

*Python
School
Copy*

Railway Reservation System

Name 1: Chandrakha P

Name 2: N Shrinidhi

Class: 12

Section: F1

Index

S.No.	Topic	Page No.
1	BONAFIDE	3
2	ACKNOWLEDGEMENT	4
3	PYTHON OVERVIEW	5
4	PROJECT DESCRIPTION	10
5	FUNCTIONS USED	11
6	TABLES USED	13
7	SOURCE CODE	15
8	SAMPLE OUTPUTS	95
9	CONCLUSION	99
10	BIBLIOGRAPHY	100



Chettinad Vidyashram

(Affiliated to Central Board of Secondary Education, New Delhi)
(Chettinad House, R.A.Puram, Chennai – 600 028)

COMPUTER SCIENCE

Certified to be the Bonafide Record of work done by

_____ of Std XII Sec _____

In the Computer Science Lab of the CHETTINAD VIDYASHRAM,
CHENNAI, during the year 2021 – 2022.

Date:

Teacher-in-charge

REGISTER NO. _____

Submitted for All India Senior Secondary Practical Examination in
Computer Science held on _____ at
Chettinad Vidyashram, Chennai – 600 028.

Principal

Internal Examiner

External Examiner

ACKNOWLEDGEMENT

I would like to express my sincere thanks to Meena Aunty, Principal Mrs. S.Amudhalakshmi for their encouragement and support to work on this Project. I am grateful to my computer science teacher UMA MAGESWARI R and to the computer science department for the constant guidance and support to complete the project.

Python Overview

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently whereas the other languages use punctuation and it has fewer syntactical constructions than other languages.

- ✓ **Python is interpreted** – Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
- ✓ **Python is interactive** – You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.
- ✓ **Python is object-oriented** – Python supports Object-Oriented style or technique of programming that encapsulates code within objects.
- ✓ **Python is Beginner's Language** – Python is a great language for beginner-level programmers and

supports the development of a wide range of applications from simple text processing to WWW browsers to games.

History of Python

- ✓ Python was developed by Guido van Rossum in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the Netherlands.
- ✓ Python is derived from many other languages, including ABC, Modula-3, C, C++, Algol-68, SmallTalk, and Unix shell and other scripting languages.
- ✓ Python is copyrighted. Like Perl, Python source code is now available under the GNU General Public License (GPL).
- ✓ Python is now maintained by a core development team at the institute, although Guido van Rossum still holds a vital role in directing its progress.

Python features

- ✓ **Easy to learn** - Python has few keywords, simple structure, and a clearly defined syntax. This allows the student to pick up the language quickly.
- ✓ **Easy to read** - Python code is more clearly defined and visible to the eyes.
- ✓ **Easy to maintain** - Python's source code is fairly easy to maintain.
- ✓ **A broad standard library** - Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.
- ✓ **Interactive mode** - Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.
- ✓ **Portable** - Python can run on a wide variety of hardware platforms and has the same interface on all platforms.
- ✓ **Extendable** - You can add low-level modules to the Python interpreter. These modules enable

programmers to add to or customize their tools to be more efficient.

- ✓ **Databases** - Python provides interfaces to all major commercial databases.
- ✓ **GUI Programming** - Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.
- ✓ **Scalable** - Python provides a better structure and support for large programs than shell scripting.

Other features:

- ✓ It supports functional and structured programming methods as well as OOP.
- ✓ It can be used as a scripting language or can be compiled to byte-code for building large applications.
- ✓ It provides very high-level dynamic data types and supports dynamic type checking.
- ✓ It supports automatic garbage collection.
- ✓ It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java.

Project Description

Railway Reservation System is a Python-based project. This has been developed using Python and MySQL. It is a computerised system of reserving the train seats in advance mainly used for a long route. This facilitates the users to enquire the trains available on the basis of source and destination, booking of tickets, cancellation of tickets, enquire about the status of booked ticket, etc. The aim is to create a database maintaining records of different trains, train status and passengers. Railway Reservation System can lead to error free and fast management system. The admin of the project, with the help of a password, can enter new train record, display all train records, modify train records and delete train records.

Functions Used

menu()	To display starting menus.
table()	To insert predefined train details to table.
HandlerMainMenu()	To display the available options in admin and main menu.
HandlerUserMenu()	To call specific user functions based on choice.
get_all_train()	To display all the available trains.
get_train()	To display all the available trains in specific route.
booking()	To accept the details of trains for booking.
book_ticket()	To fetch and display the desired trains from database.
GetPassengerInfo()	To accept the details and seats of the passengers.
GetPassengerInfoAndBook()	To insert the details in the database.
modify_details()	To fetch the details of the given PID from database.
promptandmodifybooking()	To prompt for modification and accept data if confirmed
updatemodifiedpassengerinfo()	To update the accepted details.
cancellation()	To get the details of booking with the PNR.

Tables Used

train_master	To store train details.
booking	To store ticket details.
passenger	To store passenger details.

```
mysql> desc train_master;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| Train_number | decimal(10,0) | NO |   |   |   |
| Train_Name | char(30) | NO |   |   |   |
| Start_place | char(30) | NO |   |   |   |
| End_place | char(30) | NO |   |   |   |
| Date_of_departure | date | NO |   |   |   |
| Date_of_arrival | date | NO |   |   |   |
| Departure_time | time | NO |   |   |   |
| Arrival_time | time | NO |   |   |   |
| No_of_seats | decimal(10,0) | NO |   |   |   |
| No_of_seats_remaining | decimal(10,0) | NO |   |   |   |
| Fare | decimal(10,0) | NO |   |   |   |
+-----+-----+-----+-----+-----+-----+
```

```
mysql> desc booking;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| Train_number | decimal(10,0) | NO |   |   |   |
| PID | decimal(10,0) | NO |   |   |   |
| travel_date | date | YES |   | NULL |   |
| PNR | decimal(10,0) | NO |   |   |   |
| seat_number | decimal(5,0) | NO |   |   |   |
| status | char(30) | YES |   | NULL |   |
+-----+-----+-----+-----+-----+-----+
```

