**EUROPEAN UNIVERSITY OF LEFKE**

Faculty of Engineering Department of Computer Engineering



COMP 337

DATABASE MANAGEMENT SYSTEMS

Hospital Management System

|  |  |
| --- | --- |
| **Name**  **SAMET BARIŞ ÖNCÜL** | **Std Id**  **20121758** |
| **Name**  **MUSTAFA ÖZER** | **Std Id**  **196118** |

**Submitted to:**

Salman Khan

**Grading Sheet**

|  |  |  |  |
| --- | --- | --- | --- |
| **Grading Criteria** | **Marks** | **Team Member 1** | **Team Member 2** |
| 1. Programming Language and Connection | 03 |  |  |
| 2. Functionality | 05 |  |  |
| 3. ER Diagrams | 05 |  |  |
| 4. Database Working | 07 |  |  |
| 5. GUI Interface | 03 |  |  |
| 6. Report and Format | 07 |  |  |
| 7. Viva | 10 |  |  |
| Total | 40 |  |  |

# Teacher Signature:

**Date:**

# Term Project Report: GUI-Based Desktop Application

Create a Table of Contents, a List of Figures and List of abbreviations (if any)

# Project Overview

* 1. **Problem of Interest:**

We choose hospital management system to help the management of patients more easily. Interface is easy to use we kept it simple and readable for everyone. System basically keeps track of incoming patients like how long they are going to stay, which medications they are going to use, how often they will use and what will happen after they are discharged.

# Team Details

Team Members

* + - SAMET BARIŞ ÖNCÜL – decide on project, connected database and programming language together
    - MUSTAFA ÖZER – decided and designed the GUI, made sure connection is safe between backend and frontend.

# Job Descriptions

* + - MUSTAFA ÖZER - Led the development of the front end and data handling logic.
    - SAMET BARIŞ ÖNCÜL - Focused on database design and backend development.
    - SAMET BARIŞ ÖNCÜL - Assisted in frontend development and testing..

# Programming Language Choice

* We decided on Python. It was very easy to use and understand. User friendly, large library support and community support. We found plenty of documents to help us understand the topic and basic syntax. We used modules like tkinter, messagebox, mysql.connector, random, time and datetime.

.

