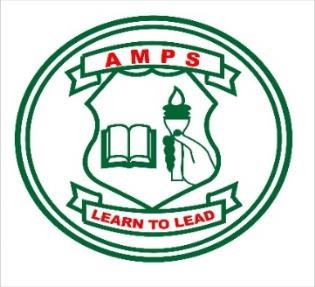
**ALWIN MEMORIAL PUBLIC SCHOOL**

# IndiraNagar, Selaiyur, Chennai–600073



AISSCE 2022–2023

**NAME: Joseph Mathew Pazhayakalam**

**REGISTER NO:**

**STD: XII**

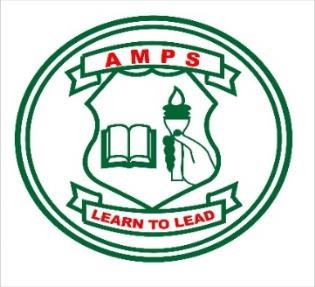
**SEC: C**

**SUBJECT: Computer Science**

**Student Information Management System**

**ALWIN MEMORIAL PUBLIC SCHOOL**

# IndiraNagar, Selaiyur, Chennai–600073



**BONAFIDE CERTIFICATE**

# This is to certify that the project titled *Student Information Management System* is a Bonafide work done by *Joseph Mathew Pazhayakalam* in partial fulfillment of the requirements in Computer Science practical as prescribed by CBSE for AISSCE 2022– 2023.

**REGISTER NO.**

# TEACHER-IN-CHARGE PRINCIPAL

# INTERNAL EXAMINER EXTERNAL EXAMINER

**TABLE OF CONTENT:**

**1. Acknowledgment**

**2. Abstract**

**3. Requirement Analysis**

**3.1. Hardware requirements**

**3.2. Software requirements**

**4. Applications**

**5. Design**

**5.1 Block diagram**

**5.2 Flow chart**

**5.3 Algorithm**

**6. Source Code**

**7. Screen Shots**

**8. Future Enhancements**

**9. Bibliography**

**Acknowledgement**

We wish to express our sincere thanks towards our Beloved Principal

Mrs. D.V. Lavanya Alwin Memorial Public School for guiding us to cause the successful outcome of the project work.

We wish to our deep gratitude to our guide teacher Mrs.Subashini for the expert help & and valuable guidance comments and suggestions

We also place on record, our Sincere Thanks and gratitude to one and all who have Directly or Indirectly have lent their helping hand in this venture

**ABSTRACT**

Student Information Management System can be used by education institutes to maintain the records of students easily. Achieving this objective is difficult using a manual system as the information is scattered, can be redundant and collecting relevant information may be very time consuming. All these problems are solved using this project.

The project focus on the Adding, Deleting, Updating and Viewing of profile of Students and storing in database. In the data collected Admission number is used to locate a single profile out of the thousands available in the Database

We primarily use python as Frontend in this project to create user-friendly Gui using Tkinter Module and logic behind different process available throughout the program etc…. and we use MySQL as Back end to store various Student details in one Table and another table to store all the Account that can access this program. We also include PyMySQL Module used to establish a connection between Python Gui(Front-end) and MySQL Database(Back-end)

**REQUIREMENT ANALYSIS**

**1. Hardware Requirement**

* + x86 or 64-bit CPU (Intel / AMD architecture)
  + 4 GB RAM
  + 5 GB free disk space

**2.** **Software Requirement**

* Python (Version Above 2.9)
* My SQL
* Any version of windows XP or higher

**APPLICATION**

The student management system can be easily expanded to fit a variety of educational institutions like:

* Schools
* Universities and College
* Tuition centers

This program can be also be converted to work for other fields like:

* Hospitals
* Gyms
* MNC etc…

**DESIGN**

**1.BLOCK DIAGRAM**

**2. EXPLANATION**

The objective of this interface is to manage student profile for different students who are enrolled in the institution. The institution’s administrator shall be able to save all the relevant details pertaining the student’s information on the accounts he holds in the application. These details shall be saved in the database in encrypted format. This will help the administrator to locate and manage different student profile for each student in the local database. The user shall be able to add records edit and delete record using the system. The user has to login to the system in order to use this tool. It takes user login and password.

**FLOWCHART**

# 2. Program Flow



**ALGORITHM**

Sign-in window

1. Input Username, Password to add to database
2. Read Username, Password
3. If (Username does not exist) add Username, hashed Password
4. Else show message “User already exists”

Login window

1. Input Username, Password
2. Read Username, Password
3. If (Username in database and Password is correct) Student information management system
4. Else show message “Wrong credentials”

