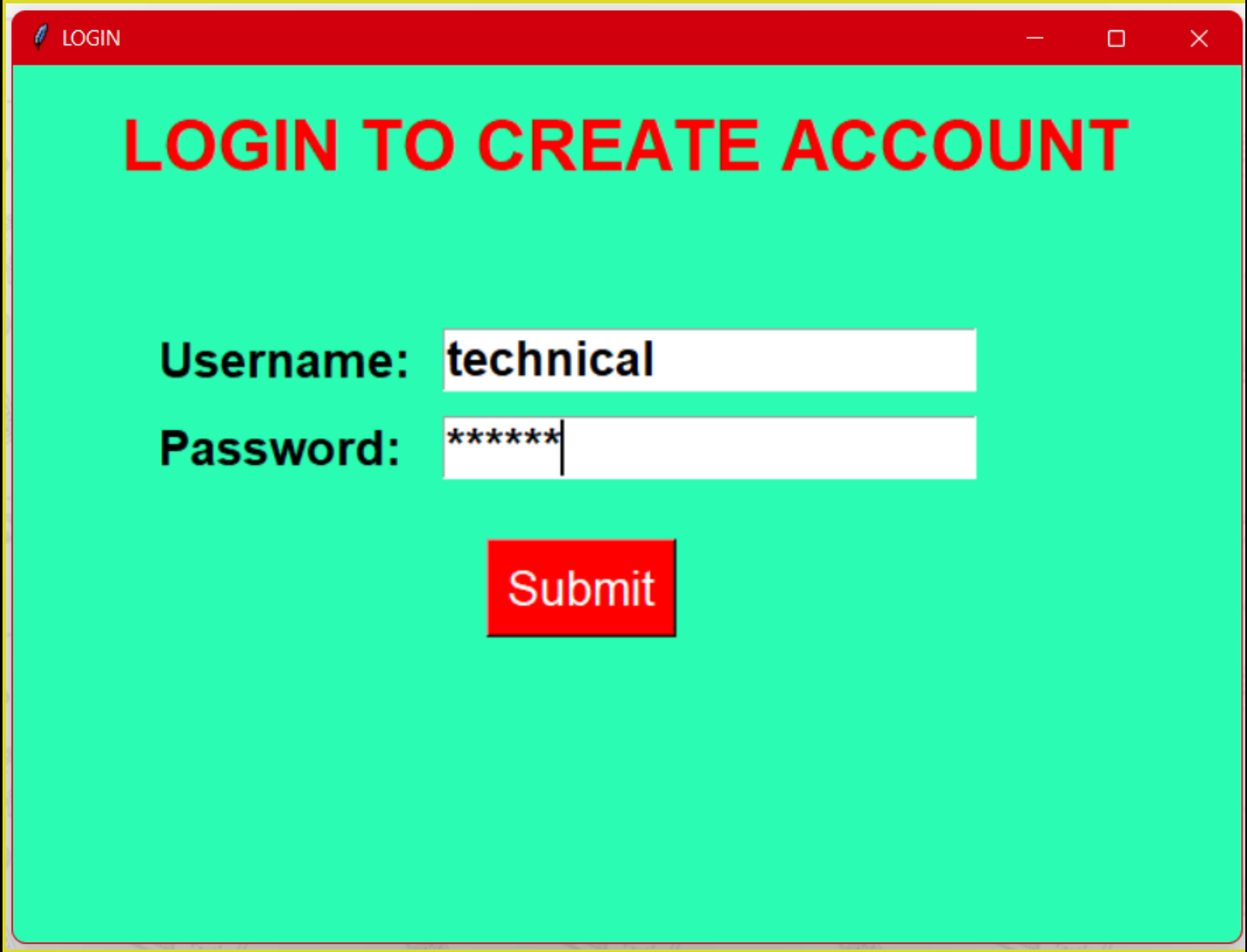
```
label.pack()
message = '''
New Era Public School, is a school in Mayapuri, New Delhi, India.

It was founded in 1960 by the New Era Education Society in Delhi.

The project of Mr. R.L. Chopra and Mrs. Usha Chopra came into existence in 1965.

The school is an All India Senior Secondary school, affiliated with the Central Board of Secondary Education

The school came into existence in one room in the year 1960.

The school has another branch in Pochampur, Dwarka with the same name.

OUR MISSION

Our primary objective is to facilitate the all round development of all children in a safe and nurturing environment,

enabling every child to develop into a well-adjusted, confident, free-thinking global citizen.

New Era seeks to empower each student to actualize their potential, to be confident in their endeavours whilst imbibing values of humility, work ethic, honesty, duty and legacy.

True Education means preparation for life and moves beyond the confines of academics and extra-curricular activities.

We aim to help a student develop emotionally and spiritually through a strong value system that has been incorporated in the curriculum.

New Era seeks to instill the time tested values in children to ensure that in their process of growing up, they learn to be honest, compassionate, truthful, sincere and a well-rounded citizens.'''
```

```
    text_box = Text(ws,height=30, width=120)
    text_box.pack(expand=True)
    text_box.insert('end', message)
    text_box.config(state='disabled',font="Roboto",bg="#2bfcb3",fg="black")

    ws.mainloop()




if __name__ == '__main__':
    print("For this program to work you need to install some
modules:\n1.Pillow(pip install Pillow)\n2.tkinter(pip install
tkinter)\n3.webbrowser(pip install webbrowser)\n4.mysql.connector(pip
intall mysql.connector)")
    # time.sleep(5)
    root=Tk()
    root.geometry("1000x600")
    root.title("SCHOOL MANAGMENT SYSTEM")
    root.iconbitmap('Photos/python.ico')                    #making icon
    my_image = ImageTk.PhotoImage(Image.open("Photos/cloud.png"))
#Bringing photo
    my_image1 = ImageTk.PhotoImage(Image.open("Photos/neps.png"))
    # my_label = Label(root,image=my_image)
    # my_label.pack()
    my_canvas=Canvas(root,width=1000,height=600)
#Making canvas
    my_canvas.pack(fill="both",expand=True)                    #Packing
canvas
    my_canvas.create_image(0,0,image=my_image,anchor="nw")
#creating image of canvas
    my_canvas.create_image(390,40,image=my_image1,anchor="nw")
#creating image of canvas
    fontStyle = tkFont.Font(family="Calluna", size=30)          #defining
```

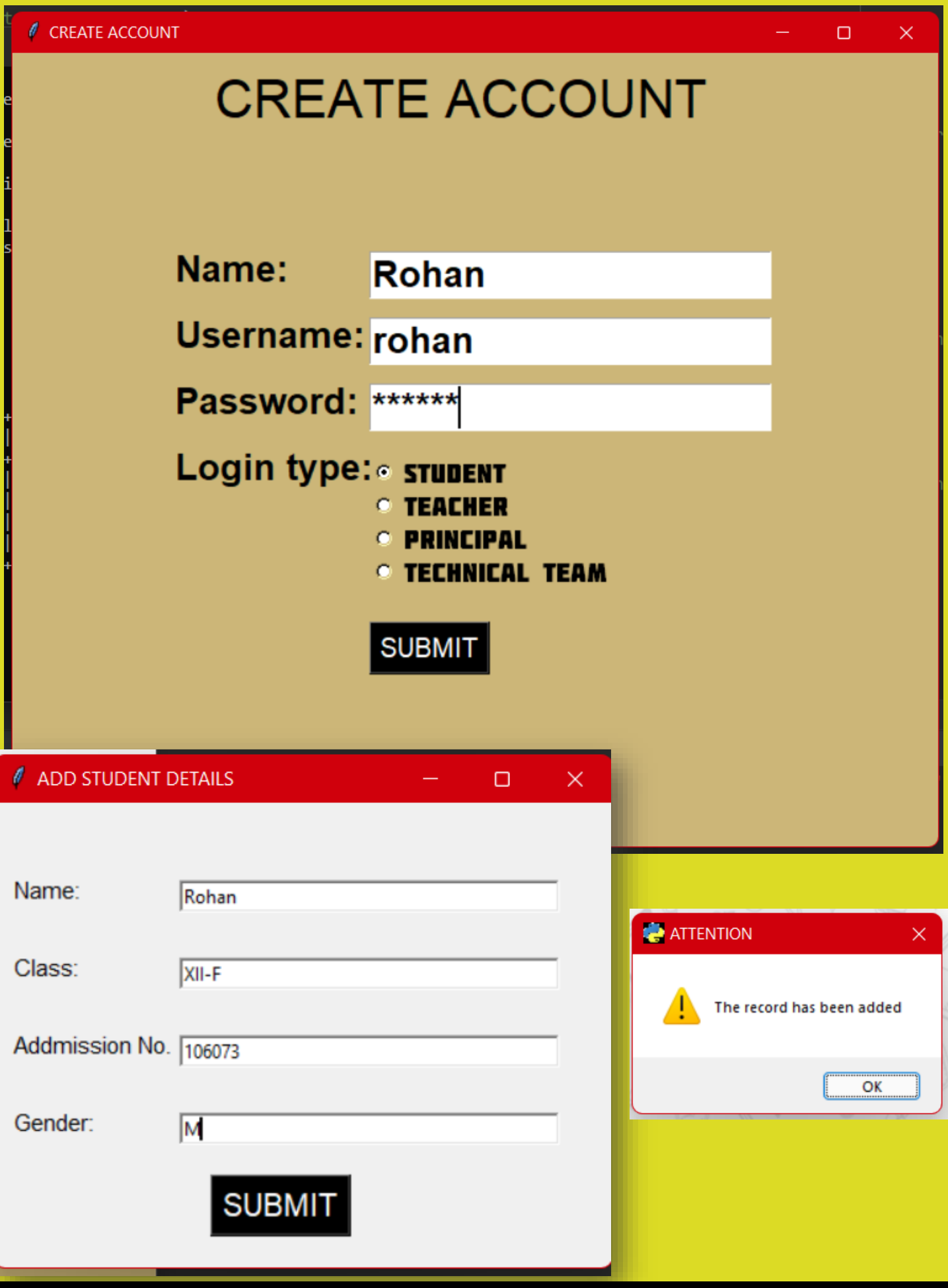
```
font style
    my_canvas.create_text(510,20,text="WELCOME TO SCHOOL
DATABASE",font=fontStyle,fill="red") # creating a text
    fontStyle1 = tkFont.Font(family="Calluna", size=20)
    my_button1=Button(root,text="CREATE
ACCOUNT",font=fontStyle1,padx=40,pady=5,bg="white",fg="blue",command=c
reateaccount).place(x=320,y=220)

my_button2=Button(root,text="LOGIN",font=fontStyle1,padx=123,pady=5,bg=
"white",fg="blue",command=loginscreen).place(x=320,y=300)

my_button3=Button(root,text="ADMISSION",font=fontStyle1,padx=88,pady=5,
bg="white",fg="blue",command=admissionshow).place(x=320,y=380)
    my_button4=Button(root,text="ABOUT
SCHOOL",font=fontStyle1,padx=58,pady=5,bg="white",fg="blue",command=stu
dent_aboutschoolshow).place(x=320,y=460)
    my_button5=Button(root,text="GO TO
WEBSITE",bg="white",fg="blue",command=clicktoviewneps).place(x=900,y=5
50)

my_button6=Button(root,text="EXIT",bg="white",fg="blue",command=root.de
stroy).place(x=0,y=550)
    uuuuu=' '*13
    Label1=Label(root,text="Copyright ©-Don't copy this
code\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t\t"+uuuuu+"ver-
2.0.1",anchor=W,width=600).place(x=0,y=580)
    root.mainloop()
```