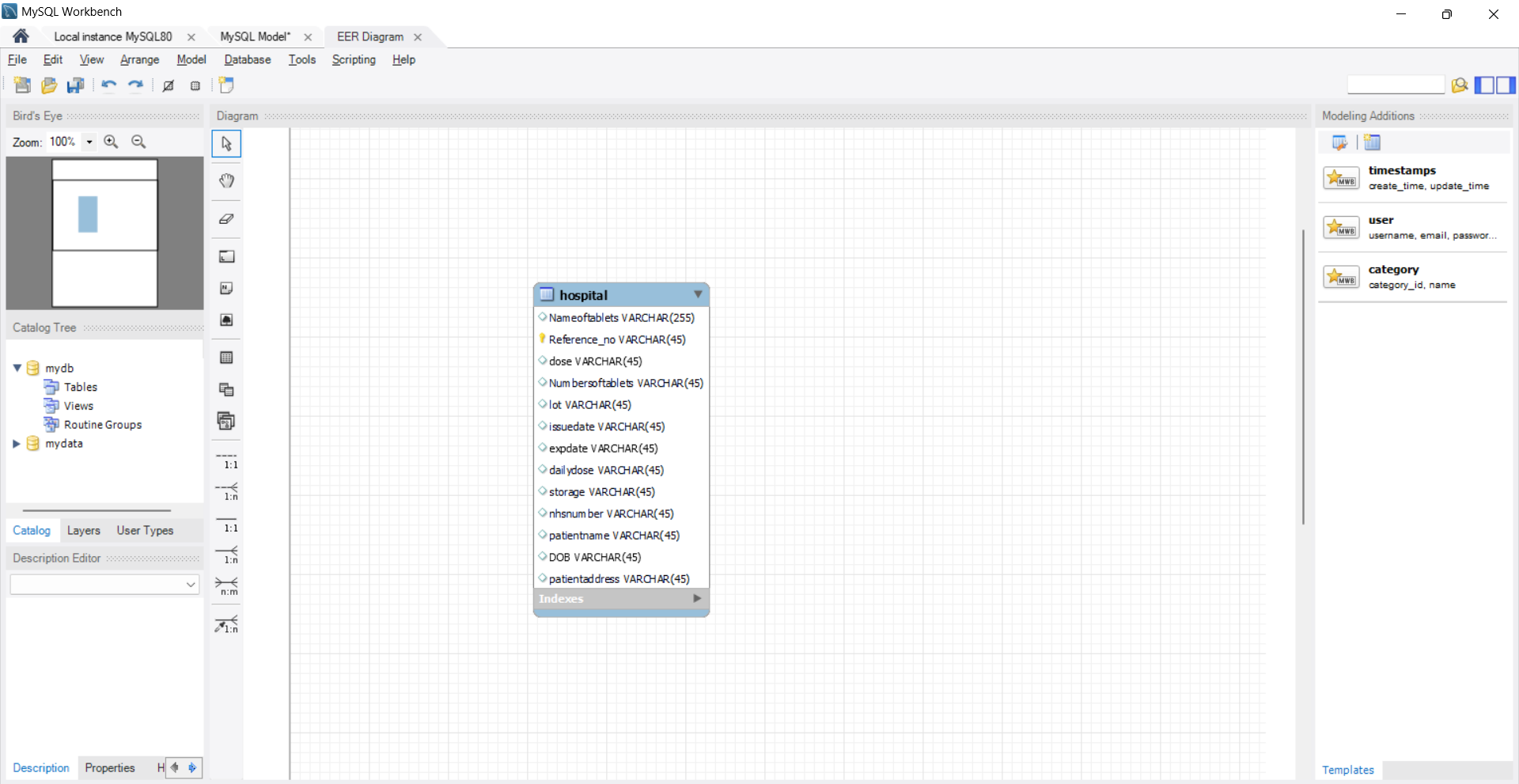
# Database and Relations

* 1. **Database Choice**
     + We choose MySQL because we have been studying MySQL for a semester so we were familiar with syntax and query. Connection to python was successful due to python’s large choice of modules.

# Database Relations

In our application, we found it appropriate to use one database table instead of four