INTRODUCTION

India is a vast country with a large population. From personal experience, we have seen many problems such as accidents caused due to mismanagement of space.The parking system is inefficient, and there are many ways to improve it. Our Computer Science project is a Vehicle Parking management system - for managing the records of the vehicles entering and exiting in a parking area. The Admin can easily retrieve data if a vehicle has entered the parking space.

Nowadays, in public places - malls, multiplex systems, hospitals, offices, market areas - we face the crucial problem of parking. The vehicle parking space has many lanes/slots for car parking. But, to park one's vehicle, one has to look through them all.

Moreover, this involves a lot of manual labour and investment. Instead of paying the price of being towed, we can park with safety and security at a low cost.The parking control system has integrated many security devices such as parking control gates, toll gates, time and attendance machines, and car tallying systems. These features are necessary to secure your vehicle. They also help us to evaluate the fee structure for every vehicle in the parking lot.

The objective of this project is to build an efficient Vehicle Parking management system. Hence, there will be better time management and control of vehicles by using number plate recognition. Our system will track the entry and exit of cars, maintain a listing of the vehicles within the parking space, and determine if a parking lot is vacant or not. It will determine the cost per vehicle according to their time consumption.

OBJECTIVE OF THE PROJECT

Our project combats the problem of inefficient parking using Python and MySQL Software. We wish to provide a safer environment and better parking. We can park our vehicle in our own slot by paying.Because of that there are no towing problems.There is no risk for the vehicle owner for parking the car.In case of any damages and problem of vehicle that will claim by parking management.As the world is facing many threads daily, robberies are done easily with no track to trace, bomb blasts occur with the use of vehicle, so if a proper system is adopted each and every record can be saved and anyone can be track easily therefore mainly is to make a better and fast software, most important user-friendly. Maintain records in a short period of time and also determine if the parking area is full or not.It will enhance the visitor's experience.

In the modern age, many people have vehicles. Vehicle is now a basic need. Every place is under the process of urbanization. There are many corporate offices and shopping centers etc. There are many recreational places where people used to go for refreshment. So, all these places need a parking space where people can park their vehicles safely and easily. Every parking area needs a system that records the details of vehicles to give the facility. These systems might be computerized or non-computerized. With the help of a computerized system we can deliver a good service to customers who want to park their vehicle into any organization's premises.Vehicle parking management system is an automatic system which delivers data processing at a very high speed in a systematic manner. Parking is a growing need of the time. Development of this system is very useful in this area of the field. The proposed vehicle parking application can be used by any organization. By using our system they can maintain records very easily. Our system covers every area of parking management.

PROJECT REQUIREMENTS

1. Python Software and its modules-Python is an object oriented programming language. It is freely available, and it is open source.You can be downloaded from www.python.org.
2. MySQL Software - MySQL helps us efficiently manage our databases with multiple tables. You can download MySQL software from www.mysql.com.

PROPOSED SYSTEM

We have devised a project to allocate parking spots based on vehicle numbers to improvise the parking system. The system efficiently allots different parking spots and calculates the fee needed after its usage. A permanent record of vehicle numbers is maintained for safety concerns, along with a temporary record to operate the program with ease.

MODULES USES AND ITS DESCRIPTION

- **Tkinter:** Used for creating a GUI interface.The tkinter package (“Tk interface”) is the standard Python interface to the Tk GUI toolkit. Both Tk and tkinter are available on most Unix platforms, as well as on Windows systems.

- **Mysqlconnector:** Used for connecting MySQL and Python.**MySQL Connector** provides connectivity to the **MySQL** server for client-server programs. APIs provide low-level access to **MySQL** resources using either the classic **MySQL** protocol or X Protocol.

- **Random:** Used for generating random numbers in a given range.

- **Math:** Used for mathematical calculations

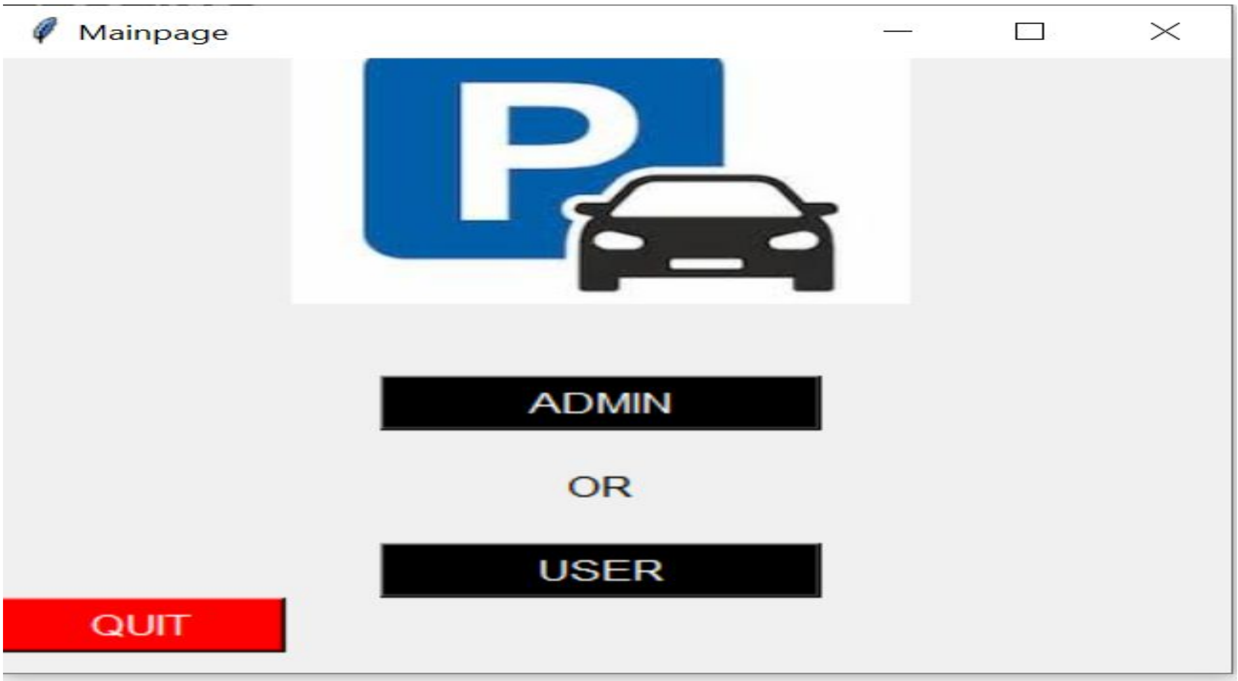
- **Datetime:** Used for generating the current time and date.

