**Department of Computer Science**

|  |  |
| --- | --- |
| Database Systems Project | |
|  |

**BSCS(Hons)**

**4th Semester**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Speedo Bus Facilitation System

**Submitted to**

**Sir Muhammad Hafeez**

Department of Computer Science

GC University, Lahore

**Submitted by**

Rizwan Ashraf 0084-BSCS-20

Muhammad Rameez 0090-BSCS-20

**Section:** C

Table of Contents

[**History of Speedo Bus** 3](#_Toc15716)

[**Define Problems and Constraints** 3](#_Toc15717)

[**Define Objectives** 4](#_Toc15718)

[**Define Scope and Boundaries** 4](#_Toc15719)

[**Boundary** 4](#_Toc15720)

[**Story Board of an Facilitation** 4](#_Toc15721)

[**Noun Verb Analysis** 5](#_Toc15722)

[**Entity RelationshipDiagram** 6](#_Toc15723)

[**Abnormal Form** 6](#_Toc15724)

[**Normalization-Form** 7](#_Toc15725)

[**Implementation** 7](#_Toc15727)

[**Relational Model of the Database** 14](#_Toc15728)

[**Testing** 14](#_Toc15729)

Speedo Bus Facilitation

The Centralized network of [public transport in Lahore](https://www.zameen.com/blog/public-transport-lahore.html) has now made it easier than ever to travel throughout the city in low cost, safely, and comfortably. Along with the [Lahore Metrobus Service (LMBS)](https://www.zameen.com/blog/lahore-metrobus-system.html) and the [Orange Line of the Lahore Rapid Mass Transit System (LRMTS)](https://www.zameen.com/blog/orange-line-metro-train-features-benefits.html), several feeder buses, also called Speedo buses, help the citizens commute from one point in the city to another, using various predefined routes. Let’s cover the Speedo bus routes in Lahore that have been launched in Phase 1 of the Lahore Metrobus Service to facilitate commute over 14 default routes.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# History of Mass Transit

The LMBS is managed by the Punjab Mass transit Authority (PMA) and was launched in 2013. In the first phase of the project, nearly 200 buses were connected to the central Metrobus route in partnership with Daewoo Pakistan Express Bus Service Ltd. to cover 14 predetermined feeder bus routes. These buses now travel along with regular traffic on their set routes, picking up passengers from predetermined bus stops to drop them off at the nearest Metrobus station from where it is easier than ever to travel all over the city rapidly.

Daewoo has been operating inter-city buses in Pakistan for nearly two decades and is now overseeing all the intra-city buses in Lahore that run on the 14 Speedo bus routes.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Define Problems and Constraints

The company has number of the buses working in Lahore, they want to store data including the bus number, bus driver data, bus route details, bus location data.

Similarly, they want to save the data of customers/Passengers their Bus cards, check in and check out time, their card current balance and the previous balance.

Moreover, they want to store the data of their staff including details of driver, bus attendant, and their other workers.

The biggest constraint in this database is that we must make the data separated from driver to passenger and bus attendant because the mass transit allows bus attendant to recharge their card more then a normal passenger. However, a normal passenger can charge their card up to 1000Rs only.

Similarly, the driver card has different facilities then a bus attendant and passenger card have. Driver uses his bus card to set the next route for new trip however bus attendant and passengers uses its card to check in for new ride.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Define Objectives

Our objective is to create the database for the speedo bus to keep detailed record of all check in and check out details of passengers and the bus card.

This will store the records of all drivers, bus attendants and passengers.

This system will also keep record of details related to bus, their routes and the bus stops.

\_\_\_\_\_\_\_\_\_\_\_\_\_

# Define Scope and Boundaries

Our main entities are

Bus, Bus Stop, Stops, Route, Driver, Bus Attendant, Card and Token.

The driver drives the bus with their card, the passenger travel in the bus with their card, the bus attendant can manage bus fare with bus card bus, the bus covers their route, bus stops at bus stop, route have bus stop, bus stop derived from stop or passenger have token. Passenger checkin or checkout at bus stop.

Our scope will be around these entities.

# Boundary

The machine which run this system is not too heavy. This system will run on home computer also.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Story Board of an Organization

Speedo Bus works with multiple buses and multiple routes in a same city. Each Bus have its id, number, location, departure time. Bus also required the details of driver and routes. However, every driver has some attribute like id, name, cnic, address, age, experience, salary and his card. Every Driver can drive only one bus, but one bus is driven by more then one driver. Bus must have at least one route but one route can be a path of more than one bus. Each route have its number, starting point, ending point, distance, and bus details. Driver can drive the bus with or without passengers but passenger must have only one bus. Passenger must have its id, bus number, and card number. One card can be used in only one bus at a time and it can be used single time in a bus. Bus have many bus stops and each bus stop have its id, name, number, and bus number. Bus stop is derived from the stops but one stops have many bus stops. Stop contains all the bus stops details having its id, name.

