Lab 3 - DDL Commands

Introduction to SQL DDL Commands

Data Definition Language or DDL commands in standard query language(SQL) are used to describe/define the database schema. These commands deal with database schema creation and its further modifications. Some popularly known DDL commands are CREATE, ALTER, DROP, TRUNCATE, and COMMENT.

Command	**Description**
CREATE	Used for creating database objects like a database and a database table.
ALTER	Used for modifying and renaming elements of an existing database table.
DROP	Used for removing an entire database or a database table.
TRUNCATE	Used to remove all the records from a database table.
COMMENT	Used to write comments within SQL queries.

1. CREATE

CREATE is a data definition language(DDL) command that is used for creating database objects such as database and database table.

The syntax for creating a database is as follows :

```
CREATE DATABASE database_name;

CREATE TABLE table_name
(
column_name_1  datatype [NULL | NOT NULL],
column_name_2  datatype [NULL | NOT NULL],
.
.
.
column_name_n  datatype [NULL | NOT NULL] )
```

2. ALTER

ALTER command in SQL is used to add, rename or modify, drop/delete columns in an existing database table. It can further be used to add and remove various constraints on an existing database table.

The syntax used for altering a table in SQL by adding a new column is as follows :

```
ALTER TABLE table_name
ADD (Columnname_1  datatype)
```

3. TRUNCATE

TRUNCATE TABLE command is used to remove all the data records from the database table. It deletes all the rows permanently. we cannot perform a rollback operation to undo a TRUNCATE command.

The generic syntax used for writing TRUNCATE command is as follows :

```
TRUNCATE TABLE table_name;
```

4. DROP

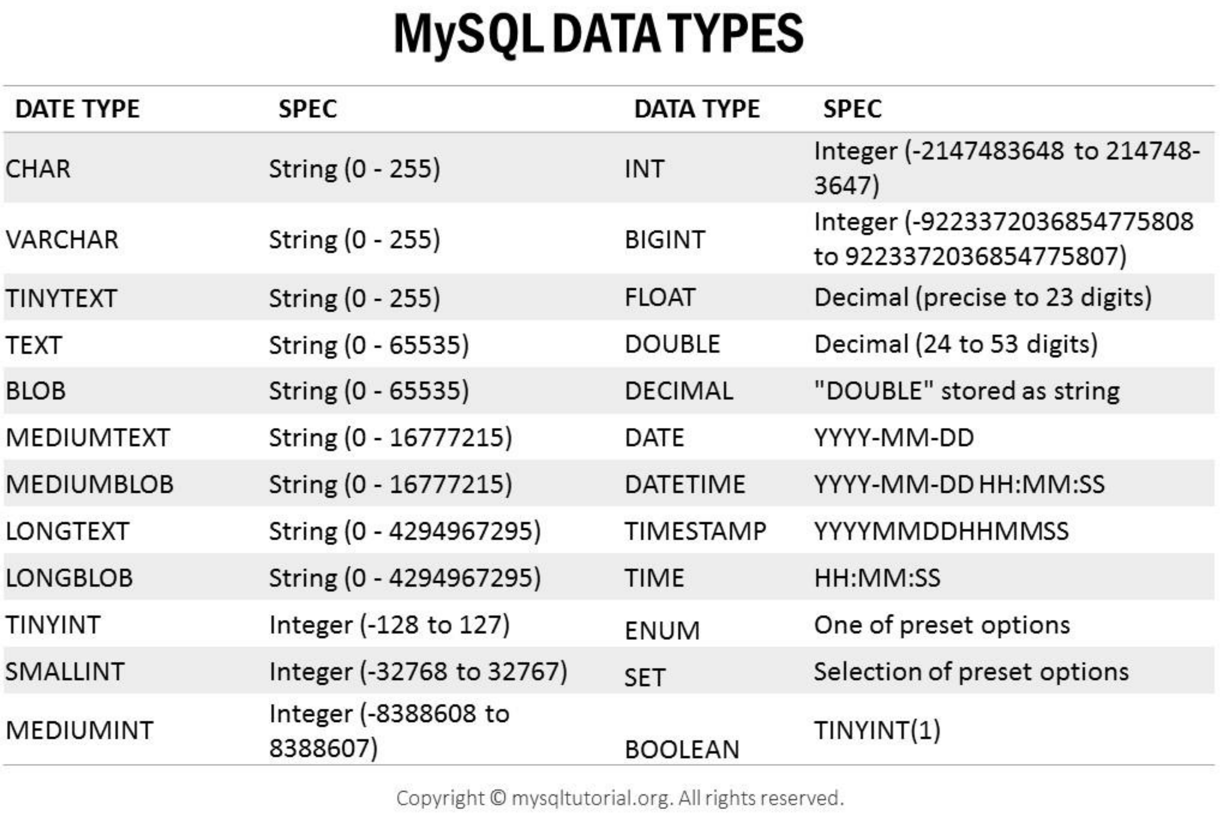
DROP TABLE SQL command is used to delete a database object from the database. We can even delete the database using the DROP command. We cannot perform a rollback operation to undo a DROP database/table command.

