School of Computer Science Engineering and Technology

Assignment No. 1

Objectives

1. Students will be able to learn SQL.
2. Student will be able to learn DDL commands and data types.
3. Student will be able to learn DML commands.

Case Study

Image Source: [https://techcrunch.com/wp-content/uploads/2013/02/india_bangalore_bus_.jpg?w=1390&crop=1]

Problem: Red Bus

RedBus is India's largest online bus ticketing platform that has transformed bus travel in the country by bringing ease and convenience to millions of Indians who travel using buses. Founded in 2006, redBus is part of India's leading online travel company MakeMyTrip Limited.

Goal:

The bus reservation system facilitates the passengers to enquire about the buses available based on the source and destination, booking and cancellation of tickets, enquire about the status of the booked ticket,etc. The aim of case study is to design and develop a database maintaining the records of different buses,buses status, and passengers. The record of buses includes its number, name, source, destination, and days on which it is available, whereas the record of bus status includes dates for which tickets can be booked, the total number of seats available, and the number of seats already booked. The database must be developedon MySQL platform.

Description:

Passengers can book their tickets for the bus in which seats are available. For this, the passenger has to provide the desired bus type and the date for which the ticket is to be booked. Before booking a ticket for apassenger, the validity of the bus type and booking date is checked. Once the bus number and booking dateare validated, it is checked whether the seat is available. If yes, the ticket is booked with confirm status and the corresponding ticket ID is generated which is stored along with other details of the passenger. Afterall the available tickets are booked, certain numbers of tickets are booked with waiting status. If the waitinglot is also finished, then tickets are not booked and a message of non-availability of seats is displayed. The ticket once booked can be cancelled at any time. For this, the passenger must provide the ticket ID(the unique key). The ticket ID is searched, and the corresponding record is deleted. With this, the firstticket with a waiting status also gets confirmed.

Assumptions:

Since the reservation system is very large in reality, it is not feasible to develop the case study to that extent and prepare documentation at that level. Therefore, a small sample case study has been created to demonstrate the working of the reservation system. To implement this sample case study, some assumptions have been made, which are as follows:

- The number of buses has been restricted to 7-10.
- The booking is considered only for 10 days from the current date.
- Only two categories of tickets can be booked, namely, AC and Non-AC.
- The status of the ticket will be either confirm or not confirmed.
- The source and destination of the buses have been

restricted to Tables will be created as follows:

1. **List_bus:** This table consists of details about all the available buses. The information stored in this table includes bus number, bus name, source, destination, fair for AC ticket, fair for generalticket, and weekdays on which bus is available. Constraints: The bus no. should be unique.

2. **Status_bus:** This table consists of details about the dates on which ticket can be booked for a bus and the status of the availability of tickets. The information stored in this table includes busnumber, bus date, total number of AC seats, total number of general seats, number of AC seats booked, and number of non-AC seats booked.

3. **Passenger_details:** This table consists of details about the booked tickets. The information stored in this table includes ticket ID, bus number, date for which ticket is booked, name, age, sex and address of the passenger, status of reservation (either confirmed or waiting), and categoryfor which ticket is booked. Constraints: Passenger_id should be unique.

List_Buses

Bus_no	Bus_name	Source	Destination	Fare	Date	Time
UP301	Zing bus	Pari Chauk	Prayagraj	1010	2022-08-21	21:30:00
HR302	Cargo	Sector 62	Chandigarh	1020	2022-08-23	22:44:00
UP101	Blue world	Pari Chauk	Lucknow	1240	2022-09-24	08:05:00
HR303	Maheshwaram	ISBT	Ambala	1010	2022-09-25	07:22:00
UP505	Goluxury	ISBT	Kanpur	2250	2022-09-27	19:38:00
DL701	Vaishnavi	ISBT	Chandigarh	1550	2022-09-28	23:55:00
DL306	Shatabdi	ISBT	Dehradun	1007	2022-09-29	20:45:00
UP501	Safar	Pari Chauk	Varanasi	1080	2022-09-30	08:35:00

Datatypes Used:
Bus_no **varchar();** Bus_name **varchar();** Source **varchar();** Destination **varchar();**Fare **int;** Date **date;** Time **time;**

Bus_Status

Bus_no	Available_seats	Booked_seats
UP301	35	25
HR302	41	24
UP101	10	51
HR303	14	48
UP505	01	59
DL701	08	46
DL306	17	45
UP501	50	00

Datatypes Used:
Bus_no **varchar();** Available_seats **int;** Booked_seats **int;**

Passengers

Passenger_id	P_Name	Bus_no	Gender	Age	Status
SCS012	Arjun	UP301	M	17	Confirm
SCS013	Anamika	HR302	F	15	Confirm
SCS012	Divya	UP101	F	65	Waiting
SCS014	Diya	HR303	F	19	Confirm
SCS015	Abhishek	UP505	M	57	Confirm
SCS090	Shiva	DL701	M	16	Waiting
SCS071	Rahul	DL306	M	21	Confirm
SCS043	Rupam	UP501	F	22	Confirm
SCS017	Hina	UP301	F	23	Waiting
SCS022	Alam	HR302	M	21	Waiting
SCS056	Satya	UP101	M	23	Confirm

Datatype Used:
Passenger_id **Varchar();**P_name **Varchar();** Bus_no **varchar();** Gender **varchar():** Age **int;** Status **varchar();**

Questions:

1. **Create a database named as Redbus online ticket booking system.**
2. **Use Redbus to enter in created Database**
3. **Create three tables named as List_buses, bus_status and passengers.**
4. **Display all columns and data of the table passengers.**
5. **Display all records from the passengers table for male passengers.**
6. **Find all records in the passengers table with null values in P_name column.**
7. **Find all the female passengers whose age is above 18.**
8. **Find all the buses whose source is either ISBT or Pari Chauk.**
9. **Add a column name as blood group in passengers table.**
10. **Delete the recently added column in the passengers table.**
11. **Create a new table new_passengers by coping all data from passengers table. Display all from new_passengers table.**
12. **Change datatype of any column of new_passengers table.**
13. **Change any column name of new_passengers table.**
14. **Delete the passengers from new_passengers table whose age is >50.**
15. **Change the age of shiva to 17.**