جامعة الأمير سلطان

PRINCE SULTAN
UNIVERSITY

PSU.1998

Riyadh Bus Reservation System

Department of Computer & Information Sciences

CS340 Project deliverable 1

Sept 24th, 2023

Instructor: Dr. Roohi Jan

Section: 799

Prepared By:

Sarah AlJurbua 220410528

Noura AlAngari 220410015

Nouf Abduljabbar 220410775

Nour Fatoom 220410494

Table of Contents

Phase 1- Describing the Information System to Be Developed

1. Introduction

This document serves as the comprehensive foundation for the Riyadh Bus Reservation System project, including a series of interrelated phases. Our aim is to meticulously define the project's scope, clarify its purpose, explain the organizational context, outline essential data requirements, and specify functional and non-functional requirements, employing diverse data presentation techniques. IEEE and ACM professional codes of ethics will be embedded into the project. Begin by providing a clear overview of the Information System to be developed. We delve into the intricate task of ER/EER Modeling, which forms the foundation for the system's data structure. Then, we focus on Designing the Database and Normalization to ensure data integrity and efficiency. As the project transitions from design to implementation, we proceed to create tables using Oracle 10g, transforming our conceptual model into a practical database system. Finally, we conclude the project by crafting queries using Oracle, enabling users to interact with the system.

1.1. Background

In earlier years, transportation from one destination to the other has been significantly difficult and time-consuming because of poor systems, opportunities, and poor system management. However, nowadays numerous improvements to the existing transportation infrastructure have been undertaken and improved. Among these interventions are tour planners, ticketing systems, travel guides, etc. Bus travel is a significant and expanding industry in many nations; nevertheless, the current manual process of making bus reservations is time-consuming and physically demanding. In order to make bus reservations easier for customers, a useful and resource-frugal mechanism must be developed.

1.2. Description of the Organization

The Riyadh Bus Reservation System (RBRS) is an innovative solution aimed at transforming bus transportation services in Riyadh and potentially catering to various public transportation authorities, bus companies, and other entities involved in managing or operating bus services within urban regions. RBRS stands as a technological innovation It replaces the old, complicated, and time-consuming booking system with a streamlined and user-friendly database platform for bus reservations, optimizing the reservation process for both passengers and transport service providers. Tailored for Riyadh city, the adaptable RBRS framework holds the potential for broader application, addressing the evolving needs and requirements of diverse organizations and government sectors.

1.3. Purpose and Scope of the IS

Its primary purpose is to improve all aspects of bus reservation, from consumer booking to operational management. The information system's basic functionality allows consumers to quickly and easily book bus tickets online, select their preferred routes, and choose seats, greatly streamlining the reservation process and lowering the time and effort needed. A better travel experience is also made possible by the system's real-time updates on bus schedules, which ensure customers are informed of any delays or modifications. The information system of RBRS is not merely a tool for booking bus tickets but a comprehensive solution that encompasses customer service, operational efficiency, environmental responsibility, and community engagement.

2. Riyadh Bus Reservation System Scenario

1) User Registration and Login:

A **user** visits the Riyadh City Bus Reservation System website or mobile app and creates an account by providing their personal information, such as **userFirstName, userLastName, Gender, dateOfBirth, mobile, password, and email.**

Once registered, the user can log in to their account using their credentials, ensuring secure access to the reservation system.

2) Bus Routes and Schedules:

The system displays the **Bus** information including **busNumber, capacity, manufacture, year_manufacture, Model,** and the available bus **routes** within Riyadh city, including their **route_name, departureLocation, arrivalLocation, distance, and estimatedDuration.**

Users can search for specific routes or browse through the list to find the desired bus service.

3) Payment:

After selecting a preferred bus route the system shall display **Payment** which includes **payment_amount, Payment_date, Payment_method, and seat_number.**

4) Profile Management:

Passengers can view their profile information, which includes **userFirstName, userLastName, mobile, email**, and a list of their past and upcoming bus ride reservations. Passengers have the option to edit their profile information, such as updating their contact details or changing their password.

5) Bus Ride Cancellation:

If a passenger initiates a bus ride reservation but fails to complete the payment, the system allows them to cancel the reservation. The system releases the reserved seats for other passengers to book.

6) Payment Verification:

Passengers are prompted to enter their payment information, which may include credit card details or other payment methods. The system securely processes the payment and verifies its success. Upon successful payment verification, the system sends a payment confirmation message to the passenger's email address or mobile number.

7) Bus Driver Assignment:

A driver logs into their account and views their profile, which includes **driverFirstName, driverLastName, licenseNumber**, and a list of upcoming bus ride assignments. The system assigns bus rides to the driver based on their availability and assigns a unique **rideID** to each assignment.

8) Bus Ride History:

Passengers can view their complete bus ride history, including past reservations and completed journeys. The history includes details such as **route_name, departureLocation, arrivalLocation, and estimatedDuration.**

3. Data requirements of the Riyadh Bus Reservation System

3.1. Entity Definition Table

Entity name	Identifier	Attribute	Description
User	User_ID	Name (FName, LName) Gender dateOfBirth mobile password email	Represents a user of the Bus reservation system. User is considered a super-entity which can be of two types, driver, and passenger.

Bus	Bus_ID	busNumber capacity manufacture year_manufacture Model	Represents a bus that operates within the system. Buses have unique identifiers, bus numbers, maximum seating capacity, manufacture, year manufacture, and model.
Route	Route_ID	route_name departureLocation arrivalLocation distance estimatedDuration	**Source & destination:** the starting point of the route and the destination point of the route. **Distance:** the distance between the source and the destination in kilometers. **Duration:** the estimated time of the journey (in hours, mins.. etc.)
Schedule	Schedule_ID	departureTime arrivalTime	Represents the schedule of a bus schedule including the departure time and arrival time for each bus on a specific route.
Payment	Payment_ID	payment_amount Payment_date Payment_method	Represents a payment made by a user for a reservation. Payments have unique identifiers and a link to a specific reservation.
Driver	Driver_ID	license_number hire_date	Represents a bus driver employed by an agency. Drivers have unique identifiers, name, and driver's license numbers.
Passenger	passenger_ID	-	Represents a passenger making a reservation. Passengers have unique identifiers, names, and ages.
Ticket	Ticket_ID	BookingDate TicketPrice SeatNumber	Represents the booking date, the price of the ticket, and the seat number.

3.2. Relation Definition Table

Relation Name	Type	Attributes	Entities	Description
Has Bus Schedule	Binary one-to-many-many TP: TP	-	BusSchedule Bus	This table holds information about buses available for reservation. Each bus is uniquely identified by a bus_ID. The busNumber attribute represents the unique identification number of the bus. The capacity attribute specifies the maximum number of passengers that the bus can accommodate.
Buy Tickets	Ternary One-to-many-many TP: TP	BookingDate	Ticket Passenger BusSchedule	Represents the relationship where a user makes a reservation for a specific schedule. A user can make multiple reservations, and a reservation is associated with a single user, schedule, passenger, and payment.
Follows	one-to-many TP: TP	-	Bus Route	Represents the relationship where a bus follows a specific route. A route can be followed by multiple buses, but a bus follows a single route.
Make Payment	One-to-one TP: TP	-	payment ticket	This table represents the payment details associated with the bus reservation and the bus reservation system. Each payment is uniquely identified by payment_ID. The reservation_ID attribute references the reservation entity, indicating that is your vision for

				which the payment is made. The amount is attributed to the payment amount.
Assigned to	one-to-one TP: PT	**-**	Bus driver	Represents the relationship where a driver is assigned to a bus. A bus can have a single driver assigned to it, and a driver can be assigned to a single bus.

3.3. Attribute Definition Table

User Table:

Attribute Name	Description
userFirstName	Unique identifier for a user
userLastName	User’s username
gender	User’s gender
email	User’s email
password	User’s password
mobile	User’s mobile number

Bus Table:

Attribute Name	Description
capacity	Maximum capacity of the bus
busNumber	Bus number
manufacturer	Bus manufacturer
yearManufacturer	Bus year of manufacture
model	Bus model name/ number

Route table:

Attribute Name	Description
route_name	The name of the specific route
departureLocation	The departure station or port
arrivalLocation	The arrival station
Distance	The distance between the stations
estimatedDuration	The time duration for the trip

Payment Table:

Attribute Name	**Description**
payment_amount	Date of the payment
Payment_date	Payment amount
Payment_method	payment method (ex: cash, card …)

Driver Table:

Attribute Name	**Description**
hire_date	Driver's hire date
license_number	Driver's license number

4. Functional and Non-Functional System Requirements

4.1. Functional System Requirements

Passenger:

-The system shall ask the passenger to create an account.
-The system shall ask the passenger to enter information for account creation.
-The system shall save the passengers' information.

-The system shall verify the passengers' accounts.
-The system shall allow the passenger to display their profile.
-The system shall allow passengers to delete their profile.
-The system shall allow passengers to edit their profile.
-The system shall allow the passenger to view available bus rides.
-The system shall allow passengers to reserve bus rides.
-The system shall verify successful reservations.
-The system shall ask the passenger for the desired payment method.
-The system shall verify the payment.
-The system shall send verification messages after successful payment.
-The system shall allow the passenger to cancel a bus ride reservation if the payment isn't complete.
-The system shall allow the passenger to view their recent (1 week) bus ride history.

Driver:

-The system shall ask the driver to create an account.
-The system shall ask the driver to enter information for account creation.
-The system shall save the drivers' information.
-The system shall verify the driver's account.
-The system shall allow the driver to display their profile.
-The system shall allow drivers to delete their profiles.
-The system shall allow drivers to edit their profiles.
-The system shall assign bus rides for the driver.
-The system shall inform the driver about the bus ride duration.
-The system shall allow the driver to view bus ride information.

4.2. Non-functional System Requirements

1. Performance Requirements:

1.1. The system shall allow the passenger to access the system within 10 seconds after successfully logging in.
1.2. The system shall send the passengers' verification messages in no more than 7 seconds upon approved payment.
1.3. The system shall accommodate a minimum of 500 users concurrently.

2. Safety Requirements:

2.1. The system shall undergo various testing prior to release.
2.2. The system shall upgrade its database continuously every 12 hours.

3. Reliability Requirements:

3.1. The systems' failure rate should not be more than 1 failure per 150 operations.

3.2. The system downtime should be 20 minutes per month maximum.

4. Usability Requirements:

4.1. The system shall allow the passenger to book a bus ride within 3 minutes.

4.2. The system shall allow users to view available bus rides in no more than 7 seconds.

5. Security Requirements:

5.1. The system shall notify the passenger if a login attempt occurs.

5.2. The system shall send a verification if payment is complete.

5.3. The system shall block the passenger's account after 3 consecutive failed login attempts.

5.4. The system shall have specific restrictions on the passenger's password.

6. Data Integrity Requirements:

6.1. The system shall constantly update the information on the system if any changes have been made.

6.2. The system shall perform a backup for every updated database.

7. Availability Requirements:

7.1. The system shall be available 95% of the time.

7.2. The system shall be available 97% of the time from 6 AM to 6 PM.

8. Constraints:

8.1. The system shall provide necessary information in both Arabic and English form.

8.2. The passenger's password should not exceed 12 characters.

8.3. The passenger's password should not be less than 8 characters.

9. Portability:

9.1. Modifying the iOS version of the system to run on Android and Windows should require changing 15% of the source code maximum.

5. Relevant clauses from the IEEE and ACM Professional Code of Ethics

5.1. IEEE Code of Ethics

- To accept the responsibility for making the right decision for the welfare of the public.
- To be realistic and honest with the available data.
- To improve the understanding of the technology and the appropriate system.

- To reject bribery in all possible forms.
- To avoid injuring the property and employment by harmful actions.
- To treat all people equally regardless of their race, religion, and gender.
- To assist and help colleagues and workers in the system development.

5.2. ACM Code of Ethics

5.2.1. General Ethical Principles

1.1. Contribute to society and human well-being.
1.2. Avoid harm.
1.3. Be honest and trustworthy.
1.4. Be fair and take the right action.
1.5. Respect the required work to produce creative works and new ideas.
1.6. Respect the privacy of the work.
1.7. Honor confidentiality.

5.2.2. Professional Responsilities

2.1. strive to achieve high quality in producing professional work.
2.2. Maintain high standards of in competence conducting ethical practices.
2.3. Respect and follow existing rules in your work.
2.4. Accept and provide the appropriate review in the profession.
2.6. Perform work only in the area of competence.

5.2.3. Professional Leadership Principles

3.1. Ensure that the public good is the central concern all of the time.
3.2. Encourage acceptance and evaluate fulfillment of social responsibility by all members of the organization.
3.3. Manage resources to increase the quality of working life.
3.5. Create opportunities for members of the organization.
3.6. Use care when changing and modifying the system.

5.2.4. Compliance with the Code

4.1. Promote and respect the principles of the code.
4.2. Treat any violation of the code as inconsistent with membership in the ACM.

6. Teamwork Distribution and Strategy

6.1. Strategy

One of the most important strategies for demonstrating teamwork is to establish clear roles and responsibilities for each team member. This ensures that everyone understands their specific tasks and areas of expertise, and it promotes a sense of accountability within the team. According to phase one, each participant was responsible for her own task.

Each team member actively shared their information, ideas, and feedback with one another. This was done through regular meetings and instant messaging, each participant tracked their progress and provided real-time updates.

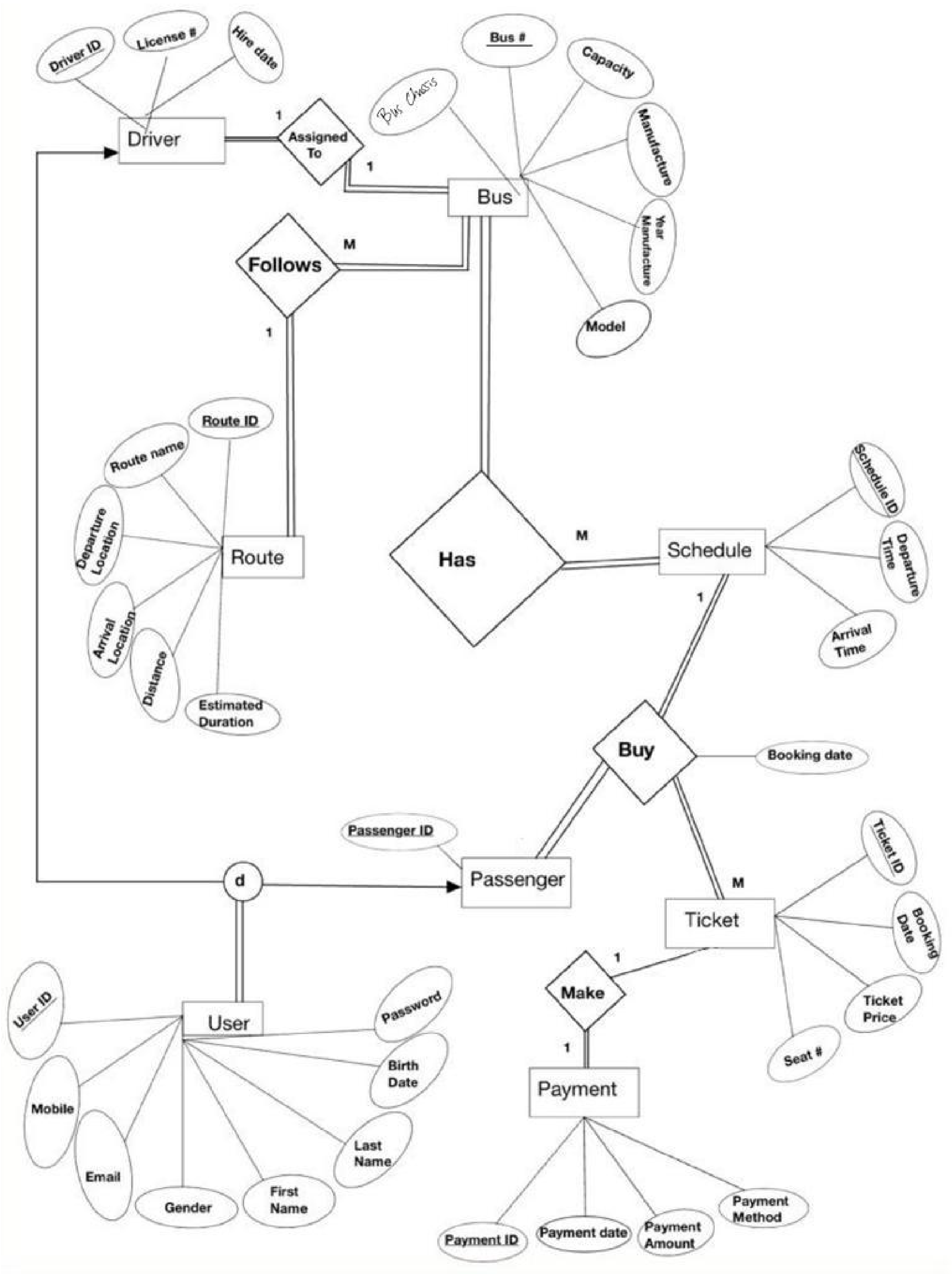
Active participation is essential for demonstrating teamwork. In our project, all team members were actively engaged in discussions, contributing ideas, providing information, and offering constructive feedback and advice. Throughout the project, team members assist and motivate one another.

6.2. Individual Contribution

Name	**Contribution**
Sarah AlJurbua	1-Description of the organization. 2-description of the purpose & scope of the information system. 3- Data scenarios
Noura AlAngari	1-Functional system requirement. 2-Nonfunctional system requirements.
Nouf Alabduljabbar	1-Data requirements. 2- Contribution of team members.
Nour Fatoom	1-Identify relevant clauses from IEEE and ACM.

