Priyansh Salian

2003148

C31

Experiment 10

Problem Statement:

Python program to demonstrate MYSQL database connectivity with python. Create a GUI based application using widgets Entry, Label, Text, Button, RadioButton, CheckButton, ListBox, Menu, Spinbox **(any five)**.

Save the details in a database and read back from file on python prompt.

Theory:

MySql DB:

MySQL is **a relational database management system (RDBMS) developed by Oracle that is based on structured query language (SQL)**. A database is a structured collection of data. It may be anything from a simple shopping list to a picture gallery or a place to hold the vast amounts of information in a corporate network.

In particular, a relational database is a digital store collecting data and organizing it according to the relational model. In this model, tables consist of rows and columns, and relationships between data elements all follow a strict logical structure. An RDBMS is simply the set of software tools used to actually implement, manage, and query such a database.

MySQL is integral to many of the most popular software stacks for building and maintaining everything from customer-facing web applications to powerful, data-driven B2B services. Its open-source nature, stability, and rich feature set, paired with ongoing development and support from Oracle, have meant that internet-critical organizations such as Facebook, Flickr, Twitter, Wikipedia, and YouTube all employ MySQL backends.

Code :

```
from tkinter import *
from PIL import Image, ImageTk
from tkinter import ttk, messagebox
import pymysql, os
import credentials as cr

class SignUp:
    def __init__(self, root):
        self.window = root
        self.window.title("Sign Up")
        self.window.geometry("1280x800+0+0")
        self.window.config(bg = "white")

        self.bg_img = ImageTk.PhotoImage(file="Images/photo1.jpeg")
        background =
Label(self.window,image=self.bg_img).place(x=0,y=0,relwidth=1,relhei
ght=1)

        frame = Frame(self.window, bg="white")
        frame.place(x=350,y=100,width=500,height=550)

        title1 = Label(frame, text="Sign Up", font=("times new
roman",25,"bold"),bg="white").place(x=20, y=10)
        title2 = Label(frame, text="Join with us", font=("times new
roman",13),bg="white", fg="gray").place(x=20, y=50)

        f_name = Label(frame, text="First name",
font=("helvetica",15,"bold"),bg="white").place(x=20, y=100)
        l_name = Label(frame, text="Last name",
font=("helvetica",15,"bold"),bg="white").place(x=240, y=100)

        self.fname_txt = Entry(frame,font=("arial"))
        self.fname_txt.place(x=20, y=130, width=200)

        self.lname_txt = Entry(frame,font=("arial"))
        self.lname_txt.place(x=240, y=130, width=200)

        email = Label(frame, text="Email",
font=("helvetica",15,"bold"),bg="white").place(x=20, y=180)

        self.email_txt = Entry(frame,font=("arial"))
        self.email_txt.place(x=20, y=210, width=420)

        sec_question = Label(frame, text="Security questions",
font=("helvetica",15,"bold"),bg="white").place(x=20, y=260)
        answer = Label(frame, text="Answer",
font=("helvetica",15,"bold"),bg="white").place(x=240, y=260)
```