tables. If more is desired, we can solve this problem with future updates. On the other hand, there are 13 data entries in our table named 'hospital'. We recorded the necessary special information about the patient and the necessary procedures in this table.

*ER Diagram is here;*



# Functionality

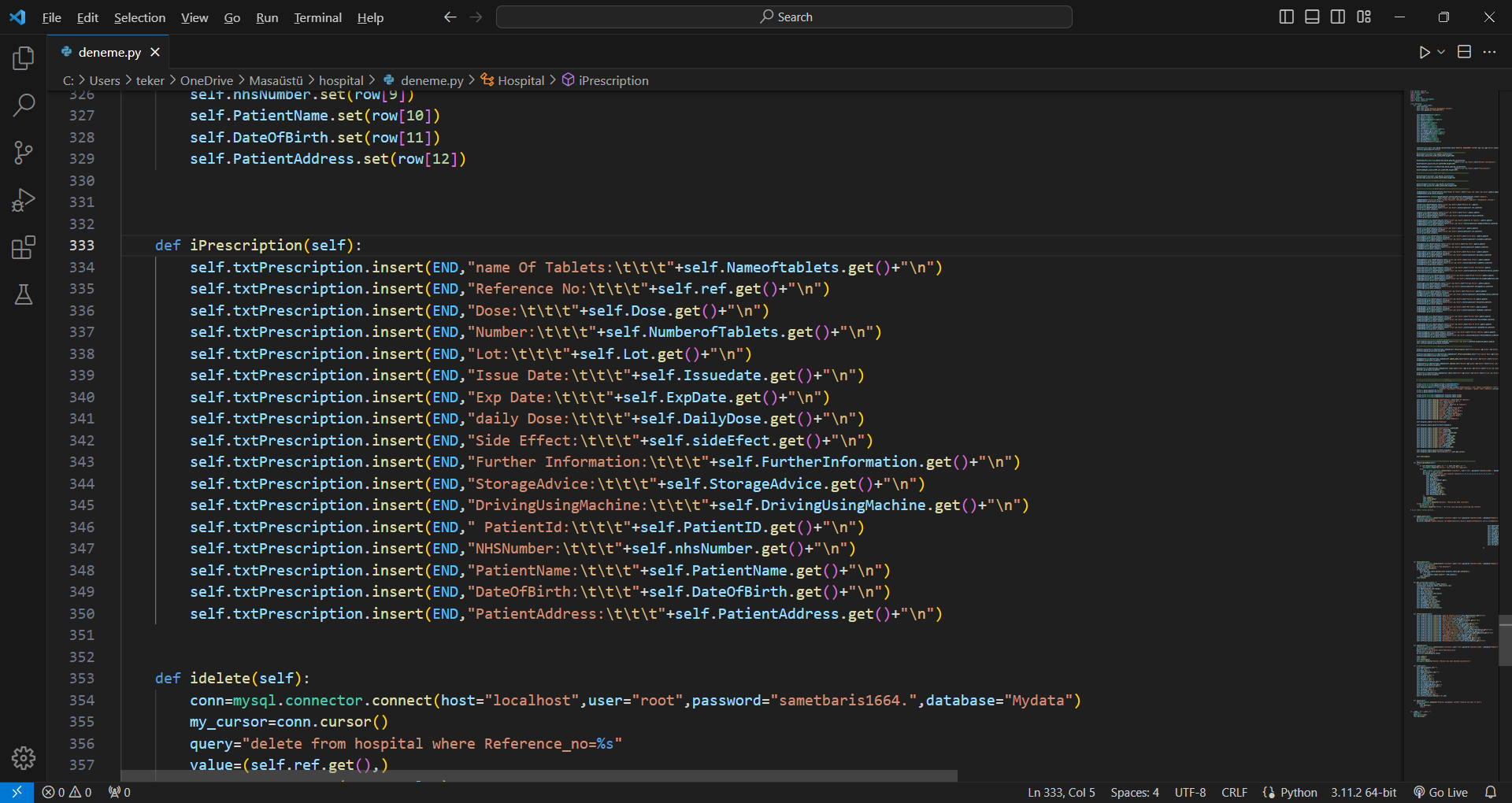
There are 6 functions in total in our application.