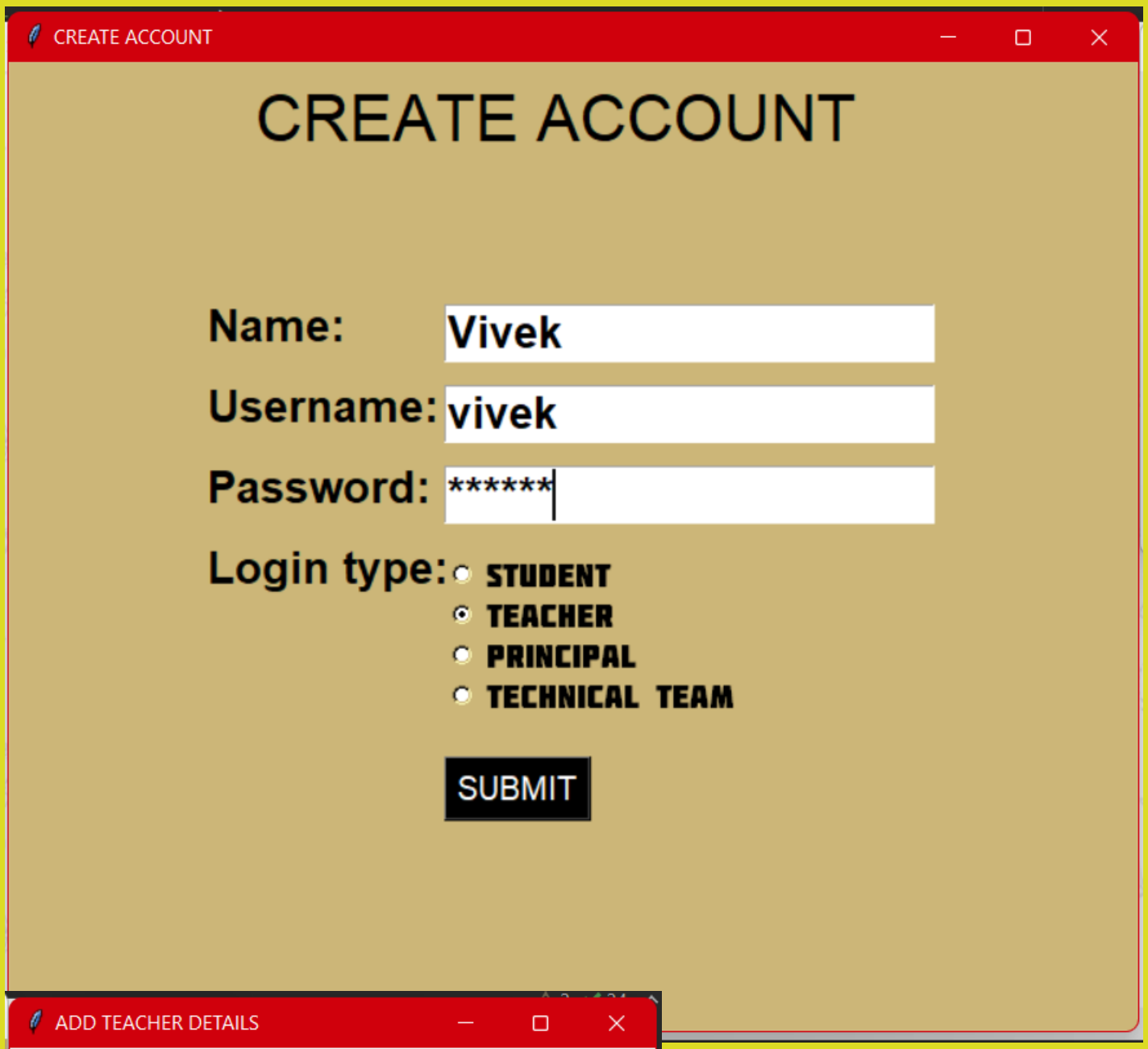
OUTPUT

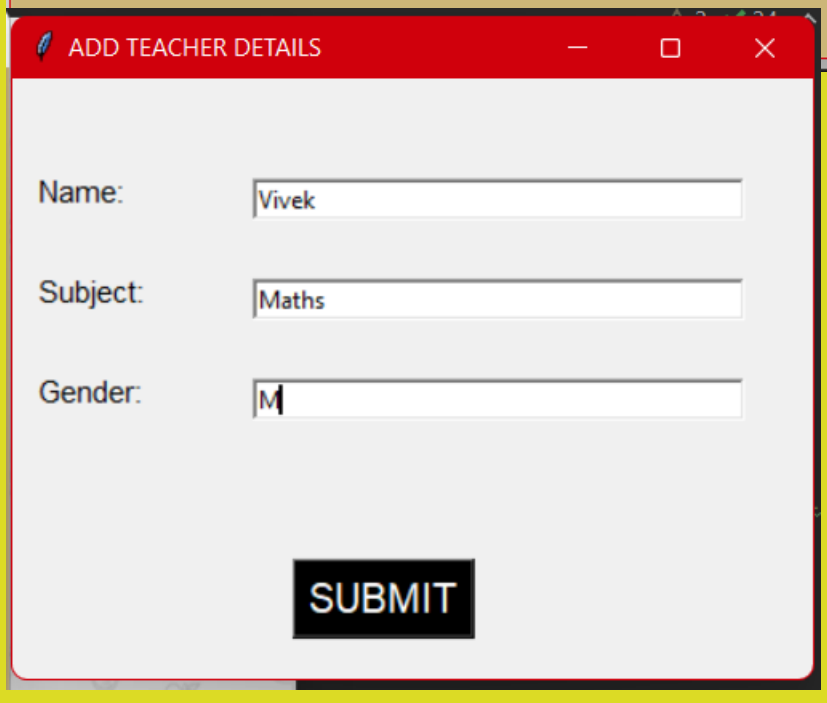

SCHOOL MANAGMENT SYSTEM
WELCOME TO SCHOOL DATABASE
ESTD. 1960
CREATE ACCOUNT
LOGIN
ADMISSION
ABOUT SCHOOL
EXIT
GO TO WEBSITE
Copyright ©-Don't copy this code
ver-2.0.1

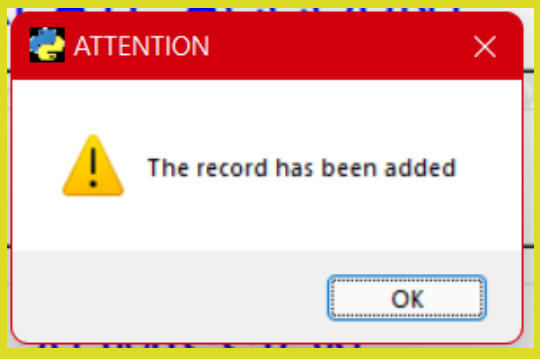



```
mysql> select * from login;
+-----------+----------+----------------+
| Username  | Password | Logintype      |
+-----------+----------+----------------+
| jasjeev   | jas123   | student        |
| aryan     | ary123   | student        |
| lalit     | lal123   | teacher        |
| technical | tec123   | technical team |
+-----------+----------+----------------+
4 rows in set (0.01 sec)
```


CREATE ACCOUNT
CREATE ACCOUNT
Name:
Rohan
Username:
rohan
Password:
******
Login type:
STUDENT
TEACHER
PRINCIPAL
TECHNICAL TEAM
SUBMIT
ADD STUDENT DETAILS
Name:
Rohan
Class:
XII-F
Addmission No.
106073
Gender:
M
SUBMIT
ATTENTION
The record has been added
OK


CREATE ACCOUNT
CREATE ACCOUNT
Name:
Vivek
Username:
vivek
Password:
******
Login type:
STUDENT
TEACHER
PRINCIPAL
TECHNICAL TEAM
SUBMIT