Passenger can travel with card or with token but one card or one token must have only one passenger. Bus card includes its id, number, balance, limit. Token have its id, number, and valid time. Many Buses can stop at one bus stop and one bus can stop at multiple bus stops.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Noun Verb Analysis

The Noun Verb Analysis of the Following Scenario are as follows:

Diagram, engineering drawing

Description automatically generated \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_



# Entity RelationshipDiagram

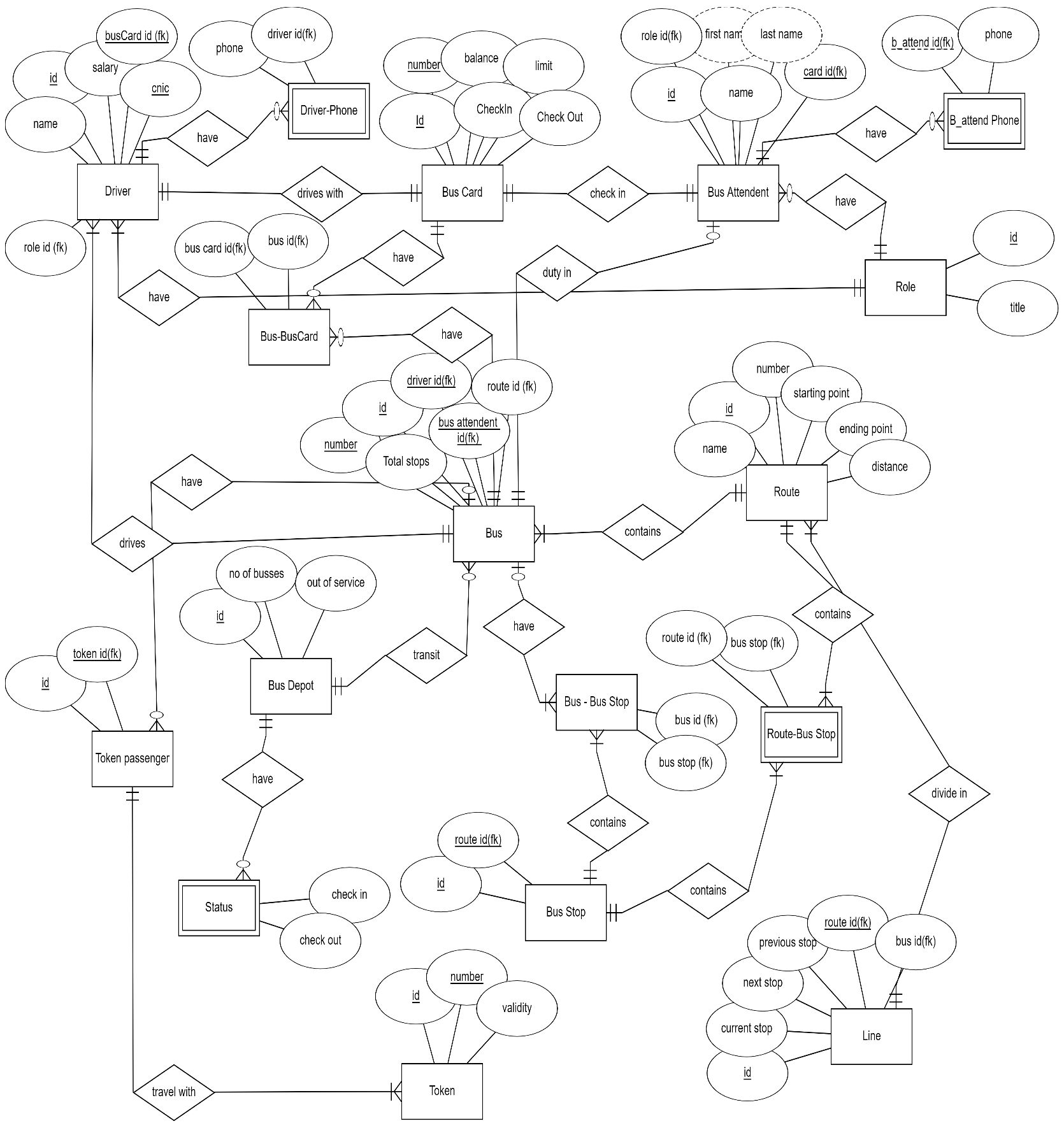
## Abnormal Form

A close-up of a computer keyboard

Description automatically generated with low confidence

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

## Normalization-Form



\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Implementation

CREATE DATABASE SPEEDO

- - USE BUS

CREATE TABLE Bus\_Depot

( id NUMBER(10) NOT NULL,

no\_of\_busses NUMBER(10) NOT NULL,

out\_of\_service NUMBER(10) NOT NULL,

PRIMARY KEY (id));

ALTER TABLE bus\_depot

ADD Name varchar(255);

create table Role -----------------

( RoleId number(10) not null,

Title varchar(20) not null,

Card\_Limit number(4) not null )

insert into role(roleid,title, card\_limit) values (1,'Bus Attendent', 5000);

insert into role(roleid,title, card\_limit) values (2,'Driver', 9000);

insert into role(roleid,title, card\_limit) values (3,'Passenger', 1000);

CREATE TABLE Bus\_Depot ----------------

(

id NUMBER(10) NOT NULL,

no\_of\_busses NUMBER(10) NOT NULL,

bus\_check\_in Varchar2(20),

bus\_check\_out Varchar2(20),

out\_of\_service NUMBER(10) NOT NULL,

PRIMARY KEY (id)