The basic syntax for writing DROP command to delete a database in SQL is as follows :

```
DROP DATABASE database_name;
```

Miscellaneous Commands:-

- show databases;
  - Show all the databases on the server
- show tables;
  - Show all the tables of the present database
- show columns from table EMPLOYEE;
- drop table *t_name*;
  - Delete the entire table *t_name*
- drop database *db_name*;
  - Delete the entire database *db_name*

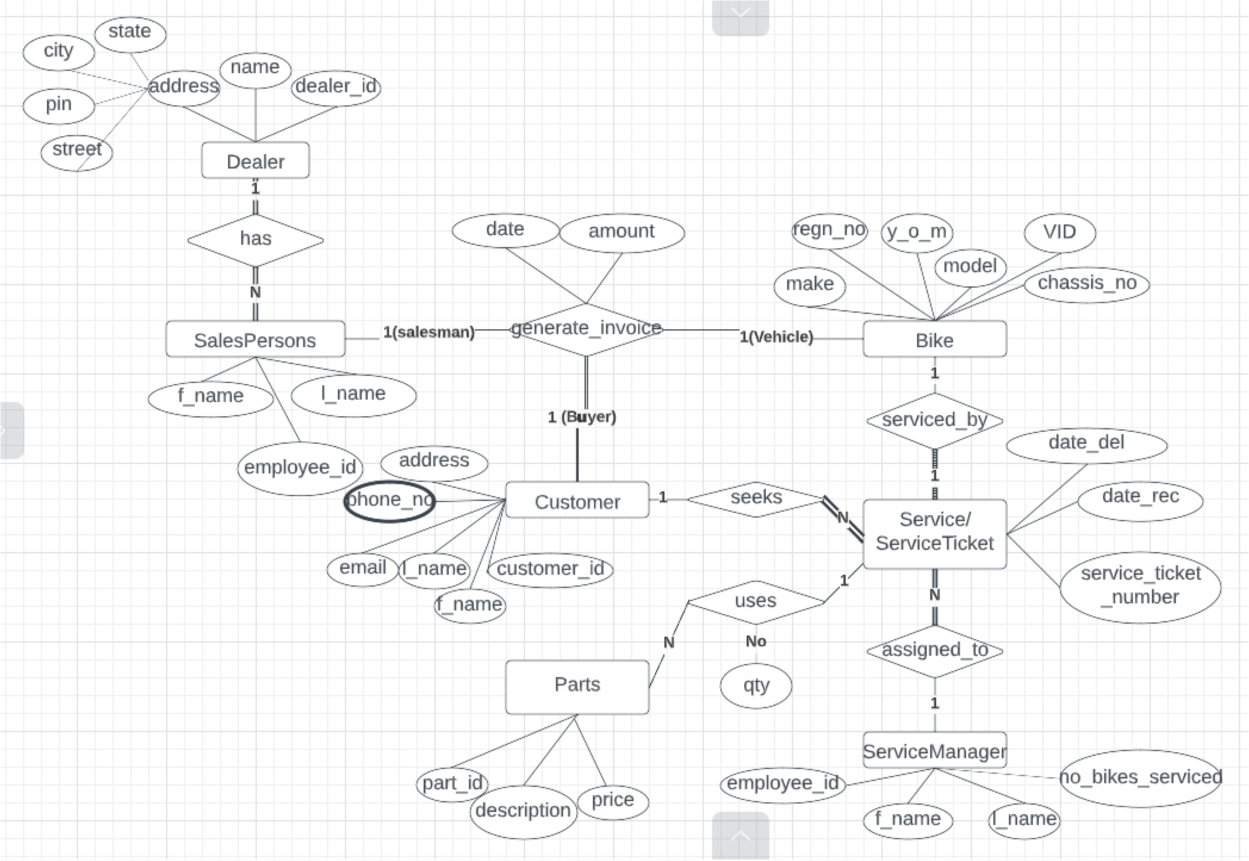


MySQL DATA TYPES

DATE TYPE	SPEC	DATA TYPE	SPEC
CHAR	String (0 - 255)	INT	Integer (-2147483648 to 214748-3647)
VARCHAR	String (0 - 255)	BIGINT	Integer (-9223372036854775808 to 9223372036854775807)
TINYTEXT	String (0 - 255)	FLOAT	Decimal (precise to 23 digits)
TEXT	String (0 - 65535)	DOUBLE	Decimal (24 to 53 digits)
BLOB	String (0 - 65535)	DECIMAL	"DOUBLE" stored as string
MEDIUMTEXT	String (0 - 16777215)	DATE	YYYY-MM-DD
MEDIUMBLOB	String (0 - 16777215)	DATETIME	YYYY-MM-DD HH:MM:SS
LONGTEXT	String (0 - 4294967295)	TIMESTAMP	YYYYMMDDHHMMSS
LONGBLOB	String (0 - 4294967295)	TIME	HH:MM:SS
TINYINT	Integer (-128 to 127)	ENUM	One of preset options
SMALLINT	Integer (-32768 to 32767)	SET	Selection of preset options
MEDIUMINT	Integer (-8388608 to 8388607)	BOOLEAN	TINYINT(1)

Demo Exercise

Consider the following ER Diagram for an E-Bike dealership. The dealership sells both new and used E-bikes, and it operates a service facility.



