A picture containing calendar

Description automatically generated

**AMERICAN INTERNATIONAL UNIVERSITY–BANGLADESH (AIUB)**

**Dept. of Computer Science**

**Faculty of Science and Technology**

**CSC 2108: Introduction To Database**

**Spring 2022-2023**

**Section: H**

**Group:10**

**Project On**

**Metro Rail Management System**

**Supervised By**

**KAWSER IROM RUSHEE**

**Submitted By:**

|  |  |
| --- | --- |
| **Name** | **ID** |
| **1. Abdullah Adnan Abul Kalam** | **22-47846-2** |
| **2. Md Sakib Hasan** | **22-47142-1** |
| **3. Md Arifur Rahman** | **22-47110-1** |
| **4. Abdullah Al Taieb** | **22-48028-2** |

**Content List:**

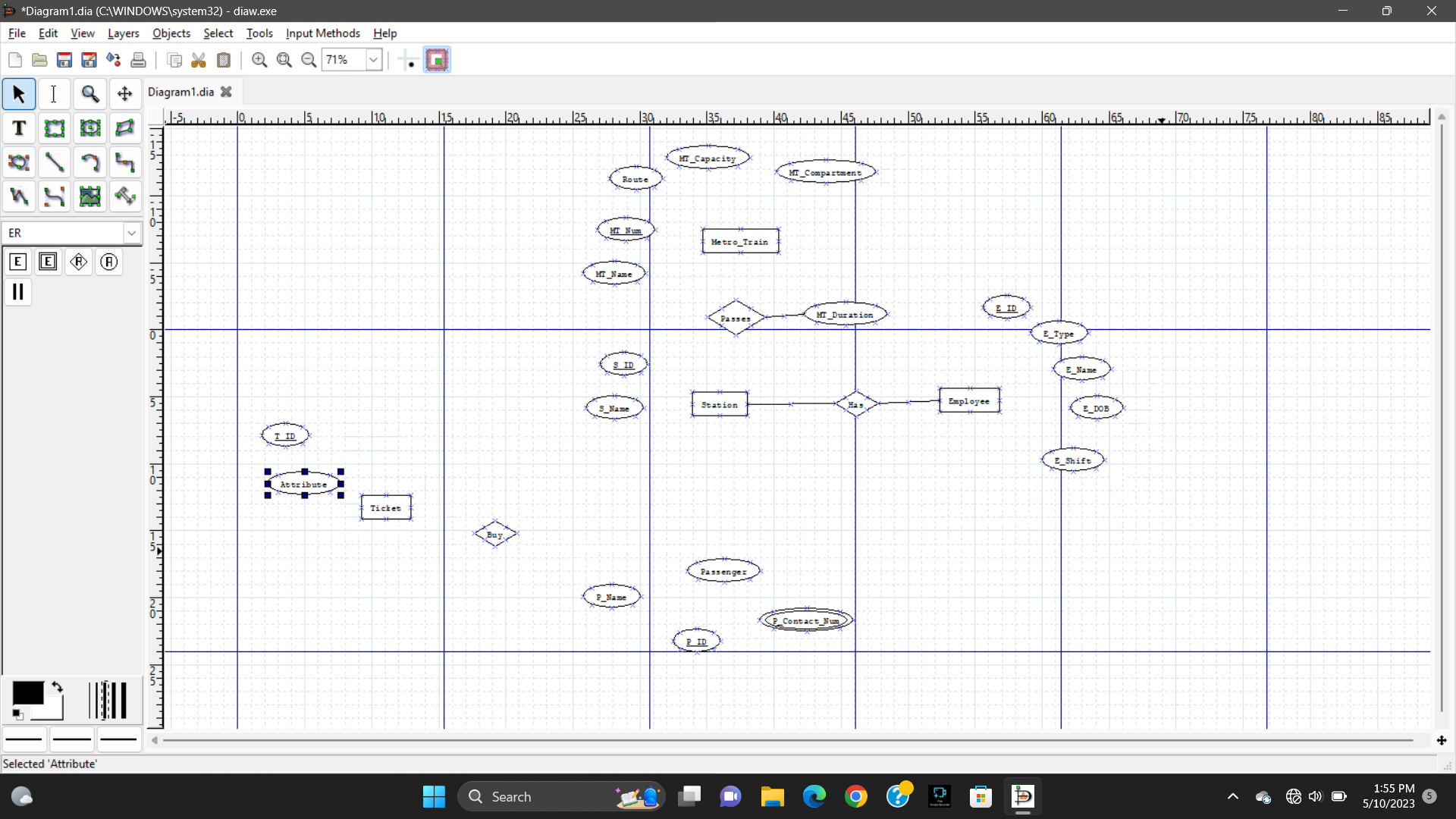
1. Case Scenario
2. ER Diagram
3. Normalization
4. Table Creation
5. Data Insertion
6. Query Writing
7. Conclusion

**Case Scenario:**

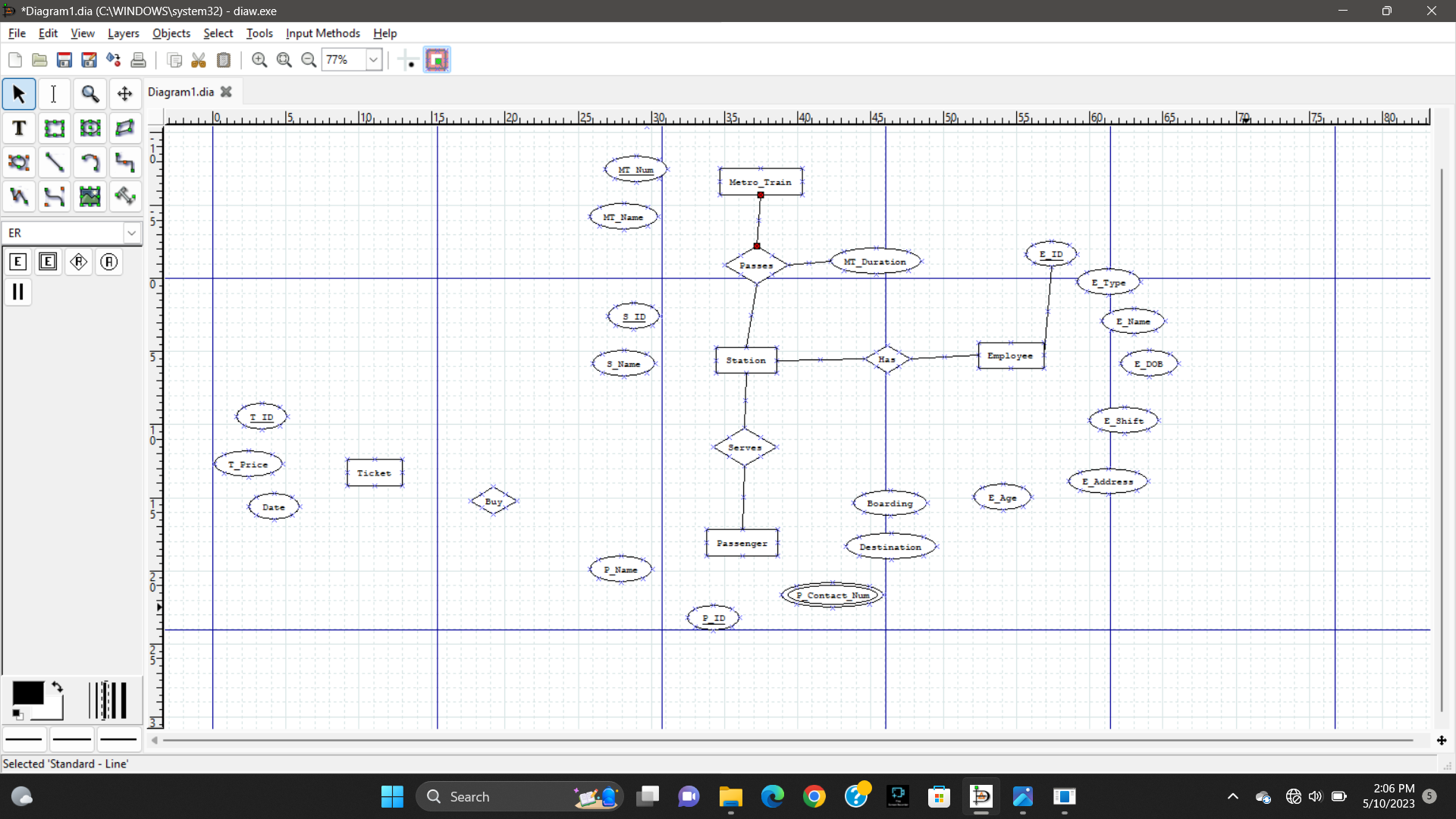
The main entities of this metro rail management system are MetroTrain, Station, Employee,Passenger and Ticket. Each station has unique Id and also they have naame. Metro Train passes station and each train has name,number, capacity, Route, and Compartment.While passing the station duration is stored. Many train passes many station. Station serves many passenger. Each passenger has name , unique Id , Contact number , Destination and Boarding place. One Passenger can have multiple contact number. Station has employees. Each Employee has unique Id . They have Name , Age , Date of Birth, their working type and ContactNumber . One employee can have multiple contact number. Employee also has also specific Address. Inside that Adress there is street number , street name , city and country. Every passenger buys ticket.And one ticket can be bought by only one passenger . One passenger can buy multiple ticket. Each Ticket has their unique Id. There is also the price of the ticket and the date when the ticket was purchased.

**ER-Diagram:**

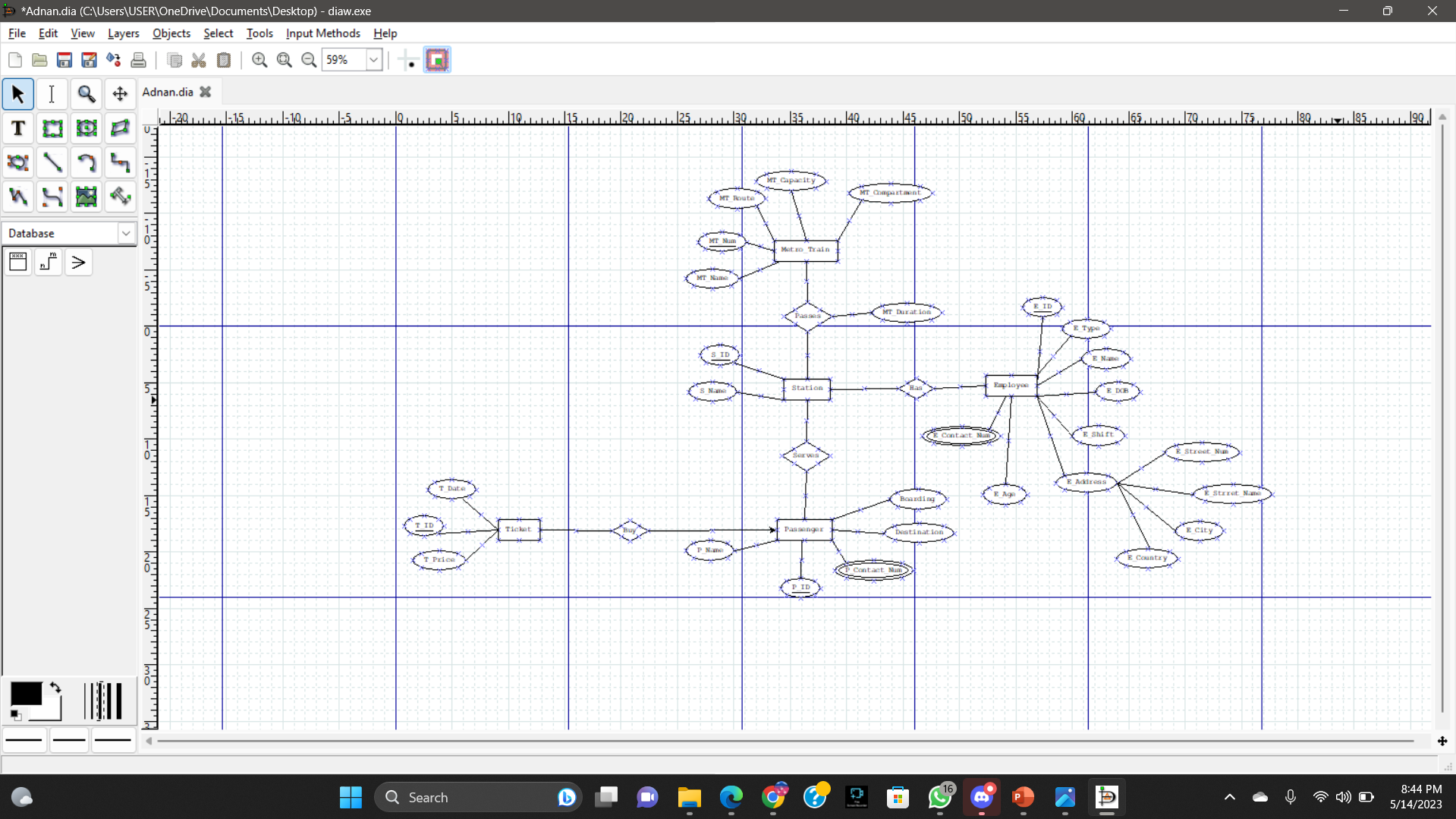
Screenshot of ER diagram just after beginning-