Phase 2-
ER/EER Modeling

7. EER Diagram



8. Entity Definition Table

Entity name	Identifier	Attribute	Description
User	User_ID	userFirstName userLastName Gender dateOfBirth mobile password email	Represents a user of the Bus reservation system. User is considered a super-entity which can be of two types, driver, and passenger.
Bus	Bus_ID	busNumber capacity manufacture year_manufacture Model	Represents a bus that operates within the system. Buses have unique identifiers, bus numbers, maximum seating capacity, manufacture, year manufacture, and model.
Route	Route_ID	route_name departureLocation arrivalLocation distance estimatedDuration	**Source & destination:** the starting point of the route and the destination point of the route. **Distance:** the distance between the source and the destination in kilometers. **Duration:** the estimated time of the journey (in hours, mins.. etc.)
Schedule	Schedule_ID	departureTime arrivalTime	Represents the schedule of a bus schedule including the departure time and arrival time for each bus on a specific route.
Payment	Payment_ID	payment_amount Payment_date Payment_method	Represents a payment made by a user for a reservation. Payments have unique identifiers and a link to a specific reservation.

Driver	Driver_ID	license_number hire_date	Represents a bus driver employed by an agency. Drivers have unique identifiers, names, and driver’s license numbers.
Passenger	passenger_ID	-	Represents a passenger making a reservation. Passengers have unique identifiers, names, and ages.
Ticket	Ticket_ID	BookingDate TicketPrice SeatNumber	Represents the booking date, the price of the ticket, and the seat number.

8.1. Relation Definition Table

Relation Name	Type	Attributes	Entities	Description
Has Bus Schedule	Ternary one-to-many-many TP: TP	-	BusSchedule Bus Route	This table holds information about buses available for reservation. Each bus is uniquely identified by a bus_ID. The busNumber attribute represents the unique identification number of the bus. The capacity attribute specifies the maximum number of passengers that the bus can accommodate.
Buy Tickets	Ternary One-to-many-many TP: TP	BookingDate Num_seat	Ticket Passenger BusSchedule	Represents the relationship where a user makes a reservation for a specific schedule. A user can make multiple reservations, and a reservation is associated with a single user, schedule, passenger, and payment.
Follows	one-to-many	**-**	Bus	Represents the relationship

	TP: TP		Route	where a bus follows a specific route. A route can be followed by multiple buses, but a bus follows a single route.
Make Payment	one-to-many TP: TP	-	payment ticket	This table represents the payment details associated with the bus reservation and the bus reservation system. Each payment is uniquely identified by payment_ID. The reservation_ID attribute references the reservation entity, indicating that is your vision for which the payment is made. The amount is attributed to the payment amount.
Assigned to	one-to-one TP: PT	**-**	Bus driver	Represents the relationship where a driver is assigned to a bus. A bus can have a single driver assigned to it, and a driver can be assigned to a single bus.

9. Business Rules

-The passenger shall be able to access the website or application with their username and password.
- The driver shall be able to access the website or application with their username and password.
-The system shall only allow passengers with an account to reserve bus rides.
-The passenger shall be able to select their desired payment method.
-Each bus ride shall have at least one driver.
-Each bus ride shall have at least one passenger.
-The system shall save every information entered by the passenger or driver to their associated database.
-The system shall synchronize the data to the server within 8 hours.
-The system shall update its database every 12 hours.

-The system shall provide the necessary description of the bus ride for the driver and passenger.
-The system shall provide Arabic and English language options.

10. Relevant clauses from the IEEE and ACM Professional Code of Ethics

10.1. IEEE Code of Ethics

- To hold paramount the safety, health, and welfare of the public, to strive to comply with ethical design and sustainable development practices.
- To protect the privacy of others, and to promptly disclose factors that might endanger the public or the environment.
- To maintain and improve our technical competence and to undertake technological tasks for others only if qualified by training or experience, or after full disclosure of pertinent limitations.
- To uphold the highest standards of integrity, responsible behavior, and ethical conduct in professional activities.
- To treat all persons fairly and with respect, and to not engage in discrimination based on characteristics such as race, religion, gender, disability, age, or national origin.
- To support colleagues and co-workers in following this code of ethics, to strive to ensure the code is upheld, and to not retaliate against individuals reporting a violation.

10.2. ACM Code of Ethics

10.2.1. General Ethical Principles

1.1 Make a contribution to society and the welfare of people.
1.2 Respect other people's privacy
1.3 Be reliable and sincere.
1.4 Be fair and make an effort to avoid discrimination.
1.5 Protect others from harm.

10.2.2. Professional Responsibilities

2.1 Aim for the highest levels of excellence, efficacy, and dignity.
2.2 Obtain and uphold professional competence.
2.3 Know and abide by the rules and regulations that govern the system.
2.4 Give due credit for any creative works.
2.5 Protect the privacy of important information.

2.6 Respect commitments, agreements, and contracts.
2.7 Increase knowledge of computing and its effects among the people

10.2.3. Professional Leadership Principles

3.1 Encourage public understanding and awareness of computing.
3.2 Develop and support laws that encourage the wise use of technology.
3.4 Only use resources in a way that serves the general good.
3.4 Educate people about responsible computing and encourage their adoption.

10.2.4. Compliance with the Code

4.1 Learn the Code thoroughly, read and comprehend the ACM Code of Ethics. Make sure you understand the rules and regulations it lays out.
4.2 Secure user data and prevent it from unauthorized access by putting in place the necessary safeguards.

11. Teamwork Distribution and Strategy

11.1. Strategy

Determining each team member's precise tasks and responsibilities is one of the most crucial tactics for exhibiting collaboration. This builds a sense of accountability among team members and guarantees that everyone is aware of their particular responsibilities and areas of expertise. Phase one stipulated that each participant was in charge of her own assignment.

Each team member engaged in active communication with the others, exchanging ideas, information, and criticism. Regular meetings and instant messaging were used for this; each participant kept track of their development and gave real-time updates.

To exhibit collaboration, one must actively participate. Every team member actively participated in our project by bringing ideas to the table, exchanging information, and giving helpful criticism and guidance. During the project, team members support and encourage one another.

11.2. Individual Contribution

Name	Contribution
Sarah AlJurbua	EER diagram
Noura AlAngari	Business Rules
Nouf Alabduljabbar	Relevant clauses from the IEEE and ACM Professional Code of Ethics
Nour Fatoom	EER Diagram

Riyadh Bus Reservation System

Department of Computer & Information Sciences

CS340 Project deliverable 3

Oct 25th, 2023

Instructor: Dr. Roohi Jan

Section: 799

Prepared By:

Sarah AlJurbua 220410528

Noura AlAngari 220410015

Nouf Abduljabbar 220410775

Nour Fatoom 220410494

Phase 3
Designing Database and Normalization

12. Data Dictionary

User

Column name	Key type	Constraints	FK table	FK column	Data type	Length
UserID	PK	Not null, unique			Varchar	9
F_Name		Not null			Varchar	15
L_Name		Not null			Varchar	15
Gender		Not null			Char	1
Mobile		Not null, unique			Char	10
Email		Not null			Varchar	50
Password		Not null			Varchar	20
DOB		Not null			dd-mm-yyyy	10

Driver

Column name	Key type	Constraints	FK table	FK column	Data type	Length
DriverID	PK, FK	Not null, unique	User	UserID	Varchar	9
Licenses#		Not null, unique			Varchar	9
Hire_date		Not null			Date	dd-mm-yyyy

Passenger

Column name	Key type	Constraints	FK table	FK column	Data type	Length
PassengerID	PK, FK	Not null, unique	User	UserID	Varchar	9
Seat#		Not null, unique			Number	9
BookingDate		Not null			Date	dd-mm-yyyy

Schedule

Column name	Key type	Constraints	FK table	FK column	Data type	Length
ScheduleID	PK	Not null, unique			Varchar	9
Dep_time		Not null			Varchar	10
Arr_time		Not null			Varchar	10

Bus

Column name	Key type	Constraints	FK table	FK column	Data type	Length
Bus ID	PK	Not null, unique			Varchar	9
Bus number		Not null, unique			Number	4
Capacity		Not null			Varchar	10
Manufacture		Not null			Varchar	10
Year manufacture		Not null			Date	dd-mm-yy yy
Model		Not null			Varchar	10

Route

Column name	Key type	Constraints	FK table	FK column	Data type	Length
Route ID	PK	Not null. unique			Varchar	9
Route Name		Not null, Unique			Varchar	15
Departure location		Not null			Varchar	20
Arrival location		Not null			Varchar	20
Distance		Not null			Varchar	20
Estimated duration		Not null			Varchar	9

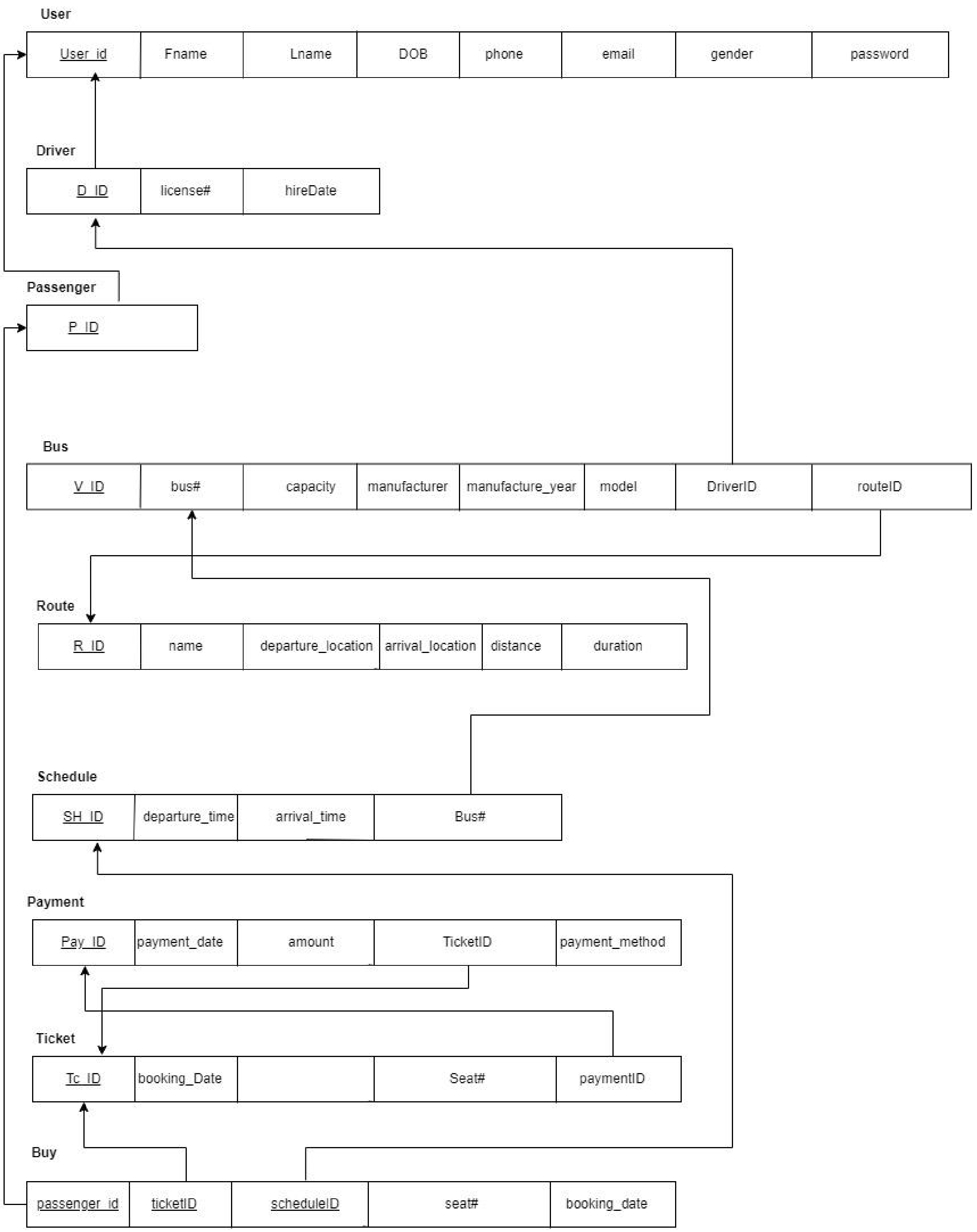
Ticket

Column name	Key type	Constraints	FK table	FK column	Data type	Length
Ticket ID	PK	Not null. unique			Varchar	9
Booking Date		Not null	Passenger	Booking Date	Date	dd-mm-yy yy
Ticket price		Not null			Varchar	9
Seat number		Not null, Unique	Passenger	seat #	Number	9

Payment

Column name	Key type	Constraints	FK table	FK column	Data type	Length
Payment ID	PK	Not null. unique			Varchar	10
Payment amount	FK	Not null	Ticket	Ticket Price	Varchar	9
Payment date		Not null			Date	dd-mm-yy yy
Payment method		Not null			Varchar	15

13. Relational Model



User

User_id	Fname	Lname	DOB	phone	email	gender	password

Driver

D_ID	license#	hireDate

Passenger

P_ID

Bus

V_ID	bus#	capacity	manufacturer	manufacture_year	model	DriverID	routeID

Route

R_ID	name	departure_location	arrival_location	distance	duration

Schedule

SH_ID	departure_time	arrival_time	Bus#

Payment

Pay_ID	payment_date	amount	TicketID	payment_method

Ticket

Tc_ID	booking_Date		Seat#	paymentID

Buy

passenger_id	ticketID	scheduleID	seat#	booking_date

14. Relevant clauses from the IEEE and ACM Professional Code of Ethics

14.1. IEEE Code of Ethics

- To accept the responsibility for making the right decision for the welfare of the public.
- To be realistic and honest with the available data.
- To improve the understanding of the technology and the appropriate system.
- To reject bribery in all possible forms.
- To avoid injuring the property and employment by harmful actions.
- To treat all people equally regardless of their race, religion, and gender.
- To assist and help colleagues and workers in the system development.

14.2. ACM Code of Ethics

14.2.1. General Ethical Principles

1.1. Contribute to society and human well-being.
1.2. Avoid harm.
1.3. Be honest and trustworthy.
1.4. Be fair and take the right action.
1.5. Respect the required work to produce creative works and new ideas.
1.6. Respect the privacy of the work.
1.7. Honor confidentiality.

14.2.2. Professional Responsibilities

2.1. strive to achieve high quality in producing professional work.
2.2. Maintain high standards of competence in conducting ethical practices.
2.3. Respect and follow existing rules in your work.
2.4. Accept and provide the appropriate review in the profession.
2.6. Perform work only in the area of competence.

14.2.3. Professional Leadership Principles

3.1. Ensure that the public good is the central concern all of the time.

3.2. Encourage acceptance and evaluate fulfillment of social responsibility by all members of the organization.
3.3. Manage resources to increase the quality of working life.
3.5. Create opportunities for members of the organization.
3.6. Use care when changing and modifying the system.

14.2.4. Compliance with the Code

4.1. Promote and respect the principles of the code.
4.2. Treat any violation of the code as inconsistent with membership in the ACM.

15. Teamwork Distribution and Strategy

15.1. Strategy

Setting defined tasks and duties for each team member is one of the most crucial tactics for exhibiting collaboration. As a result, the team feels more accountable and everyone is guaranteed to grasp their own responsibilities and areas of expertise. In the first phase, every participant was in charge of her own assignment.

Every member of the team actively communicated with the others, offering ideas, information, and criticism. Each participant recorded their progress and gave real-time updates, and this was accomplished through frequent meetings and instant chat.

To demonstrate collaboration, involvement must be active. Every member of the team actively participated in our project, bringing ideas, knowledge, and advice to the table as well as constructive criticism. Team members support and encourage one another during the project.