Create tables for the below-mentioned Relational design and add check, default, not null and unique constraints.

1)PK and FK constraints for all applicable Relations

2)For Fname , Lname in salesPerson table and customer table add not null constraints value

3) Add default constraint for service ticket

4) VIN should be unique

5)Add check constraints to DATE in invoice (data should be before 2021)

6)Rename table

7) use Truncate and drop commands

Solution:

1) Create tables:

i)Create table Dealer (Dealer_ID int NOT NULL, Name varchar(255), state varchar(255), city varchar(20),PIN int, street varchar(20),PRIMARY KEY(Dealer_ID));

ii)Create table SalesPerson (SalesPersonID int NOT NULL, FName varchar(255),LName varchar(255),Dealer_ID int,PRIMARY KEY (SalesPersonID), FOREIGN KEY (Dealer_ID) REFERENCES Dealer (Dealer_ID));

iii) Create table Customer (Cust_ID int NOT NULL, FName varchar(255), LName varchar(255),Address varchar(100),Email varchar(255),Mobile varchar(10),PRIMARY KEY (Cust_ID));

iv)Create table Bike (VIN int NOT NULL, Make varchar(30), Model varchar(30),
Reg_No varchar(10), Y_O_M int, Chasis_No varchar(20),PRIMARY KEY (VIN));

v)Create table Service_Manager (M_emp_id int NOT NULL,F_Name varchar(255),L_Name varchar(255),no_of_bike_service int,PRIMARY KEY (M_emp_id));