Screenshot of ER diagram in the middle:



Final ER diagram:



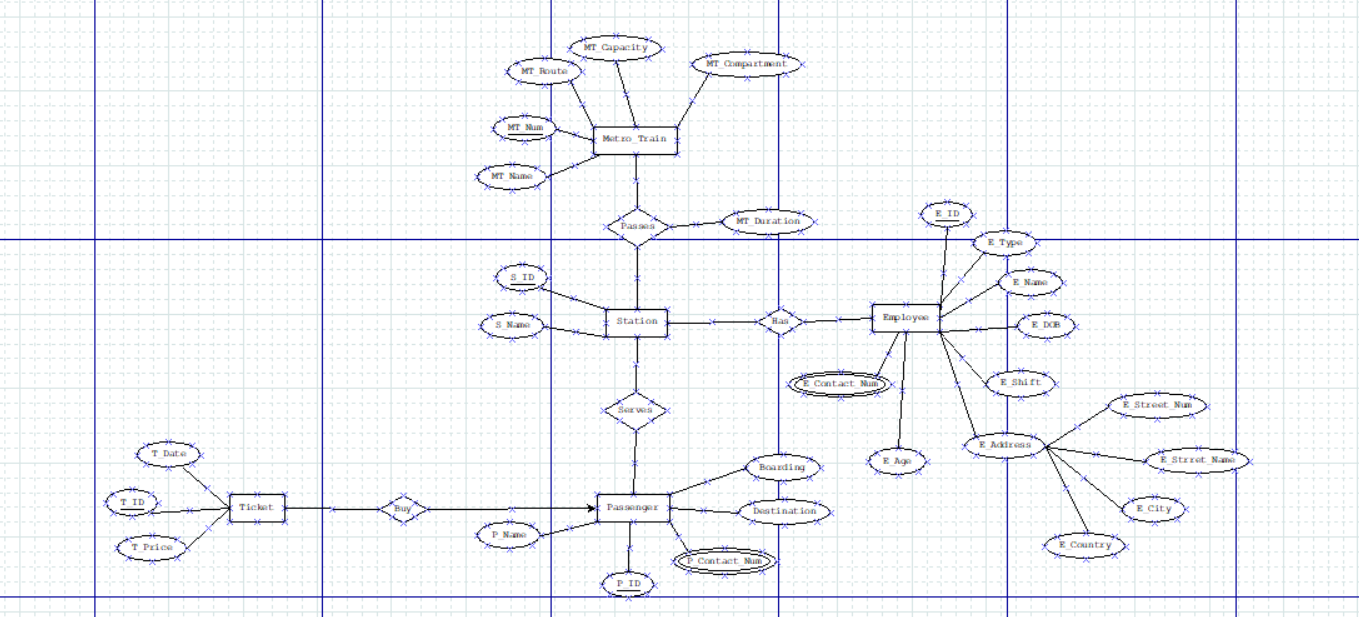


Figure: ER-Diagram

Normalization:

**Passes**

**Metro Train Passes Station:**

**UNF**

Passes (MT\_Name, MT\_Num, MT\_Route, MT\_Capacity,MT\_Compartment,MT\_Duration, S\_ID, S\_Name)

**1NF:**

There is no multi valued attribute, It is already 1NF.

1. MT\_Name, MT\_Num, MT\_Route, MT\_Capacity,MT\_Compartment, MT\_Duration, S\_ID, S\_Name

**2NF:**

1. MT\_Name, MT\_Num, MT\_Route, MT\_Capacity,MT\_Compartment, MT\_Duration,

2. S\_ID, S\_Name

**3NF:**

There is no transitive relationship, It is already 3NF.

1. MT\_Name, MT\_Num, MT\_Route, MT\_Capacity,MT\_Compartment, MT\_Duration,

2. S\_ID, S\_Name

**TABLE CREATION:**

1. MT\_Name, MT\_Num, MT\_Route, MT\_Capacity,MT\_Compartment, MT\_Duration,

2. S\_ID, S\_Name

3. **MT\_Num**, **S\_ID**

**Serves**

**Station Serves Passenger:**

**UNF:**

S\_ID, S\_Name, P\_ID, P\_Name,P\_Contact\_Num,Destination,Boarding

**1NF:**

P­\_Contact\_Num is multivalued attribute

1. S\_ID, S\_Name, P\_ID, P\_Name, P\_Contact\_Num, Destination, Boarding

**2NF:**

1. S\_ID, S\_Name

2. P\_ID, P\_Name, P\_Contact\_Num, Destination, Boarding

**3NF:**

There is no transitive relationship, It is already 3NF.

1. S\_ID, S\_Name

2. P\_ID, P\_Name, P\_Contact\_Num, Destination, Boarding

**TABLE CREATION:**

1. S\_ID, S\_Name

2. P\_ID, P\_Name, P\_Contact\_Num, Destination, Boarding

3. **S\_ID**, **P\_ID**

**Has**

**Station Has Employee:**

**UNF:**

Has (S\_ID, S\_Name, E\_ID, E\_Type, E\_Name, E\_DOB, E\_Shift, E\_Street\_Num, E\_Street\_Name, E\_City, E\_Country, E\_Age, E\_Contact\_Num)

**1NF:**

E\_Contact\_Num is multivalued attribute.

1. S\_ID, S\_Name, E\_ID, E\_Type, E\_Name, E\_DOB, E\_Shift, E\_Street\_Num, E\_Street\_Name, E\_City, E\_Country, E\_Age, E\_Contact\_Num

**2NF:**

1. S\_ID, S\_Name

2. E\_ID, E\_Type, E\_Name, E\_DOB, E\_Shift, E\_Street\_Num, E\_Street\_Name, E\_City, E\_Country, E\_Age, E\_Contact\_Num

**3NF:**

1. S\_ID, S\_Name

2. E\_ID, E\_Type, E\_Name, E\_DOB, E\_Shift, E\_Age, E\_Contact\_Num

3. E\_Street\_Num, E\_Street\_Name, E\_City, E\_Country

**TABLE CREATION:**

1. S\_ID, S\_Name

2. E\_ID, E\_Type, E\_Name, E\_DOB, E\_Shift, E\_Age, E\_Contact\_Num, E\_Address\_ID

3. E\_Address\_ID, E\_Street\_Num, E\_Street\_Name, E\_City, E\_Country

4. **S\_ID**, **E\_ID**

**BUY**

**Passenger Buy Ticket:**

**UNF:**

BUY (P\_ID, P\_Name, P\_Contact\_Num, Destination, Boarding, T\_ID, T\_Price, T\_Date)

**1NF:**

P\_Contact\_Num is multivalued attribute.

1. P\_ID, P\_Name, P\_Contact\_Num, Destination, Boarding, T\_ID, T\_Price, T\_Date

**2NF:**

1. P\_ID, P\_Name, P\_Contact\_Num, Destination, Boarding

2. T\_ID, T\_Price, T\_Date

**3NF:**

There is no transitive relationship, It is already 3NF.

