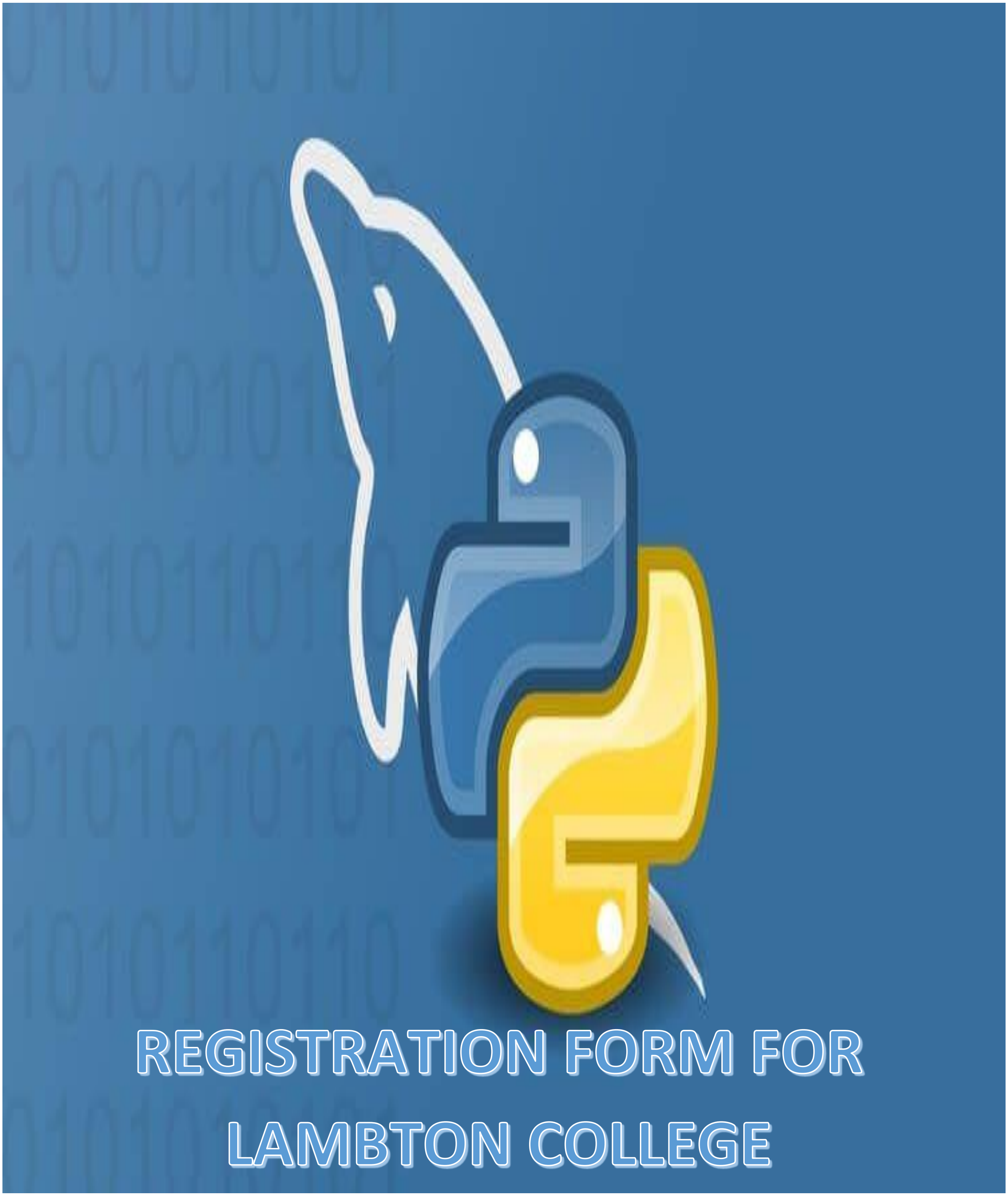
PYTHON 2



REGISTRATION FORM FOR LAMBTON COLLEGE

TABLE OF CONTENTS

Project description-- 3

Group members-- 4

Project scope

- Tk interface---5-8
- SQLite database---9-12

Technology--13

- Pycharm
- Sqlite3

Group member's Task---13

Project description

In this project, we have created registration form for a college which is linked to database. Firstly, we made templates by using images and widgets and create registration form. This form includes first name, last name, contact no. email. Security questions and answers, passwords and term and conditions.

The name of this form is REGISTRATION FORM FOR LAMBTON COLLEGE.

We used PYCHARM and SQLLite3 IDE for this project.

We used libraries such as tkinter and pillow, sqlite3,OOP,database intersection.

At the end of this project, user can register by entering personal information. If user do not fill all the field the pop-up window will show that “all field are required”. if passwords do not match with other then pop-up window show “password and confirm password is not matching”.

Group members

Members:

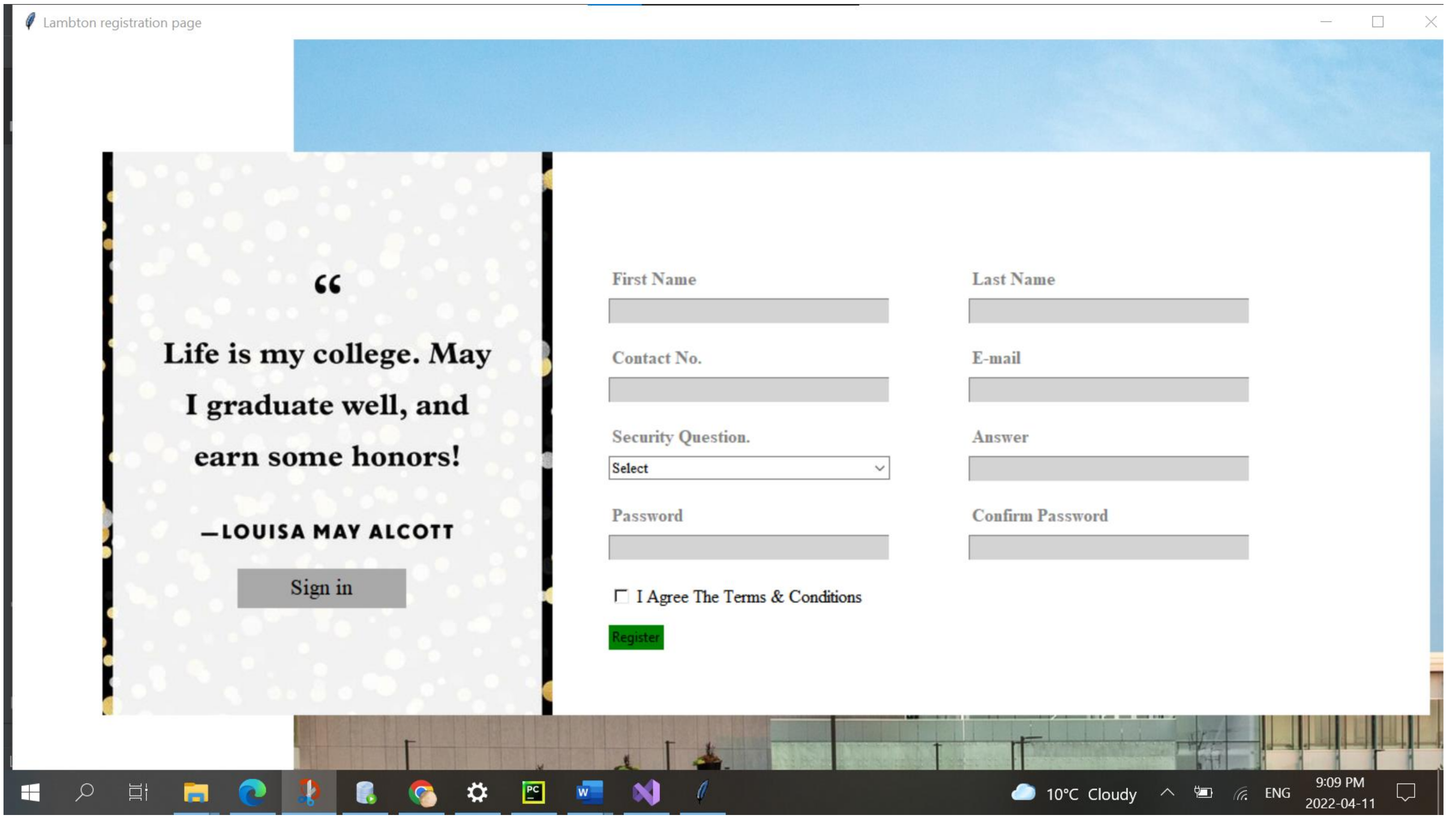
- AMANDEEP (C0840378)
- GURUSHARAN SINGH (C0828270)
- PRABHJOT KAUR KINGRA (C0834571)
- ROMANJIT KAUR (C0798336)
- SRUTHY AMEY SAMUEL (C0834672)