15.2. Individual Contribution

Name	**Contribution**
Sarah AlJurbua	Relevant clauses from the IEEE and ACM professional Code of Ethics
Noura AlAngari	Data Dictionary
Nouf Alabduljabbar	Normalized Relational Model
Nour Fatoom	Data Dictionary

Riyadh Bus Reservation System

Department of Computer & Information Sciences

CS340 Project deliverable 4

Instructor: Dr. Roohi Jan

Section: 799

Prepared By:

Sarah AlJurbua 220410528

Noura AlAngari 220410015

Nouf Abduljabbar 220410775

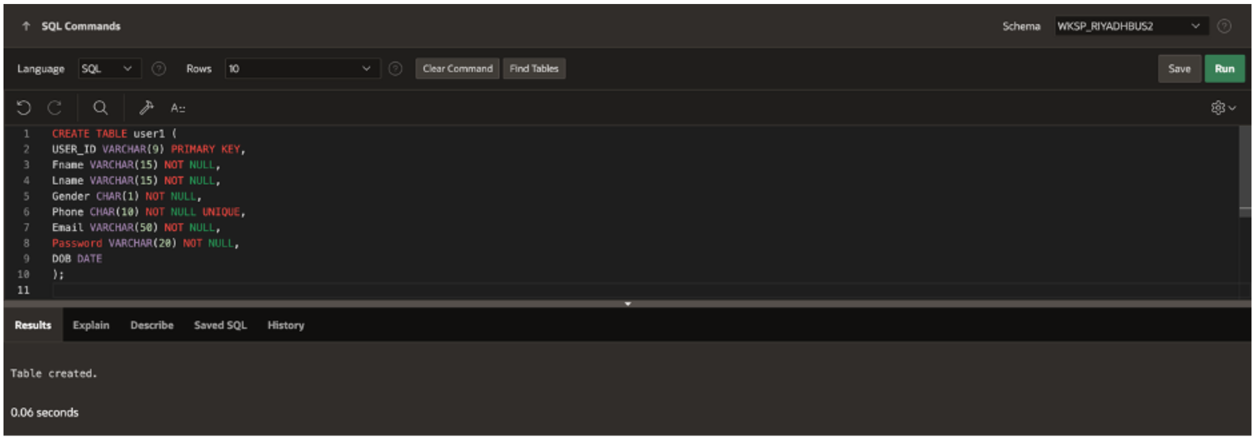
Nour Fatoom 220410494

Phase 4
Creating Tables Using Oracle 10g

16. Creating Tables

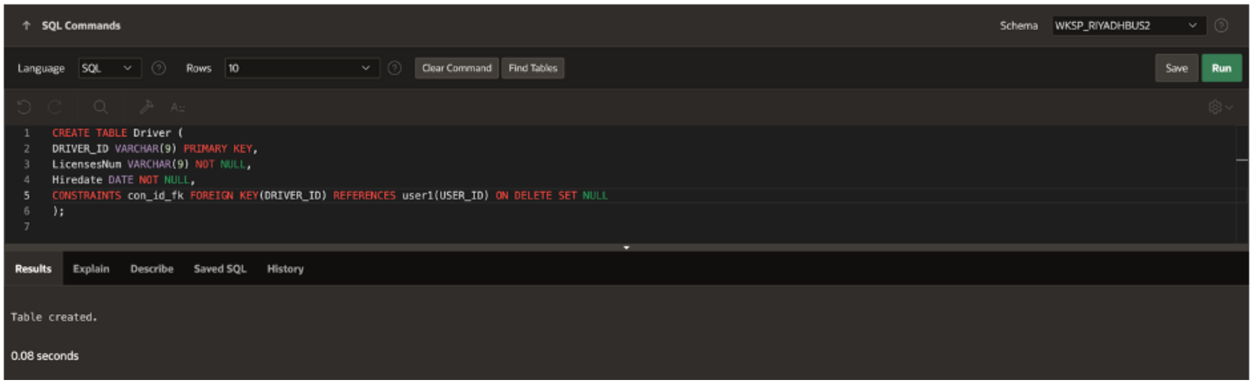
A new workspace has been created to ensure that tables don't exist before creation. Which included creating tables based on entities and inserting 10 records for each entity.

The SQL file is attached in Moodle along with the phase 4 submission.



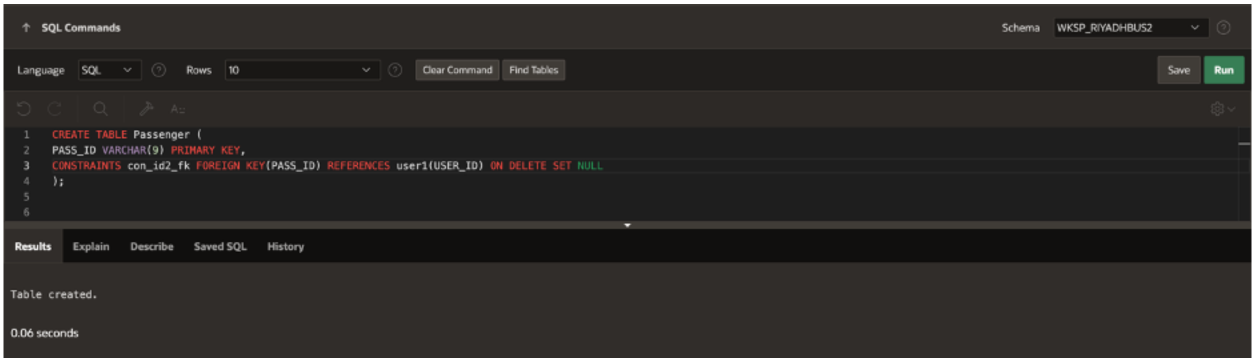
```
CREATE TABLE user1 (
USER_ID VARCHAR(9) PRIMARY KEY,
Fname VARCHAR(15) NOT NULL,
Lname VARCHAR(15) NOT NULL,
Gender CHAR(1) NOT NULL,
Phone CHAR(10) NOT NULL UNIQUE,
Email VARCHAR(50) NOT NULL,
Password VARCHAR(20) NOT NULL,
DOB DATE
);
```

```
Table created.

0.06 seconds
```



```
CREATE TABLE Driver (
DRIVER_ID VARCHAR(9) PRIMARY KEY,
LicensesNum VARCHAR(9) NOT NULL,
Hiredate DATE NOT NULL,
CONSTRAINTS con_id_fk FOREIGN KEY(DRIVER_ID) REFERENCES user1(USER_ID) ON DELETE SET NULL
);
```

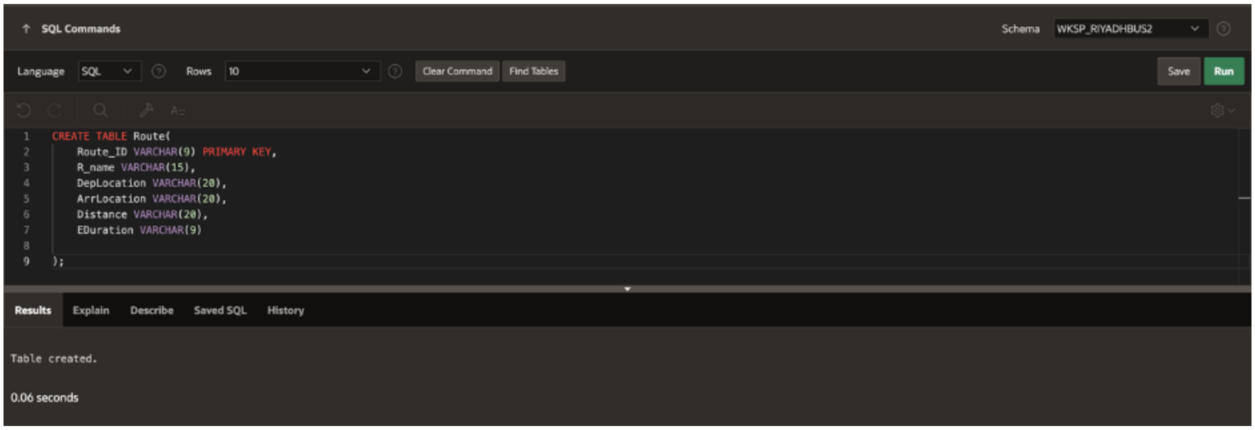
```
Table created.

0.08 seconds
```



```sql
CREATE TABLE Passenger (
PASS_ID VARCHAR(9) PRIMARY KEY,
CONSTRAINTS con_id2_fk FOREIGN KEY(PASS_ID) REFERENCES user1(USER_ID) ON DELETE SET NULL
);
```

```
Table created.
```

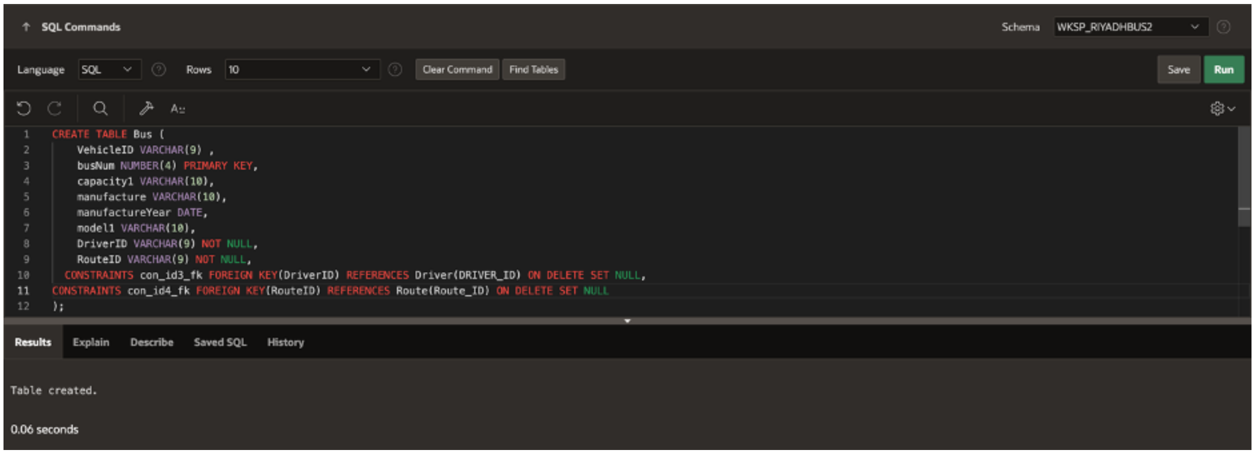
0.06 seconds



```sql
CREATE TABLE Route(
    Route_ID VARCHAR(9) PRIMARY KEY,
    R_name VARCHAR(15),
    DepLocation VARCHAR(20),
    ArrLocation VARCHAR(20),
    Distance VARCHAR(20),
    EDuration VARCHAR(9)

);
```

```
Table created.
```

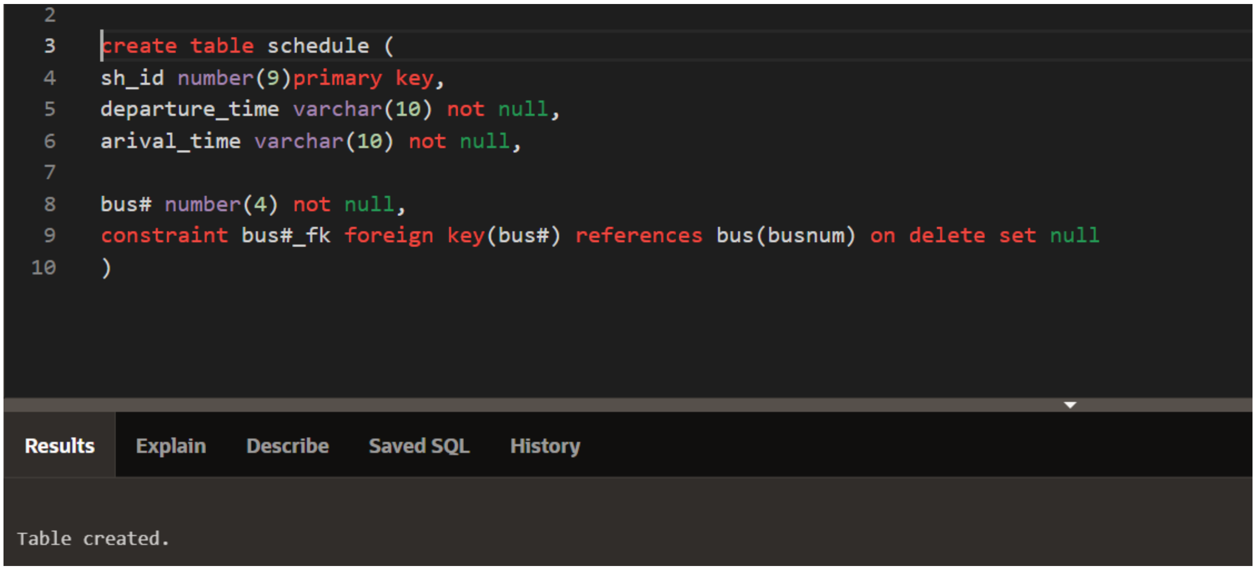
0.06 seconds



```sql
CREATE TABLE Bus (
    VehicleID VARCHAR(9) ,
    busNum NUMBER(4) PRIMARY KEY,
    capacity1 VARCHAR(10),
    manufacture VARCHAR(10),
    manufactureYear DATE,
    model1 VARCHAR(10),
    DriverID VARCHAR(9) NOT NULL,
    RouteID VARCHAR(9) NOT NULL,
  CONSTRAINTS con_id3_fk FOREIGN KEY(DriverID) REFERENCES Driver(DRIVER_ID) ON DELETE SET NULL,
CONSTRAINTS con_id4_fk FOREIGN KEY(RouteID) REFERENCES Route(Route_ID) ON DELETE SET NULL
);
```

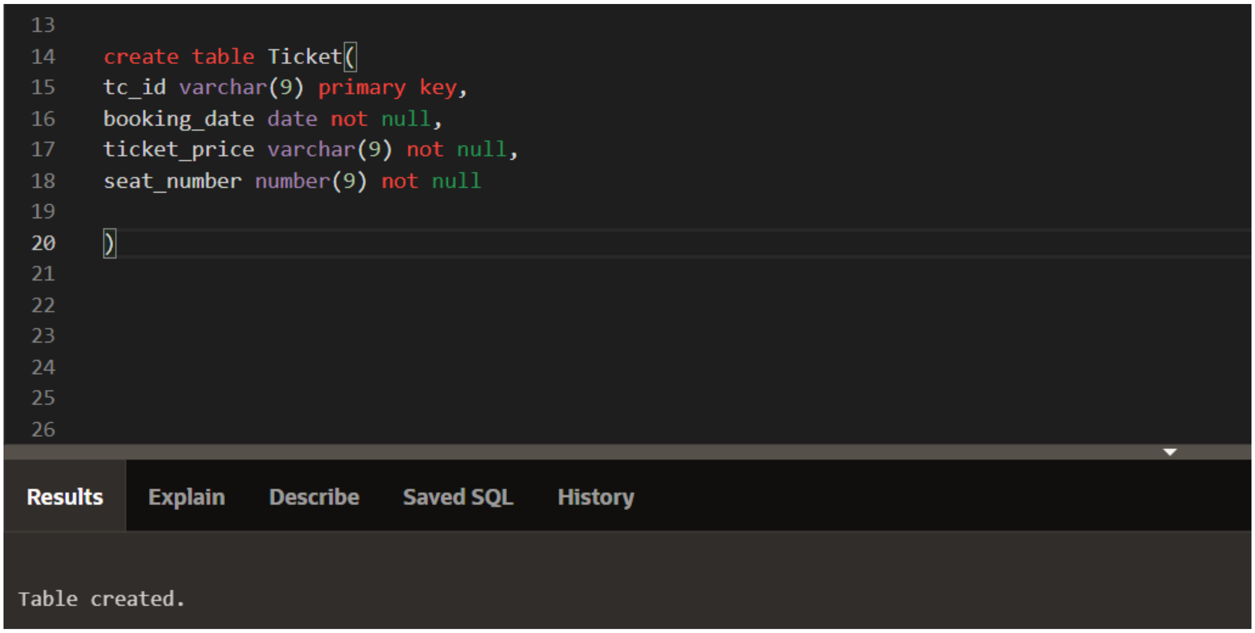
```
Table created.
```

0.06 seconds



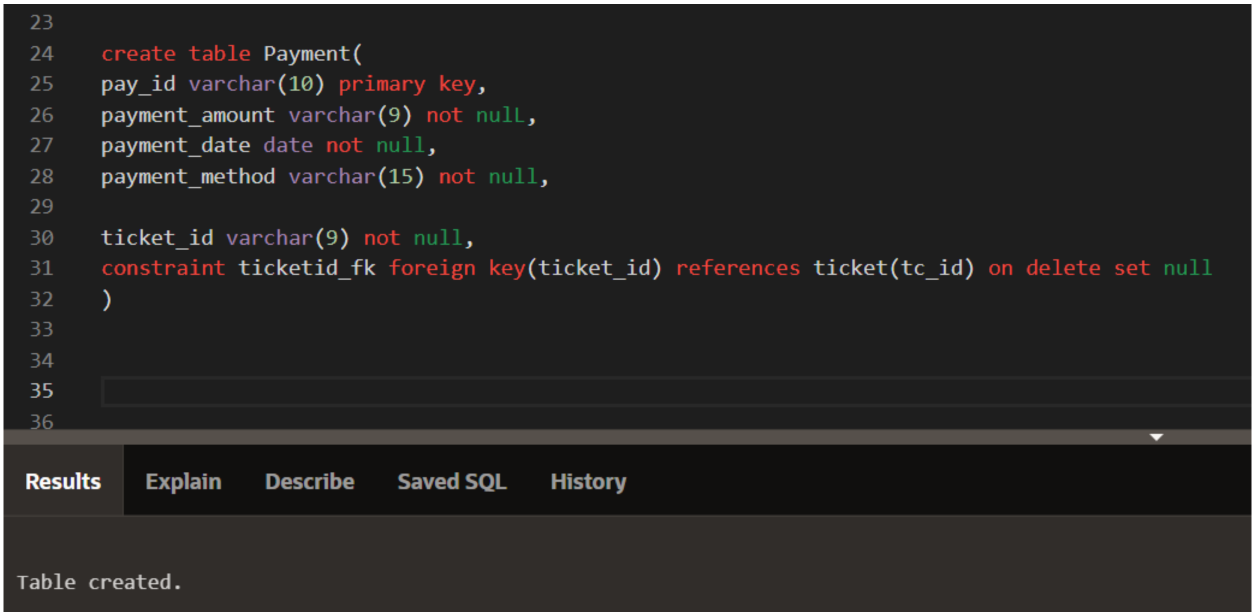
```
create table schedule (
sh_id number(9)primary key,
departure_time varchar(10) not null,
arival_time varchar(10) not null,

bus# number(4) not null,
constraint bus#_fk foreign key(bus#) references bus(busnum) on delete set null
)
```

```
Table created.
```



