

Department of Computer Science

**Semester Project**

**Railway Reservation Management System**

**Submitted by:**

Muhammad Hammad (243545)

Muhammad Umair (243587)

Muhammad Faisal (243584)

**Course Instructor:**

Sir Faisal Idrees

Summers 2025

Contents

[Railway Reservation Management System Project Report 3](#_Toc200285052)

[Project Evaluation Criteria 3](#_Toc200285053)

[Abstract 3](#_Toc200285054)

[Introduction 3](#_Toc200285055)

[Problem Statement 3](#_Toc200285056)

[Goal of the Project 4](#_Toc200285057)

[System Design and Implementation 4](#_Toc200285058)

[Entity Relationship Diagram (ERD) 5](#_Toc200285059)

[Relational Model/Relational Schema 6](#_Toc200285060)

[SQL Implementation 7](#_Toc200285061)

[DDL (Creating The DataBase) 7](#_Toc200285062)

[DDL (Creating Tables) 7](#_Toc200285063)

[DML (Insertion) 10](#_Toc200285064)

[Select Queries (Subqueries and Joins ) Procedure Query 13](#_Toc200285065)

[View Query 13](#_Toc200285066)

[Join Query 14](#_Toc200285067)

[Results & Outputs 14](#_Toc200285068)

[Train Table 15](#_Toc200285069)

[Admin Table 15](#_Toc200285070)

[Passenger Table 15](#_Toc200285071)

[Feedback Table 16](#_Toc200285072)

[Station Table 16](#_Toc200285073)

[User Account Table 16](#_Toc200285074)

[Booking Table 17](#_Toc200285075)

[Conclusion 17](#_Toc200285076)

[References 17](#_Toc200285077)

# Railway Reservation Management System Project Report

## Project Evaluation Criteria

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Roll#** | **Name** | **Report (15%)** | |  | | --- | | **Correction (10%)** |  |  | | --- | |  | | |  | | --- | | **Correction (10%)** |  |  | | --- | |  | | |  | | --- | | **Concept Understanding (20%)** |  |  | | --- | |  | | |  | | --- | | **Grip (15%)** |  |  | | --- | |  | | |  | | --- | | **Code (30%)** |  |  | | --- | |  | |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

## Abstract

This project presents a database-driven Railway Reservation Management System designed to handle train schedules, ticket bookings, and user information efficiently. Built using SQL Server, the system automates tasks such as seat availability checks, booking validations, and feedback storage. It ensures data accuracy, prevents double bookings, and supports both passengers and administrators with easy access to relevant information. The project highlights how a relational database can effectively manage the core operations of a modern railway reservation system.

## Introduction

The rapid advancement of technology has significantly transformed the way public transportation systems operate. One of the critical components of any transportation service is an efficient reservation system. This project focuses on developing a Railway Reservation Management System that leverages a structured relational database to manage train operations, ticket bookings, scheduling, passenger details, and feedback. The system is designed to replace traditional manual methods with a digital, automated approach that ensures data accuracy, operational efficiency, and improved user experience for both administrators and passengers.

## Problem Statement

Manual reservation systems are prone to errors, inefficiencies, and data loss, often resulting in overbooked trains, scheduling conflicts, and lack of accountability. There is also no centralized way to track passenger feedback, train performance, or seat availability in real-time. These shortcomings lead to poor customer satisfaction and operational difficulties for railway staff. Hence, there is a pressing need for a centralized, database-driven system that automates and secures the entire reservation process.

## ****Goal of the Project****

The primary goal of this project is to design and implement a **database system** that manages the core functionalities of a railway reservation platform. This includes automating ticket bookings, managing train schedules, preventing double bookings, storing passenger and admin details, and enabling meaningful data retrieval through views and procedures. The system aims to ensure data consistency, enhance reliability, and provide a foundation for potential expansion into a full web or desktop application in the future.