Leader: SRUTHY AMEY SAMUEL

Project scope

Tk interface

By using python libraries, we created registration form which is shown below



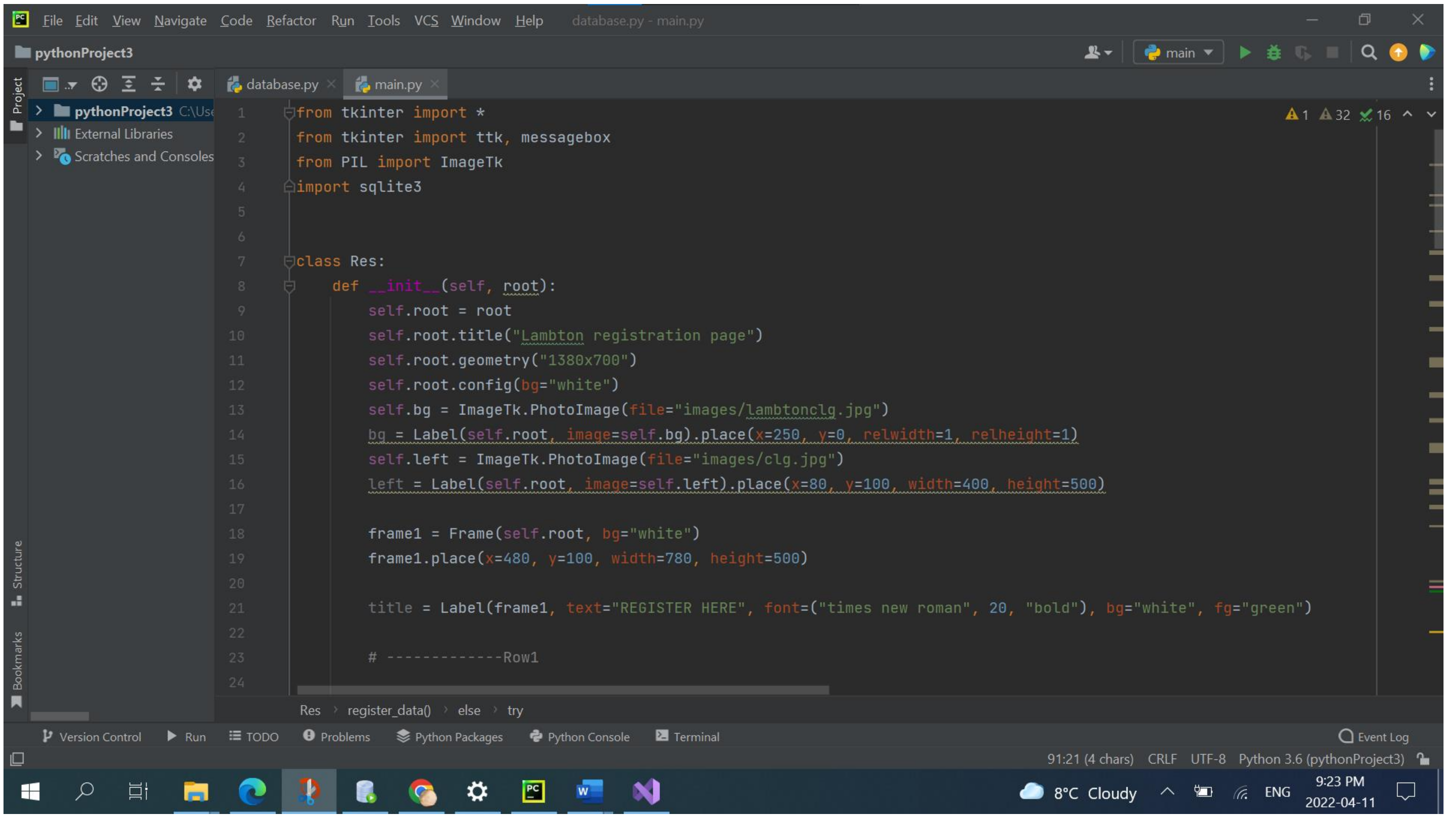
“

Life is my college. May I graduate well, and earn some honors!