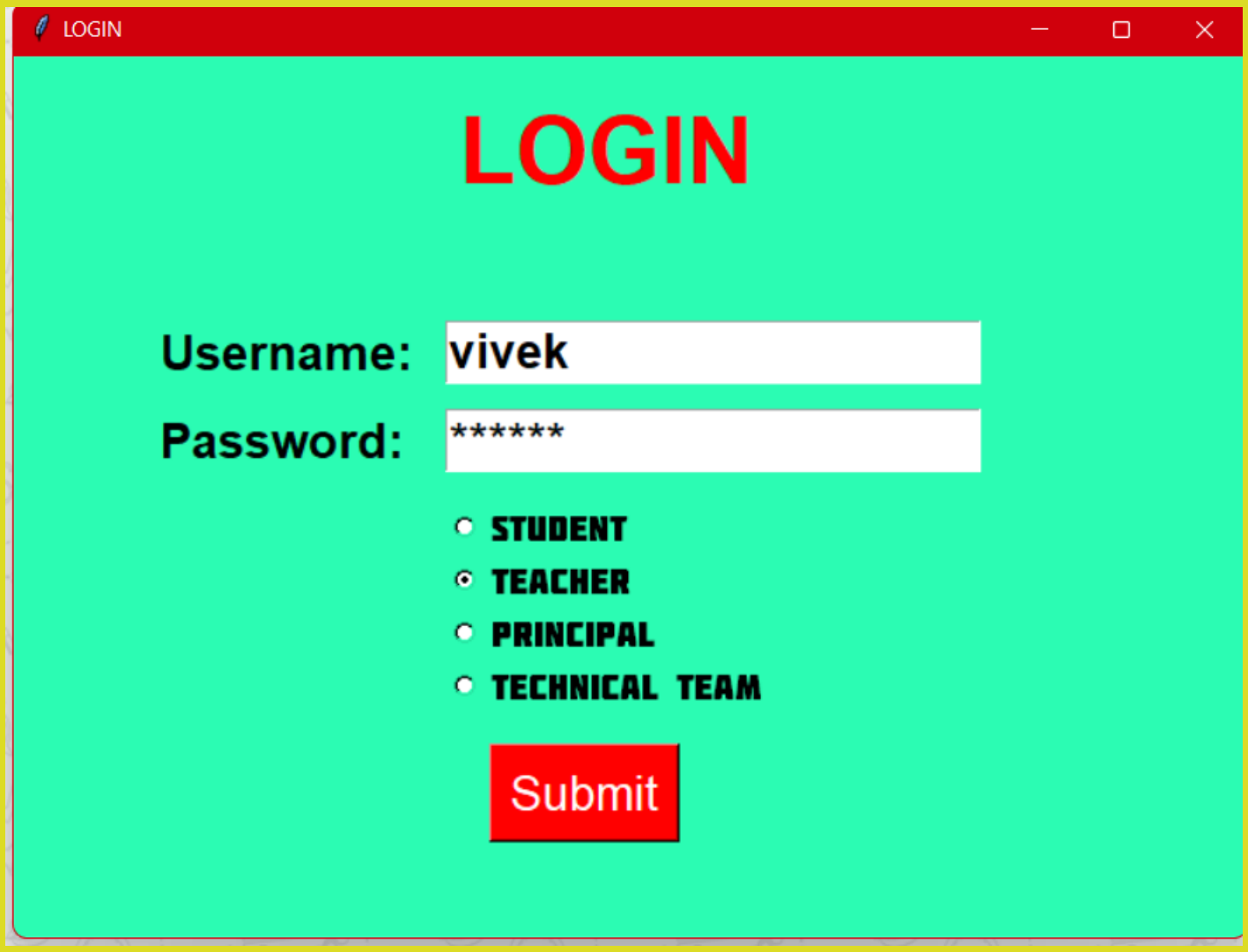

ADD TEACHER DETAILS
Name:
Vivek
Subject:
Maths
Gender:
M
SUBMIT


ATTENTION
The record has been added
OK

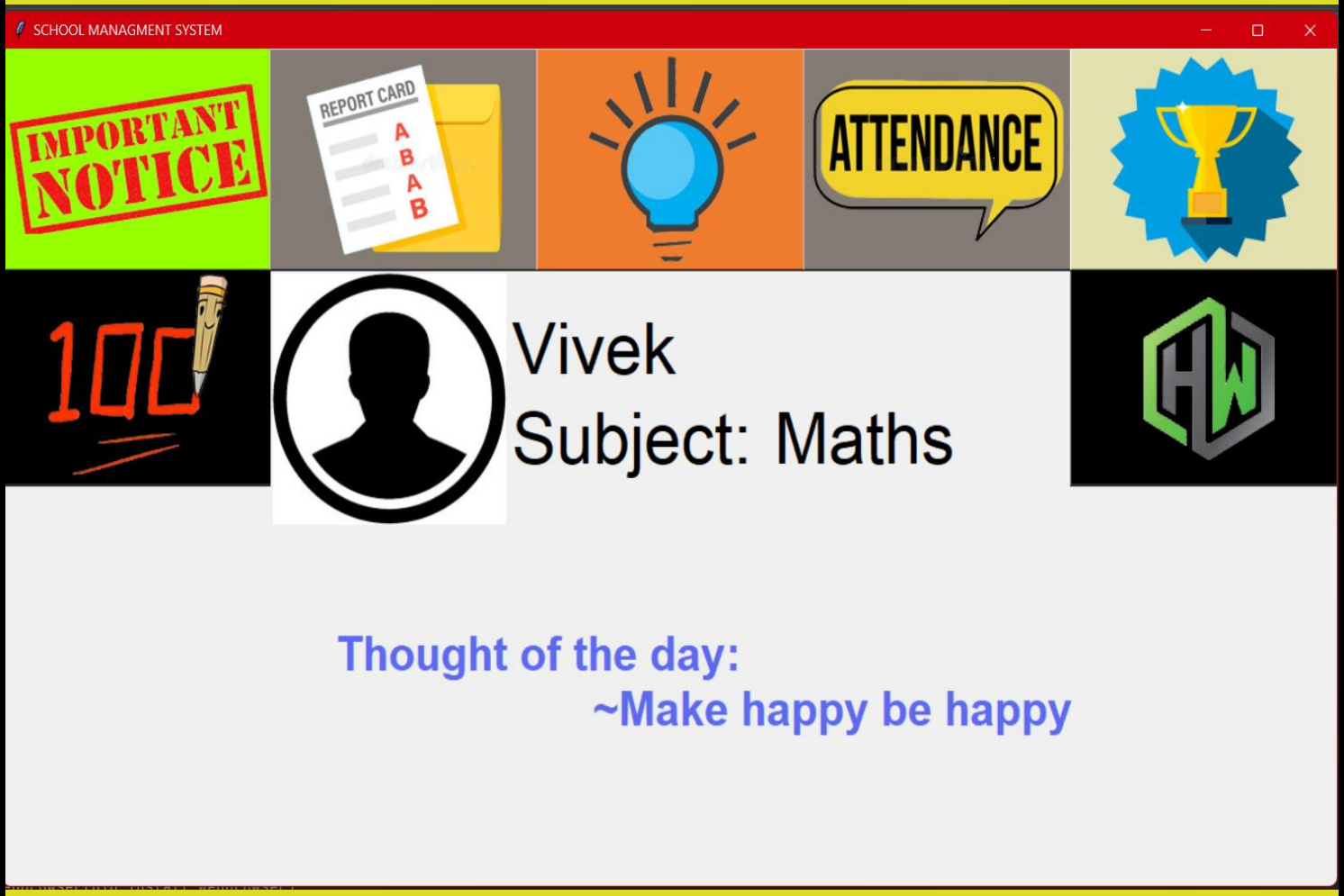
```
mysql> select * from login;
+-----------+----------+----------------+
| Username  | Password | Logintype      |
+-----------+----------+----------------+
| jasjeev   | jas123   | student        |
| aryan     | ary123   | student        |
| lalit     | lal123   | teacher        |
| technical | tec123   | technical team |
| rohan     | roh123   | student        |
| vivek     | viv123   | teacher        |
+-----------+----------+----------------+
6 rows in set (0.00 sec)
```

```
+----------+----------+--------+-------+-----------------+-----------+--------+
| Username | Password | Name   | Class | Admissionnumber | Logintype | Gender |
+----------+----------+--------+-------+-----------------+-----------+--------+
| jasjeev  | jas123   | Jasjeev| XII-F | 100073          | student   | M      |
| aryan    | ary123   | Aryan  | XII-F | 144534          | student   | M      |
| rohan    | roh123   | Rohan  | XII-F | 106073          | student   | M      |
+----------+----------+--------+-------+-----------------+-----------+--------+
```

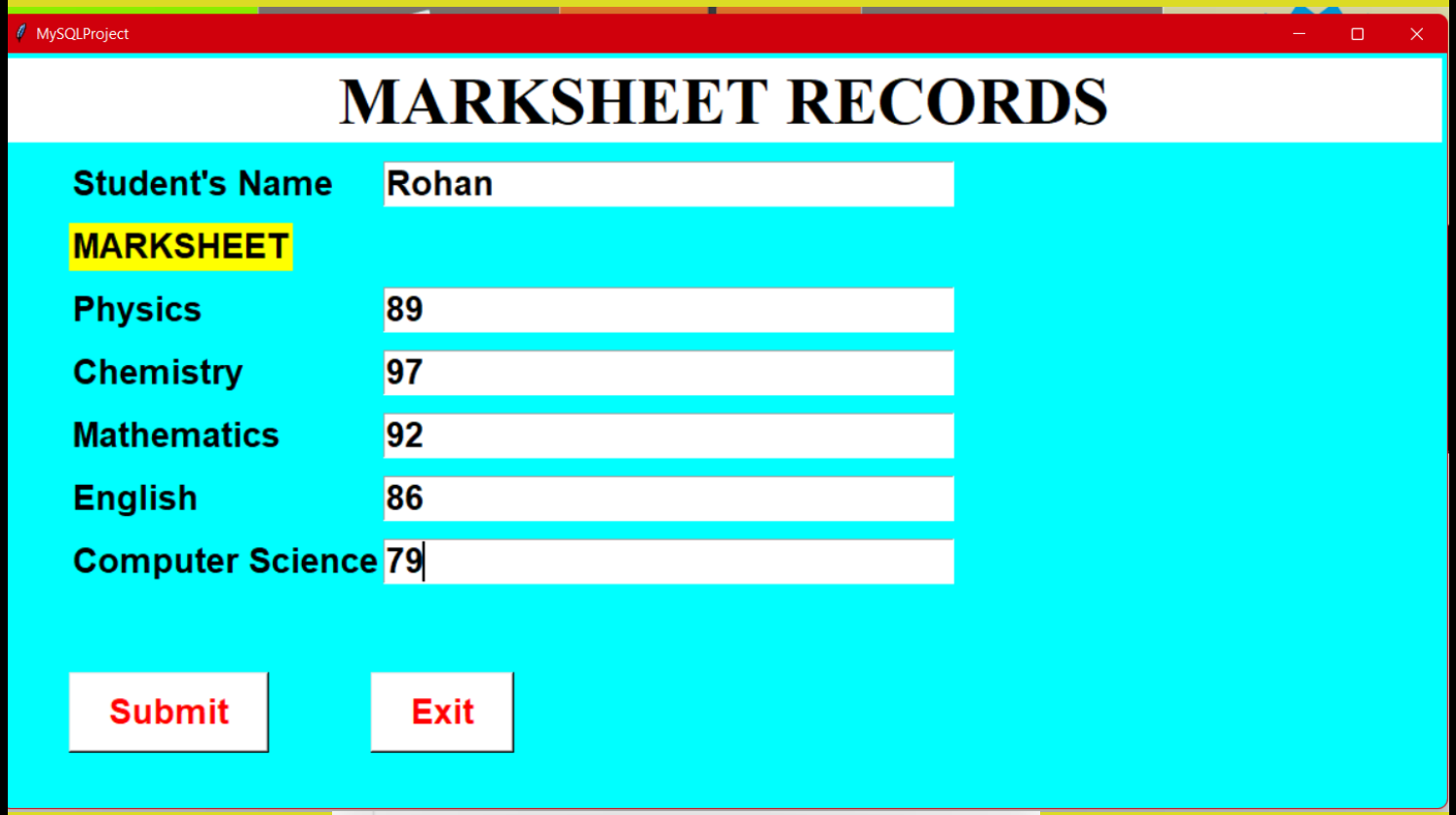
```
mysql> select * from loginteacher;
+----------+----------+---------+---------+-----------+--------+
| Username | Password | Name    | Subject | Logintype | Gender |
+----------+----------+---------+---------+-----------+--------+
| lalit    | lal123   | Lalit   | CS      | teacher   | M      |
| teacher  | tea123   | Teacher | Maths   | teacher   | F      |
| vivek    | viv123   | Vivek   | Maths   | teacher   | M      |
+----------+----------+---------+---------+-----------+--------+
```