Main Window

1. Click Add to add new Student details

1.1 Input First Name, Last Name, Course, Subject,

Year, Age, Gender, Birthday, Contact, Email

1.2 Add to database

2. Click to View Details to View existing Student details

2.1 Input student’s contact number to search

2.2 Search using contact number in database

2.3 Retrieve details and display them

3. Click Update to update Student’s details

3.1 Input student’s contact number to search

3.2 Search using contact number in database

3.3 retrieve details and display them

3.4 update necessary details

3.4.1 Click Submit to update changes in database

3.4.2 Else click Cancel to abort making changes

4. Click Delete to delete Student’s details

4.1 Input student’s contact number to search

4.2 Search using contact number in database

4.3 click submit to delete record from database

5. Click Clear to Clear the window

6. Click Exit to quit the program

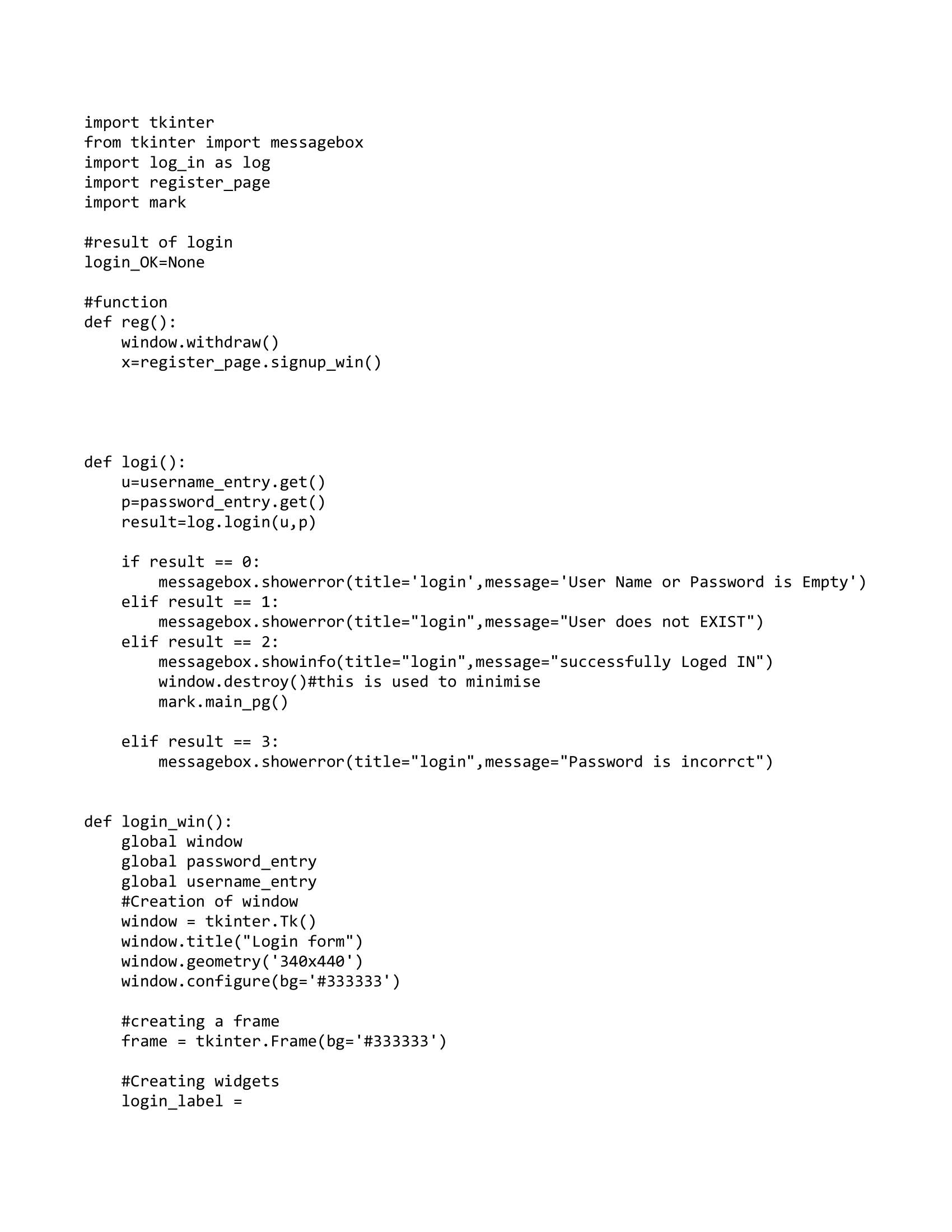
**SOURCE CODE:**

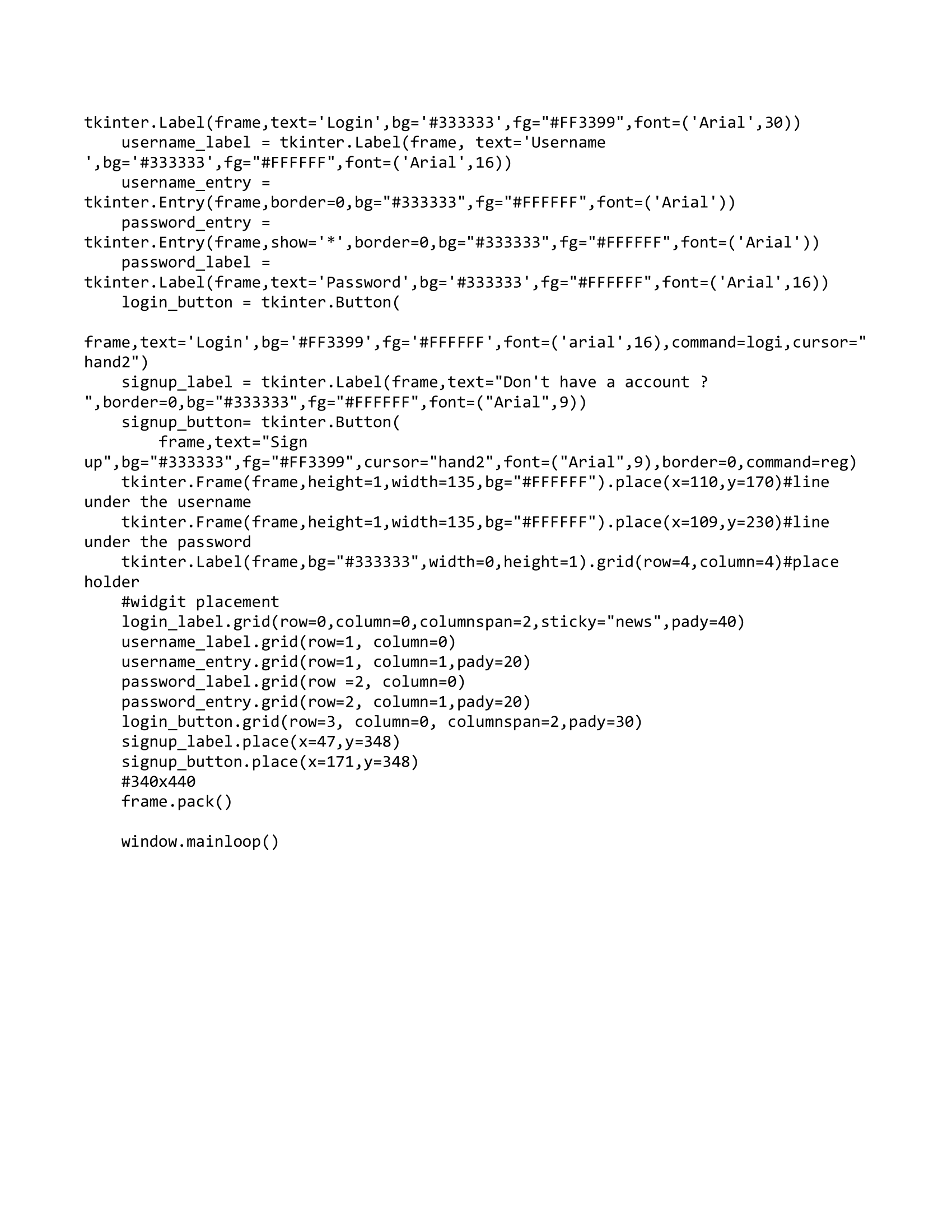
**Main.py**

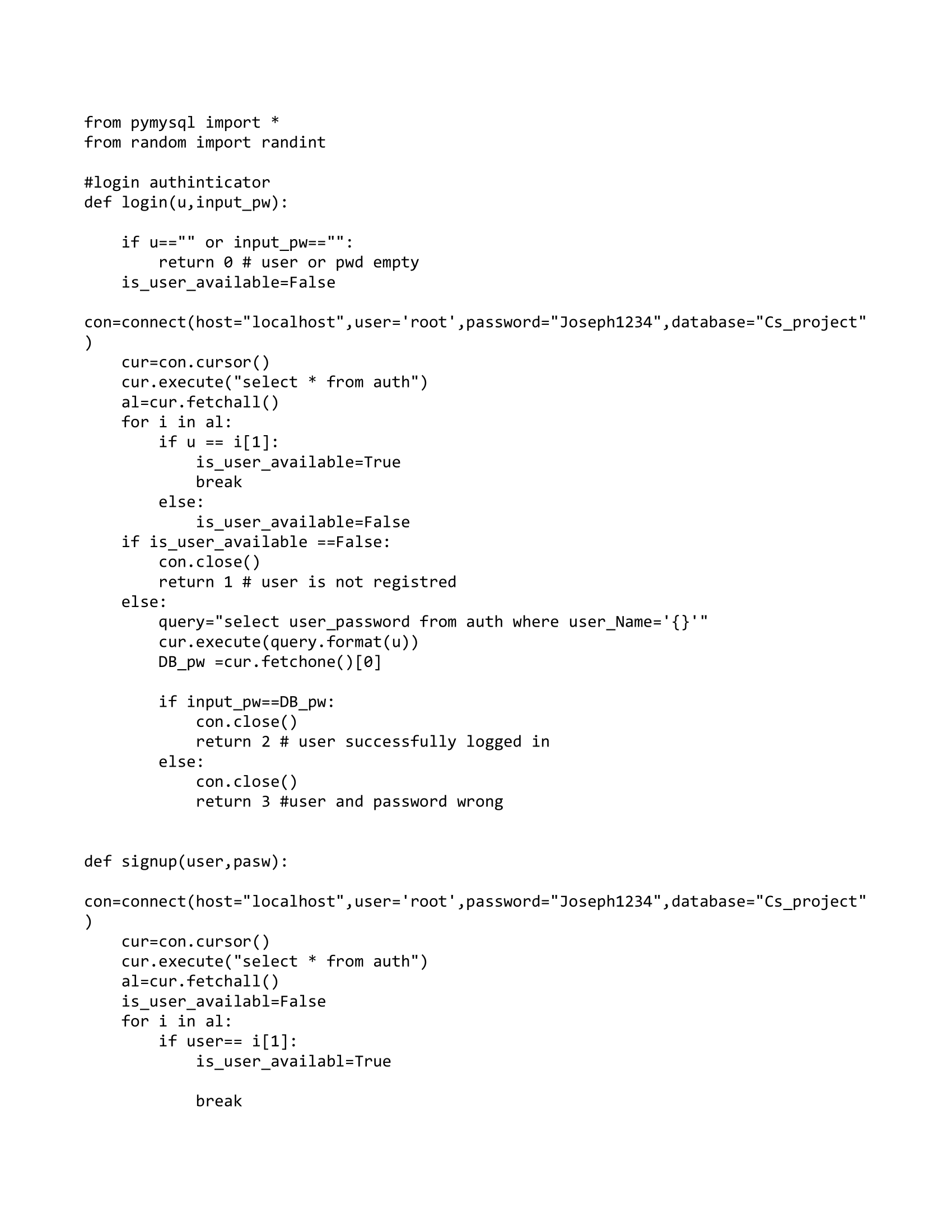
import login\_page

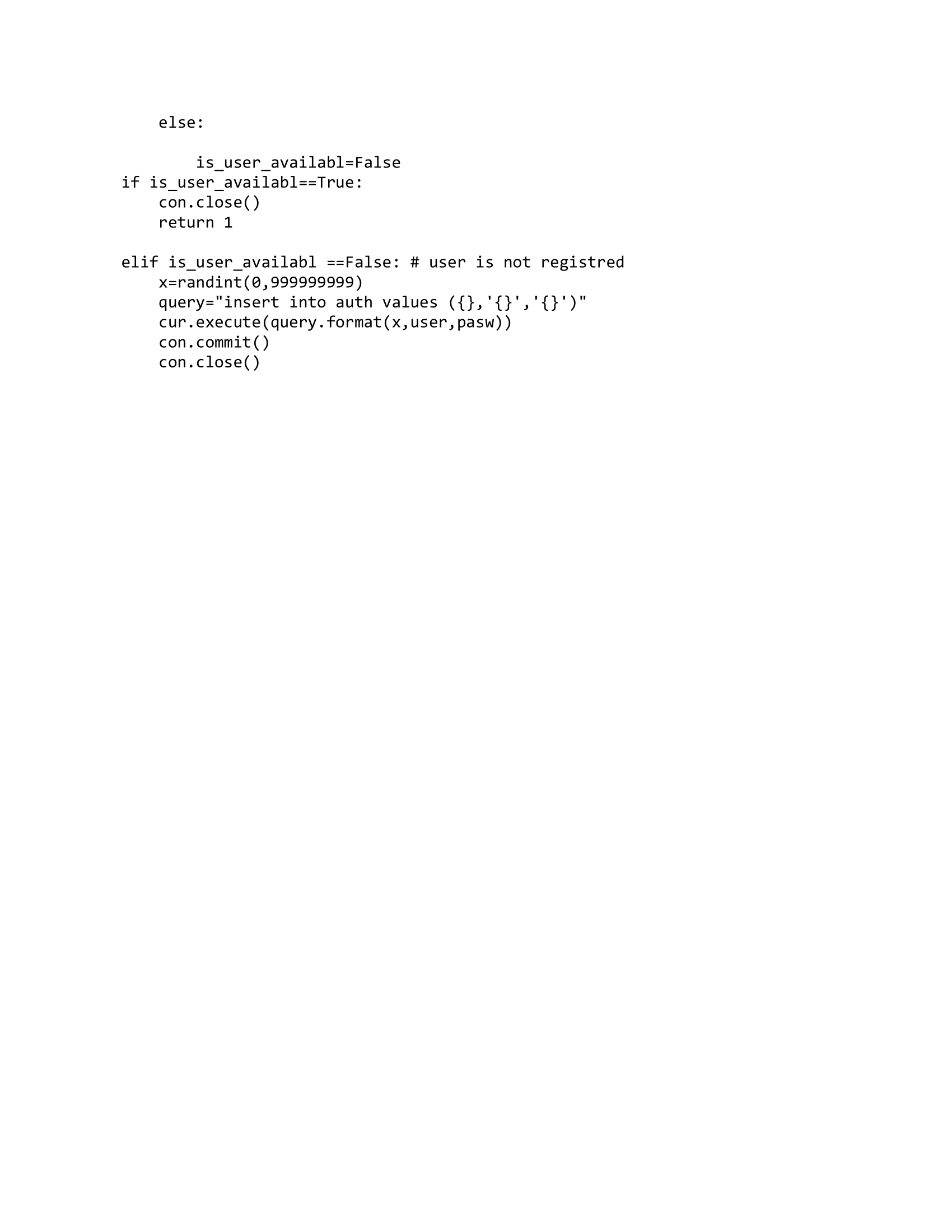
login\_page.login\_win()