);

desc bus\_depot;

insert into bus\_depot(id, no\_of\_busses, bus\_check\_in, bus\_check\_out, out\_of\_service, name ) values (1, 20 , 'May 12, 2022', 'May 13, 2022', 3, 'main');

insert into bus\_depot(id, no\_of\_busses, bus\_check\_in, bus\_check\_out, out\_of\_service, name ) values (2, 10 , 'May 13, 2022', 'May 14, 2022', 4, 'main');

insert into bus\_depot(id, no\_of\_busses, bus\_check\_in, bus\_check\_out, out\_of\_service, name ) values (3, 12 , 'May 14, 2022', 'May 15, 2022', 1, 'main');

insert into bus\_depot(id, no\_of\_busses, bus\_check\_in, bus\_check\_out, out\_of\_service, name ) values (4, 10 , 'May 16, 2022', 'May 16, 2022', 6, 'main');

insert into bus\_depot(id, no\_of\_busses, bus\_check\_in, bus\_check\_out, out\_of\_service, name ) values (5, 15 , 'May 17, 2022', 'May 17, 2022', 16, 'main');

insert into bus\_depot(id, no\_of\_busses, bus\_check\_in, bus\_check\_out, out\_of\_service, name ) values (6, 1 , 'May 18, 2022', 'May 18, 2022', 14, 'main');

insert into bus\_depot(id, no\_of\_busses, bus\_check\_in, bus\_check\_out, out\_of\_service, name ) values (6, 1 , 'May 18, 2022', 'May 18, 2022', 14, 'main');

insert into bus\_depot(id, no\_of\_busses, bus\_check\_in, bus\_check\_out, out\_of\_service, name ) values (8, 12 , 'May 20, 2022', 'May 21, 2022', 12, 'main');

insert into bus\_depot(id, no\_of\_busses, bus\_check\_in, bus\_check\_out, out\_of\_service, name ) values (9, 11 , 'May 22, 2022', 'May 22, 2022', 22, 'terminal');

insert into bus\_depot(id, no\_of\_busses, bus\_check\_in, bus\_check\_out, out\_of\_service, name ) values (10, 4 , 'May 23, 2022', 'May 23, 2022', 2, 'terminal');

insert into bus\_depot(id, no\_of\_busses, bus\_check\_in, bus\_check\_out, out\_of\_service, name ) values (11, 2 , 'May 24, 2022', 'May 25, 2022', 1, 'terminal');

select \* from bus\_depot;

select \* from bus\_depot where out\_of\_service > 10;

select \* from bus\_depot where out\_of\_service between 10 and 20;

CREATE TABLE Token ----------------

(

TokenId NUMBER(10) NOT NULL,

valid\_Time Varchar2(20),

num NUMBER(30) NOT NULL,

PRIMARY KEY (TokenId)

)

desc token;

insert into token(tokenid, valid\_time, num) values (1, '1 hour', 281033898213);

insert into token(tokenid, valid\_time, num) values (2, '1.5 hour', 2231233898213);

insert into token(tokenid, valid\_time, num) values (3, '5 hour', 2231233898213);

insert into token(tokenid, valid\_time, num) values (4, '5.4 hour', 22312143898213);

insert into token(tokenid, valid\_time, num) values (5, '4 hour', 124123112312312);

insert into token(tokenid, valid\_time, num) values (6, '5 hour', 12412235212312);

insert into token(tokenid, valid\_time, num) values (7, '5.32 hour', 124123523242312);

insert into token(tokenid, valid\_time, num) values (8, '24 hour', 12412323242312);

insert into token(tokenid, valid\_time, num) values (9, '23 hour', 12412323225712);

insert into token(tokenid, valid\_time, num) values (10, '3 hour', 12412329325712);

insert into token(tokenid, valid\_time, num) values (11, '3 hour', 1241232325712);

insert into token(tokenid, valid\_time, num) values (12, '13 hour', 1241212232325712);

insert into token(tokenid, valid\_time, num) values (13, '13.4 hour', 124124232325712);

select \* from token;

select \* from token order by num desc;

CREATE TABLE Token\_Passenger ----------------

(id NUMBER(10) NOT NULL,

TokenId NUMBER(30) NOT NULL,

PRIMARY KEY (id),

CONSTRAINT FK\_Token FOREIGN KEY (TokenId)

REFERENCES Token(TokenId))

desc token\_passenger;

insert into token\_passenger(id, tokenid) values (1, 3);

insert into token\_passenger(id, tokenid) values (2, 1);

insert into token\_passenger(id, tokenid) values (3, 2);

insert into token\_passenger(id, tokenid) values (4, 5);

insert into token\_passenger(id, tokenid) values (5, 4);

insert into token\_passenger(id, tokenid) values (7, 13);

insert into token\_passenger(id, tokenid) values (8, 10);

insert into token\_passenger(id, tokenid) values (9, 9);

insert into token\_passenger(id, tokenid) values (10, 6);

insert into token\_passenger(id, tokenid) values (11, 8);

insert into token\_passenger(id, tokenid) values (12, 7);

insert into token\_passenger(id, tokenid) values (13, 12);

select \* from token\_passenger where tokenId <> 2 or tokenId < 3;

select \* from token\_passenger where tokenId <> 2 and tokenId < 3;

CREATE TABLE Route ----------------

(

RouteId NUMBER(10) NOT NULL,

R\_Number NUMBER(10) NOT NULL,

Distance NUMBER(10) NOT NULL,

Starting\_point varchar(255) not null,

Ending\_point varchar(255) not null,

PRIMARY KEY (RouteId)