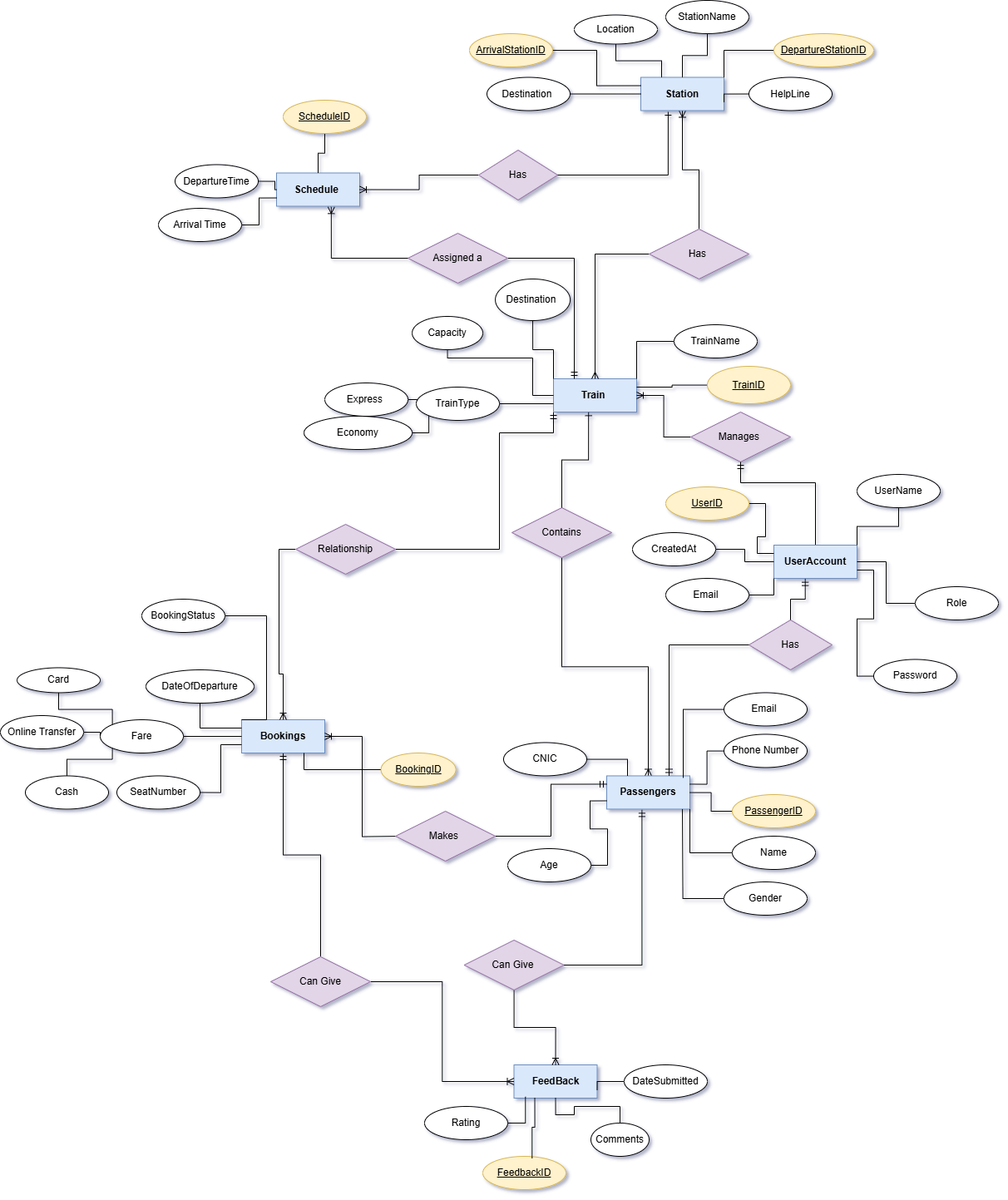
# System Design and Implementation

The system design of the Railway Reservation Management System is based on a structured and normalized relational database model. The **Entity-Relationship** (ER) Diagram outlines the key entities involved in the system, such as Admin, Train, Passenger, Station, Schedules, Bookings, and Feedback along with their relationships and attributes.

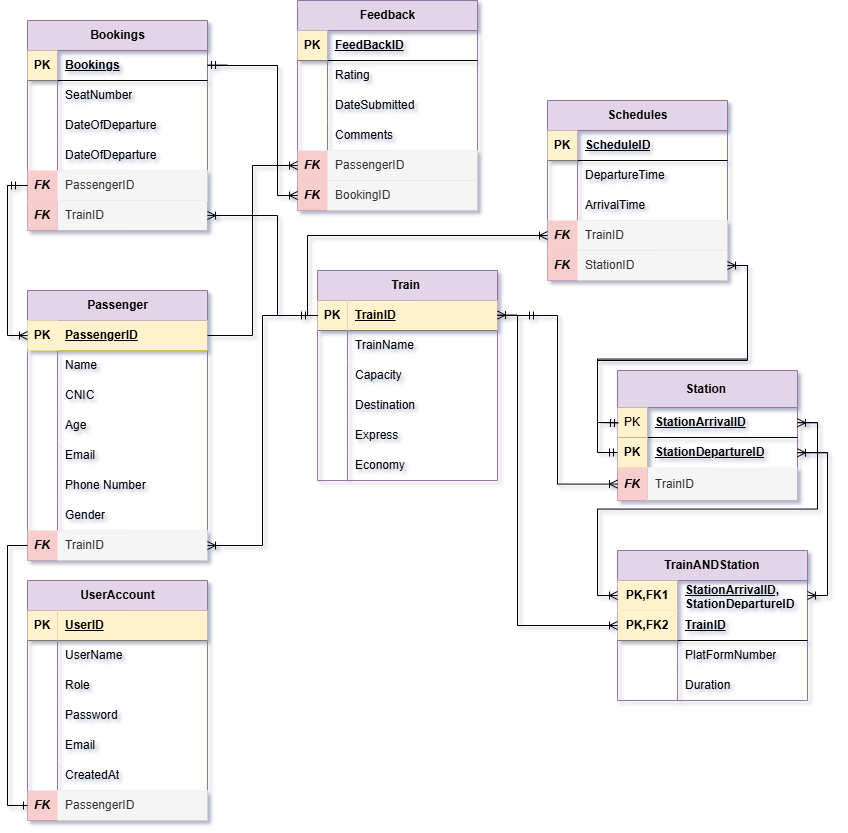
The ER model is then translated into a **Relational Schema**, which defines the tables, primary keys, foreign keys, and constraints necessary to ensure data integrity. The implementation phase involves creating these tables in SQL Server, inserting sample data, and developing views, stored procedures, and triggers to handle key operations such as ticket booking, availability checking, and feedback management.