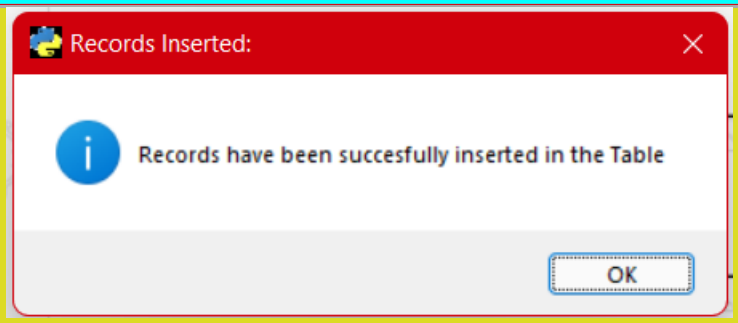



```
mysql> select * from login;
+-----------+----------+----------------+
| Username  | Password | Logintype      |
+-----------+----------+----------------+
| jasjeev   | jas123   | student        |
| aryan     | ary123   | student        |
| lalit     | lal123   | teacher        |
| technical | tec123   | technical team |
| rohan     | roh123   | student        |
| vivek     | viv123   | teacher        |
+-----------+----------+----------------+
6 rows in set (0.00 sec)
```



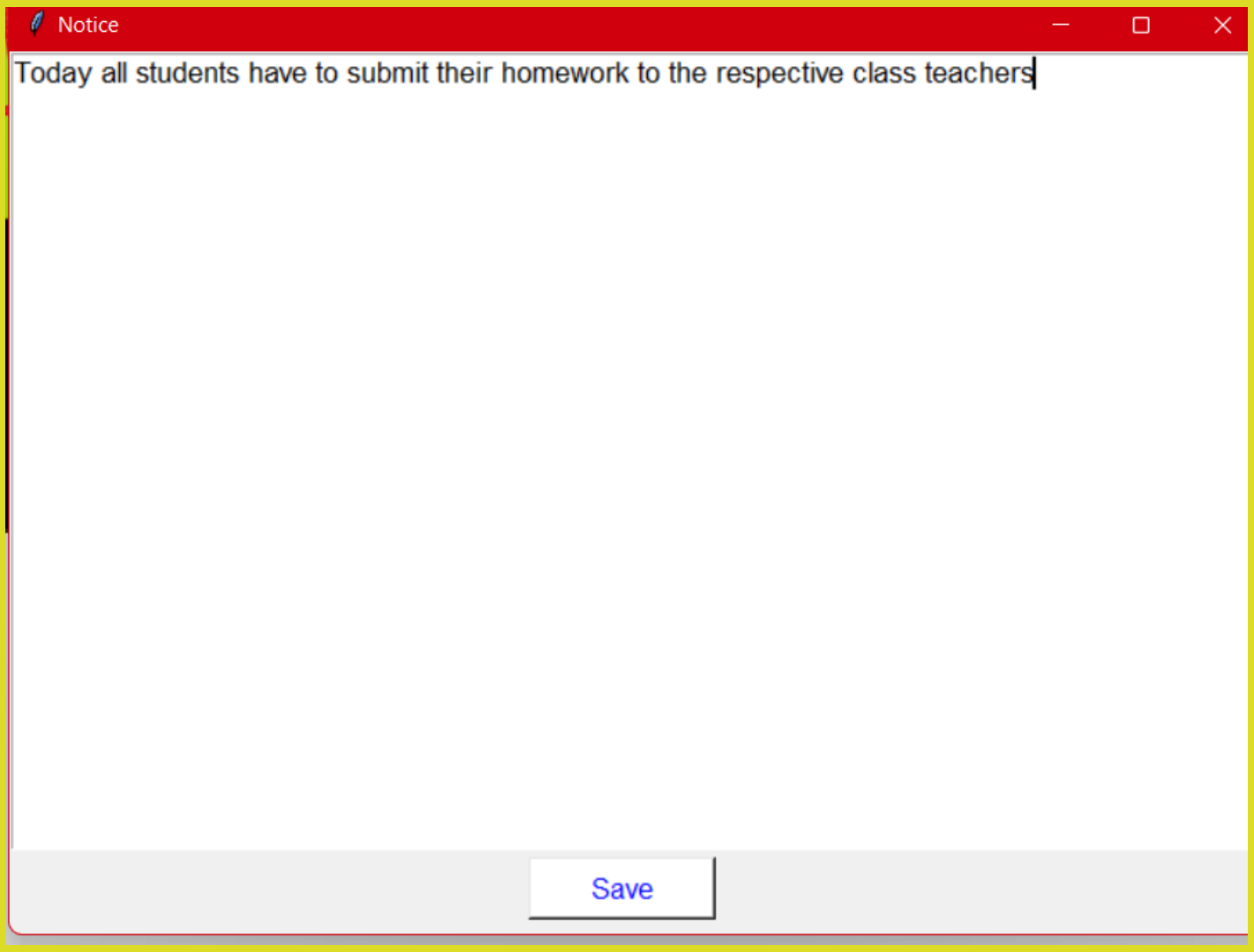

```
mysql> select * from loginteacher;
+----------+----------+---------+---------+-----------+--------+
| Username | Password | Name    | Subject | Logintype | Gender |
+----------+----------+---------+---------+-----------+--------+
| lalit    | lal123   | Lalit   | CS      | teacher   | M      |
| teacher  | tea123   | Teacher | Maths   | teacher   | F      |
| vivek    | viv123   | Vivek   | Maths   | teacher   | M      |
+----------+----------+---------+---------+-----------+--------+
```