1. P\_ID, P\_Name, P\_Contact\_Num, Destination, Boarding

2. T\_ID, T\_Price, T\_Date

**TABLE CREATION:**

1. P\_ID, P\_Name, P\_Contact\_Num, Destination, Boarding

2. T\_ID, T\_Price, T\_Date, P\_ID

**Temporary Table:**

1. MT\_Name, MT\_Num, MT\_Route, MT\_Capacity,MT\_Compartment, MT\_Duration,

2. S\_ID, S\_Name

3. **MT\_Num**, **S\_ID**

~~3. S\_ID, S\_Name~~

4. P\_ID, P\_Name, P\_Contact\_Num, Destination, Boarding

**5. S\_ID**, **P\_ID**

~~5. S\_ID, S\_Name~~

6. E\_ID, E\_Type, E\_Name, E\_DOB, E\_Shift, E\_Contact\_Num, **E\_Address\_ID**

7. E\_Address\_ID, E\_Street\_Num, E\_Street\_Name, E\_City, E\_Country

4. **S\_ID**, **E\_ID**

~~8. P\_ID, P\_Name, P\_Contact\_Num, Destination, Boarding~~

9. 2. T\_ID, T\_Price, T\_Date, **P\_ID**

**Final Table:**

1. MT\_Num , MT\_Name, MT\_Route, MT\_Capacity,MT\_Compartment, MT\_Duration

2. S\_ID, S\_Name

3. **MT\_NUM**, **S\_ID**

4. P\_ID, P\_Name, P\_Contact\_Num1, P\_Contact\_Num2, Destination, Boarding

**5. S\_ID**, **P\_ID**

6. E\_ID, E\_Type, E\_Name, E\_DOB, E\_Shift, E\_Contact\_Num1, E\_Contact\_Num2, **E\_Address\_ID**

7. E\_Address\_ID, E\_Street\_Num, E\_Street\_Name, E\_City, E\_Country

8. **S\_ID**, **E\_ID**

9. T\_ID, T\_Price, T\_Date, P\_ID

**TABLE CREATION QUERY:**

**Table-01: Metro\_Info**

CREATE TABLE Metro\_Info

( MT\_Num int Primary Key,

MT\_Name VARCHAR2(20) NOT NULL,

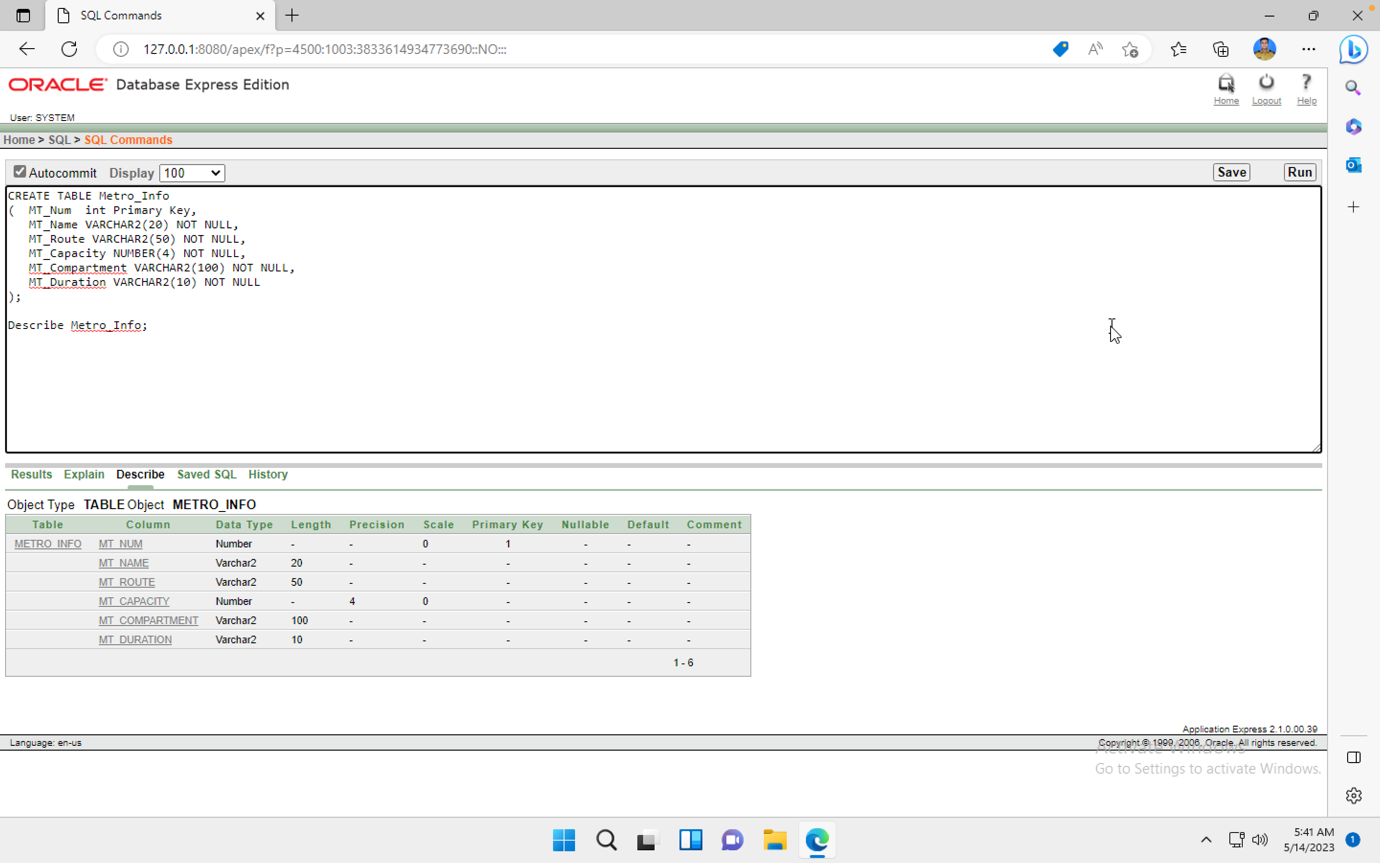
MT\_Route VARCHAR2(50) NOT NULL,

MT\_Capacity NUMBER(4) NOT NULL,

MT\_Compartment VARCHAR2(100) NOT NULL,

MT\_Duration VARCHAR2(10) NOT NULL

);

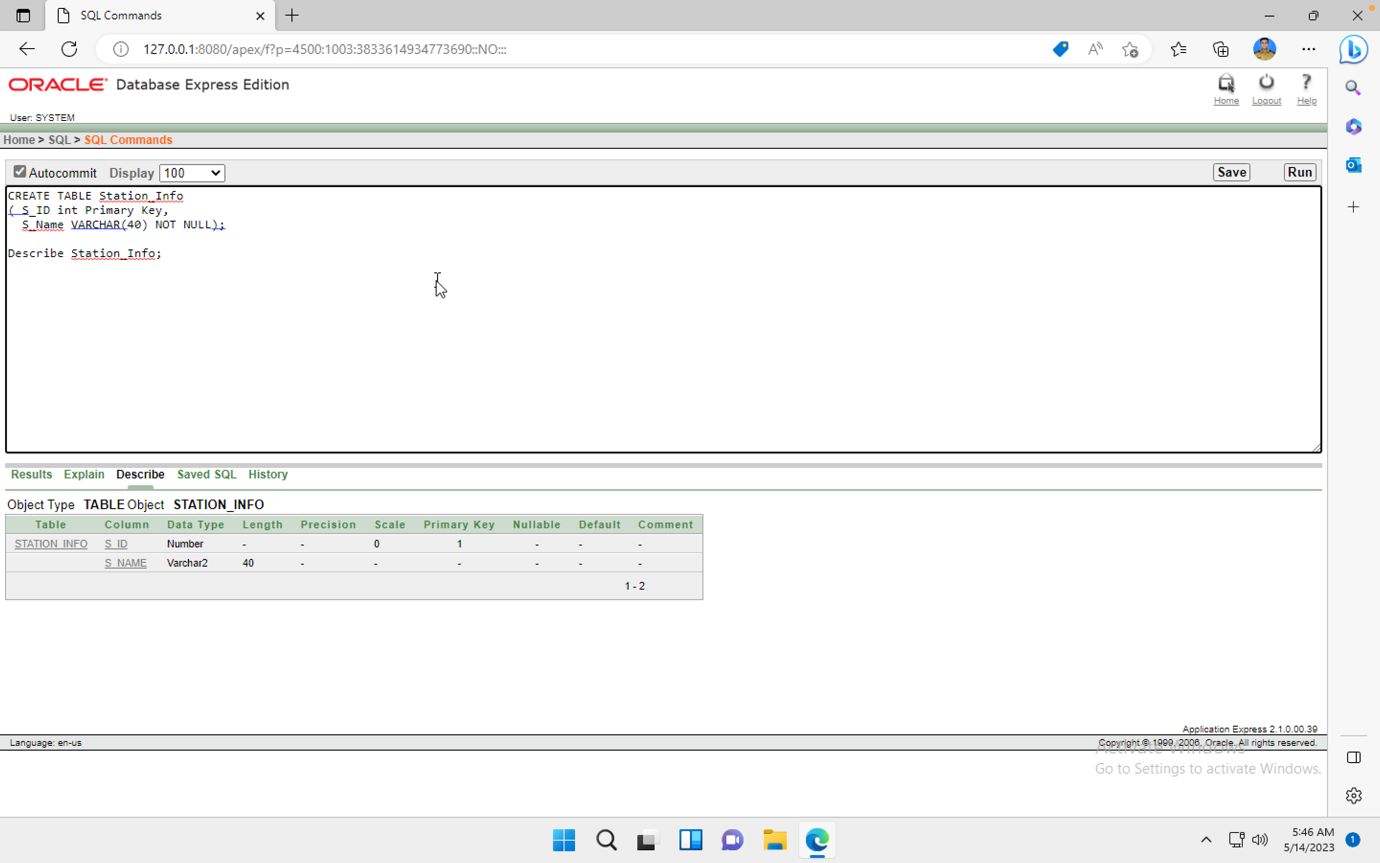


**Table-02: Station\_Info**

CREATE TABLE Station\_Info

( S\_ID int Primary Key,

S\_Name VARCHAR(40) NOT NULL);



**Table-03: Metro\_Station**

CREATE TABLE Metro\_Station

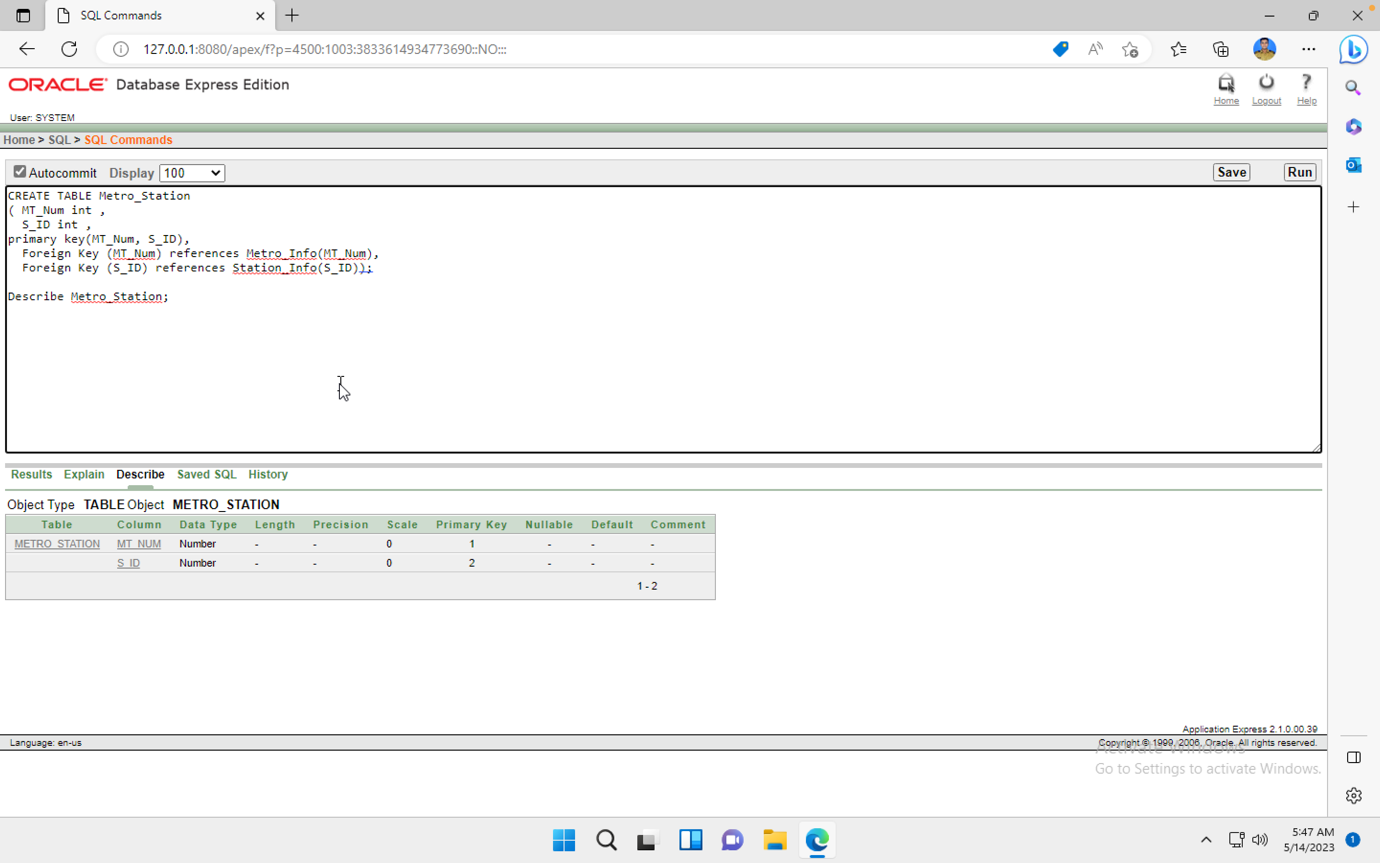
( MT\_Num int ,

S\_ID int ,

primary key(MT\_Num, S\_ID),

Foreign Key (MT\_Num) references Metro\_Info(MT\_Num),

Foreign Key (S\_ID) references Station\_Info(S\_ID));