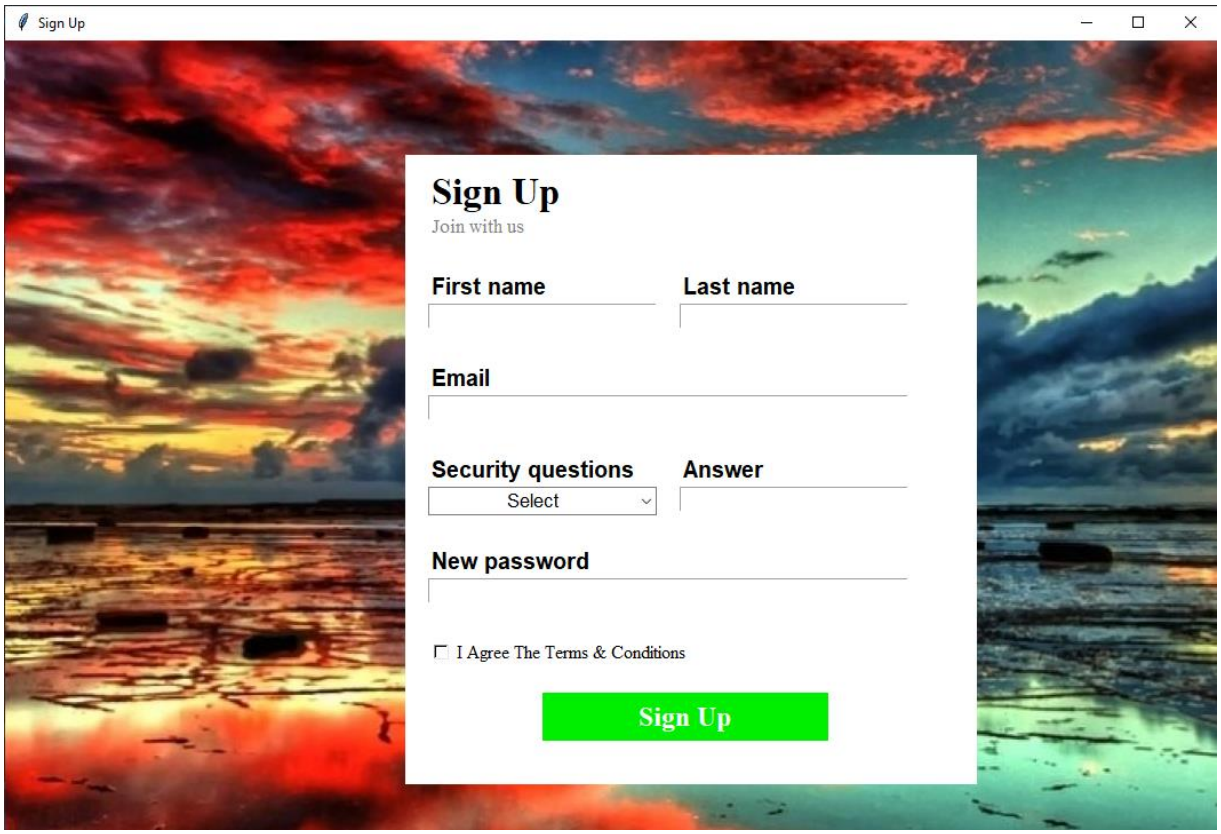
```
        self.questions =
ttk.Combobox(frame,font=("helvetica",13),state='readonly',justify=CE
NTER)
        self.questions['values'] = ("Select","What's your pet
name?","Your first teacher name","Your birthplace", "Your favorite
movie")
        self.questions.place(x=20,y=290,width=200)
        self.questions.current(0)

        self.answer_txt = Entry(frame,font=("arial"))
        self.answer_txt.place(x=240, y=290, width=200)

        password =  Label(frame, text="New password",
font=("helvetica",15,"bold"),bg="white").place(x=20, y=340)

        self.password_txt = Entry(frame,font=("arial"))
        self.password_txt.place(x=20, y=370, width=420)

        self.terms = IntVar()
        terms_and_con = Checkbutton(frame,text="I Agree The Terms &
Conditions",variable=self.terms,onvalue=1,offvalue=0,bg="white",font
=("times new roman",12)).place(x=20,y=420)
        self.signup = Button(frame,text="Sign
Up",command=self.signup_func,font=("times new roman",18,
"bold"),bd=0,cursor="hand2",bg="green2",fg="white").place(x=120,y=47
0,width=250)

    def signup_func(self):
        if self.fname_txt.get()=="" or self.lname_txt.get()=="" or
self.email_txt.get()=="" or self.questions.get()=="Select" or
self.answer_txt.get()=="" or self.password_txt.get() == "":
            messagebox.showerror("Error!","Sorry!, All fields are
required",parent=self.window)

        elif self.terms.get() == 0:
            messagebox.showerror("Error!","Please Agree with our
Terms & Conditions",parent=self.window)

        else:
            try:
                connection = pymysql.connect(host=cr.host,
user=cr.user, password=cr.password, database=cr.database)
                cur = connection.cursor()
                cur.execute("select * from student_register where
email=%s",self.email_txt.get())
                row=cur.fetchone()

                # Check if th entered email id is already exists or
not.
                if row!=None:
```

```
                    messagebox.showerror("Error!","The email id is
already exists, please try again with another email
id",parent=self.window)
                else:
                    cur.execute("insert into student_register
(f_name,l_name,email,question,answer,password)
values(%s,%s,%s,%s,%s,%s)",
                                        (
                                            self.fname_txt.get(),
                                            self.lname_txt.get(),
                                            self.email_txt.get(),
                                            self.questions.get(),
                                            self.answer_txt.get(),
                                            self.password_txt.get()
                                        ))
                    connection.commit()
                    connection.close()
                    messagebox.showinfo("Congratulations!","Register
Successful",parent=self.window)
                    self.reset_fields()
            except Exception as e:
                messagebox.showerror("Error!",f"Error due to
{str(e)}",parent=self.window)

    def reset_fields(self):
        self.fname_txt.delete(0, END)
        self.lname_txt.delete(0, END)
        self.email_txt.delete(0, END)
        self.questions.current(0)
        self.answer_txt.delete(0, END)
        self.password_txt.delete(0, END)

if __name__ == "__main__":
    root = Tk()
    obj = SignUp(root)
    root.mainloop()
```

