

Mini Project : TRAIN TICKET BOOKING APP

Objective:

Your project is to implement **TRAIN TICKET BOOKING APP** with options to **signIN** , **signUP**, booking ticket , cancelling ticket , checking train availabilities etc...

User Interface:

The application when executed, should display a menu as given below.
\$./myirctc

-----**MENU-1**-----

- 1: sign-up (to create new login)
- 2: sign-in (to login existing users)

✓ If option **1** chosen , then provide , **USR_NAME** and **PASSWORD**.

- ◆ Collect **USR_NAME** and **PASSWORD** and store in one file i.e., **usr_logins**.
- ◆ **PASSWORD** length minimum **8** characters long with at least **one** uppercase, lowercase, numeric and special.
- ◆ If not following above rules , print **error MSG** and ask to provide input again.
- ◆ If **5** times given wrong **PASSWORD** , ask to **signUP** again by printing **error MSG** like , **TIME OUT**.
- ◆ **USR_NAME** must be unique. Means , verify , is there any login exist or not with same **USR_NAME** , if exist , ask them to provide another **USR_NAME**.

✓ If option **2** chosen , then provide existing user details i.e., **USR_NAME** and **PASSWORD**.

- Check the given login details in a **usr_logins** file.
- If not found , ask to login again for another 3 times.
- If attempts completed , then print MSG like ., “You have reached max no.of attempts, please wait for 5 mins to login again.
- If login details found , then print another **MENU** to choose.

Print all available trains info.

Train info should contains train_number , train_name , source , destination.

-----**MENU-2**-----

R/r : To reserve ticket

C/c : To cancell ticket

B/b : Booking details

Q/q: Quit from app

- ✓ If option **R/r** choosen , then provide below asked info to reserve train ticket.

- ◆ **Train_number , Date of journey , No.of_seats to reserve**
Depends on no.of_seats, provide passingers details like , **name , age , male/female.**
- ◆ Store all passinger details in a file by assigning seat numbers to passingers.
- ◆ Consider , you have 5 seats available , but want to reserver 10 seats.
So, book first 5 seats for first 5 passingers and print equivalent **error MSG** on screen.

- ✓ If option **C/c** choosen , then provide below asked info to cancel ticket .

- ◆ **Train_number , Date of journey seat numbers to cancel**
- ◆ After cancelling tickets, please update passinger details in a file.
- ◆ Update the no.of seats availablity.

- ✓ If option **B/b** choosen , then provide below asked info to display booking details.

- ◆ **Train_number , Date of journey , seat numbers**
 - ◆ Open the file where passinger details saved and print only supplied seat numbers info.
-

Requirements:

- Maintain separate files to store **all trains_info** and **passingers_info** on particular data.
- **Separate structures** require to store **train_info** and **seats_info**.
- Train_info should contain , **train_name** , **train_number** , **source** , **destination** , **no.of seats**.
- Seats_info should contain , **total_seats** , **waiting_list** , **current_booking_seat_number**.
- Make sure that , duplicate trains should not be there. Means , train_name and train_number should be unique.

DELIVERABLES:

1. This app should contain **user-defined** functions for each and every task.
2. Use makefile and make tool to manage the project.
3. Use **readme.txt** to explain the usage of the project, how to compile, execute etc..
4. If we **re-launch** the app , old / previows data should be available.
5. Deliver the project, in a folder(named your ID), containing all source files, headerfiles, makefile, and readme.txt.

PROJECT approach :

1. Use of structure pointer with dynamic memory allocation.
2. FileHandling : File based functions like **save()**, **syncfile()**, should use to store data in file.
3. For every function , separate file should be implemented.
4. Implement this application by using SLL.

SUGGESTION:

- A) Use typedef, enum, union where-ever applicable.
- B) Use separate header file to keep all structure, union , and typedefs.

Note1: Maintain one / more special **USR_ID** for adding **new train_info** to trains list.

Note2: To reserver tickets , consider **only 3 days** from the current date.