|  |  |
| --- | --- |
| **EX NO :1 A** | **Creating and Managing Tables using DDL commands** |
| **DATE** | **17-12-2024** |

# Aim

To execute Data Definition Language 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, RENAME and ALTER.

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

**User table:**

|  |  |
| --- | --- |
| **Column** | **Data Type** |
| UserID | INT/NUMBER (10) |
| Name | VARCHAR2(25) |
| Email | VARCHAR2(25) |
| Password | VARCHAR2(25) |
| Phone | VARCHAR2(20) |

**Event table:**

|  |  |
| --- | --- |
| **Column** | **Data Type** |
| EventID | INT/NUMBER (10) |
| Name | VARCHAR2(25) |
| Date | DATE |
| Time | VARCHAR |
| VenueID | INT/NUMBER (10) |
| Description | VARCHAR2(500) |

**Venue table:**

|  |  |
| --- | --- |
| **Column** | **Data Type** |
| VenueID | INT/NUMBER (10) |
| Name | VARCHAR2(25) |
| Address | VARCHAR2(25) |
|  |  |

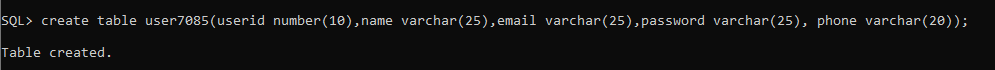
|  |  |
| --- | --- |
| City | VARCHAR2(25) |
| State | VARCHAR2(25) |
| Country | VARCHAR2(25) |

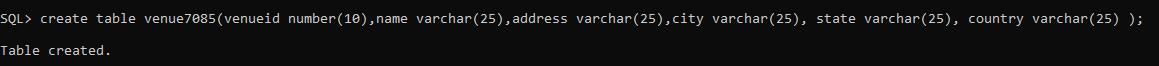
**Ticket table:**

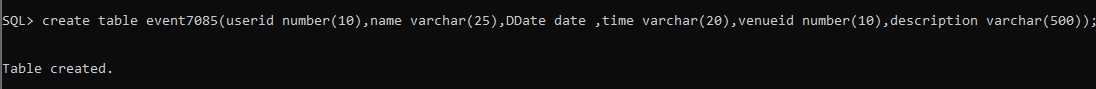
|  |  |
| --- | --- |
| **Column** | **Data Type** |
| TicketID | INT/NUMBER (10) |
| EventID | INT/NUMBER (10) |
| UserID | INT/NUMBER (10) |
| SeatNumber | VARCHAR2(20) |
| Price | INT/NUMBER (10) |
| Status | VARCHAR2(50) |

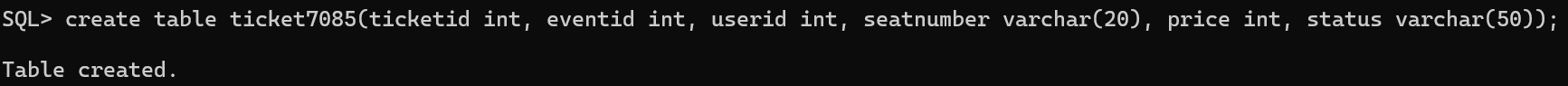
# Queries (DDL Commands)

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

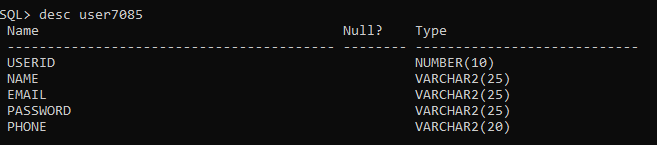


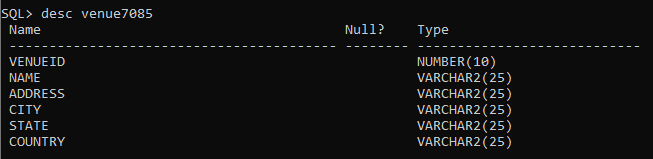


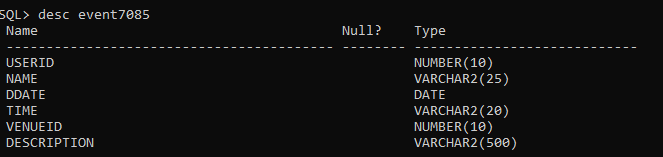


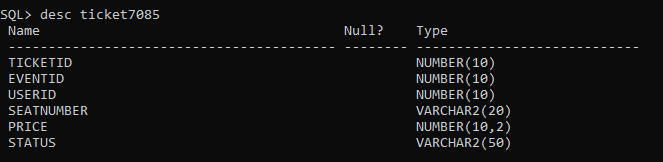


1. Describe the tables.

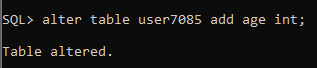




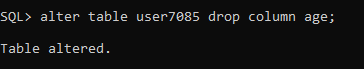




1. Alter the User table to add a new columnAge.



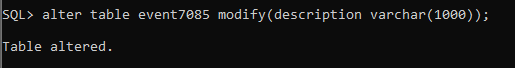
1. Drop the newly added columnAge.