)

insert into route(routeid,r\_number,distance,starting\_point,ending\_point) values (1, 15, 20, 'Bhatti Chowk', 'RA Bazar');

insert into route(routeid,r\_number,distance,starting\_point,ending\_point) values (2, 23, 20, 'Railway Station', 'Rana Town');

insert into route(routeid,r\_number,distance,starting\_point,ending\_point) values (3, 2, 20, 'Nasir Bagh', 'Rana Town');

insert into route(routeid,r\_number,distance,starting\_point,ending\_point) values (4, 3, 20, 'Railway Station', 'Rana Town via Mini market');

insert into route(routeid,r\_number,distance,starting\_point,ending\_point) values (5, 4, 30, 'Jilani Park', 'Cantt');

insert into route(routeid,r\_number,distance,starting\_point,ending\_point) values (6, 5, 30, 'Multan Road ', 'Muzafferabad');

insert into route(routeid,r\_number,distance,starting\_point,ending\_point) values (7, 6, 30, 'Katchery ', 'Janazgah');

insert into route(routeid,r\_number,distance,starting\_point,ending\_point) values (8, 7, 30, 'Mini Market ', 'Bhatti Chowk');

insert into route(routeid,r\_number,distance,starting\_point,ending\_point) values (9, 7, 30, 'Max Opera ', 'Cavilary Chowk');

insert into route(routeid,r\_number,distance,starting\_point,ending\_point) values (10, 8, 25, 'Amritsar Chowk Opera ', 'Wagha Border');

insert into route(routeid,r\_number,distance,starting\_point,ending\_point) values (11, 9, 33, 'Shahdra', 'Gujumata');

select \* from route;

select \* from route where starting\_point Like('M%');

select \* from route where starting\_point in('Jilani Park', 'Shahdra')

CREATE TABLE Bus ----------------

(

BusId NUMBER(10) NOT NULL,

B\_Number NUMBER(10) NOT NULL,

Location NUMBER(10) ,

departureTime number(20) not null,

RouteId NUMBER(10) NOT NULL,

PRIMARY KEY (BusId),

CONSTRAINT FK\_RouteId FOREIGN KEY (RouteId)

REFERENCES Route(RouteId)

)

CREATE TABLE Line ----------------

(

LineId NUMBER(10) NOT NULL,

Current\_Stop Varchar2(30) ,

Previous\_Stop Varchar2(30) ,

Next\_Stop Varchar2(30) ,

RouteId NUMBER(10) NOT NULL,

BusId NUMBER(10) NOT NULL,

PRIMARY KEY (LineId),

CONSTRAINT FK\_RId FOREIGN KEY (RouteId)

REFERENCES Route(RouteId),

CONSTRAINT FK\_BusId FOREIGN KEY (BusId)

REFERENCES Bus(BusId)

)

CREATE TABLE Card ----------------

(

CardId NUMBER(10) NOT NULL,

Balance NUMBER(10) ,

C\_Number Number(30) not null ,

PRIMARY KEY (CardId)

)

desc card;

insert into card(cardid, balance, c\_number) values (1, 200, 4242342343);

insert into card(cardid, balance, c\_number) values (2, 300, 42423423343);

insert into card(cardid, balance, c\_number) values (3, 400, 424533543343);

insert into card(cardid, balance, c\_number) values (4, 2300, 2343232432432);

insert into card(cardid, balance, c\_number) values (6, 9000, 273178293721937);

insert into card(cardid, balance, c\_number) values (7, 1000, 273178721937);

insert into card(cardid, balance, c\_number) values (8, 1000, 27317832721937);

insert into card(cardid, balance, c\_number) values (9, 2000, 27317832721937);

insert into card(cardid, balance, c\_number) values (10, 5000, 273214721937);

insert into card(cardid, balance, c\_number) values (11, 2000, 27322314721937);

insert into card(cardid, balance, c\_number) values (12, 2400, 27322314722417);

insert into card(cardid, balance, c\_number) values (13, 4300, 2732231873497);

select \* from card;

select Max(balance), Min(Balance) from card;

select Sum(Balance) from card;

select count(Balance) from card;

CREATE TABLE Attendent ----------------

(

AttendentId NUMBER(10) NOT NULL,

Name Varchar2(30) ,

Shift Varchar2(20) ,

Address Varchar2(255) ,

Cnic Number(30) not null ,

Salary Number(6) ,

BusId Number(30) not null ,

CardId Number(30) not null ,

PRIMARY KEY (AttendentId),

CONSTRAINT FK\_BId FOREIGN KEY (BusId)

REFERENCES Bus(BusId),

CONSTRAINT FK\_CardId FOREIGN KEY (CardId)

REFERENCES Card(CardId)

)

CREATE TABLE Driver ----------------

(

DriverId NUMBER(10) NOT NULL,

FirstName Varchar2(30) ,

LastName Varchar2(30) ,

Shift Varchar2(20) ,

Address Varchar2(255) ,

Cnic Number(30) not null ,

Salary Number(6) ,

BusId Number(30) not null ,

CardId Number(30) not null ,

PRIMARY KEY (DriverId),

CONSTRAINT FK\_BuId FOREIGN KEY (BusId)

REFERENCES Bus(BusId),

CONSTRAINT FK\_CId FOREIGN KEY (CardId)

REFERENCES Card(CardId)

)

Insert into driver(id, firstname,lastname, shift,address, cnic, salary, busid, cardid) values (8, 'Muhammad', 'Rameez', 'Morning', '123-C Block', 123, 45000, 2, 3);

CREATE TABLE Passenger ----------------

(

PassengerId NUMBER(10) NOT NULL,

BusId Number(30) not null ,

CardId Number(30) not null ,

PRIMARY KEY (PassengerId),

CONSTRAINT FK\_BubId FOREIGN KEY (BusId)