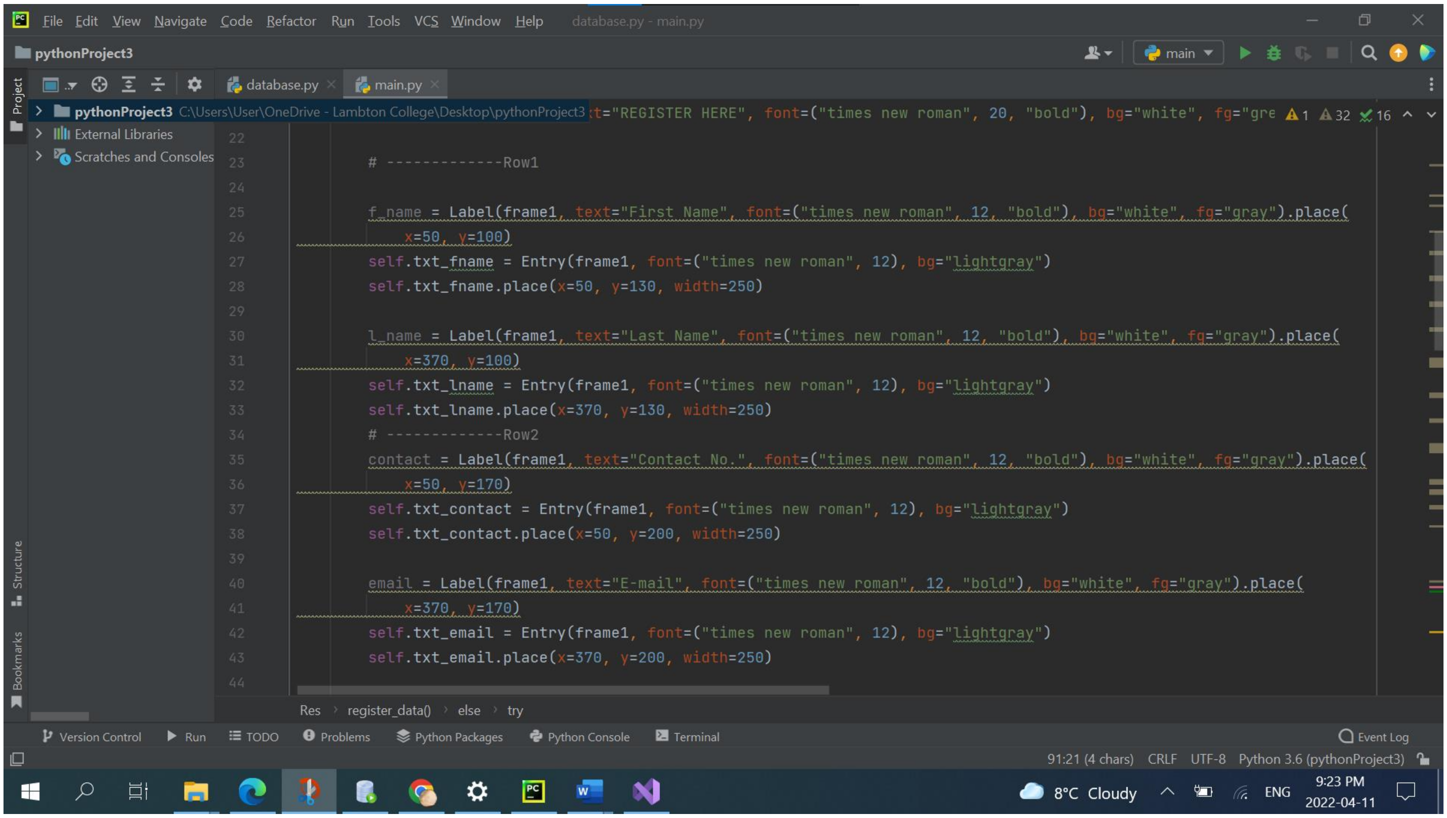
—LOUISA MAY ALCOTT

In the python code, we import tkinter , PIL for import images ,ttk for combo boxes .we made a class named as RES and use constructor def_init_ so that we call an object which is created from a class. In our code, root is an object of class you can see in the main.py program.