Source Code

```
def
GetPassengerInfoAndBook(PassengerNumber,TrainNumber,DateOfTravel,PNRN
umber):

    import mysql.connector

    con=mysql.connector.connect(host='localhost', user='root', password='root',
charset='utf8', database='railways')

    cur=con.cursor()
    errorflag=0
    ErrorString=""

    PassengerAddress=AddressField.get(1.0,"end-1c")

    if len(PassengerAddress)>100:
        errorflag=1
        ErrorString=ErrorString+"Address Too Long\n"

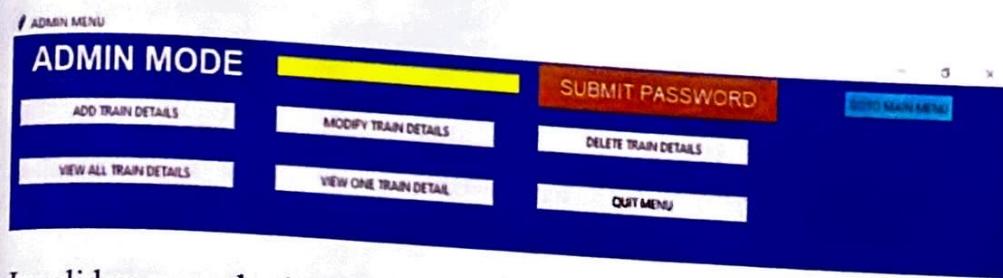
    PassengerPhoneNum=PhoneNumberField.get(1.0,"end-1c")
    try:
        PhoneNumValidated=int(PassengerPhoneNum)
        if len(PassengerPhoneNum)!=10:
            errorflag=1
            ErrorString=ErrorString+"Incorrect Phone Number\n"
    except ValueError:
        errorflag=1
        ErrorString=ErrorString+"Invalid Phone Number\n"
```

Sample Outputs

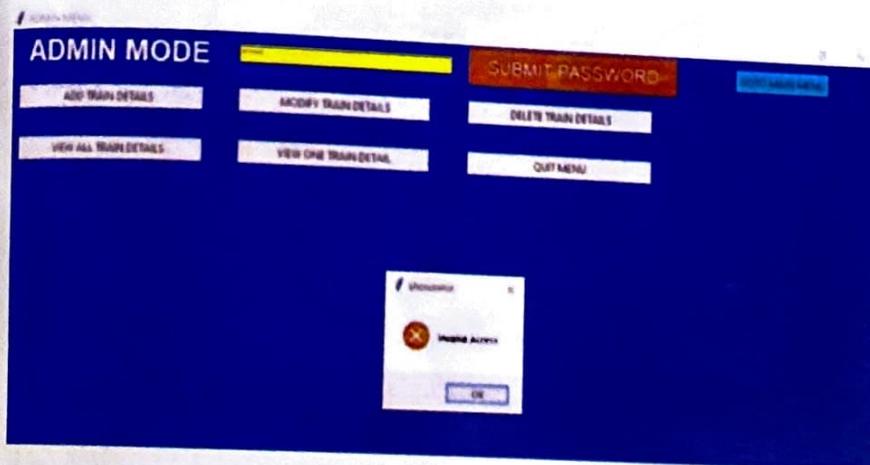
Main Screen:



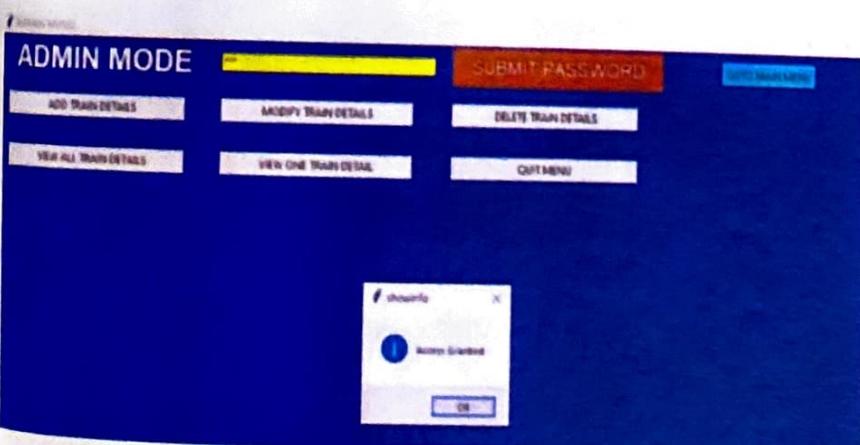
Admin Menu:



Invalid password rejected:



Now correct password is accepted:



Conclusion

- ✓ **Web-based project** – The project could be made web-based so that all users across could access the same from any part and from any device.
- ✓ **Classes** – The users could be given another option of various classes of seats (1st class, 2nd class, etc.) and the fare could be changed accordingly.
- ✓ **Stations between two places** – There could be an option of getting in and out of a train from stations in between the given start place and destination so that it would facilitate the easy travel for passengers.

Bibliography

- ✓ https://www.tutorialspoint.com/python/python_overview.htm
- ✓ <https://pythonguides.com>
- ✓ <https://likegeeks.com>
- ✓ <https://www.geeksforgeeks.org>
- ✓ <https://www.w3schools.com>
- ✓ Computer Science with Python Textbook for class 12
By Sumita Arora.