This design ensures that the system is **modular, scalable, and consistent**, making it capable of handling real-world railway operations with efficiency and accuracy.

## Entity Relationship Diagram (ERD)



## Relational Model/Relational Schema



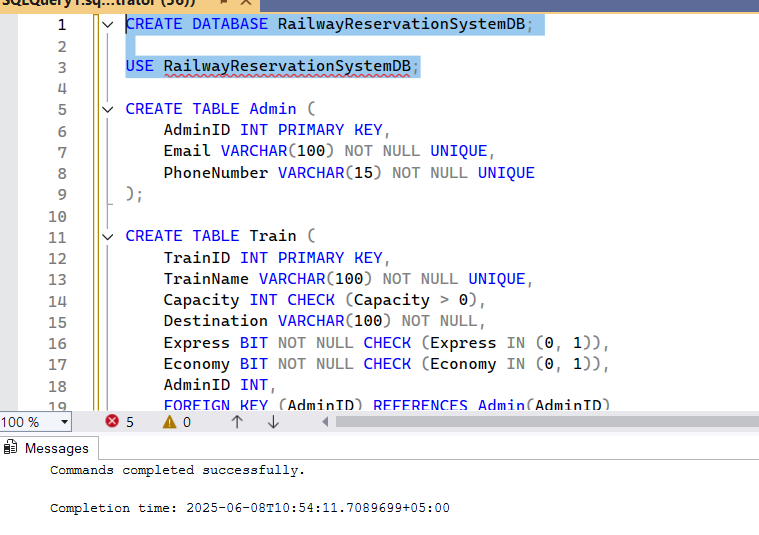
# SQL Implementation

## DDL (Creating The DataBase)

CREATE DATABASE RailwayReservationSystemDB;

USE RailwayReservationSystemDB;

**Screenshot:**



## DDL (Creating Tables)

CREATE TABLE Admin (

AdminID INT PRIMARY KEY,

Email VARCHAR(100) NOT NULL UNIQUE,

PhoneNumber VARCHAR(15) NOT NULL UNIQUE

);

CREATE TABLE Train (

TrainID INT PRIMARY KEY,

TrainName VARCHAR(100) NOT NULL UNIQUE,

Capacity INT CHECK (Capacity > 0),

Destination VARCHAR(100) NOT NULL,

Express BIT NOT NULL CHECK (Express IN (0, 1)),

Economy BIT NOT NULL CHECK (Economy IN (0, 1)),

AdminID INT,

FOREIGN KEY (AdminID) REFERENCES Admin(AdminID)

);

CREATE TABLE Passenger (

PassengerID INT PRIMARY KEY,

Name VARCHAR(100) NOT NULL,

CNIC VARCHAR(15) NOT NULL UNIQUE,

Age INT CHECK (Age > 0),

Email VARCHAR(100) NOT NULL UNIQUE,

PhoneNumber VARCHAR(15) NOT NULL UNIQUE,

Gender VARCHAR(10) CHECK (Gender IN ('Male', 'Female', 'Other')),

TrainID INT,

FOREIGN KEY (TrainID) REFERENCES Train(TrainID)

);

CREATE TABLE Station (

StationArrivalID INT UNIQUE,

StationDepartureID INT,

TrainID INT,

PRIMARY KEY (StationArrivalID, StationDepartureID),

FOREIGN KEY (TrainID) REFERENCES Train(TrainID)

);

CREATE TABLE Schedules (

ScheduleID INT PRIMARY KEY,

DepartureTime DATETIME NOT NULL,

ArrivalTime DATETIME NOT NULL,

TrainID INT,

StationID INT,

FOREIGN KEY (TrainID) REFERENCES Train(TrainID),

FOREIGN KEY (StationID) REFERENCES Station(StationArrivalID)