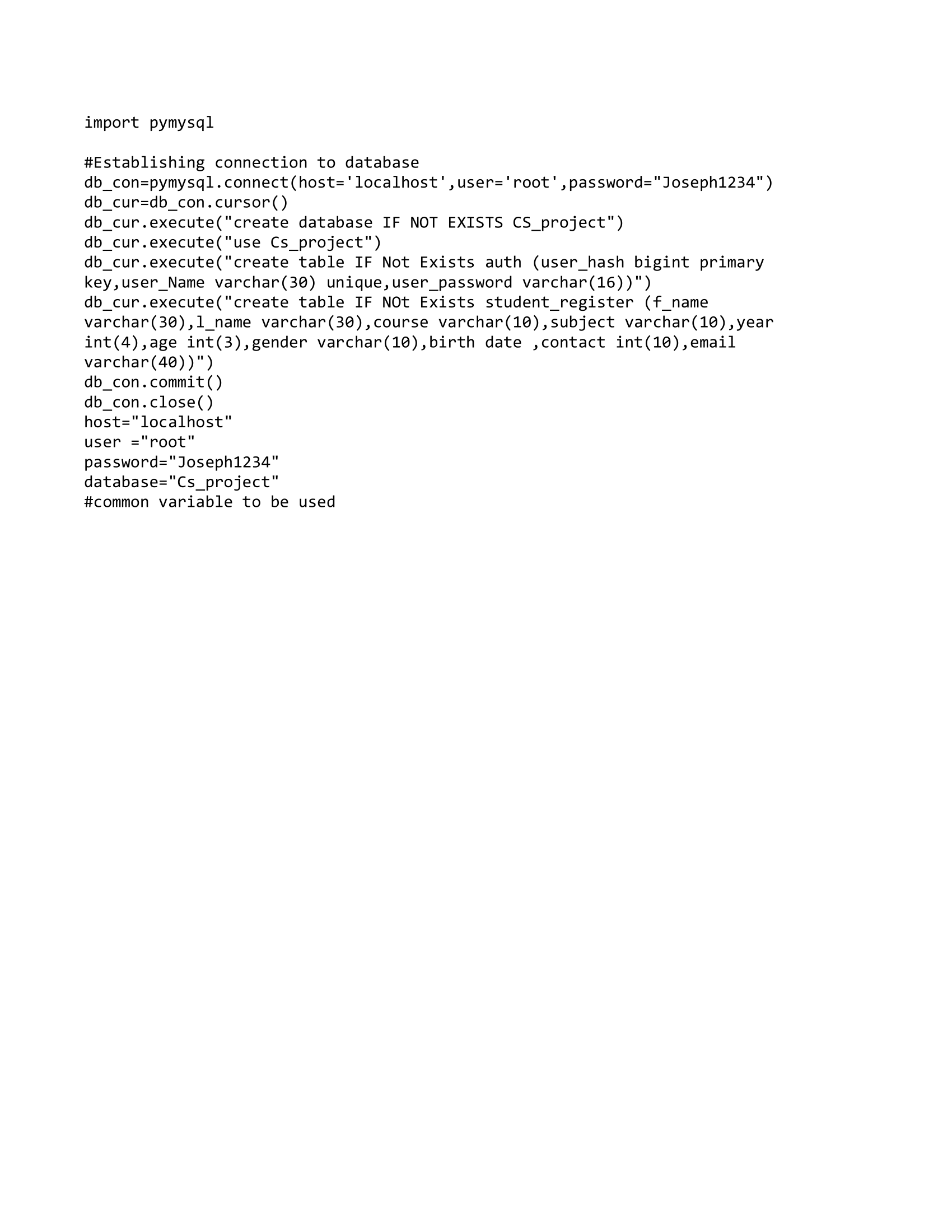
**login\_page.py**

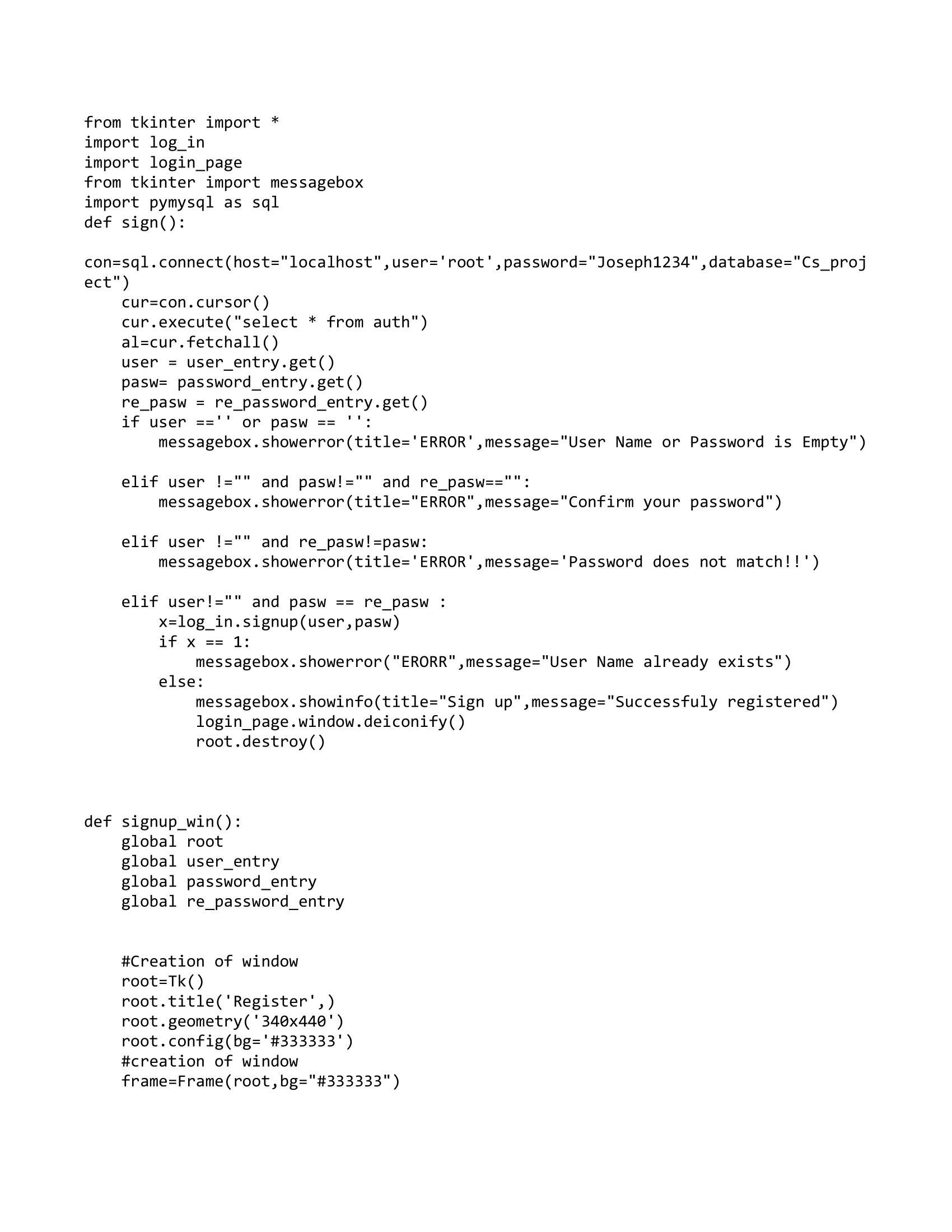