vi) Create table Service_Ticket (Service_ID int NOT NULL,VIN int,Cust_ID int, Date_Rec DATE,Date_Del DATE, M_emp_id int,Primary KEY(Service_ID), FOREIGN KEY (VIN) REFERENCES Bike(VIN), FOREIGN KEY (Cust_ID) REFERENCES Customer(Cust_ID),FOREGIN KEY (M_emp_id) REFERENCES Service_Manager(M_emp_id));

vii)Create table Parts (P_ID int NOT NULL, Description varchar(255),Qty int,Price DECIMAL(10,2), Service_ID int ,PRIMARY KEY (P_ID), FOREIGN KEY (Service_ID) REFERENCES Service_Ticket (Service_ID));

viii)Create table Invoice (SalesPersonID int, VIN int, Cust_ID int,Date DATE, Total_Bill decimal(10,2),PRIMARY KEY(SalesPersonID, VIN, Cust_ID), FOREIGN KEY (VIN) REFERENCES Bike(VIN), FOREIGN KEY (SalesPersonID) REFERENCES SalesPerson (SalesPersonID),FOREIGN KEY (Cust_ID) REFERENCES Customer(Cust_ID));

ix)Create table cust_phone(Cust_ID int,phone_No varchar(10),PRIMARY KEY(Cust_ID int,phone_No), FOREIGN KEY (Cust_ID) REFERENCES Customer(Cust_ID));

2) ALTER TABLE SalesPerson MODIFY FName varchar(255) NOT NULL, MODIFY LName varchar(255) NOT NULL;

3)ALTER table Bike ADD UNIQUE (VIN);

4)ALTER table Invoice ADD CHECK(Date>=2021);

5) Rename table parts on Accessories;

6) Truncate black_bike;

7) Drop balck_bike;

Assignment:

Railway Reservation System

Users check for availability of seats/Fare etc. in Trains and also books/cancels Train Tickets from particular source stations to destination stations. An user may be an admin (reservations clerks) or a passenger.

Every train has at least one compartment and at most 5 compartments. Each compartment is of type I class/ II Class / III Class corresponding to the ticket class. Number of seats in each type of compartment is fixed and is equal to 16 , 30, and 60 respectively.

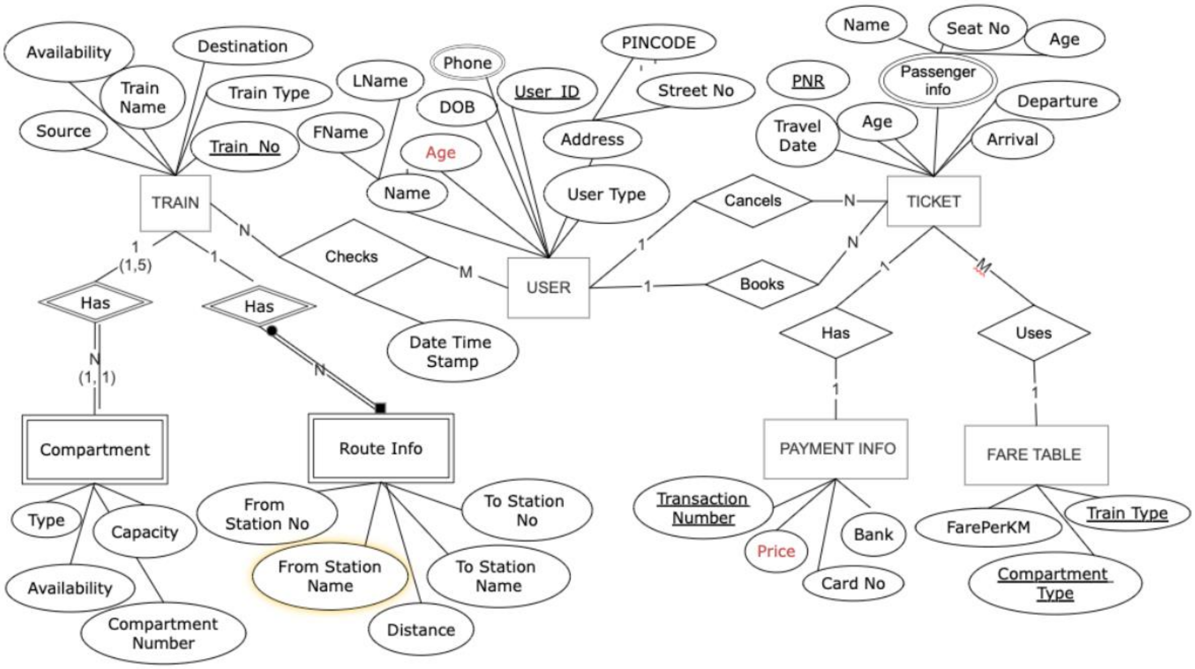
Every train has associated route information that stores distance between each pair of stations along the route