****

**Table-04: Passenger\_Info**

CREATE TABLE Passenger\_Info

( P\_ID int Primary Key,

P\_Name VARCHAR2(50) NOT NULL,

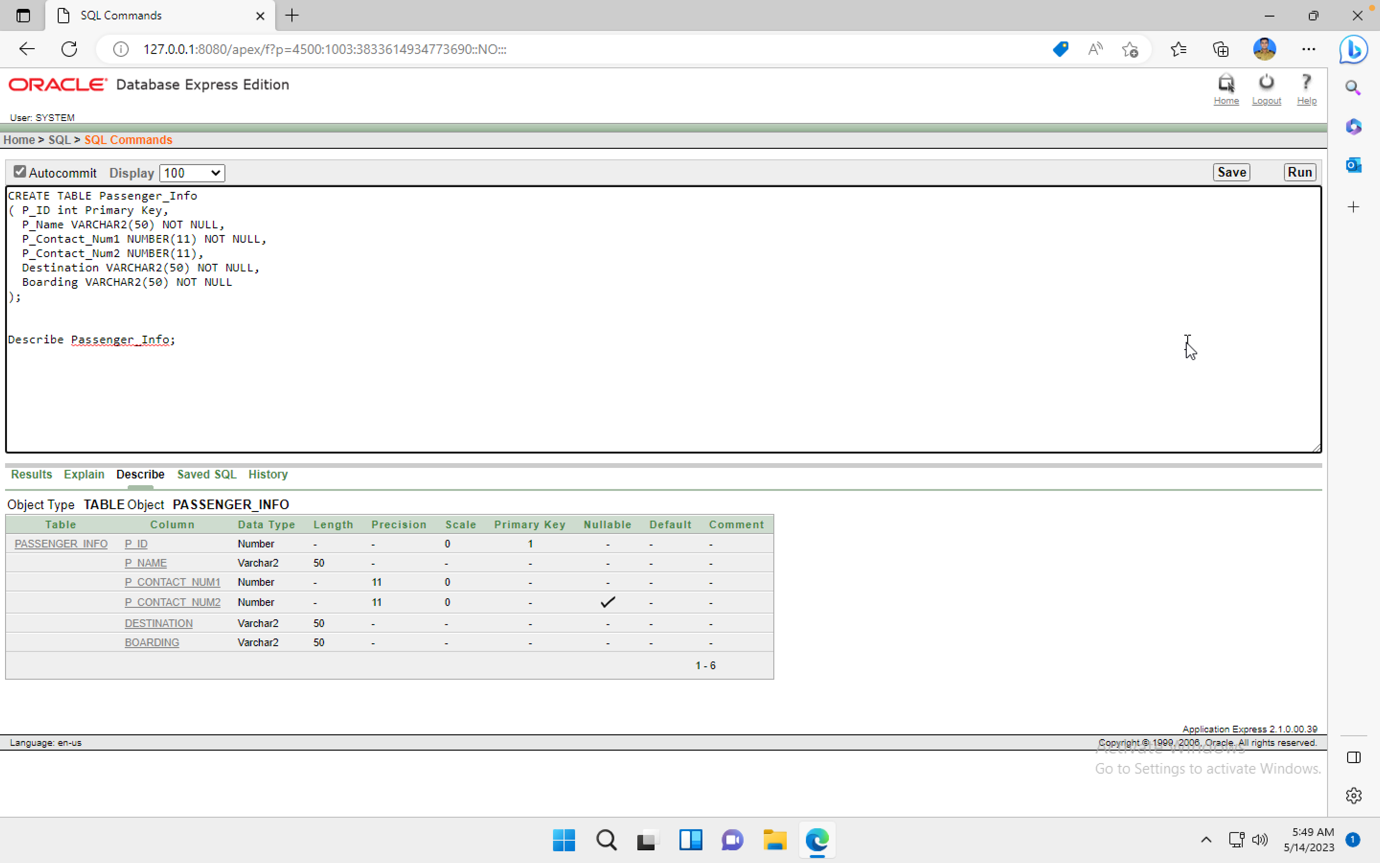
P\_Contact\_Num1 NUMBER(11) NOT NULL,

P\_Contact\_Num2 NUMBER(11),

Destination VARCHAR2(50) NOT NULL,

Boarding VARCHAR2(50) NOT NULL

);

****

**Table-05: Station\_Passenger**

CREATE TABLE Station\_Passenger

( S\_ID int,

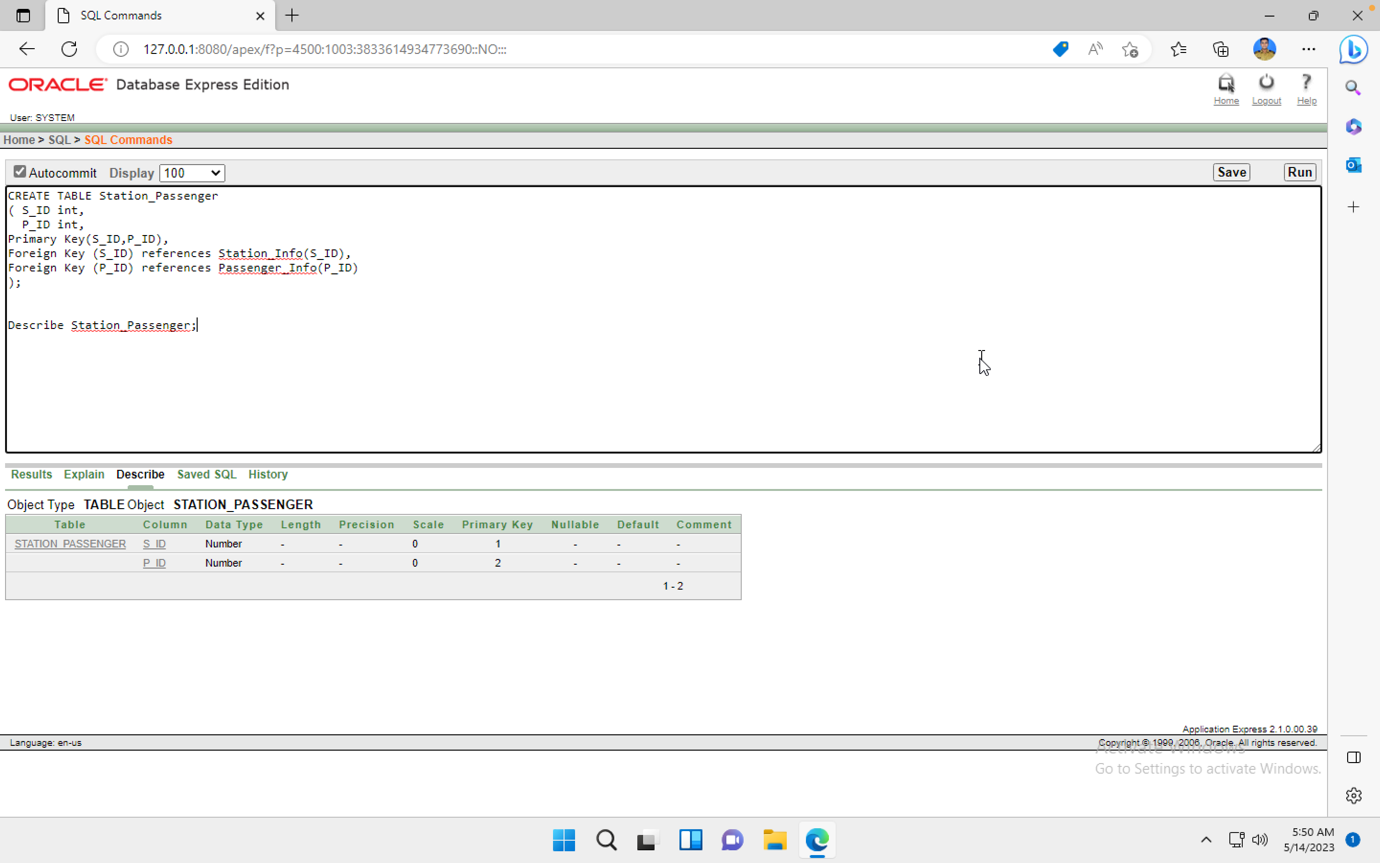
P\_ID int,

Primary Key(S\_ID,P\_ID),

Foreign Key (S\_ID) references Station\_Info(S\_ID),

Foreign Key (P\_ID) references Passenger\_Info(P\_ID)

);

****

**Table-06: Employee\_Address**

CREATE TABLE Employee\_Address

(

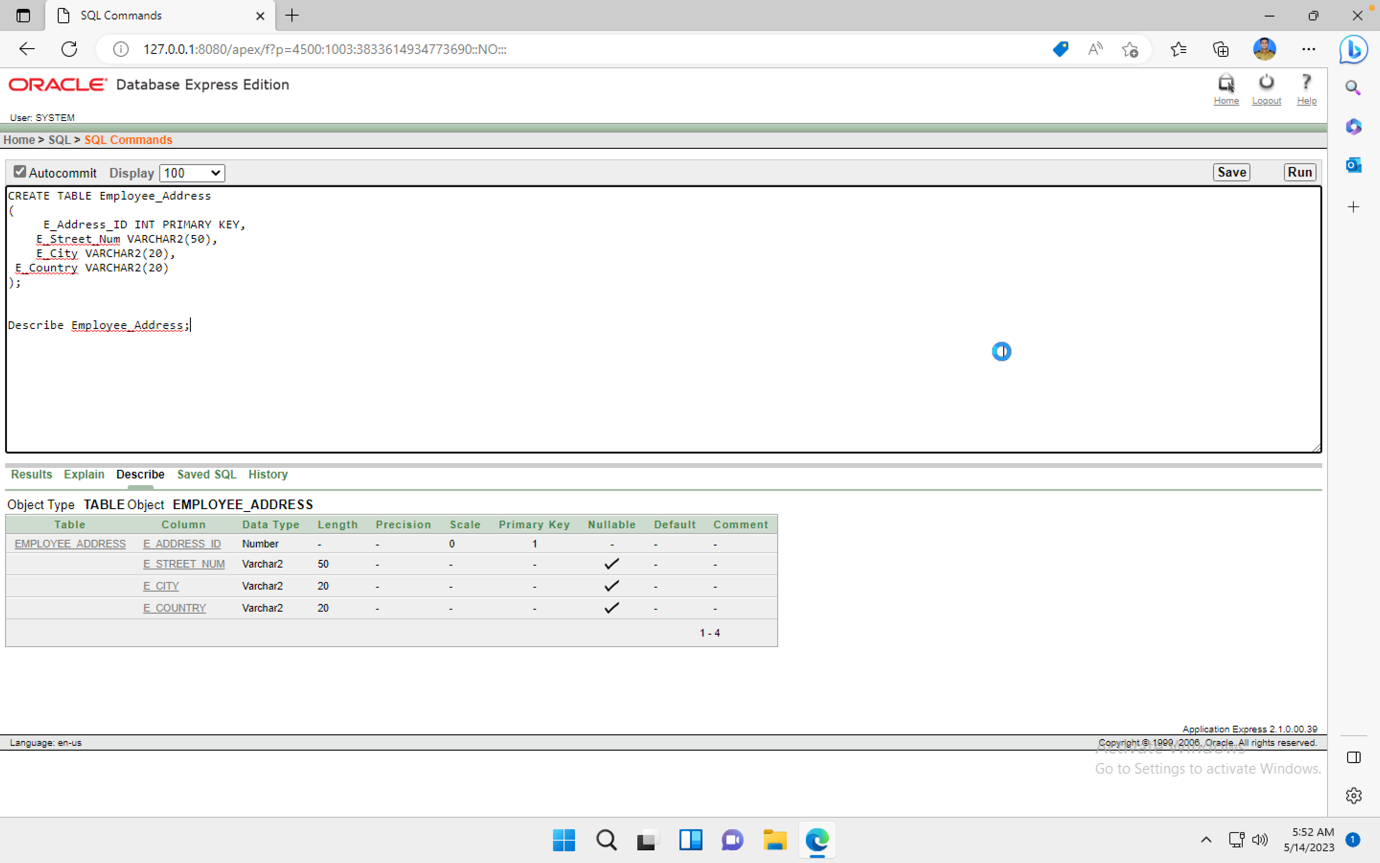
E\_Address\_ID INT PRIMARY KEY,

E\_Street\_Num VARCHAR2(50),

E\_City VARCHAR2(20),

E\_Country VARCHAR2(20)