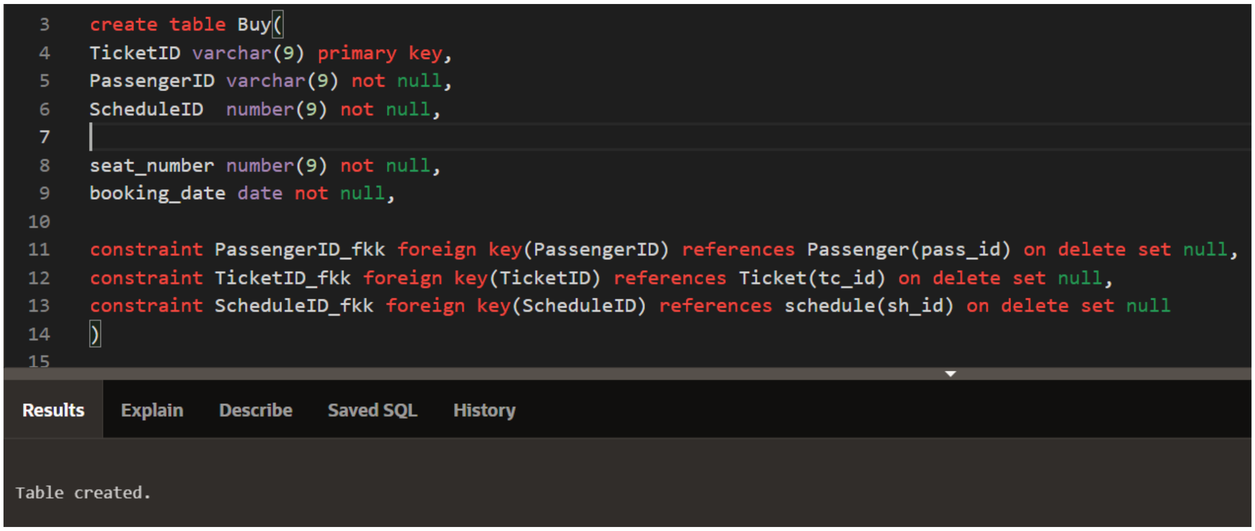
```
create table Ticket(
tc_id varchar(9) primary key,
booking_date date not null,
ticket_price varchar(9) not null,
seat_number number(9) not null

)
```

```
Table created.
```



```sql

create table Payment(
pay_id varchar(10) primary key,
payment_amount varchar(9) not nulL,
payment_date date not null,
payment_method varchar(15) not null,

ticket_id varchar(9) not null,
constraint ticketid_fk foreign key(ticket_id) references ticket(tc_id) on delete set null
)



```

```
Table created.
```



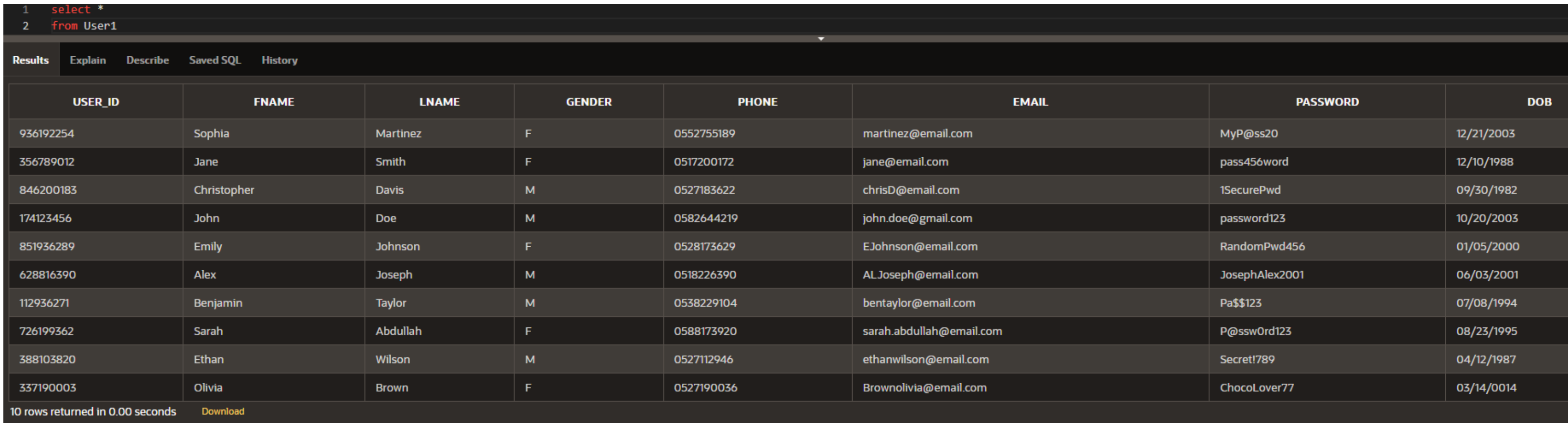
```sql
create table Buy(
TicketID varchar(9) primary key,
PassengerID varchar(9) not null,
ScheduleID  number(9) not null,

seat_number number(9) not null,
booking_date date not null,

constraint PassengerID_fkk foreign key(PassengerID) references Passenger(pass_id) on delete set null,
constraint TicketID_fkk foreign key(TicketID) references Ticket(tc_id) on delete set null,
constraint ScheduleID_fkk foreign key(ScheduleID) references schedule(sh_id) on delete set null
)
```

```
Table created.
```

17. Inserting data

1) **Inserting 10 records in User1 table:**

```
INSERT INTO user1 VALUES ('174123456', 'John', 'Doe', 'M', '0582644219',
'john.doe@email.com', 'password123', TO_DATE(‘20-OCT-2003’,'DD-MON-YYYY'));

INSERT INTO user1 VALUES ('356789012', 'Jane', 'Smith', 'F', '0517200172', 'jane@email.com',
'pass456word', TO_DATE('10-12-1988','DD-MON-YYYY'));

INSERT INTO user1 VALUES ('726199362', 'Sarah', 'Abdullah', 'F', '0588173920',
'sarah.abdullah@email.com','P@ssw0rd123',TO_DATE('23-AUG-1995','DD-MON-YYYY));

INSERT INTO user1 VALUES ('388103820', 'Ethan', 'Wilson', 'M', '0527112946',
'ethanwilson@email.com', 'Secret!789', TO_DATE('12-APR-1987','DD-MON-YYYY'));

INSERT INTO user1 VALUES ('851936289', 'Emily', 'Johnson', 'F', '0528173629',
'EJohnson@email.com', 'RandomPwd456', TO_DATE('05-JAN-2000','DD-MON-YYYY'));

INSERT INTO user1 VALUES ('846200183', 'Christopher', 'Davis', 'M', '0527183622',
'chrisD@email.com', '1SecurePwd', TO_DATE('30-SEP-1982','DD-MON-YYYY'));

INSERT INTO user1 VALUES ('112936271', 'Benjamin', 'Taylor', 'M', '0538229104',
'bentaylor@email.com', 'Pa$$123', TO_DATE('08-JUL-1994','DD-MON-YYYY'));

INSERT INTO user1 VALUES ('337190003', 'Olivia', 'Brown', 'F', '0527190036',
'Brownolivia@email.com', 'ChocoLover77', TO_DATE('14-MAR-14','DD-MON-YYYY'));

INSERT INTO user1 VALUES ('936192254', 'Sophia', 'Martinez', 'F', '0552755189',
'martinez@email.com', 'MyP@ss20',TO_DATE('21-DEC-2003','DD-MON-YYYY'));

INSERT INTO user1 VALUES ('628816390', 'Alex', 'Joseph', 'M', '0518226390',
'ALJoseph@email.com', 'JosephAlex2001',TO_DATE('03-JUN-2001','DD-MON-YYYY'));
```



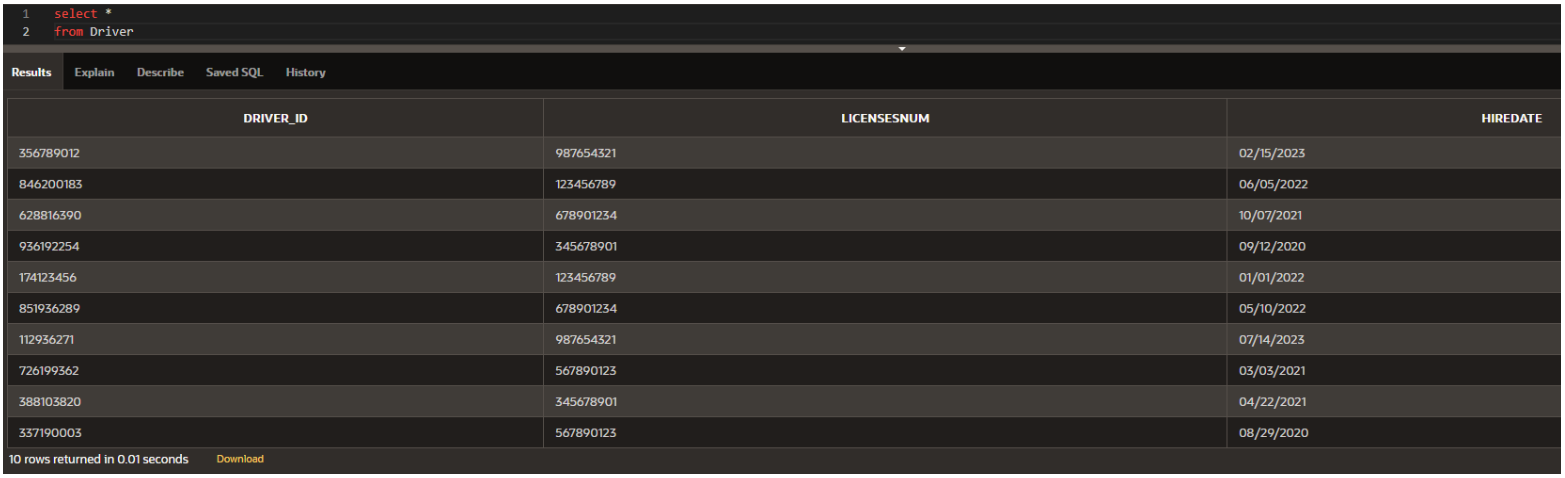
```
select *
from User1
```

USER_ID	FNAME	LNAME	GENDER	PHONE	EMAIL	PASSWORD	DOB
936192254	Sophia	Martinez	F	0552755189	martinez@email.com	MyP@ss20	12/21/2003
356789012	Jane	Smith	F	0517200172	jane@email.com	pass456word	12/10/1988
846200183	Christopher	Davis	M	0527183622	chrisD@email.com	1SecurePwd	09/30/1982
174123456	John	Doe	M	0582644219	john.doe@gmail.com	password123	10/20/2003
851936289	Emily	Johnson	F	0528173629	EJohnson@email.com	RandomPwd456	01/05/2000
628816390	Alex	Joseph	M	0518226390	ALJoseph@email.com	JosephAlex2001	06/03/2001
112936271	Benjamin	Taylor	M	0538229104	bentaylor@email.com	Pa$$123	07/08/1994
726199362	Sarah	Abdullah	F	0588173920	sarah.abdullah@email.com	P@ssw0rd123	08/23/1995
388103820	Ethan	Wilson	M	0527112946	ethanwilson@email.com	Secret!789	04/12/1987
337190003	Olivia	Brown	F	0527190036	Brownolivia@email.com	ChocoLover77	03/14/0014

10 rows returned in 0.00 seconds

2) Inserting 10 records in Driver table:

```
INSERT INTO Driver VALUES ('174123456', '123456789', TO_DATE('01-JAN-2022',
'DD-MON-YYYY'));

INSERT INTO Driver VALUES ('356789012', '987654321', TO_DATE('15-FEB-2023',
'DD-MON-YYYY'));

INSERT INTO Driver VALUES ('726199362', '567890123', TO_DATE('03-MAR-2021',
'DD-MON-YYYY'));

INSERT INTO Driver VALUES ('388103820', '345678901', TO_DATE('22-APR-2021',
'DD-MON-YYYY'));

INSERT INTO Driver VALUES ('851936289', '678901234', TO_DATE('10-MAY-2022',
'DD-MON-YYYY'));

INSERT INTO Driver VALUES ('846200183', '123456789', TO_DATE('05-JUN-2022',
'DD-MON-YYYY'));

INSERT INTO Driver VALUES ('112936271', '987654321', TO_DATE('14-JUL-2023',
'DD-MON-YYYY'));

INSERT INTO Driver VALUES ('337190003', '567890123', TO_DATE('29-AUG-2020',
'DD-MON-YYYY'));

INSERT INTO Driver VALUES ('936192254', '345678901', TO_DATE('12-SEP-2020',
'DD-MON-YYYY'));

INSERT INTO Driver VALUES ('628816390', '678901234', TO_DATE('07-OCT-2021',
'DD-MON-YYYY'));
```



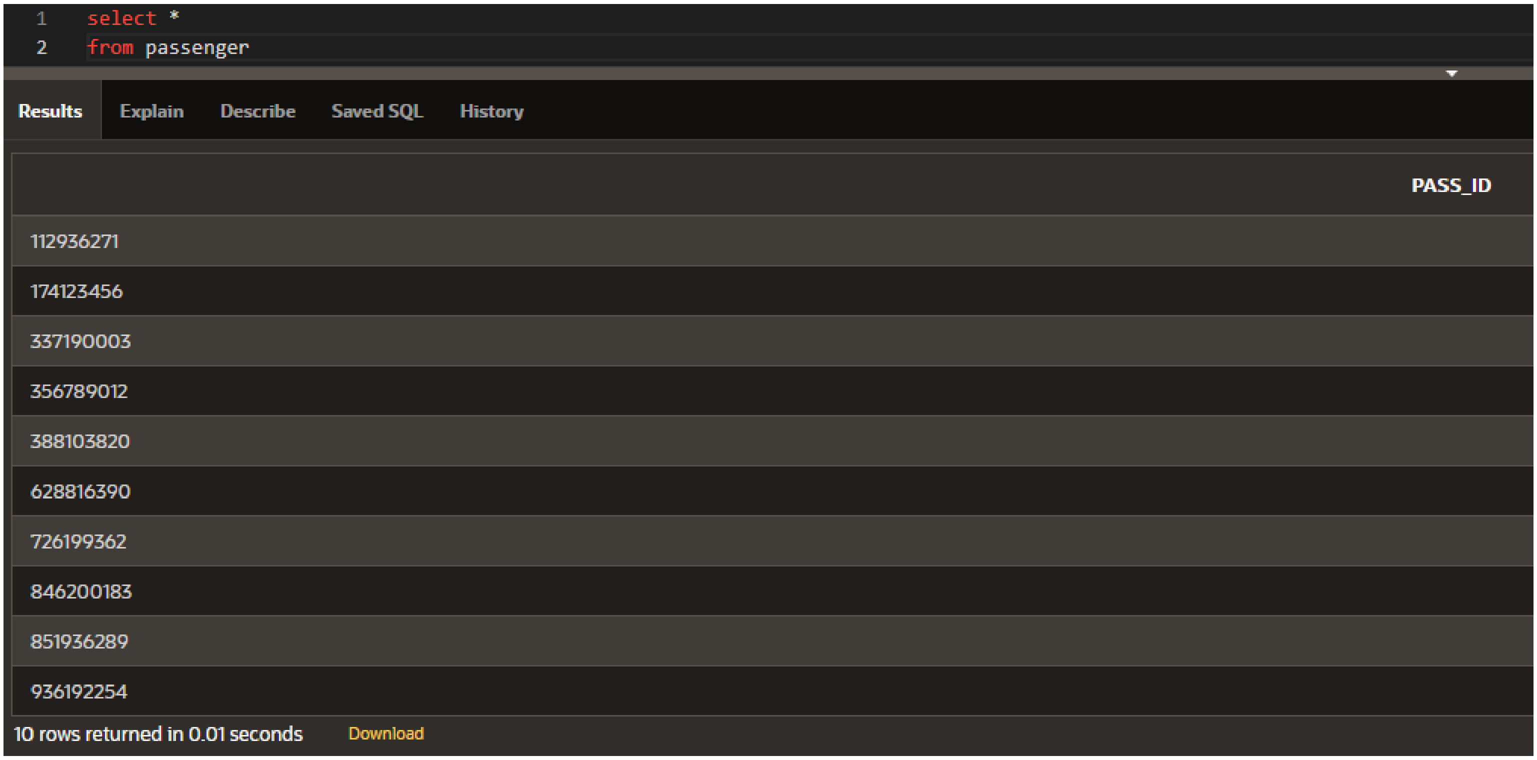
```
select *
from Driver
```

DRIVER_ID	LICENSESNUM	HIREDATE
356789012	987654321	02/15/2023
846200183	123456789	06/05/2022
628816390	678901234	10/07/2021
936192254	345678901	09/12/2020
174123456	123456789	01/01/2022
851936289	678901234	05/10/2022
112936271	987654321	07/14/2023
726199362	567890123	03/03/2021
388103820	345678901	04/22/2021
337190003	567890123	08/29/2020

10 rows returned in 0.01 seconds

3) Inserting 10 records in Passenger table:

```
INSERT INTO Passenger VALUES (‘174123456’);

INSERT INTO Passenger VALUES ('356789012');

INSERT INTO Passenger VALUES ('726199362');

INSERT INTO Passenger VALUES (‘388103820’);

INSERT INTO Passenger VALUES (‘851936289’);

INSERT INTO Passenger VALUES (‘846200183’);

INSERT INTO Passenger VALUES (‘112936271’);

INSERT INTO Passenger VALUES (‘337190003’);

INSERT INTO Passenger VALUES (‘936192254’);

INSERT INTO Passenger VALUES ('628816390');
```



```sql
select *
from passenger
```