REFERENCES Bus(BusId),

CONSTRAINT FK\_CaId FOREIGN KEY (CardId)

REFERENCES Card(CardId)

)

CREATE TABLE BusStop ----------------

(

BusStopId NUMBER(10) NOT NULL,

Name VarChar2(50) not null ,

BusId Number(30) not null ,

RouteId Number(30) not null ,

PRIMARY KEY (BusStopId),

CONSTRAINT FK\_BbId FOREIGN KEY (BusId)

REFERENCES Bus(BusId),

CONSTRAINT FK\_RoId FOREIGN KEY (RouteId)

REFERENCES Route(RouteId)

)

Insert into driver(id, name, age, salary, expe) values (1, 'Ali', 25, 20000, 2);

Insert into driver(id, name, age, salary, expe) values (2, 'Ahmed', 26, 25000, 3);

Insert into driver(id, name, age, salary, expe) values (3, 'Hashir', 22, 35000, 3);

Insert into driver(id, name, age, salary, expe) values (4, 'Meer', 20, 30000, 1);

Insert into driver(id, name, age, salary, expe) values (5, 'Adil', 21, 35000, 1);

Insert into driver(id, name, age, salary, expe) values (6, 'Bilal', 20, 45000, 4);

Insert into driver(id, name, age, salary, expe) values (7, 'Rizwan', 22, 45000, 2);

Insert into driver(id, name, age, salary, expe) values (8, 'Rameez', 20, 45000, 2);

Insert into driver(id, name, age, salary, expe) values (9, 'Mohsin', 19, 15000, 1);

Insert into driver(id, name, age, salary, expe) values (10, 'Basheer', 23, 35000, 2);

insert into driver(id, name, age , salary, expe ) values(11, 'Shoaib', 22, 12000, 1);

insert into driver(id, name, age , salary, expe ) values(12, 'Shahzaneer', 21, 200000, 2);

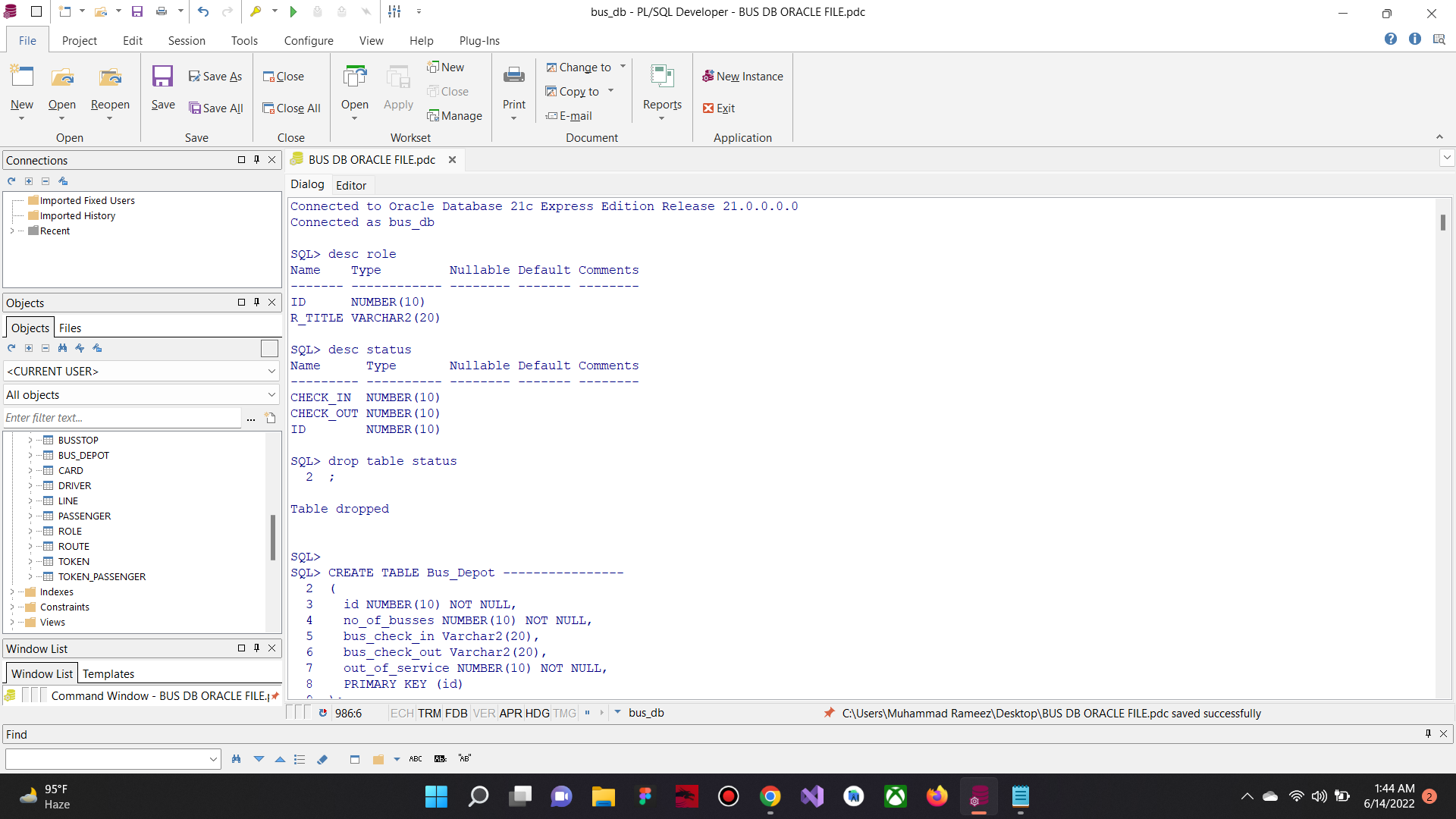
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