);

CREATE TABLE TrainANDStation (

StationArrivalID INT,

StationDepartureID INT,

TrainID INT,

PlatFormNumber VARCHAR(10) NOT NULL,

Duration INT CHECK (Duration >= 0),

PRIMARY KEY (StationArrivalID, StationDepartureID, TrainID),

FOREIGN KEY (StationArrivalID, StationDepartureID) REFERENCES Station(StationArrivalID, StationDepartureID),

FOREIGN KEY (TrainID) REFERENCES Train(TrainID)

);

CREATE TABLE Bookings (

BookingID INT PRIMARY KEY,

SeatNumber VARCHAR(10) NOT NULL,

DateOfDeparture DATE NOT NULL,

PassengerID INT,

TrainID INT,

FOREIGN KEY (PassengerID) REFERENCES Passenger(PassengerID),

FOREIGN KEY (TrainID) REFERENCES Train(TrainID)

);

CREATE TABLE Feedback (

FeedbackID INT PRIMARY KEY,

Rating INT CHECK (Rating BETWEEN 1 AND 5),

DateSubmitted DATE DEFAULT GETDATE(),

Comments TEXT,

PassengerID INT,

BookingID INT,

FOREIGN KEY (PassengerID) REFERENCES Passenger(PassengerID),

FOREIGN KEY (BookingID) REFERENCES Bookings(BookingID)

);

CREATE TABLE UserAccount (

UserID INT PRIMARY KEY,

UserName VARCHAR(50) NOT NULL,

Role VARCHAR(20) CHECK (Role IN ('Admin', 'Passenger')),

Password VARCHAR(100) NOT NULL,

Email VARCHAR(100) NOT NULL UNIQUE,

CreatedAt DATETIME DEFAULT GETDATE(),

AdminID INT,

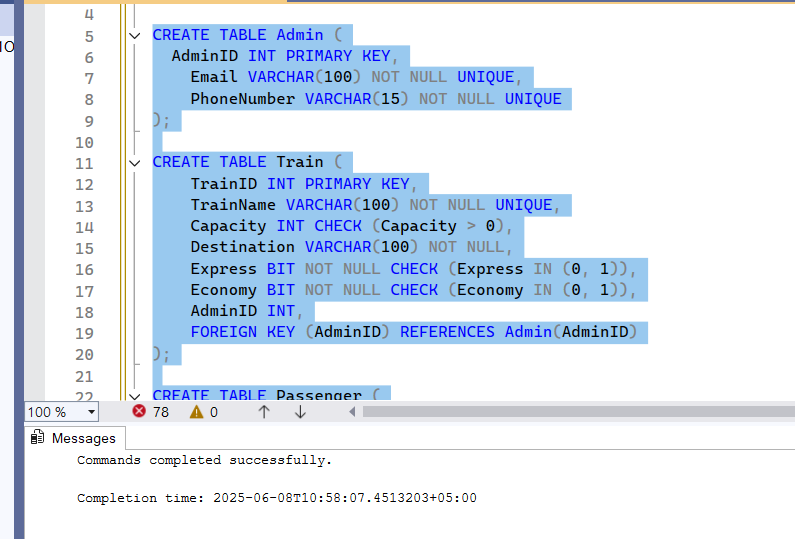
PassengerID INT,

FOREIGN KEY (AdminID) REFERENCES Admin(AdminID),

FOREIGN KEY (PassengerID) REFERENCES Passenger(PassengerID)

);

**Screenshot:**



## DML (Insertion)

INSERT INTO Admin (AdminID, Email, PhoneNumber)

VALUES

(1001, 'admin1@pakrail.gov.pk', '03001234567'),

(1002, 'admin2@pakrail.gov.pk', '03009876543'),

(1003, 'admin3@pakrail.gov.pk', '03111234567'),

(1004, 'admin4@pakrail.gov.pk', '03229876543'),

(1005, 'admin5@pakrail.gov.pk', '03331234567'),

(1006, 'admin6@pakrail.gov.pk', '03449876543'),

(1007, 'admin7@pakrail.gov.pk', '03551234567'),

(1008, 'admin8@pakrail.gov.pk', '03669876543'),

(1009, 'admin9@pakrail.gov.pk', '03771234567'),

(1010, 'admin10@pakrail.gov.pk', '03889876543');

INSERT INTO Train (TrainID, TrainName, Capacity, Destination, Express, Economy, AdminID)