PASS_ID
112936271
174123456
337190003
356789012
388103820
628816390
726199362
846200183
851936289
936192254

10 rows returned in 0.01 seconds

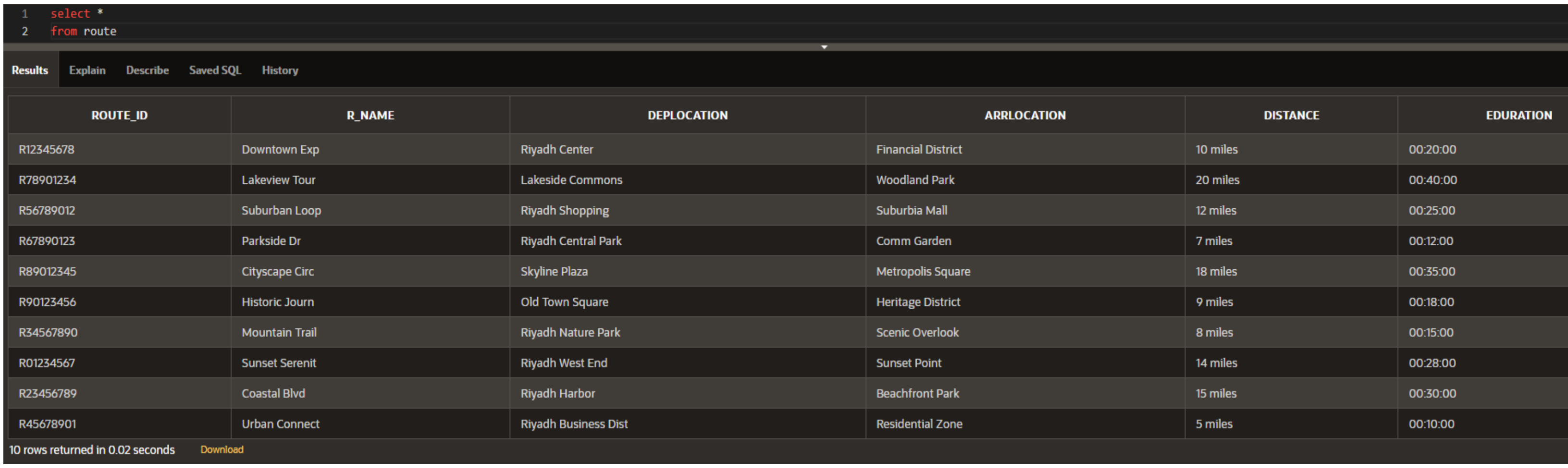
4) Inserting 10 records in Route table:

```sql
INSERT INTO Route VALUES ('R12345678', 'Downtown Exp', 'Riyadh Center', 'Financial District', '10 miles', '00:20:00');

INSERT INTO Route VALUES ('R23456789', 'Coastal Blvd', 'Riyadh Harbor', 'Beachfront Park', '15 miles', '00:30:00');

INSERT INTO Route VALUES ('R34567890', 'Mountain Trail', 'Riyadh Nature Park', 'Scenic Overlook', '8 miles', '00:15:00');

INSERT INTO Route VALUES ('R45678901', 'Urban Connect', 'Riyadh Business Dist', 'Residential Zone', '5 miles', '00:10:00');

INSERT INTO Route VALUES ('R56789012', 'Suburban Loop', 'Riyadh Shopping', 'Suburbia Mall', '12 miles', '00:25:00');

INSERT INTO Route VALUES ('R67890123', 'Parkside Dr', 'Riyadh Central Park', 'Comm Garden', '7 miles', '00:12:00');

INSERT INTO Route VALUES ('R78901234', 'Lakeview Tour', 'Lakeside Commons', 'Woodland Park', '20 miles', '00:40:00');
```

INSERT INTO Route VALUES ('R89012345', 'Cityscape Circ', 'Skyline Plaza', 'Metropolis Square', '18 miles', '00:35:00');

INSERT INTO Route VALUES ('R90123456', 'Historic Journ', 'Old Town Square', 'Heritage District', '9 miles', '00:18:00');

INSERT INTO Route VALUES ('R01234567', 'Sunset Serenit', 'Riyadh West End', 'Sunset Point', '14 miles', '00:28:00');



```
select *
from route
```

ROUTE_ID	R_NAME	DEPLOCATION	ARRLOCATION	DISTANCE	EDURATION
R12345678	Downtown Exp	Riyadh Center	Financial District	10 miles	00:20:00
R78901234	Lakeview Tour	Lakeside Commons	Woodland Park	20 miles	00:40:00
R56789012	Suburban Loop	Riyadh Shopping	Suburbia Mall	12 miles	00:25:00
R67890123	Parkside Dr	Riyadh Central Park	Comm Garden	7 miles	00:12:00
R89012345	Cityscape Circ	Skyline Plaza	Metropolis Square	18 miles	00:35:00
R90123456	Historic Journ	Old Town Square	Heritage District	9 miles	00:18:00
R34567890	Mountain Trail	Riyadh Nature Park	Scenic Overlook	8 miles	00:15:00
R01234567	Sunset Serenit	Riyadh West End	Sunset Point	14 miles	00:28:00
R23456789	Coastal Blvd	Riyadh Harbor	Beachfront Park	15 miles	00:30:00
R45678901	Urban Connect	Riyadh Business Dist	Residential Zone	5 miles	00:10:00

10 rows returned in 0.02 seconds   Download

5) **Inserting 10 records in Bus table:**

INSERT INTO Bus VALUES ('123456789', 1011, '50', 'Mercedes', TO_DATE('2019', 'YYYY'), 'C-Class', '174123456', 'R12345678');

INSERT INTO Bus VALUES ('234567899', 1022, '45', 'Volvo', TO_DATE('2019', 'YYYY'), 'V60', '356789012', 'R23456789');

INSERT INTO Bus VALUES ('345678900', 1033, '40', 'MAN', TO_DATE('2021', 'YYYY'), 'Lion Coach', '726199362', 'R34567890');

INSERT INTO Bus VALUES ('456789011', 1044, '55', 'Scania', TO_DATE('2021', 'YYYY'), 'Touring', '388103820', 'R45678901');

INSERT INTO Bus VALUES ('567890122', 1055, '48', 'Iveco', TO_DATE('2020', 'YYYY'), 'Crossway', '851936289', 'R56789012');

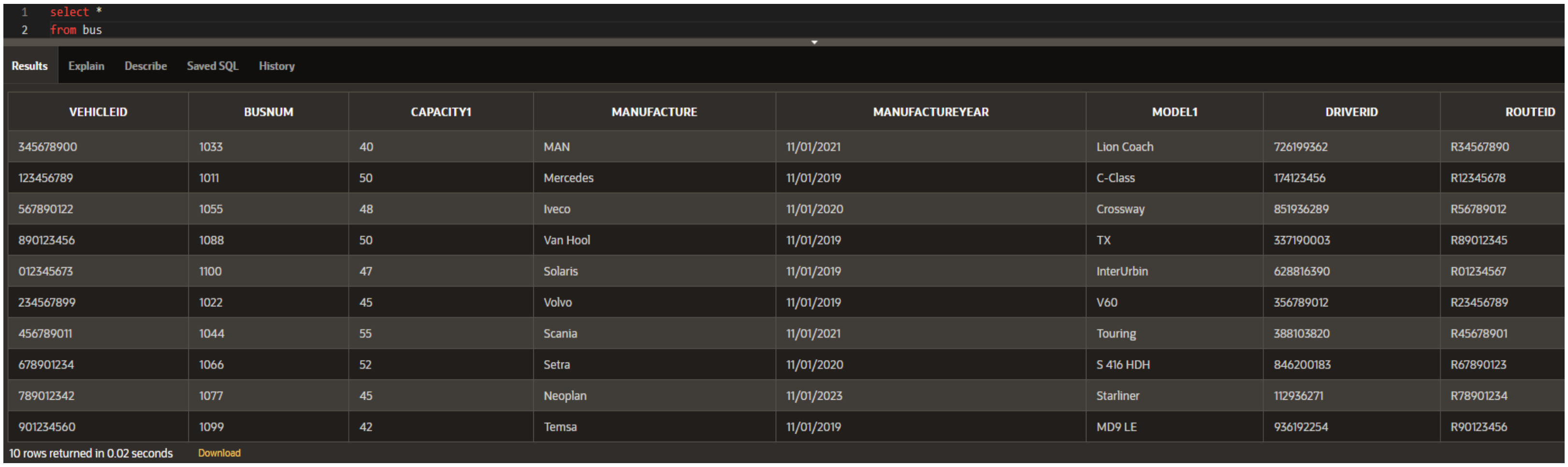
INSERT INTO Bus VALUES ('678901234', 1066, '52', 'Setra', TO_DATE('2020', 'YYYY'), 'S 416 HDH', '846200183', 'R67890123');

INSERT INTO Bus VALUES ('789012342', 1077, '45', 'Neoplan', TO_DATE('2023', 'YYYY'), 'Starliner', '112936271', 'R78901234');

INSERT INTO Bus VALUES ('890123456', 1088, '50', 'Van Hool', TO_DATE('2019', 'YYYY'), 'TX', '337190003', 'R89012345');

INSERT INTO Bus VALUES ('901234560', 1099, '42', 'Temsa', TO_DATE('2019', 'YYYY'), 'MD9 LE', '936192254', 'R90123456');

INSERT INTO Bus VALUES ('012345673', 1100, '47', 'Solaris', TO_DATE('2019', 'YYYY'), 'InterUrbin', '628816390', 'R01234567');



```
select *
from bus
```

VEHICLEID	BUSNUM	CAPACITY1	MANUFACTURE	MANUFACTUREYEAR	MODEL1	DRIVERID	ROUTEID
345678900	1033	40	MAN	11/01/2021	Lion Coach	726199362	R34567890
123456789	1011	50	Mercedes	11/01/2019	C-Class	174123456	R12345678
567890122	1055	48	Iveco	11/01/2020	Crossway	851936289	R56789012
890123456	1088	50	Van Hool	11/01/2019	TX	337190003	R89012345
012345673	1100	47	Solaris	11/01/2019	InterUrbin	628816390	R01234567
234567899	1022	45	Volvo	11/01/2019	V60	356789012	R23456789
456789011	1044	55	Scania	11/01/2021	Touring	388103820	R45678901
678901234	1066	52	Setra	11/01/2020	S 416 HDH	846200183	R67890123
789012342	1077	45	Neoplan	11/01/2023	Starliner	112936271	R78901234
901234560	1099	42	Temsa	11/01/2019	MD9 LE	936192254	R90123456

10 rows returned in 0.02 seconds

6) Inserting 10 records in schedule table:

INSERT INTO schedule VALUES (183719026, '08:00 AM', '10:00 AM', 1011);

INSERT INTO schedule VALUES (263991702, '09:30 AM', '11:30 AM', 1022);

INSERT INTO schedule VALUES (373819462, '10:45 AM', '12:45 PM', 1033);

INSERT INTO schedule VALUES (471822637, '12:15 PM', '02:15 PM', 1044);

INSERT INTO schedule VALUES (518263911, '01:45 PM', '03:45 PM', 1055);

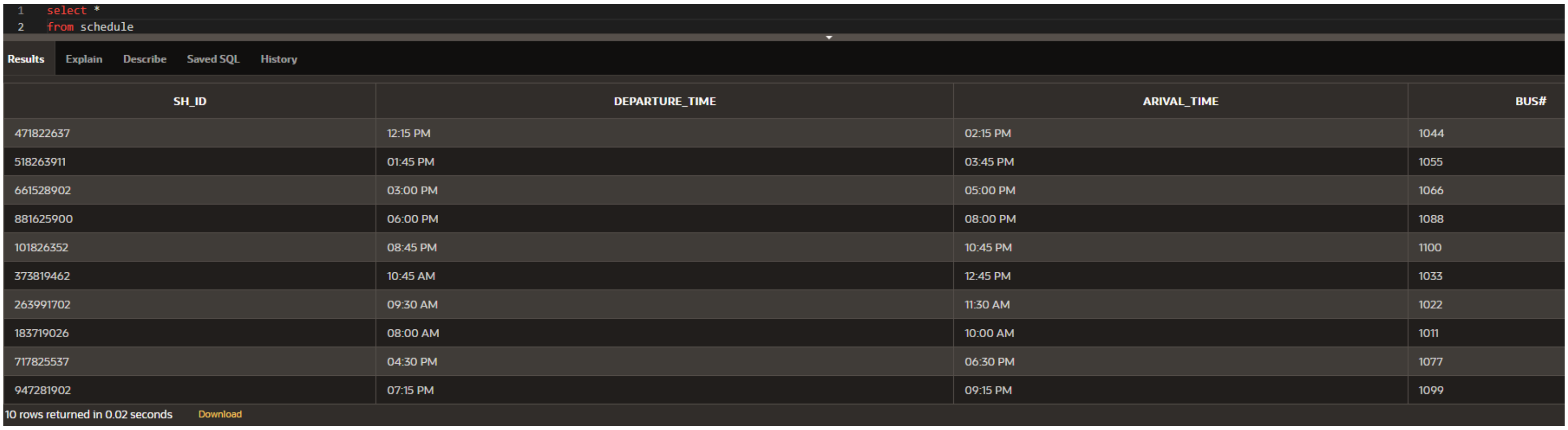
INSERT INTO schedule VALUES (661528902, '03:00 PM', '05:00 PM', 1066);

INSERT INTO schedule VALUES (717825537, '04:30 PM', '06:30 PM', 1077);

INSERT INTO schedule VALUES (881625900, '06:00 PM', '08:00 PM', 1088);

INSERT INTO schedule VALUES (947281902, '07:15 PM', '09:15 PM', 1099);

INSERT INTO schedule VALUES (101826352, '08:45 PM', '10:45 PM', 1100);



```
select *
from schedule
```

SH_ID	DEPARTURE_TIME	ARIVAL_TIME	BUS#
471822637	12:15 PM	02:15 PM	1044
518263911	01:45 PM	03:45 PM	1055
661528902	03:00 PM	05:00 PM	1066
881625900	06:00 PM	08:00 PM	1088
101826352	08:45 PM	10:45 PM	1100
373819462	10:45 AM	12:45 PM	1033
263991702	09:30 AM	11:30 AM	1022
183719026	08:00 AM	10:00 AM	1011
717825537	04:30 PM	06:30 PM	1077
947281902	07:15 PM	09:15 PM	1099

10 rows returned in 0.02 seconds

7) Inserting 10 records in Ticket table:

INSERT INTO Ticket VALUES ('T12345678', TO_DATE('2023-11-14', 'YYYY-MM-DD'), '50.00', 1);

INSERT INTO Ticket VALUES ('T23456789', TO_DATE('2023-11-14', 'YYYY-MM-DD'), '45.00', 2);

INSERT INTO Ticket VALUES ('T34567890', TO_DATE('2023-11-14', 'YYYY-MM-DD'), '40.00', 3);

INSERT INTO Ticket VALUES ('T45678901', TO_DATE('2023-11-14', 'YYYY-MM-DD'), '55.00', 4);

INSERT INTO Ticket VALUES ('T56789012', TO_DATE('2023-11-14', 'YYYY-MM-DD'), '48.00', 5);

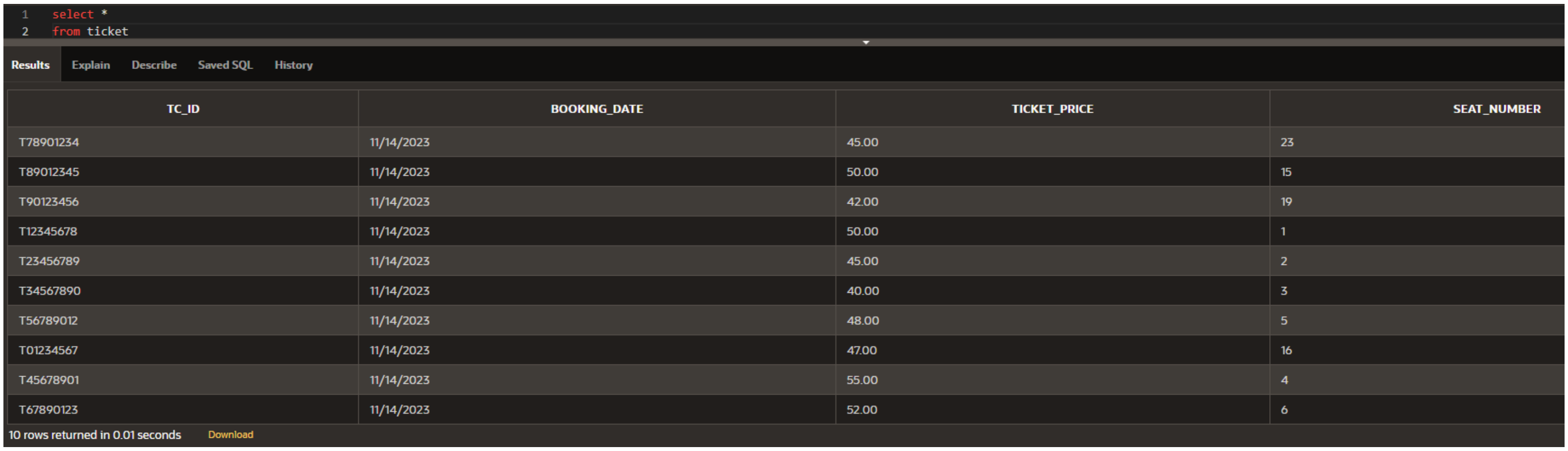
INSERT INTO Ticket VALUES ('T67890123', TO_DATE('2023-11-14', 'YYYY-MM-DD'), '52.00', 6);

INSERT INTO Ticket VALUES ('T78901234', TO_DATE('2023-11-14', 'YYYY-MM-DD'), '45.00', 23);

INSERT INTO Ticket VALUES ('T89012345', TO_DATE('2023-11-14', 'YYYY-MM-DD'), '50.00', 15);

INSERT INTO Ticket VALUES ('T90123456', TO_DATE('2023-11-14', 'YYYY-MM-DD'), '42.00', 19);

INSERT INTO Ticket VALUES ('T01234567', TO_DATE('2023-11-14', 'YYYY-MM-DD'), '47.00', 16);



```
select *
from ticket
```