****

****

**Login\_in.py**

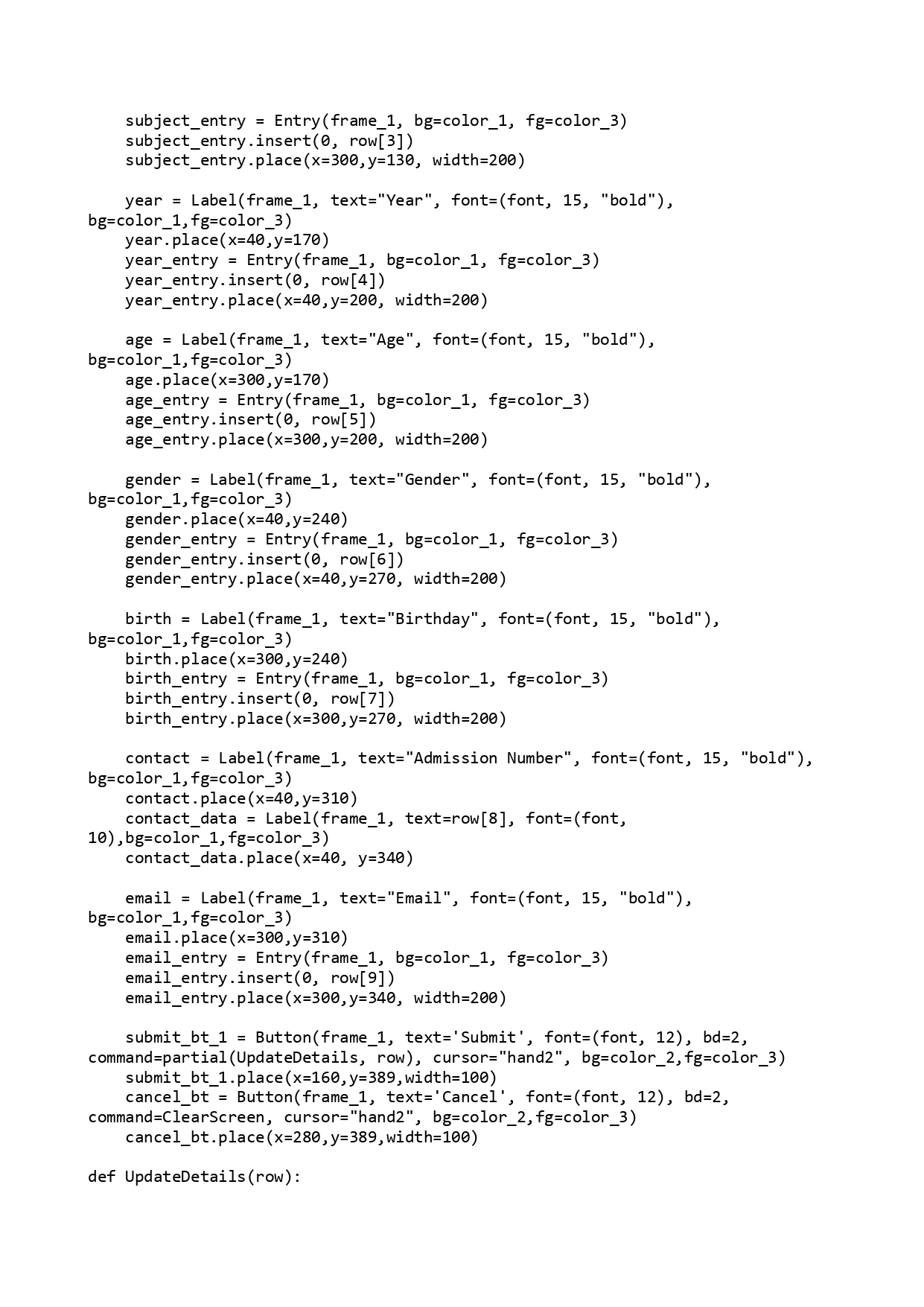
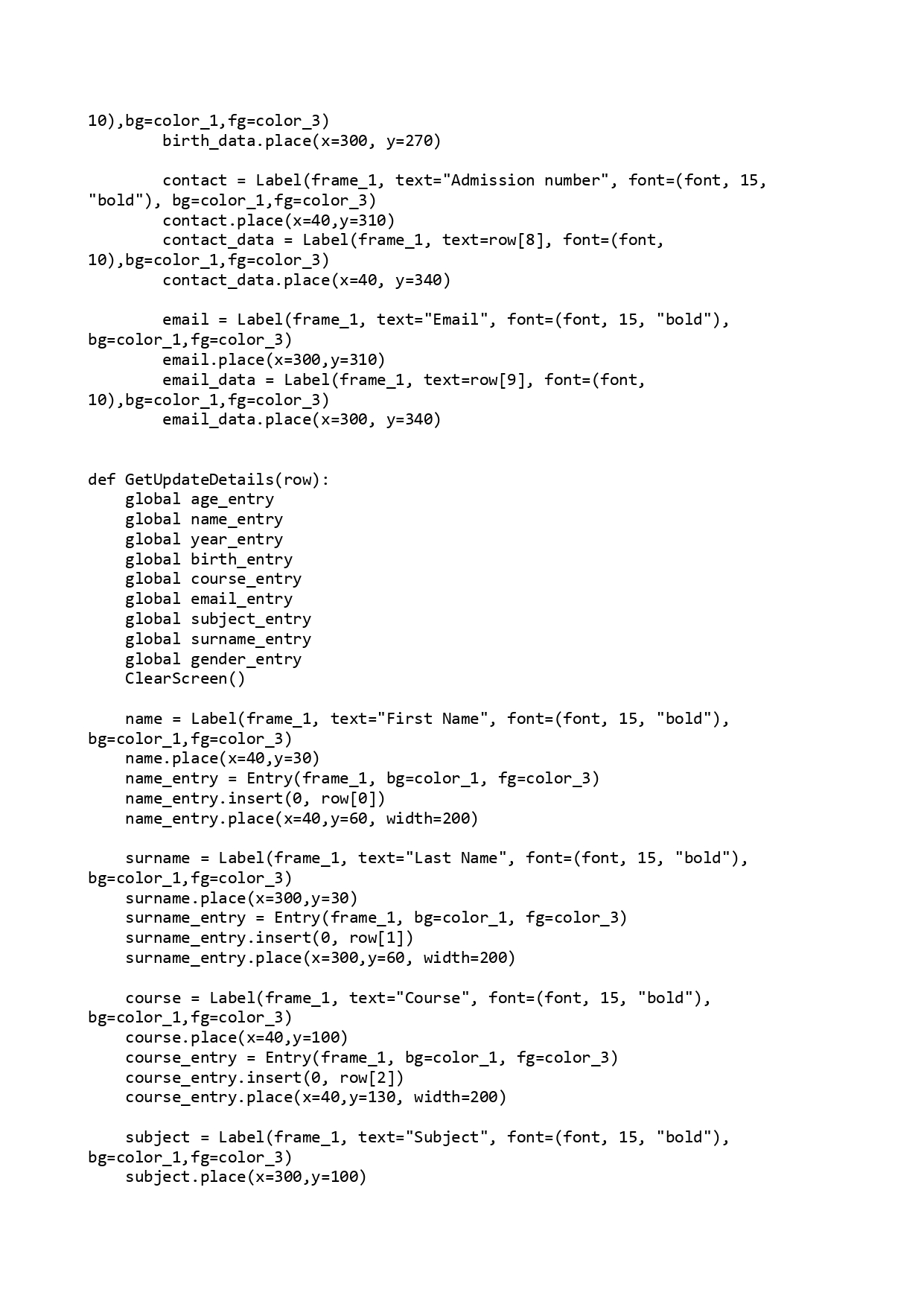