);

****

**Table-07: Employee\_Info**

CREATE TABLE Employee\_Info

(

E\_ID INT PRIMARY KEY,

E\_Type VARCHAR2(15) CHECK(E\_Type IN('Station Manager', 'Ticketing Staff', 'Security Personnel', 'Train Operators', 'Maintenance Staff', 'Cleaning Staff')) NOT NULL,

E\_Name VARCHAR2(50) NOT NULL,

E\_DOB DATE,

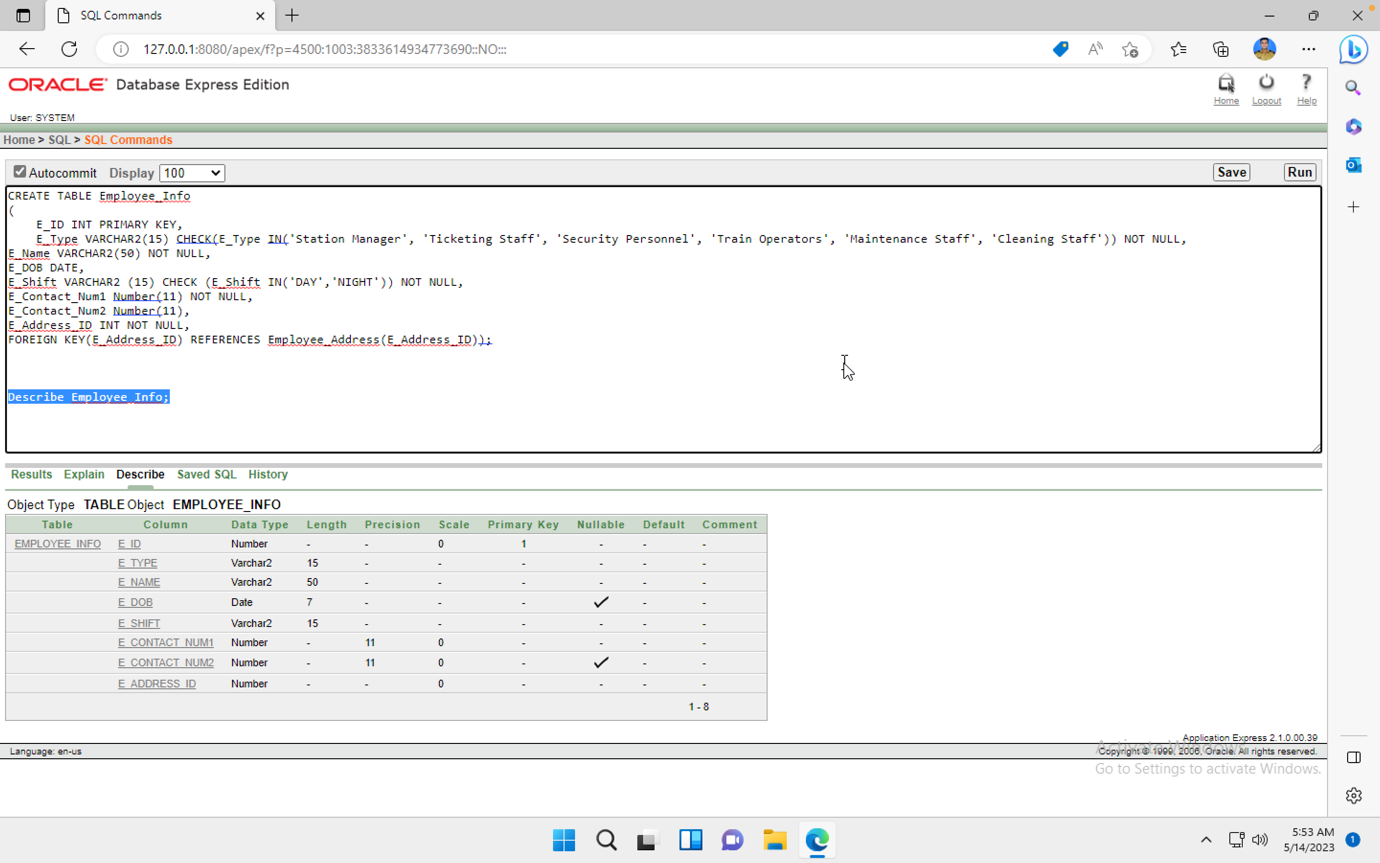
E\_Shift VARCHAR2 (15) CHECK (E\_Shift IN('DAY','NIGHT')) NOT NULL,

E\_Contact\_Num1 Number(11) NOT NULL,

E\_Contact\_Num2 Number(11),

E\_Address\_ID INT NOT NULL,

FOREIGN KEY(E\_Address\_ID) REFERENCES Employee\_Address(E\_Address\_ID));

****

**Table-08: Station\_Employee**

CREATE TABLE Station\_Employee

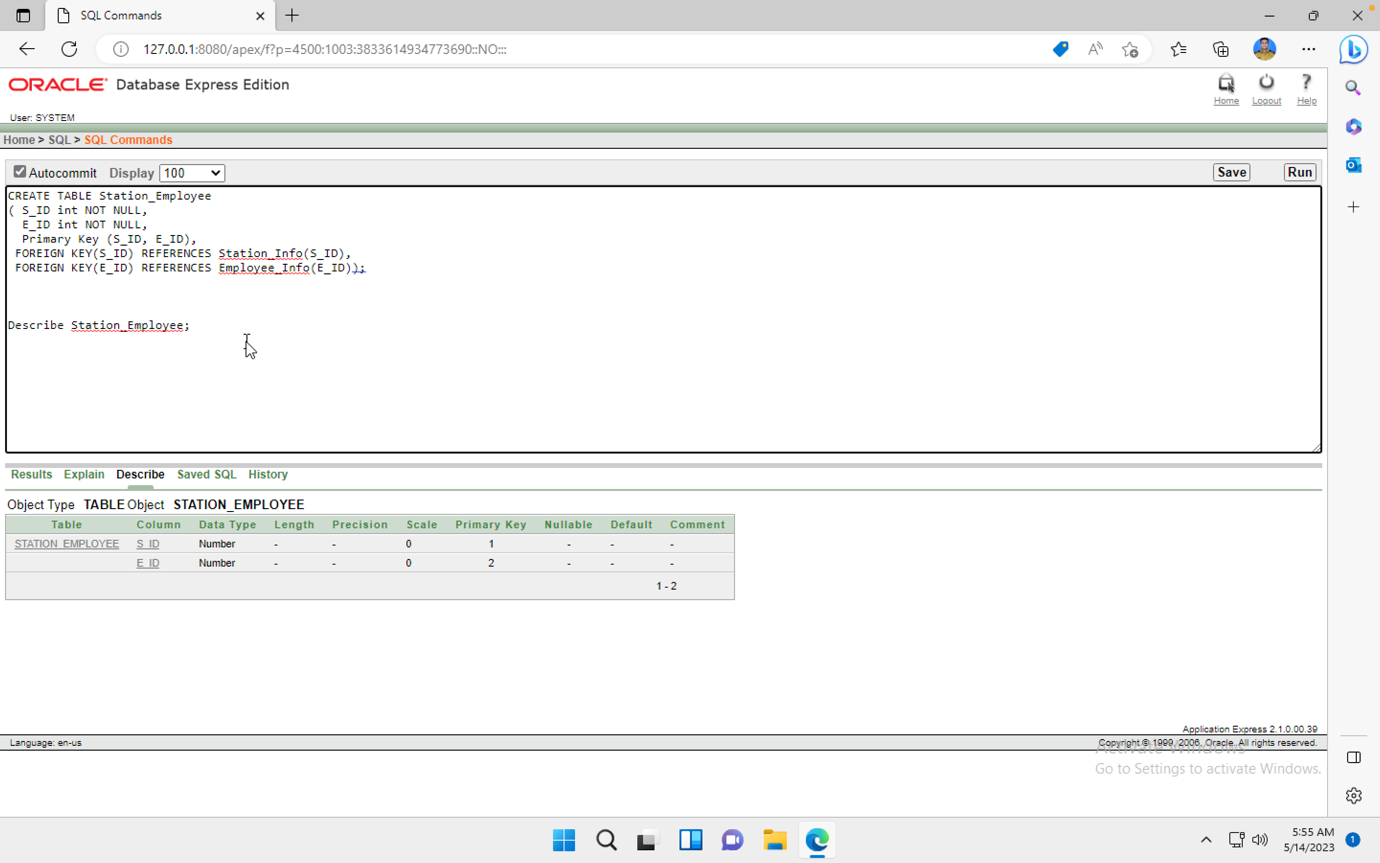
( S\_ID int NOT NULL,

E\_ID int NOT NULL,

Primary Key (S\_ID, E\_ID),

FOREIGN KEY(S\_ID) REFERENCES Station\_Info(S\_ID),

FOREIGN KEY(E\_ID) REFERENCES Employee\_Info(E\_ID));