These are Prescription, Prescription Data, Update, Delete, Clear and Exit functions.

Now let's introduce these functions.

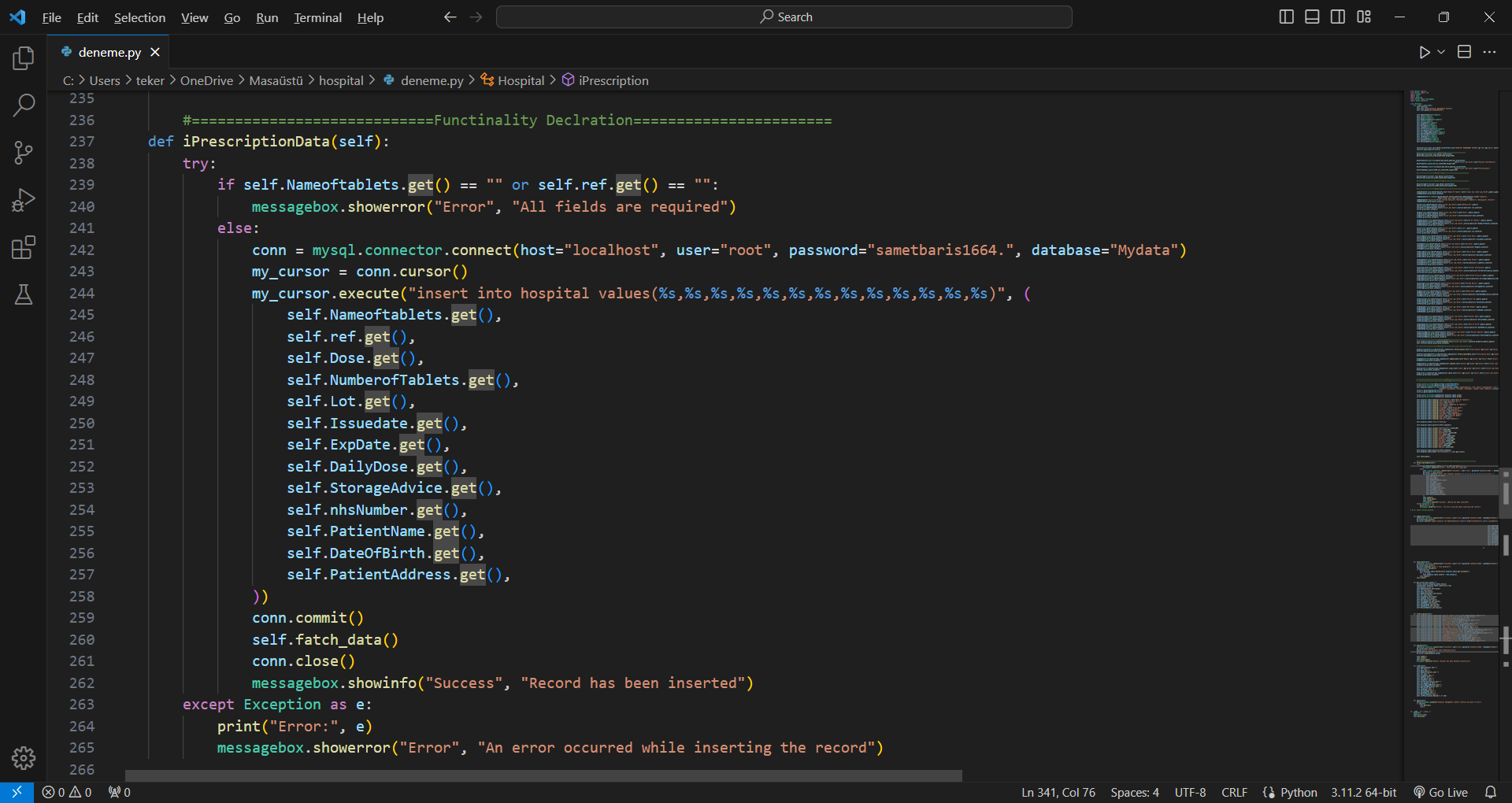
First, the Presciption function. Thanks to this function, after entering the information of the incoming patients, when we press it, this information is listed on the right side of the application.