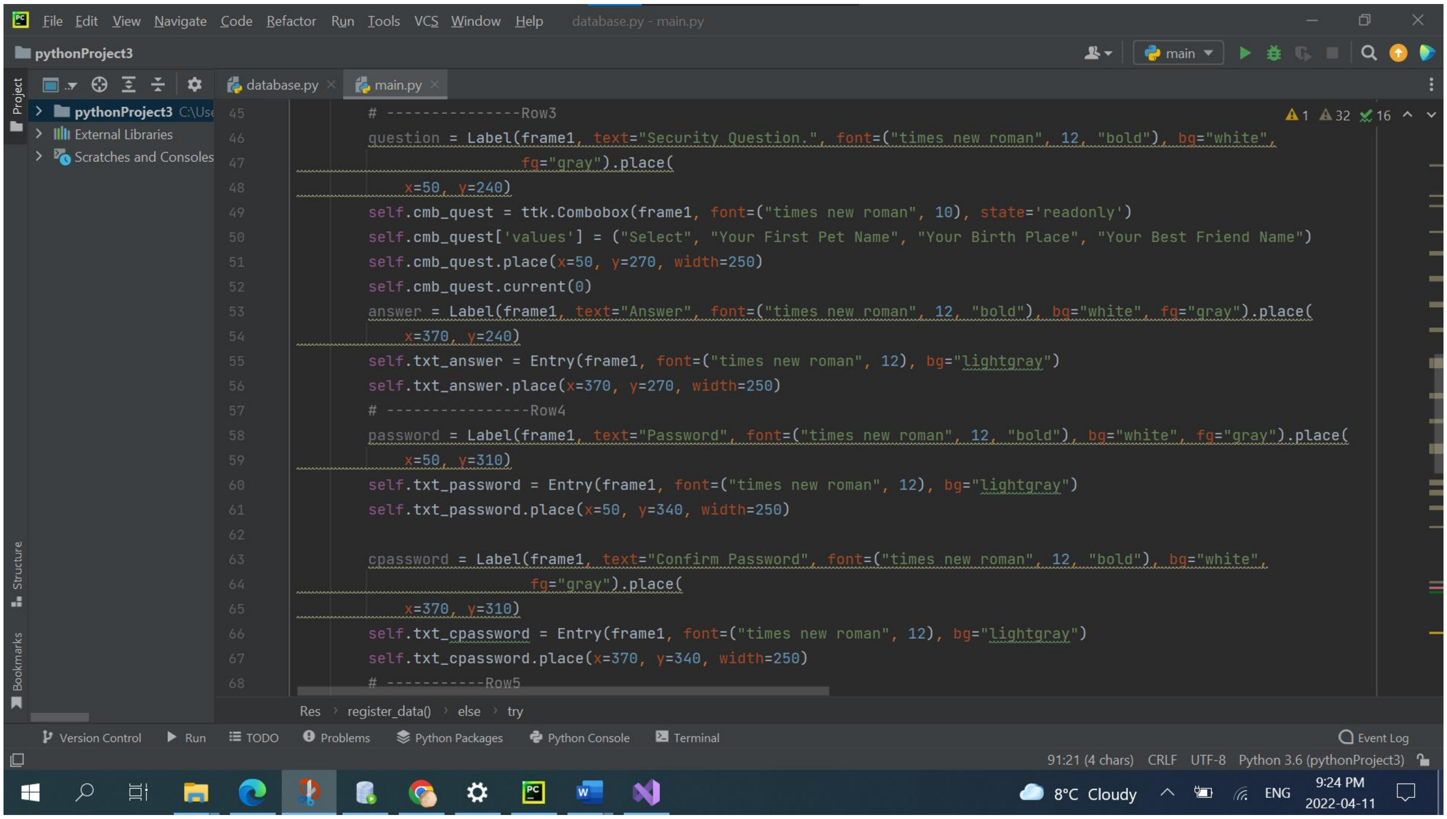
Python code



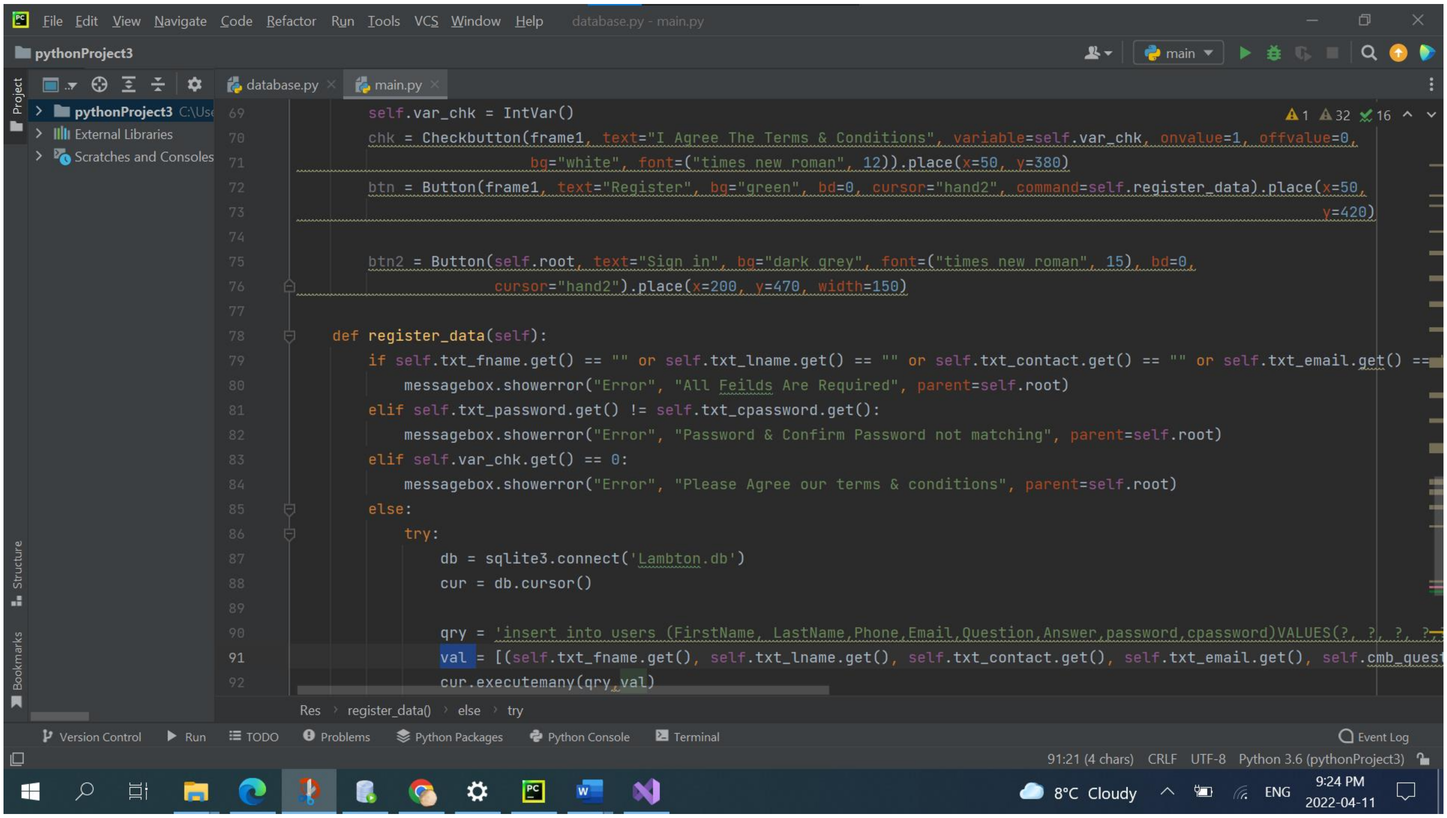
```python
from tkinter import *
from tkinter import ttk, messagebox
from PIL import ImageTk
import sqlite3


class Res:
    def __init__(self, root):
        self.root = root
        self.root.title("Lambton registration page")
        self.root.geometry("1380x700")
        self.root.config(bg="white")
        self.bg = ImageTk.PhotoImage(file="images/lambtonclg.jpg")
        bg = Label(self.root, image=self.bg).place(x=250, y=0, relwidth=1, relheight=1)
        self.left = ImageTk.PhotoImage(file="images/clg.jpg")
        left = Label(self.root, image=self.left).place(x=80, y=100, width=400, height=500)

        frame1 = Frame(self.root, bg="white")
        frame1.place(x=480, y=100, width=780, height=500)

        title = Label(frame1, text="REGISTER HERE", font=("times new roman", 20, "bold"), bg="white", fg="green")

        # ------------Row1

```



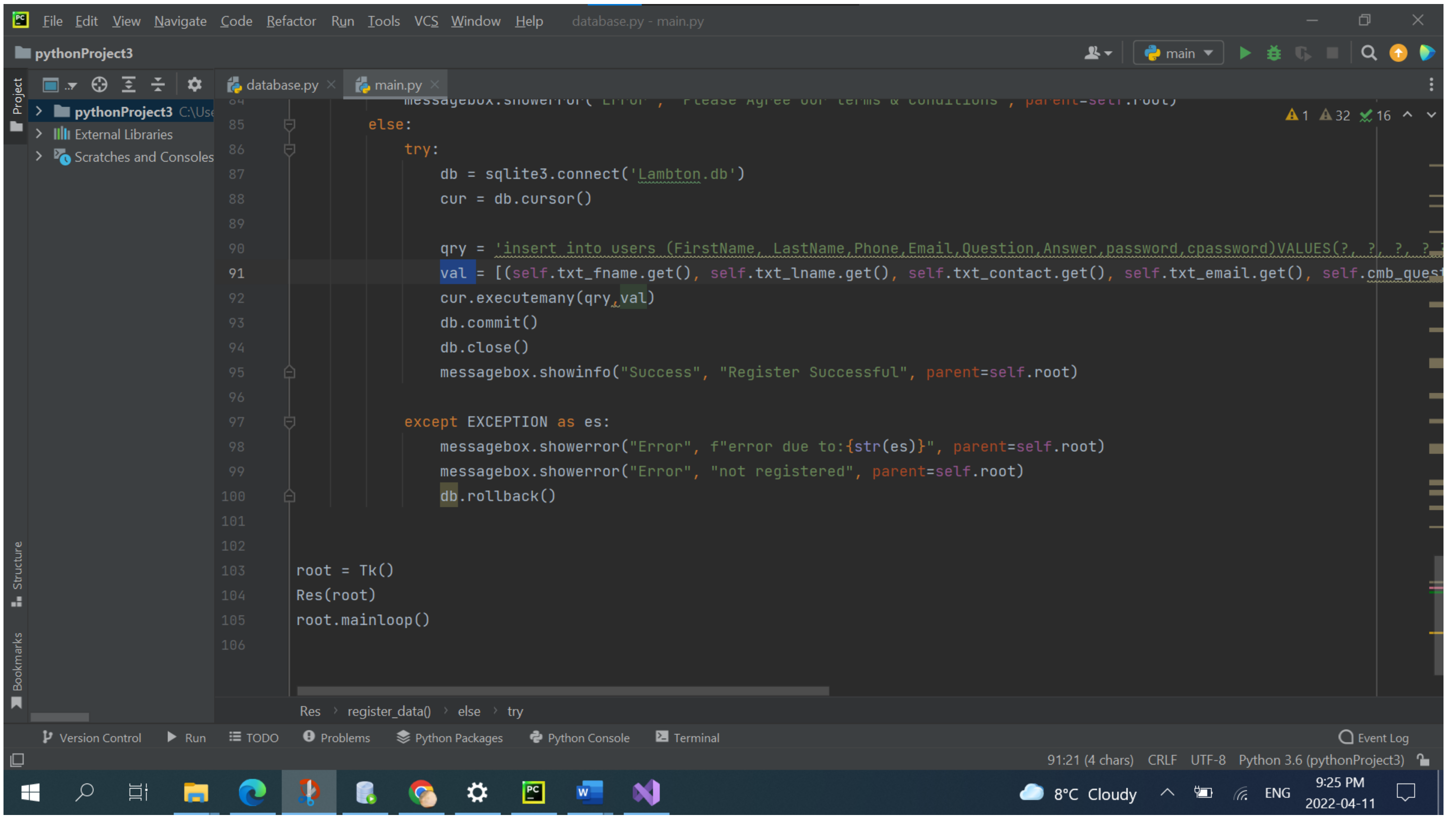
```
t="REGISTER HERE", font=("times new roman", 20, "bold"), bg="white", fg="gre

        # ------------Row1

        f_name = Label(frame1, text="First Name", font=("times new roman", 12, "bold"), bg="white", fg="gray").place(
            x=50, y=100)
        self.txt_fname = Entry(frame1, font=("times new roman", 12), bg="lightgray")
        self.txt_fname.place(x=50, y=130, width=250)

        l_name = Label(frame1, text="Last Name", font=("times new roman", 12, "bold"), bg="white", fg="gray").place(
            x=370, y=100)
        self.txt_lname = Entry(frame1, font=("times new roman", 12), bg="lightgray")
        self.txt_lname.place(x=370, y=130, width=250)
        # ------------Row2
        contact = Label(frame1, text="Contact No.", font=("times new roman", 12, "bold"), bg="white", fg="gray").place(
            x=50, y=170)
        self.txt_contact = Entry(frame1, font=("times new roman", 12), bg="lightgray")
        self.txt_contact.place(x=50, y=200, width=250)

        email = Label(frame1, text="E-mail", font=("times new roman", 12, "bold"), bg="white", fg="gray").place(
            x=370, y=170)
        self.txt_email = Entry(frame1, font=("times new roman", 12), bg="lightgray")
        self.txt_email.place(x=370, y=200, width=250)