****

**Table-09: Ticket**

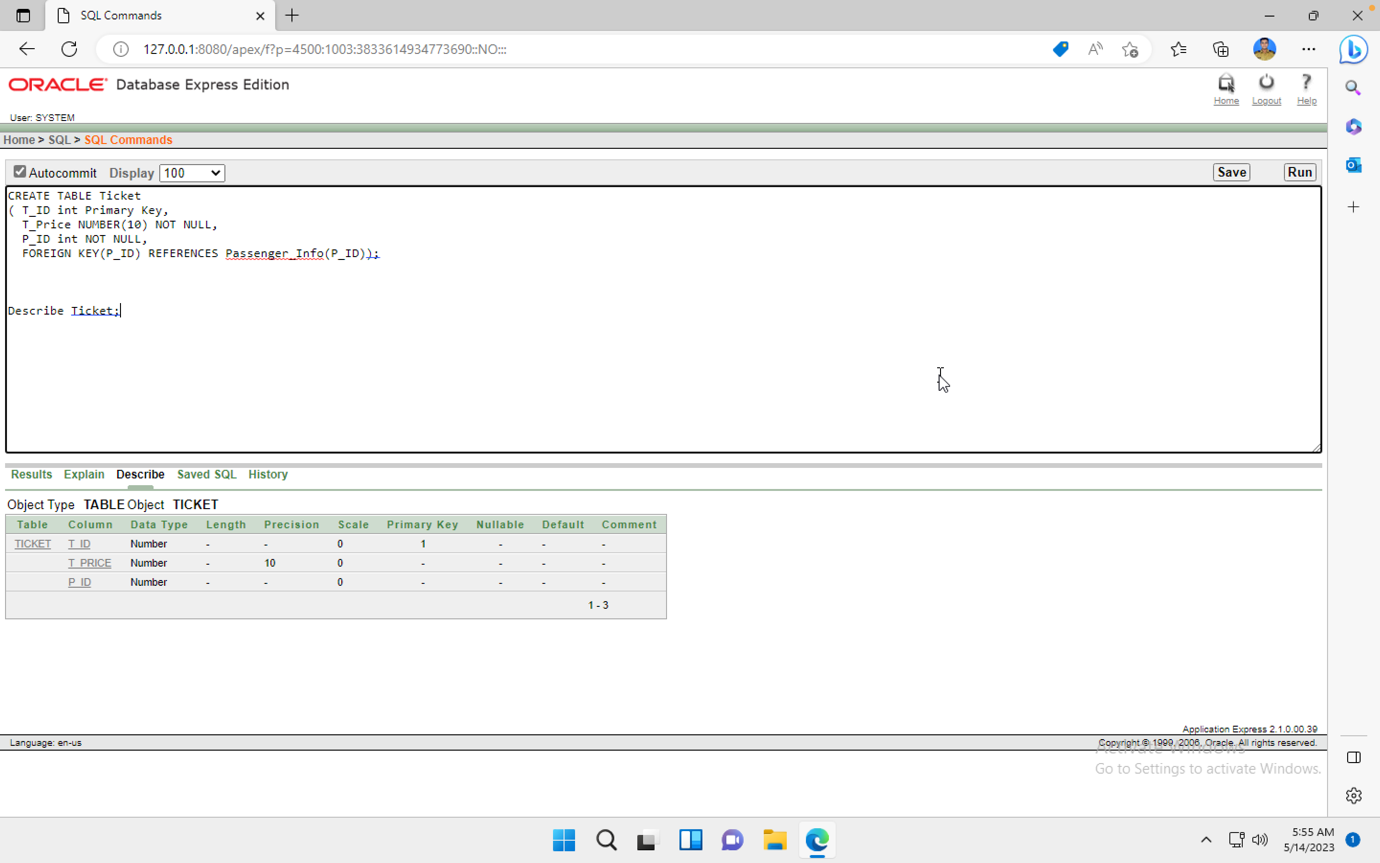
CREATE TABLE Ticket

( T\_ID int Primary Key,

T\_Price NUMBER(10) NOT NULL,

P\_ID int NOT NULL,

FOREIGN KEY(P\_ID) REFERENCES Passenger\_Info(P\_ID));

****

**Data Insertion:**

**Table-01: Metro\_Info**

Insert into Metro\_Info Values

(MT\_Num\_seq.NEXTVAL,'Bangabandhu Express', 'Uttara North',500,'20','5 Mins');

Insert into Metro\_Info Values

(MT\_Num\_seq.NEXTVAL,'Sonar Bangla Express', 'Pallabi',400,'15','5 Mins');Insert into Metro\_Info Values

(MT\_Num\_seq.NEXTVAL,'Joy Bangla Express', 'Pallabi',400,'15','5 Mins');

Insert into Metro\_Info Values

(MT\_Num\_seq.NEXTVAL,'Mahanagar Godhuli', 'Mirpur 11',600,'30','10 Mins');

Insert into Metro\_Info Values

(MT\_Num\_seq.NEXTVAL,'Udayan Express', 'Farmgate',550,'40','7 Mins');

select \*

from Metro\_Info;

**Table-02: Station\_Info**

Insert into Station\_Info Values

(S\_ID\_seq.NEXTVAL,'Pallabi');

Insert into Station\_Info Values

(S\_ID\_seq.NEXTVAL,'Mirpur 1');

Insert into Station\_Info Values

(S\_ID\_seq.NEXTVAL,'Uttara North');

Insert into Station\_Info Values

(S\_ID\_seq.NEXTVAL,'Uttara South');

Insert into Station\_Info Values

(S\_ID\_seq.NEXTVAL,'Farm gate');

select \*

from Station\_Info;

**Table-03: Metro\_Station**

Insert into Metro\_Station values

(MT\_Num\_fk1.NEXTVAL, S\_ID\_fk1.NEXTVAL);

Insert into Metro\_Station values

(MT\_Num\_fk1.NEXTVAL, S\_ID\_fk1.NEXTVAL);

Insert into Metro\_Station values

(MT\_Num\_fk1.NEXTVAL, S\_ID\_fk1.NEXTVAL);

Insert into Metro\_Station values

(MT\_Num\_fk1.NEXTVAL, S\_ID\_fk1.NEXTVAL);

Insert into Metro\_Station values

(MT\_Num\_fk1.NEXTVAL, S\_ID\_fk1.NEXTVAL);

select \*

from Metro\_Station;

**Table-04: Passenger\_Info**

Insert into Passenger\_Info values

( P\_ID\_seq.NEXTVAL, 'Abdullah Adnan',1924029299,1779169291,'Uttara North','Uttara South');

Insert into Passenger\_Info values

( P\_ID\_seq.NEXTVAL, 'Basharul Alam',1813890622,NULL,'Pallabi','Farm gate');

Insert into Passenger\_Info values

( P\_ID\_seq.NEXTVAL, 'Fatema Akhter',1344029435,1659160421,'Agargaon','Uttara North');

Insert into Passenger\_Info values

( P\_ID\_seq.NEXTVAL, 'Nowshin Ahmed',1876244229,1432362341,'Badda','Farm gate');

Insert into Passenger\_Info values

( P\_ID\_seq.NEXTVAL, 'Sakib Hasan',1652342366,1780134651,'Mirpur 11','Uttara South');

select \*

from Passenger\_Info;

**Table-05: Station\_Passenger**

Insert into Station\_Passenger values

( S\_ID\_fk2.NEXTVAL,P\_ID\_fk1.NEXTVAL);

Insert into Station\_Passenger values

( S\_ID\_fk2.NEXTVAL,P\_ID\_fk1.NEXTVAL);

Insert into Station\_Passenger values

( S\_ID\_fk2.NEXTVAL,P\_ID\_fk1.NEXTVAL);

Insert into Station\_Passenger values

( S\_ID\_fk2.NEXTVAL,P\_ID\_fk1.NEXTVAL);

Insert into Station\_Passenger values

( S\_ID\_fk2.NEXTVAL,P\_ID\_fk1.NEXTVAL);

select \*

from Station\_Passenger;

**Table-06: Employee\_Address**

Insert into Employee\_Address value

( E\_Address\_ID\_seq.NEXTVAL,'Road:06 E Block','Bashundhara, Dhaka','Bangladesh');

Insert into Employee\_Address value

( E\_Address\_ID\_seq.NEXTVAL,'Road:07 Taltola','Khilgaon, Dhaka','Bangladesh');

Insert into Employee\_Address value

( E\_Address\_ID\_seq.NEXTVAL,'Road:13/2, Kawla','Airport, Dhaka','Bangladesh');

Insert into Employee\_Address value

( E\_Address\_ID\_seq.NEXTVAL,'Road: B-6 Mugda',' Dhaka','Bangladesh');

Insert into Employee\_Address value

( E\_Address\_ID\_seq.NEXTVAL,'Road:08 C Block','Bashundhara, Dhaka','Bangladesh');

select \*

from Employee\_Address;

**Table-07: Employee\_Info**

Insert into Employee\_Info values

( E\_ID\_seq.NEXTVAL,'Station Manager', 'Jubayer Ahmed',TO\_DATE('1985-06-11', 'yyyy-mm-dd'),'NIGHT',01924226006,01715308087,E\_Address\_ID\_fk1.NEXTVAL);

Insert into Employee\_Info values

(E\_ID\_seq.NEXTVAL,'Ticketing Staff', 'Fahim Rahman',TO\_DATE('1990-01-18', 'yyyy-mm-dd'),'DAY',01736282766,01315309090,E\_Address\_ID\_fk1.NEXTVAL);

Insert into Employee\_Info values

(E\_ID\_seq.NEXTVAL,'Cleaning Staff', 'Abul Kalam',TO\_DATE('1998-05-25', 'yyyy-mm-dd'),'DAY',01435627454,NULL,E\_Address\_ID\_fk1.NEXTVAL);

Insert into Employee\_Info values

(E\_ID\_seq.NEXTVAL,'Cleaning Staff', 'Sadman Ahsan',TO\_DATE('1987-07-11', 'yyyy-mm-dd'),'NIGHT',01345466654,017262553427,E\_Address\_ID\_fk1.NEXTVAL);

Insert into Employee\_Info values

(E\_ID\_seq.NEXTVAL,'Train Operators', 'Najib Mahfuz',TO\_DATE('1999-03-07', 'yyyy-mm-dd'),'DAY',01635456454,015343426427,E\_Address\_ID\_fk1.NEXTVAL);

select \*

from Employee\_Info;

**Table-08: Station\_Employee**

Insert into Station\_Employee values

( S\_ID\_fk3.NeXTVAL,E\_ID\_fk1.NEXTVAL);

Insert into Station\_Employee values

( S\_ID\_fk3.NeXTVAL,E\_ID\_fk1.NEXTVAL);

Insert into Station\_Employee values

( S\_ID\_fk3.NeXTVAL,E\_ID\_fk1.NEXTVAL);

Insert into Station\_Employee values

( S\_ID\_fk3.NeXTVAL,E\_ID\_fk1.NEXTVAL);

Insert into Station\_Employee values

( S\_ID\_fk3.NeXTVAL,E\_ID\_fk1.NEXTVAL);

select \*

from Station\_Employee;

**Table-09: Ticket**

Insert into Ticket values

( T\_ID\_seq.NEXTVAL, 60, P\_ID\_fk2.NEXTVAL);

Insert into Ticket values

( T\_ID\_seq.NEXTVAL, 80, P\_ID\_fk2.NEXTVAL);

Insert into Ticket values

( T\_ID\_seq.NEXTVAL, 90, P\_ID\_fk2.NEXTVAL);

Insert into Ticket values

( T\_ID\_seq.NEXTVAL, 100, P\_ID\_fk2.NEXTVAL);

Insert into Ticket values

( T\_ID\_seq.NEXTVAL, 50, P\_ID\_fk2.NEXTVAL);

select \*

from Ticket;

**Sub-Query:**

1. Subquery to retrieve all passengers who have bought a ticket with a price greater than $50:

SELECT P\_Name

FROM Passenger\_Info

WHERE P\_ID IN (SELECT P\_ID FROM Ticket WHERE T\_Price > 50);

**A screenshot of a computer

Description automatically generated**

**Sub-Query-01**

1. Subquery to retrieve all station employees who work in stations located in "Pallabi":

SELECT E\_Name

FROM Employee\_Info

WHERE E\_ID IN (SELECT E\_ID FROM Station\_Employee WHERE S\_ID IN (SELECT S\_ID FROM Station\_Info WHERE S\_Name = 'Pallabi'));

A screenshot of a computer

Description automatically generated

**Sub-Query-02**

1. Subquery to retrieve all metro routes that have a capacity greater than 500:

SELECT MT\_Name

FROM Metro\_Info

WHERE MT\_Route IN (SELECT MT\_Route FROM Metro\_Info WHERE MT\_Capacity > 500);

A screenshot of a computer

Description automatically generated

**Sub-Query-03**

**Joining(Equi-join):**

1. Join the "Passenger\_Info" and "Ticket" tables on the "P\_ID" column and retrieve the passenger name and destination for each ticket.