VALUES

(2001, 'Green Line Express', 300, 'Islamabad', 1, 0, 1001),

(2002, 'Jaffar Express', 400, 'Lahore', 0, 1, 1002),

(2003, 'Karakoram Express', 350, 'Karachi', 1, 0, 1003),

(2004, 'Allama Iqbal Express', 450, 'Lahore', 0, 1, 1004),

(2005, 'Awam Express', 500, 'Peshawar', 0, 1, 1005),

(2006, 'Shalimar Express', 320, 'Karachi', 1, 0, 1006),

(2007, 'Tezgam', 280, 'Rawalpindi', 1, 0, 1007),

(2008, 'Bolan Mail', 400, 'Quetta', 0, 1, 1008),

(2009, 'Pak Business Express', 200, 'Islamabad', 1, 0, 1009),

(2010, 'Sir Syed Express', 450, 'Karachi', 0, 1, 1010);

INSERT INTO Passenger (PassengerID, Name, CNIC, Age, Email, PhoneNumber, Gender, TrainID)

VALUES

(3001, 'Ali Khan', '42101-1234567-8', 28, 'ali.khan@email.com', '03121234567', 'Male', 2001),

(3002, 'Sara Ahmed', '35202-9876543-1', 32, 'sara.ahmed@email.com', '03129876543', 'Female', 2002),

(3003, 'Usman Malik', '37405-1234567-9', 45, 'usman.malik@email.com', '03211234567', 'Male', 2003),

(3004, 'Fatima Riaz', '42201-9876543-2', 22, 'fatima.riaz@email.com', '03229876543', 'Female', 2004),

(3005, 'Bilal Akhtar', '36101-1234567-0', 35, 'bilal.akhtar@email.com', '03331234567', 'Male', 2005),

(3006, 'Hina Shah', '38102-9876543-3', 29, 'hina.shah@email.com', '03349876543', 'Female', 2006),

(3007, 'Kamran Siddiqui', '42301-1234567-1', 50, 'kamran.siddiqui@email.com', '03451234567', 'Male', 2007),

(3008, 'Ayesha Khan', '35302-9876543-4', 26, 'ayesha.khan@email.com', '03469876543', 'Female', 2008),

(3009, 'Imran Hashmi', '37501-1234567-2', 40, 'imran.hashmi@email.com', '03571234567', 'Male', 2009),

(3010, 'Nadia Akram', '42402-9876543-5', 33, 'nadia.akram@email.com', '03589876543', 'Female', 2010);

INSERT INTO Station (StationArrivalID, StationDepartureID, TrainID)

VALUES

(101, 102, 2001), -- Karachi -> Hyderabad (Green Line)

(102, 103, 2001), -- Hyderabad -> Islamabad

(104, 105, 2002), -- Quetta -> Sukkur (Jaffar Express)

(105, 106, 2002), -- Sukkur -> Lahore

(107, 108, 2003), -- Peshawar -> Rawalpindi (Karakoram)

(108, 109, 2003), -- Rawalpindi -> Karachi

(110, 111, 2004), -- Faisalabad -> Lahore (Allama Iqbal)

(112, 113, 2005), -- Karachi -> Rohri (Awam)

(113, 114, 2005), -- Rohri -> Peshawar

(115, 116, 2006); -- Lahore -> Karachi (Shalimar)

INSERT INTO Schedules (ScheduleID, DepartureTime, ArrivalTime, TrainID, StationID)

VALUES

(4001, '2025-07-01 08:00:00', '2025-07-01 10:30:00', 2001, 101), -- Green Line: Karachi -> Hyderabad

(4002, '2025-07-01 11:00:00', '2025-07-01 15:00:00', 2001, 102), -- Hyderabad -> Islamabad

(4003, '2025-07-02 07:00:00', '2025-07-02 12:00:00', 2002, 104), -- Jaffar: Quetta -> Sukkur

(4004, '2025-07-02 12:30:00', '2025-07-02 20:00:00', 2002, 105), -- Sukkur -> Lahore

(4005, '2025-07-03 09:00:00', '2025-07-03 11:30:00', 2003, 107), -- Karakoram: Peshawar -> Rawalpindi

(4006, '2025-07-03 12:00:00', '2025-07-03 18:00:00', 2003, 108), -- Rawalpindi -> Karachi

(4007, '2025-07-04 06:00:00', '2025-07-04 09:00:00', 2004, 110), -- Allama Iqbal: Faisalabad -> Lahore

(4008, '2025-07-05 22:00:00', '2025-07-06 04:00:00', 2005, 112), -- Awam: Karachi -> Rohri (overnight)

(4009, '2025-07-06 05:00:00', '2025-07-06 14:00:00', 2005, 113), -- Rohri -> Peshawar

(4010, '2025-07-07 07:30:00', '2025-07-07 10:00:00', 2006, 115); -- Shalimar: Lahore -> Karachi

INSERT INTO TrainANDStation (StationArrivalID, StationDepartureID, TrainID, PlatFormNumber, Duration)