Tickets compute fare from a Fare table that stores fare per KM rate for a given type of train (Superfast, Fast and Mail), for each class of ticket (I /II/ III Class)

Every ticket booked/canceled has associated payment information.

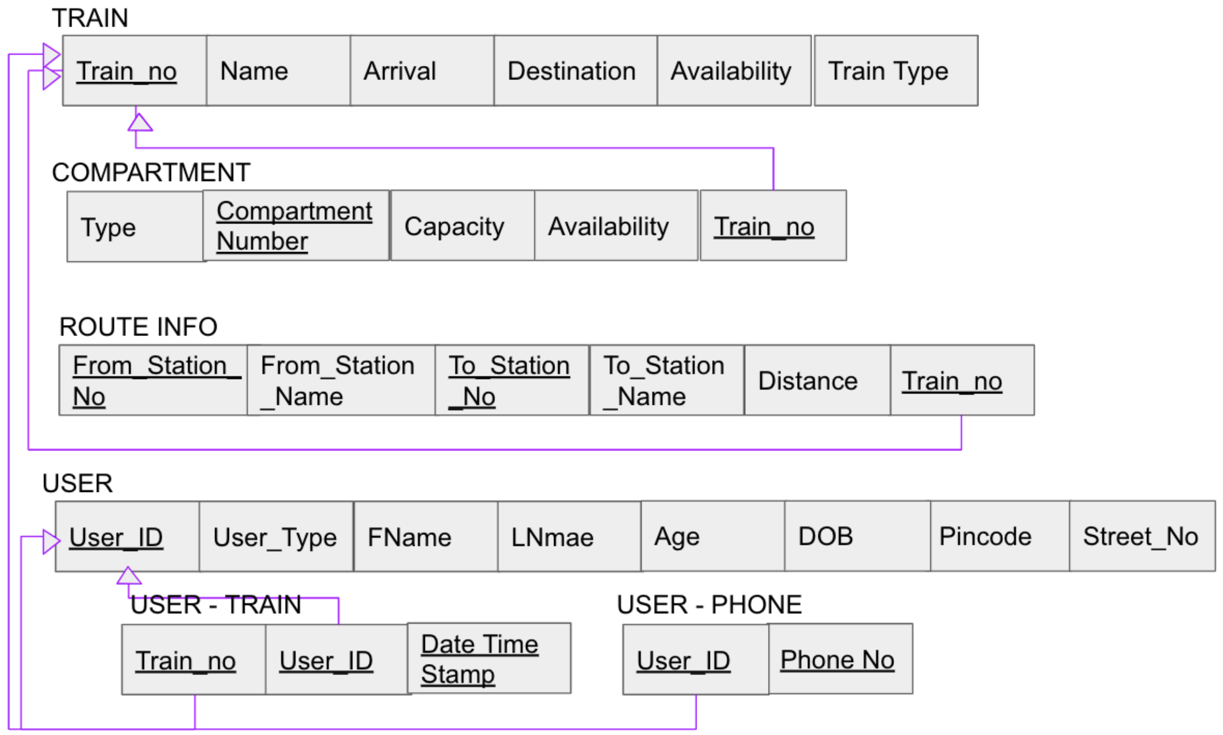
Identify entities and relationships, weak entities, Identifying relationships, total relationships, cardinalities and restrictions of each relationship.