Select P\_Name, Destination

From Passenger\_Info, Ticket

Where Passenger\_Info.P\_ID=Ticket.P\_ID;

A screenshot of a computer

Description automatically generated

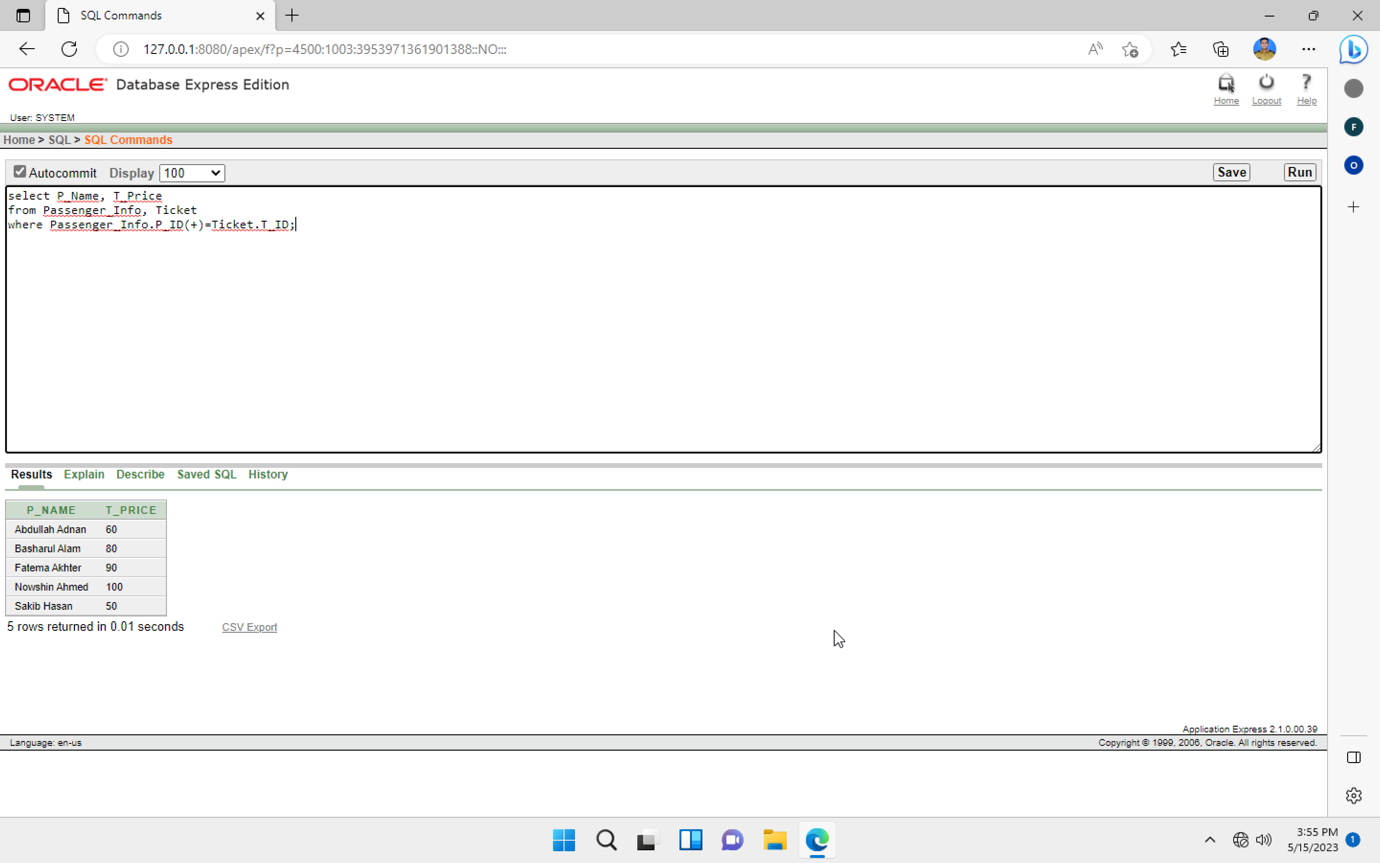
**Joining(Outer-join):**

1. Perform a left outer join between the "Passenger\_Info" and "Ticket" tables on the "P\_ID" column. Retrieve the passenger name and ticket price for each matching record, including all records from the left table (Passenger\_Info) regardless of whether there is a match in the right table (Ticket).

SELECT P\_Name, T\_Price

FROM Passenger\_Info, Ticket

WHERE Passenger\_Info.P\_ID(+)=Ticket.T\_ID;



1. Perform a full outer join between the "Passenger\_Info", "Station\_Passenger", and "Station\_Info" tables on the matching "P\_ID" and "S\_ID" columns. Retrieve the passenger name and station name for each matching record, including all records from all tables regardless of whether there is a match in the other tables.

SELECT P\_Name, S\_Name

FROM Passenger\_Info, Station\_Passenger, Station\_Info

WHERE Passenger\_Info.P\_ID = Station\_Passenger.P\_ID

AND Station\_Passenger.S\_ID = Station\_Info.S\_ID;

A screenshot of a computer

Description automatically generated

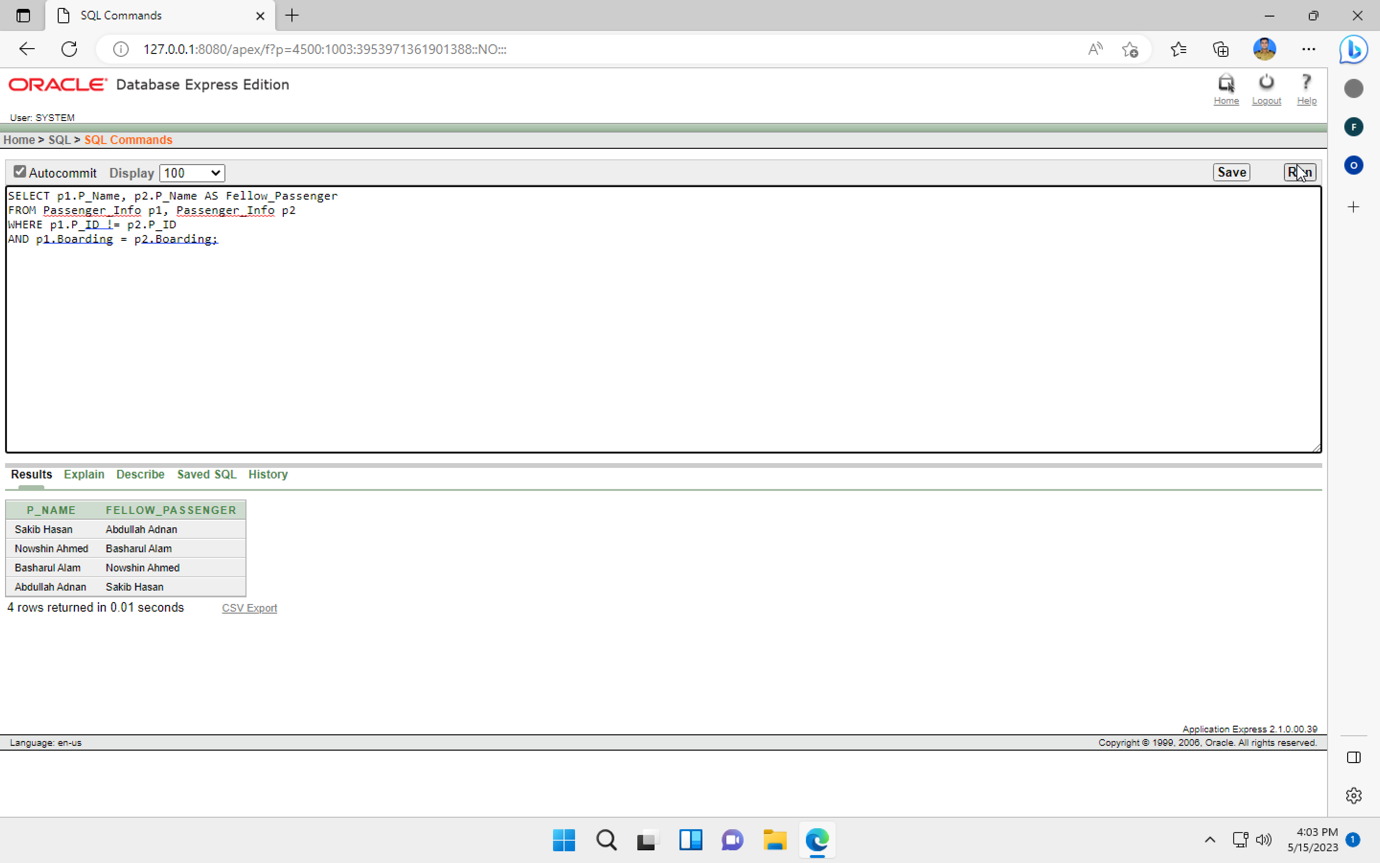
**Joining(Self-join):**

1.Query that joins the "Passenger\_Info" table with itself based on the "P\_ID" column, retrieving the names of passengers and their fellow passengers who have the same boarding location.

SELECT p1.P\_Name, p2.P\_Name AS Fellow\_Passenger

FROM Passenger\_Info p1, Passenger\_Info p2

WHERE p1.P\_ID != p2.P\_ID AND p1.Boarding = p2.Boarding;



**Simple View:**

In this query, we create a view named "Metro\_Stations" that retrieves the station ID and station name from the "Station\_Info" table. The view allows us to access this information as if it were a virtual table.