TC_ID	BOOKING_DATE	TICKET_PRICE	SEAT_NUMBER
T78901234	11/14/2023	45.00	23
T89012345	11/14/2023	50.00	15
T90123456	11/14/2023	42.00	19
T12345678	11/14/2023	50.00	1
T23456789	11/14/2023	45.00	2
T34567890	11/14/2023	40.00	3
T56789012	11/14/2023	48.00	5
T01234567	11/14/2023	47.00	16
T45678901	11/14/2023	55.00	4
T67890123	11/14/2023	52.00	6

10 rows returned in 0.01 seconds

8) Inserting 10 records in Payment table:

INSERT INTO Payment VALUES ('P123456789', '50.00', TO_DATE('2023-11-14', 'YYYY-MM-DD'), 'Credit Card', 'T12345678');

INSERT INTO Payment VALUES ('P234567890', '45.00', TO_DATE('2023-11-14', 'YYYY-MM-DD'), 'PayPal', 'T23456789');

INSERT INTO Payment VALUES ('P345678901', '40.00', TO_DATE('2023-11-14', 'YYYY-MM-DD'), 'Cash', 'T34567890');

INSERT INTO Payment VALUES ('P456789012', '55.00', TO_DATE('2023-11-14', 'YYYY-MM-DD'), 'Credit Card', 'T45678901');

INSERT INTO Payment VALUES ('P567890123', '48.00', TO_DATE('2023-11-14', 'YYYY-MM-DD'), 'Cash', 'T56789012');

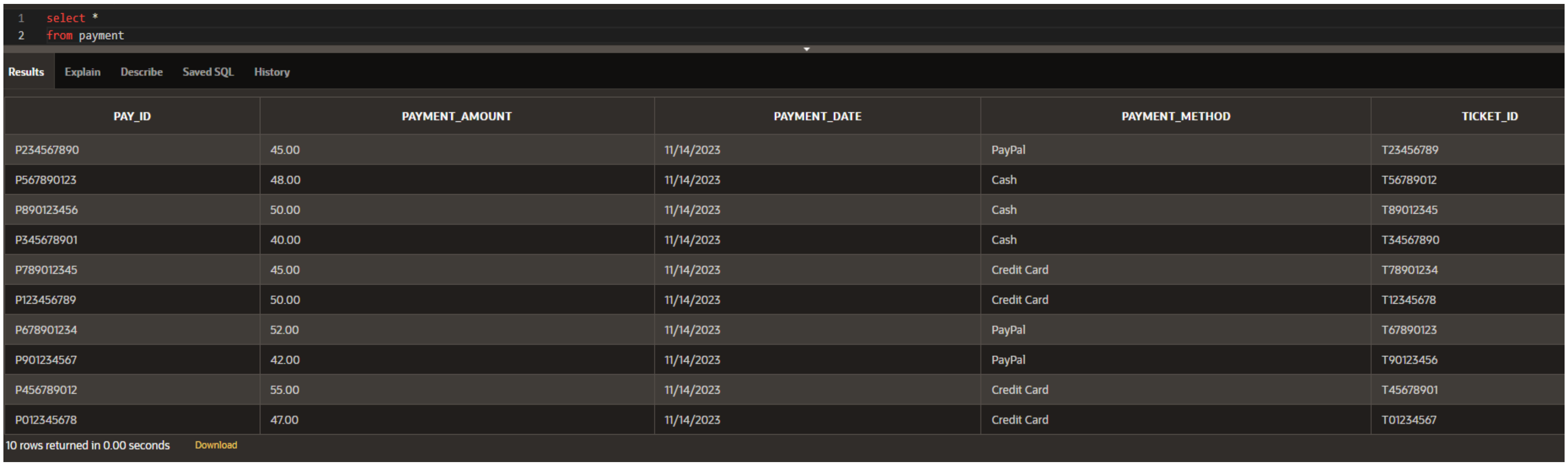
INSERT INTO Payment VALUES ('P678901234', '52.00', TO_DATE('2023-11-14', 'YYYY-MM-DD'), 'PayPal', 'T67890123');

INSERT INTO Payment VALUES ('P789012345', '45.00', TO_DATE('2023-11-14', 'YYYY-MM-DD'), 'Credit Card', 'T78901234');

INSERT INTO Payment VALUES ('P890123456', '50.00', TO_DATE('2023-11-14', 'YYYY-MM-DD'), 'Cash', 'T89012345');

INSERT INTO Payment VALUES ('P901234567', '42.00', TO_DATE('2023-11-14', 'YYYY-MM-DD'), 'PayPal', 'T90123456');

INSERT INTO Payment VALUES ('P012345678', '47.00', TO_DATE('2023-11-14', 'YYYY-MM-DD'), 'Credit Card', 'T01234567');



```
select *
from payment
```

PAY_ID	PAYMENT_AMOUNT	PAYMENT_DATE	PAYMENT_METHOD	TICKET_ID
P234567890	45.00	11/14/2023	PayPal	T23456789
P567890123	48.00	11/14/2023	Cash	T56789012
P890123456	50.00	11/14/2023	Cash	T89012345
P345678901	40.00	11/14/2023	Cash	T34567890
P789012345	45.00	11/14/2023	Credit Card	T78901234
P123456789	50.00	11/14/2023	Credit Card	T12345678
P678901234	52.00	11/14/2023	PayPal	T67890123
P901234567	42.00	11/14/2023	PayPal	T90123456
P456789012	55.00	11/14/2023	Credit Card	T45678901
P012345678	47.00	11/14/2023	Credit Card	T01234567

10 rows returned in 0.00 seconds

9) Inserting 10 records in Buy table:

INSERT INTO Buy VALUES ('T12345678', '174123456', 183719026, 1, TO_DATE('2023-11-14', 'YYYY-MM-DD'));

INSERT INTO Buy VALUES ('T23456789', '356789012', 263991702, 2, TO_DATE('2023-11-14', 'YYYY-MM-DD'));

INSERT INTO Buy VALUES ('T34567890', '726199362', 373819462, 3, TO_DATE('2023-11-14', 'YYYY-MM-DD'));

INSERT INTO Buy VALUES ('T45678901', '388103820', 471822637, 4, TO_DATE('2023-11-14', 'YYYY-MM-DD'));

INSERT INTO Buy VALUES ('T56789012', '851936289', 518263911, 5, TO_DATE('2023-11-14', 'YYYY-MM-DD'));

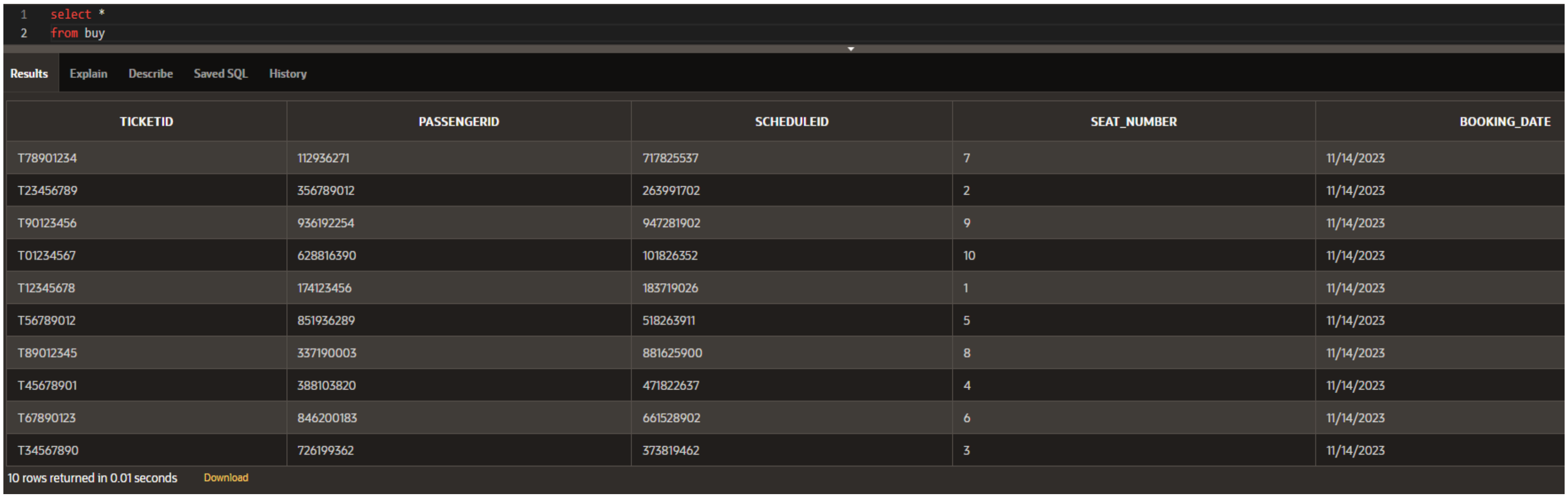
INSERT INTO Buy VALUES ('T67890123', '846200183', 661528902, 6, TO_DATE('2023-11-14', 'YYYY-MM-DD'));

INSERT INTO Buy VALUES ('T78901234', '112936271', 717825537, 7, TO_DATE('2023-11-14', 'YYYY-MM-DD'));

INSERT INTO Buy VALUES ('T89012345', '337190003', 881625900, 8, TO_DATE('2023-11-14', 'YYYY-MM-DD'));

INSERT INTO Buy VALUES ('T90123456', '936192254', 947281902, 9, TO_DATE('2023-11-14', 'YYYY-MM-DD'));

INSERT INTO Buy VALUES ('T01234567', '628816390', 101826352, 10, TO_DATE('2023-11-14', 'YYYY-MM-DD'));



```
select *
from buy
```

TICKETID	PASSENGERID	SCHEDULEID	SEAT_NUMBER	BOOKING_DATE
T78901234	112936271	717825537	7	11/14/2023
T23456789	356789012	263991702	2	11/14/2023
T90123456	936192254	947281902	9	11/14/2023
T01234567	628816390	101826352	10	11/14/2023
T12345678	174123456	183719026	1	11/14/2023
T56789012	851936289	518263911	5	11/14/2023
T89012345	337190003	881625900	8	11/14/2023
T45678901	388103820	471822637	4	11/14/2023
T67890123	846200183	661528902	6	11/14/2023
T34567890	726199362	373819462	3	11/14/2023

10 rows returned in 0.01 seconds

18. Relevant clauses from the IEEE and ACM Professional Code of Ethics

18.1. IEEE Code of Ethics

- To uphold the greatest standards of morality, responsibility, and ethics in all aspects of one's professional life.
- To make an effort to see that Students, DSA members, and supervisors uphold this code.
- To prioritize the public's safety and welfare.
- To work toward adherence to ethical design and sustainable development standards.
- To safeguard others' privacy.
- To quickly disclose elements that could put the public or the environment in danger.
- To refrain from discrimination based on factors such as race, religion, gender, age, and country of origin.
- To treat all people equally and with respect.
- To refrain from any form of harassment

18.2. ACM Code of Ethics

18.2.1. General Ethical Principles

1.1 Contribute to society and to human well-being, acknowledging that all people are stakeholders in computing.
1.2 Avoid harm.
1.3 Be honest and trustworthy.
1.4 Be fair and take action not to discriminate.
1.5 Respect privacy.
1.6 Honor confidentiality

18.2.2. Professional Responsibilities

2.1 Maintain high standards of professional competence, conduct, and ethical practice.
2.2 Know and respect existing rules pertaining to professional work.
2.3 Access computing and communication resources only when authorized or when compelled by the public good.
2.4 Design and implement systems that are robust and usably secure.

18.2.3. Professional Leadership Principles

3.1 Articulate, encourage acceptance of and evaluate fulfillment of social responsibilities by members of the organization or group.
3.2 Manage personnel and resources to enhance the quality of working life.
3.3 Articulate, apply, and support policies and processes that reflect the principles of the Code.
3.4 Create opportunities for members of the organization or group to grow as professionals.
3.5 Use care when modifying or retiring systems.

18.2.4. Compliance with the Code

4.1 Uphold, promote, and respect the principles of the Code.
4.2 Treat violations of the Code as inconsistent with membership in the ACM

19. Teamwork Distribution and Strategy

19.1. Strategy

One of the most important strategies for demonstrating collaboration is to ascertain the specific duties and obligations of each team member. As a result, team members will feel more accountable to one another and will be fully aware of their individual roles and areas of competence. Each participant was required to oversee her own assignment

Every team member actively communicated with the others, sharing knowledge, suggestions, and criticism. This was accomplished through frequent meetings and instant chat; each member provided real-time updates and tracked their own progress.

19.2. Individual Contribution

Name	**Contribution**
Sarah AlJurbua	Inserting data into tables
Noura AlAngari	Tables along with all the required integrity and column constraints
Nouf Alabduljabbar	Relevant clauses from the IEEE and ACM professional Code of Ethics
Nour Fatoom	Tables along with all the required integrity and column constraints

Riyadh Bus Reservation System

Department of Computer & Information Sciences

CS340 Project deliverable 5

Instructor: Dr. Roohi Jan

Section: 799

Prepared By:

Sarah AlJurbua 220410528

Noura AlAngari 220410015

Nouf Abduljabbar 220410775

Nour Fatoom 220410494

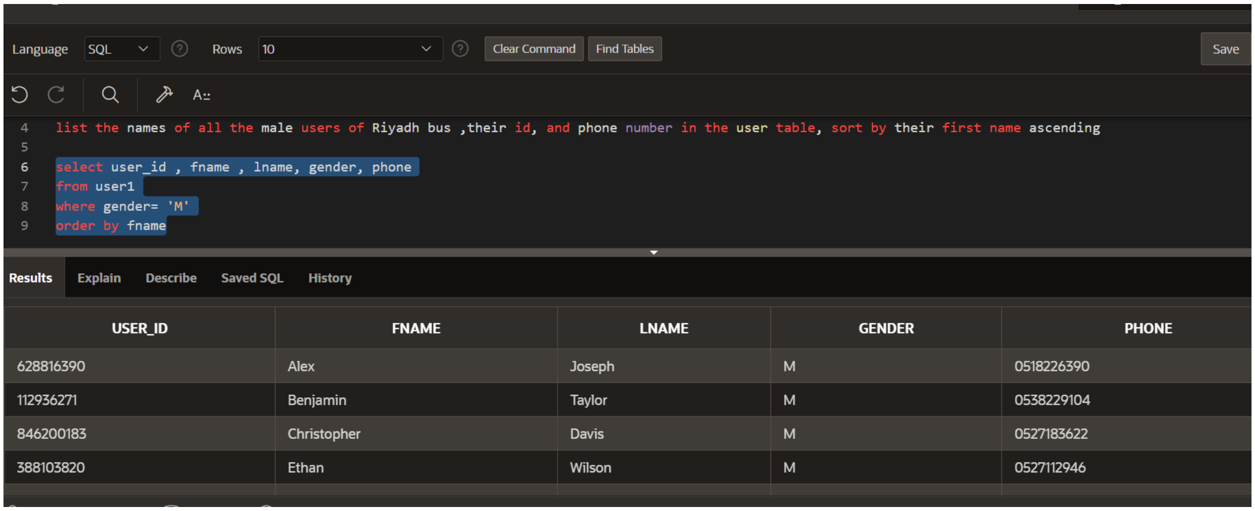
Phase 5

Creating simple and advanced queries using Oracle

- **Basic 10 SQL Queries:**

1-list the names of all the male users of Riyadh bus ,their id, and phone number in the user table, sort by their first name ascending:

select user_id , fname , lname, gender, phone
from user1
where gender= 'M'
order by fname



Language SQL

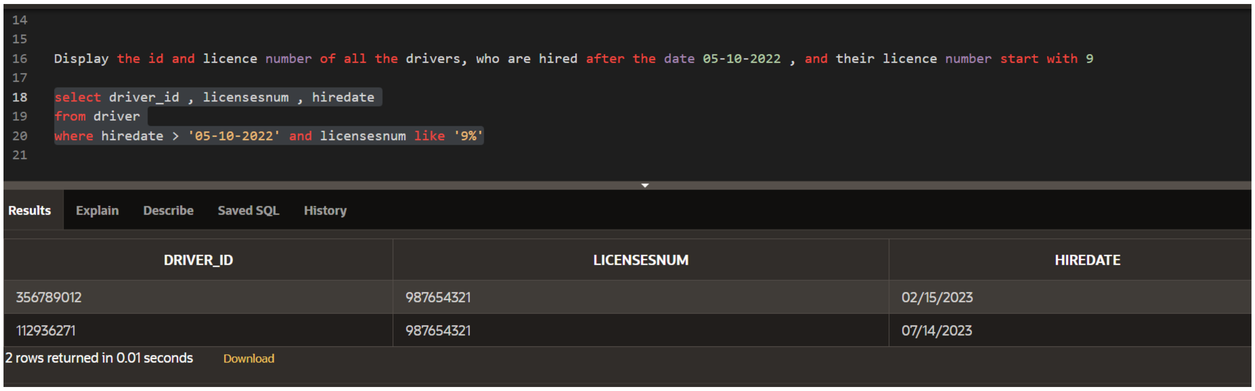
Rows 10

```
list the names of all the male users of Riyadh bus ,their id, and phone number in the user table, sort by their first name ascending

select user_id , fname , lname, gender, phone
from user1
where gender= 'M'
order by fname
```

USER_ID	FNAME	LNAME	GENDER	PHONE
628816390	Alex	Joseph	M	0518226390
112936271	Benjamin	Taylor	M	0538229104
846200183	Christopher	Davis	M	0527183622
388103820	Ethan	Wilson	M	0527112946

2-Display the id and license number of all the drivers, who are hired after the date 05-10-2022 , and their license number start with 9:

select driver_id , licensesnum , hiredate
from driver
where hiredate > '05-10-2022' and licensesnum like '9%'