VALUES

(101, 102, 2001, 'Platform 1', 150), -- Karachi-Hyderabad: 2.5 hrs

(102, 103, 2001, 'Platform 2', 240), -- Hyderabad-Islamabad: 4 hrs

(104, 105, 2002, 'Platform 3', 300), -- Quetta-Sukkur: 5 hrs

(105, 106, 2002, 'Platform 1', 450), -- Sukkur-Lahore: 7.5 hrs

(107, 108, 2003, 'Platform 2', 150), -- Peshawar-Rawalpindi: 2.5 hrs

(108, 109, 2003, 'Platform 4', 360), -- Rawalpindi-Karachi: 6 hrs

(110, 111, 2004, 'Platform 3', 180), -- Faisalabad-Lahore: 3 hrs

(112, 113, 2005, 'Platform 1', 360), -- Karachi-Rohri: 6 hrs

(113, 114, 2005, 'Platform 2', 540), -- Rohri-Peshawar: 9 hrs

(115, 116, 2006, 'Platform 5', 150); -- Lahore-Karachi: 2.5 hrs

INSERT INTO Bookings (BookingID, SeatNumber, DateOfDeparture, PassengerID, TrainID)

VALUES

(5001, 'A1', '2025-07-01', 3001, 2001),

(5002, 'B3', '2025-07-02', 3002, 2002),

(5003, 'C2', '2025-07-03', 3003, 2003),

(5004, 'D4', '2025-07-04', 3004, 2004),

(5005, 'A5', '2025-07-05', 3005, 2005),

(5006, 'B1', '2025-07-06', 3006, 2006),

(5007, 'C3', '2025-07-07', 3007, 2007),

(5008, 'D2', '2025-07-08', 3008, 2008),

(5009, 'A4', '2025-07-09', 3009, 2009),

(5010, 'B5', '2025-07-10', 3010, 2010);

INSERT INTO Feedback (FeedbackID, Rating, Comments, PassengerID, BookingID)

VALUES

(6001, 5, 'Excellent service!', 3001, 5001),

(6002, 4, 'Comfortable seats but delayed.', 3002, 5002),

(6003, 5, 'Friendly staff and clean coach.', 3003, 5003),

(6004, 3, 'AC was not cooling properly.', 3004, 5004),

(6005, 5, 'On-time arrival and departure.', 3005, 5005),

(6006, 2, 'Too crowded and noisy.', 3006, 5006),

(6007, 4, 'Good food options available.', 3007, 5007),

(6008, 3, 'Average journey experience.', 3008, 5008),

(6009, 5, 'Will definitely travel again!', 3009, 5009),

(6010, 4, 'Value for money.', 3010, 5010);

INSERT INTO UserAccount (UserID, UserName, Role, Password, Email, AdminID, PassengerID)

VALUES

(7001, 'admin1', 'Admin', 'hashed\_pwd\_1', 'admin1@pakrail.gov.pk', 1001, NULL),

(7002, 'sara.ahmed', 'Passenger', 'hashed\_pwd\_2', 'sara.ahmed@email.com', NULL, 3002),

(7003, 'admin2', 'Admin', 'hashed\_pwd\_3', 'admin2@pakrail.gov.pk', 1002, NULL),

(7004, 'usman.malik', 'Passenger', 'hashed\_pwd\_4', 'usman.malik@email.com', NULL, 3003),

(7005, 'fatima.riaz', 'Passenger', 'hashed\_pwd\_5', 'fatima.riaz@email.com', NULL, 3004),

(7006, 'admin3', 'Admin', 'hashed\_pwd\_6', 'admin3@pakrail.gov.pk', 1003, NULL),