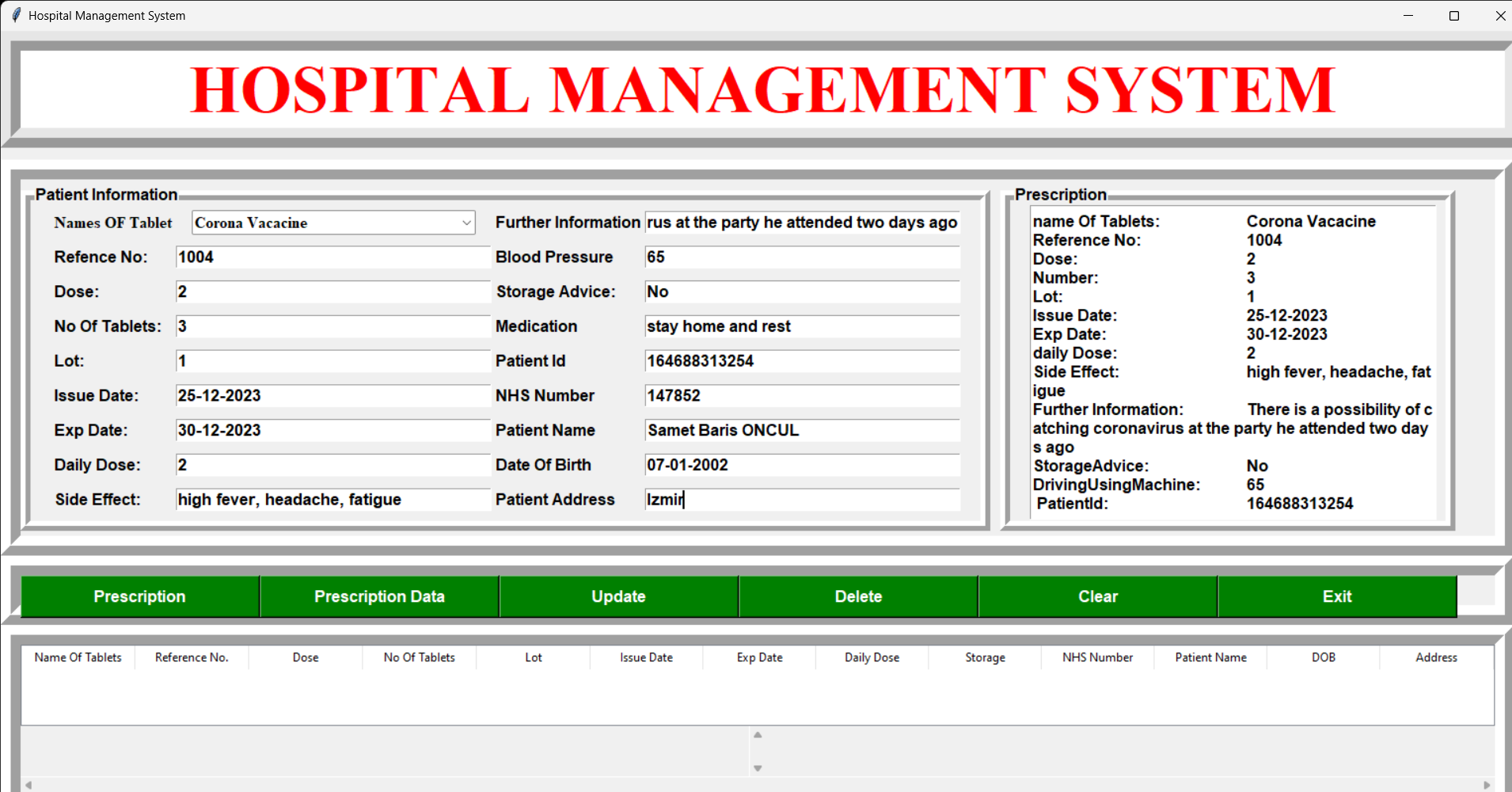
I share the screenshot and codes of the function.