- **ImageTk and Image:** python supports images of ppm or pgm. ImageTk is used for converting pics of other formats into this format.

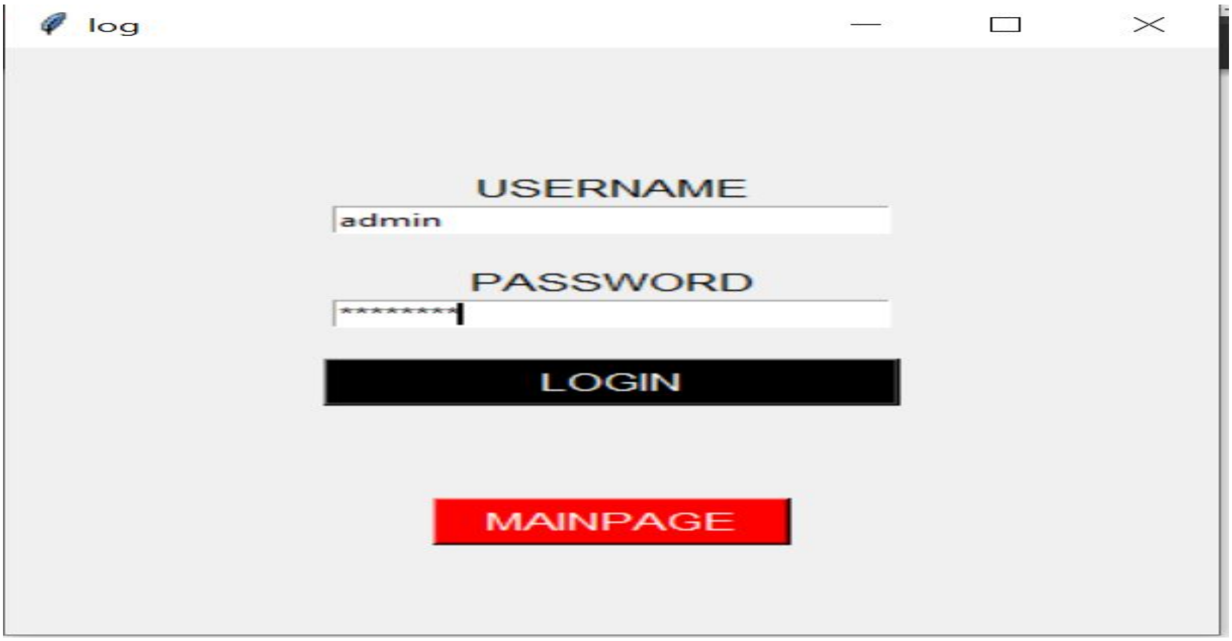
- **PIL(pillow):** Python Imaging Library is a free and open-source additional library for the Python programming language that adds support for opening, manipulating, and saving many different image file formats.

TESTING

MAINPAGE:



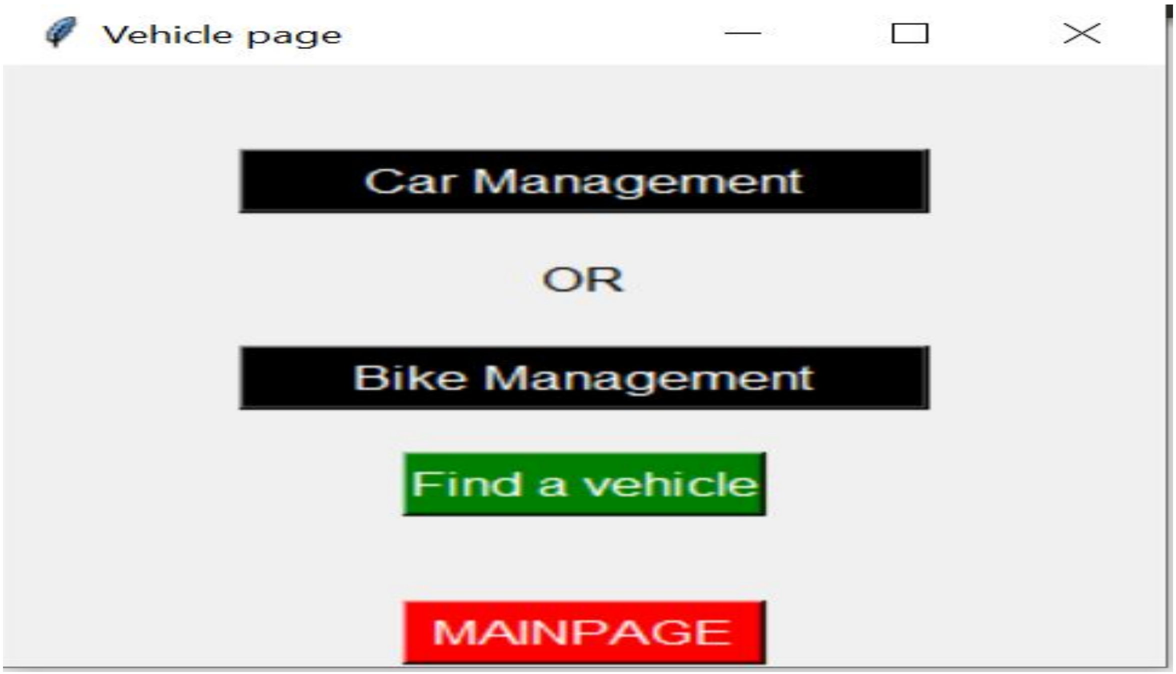
Admin login page:



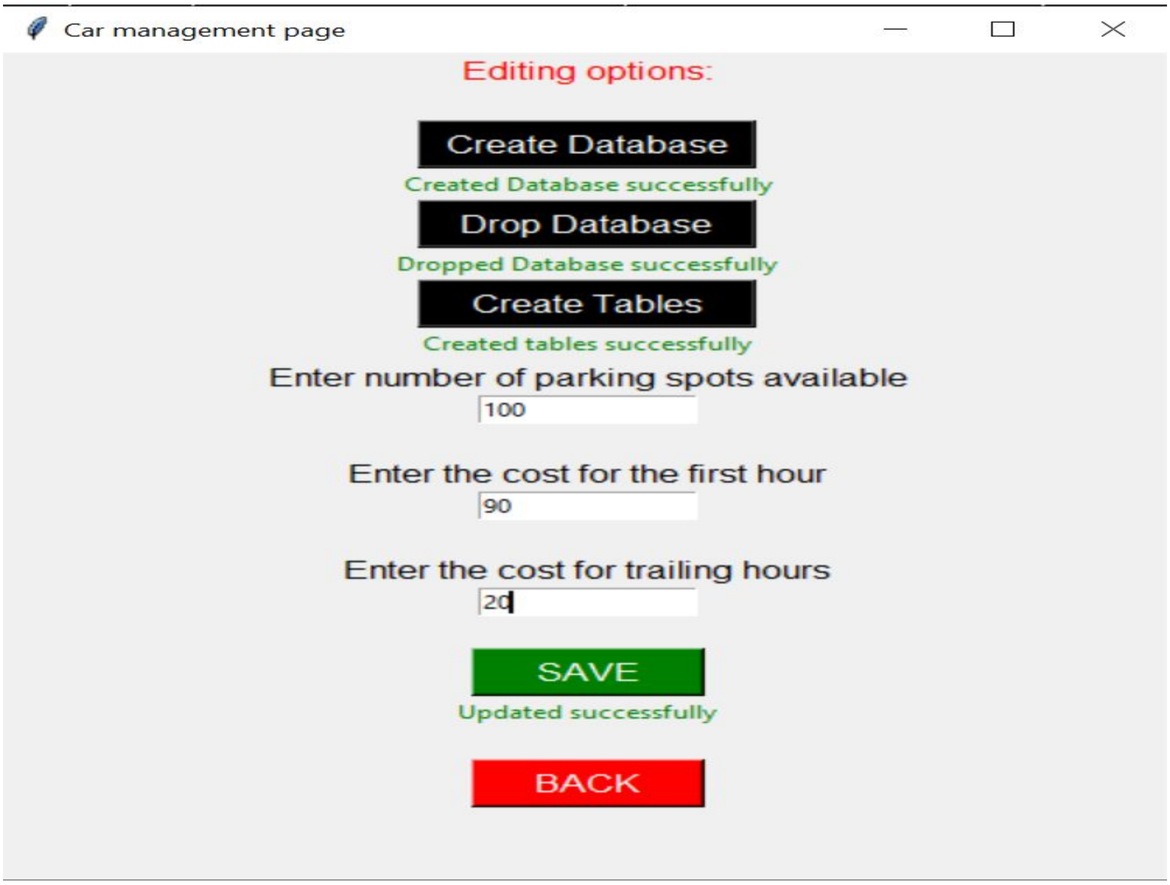
USERNAME: admin

PASSWORD: ********

Vehicle option page:



Car management page:



Created Database successfully

Dropped Database successfully

Created tables successfully

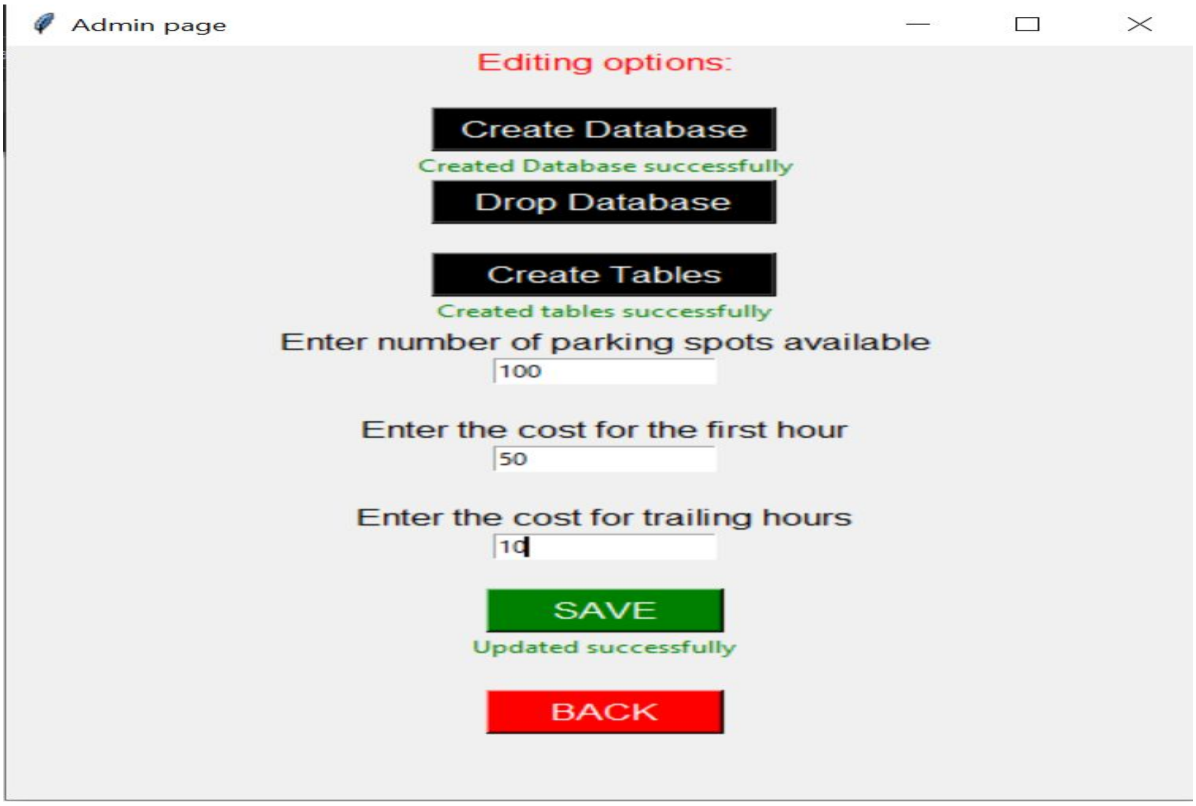
Enter number of parking spots available: 100

Enter the cost for the first hour: 90

Enter the cost for trailing hours: 20

Updated successfully

Bike management page:



Created Database successfully

Created tables successfully

Enter number of parking spots available

100

Enter the cost for the first hour

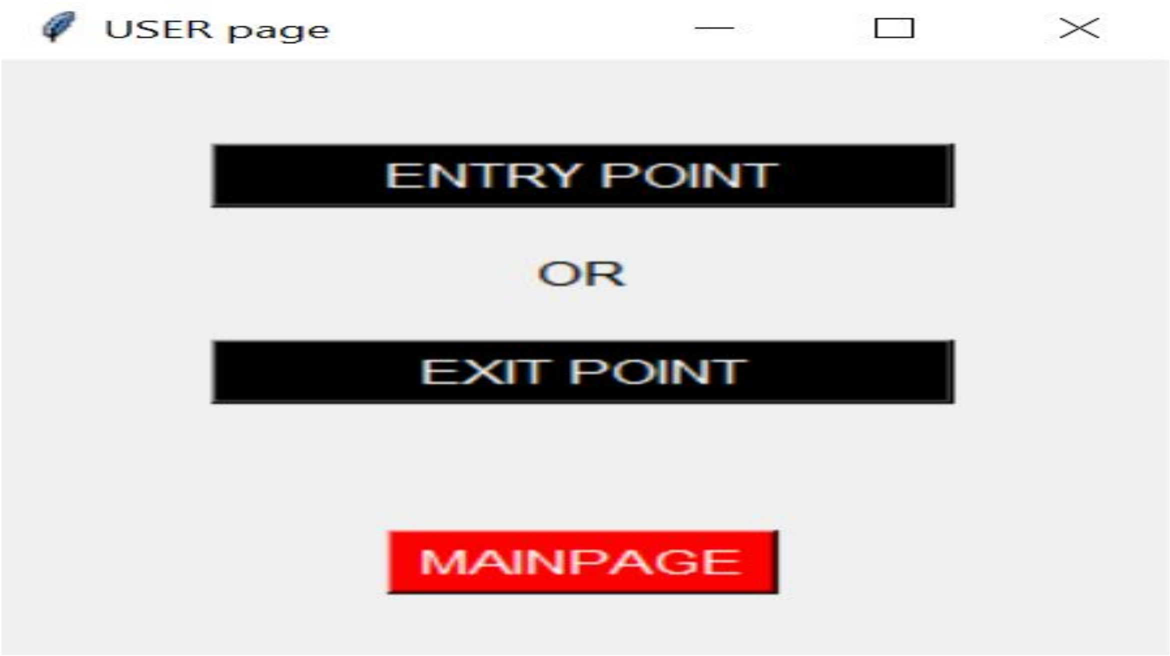
50

Enter the cost for trailing hours

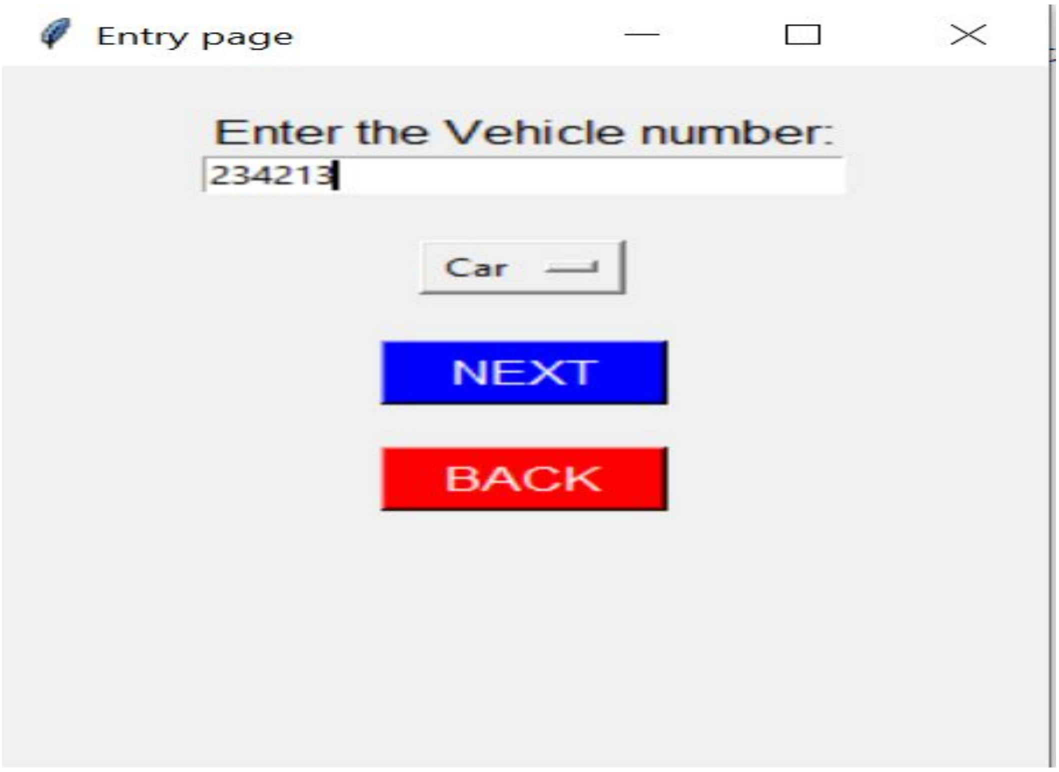
10

Updated successfully

User page:



Entry point page:



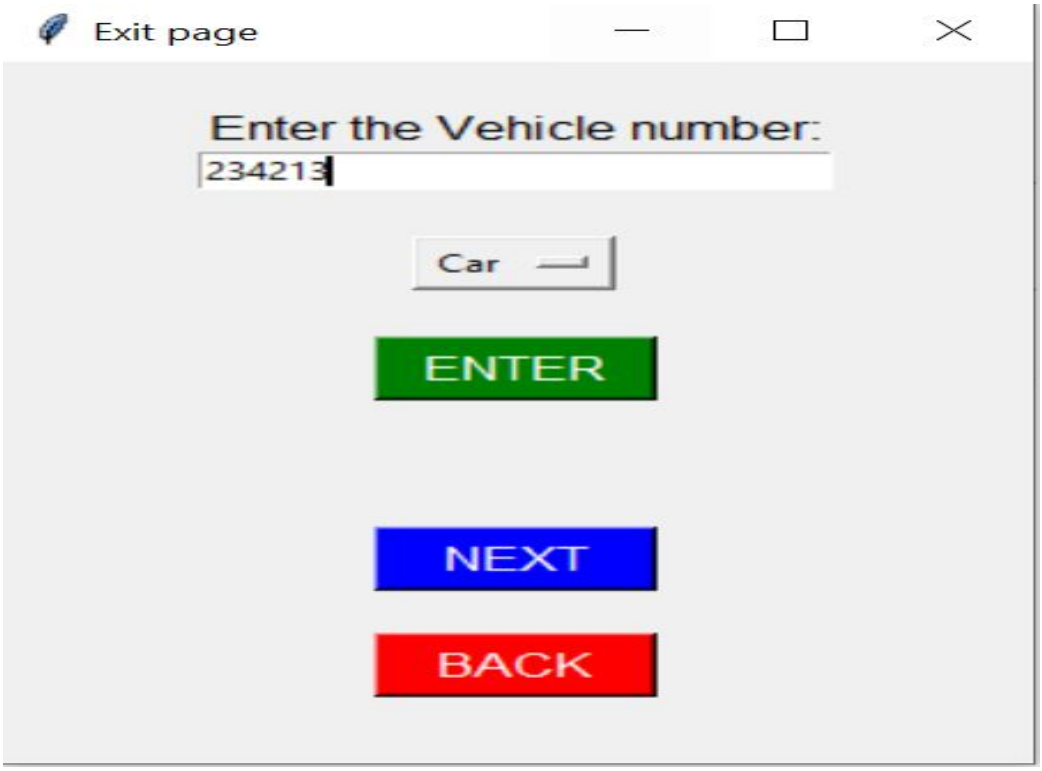
Entry page

Enter the Vehicle number:

234213

Car

Exit point page:



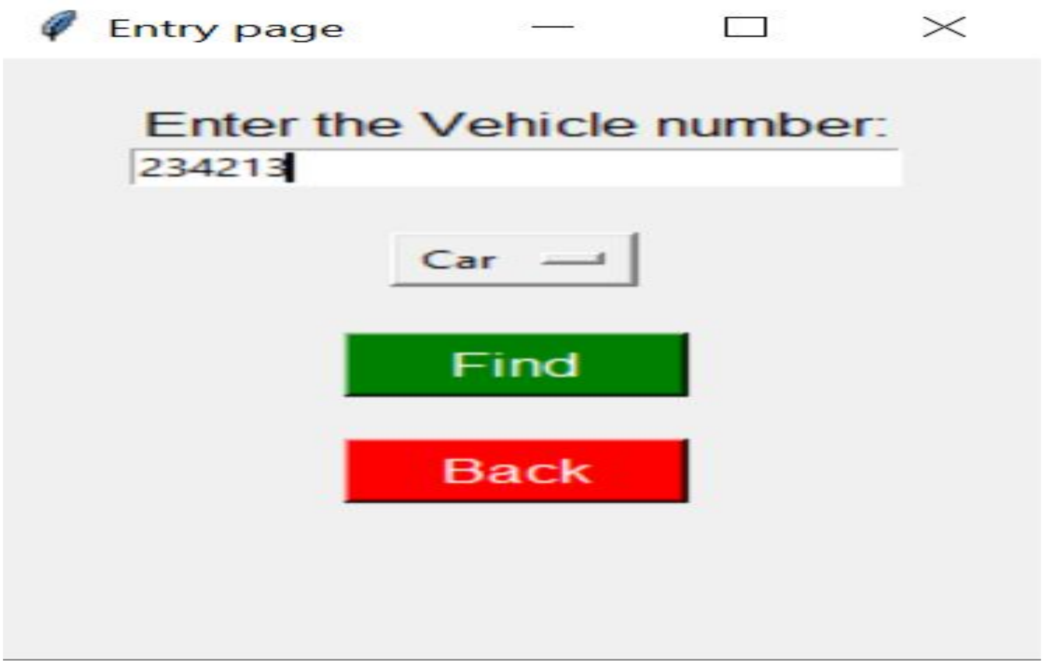
Exit page

Enter the Vehicle number:

234213

Car

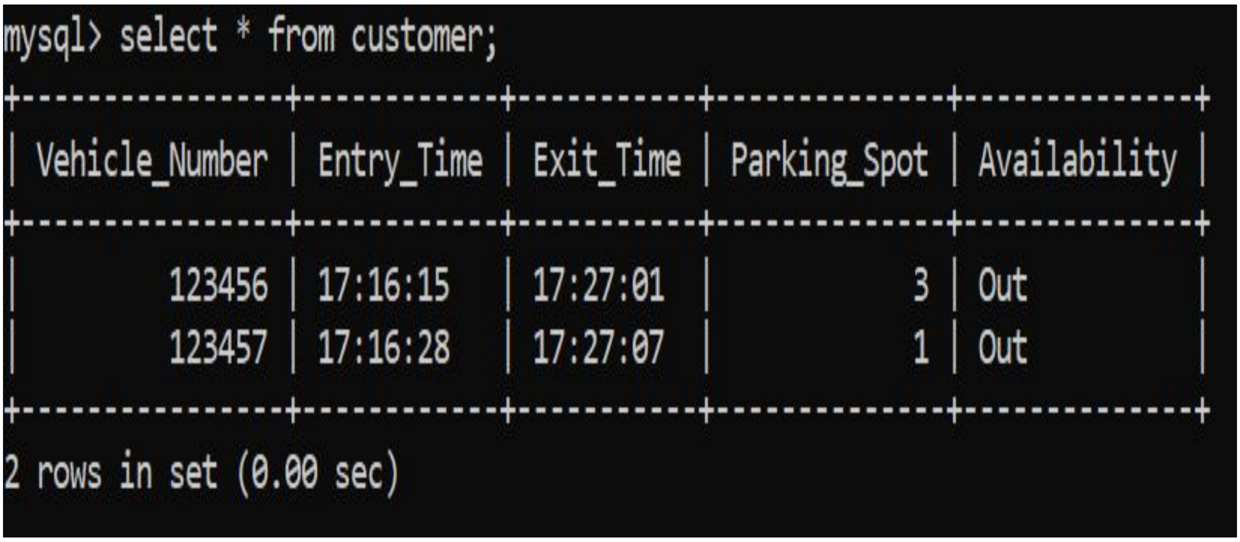
Find a vehicle page:



Enter the Vehicle number: 234213

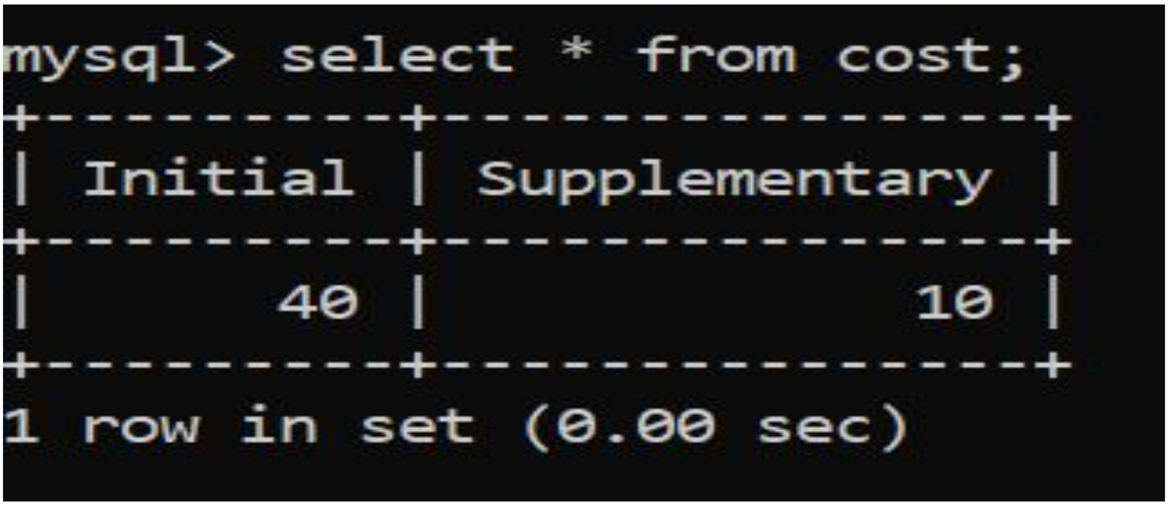
SQL TABLE SCREEN SHOTS:

Customer table:



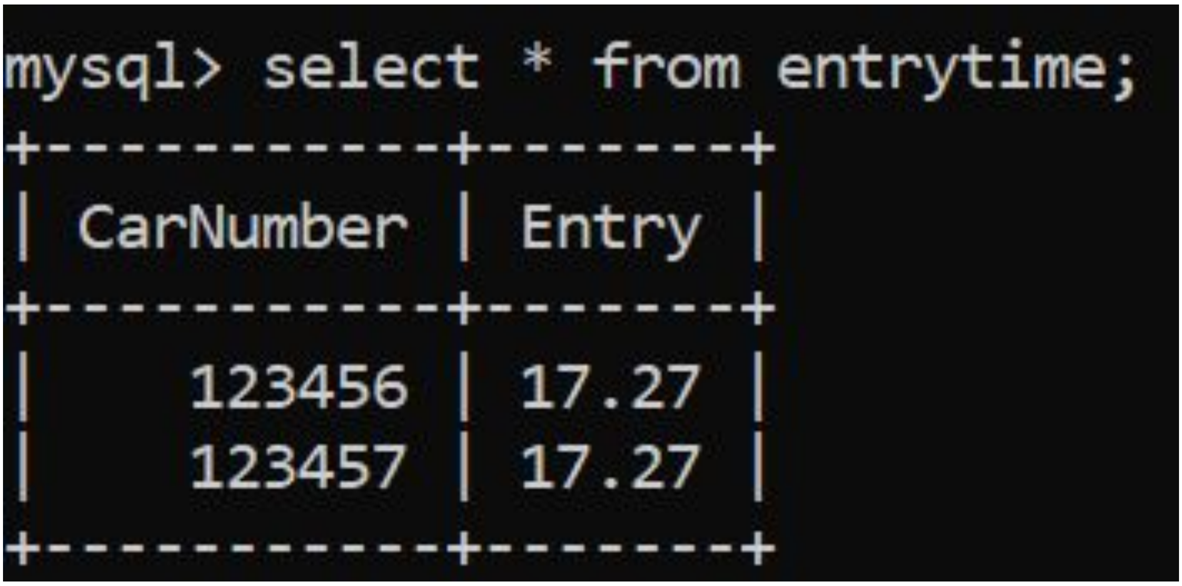
```
mysql> select * from customer;
+----------------+------------+-----------+--------------+--------------+
| Vehicle_Number | Entry_Time | Exit_Time | Parking_Spot | Availability |
+----------------+------------+-----------+--------------+--------------+
|         123456 | 17:16:15   | 17:27:01  |            3 | Out          |
|         123457 | 17:16:28   | 17:27:07  |            1 | Out          |
+----------------+------------+-----------+--------------+--------------+
2 rows in set (0.00 sec)
```

Cost table:



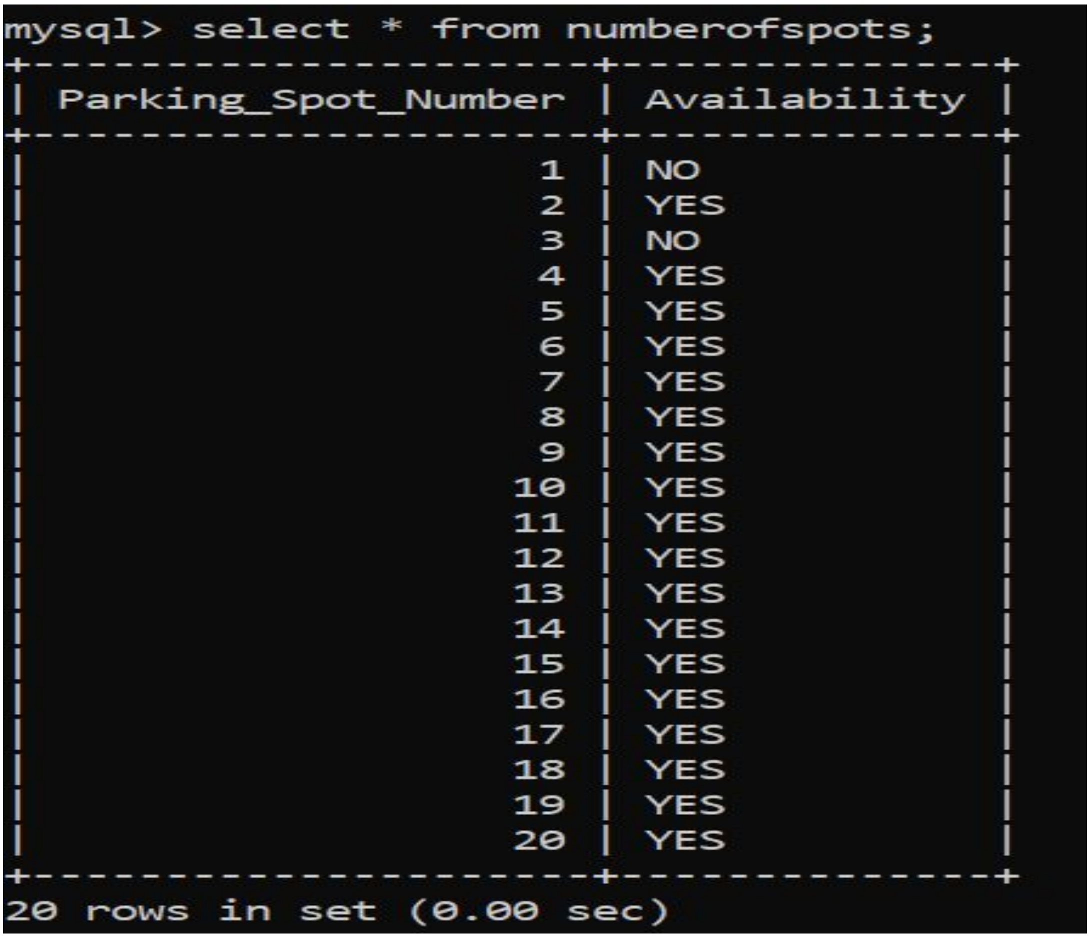
```
mysql> select * from cost;
+---------+---------------+
| Initial | Supplementary |
+---------+---------------+
|      40 |            10 |
+---------+---------------+
1 row in set (0.00 sec)
```

Entrytime table:



```
mysql> select * from entrytime;
+-----------+-------+
| CarNumber | Entry |
+-----------+-------+
|    123456 | 17.27 |
|    123457 | 17.27 |
+-----------+-------+
```

Spots table:



```
mysql> select * from numberofspots;
+---------------------+--------------+
| Parking_Spot_Number | Availability |
+---------------------+--------------+
|                   1 | NO           |
|                   2 | YES          |
|                   3 | NO           |
|                   4 | YES          |
|                   5 | YES          |
|                   6 | YES          |
|                   7 | YES          |
|                   8 | YES          |
|                   9 | YES          |
|                  10 | YES          |
|                  11 | YES          |
|                  12 | YES          |
|                  13 | YES          |
|                  14 | YES          |
|                  15 | YES          |
|                  16 | YES          |
|                  17 | YES          |
|                  18 | YES          |
|                  19 | YES          |
|                  20 | YES          |
+---------------------+--------------+
20 rows in set (0.00 sec)
```

FUTURE ENHANCEMENTS

Our project has minimized the time taken in parking a vehicle - through a better payment process and maintaining details about the customer and vehicle. Through this system, we can park our vehicles safely and securely. The efficiency of our system, with some minor modifications, can be enhanced. By integrating services like web pages and newer modules in the future, our system can progress further. We will tackle different parking systems in the future. We wish to enhance our system using more complex and efficient modules. We would also like to add a better user interface through the usage of other software.

BIBLIOGRAPHY

- www.w3schools.com
- BOOK - PYTHON IN PROGRESS
- https://pythonworld.in/practical-project/project-list/
- https://www.cs4school.com/cbse/python-project-for-class-12
- www.geeksforgeeks.com
- www.youtube.com
- www.python.org
- www.mysql.org
- www.pythonprogramming.org
- www.tutorialspoint.com
- WIKIPEDIA