ER Diagram:



Age and Price - Derived Attributes

Railway Reservation System Schema



TRAIN

Train_no	Name	Arrival	Destination	Availability	Train Type

COMPARTMENT

Type	Compartment Number	Capacity	Availability	Train_no

ROUTE INFO

From_Station_No	From_Station_Name	To_Station_No	To_Station_Name	Distance	Train_no

USER

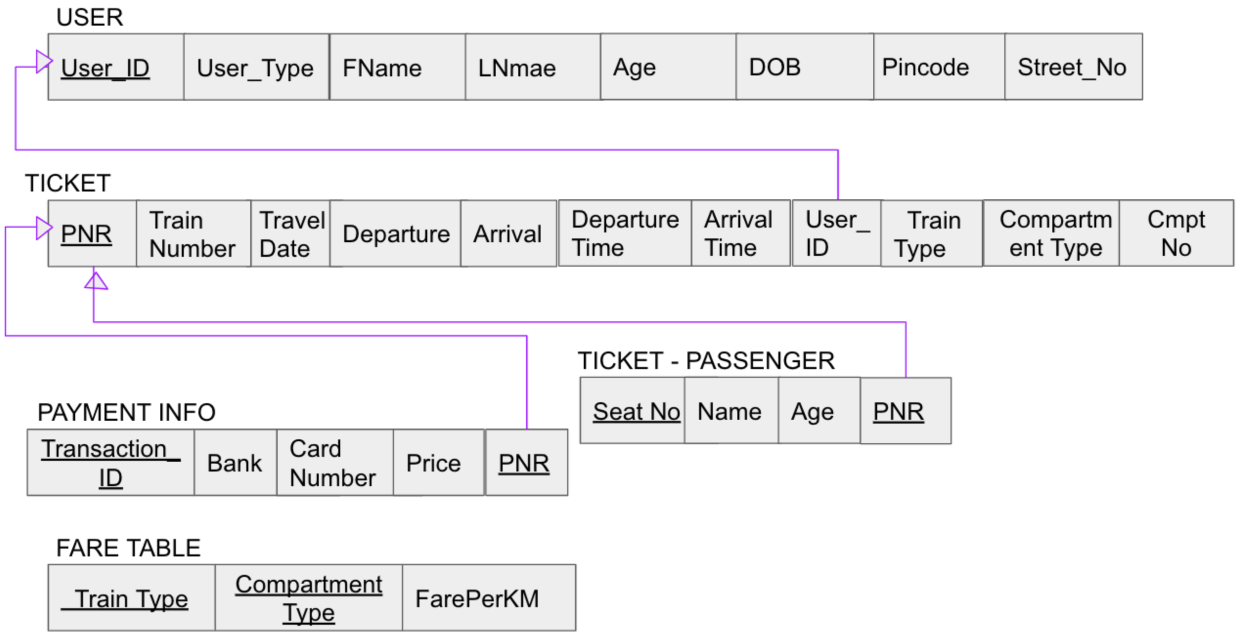
User_ID	User_Type	FName	LNmae	Age	DOB	Pincode	Street_No

USER - TRAIN

Train_no	User_ID	Date Time Stamp

USER - PHONE

User_ID	Phone No



USER

User_ID	User_Type	FName	LNmae	Age	DOB	Pincode	Street_No

TICKET

PNR	Train Number	Travel Date	Departure	Arrival	Departure Time	Arrival Time	User_ID	Train Type	Compartment Type	Cmpt No

TICKET - PASSENGER

Seat No	Name	Age	PNR

PAYMENT INFO

Transaction_ID	Bank	Card Number	Price	PNR

FARE TABLE

Train Type	Compartment Type	FarePerKM

Create tables for the Above-mentioned Relational design and add check, default, not null and unique constraints.

1)Specify PK and FK constraints for all applicable Relations

2)For Train_name, Train_Type in Train table add not null constraints

value

3) Add default constraint for compartment table setting Availability attribute to Yes

4)train name should be unique

5)Add check constraints to Ticket to check if the passenger age is above 5

6)Rename any existing table name

7) use Truncate and drop commands