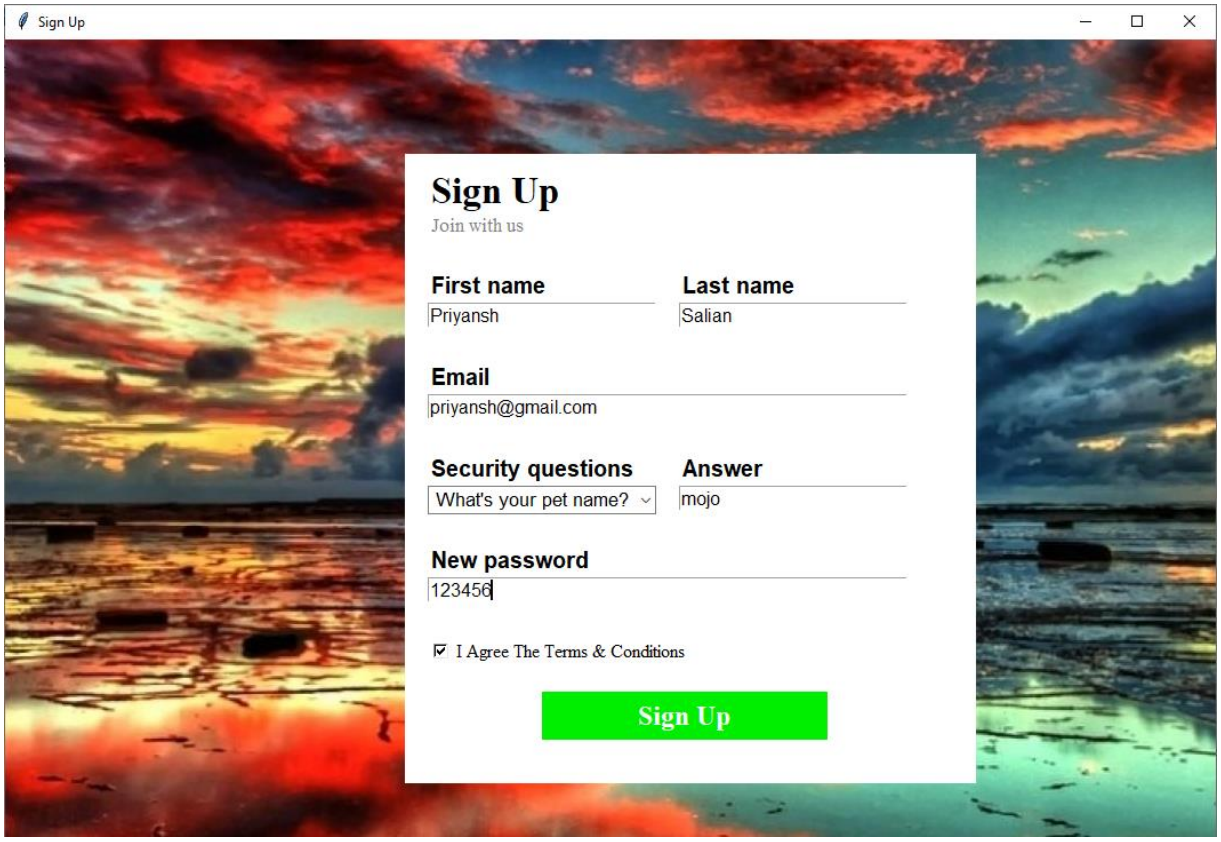
TERMINAL:

```
PS C:\Users\Admin\Desktop\PRIYANSH\College\PYTHON\CODE\EXPERIMENT 10\login-page-using-Python-and-MySQL-main> python -u "c:\Users\Admin\Desktop\PRIYANSH\C
ollege\PYTHON\CODE\EXPERIMENT 10\login-page-using-Python-and-MySQL-main\signup_page.py"
```