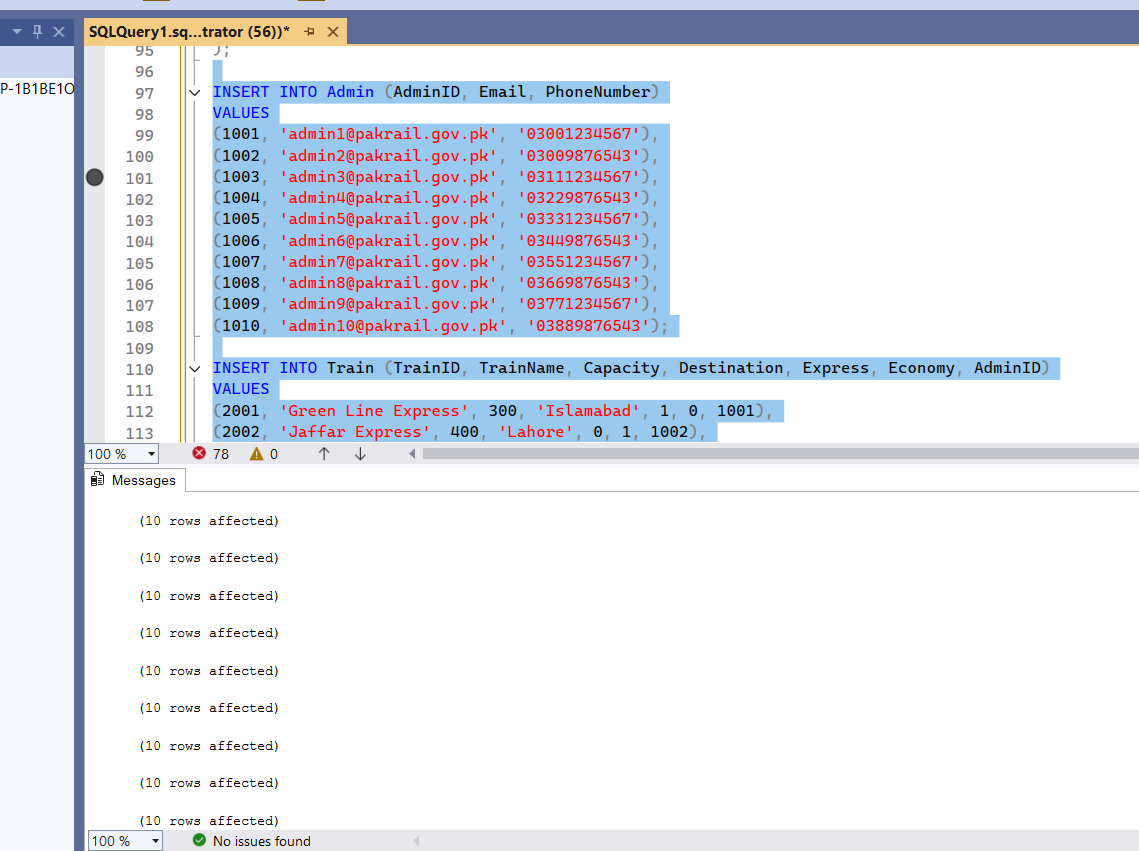
(7007, 'bilal.akhtar', 'Passenger', 'hashed\_pwd\_7', 'bilal.akhtar@email.com', NULL, 3005),

(7008, 'hina.shah', 'Passenger', 'hashed\_pwd\_8', 'hina.shah@email.com', NULL, 3006),

(7009, 'admin4', 'Admin', 'hashed\_pwd\_9', 'admin4@pakrail.gov.pk', 1004, NULL),

(7010, 'kamran.siddiqui', 'Passenger', 'hashed\_pwd\_10', 'kamran.siddiqui@email.com', NULL, 3007);

**Screen Shot:**



# Select Queries (Subqueries and Joins ) Procedure Query

CREATE VIEW PassengerBookings AS

SELECT

p.Name,

p.CNIC,

t.TrainName,

b.SeatNumber,

b.DateOfDeparture,

s.DepartureTime,

s.ArrivalTime

FROM Bookings b

JOIN Passenger p ON b.PassengerID = p.PassengerID

JOIN Train t ON b.TrainID = t.TrainID

JOIN Schedules s ON t.TrainID = s.TrainID;

### View Query

CREATE VIEW TrainSchedules AS

SELECT

t.TrainName,

sta.StationArrivalID AS FromStation,

std.StationDepartureID AS ToStation,

s.DepartureTime,

s.ArrivalTime,

DATEDIFF(MINUTE, s.DepartureTime, s.ArrivalTime) AS DurationMinutes

FROM Schedules s

JOIN Train t ON s.TrainID = t.TrainID

JOIN Station sta ON s.StationID = sta.StationArrivalID

JOIN Station std ON s.StationID = std.StationDepartureID;

### Join Query

SELECT

p.Name,

p.CNIC,

t.TrainName,

t.Destination,

b.SeatNumber,

b.DateOfDeparture,

s.DepartureTime,

s.ArrivalTime,

f.Rating,

f.Comments

FROM Passenger p

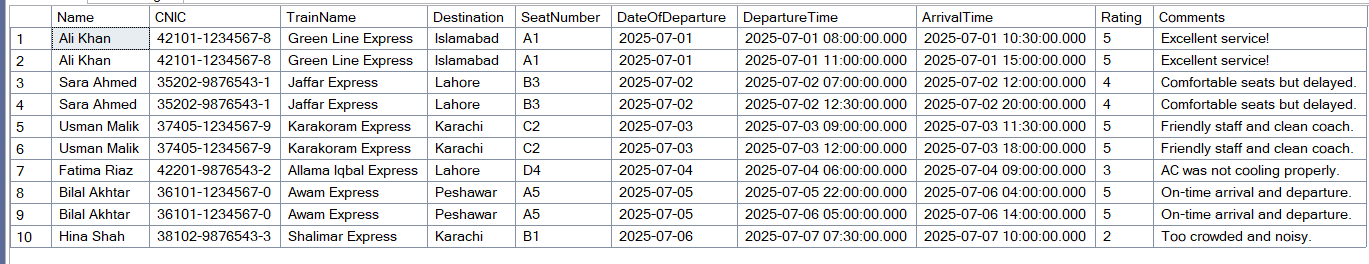
JOIN Bookings b ON p.PassengerID = b.PassengerID