1. Rename the Venue table to Location

unnamed (27)

1. Modify the size of the Event table's Description column to 1000.



1. Drop the Seat Number column from the Ticket table

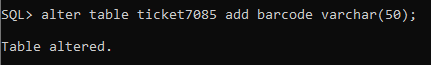
unnamed (29)

1. Rename the UserID column in the User table to ID

unnamed (30)

1. Modify the Ticket table to add a column named "Barcode" with a data

type of VARCHAR2(50).



1. Modify the Name column in the Venue table to increase its maximum

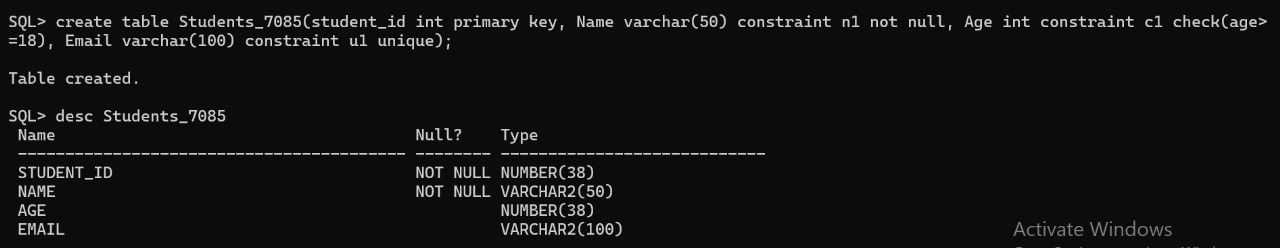
length to VARCHAR2(300).

unnamed (32)

1. Write an SQL query to create a table named Students with the following structure: StudentID (Integer, Primary Key)

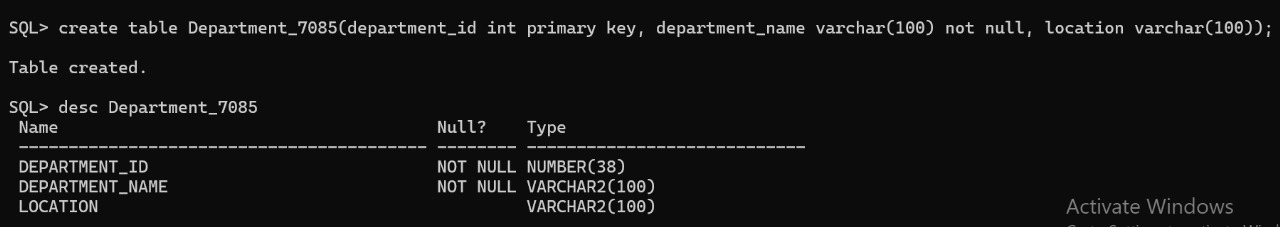
Name (Variable Character, max length 50, cannot be NULL) Age (Integer, must be between 18 and 25)

Email (Variable Character, max length 100, unique)



1. Write an SQL query to create a table named Department with the following structure: DepartmentID (Integer, Primary Key)

DepartmentName (Variable Character, max length 100, cannot be NULL) Location (Variable Character, max length 100)

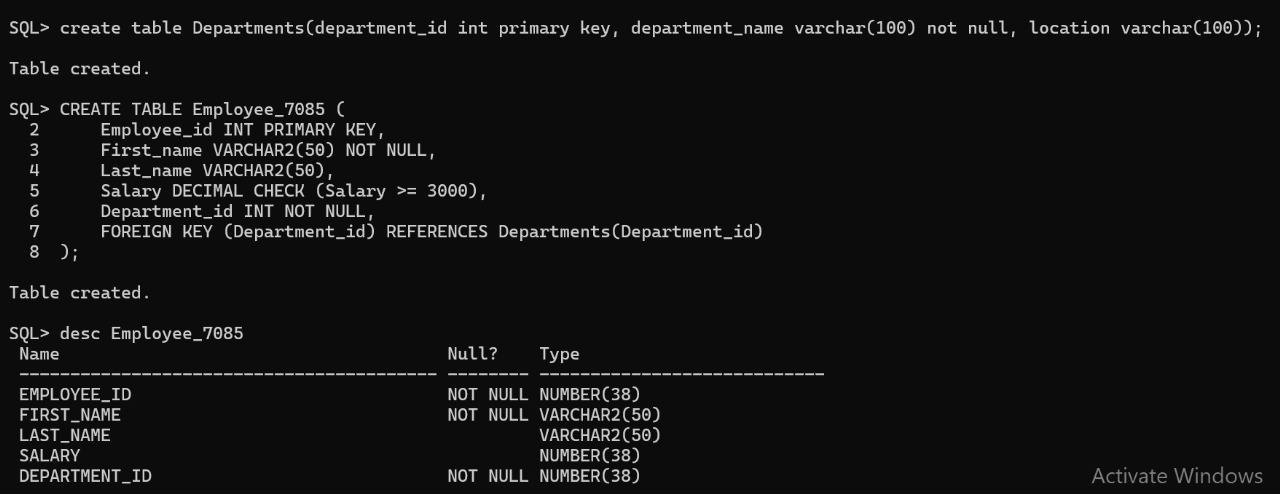


1. Write an SQL query to create a table named Employees with the following constraints: EmployeeID (Integer, Primary Key)

FirstName (Variable Character, max length 50, cannot be NULL) LastName (Variable Character, max length 50)

Salary (Decimal, must be greater than or equal to 3000)

DepartmentID (Integer, Foreign Key referencing Departments (DepartmentID), cannot be NULL)



**Result**: The above command is successfully compiled and executed.