```



```
        # --------------Row3
        question = Label(frame1, text="Security Question.", font=("times new roman", 12, "bold"), bg="white",
                         fg="gray").place(
            x=50, y=240)
        self.cmb_quest = ttk.Combobox(frame1, font=("times new roman", 10), state='readonly')
        self.cmb_quest['values'] = ("Select", "Your First Pet Name", "Your Birth Place", "Your Best Friend Name")
        self.cmb_quest.place(x=50, y=270, width=250)
        self.cmb_quest.current(0)
        answer = Label(frame1, text="Answer", font=("times new roman", 12, "bold"), bg="white", fg="gray").place(
            x=370, y=240)
        self.txt_answer = Entry(frame1, font=("times new roman", 12), bg="lightgray")
        self.txt_answer.place(x=370, y=270, width=250)
        # ---------------Row4
        password = Label(frame1, text="Password", font=("times new roman", 12, "bold"), bg="white", fg="gray").place(
            x=50, y=310)
        self.txt_password = Entry(frame1, font=("times new roman", 12), bg="lightgray")
        self.txt_password.place(x=50, y=340, width=250)

        cpassword = Label(frame1, text="Confirm Password", font=("times new roman", 12, "bold"), bg="white",
                          fg="gray").place(
            x=370, y=310)
        self.txt_cpassword = Entry(frame1, font=("times new roman", 12), bg="lightgray")
        self.txt_cpassword.place(x=370, y=340, width=250)
        # ----------Row5
```



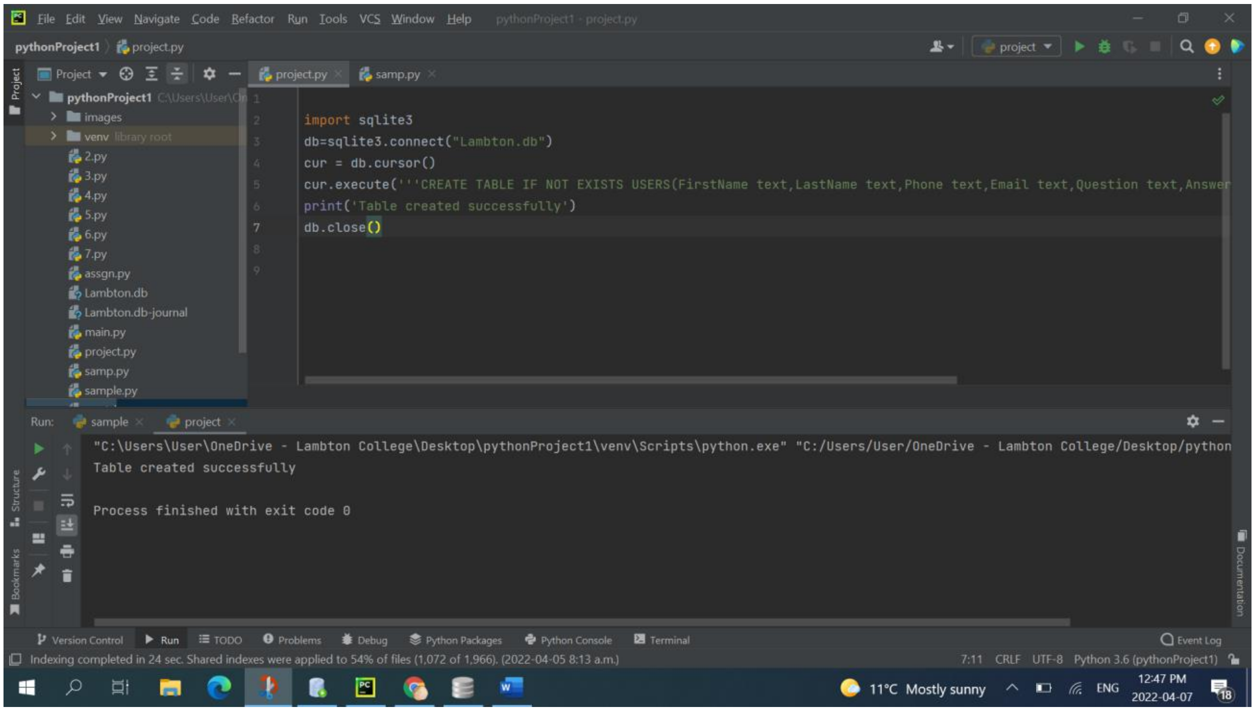
```
        self.var_chk = IntVar()
        chk = Checkbutton(frame1, text="I Agree The Terms & Conditions", variable=self.var_chk, onvalue=1, offvalue=0,
                                bg="white", font=("times new roman", 12)).place(x=50, y=380)
        btn = Button(frame1, text="Register", bg="green", bd=0, cursor="hand2", command=self.register_data).place(x=50,
                                                                                                                  y=420)

        btn2 = Button(self.root, text="Sign in", bg="dark grey", font=("times new roman", 15), bd=0,
                      cursor="hand2").place(x=200, y=470, width=150)

    def register_data(self):
        if self.txt_fname.get() == "" or self.txt_lname.get() == "" or self.txt_contact.get() == "" or self.txt_email.get() ==
            messagebox.showerror("Error", "All Feilds Are Required", parent=self.root)
        elif self.txt_password.get() != self.txt_cpassword.get():
            messagebox.showerror("Error", "Password & Confirm Password not matching", parent=self.root)
        elif self.var_chk.get() == 0:
            messagebox.showerror("Error", "Please Agree our terms & conditions", parent=self.root)
        else:
            try:
                db = sqlite3.connect('Lambton.db')
                cur = db.cursor()

                qry = 'insert into users (FirstName, LastName,Phone,Email,Question,Answer,password,cpassword)VALUES(?, ?, ?, ?
                val = [(self.txt_fname.get(), self.txt_lname.get(), self.txt_contact.get(), self.txt_email.get(), self.cmb_quest
                cur.executemany(qry,val)
```



```
            messagebox.showerror("Error", "Please Agree our terms & conditions", parent=self.root)
        else:
            try:
                db = sqlite3.connect('Lambton.db')
                cur = db.cursor()

                qry = 'insert into users (FirstName, LastName,Phone,Email,Question,Answer,password,cpassword)VALUES(?, ?, ?, ?
                val = [(self.txt_fname.get(), self.txt_lname.get(), self.txt_contact.get(), self.txt_email.get(), self.cmb_quest
                cur.executemany(qry,val)
                db.commit()
                db.close()
                messagebox.showinfo("Success", "Register Successful", parent=self.root)

            except EXCEPTION as es:
                messagebox.showerror("Error", f"error due to:{str(es)}", parent=self.root)
                messagebox.showerror("Error", "not registered", parent=self.root)
                db.rollback()


root = Tk()
Res(root)
root.mainloop()
```