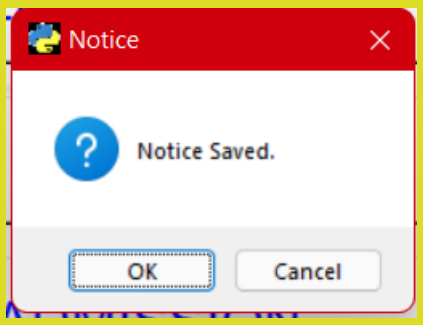


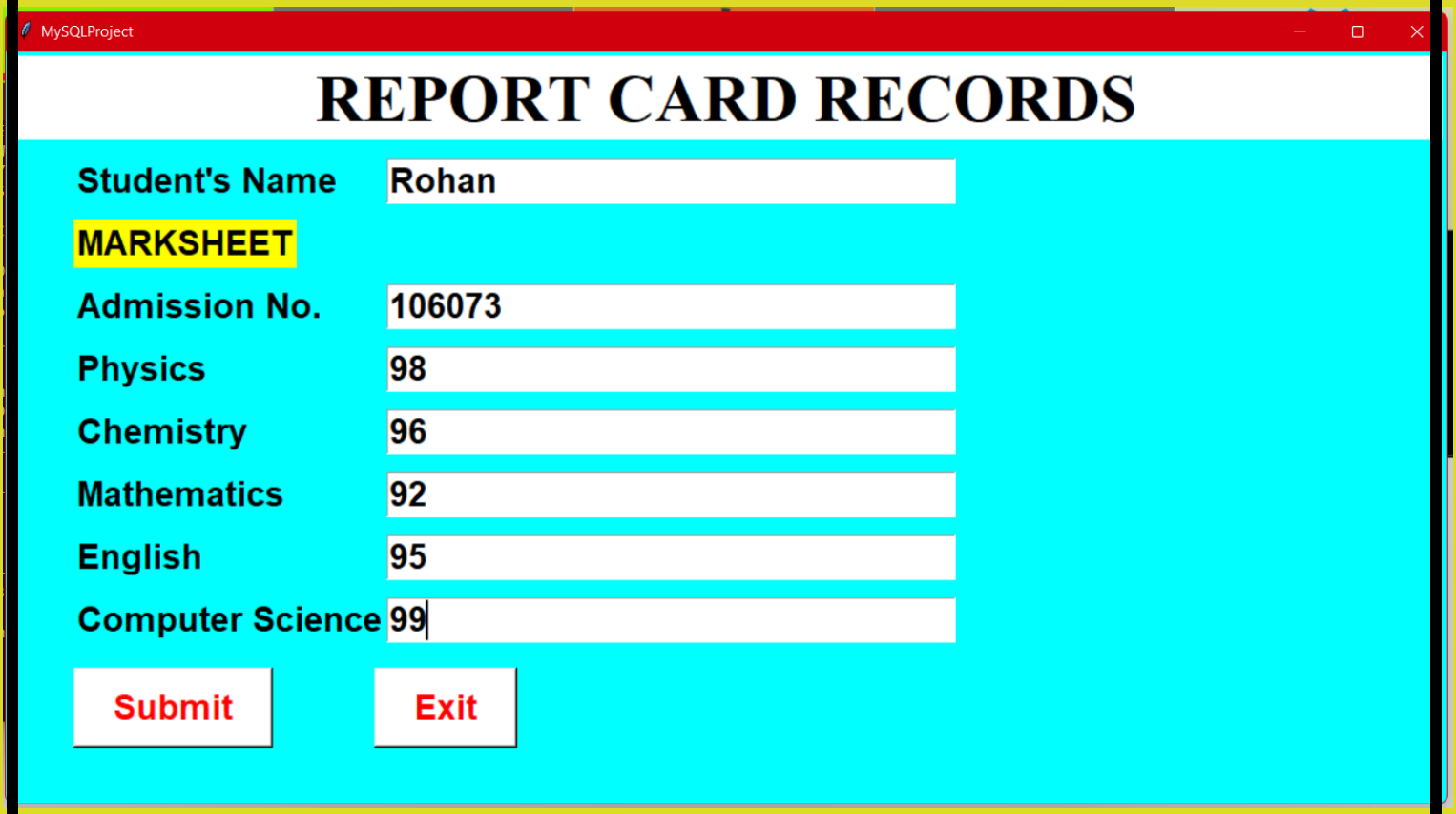


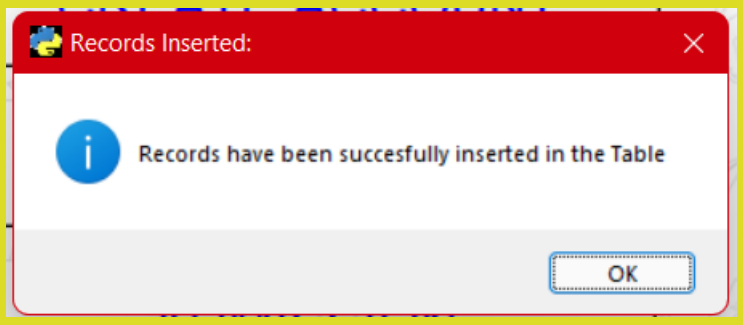

```
+--------------+---------------+-----------------+-------------------+
| Student_Name | Physics_marks | Chemistry_marks | Mathematics_marks |
+--------------+---------------+-----------------+-------------------+
| Rohan        |            89 |              97 |                92 |
+--------------+---------------+-----------------+-------------------+
1 row in set (0.00 sec)
```

```
+---------------+----------+-----------------------------+
| English_marks | Cs_marks | TotalMarksObtained_Outof500 |
+---------------+----------+-----------------------------+
|            86 |       79 |                         443 |
+---------------+----------+-----------------------------+
```


Notice
Today all students have to submit their homework to the respective class teachers
Save

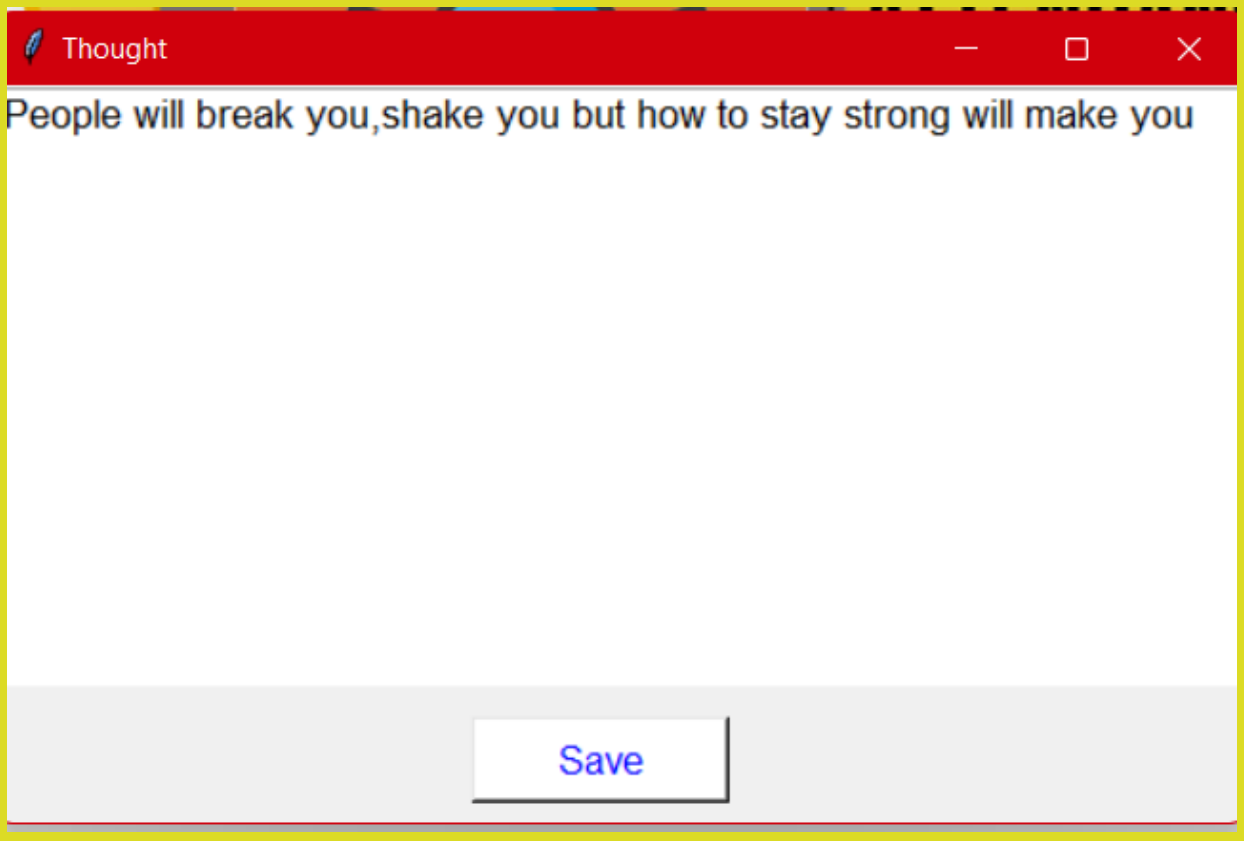

Notice
Notice Saved.
OK
Cancel

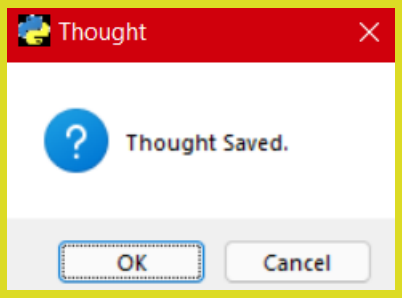


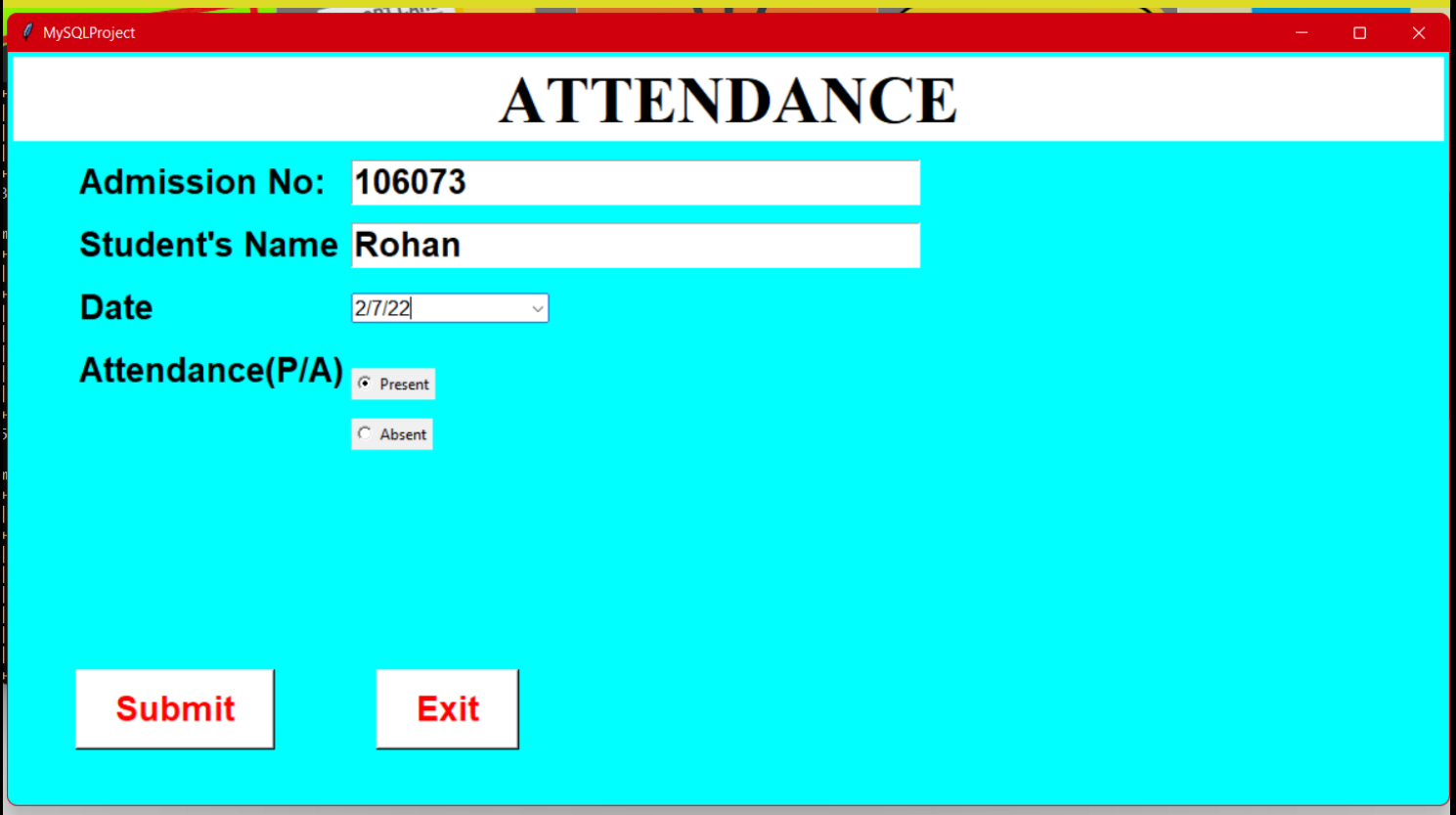


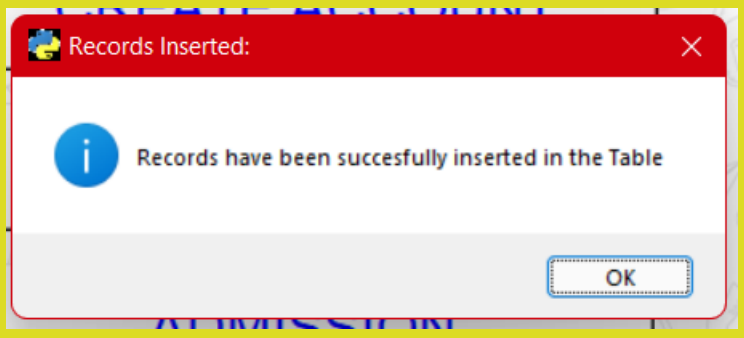
Student_Name	Admission_No	Physics_marks	Chemistry_marks
Rohan	106073	98	96