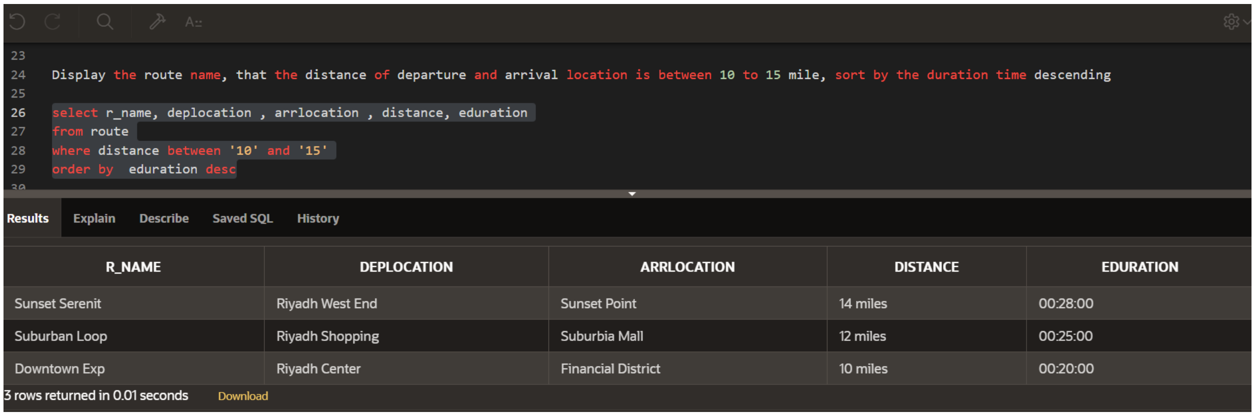
```


Display the id and licence number of all the drivers, who are hired after the date 05-10-2022 , and their licence number start with 9

select driver_id , licensesnum , hiredate
from driver
where hiredate > '05-10-2022' and licensesnum like '9%'

```

DRIVER_ID	LICENSESNUM	HIREDATE
356789012	987654321	02/15/2023
112936271	987654321	07/14/2023

2 rows returned in 0.01 seconds

3-Display the route name, that the distance of departure and arrival location is between 10 to 15 mile, sort by the duration time descending:

select r_name, deplocation , arrlocation , distance, eduration
from route
where distance between '10' and '15'
order by eduration desc



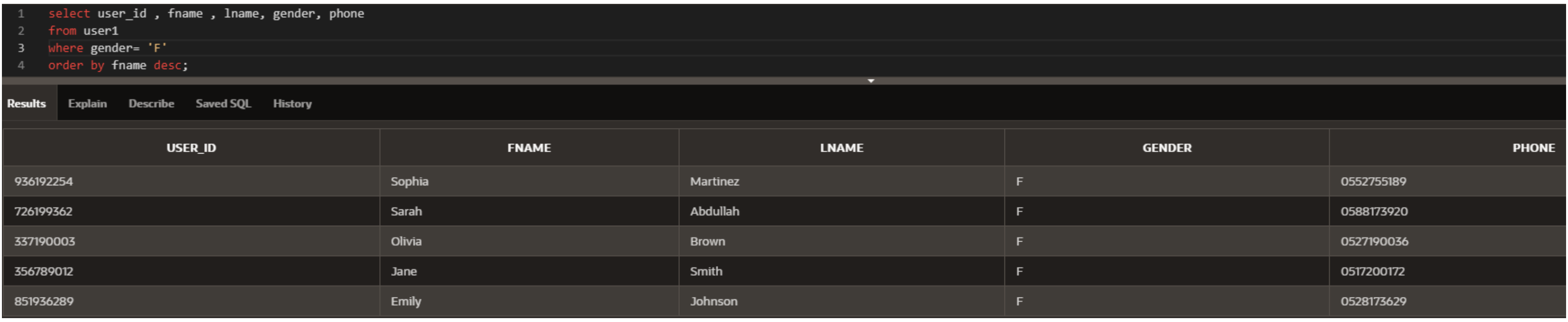
```
Display the route name, that the distance of departure and arrival location is between 10 to 15 mile, sort by the duration time descending

select r_name, deplocation , arrlocation , distance, eduration
from route
where distance between '10' and '15'
order by  eduration desc
```

R_NAME	DEPLOCATION	ARRLOCATION	DISTANCE	EDURATION
Sunset Serenit	Riyadh West End	Sunset Point	14 miles	00:28:00
Suburban Loop	Riyadh Shopping	Suburbia Mall	12 miles	00:25:00
Downtown Exp	Riyadh Center	Financial District	10 miles	00:20:00

3 rows returned in 0.01 seconds

4- List the names of all the female users of Riyadh bus ,their id, and phone number in the user table, sort by their first name descending order:

select user_id , fname , lname, gender, phone
from user1
where gender= 'F'
order by fname desc;

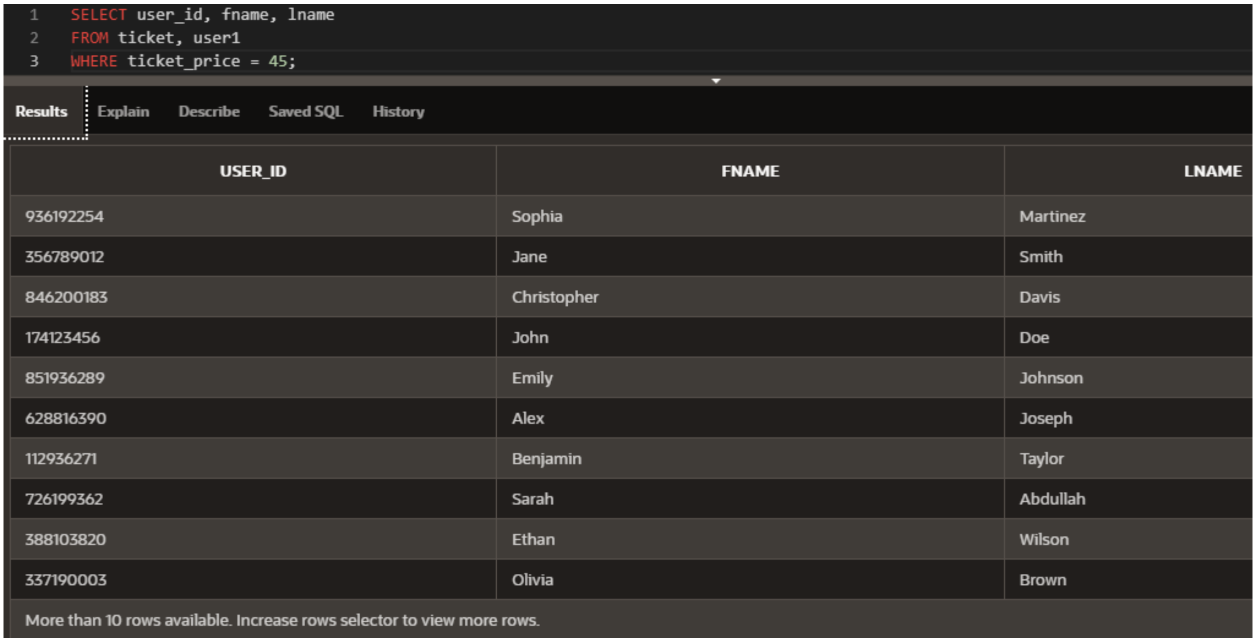


```
select user_id , fname , lname, gender, phone
from user1
where gender= 'F'
order by fname desc;
```

USER_ID	FNAME	LNAME	GENDER	PHONE
936192254	Sophia	Martinez	F	0552755189
726199362	Sarah	Abdullah	F	0588173920
337190003	Olivia	Brown	F	0527190036
356789012	Jane	Smith	F	0517200172
851936289	Emily	Johnson	F	0528173629

5- Display IDs and names of all users who made a reservation with ticket price 45:

SELECT user_id, fname, lname
FROM ticket, user1
WHERE ticket_price = 45;



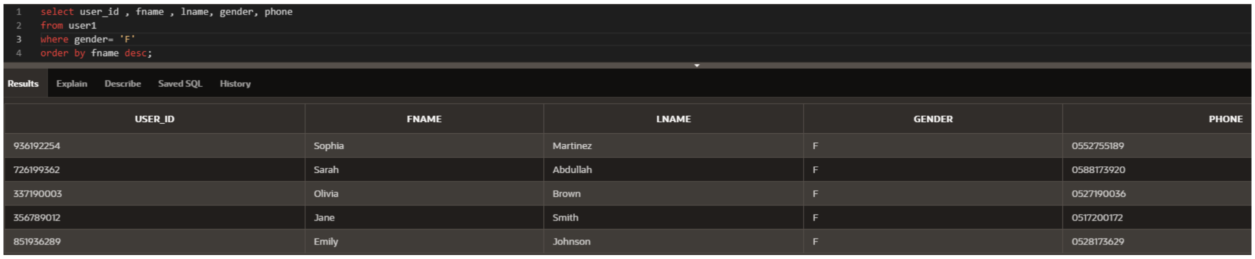
```
SELECT user_id, fname, lname
FROM ticket, user1
WHERE ticket_price = 45;
```

USER_ID	FNAME	LNAME
936192254	Sophia	Martinez
356789012	Jane	Smith
846200183	Christopher	Davis
174123456	John	Doe
851936289	Emily	Johnson
628816390	Alex	Joseph
112936271	Benjamin	Taylor
726199362	Sarah	Abdullah
388103820	Ethan	Wilson
337190003	Olivia	Brown

More than 10 rows available. Increase rows selector to view more rows.

6- Display user id and name for users whose payment method was paypal:

SELECT user_id, fname, lname
FROM payment, user1
WHERE payment_method='PayPal';



```
select user_id , fname , lname, gender, phone
from user1
where gender= 'F'
order by fname desc;
```

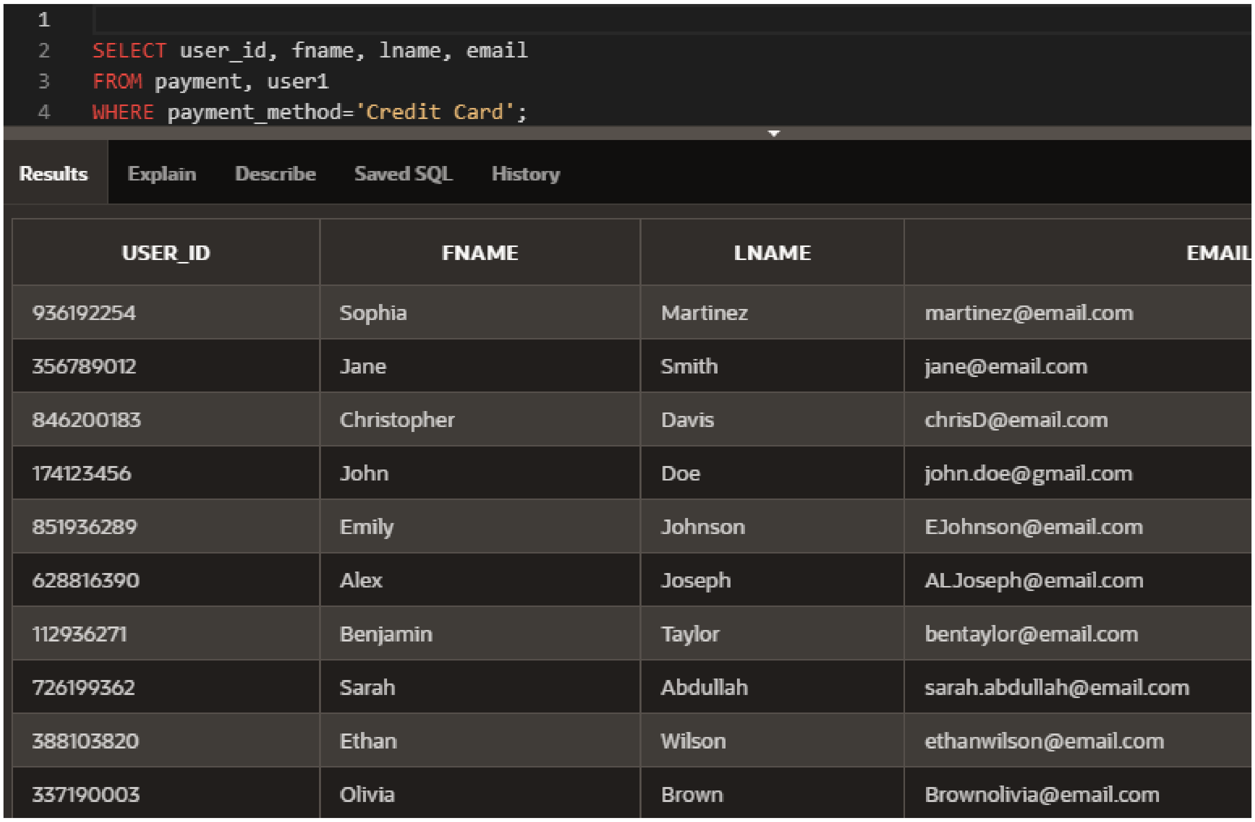
USER_ID	FNAME	LNAME	GENDER	PHONE
936192254	Sophia	Martinez	F	0552755189
726199362	Sarah	Abdullah	F	0588173920
337190003	Olivia	Brown	F	0527190036
356789012	Jane	Smith	F	0517200172
851936289	Emily	Johnson	F	0528173629

7- Display user id and name and emails for users whose payment method was credit card:

SELECT user_id, fname, lname, email

FROM payment, user1

WHERE payment_method='Credit Card';



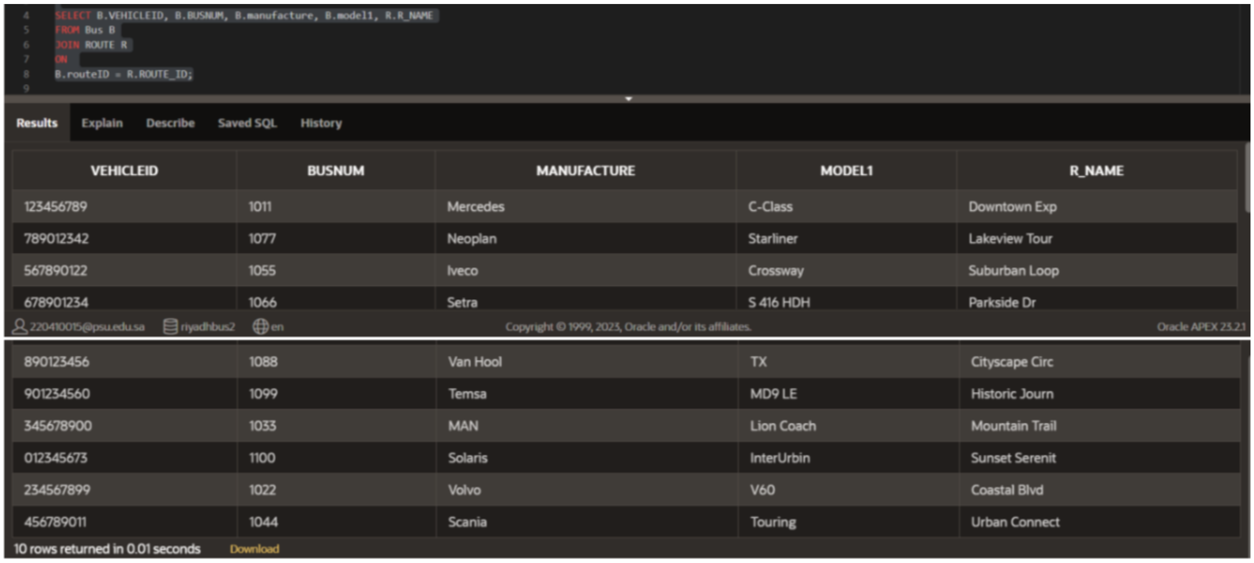
```
SELECT user_id, fname, lname, email
FROM payment, user1
WHERE payment_method='Credit Card';
```

USER_ID	FNAME	LNAME	EMAIL
936192254	Sophia	Martinez	martinez@email.com
356789012	Jane	Smith	jane@email.com
846200183	Christopher	Davis	chrisD@email.com
174123456	John	Doe	john.doe@gmail.com
851936289	Emily	Johnson	EJohnson@email.com
628816390	Alex	Joseph	ALJoseph@email.com
112936271	Benjamin	Taylor	bentaylor@email.com
726199362	Sarah	Abdullah	sarah.abdullah@email.com
388103820	Ethan	Wilson	ethanwilson@email.com
337190003	Olivia	Brown	Brownolivia@email.com

8- Display the bus and its routes details:

SELECT B.VEHICLEID, B.BUSNUM, B.manufacture, B.model1, R.R_NAME FROM Bus B
JOIN ROUTE R
ON

B.routeID = R.ROUTE_ID;



```
SELECT B.VEHICLEID, B.BUSNUM, B.manufacture, B.model1, R.R_NAME
FROM Bus B
JOIN ROUTE R
ON
B.routeID = R.ROUTE_ID;
```

VEHICLEID	BUSNUM	MANUFACTURE	MODEL1	R_NAME
123456789	1011	Mercedes	C-Class	Downtown Exp
789012342	1077	Neoplan	Starliner	Lakeview Tour
567890122	1055	Iveco	Crossway	Suburban Loop
678901234	1066	Setra	S 416 HDH	Parkside Dr
890123456	1088	Van Hool	TX	Cityscape Circ
901234560	1099	Temsa	MD9 LE	Historic Journ
345678900	1033	MAN	Lion Coach	Mountain Trail
012345673	1100	Solaris	InterUrbin	Sunset Serenit
234567899	1022	Volvo	V60	Coastal Blvd
456789011	1044	Scania	Touring	Urban Connect