SQLite database

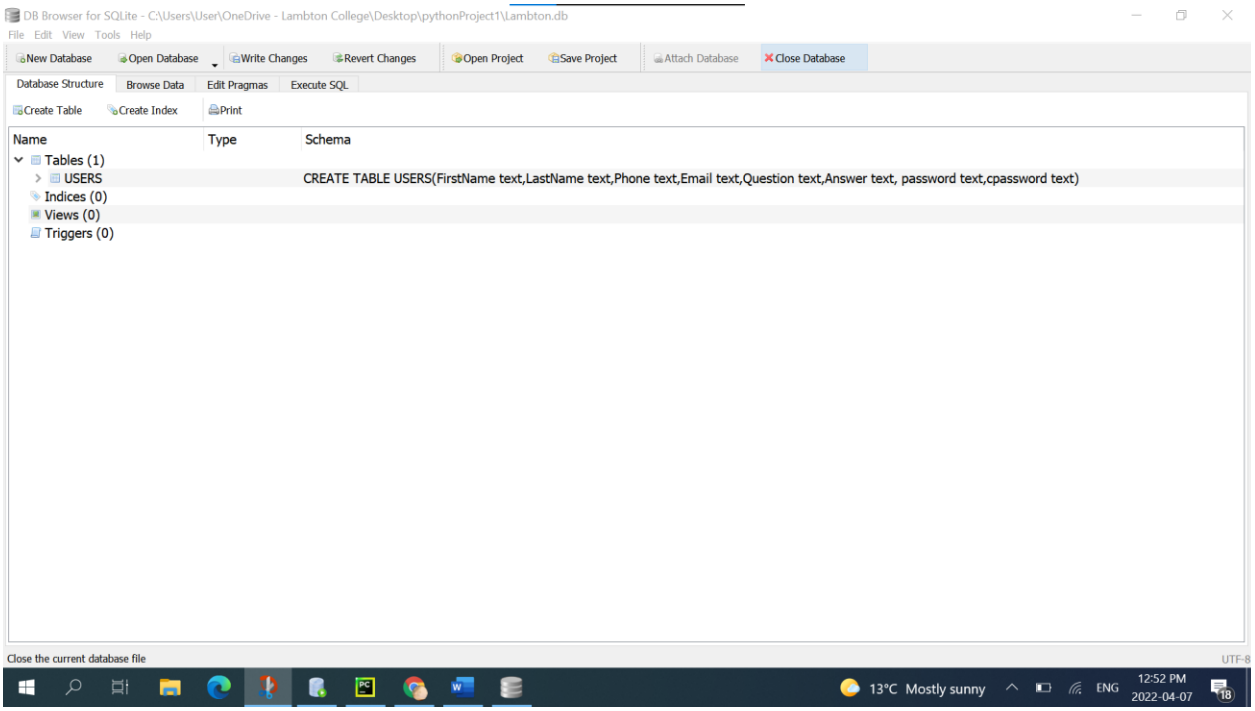
After python code we connected it with database. When user enter any entry in the form we can fetch it and stored in the database .



```
import sqlite3
db=sqlite3.connect("Lambton.db")
cur = db.cursor()
cur.execute('''CREATE TABLE IF NOT EXISTS USERS(FirstName text,LastName text,Phone text,Email text,Question text,Answer
print('Table created successfully')
db.close()
```

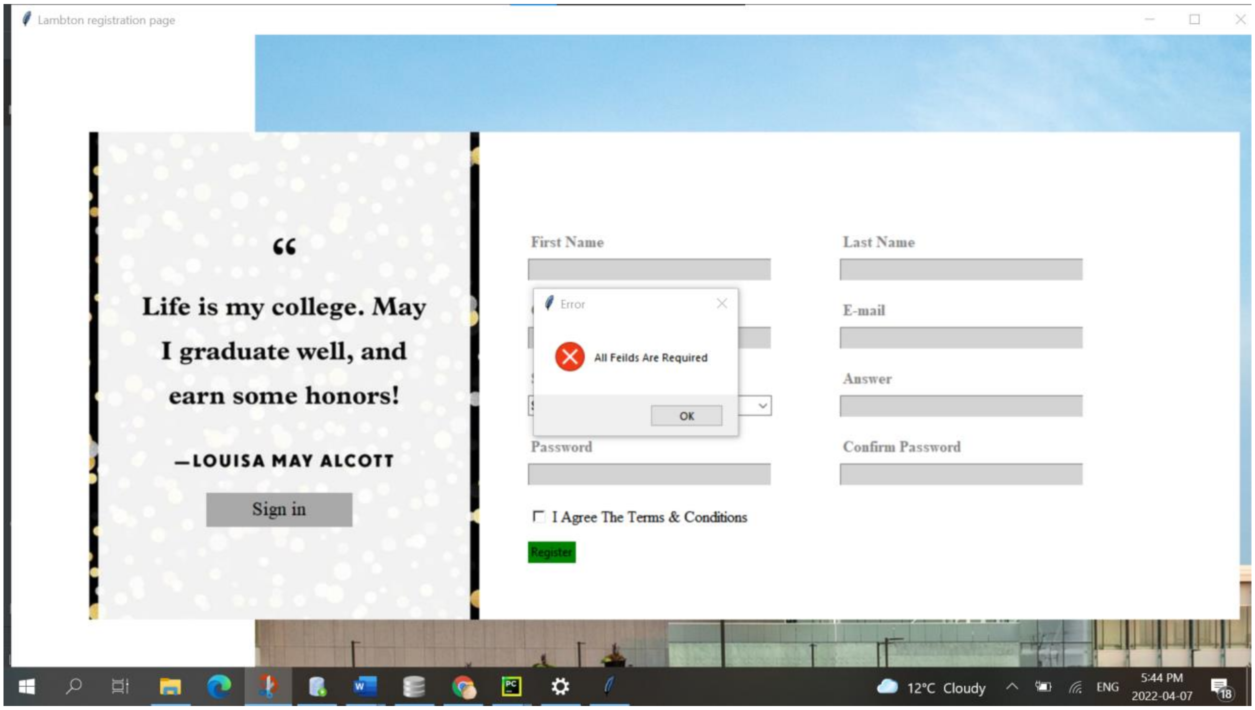
```
"C:\Users\User\OneDrive - Lambton College\Desktop\pythonProject1\venv\Scripts\python.exe" "C:/Users/User/OneDrive - Lambton College/Desktop/python
Table created successfully

Process finished with exit code 0
```

In the above screenshot we connect PyCharm to sqlite3 and create table. Our table is created successfully.