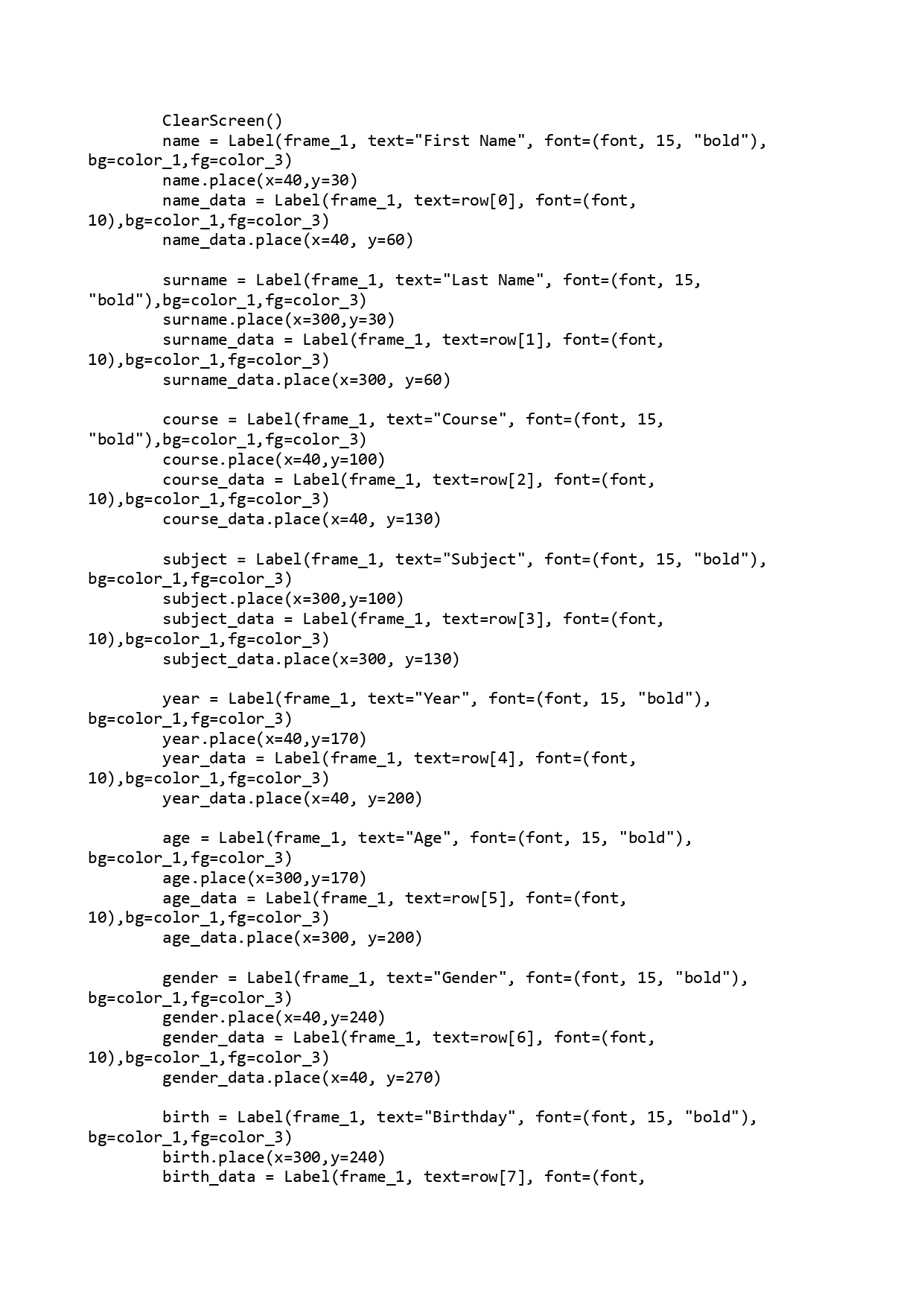
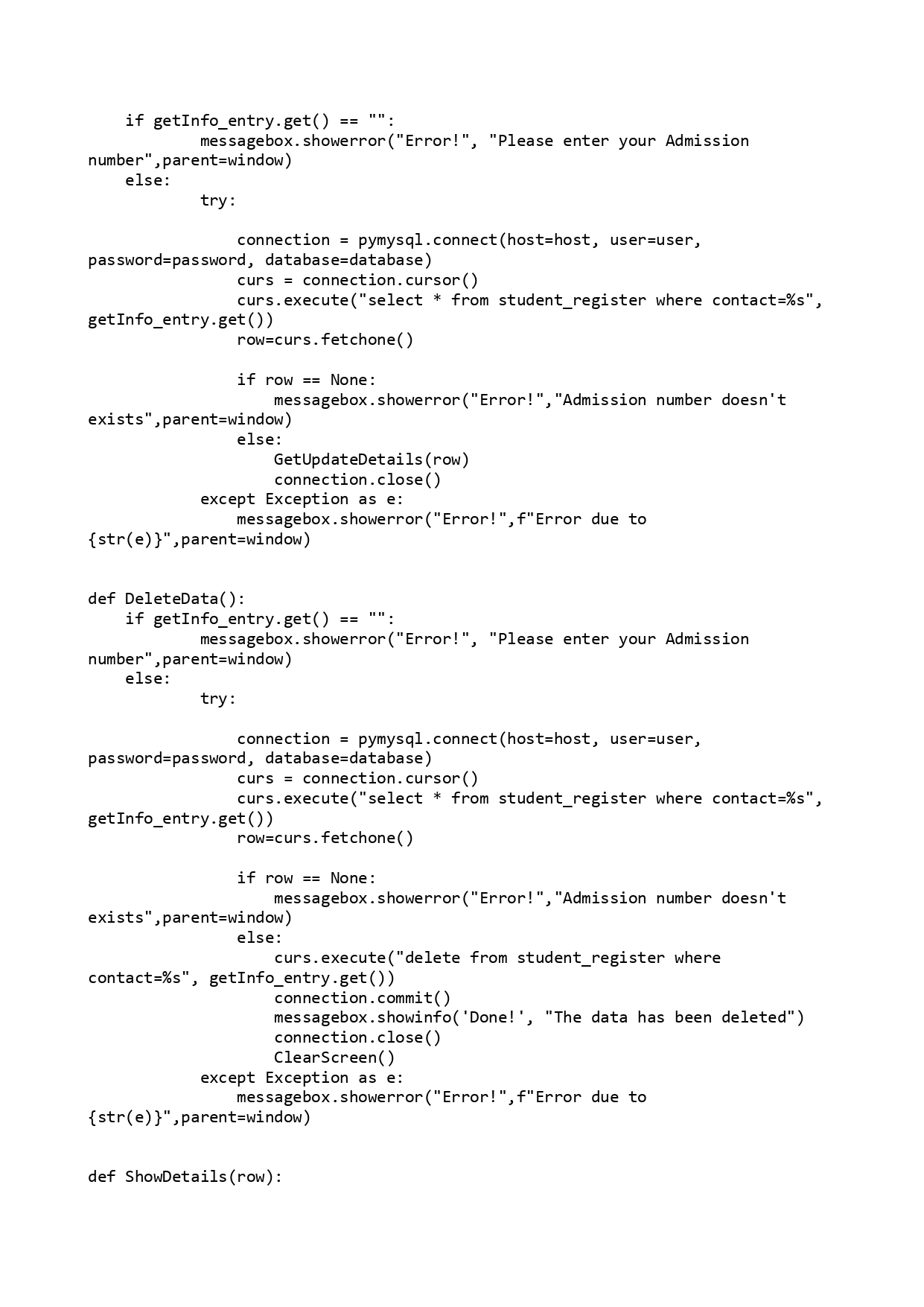
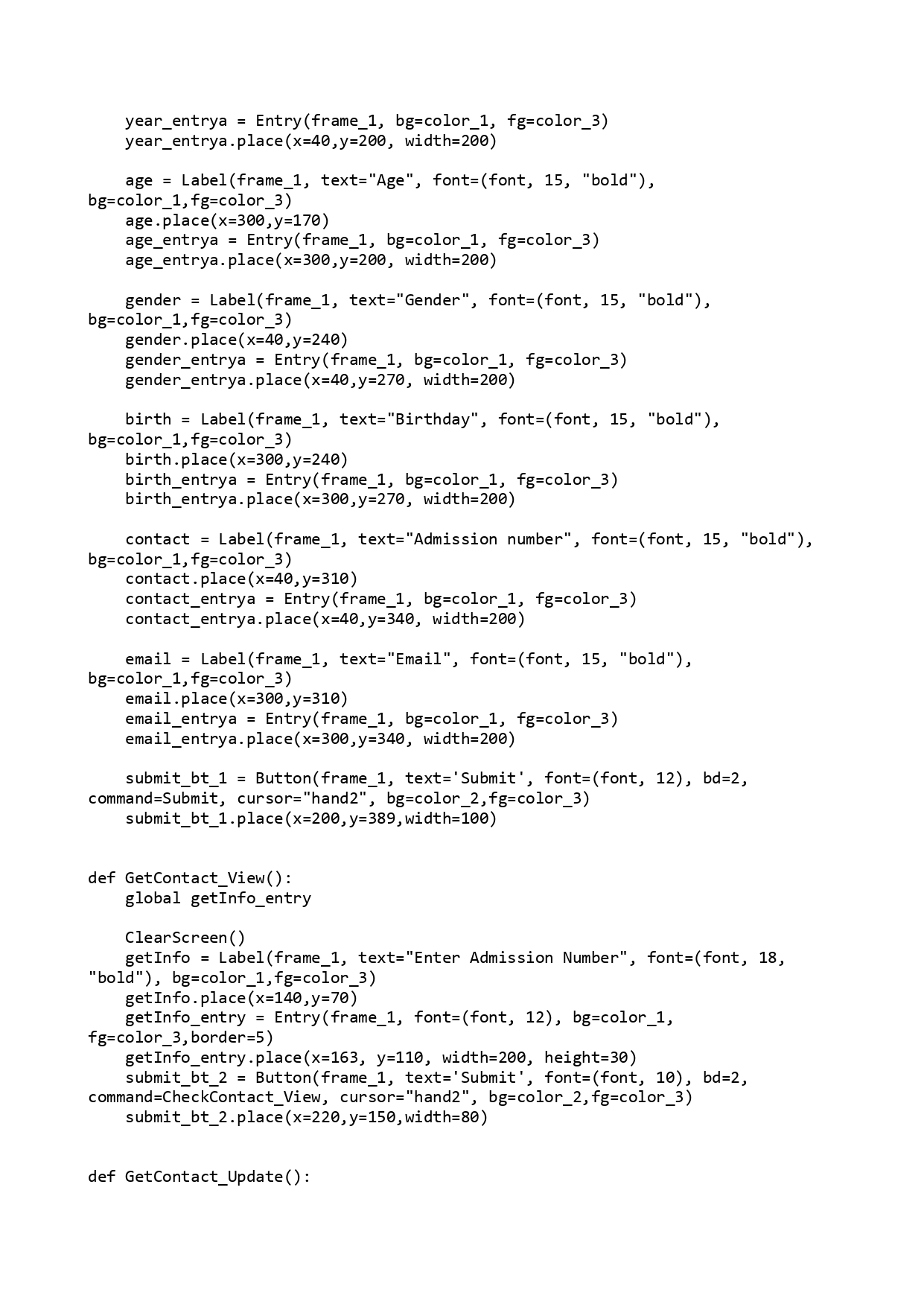
****