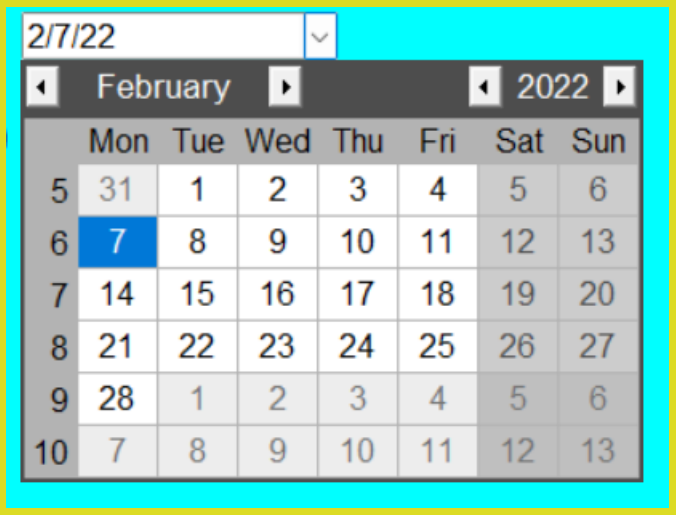
Mathematics_marks	English_marks	Cs_marks	TotalMarksObtained_Outof500
92	95	99	480


Thought
People will break you,shake you but how to stay strong will make you
Save

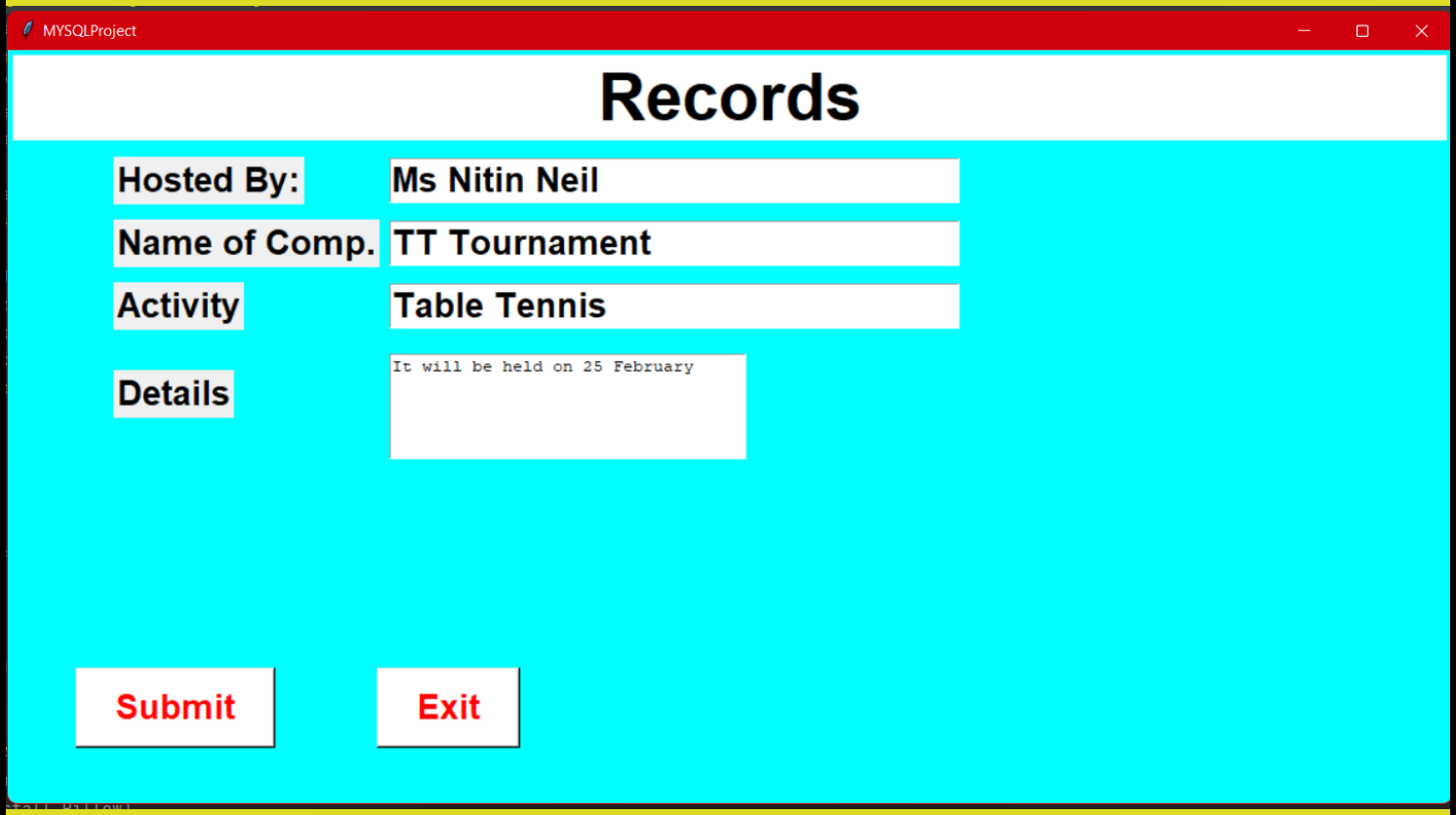

Thought
Thought Saved.
OK
Cancel

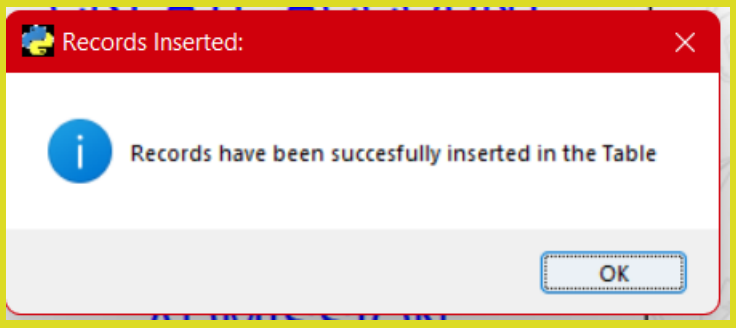






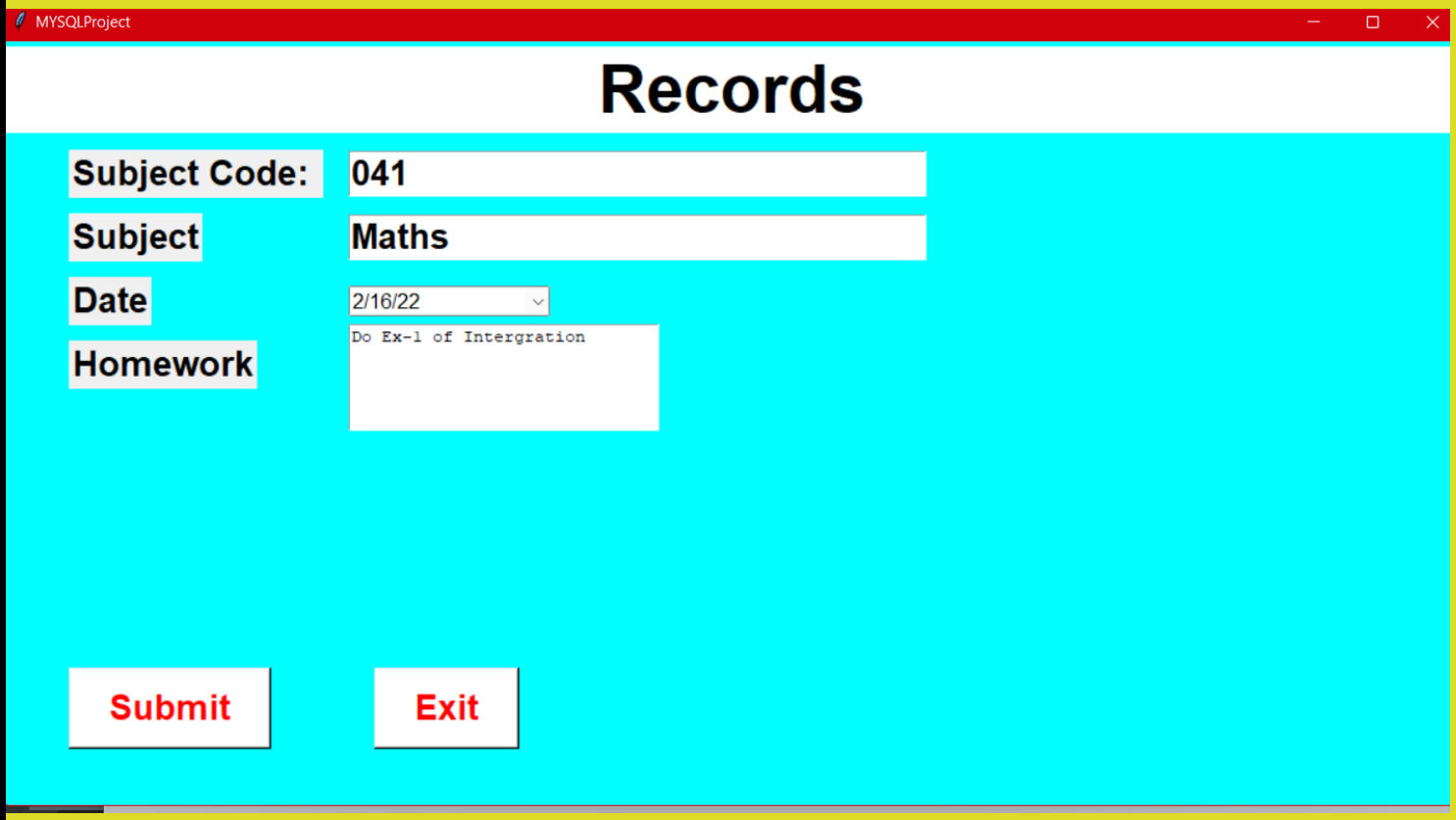
Admission_No	Student_Name	Date_of_Attendance	Attendance
106073	Rohan	2022-02-07	P