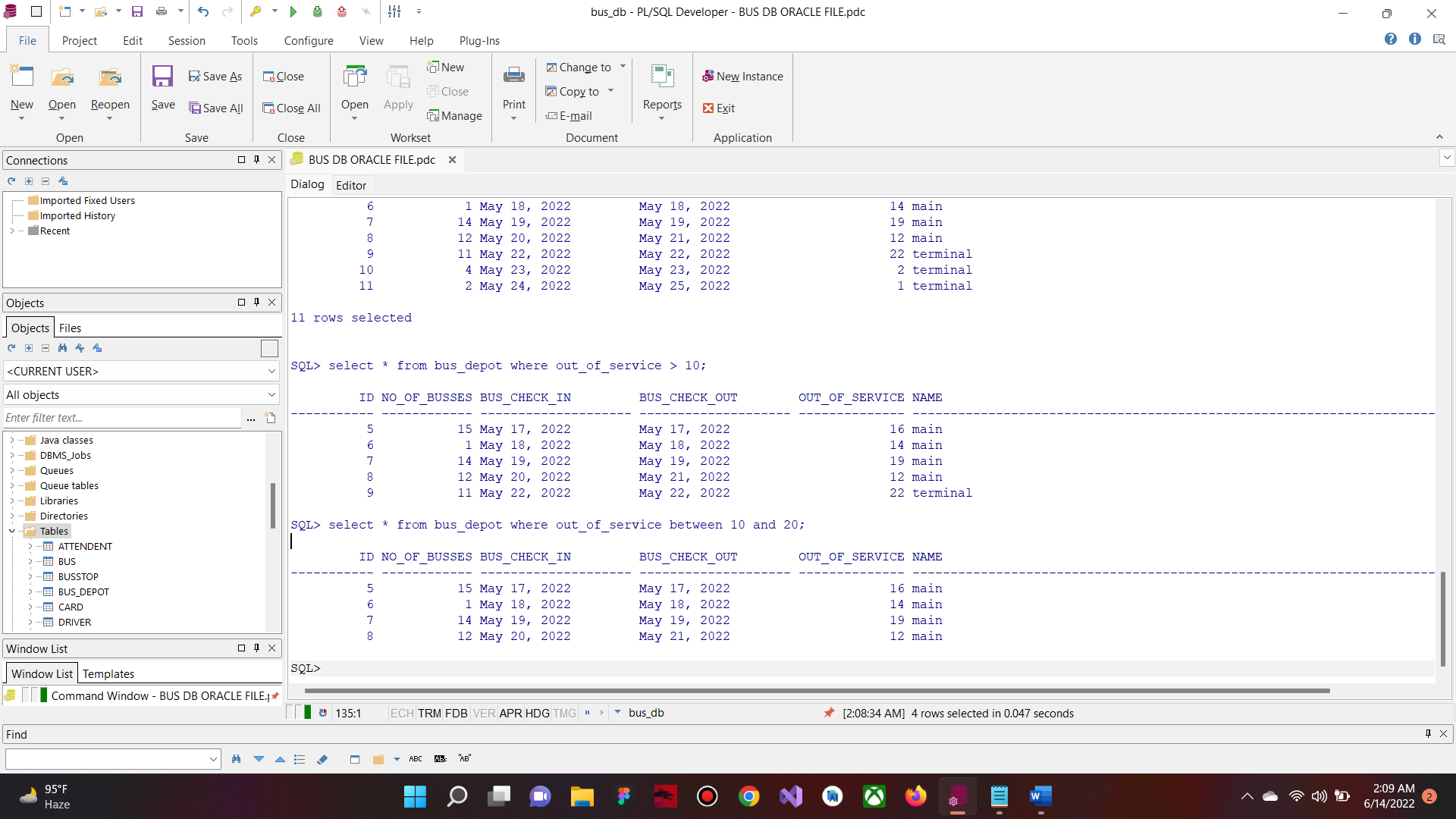
# Relational Model of the Database

