# Railway Reservation System - README

# Railway Reservation System (ADO.NET C#)

## Overview
This project is a console-based Railway Reservation System built in C# using ADO.NET for database operations.
It supports both Admin and User functionalities, with ticket booking, cancellation, refund, and availability tracking.

## Features

### Admin
- View all bookings and reports.
- Delete bookings and process refunds.
- Add/modify trains and class seat availability.

### User
- View trains and available seats.
- Book tickets by selecting train number, class type (Sleeper, 3rd AC, 2nd AC), and number of passengers.
- View current booking status using PNR.
- Get a detailed PDF ticket with all booking information.

## Database Structure

- **Admins**: Stores admin login credentials.
- **Customers**: Stores customer details.
- **TrainMaster**: Stores train details.
- **TrainClasses**: Stores class types for each train with available seats and prices (composite key: TrainNumber, ClassType).
- **Reservations**: Stores booking details including number of passengers and total amount.
- **Cancellations**: Stores cancellation records.
- **Refunds**: Stores refund records.

## How to Run

1. **Setup the database**:
   - Open SQL Server Management Studio.
   - Run the `SQL/seed_full.sql` script to create and populate the database.

2. **Configure the connection**:
   - Open `DatabaseConnection.cs`.
   - Set your SQL Server connection string.

3. **Build the project**:
   - Open the solution in Visual Studio.
   - Build the solution (`Ctrl+Shift+B`).

4. **Run the application**:
   - Execute the program.
   - Choose Admin or User mode.

- Follow on-screen instructions.

## Booking Flow

1. User logs in or registers.
2. User selects train by number and enters class type (`sl`, `3a`, `2a`).
3. User specifies travel date and number of passengers.
4. System checks seat availability and calculates total amount.
5. On confirmation, the booking is saved, seats reduced, and PDF ticket generated.

## Exception Handling

- Custom exceptions (`InvalidInputException`, `BookingNotFoundException`, `DataAccessException`) ensure robust error handling.
- System exceptions like `SqlException` and `IOException` are caught and handled gracefully.

## Example Ticket Details

- PNR (Booking ID)
- Passenger name
- Train name & number
- Source & destination
- Travel date
- Class type
- Number of passengers
- Total amount
- Booking date