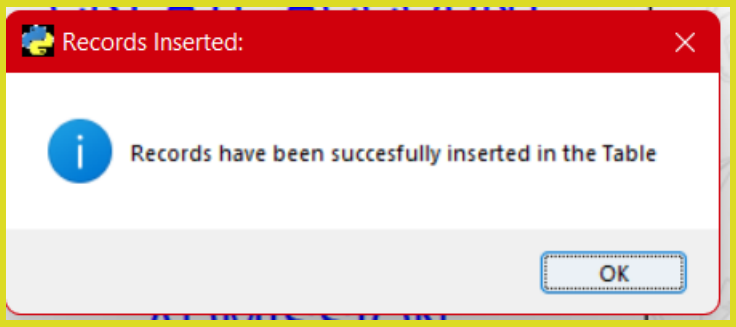




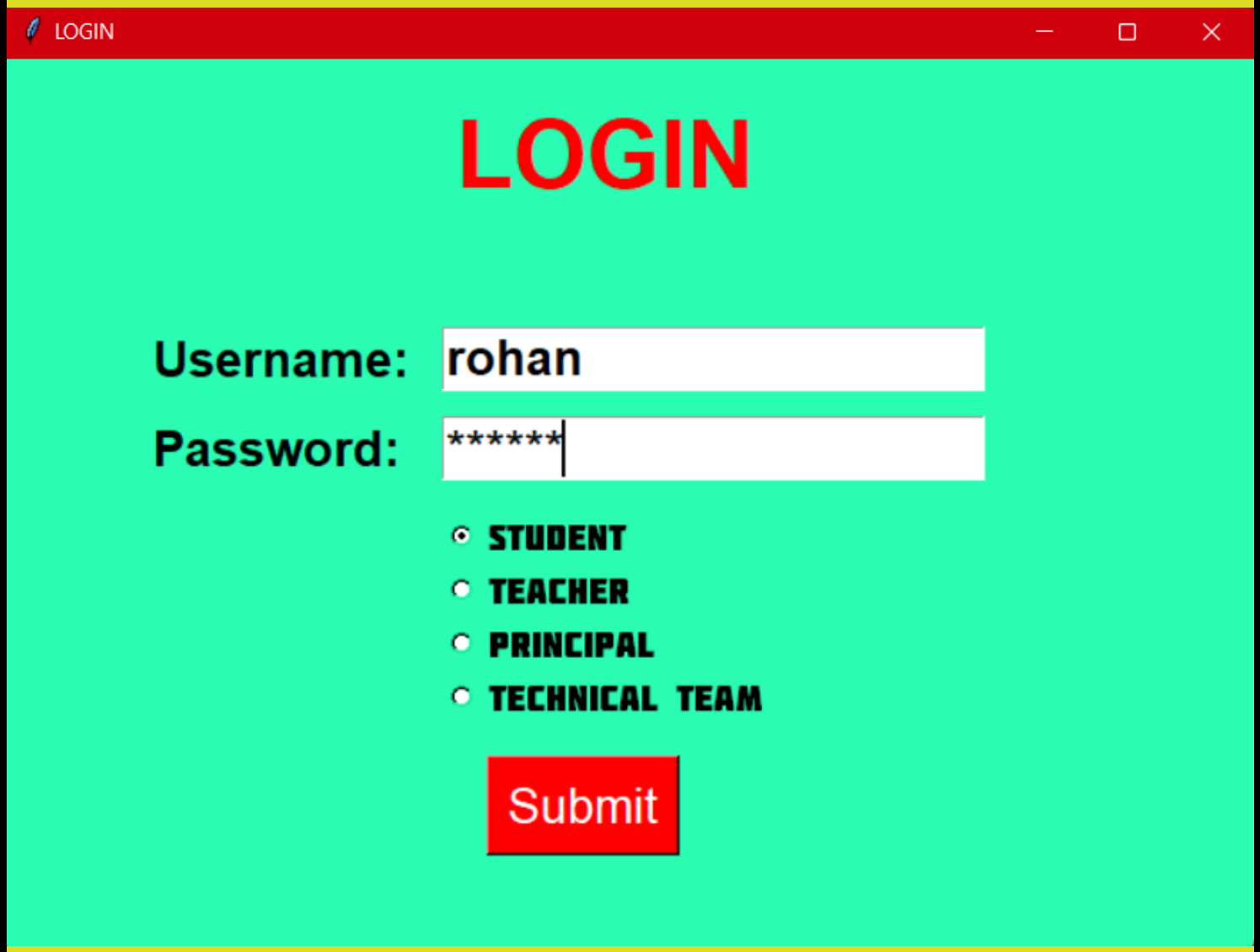

```
+---------------+---------------------+
| hosted_by     | Name_of_competition |
+---------------+---------------------+
| Ms Nitin Neil | TT Tournament       |
 |
+---------------+---------------------+
```

```
+--------------+-----------------------------------+
| Activity     | Details                           |
+--------------+-----------------------------------+
| Table Tennis | It will be held on 25 February    |

+--------------+-----------------------------------+
```

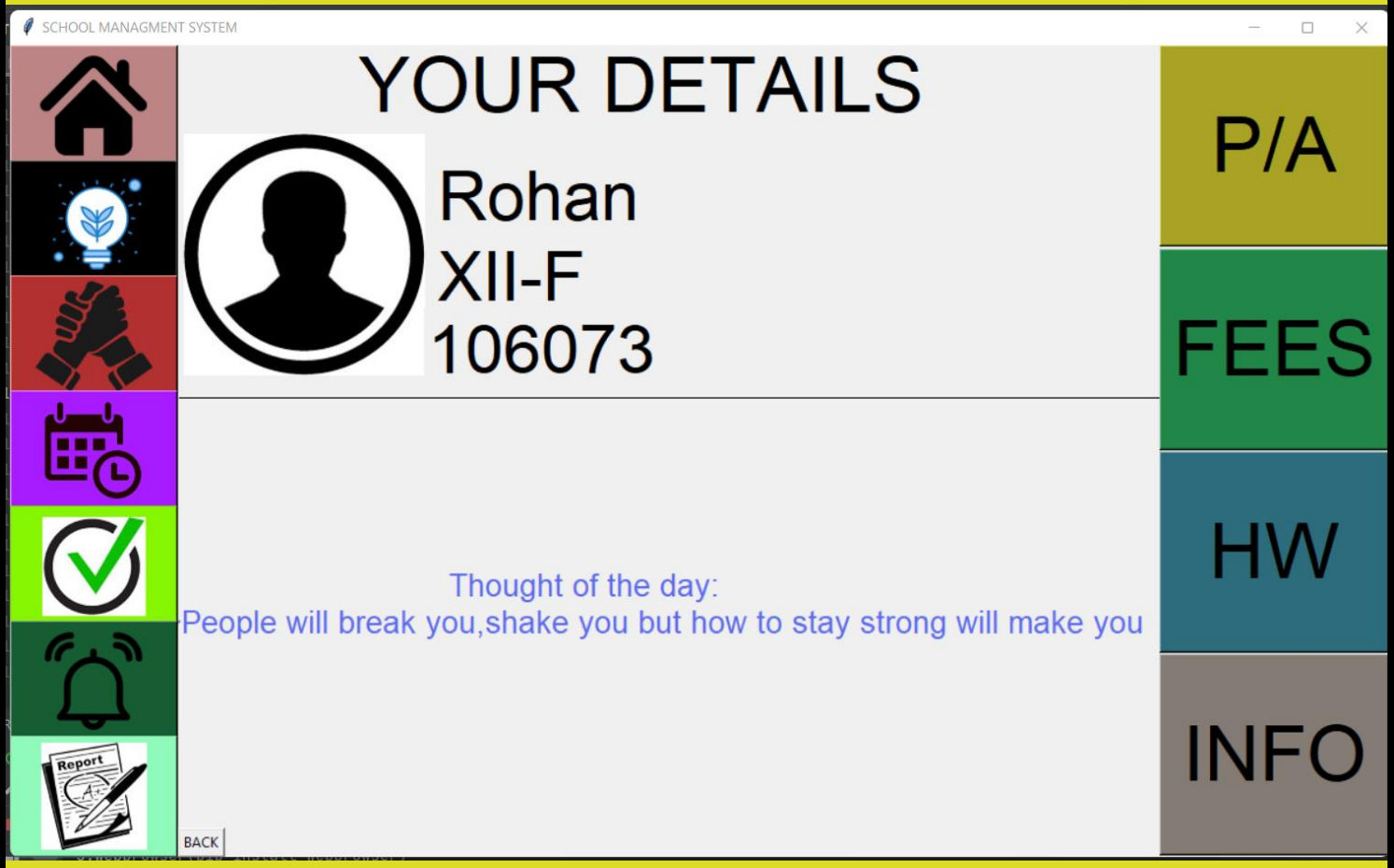





```
+--------------+---------+------------+--------------------------+
| Subject_Code | Subject | Date       | Homework                 |
+--------------+---------+------------+--------------------------+
| 041          | Maths   | 2022-02-16 | Do Ex-1 of Intergration  |
|
+--------------+---------+------------+--------------------------+
```




```
mysql> select * from login;
+-----------+----------+----------------+
| Username  | Password | Logintype      |
+-----------+----------+----------------+
| jasjeev   | jas123   | student        |
| aryan     | ary123   | student        |
| lalit     | lal123   | teacher        |
| technical | tec123   | technical team |
| rohan     | roh123   | student        |
| vivek     | viv123   | teacher        |
+-----------+----------+----------------+
6 rows in set (0.00 sec)
```




```
mysql> select * from loginstudent;
+----------+----------+---------+-------+-----------------+-----------+--------+
| Username | Password | Name    | Class | Admissionnumber | Logintype | Gender |
+----------+----------+---------+-------+-----------------+-----------+--------+
| jasjeev  | jas123   | Jasjeev | XII-F | 100073          | student   | M      |
| aryan    | ary123   | Aryan   | XII-F | 144534          | student   | M      |
| rohan    | roh123   | Rohan   | XII-F | 106073          | student   | M      |
+----------+----------+---------+-------+-----------------+-----------+--------+
```

ATTENDANCE

Admission No.	Student's Name	Date	Attendance(P/A)
106073	Rohan	2022-02-07	P

Exit

Admission_No	Student_Name	Date_of_Attendance	Attendance
106073	Rohan	2022-02-07	P

MARKSHEET

Student's Name	Physics	Chemistry	Mathematics	English	Computer Science	Total Marks
Rohan	89	97	92	86	79	443