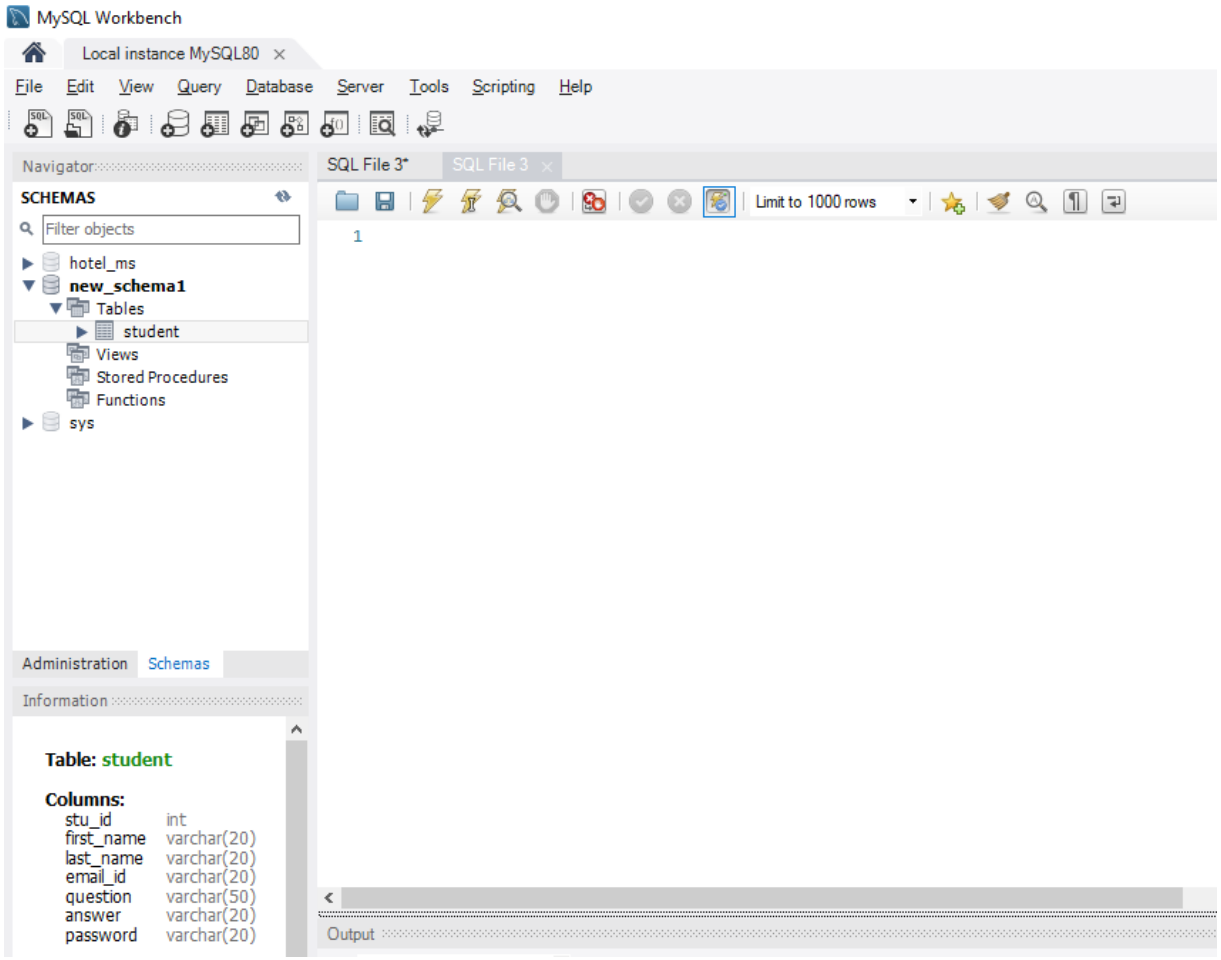
OutPut:



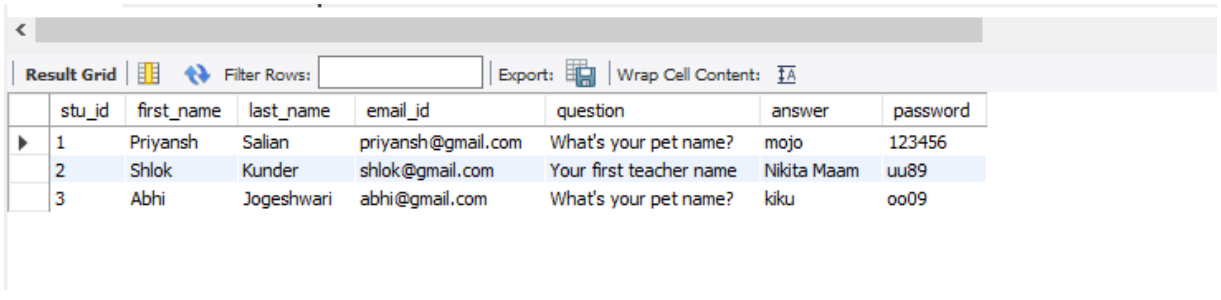
Filling Details:



MySql Workbench:



After inserting data:



Result Grid | Filter Rows: | Export: | Wrap Cell Content:

stu_id	first_name	last_name	email_id	question	answer	password
1	Priyansh	Salian	priyansh@gmail.com	What's your pet name?	mojo	123456
2	Shlok	Kunder	shlok@gmail.com	Your first teacher name	Nikita Maam	uu89
3	Abhi	Jogeshwari	abhi@gmail.com	What's your pet name?	kiku	oo09