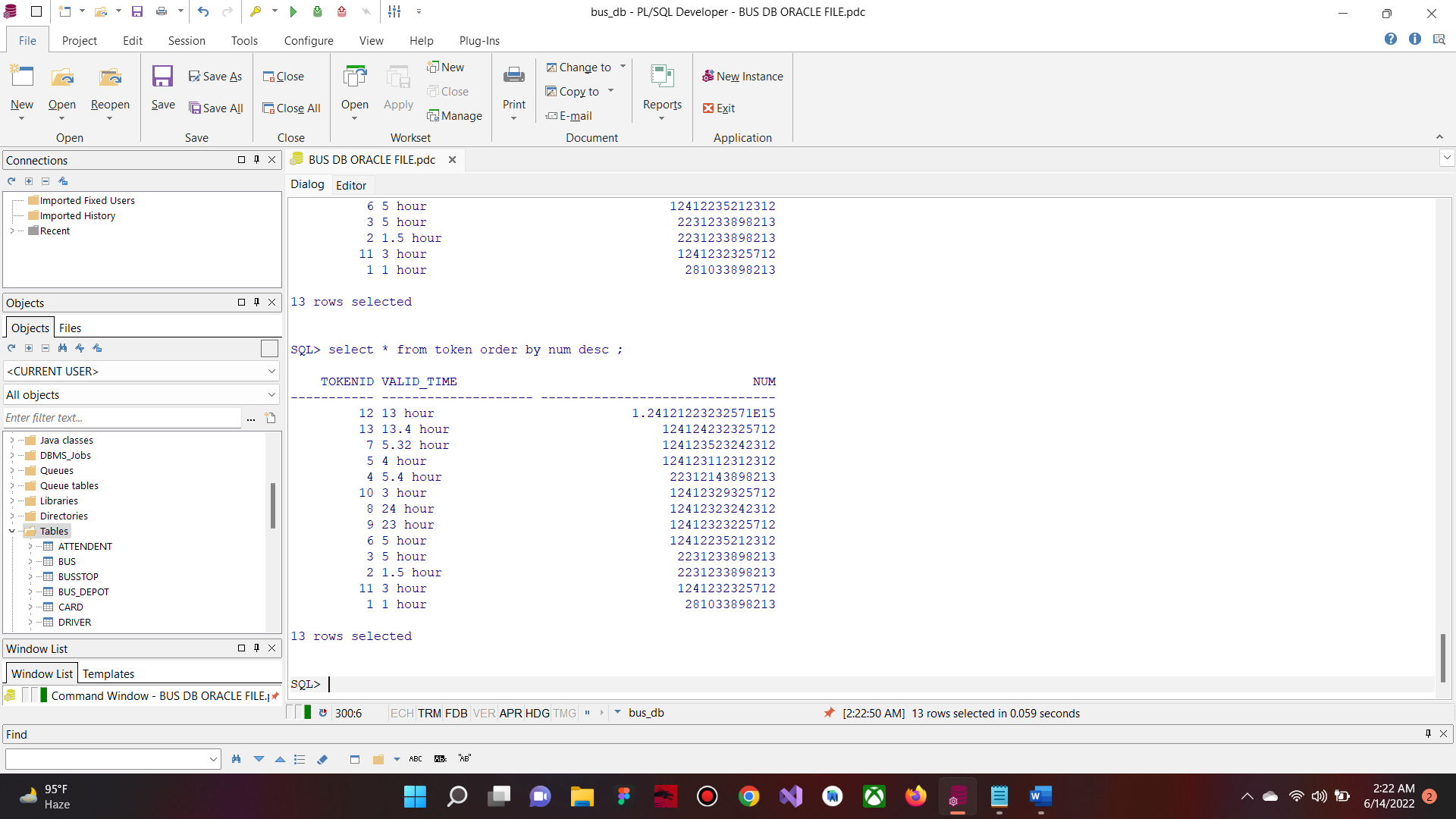
A picture containing engineering drawing