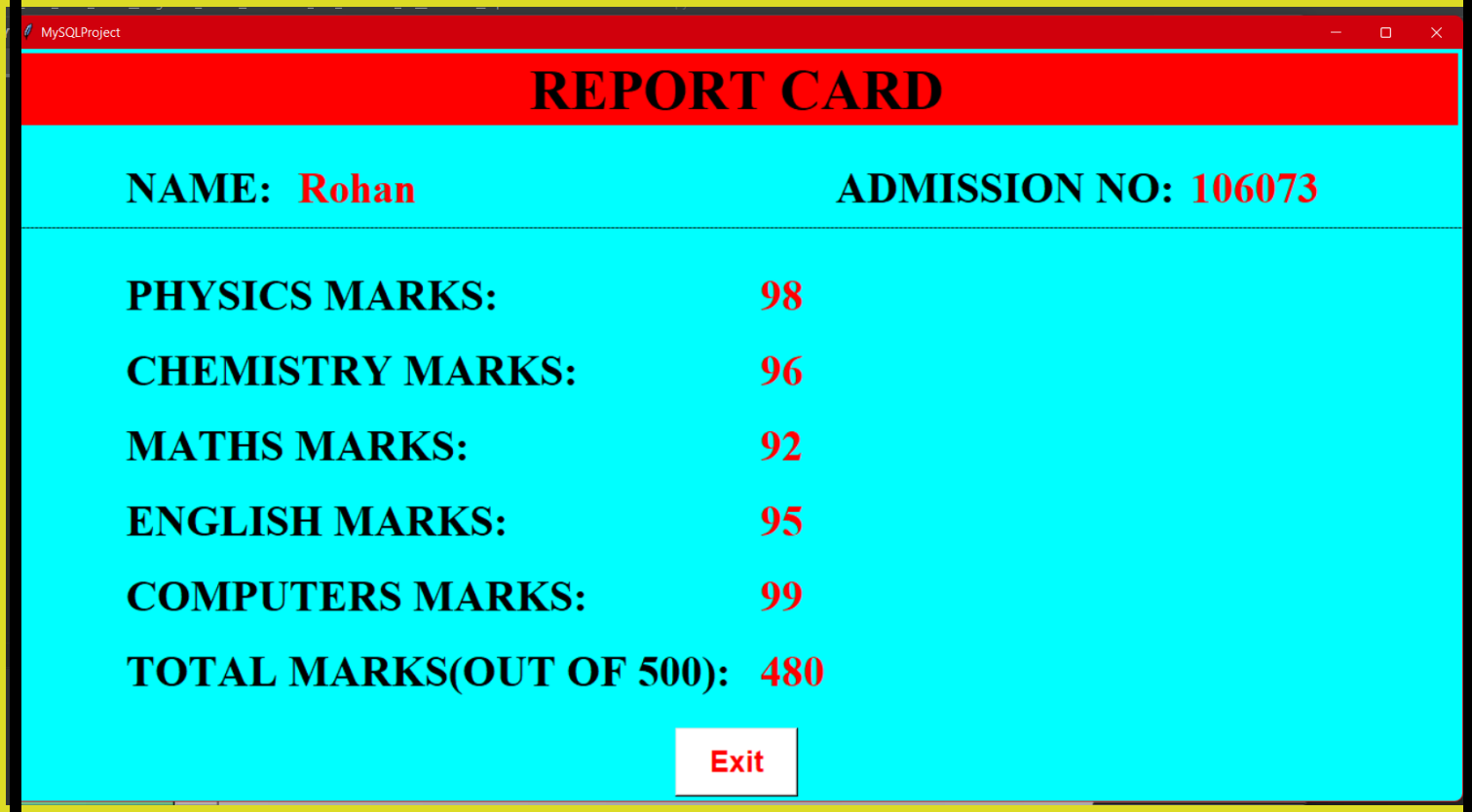
Exit

```
+-------------+---------------+-----------------+-------------------+
| Student_Name | Physics_marks | Chemistry_marks | Mathematics_marks |
+-------------+---------------+-----------------+-------------------+
| Rohan       |            89 |              97 |                92 |
+-------------+---------------+-----------------+-------------------+
1 row in set (0.00 sec)
```

```
+---------------+----------+-------------------------------+
| English_marks | Cs_marks | TotalMarksObtained_Outof500 |
+---------------+----------+-------------------------------+
|            86 |       79 |                           443 |
+---------------+----------+-------------------------------+
```

Notice

~Today all students have to submit their homework to the respective class teache
rs



Student_Name	Admission_No	Physics_marks	Chemistry_marks
Rohan	106073	98	96

Mathematics_marks	English_marks	Cs_marks	TotalMarksObtained_Outof500
92	95	99	480

ATTENDANCE

Admission Nc	Student's Nar	Date	Attendance(P/A)
106073	Rohan	2022-02-07	P

Exit

Student's Homework

Subject_Code	Subject	Date	Homework
041	Maths	2022-02-16	Do Ex-1 of Intergration

Exit

```
+--------------+---------+------------+--------------------------+
| Subject_Code | Subject | Date       | Homework                 |
+--------------+---------+------------+--------------------------+
| 041          | Maths   | 2022-02-16 | Do Ex-1 of Intergration  |
 |
+--------------+---------+------------+--------------------------+
```

MYSQLProject

ADMISSION NOTICE

NEW ERA PUBLIC SCHOOL

(Certified ISO 9001 : 2015)

MAYAPURI ROAD, NEW DELHI - 110064

Phones : 25494277, 25494439, 25495579

E-mail : uchpneps@gmail.com

Website : www.newerapublicschool.in

NEW ERA PUBLIC SCHOOL

61 years of Excellence in Education

International Dimension in Schools 2020-23

Mission

Character | Commitment | Consistency

Vision

No Child Left Behind

REGISTRATION OPEN

SESSION 2022-23

ENTRY LEVEL (CLASS - I)

Contact School office for Registration Form between 10 a.m. and 12 noon on all working days till 07.01.2022

- Age Criteria : 5 Plus as on 31st March, 2022
- EWS & Disadvantaged group - 25% seats as per rules

Features :

Air Conditioned State of the Art :

Libraries | Auditoriums | ATL Tinkering Lab | AV Rooms | English Language Lab | Maths Lab

Plethora of Sports Facilities :

Basketball | Volleyball | Badminton | Table Tennis | Football | Cricket | Martial Arts

PythonGuides

New Era Public School, is a school in Mayapuri, New Delhi, India.

It was founded in 1960 by the New Era Education Society in Delhi.

The project of Mr. R.L. Chopra and Mrs. Usha Chopra came into existence in 1965.

The school is an All India Senior Secondary school, affiliated with the Central Board of Secondary Education

The school came into existence in one room in the year 1960.

The school has another branch in Pochampur, Dwarka with the same name.

OUR MISSION

Our primary objective is to facilitate the all round development of all children in a safe and nurturing environment,

enabling every child to develop into a well-adjusted, confident, free-thinking global citizen.

```
mysql> use school;
Database changed
mysql> show tables
    -> ;
+----------------------+
| Tables_in_school     |
+----------------------+
| login                |
| loginstudent         |
| loginteacher         |
| student_attendance   |
| student_competition  |
| student_hw           |
| student_marksheet    |
| student_report       |
+----------------------+
8 rows in set (0.01 sec)
```

```
mysql> select * from login;
+-----------+----------+----------------+
| Username  | Password | Logintype      |
+-----------+----------+----------------+
| jasjeev   | jas123   | student        |
| aryan     | ary123   | student        |
| lalit     | lal123   | teacher        |
| technical | tec123   | technical team |
| rohan     | roh123   | student        |
| vivek     | viv123   | teacher        |
+-----------+----------+----------------+
6 rows in set (0.00 sec)
```

```
mysql> select * from loginstudent;
+----------+----------+---------+-------+-----------------+-----------+--------+
| Username | Password | Name    | Class | Admissionnumber | Logintype | Gender |
+----------+----------+---------+-------+-----------------+-----------+--------+
| jasjeev  | jas123   | Jasjeev | XII-F | 100073          | student   | M      |
| aryan    | ary123   | Aryan   | XII-F | 144534          | student   | M      |
| rohan    | roh123   | Rohan   | XII-F | 106073          | student   | M      |
+----------+----------+---------+-------+-----------------+-----------+--------+
3 rows in set (0.00 sec)

mysql> select * from loginteacher;
+----------+----------+---------+---------+-----------+--------+
| Username | Password | Name    | Subject | Logintype | Gender |
+----------+----------+---------+---------+-----------+--------+
| lalit    | lal123   | Lalit   | CS      | teacher   | M      |
| teacher  | tea123   | Teacher | Maths   | teacher   | F      |
| vivek    | viv123   | Vivek   | Maths   | teacher   | M      |
+----------+----------+---------+---------+-----------+--------+
3 rows in set (0.01 sec)
```

```
mysql> select * from student_attendance;
+--------------+--------------+--------------------+------------+
| Admission_No | Student_Name | Date_of_Attendance | Attendance |
+--------------+--------------+--------------------+------------+
| 106073       | Rohan        | 2022-02-07         | P          |
+--------------+--------------+--------------------+------------+
1 row in set (0.00 sec)

mysql> select * from student_competition;
+---------------+----------------------+--------------+----------------------------------+
| hosted_by     | Name_of_competition  | Activity     | Details                          |
+---------------+----------------------+--------------+----------------------------------+
| Ms Nitin Neil | TT Tournament        | Table Tennis | It will be held on 25 February
 |
+---------------+----------------------+--------------+----------------------------------+
1 row in set (0.00 sec)
```

```
mysql> select * from student_hw;
+--------------+---------+------------+-------------------------+
| Subject_Code | Subject | Date       | Homework                |
+--------------+---------+------------+-------------------------+
| 041          | Maths   | 2022-02-16 | Do Ex-1 of Intergration
 |
+--------------+---------+------------+-------------------------+
1 row in set (0.00 sec)

mysql> select * from student_marksheet;
+--------------+---------------+-----------------+------------------+--------------+----------+---------------------------+
| Student_Name | Physics_marks | Chemistry_marks | Mathematics_marks | English_marks | Cs_marks | TotalMarksObtained_Outof500 |
+--------------+---------------+-----------------+------------------+--------------+----------+---------------------------+
| Rohan        |            89 |              97 |               92 |           86 |       79 |                       443 |
+--------------+---------------+-----------------+------------------+--------------+----------+---------------------------+
1 row in set (0.00 sec)
```

```
mysql> select * from student_report;
+--------------+-------------+---------------+-----------------+------------------+--------------+----------+---------------------------+
| Student_Name | Admission_No | Physics_marks | Chemistry_marks | Mathematics_marks | English_marks | Cs_marks | TotalMarksObtained_Outof500 |
+--------------+-------------+---------------+-----------------+------------------+--------------+----------+---------------------------+
| Rohan        |       106073 |            98 |              96 |               92 |           95 |       99 |                       480 |
+--------------+-------------+---------------+-----------------+------------------+--------------+----------+---------------------------+
1 row in set (0.00 sec)
```


Thank
you