**JAVA FOUNDATION TRAINING**

**BATCH 8**

**Name:** Rajalakshmi Ganesh             **Case Study:** TicketBookingSystem

**Tasks 1: Database Design**

**1. Create the database named "TicketBookingSystem"**

🡪 CREATE DATABASE TicketBookingSystem;

🡪 USE TicketBookingSystem;

**2. Write SQL scripts to create the mentioned tables with appropriate data types, constraints, and relationships (Venu, Event, Customers, Booking)**

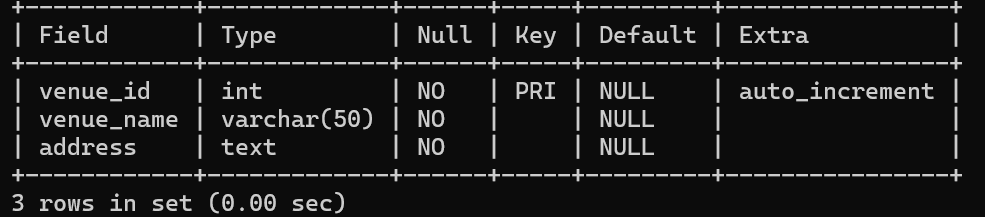
* **Venue**

**🡪** CREATE TABLE Venue (

venue\_id INT AUTO\_INCREMENT PRIMARY KEY,

venue\_name VARCHAR (50) NOT NULL,

address TEXT NOT NULL);

****

* **Event**

**🡪** CREATE TABLE Event (

event\_id INT AUTO\_INCREMENT PRIMARY KEY,

event\_name VARCHAR (100) NOT NULL,

event\_date DATE NOT NULL,

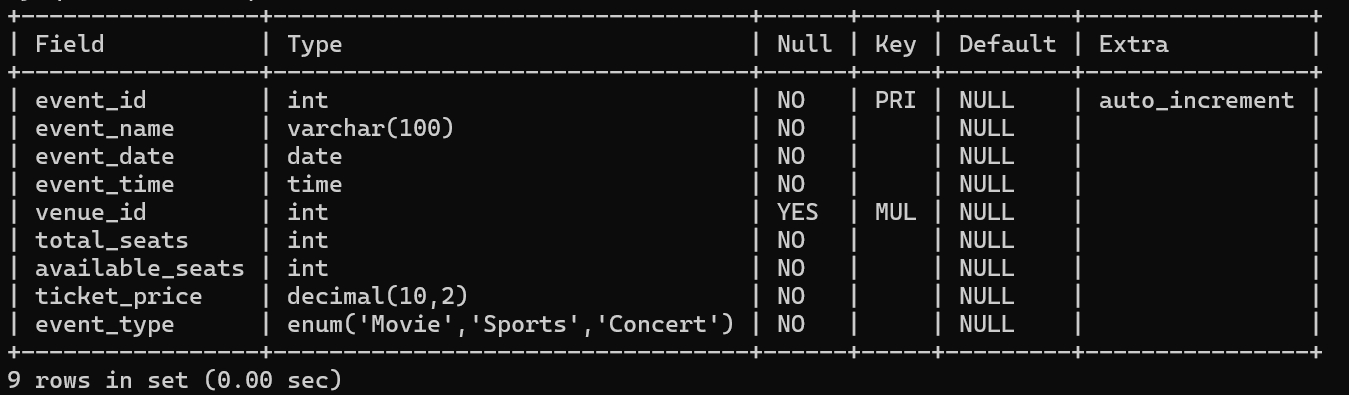
event\_time TIME NOT NULL,

venue\_id INT,

total\_seats INT NOT NULL CHECK (total\_seats > 0),

available\_seats INT NOT NULL CHECK (available\_seats >= 0),

ticket\_price DECIMAL (10,2) NOT NULL CHECK (ticket\_price >= 0), event\_type ENUM ('Movie', 'Sports', 'Concert') NOT NULL,

FOREIGN KEY (venue\_id) REFERENCES Venue (venue\_id)); ****

* **Customer**

**🡪** CREATE TABLE Customer (

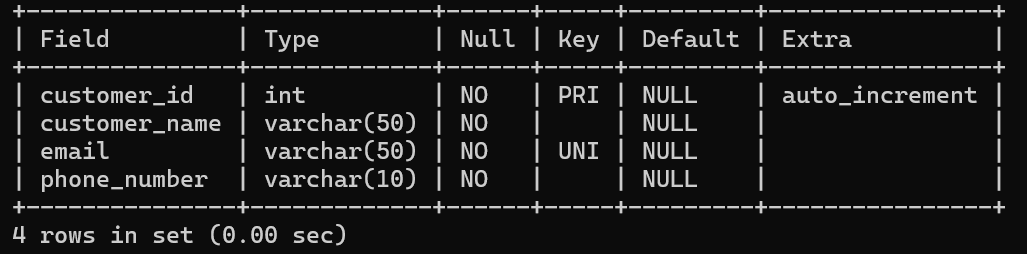
customer\_id INT AUTO\_INCREMENT PRIMARY KEY,

customer\_name VARCHAR (50) NOT NULL,

email VARCHAR (50) UNIQUE NOT NULL,

phone\_number VARCHAR (10) NOT NULL

);

****

* **Booking**

**🡪** CREATE TABLE Booking (

booking\_id INT AUTO\_INCREMENT PRIMARY KEY,

customer\_id INT NOT NULL,

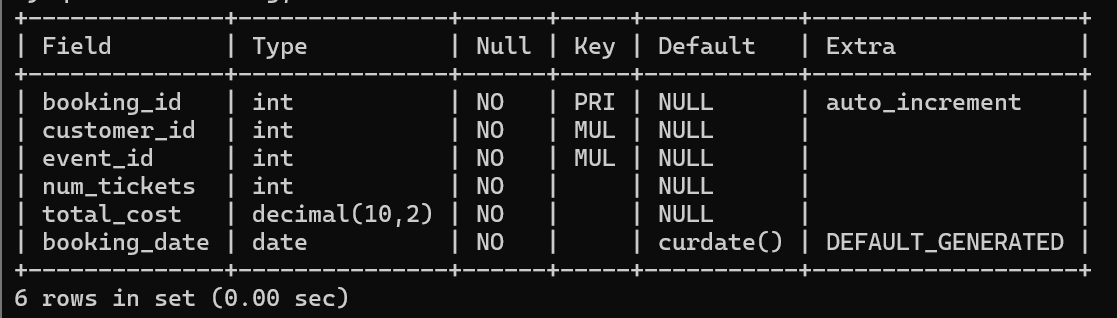
event\_id INT NOT NULL,

num\_tickets INT NOT NULL CHECK (num\_tickets > 0),

total\_cost DECIMAL(10,2) NOT NULL CHECK (total\_cost >= 0), booking\_date DATE NOT NULL DEFAULT (CURDATE ()),

FOREIGN KEY (customer\_id) REFERENCES Customer(customer\_id),

FOREIGN KEY (event\_id) REFERENCES Event(event\_id));

****

**3. Create an ERD (Entity Relationship Diagram) for the database.**