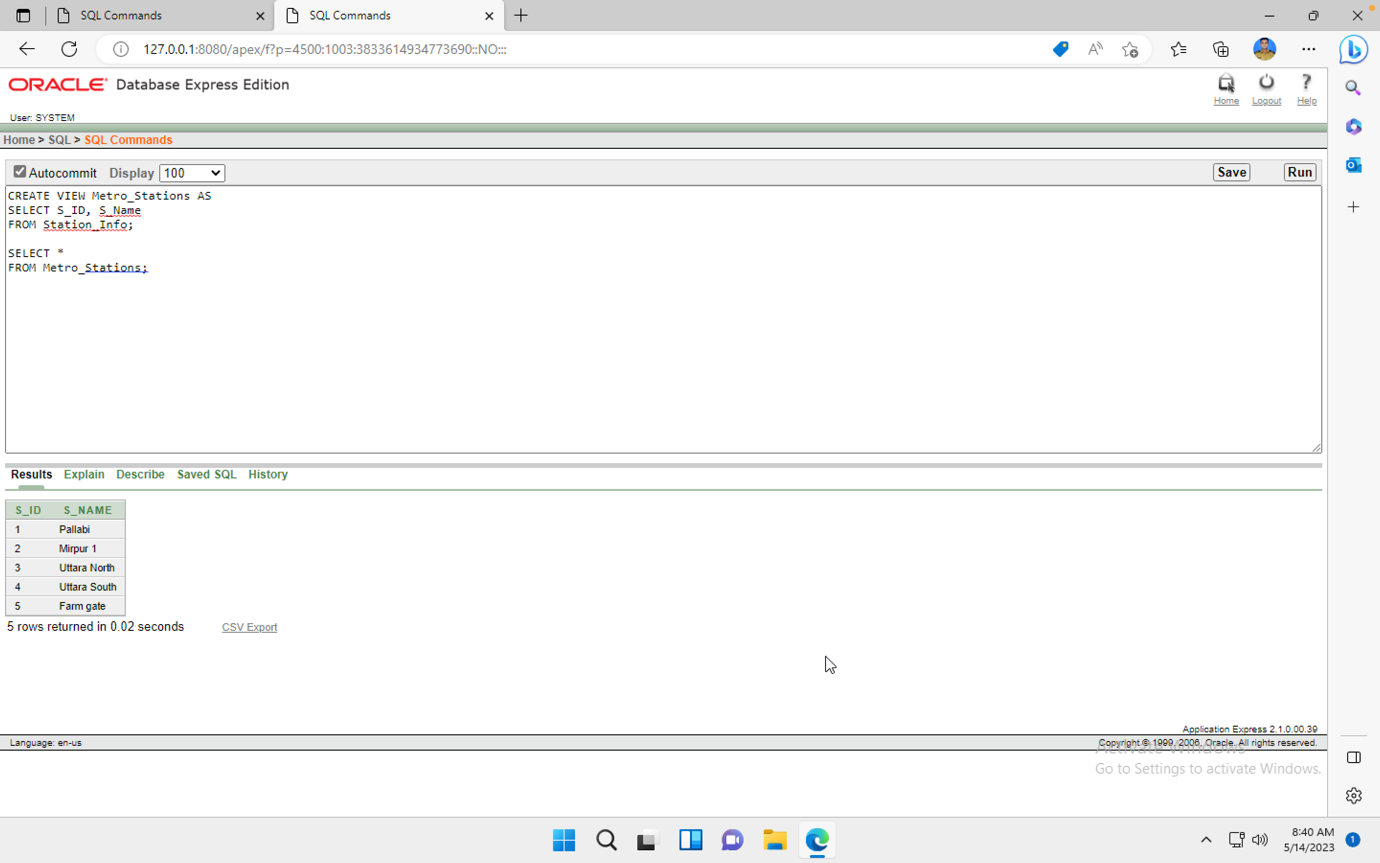
CREATE VIEW Metro\_Stations AS

SELECT S\_ID, S\_Name

FROM Station\_Info;

SELECT \*

FROM Metro\_Stations;



**Complex View:**

We create a view called "Passenger\_Ticket\_Info" that combines data from the "Passenger\_Info" and "Ticket" tables. The view includes the passenger ID, name, destination, and ticket price for each matching record. It includes all records from the left table (Passenger\_Info) even if there is no match in the right table (Ticket).

CREATE VIEW Passenger\_Ticket\_Info AS

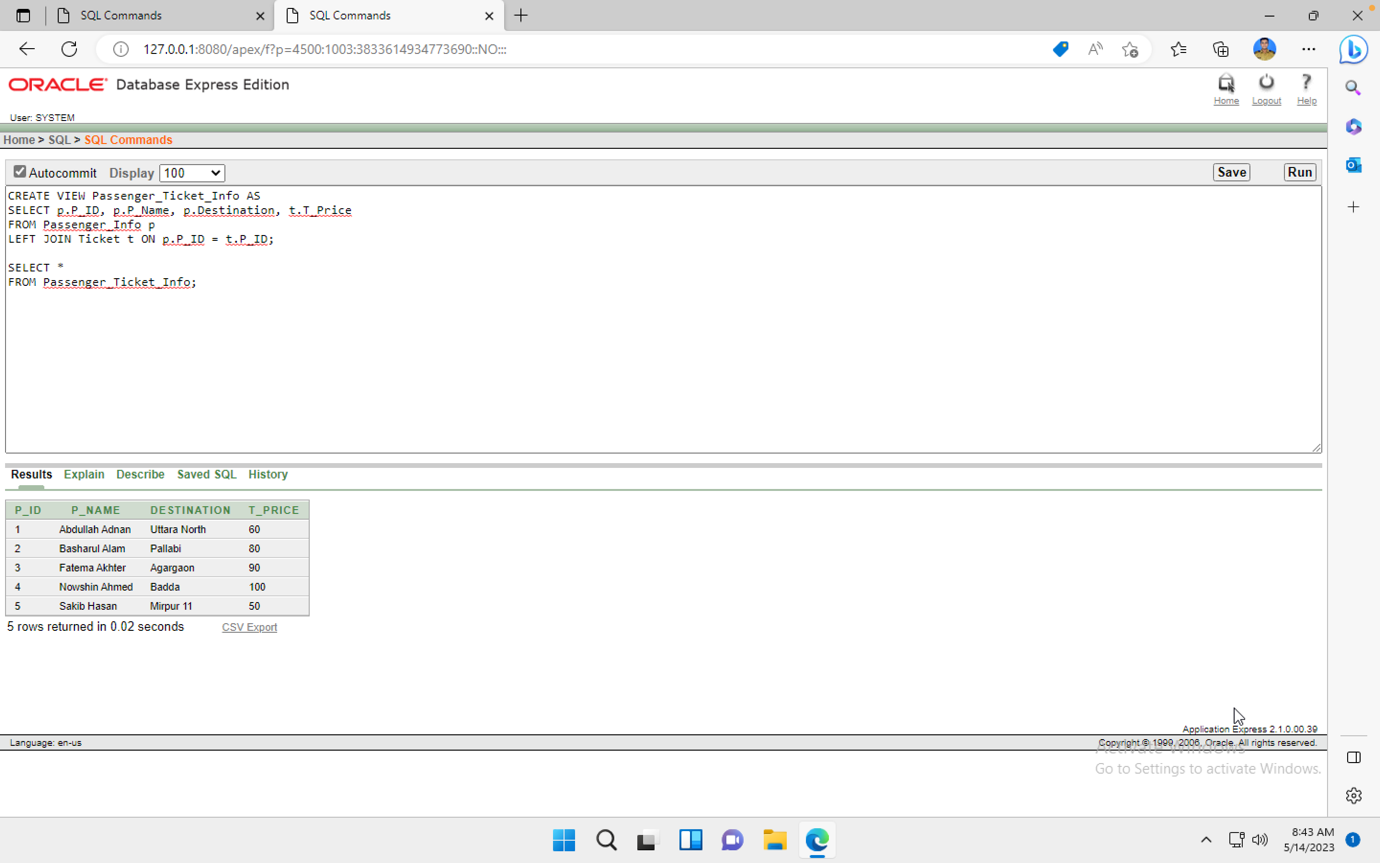
SELECT p.P\_ID, p.P\_Name, p.Destination, t.T\_Price

FROM Passenger\_Info p

LEFT JOIN Ticket t ON p.P\_ID = t.P\_ID;

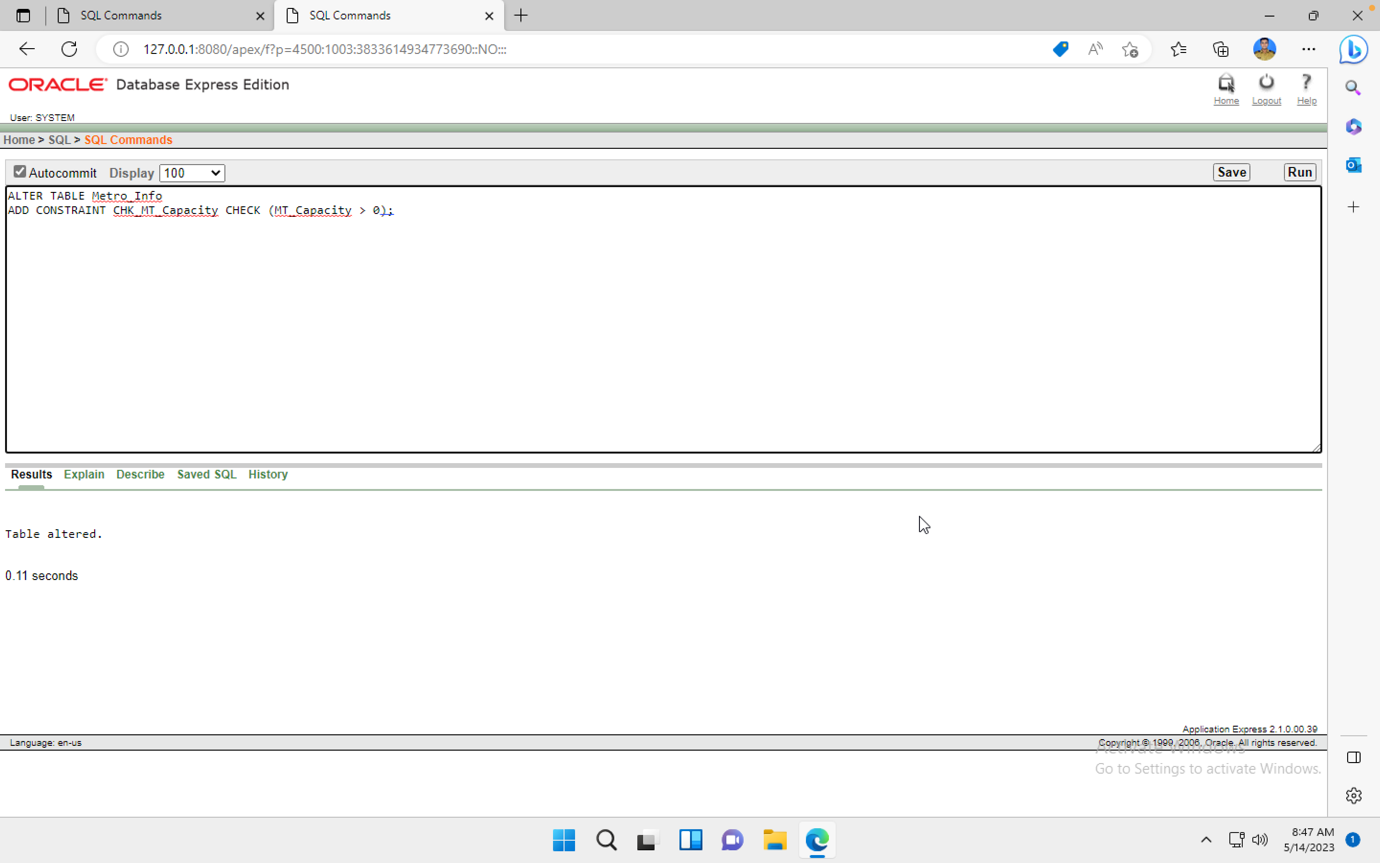
SELECT \*

FROM Passenger\_Ticket\_Info;



**Addition of Constraint:**

1.Query that adds a check constraint named "CHK\_MT\_Capacity" to the "Metro\_Info" table, ensuring that the "MT\_Capacity" column has values greater than 0.



**Conclusion:**

In conclusion, a thorough and effective management system for metro rail operations has been created as part of the metro rail management database project. The database schema includes the essential linkages and constraints, as well as tables for metros, stations, travelers, staff, and ticketing. It makes it possible to manage, save, and retrieve information seamlessly while preserving the quality and integrity of the data. The initiative aids in efficient decision-making, analysis, and reporting, which helps the metro system run smoothly. The metro rail management database project, taken as a whole, offers a strong foundation for improving operational effectiveness, passenger management, employee administration, and ticketing activities in a metro rail setting.