# Ex. No. 1(a) – Creating and Managing Tables

**Aim**

To execute DDL commands and get the desired output.

# Description

# DDL refers to "Data Definition Language", a subset of SQL statements that change the structure of the database schema in some way, typically by creating, deleting, or modifying schema objects such as databases, tables, and views. Most Impala DDL statements start with the keywords CREATE , DROP , or ALTER .

In this schema, we have four main tables: User, Event, Venue, and Ticket.

|  |  |
| --- | --- |
| **User table:** |  |
| **Column** | **Data Type** |
| UserID | NUMBER(10) |
| Name | VARCHAR2(255) |
| Email | VARCHAR2(255) |
| Password | VARCHAR2(255) |
| Phone | VARCHAR2(20) |
|  |  |
|  |  |
| **Event table:** |  |
| **Column** | **Data Type** |
| EventID | NUMBER(10) |
| Name | VARCHAR2(255) |
| Date | DATE |
| Time | TIMESTAMP |
| VenueID | NUMBER(10) (Foreign key) |
| Description | VARCHAR2(500) |
| **Venue table:** |  |
| **Column** | **Data Type** |
| VenueID | NUMBER(10) |
| Name | VARCHAR2(255) |
| Address | VARCHAR2(255) |
| City | VARCHAR2(255) |
| State | VARCHAR2(255) |
| Country | VARCHAR2(255) |
| **Ticket table:** |  |
| **Column** | **Data Type** |
| TicketID | NUMBER(10) |
| EventID | NUMBER(10) (Foreign key) |
| UserID | NUMBER(10) |
| SeatNumber | VARCHAR2(20) |
| Price | NUMBER(10, 2) |
| Status | VARCHAR2(50) |

## Queries (DDL Commands)

## Create User, Event, Venue, and ticket tables based on the given schema.

## User table: Create table user\_1128(User\_ID number(10), Name varchar(255), Email varchar(255), Password varchar(255),Phone number(20));

## 

## Event table: Create table event\_1128(Event\_ID number(10), Name varchar(255), Date\_event DATE, Time\_event TIMESTAMP, Venue\_ID number(10), Description varchar(255));

## 

## Venue table: Create table venue\_1128(Venue\_ID number(10), Name varchar(255),Address varchar(255), City varchar(255),State varchar(255), Country varchar(255));

## 

## Ticket table: Create table Ticket\_1128(Ticket\_ID number(10), Event\_ID number(10),User\_ID number(10),Seat\_Number number(10),Price number(10,2),Status varchar(255));

## 

## 2. Describe the tables.

## User Table: desc user\_1128;

## 

## Event Table: desc event\_1128;

## 

## Venue Table: desc venue\_1128;

## 

## Ticket Table: desc Ticket\_1128;

## 

## Alter the User table to add a new column Age.

## Query: alter table user\_1128 add Age number(10);

## 

## Drop the newly added column Age.

## Query: alter table user\_1128 drop column Age;

## 

## Rename the Venue table to Location

## Query: alter table venue\_1128 rename to Location;

## 

## 

## Modify the size of the Event table & Description column to 1000.

## Query: alter table event\_1128 modify Description varchar(1000);

## 

## Drop the SeatNumber column from the Ticket table

## Query: alter table Ticket\_1128 drop column Seat\_Number;

## 

## 

## Add a unique constraint on the Email column in the User table

## Query: alter table user\_1128 add constraint Email UNIQUE(Email);

## 

## Rename the UserID column in the User table to ID

## Query: alter table user\_1128 rename column User\_ID to ID;

## 

## Modify the Ticket table to add a column named Barcode with a data type of VARCHAR2(50).

## Query: alter table Ticket\_1128 add Barcode varchar2(50);

## 

## Modify the Name column in the Venue table to increase its maximum length to

## VARCHAR2(300)

## Query: alter table Location modify Name varchar (300);

## 

## 

## 12. Add a foreign key constraint on the VenueID column in the Event table, referencing

## the Venue table.

## Query: alter table event\_1128 modify Venue\_ID number references Location (Venue\_ID);

## 

## 13. Add a CHECK constraint to check whether the UserID is between 101 and 105

## Query: alter table user\_1128 add constraint ck check(ID between 101 and 105);

## 

## 14. Add a unique constraint to Phone column of the User table.

## Query: alter table user\_1125 add constraint Phone UNIQUE(Phone);

## 

## 15. Truncate the user table.

## Query: truncate table user\_1128;

## 

## Result:

## The DDL commands are executed successfully and shown the valid output.