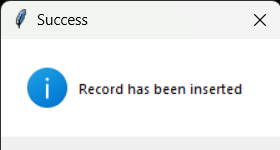


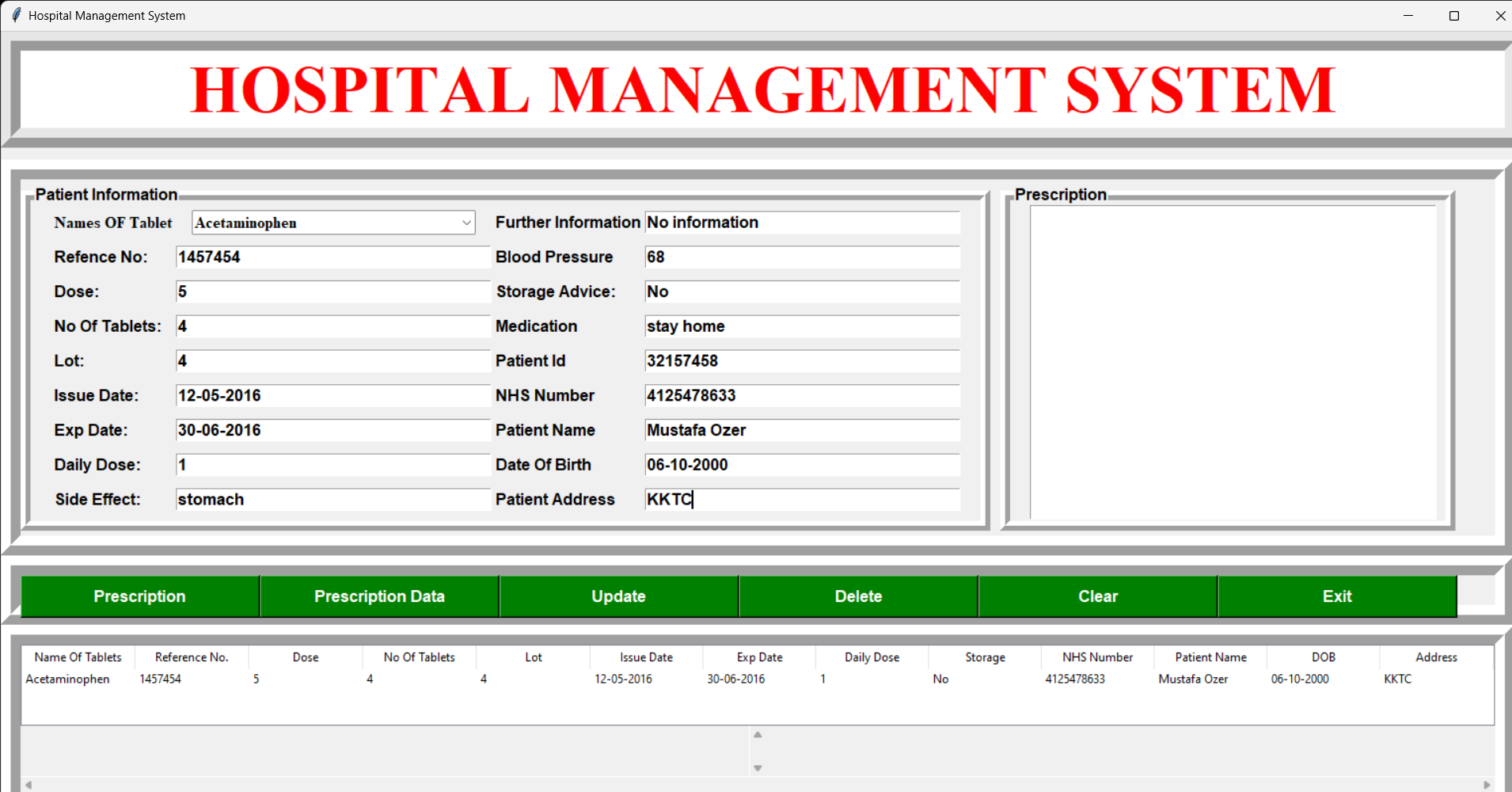
Our second function is prescription data, our method of adding data. Thanks to this function, we successfully save the information received from the patient to our database.