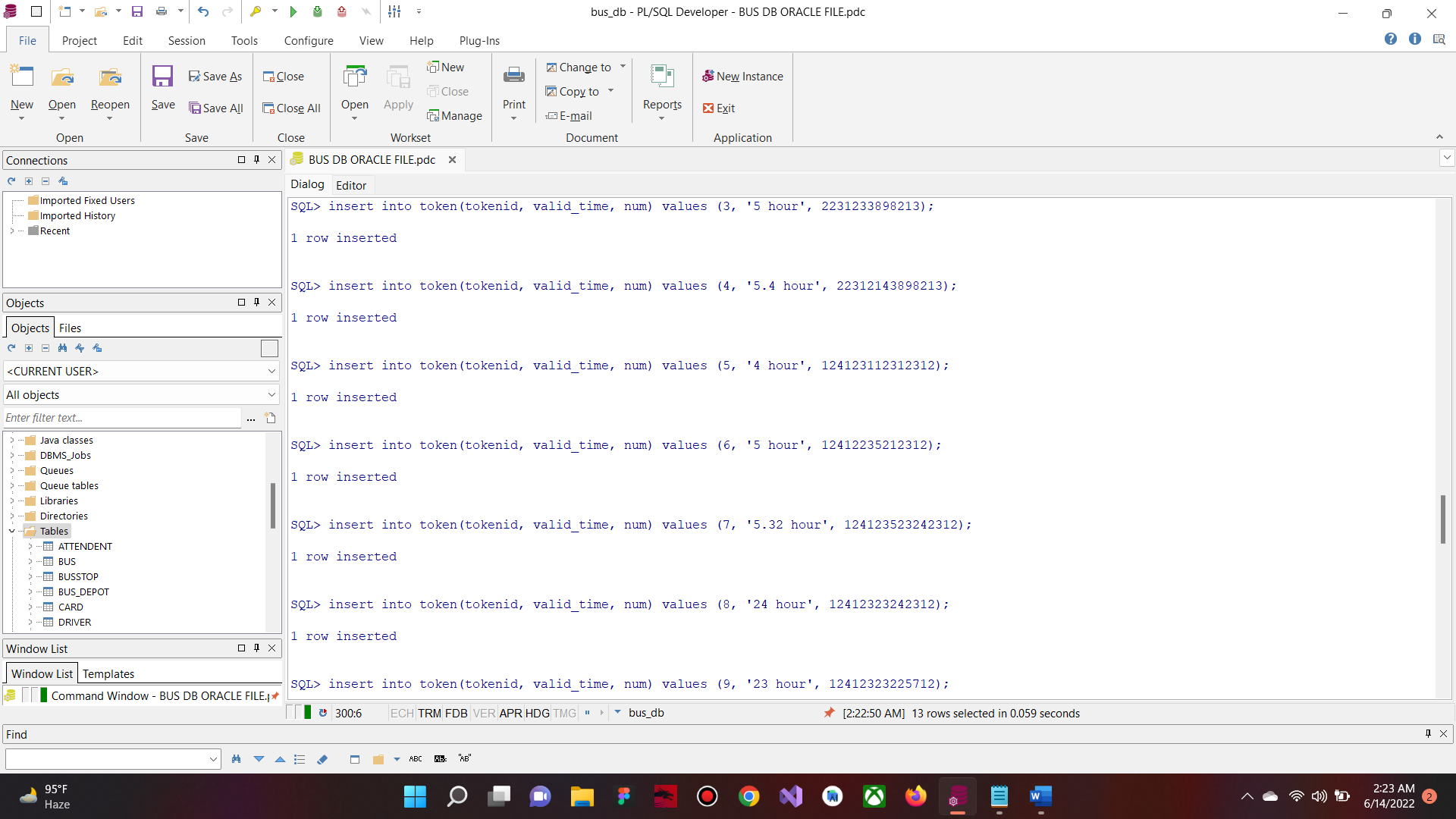
Description automatically generated

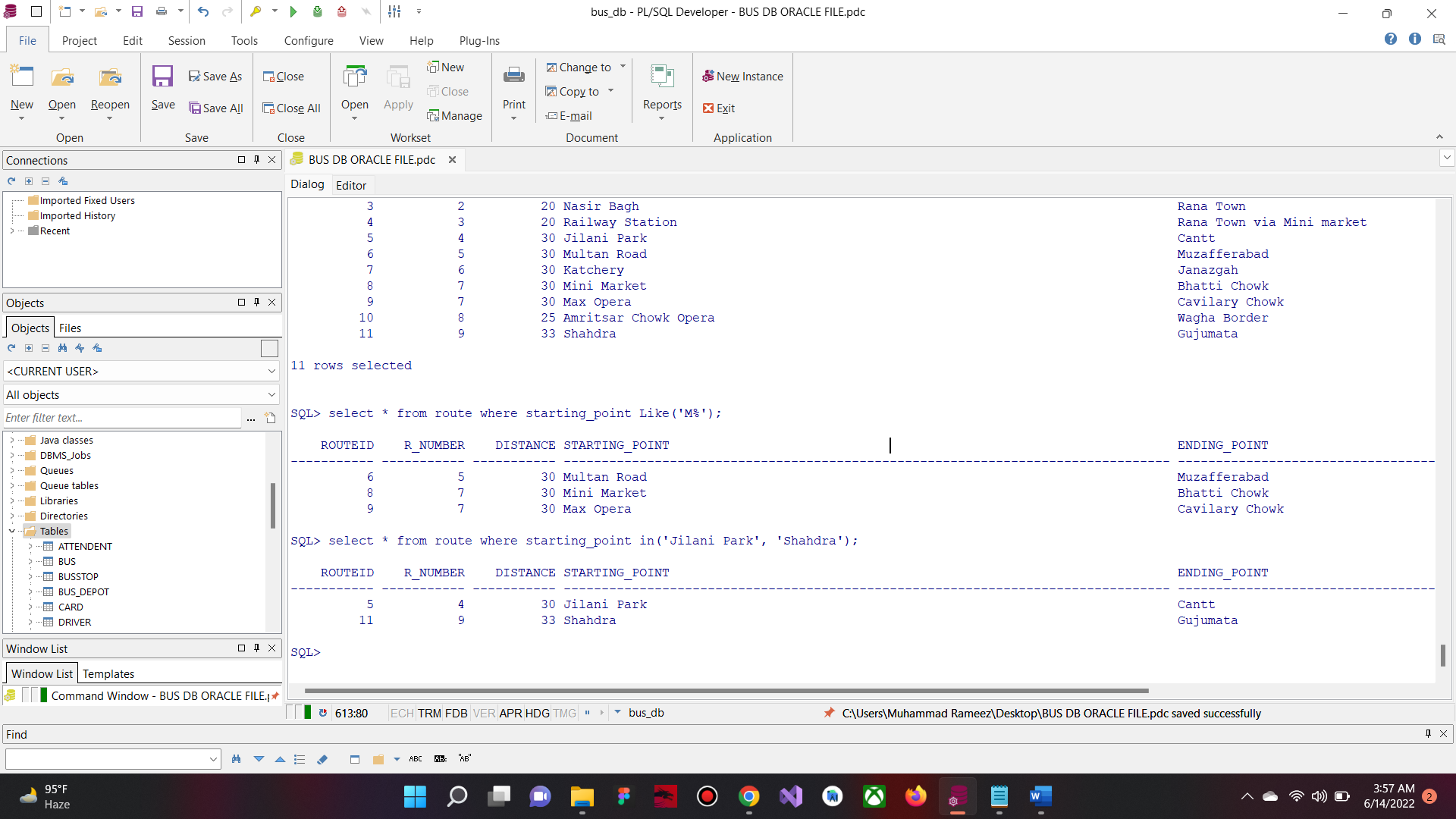
# Testing











\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_