**database\_connection.py**

**register\_page.py**

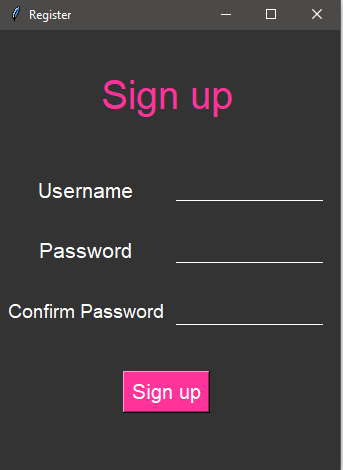
****

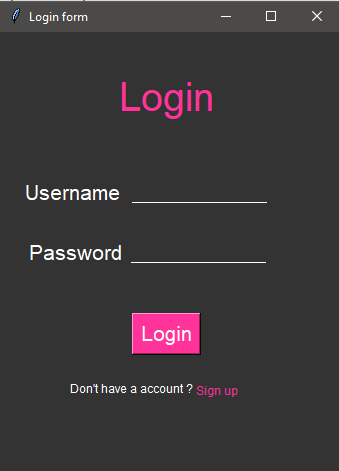
**mark.py**



**SCREEN SHOTS:**

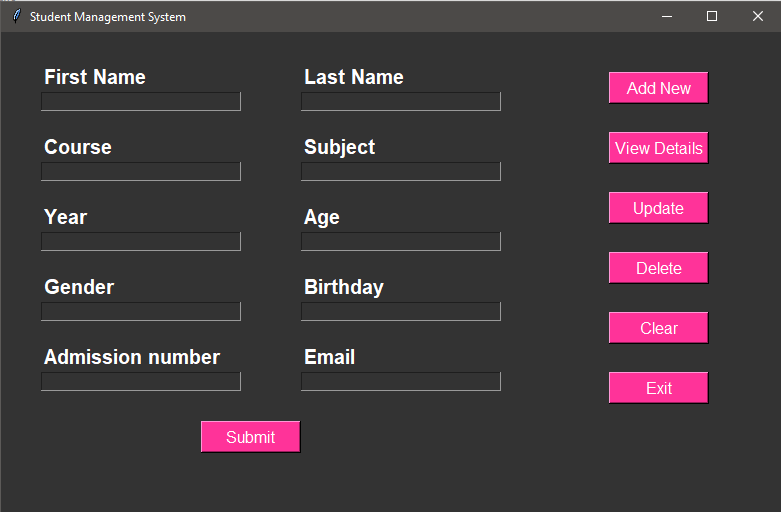
Sign up:

****

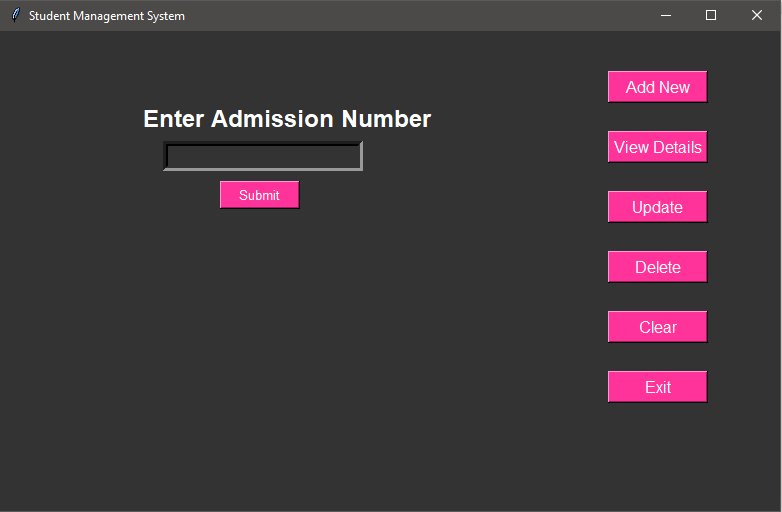
****Login page:

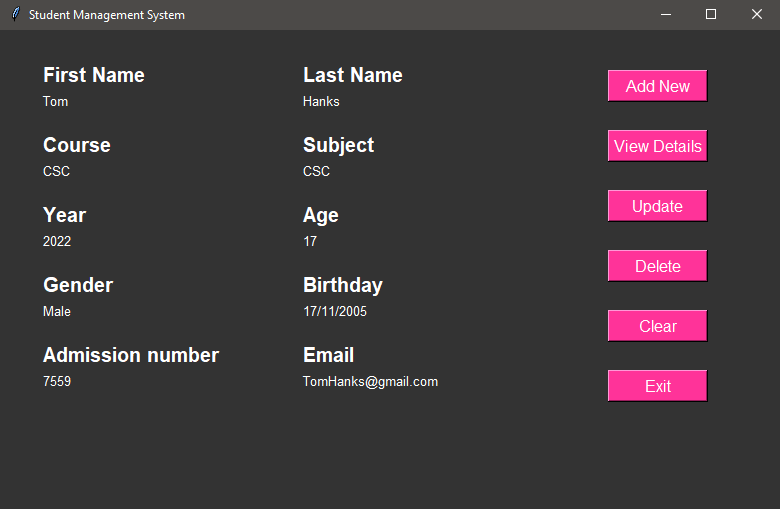
Management page:

Add page:

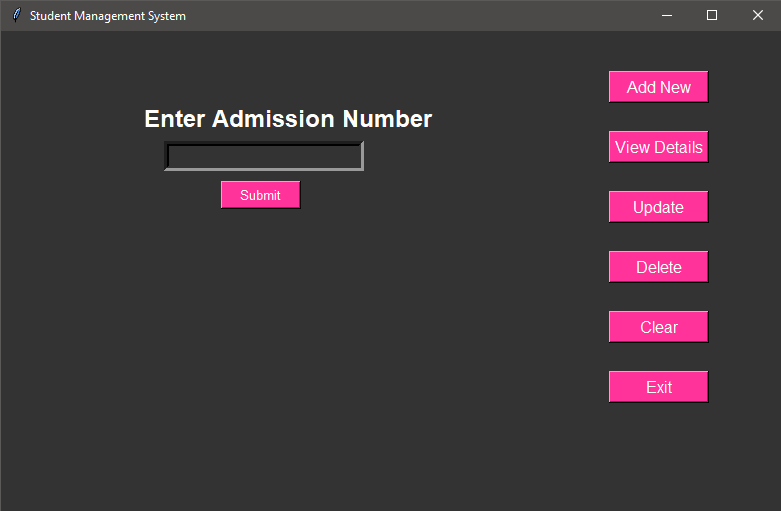
****

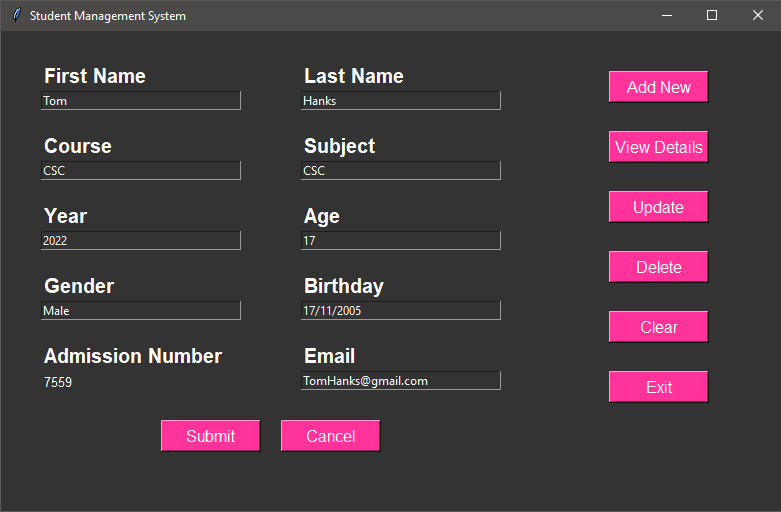
View Details:



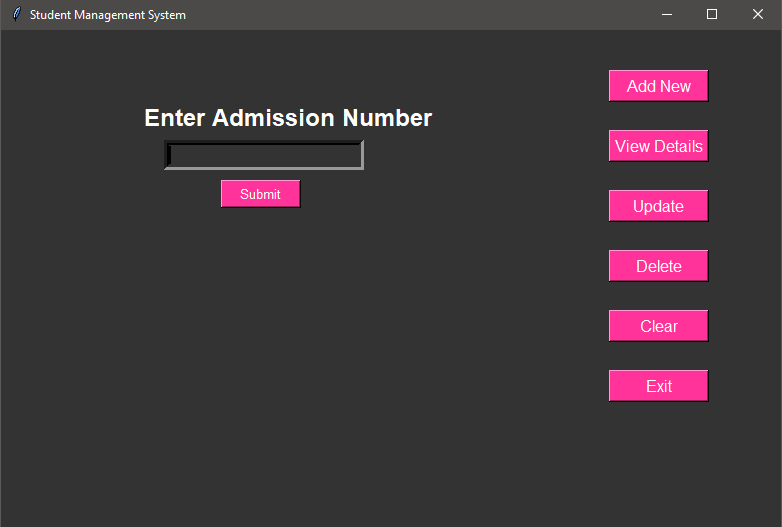
****

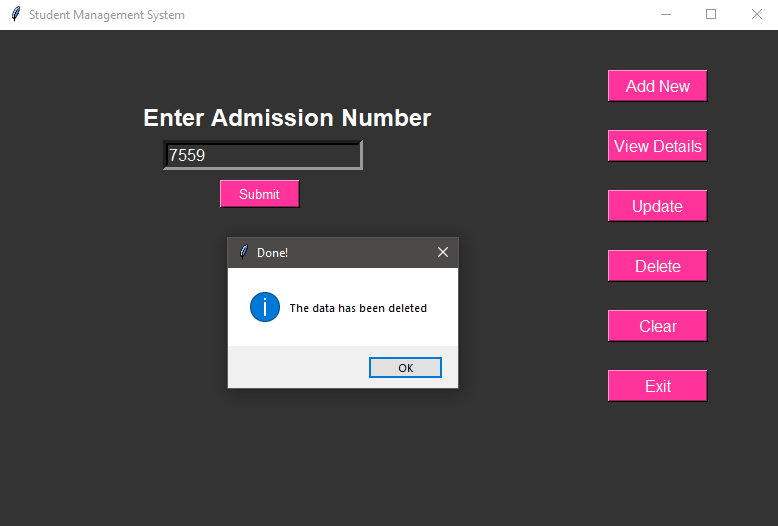
Update details:





Delete:



****

Clear:



**FUTURE ENHANCEMENT**

* Using python classes to make the code more efficient.
* Dynamic and More user-friendly GUI.
* Save password by encrypting it
* Add a student login to view information
* The Student Information Management System (SIMS) can be enhanced to include some other functionality like marks, attendance management.
* Talent management of students based on their performance evaluation can be added.
* Social networking can also be added wherein students can interact with each other
* Online class functionality can be added
* Online exam functionality can be added

**Bibliography:**

* www.w3schools.com
* Informatics practices by Sumita Arora
* https://realpython.com/python-gui-tkinter
* https://docs.python.org/3/library/tkinter.messagebox.html
* https://docs.python.org/3/library/tkinter.html
* https://pypi.org/project/PyMySQL/#documentation
* https://docs.python.org/3/installing/index.html
* https://www.color-hex.com/
* https://docs.python.org/3/library/functions.html
* https://docs.python.org/3/library/functools.html
* https://docs.python.org/3/library/functools.html
* https://docs.python.org/3/tutorial/modules.html
* http://www.google.com/