JOIN Train t ON b.TrainID = t.TrainID

JOIN Schedules s ON t.TrainID = s.TrainID

LEFT JOIN Feedback f ON p.PassengerID = f.PassengerID AND b.BookingID = f.BookingID;

**Screenshot**:



## Query Optimization Techniques

**Normalization**: Ensuring the database is normalized to eliminate redundancy and maintain data integrity.

**Efficient Joins**: Using appropriate join types (INNER, LEFT, RIGHT, FULL) based on the query requirements.

# Results & Outputs

select \* From Train;

select \* From Feedback;

select \* From Schedules;

select \* From Passenger;

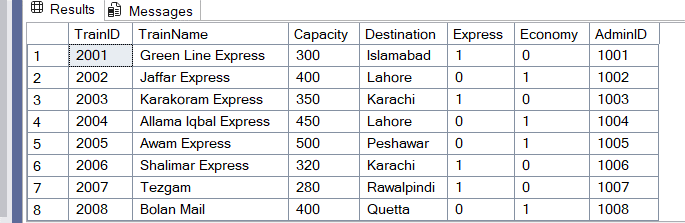
select \* From Station;

select \* From UserAccount;

select \* From Bookings;

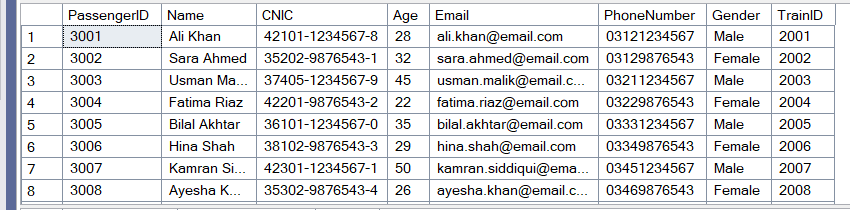
**Screenshots:**

## Train Table

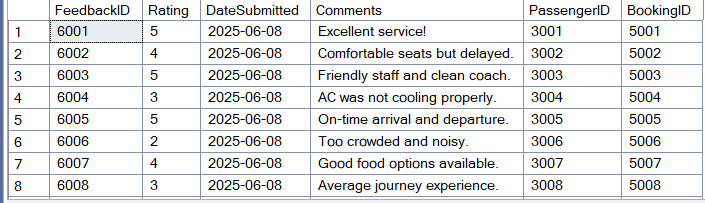


## Admin Tble

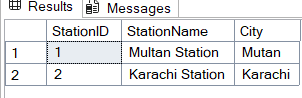
## Passenger Table



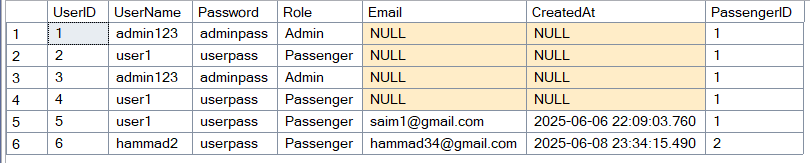
## Feedback Table



## Station Table

****

## User Account Table

****

## Booking Table

# Conclusion

The Railway Reservation Management System successfully streamlines the process of booking, managing, and tracking train journeys. By integrating multiple modules, such as Train Management, Passenger Records, Schedule Handling, Booking System, Feedback Collection, and Admin Controls, the system ensures efficient operations, reduces manual errors, and improves the overall passenger experience.

With its user-friendly interface and structured database management using SQL Server, the project not only automates core functionalities but also lays the groundwork for future enhancements like online payments, seat selection, and real-time train tracking. This system reflects how technology can modernize traditional railway systems, making them faster, more reliable, and more accessible.

# References

1. Microsoft. (n.d.). *SQL Server Management Studio (SSMS)*. Retrieved from <https://learn.microsoft.com/en-us/sql/ssms>
2. W3Schools. (n.d.). *SQL Tutorial*. Retrieved from <https://www.w3schools.com/sql/>
3. TutorialsPoint. (n.d.). *DBMS Tutorial*. Retrieved from <https://www.tutorialspoint.com/dbms/index.htm>
4. GeeksforGeeks. (n.d.). *Database Management Systems (DBMS)*. Retrieved from <https://www.geeksforgeeks.org/dbms/>
5. Date, C. J. (2004). *An Introduction to Database Systems* (8th ed.). Pearson Education.