DB Browser for SQLite - C:\Users\User\OneDrive - Lambton College\Desktop\pythonProject1\Lambton.db

Name	Type	Schema
Tables (1)		
USERS		CREATE TABLE USERS(FirstName text,LastName text,Phone text,Email text,Question text,Answer text, password text,cpassword text)
Indices (0)		
Views (0)		
Triggers (0)		



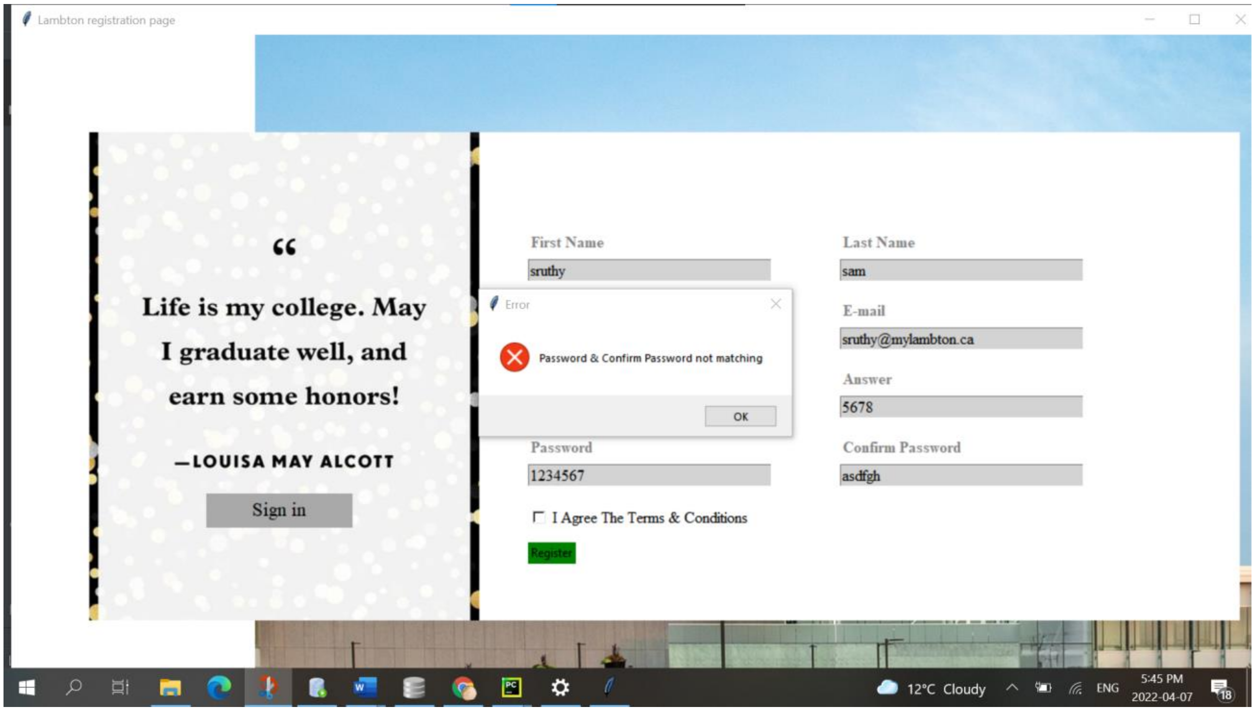
Error

All Feilds Are Required

OK

□ I Agree The Terms & Conditions

In the above screenshot if the user do not fill all entries , a pop-up box will appear “all fields are required”.


Lambton registration page
“
Life is my college. May I graduate well, and earn some honors!
—LOUISA MAY ALCOTT
Sign in
Register
12°C Cloudy
ENG
5:45 PM
2022-04-07

First Name: sruthy

Last Name: sam

E-mail: sruthy@mylambton.ca

Answer: 5678

Password: 1234567

Confirm Password: asdfgh

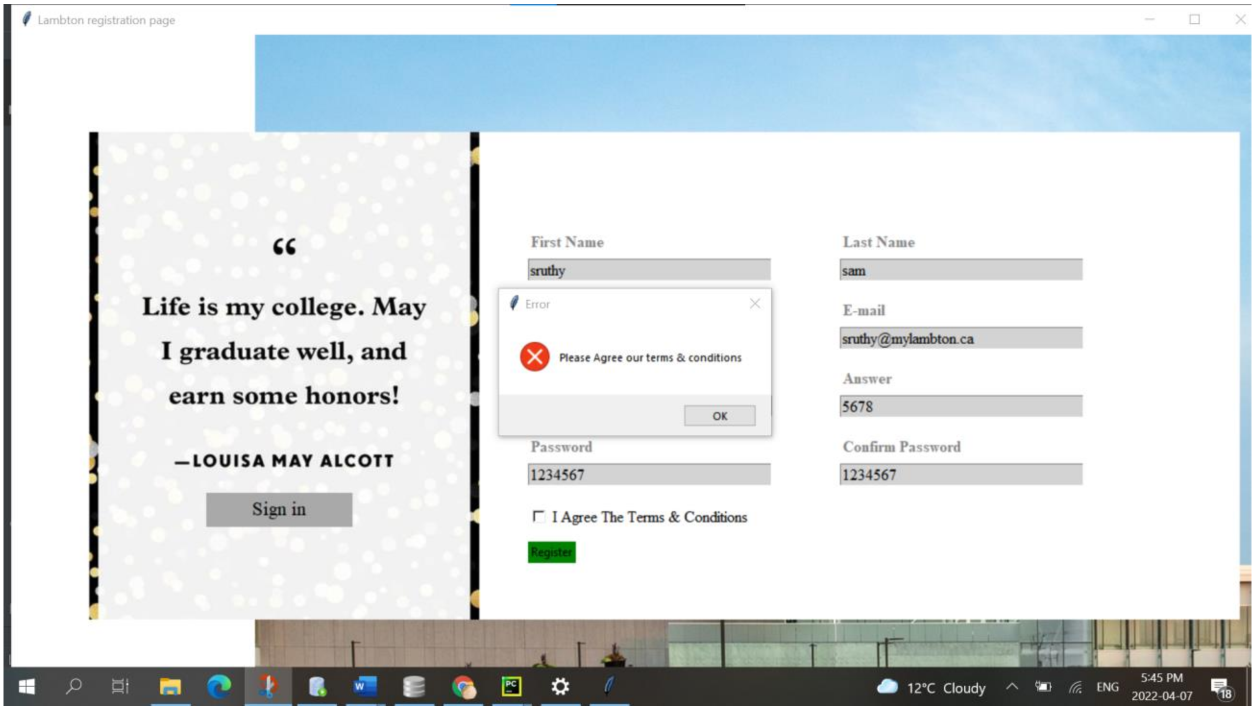
Error

Password & Confirm Password not matching

OK

□ I Agree The Terms & Conditions

If the password and confirm password not matching.



Life is my college. May I graduate well, and earn some honors!