I share the screenshot and codes of the function

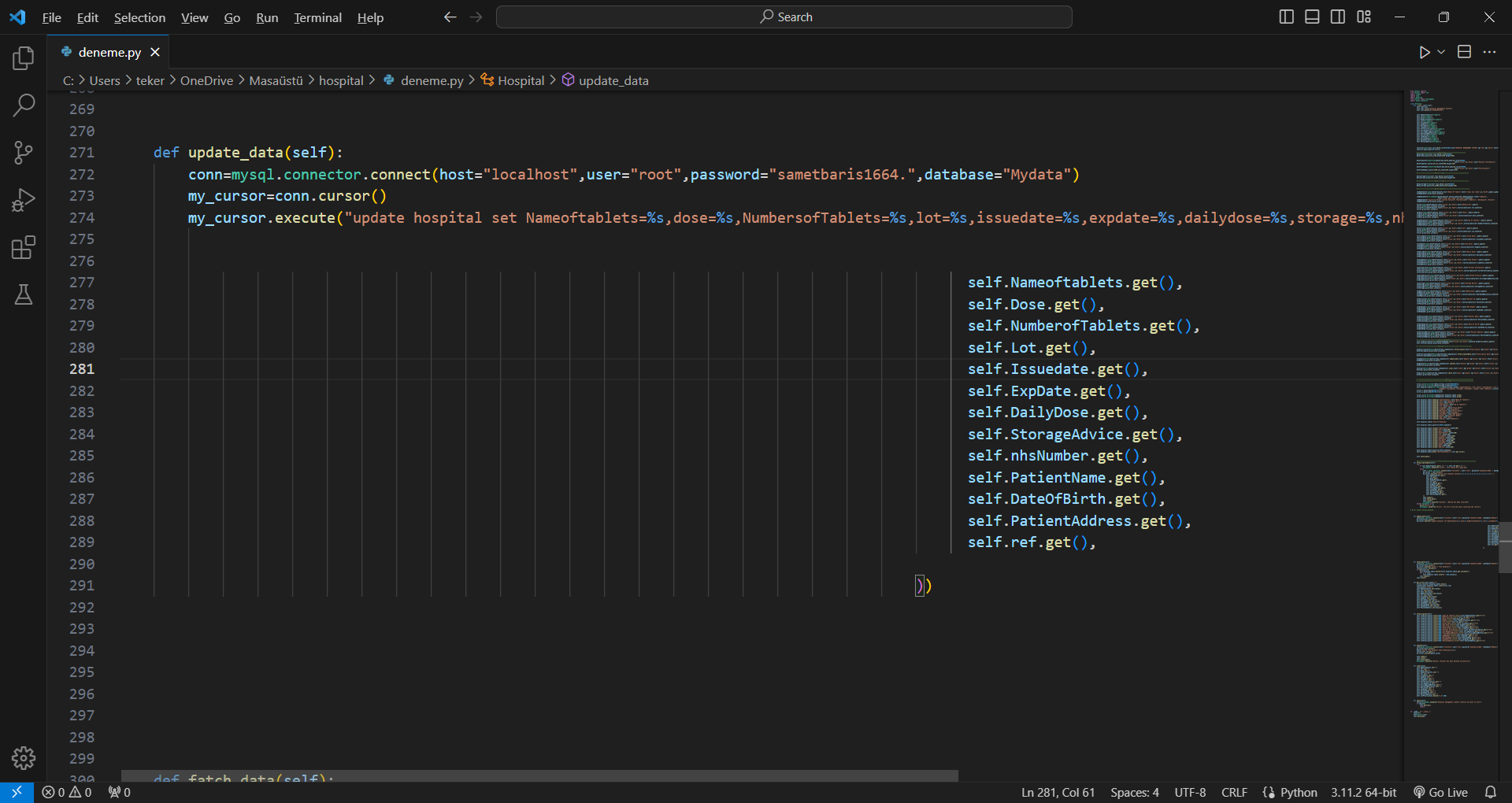




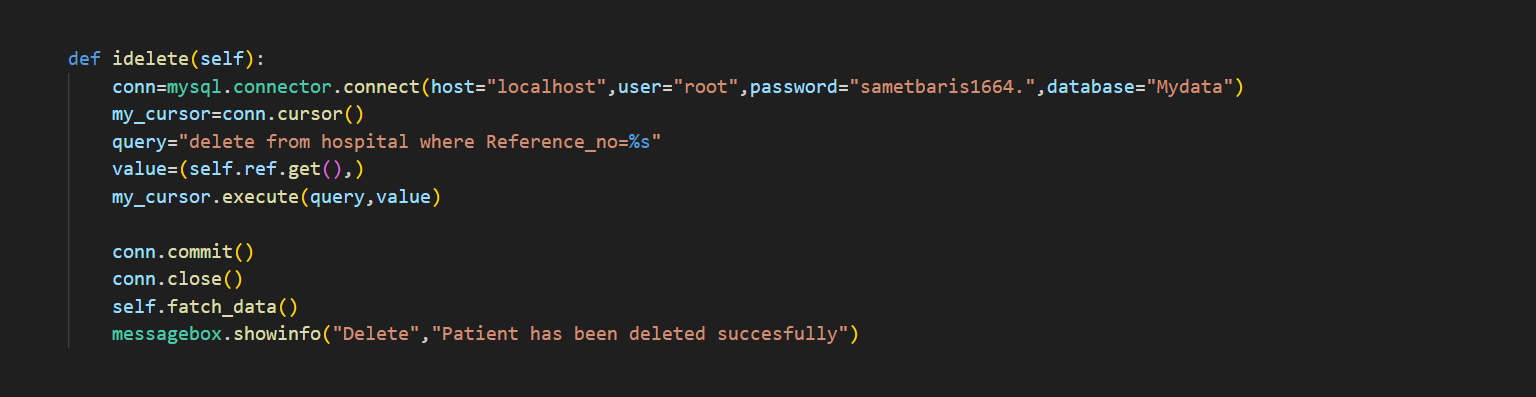


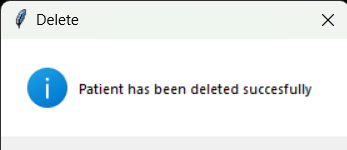


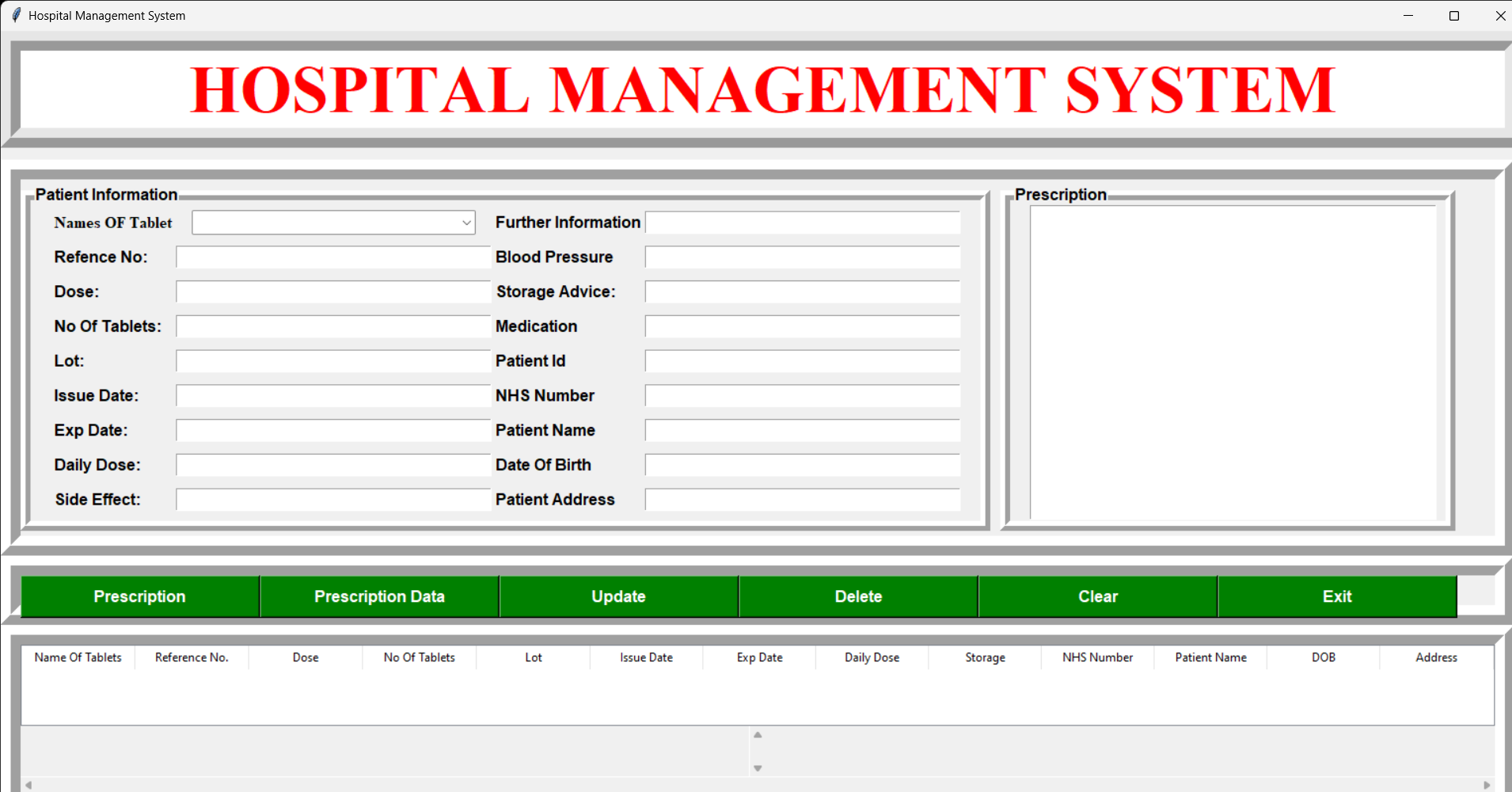
Our third function is the Update function. We wrote the front-end codes for the user part of this function, but there was an error in the codes when updating from the database, so it does not work properly. We will fix this in future updates :)



Our fourth function is the Delete command. Thanks to this command, we can select and delete patients in the list. I share the source codes and the screenshot from the application.



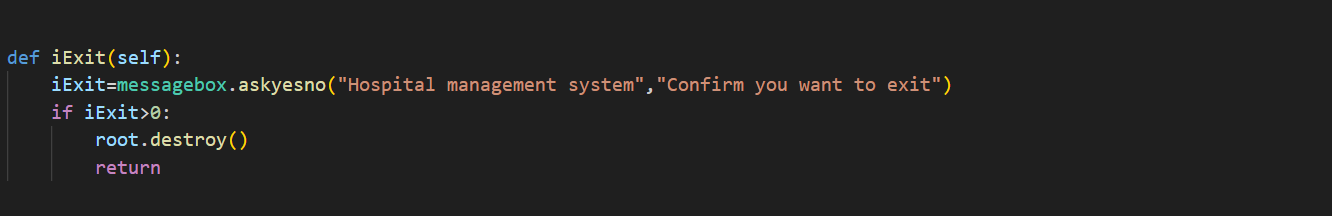


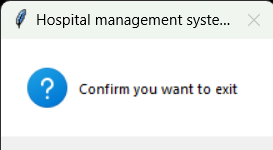


Our fifth function is to delete the data from the beginning and use it again if there is a mistake while entering the data of the patients. As a sort of 'Correction'. The codes is below.



Our last function helps us exit the application.





# Screenshots and Demonstration

Since I shared it in detail in chapter 4, I did not share the same things again in chapter 5.

# Overall Assessment

In our project, we learned where we came by using the SQL codes we learned throughout this year. My friend and I saw what we could do with a MySQL database application using the Python programming language.

Among the difficulties we encountered was how we could make the front-end part of our application and where we could find the codes for it. We solved these easily by doing a little research. There will be a discussion about how we can make a better system by responding to our users' opinions about future improvements. In addition, there may be improvements in the interface for nurses, doctors and similar healthcare professionals who require a password when entering the application.

# Conclusion

In general, when we look at the application from start to finish, we have created a hospital management system. In this system, when the patient comes to the hospital, we created a system to register himself in the hospital system and get his information. As I mentioned before, while writing the codes, the difficulty was that we did not know some of the places, so we found out what we could do while writing these codes. In short, this was our application. Stay tuned for updates coming soon. Best regards..