—LOUISA MAY ALCOTT

First Name: sruthy

Last Name: sam

E-mail: sruthy@mylambton.ca

Answer: 5678

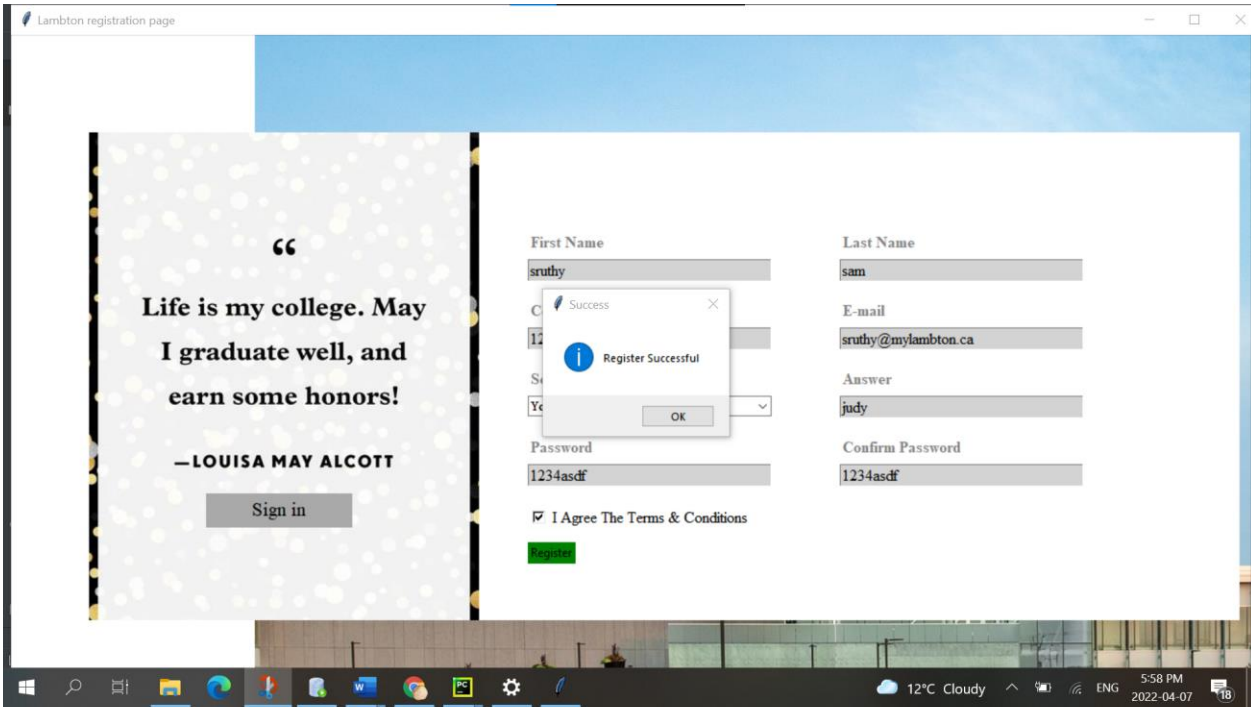
Password: 1234567

Confirm Password: 1234567

□ I Agree The Terms & Conditions

Please Agree our terms & conditions

If user do not select terms and condition.



Life is my college. May I graduate well, and earn some honors!

—LOUISA MAY ALCOTT

First Name: sruthy

Last Name: sam

E-mail: sruthy@mylambton.ca

Answer: judy

Password: 1234asdf

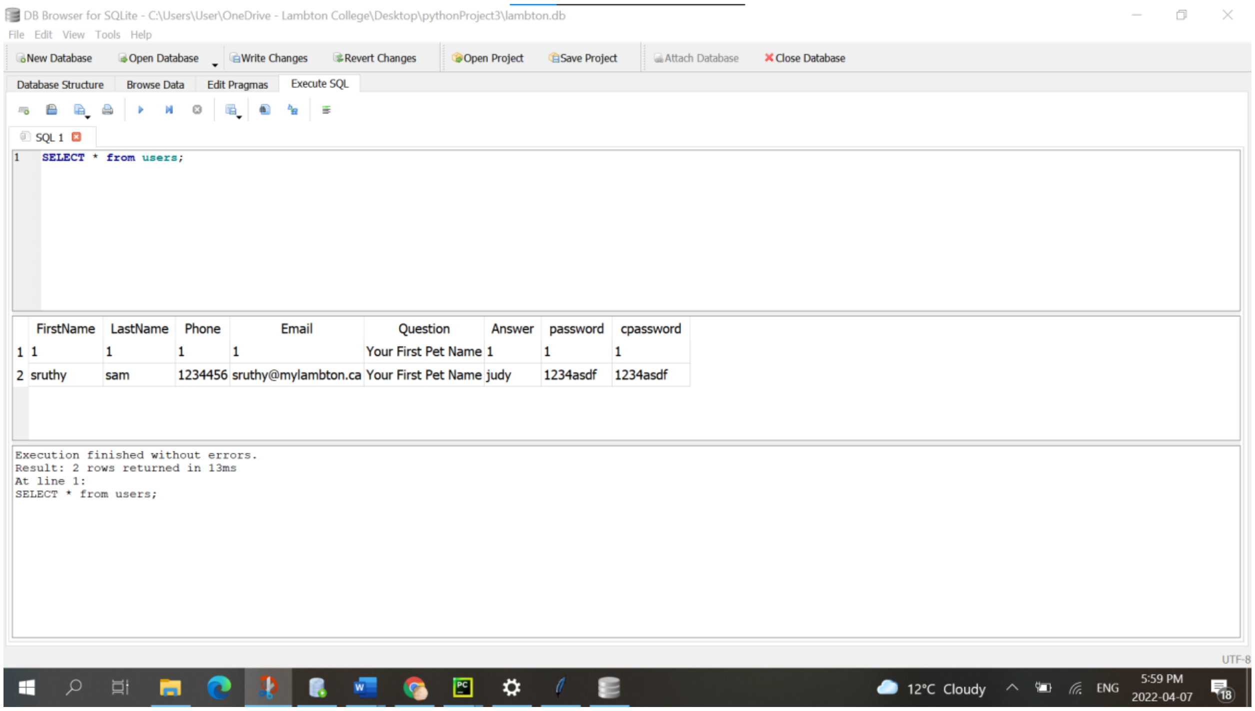
Confirm Password: 1234asdf

☑ I Agree The Terms & Conditions

Register Successful

In the above screenshot, user entered all entries and pop-up window shows register successful.

By using SELECT statement we can check user's entries which is inserted in our database. In the below image you can see



```sql
SELECT * from users;
```

	FirstName	LastName	Phone	Email	Question	Answer	password	cpassword
1	1	1	1	1	Your First Pet Name	1	1	1
2	sruthy	sam	1234456	sruthy@mylambton.ca	Your First Pet Name	judy	1234asdf	1234asdf

```
Execution finished without errors.
Result: 2 rows returned in 13ms
At line 1:
SELECT * from users;
```

TECHNOLOGY

For this project we used PyCharm and sqlite3 db browser

Tasks

All Group members referred to different materials and collected ideas to make the interface more interactive.

OOP Concepts:

Amandeep

Gursharan Singh

GUI framework: Romanjit Kaur

Sruthy Amey Samuel

Database Connectivity: Prabhjot Kaur Kingra