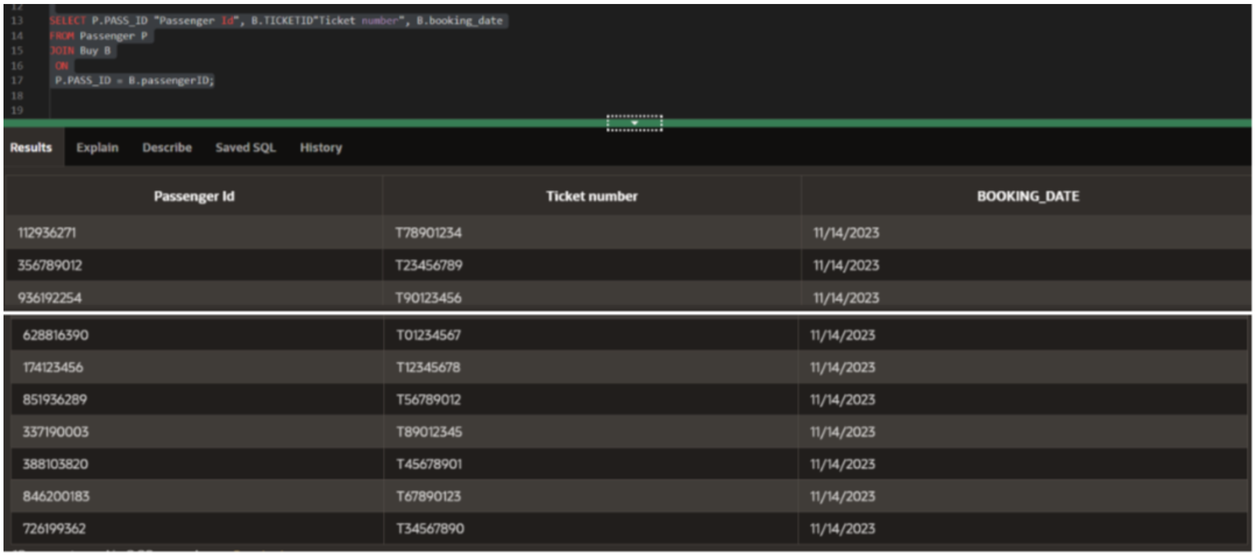
10 rows returned in 0.01 seconds

9- Display the passenger and booking info.:

SELECT P.PASS_ID "Passenger Id", B.TICKETID"Ticket number", B.booking_date FROM Passenger P
JOIN Buy B

ON

P.PASS_ID = B.passengerID;

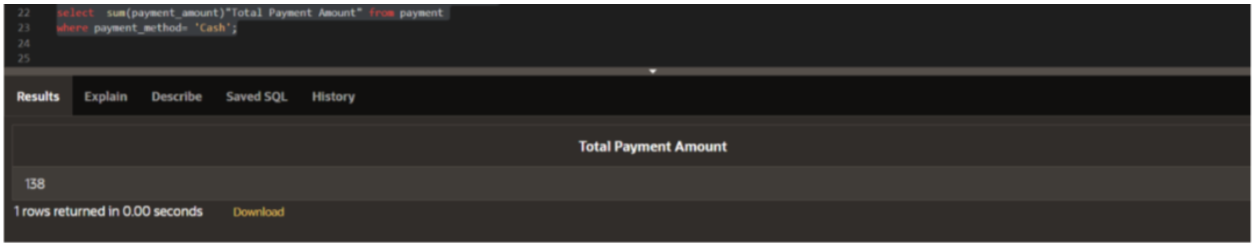


```
SELECT P.PASS_ID "Passenger Id", B.TICKETID"Ticket number", B.booking_date
FROM Passenger P
JOIN Buy B
 ON
 P.PASS_ID = B.passengerID;
```

Passenger Id	Ticket number	BOOKING_DATE
112936271	T78901234	11/14/2023
356789012	T23456789	11/14/2023
936192254	T90123456	11/14/2023
628816390	T01234567	11/14/2023
174123456	T12345678	11/14/2023
851936289	T56789012	11/14/2023
337190003	T89012345	11/14/2023
388103820	T45678901	11/14/2023
846200183	T67890123	11/14/2023
726199362	T34567890	11/14/2023

10- Display the total payment which was made by Cash method:

Select sum(payment_amount)"Total Payment Amount" from payment where payment_method= 'Cash';



```
select  sum(payment_amount)"Total Payment Amount" from payment
where payment_method= 'Cash';
```

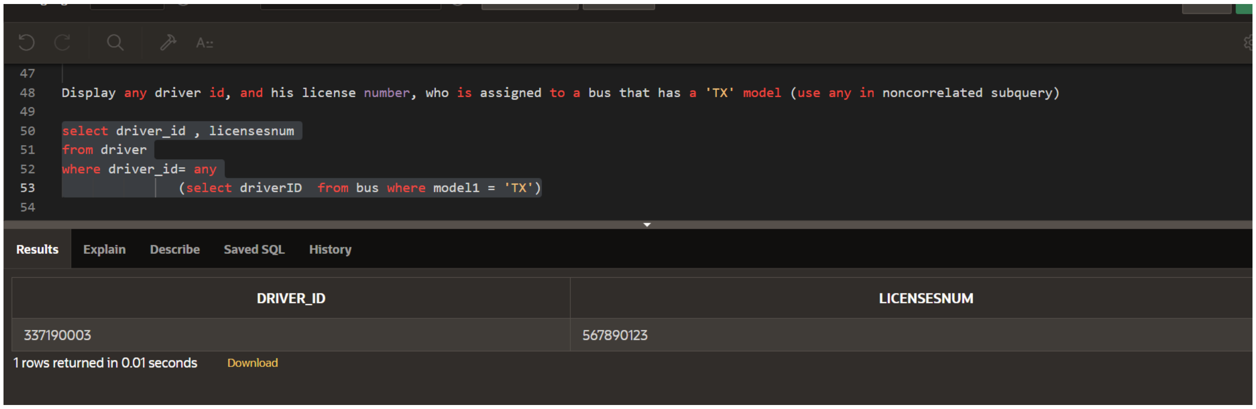
Total Payment Amount
138

1 rows returned in 0.00 seconds

- ## Advanced 10 SQL Queries:

1-Display any driver id, and his license number, who is assigned to a bus that has a 'TX' model (use any in noncorrelated subquery)

select driver_id , licensesnum
from driver
where driver_id= any
  (select driverID from bus where model1 = 'TX')



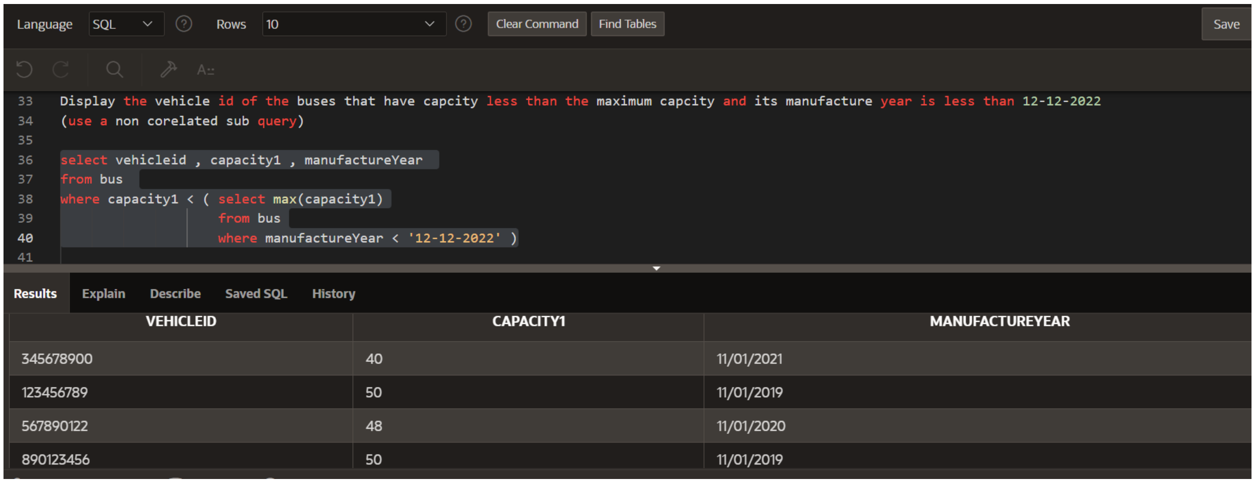
```
Display any driver id, and his license number, who is assigned to a bus that has a 'TX' model (use any in noncorrelated subquery)

select driver_id , licensesnum
from driver
where driver_id= any
            (select driverID  from bus where model1 = 'TX')
```

DRIVER_ID	LICENSESNUM
337190003	567890123

1 rows returned in 0.01 seconds   Download

2-Display the vehicle id of the buses that have capacity less than the maximum capacity and its manufacture year is less than 12-12-2022
(use a non correlated subquery)

select vehicleid , capacity1 , manufactureYear
from bus
where capacity1 < (select max(capacity1)
                from bus
                where manufactureYear < '12-12-2022')



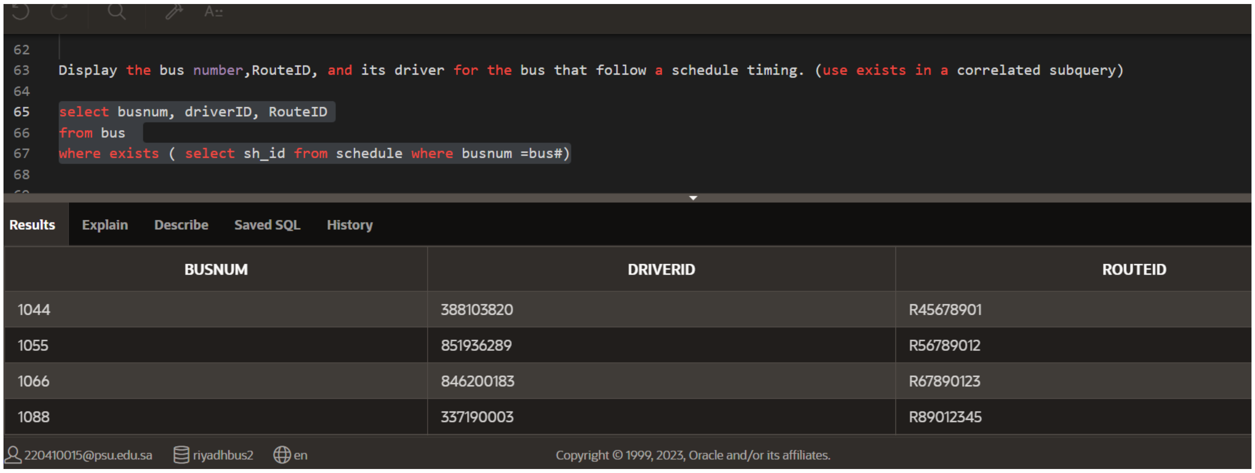
```
Display the vehicle id of the buses that have capcity less than the maximum capcity and its manufacture year is less than 12-12-2022
(use a non corelated sub query)

select vehicleid , capacity1 , manufactureYear
from bus
where capacity1 < ( select max(capacity1)
                    from bus
                    where manufactureYear < '12-12-2022' )
```

VEHICLEID	CAPACITY1	MANUFACTUREYEAR
345678900	40	11/01/2021
123456789	50	11/01/2019
567890122	48	11/01/2020
890123456	50	11/01/2019

3-Display the bus number,RouteID, and its driver for the bus that follows a schedule timing. (use exists in a correlated subquery)

select busnum, driverID, RouteID
from bus

where exists (select sh_id from schedule where busnum =bus#)

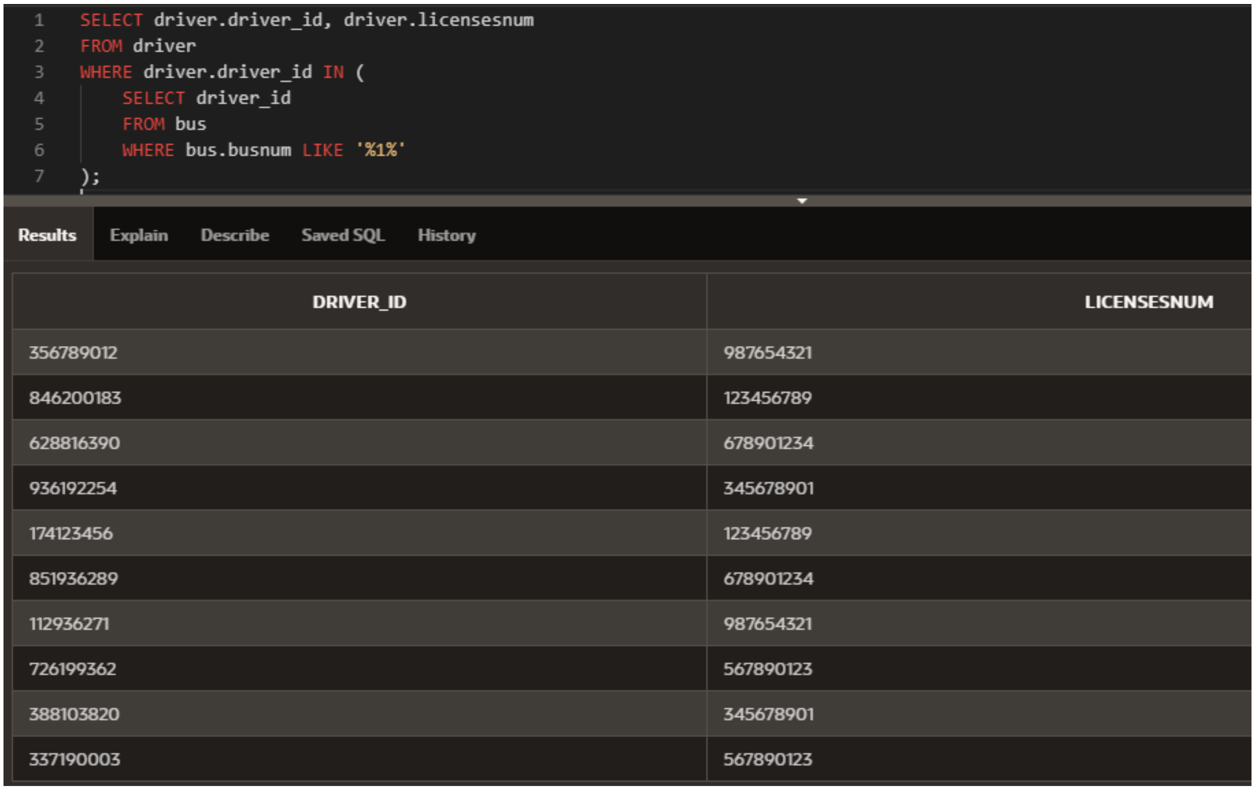


```
Display the bus number,RouteID, and its driver for the bus that follow a schedule timing. (use exists in a correlated subquery)

select busnum, driverID, RouteID
from bus
where exists ( select sh_id from schedule where busnum =bus#)
```

BUSNUM	DRIVERID	ROUTEID
1044	388103820	R45678901
1055	851936289	R56789012
1066	846200183	R67890123
1088	337190003	R89012345

4- Display driver id and license number for any driver assigned to a bus with bus number including the number 1 use any in noncorrelated subquery

```
SELECT driver.driver_id, driver.licensesnum

FROM driver

WHERE driver.driver_id IN (

   SELECT driver_id

   FROM bus

   WHERE bus.busnum LIKE '%1%'

);
```



```
SELECT driver.driver_id, driver.licensesnum
FROM driver
WHERE driver.driver_id IN (
    SELECT driver_id
    FROM bus
    WHERE bus.busnum LIKE '%1%'
);
```

DRIVER_ID	LICENSESNUM
356789012	987654321
846200183	123456789
628816390	678901234
936192254	345678901
174123456	123456789
851936289	678901234
112936271	987654321
726199362	567890123
388103820	345678901
337190003	567890123

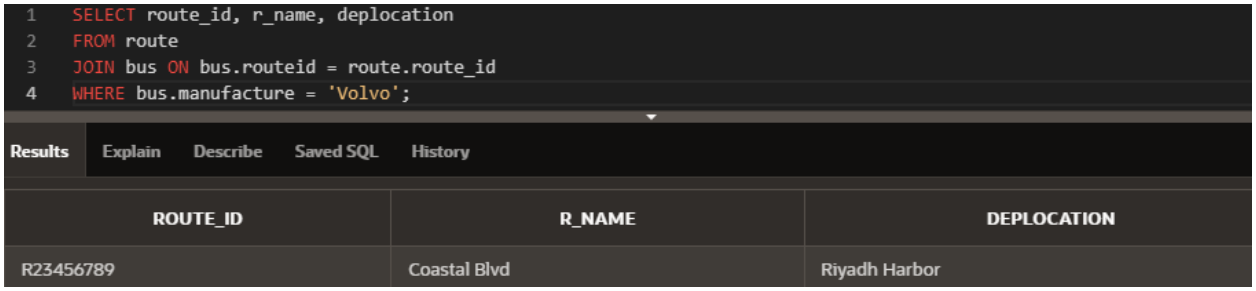
5- Display route id and route name and departure location for every bus with a model manufacture of 'Volvo'

SELECT route_id, r_name, deplocation

FROM route

JOIN bus ON bus.routeid = route.route_id

WHERE bus.manufacture = 'Volvo';



```
SELECT route_id, r_name, deplocation
FROM route
JOIN bus ON bus.routeid = route.route_id
WHERE bus.manufacture = 'Volvo';
```

ROUTE_ID	R_NAME	DEPLOCATION
R23456789	Coastal Blvd	Riyadh Harbor

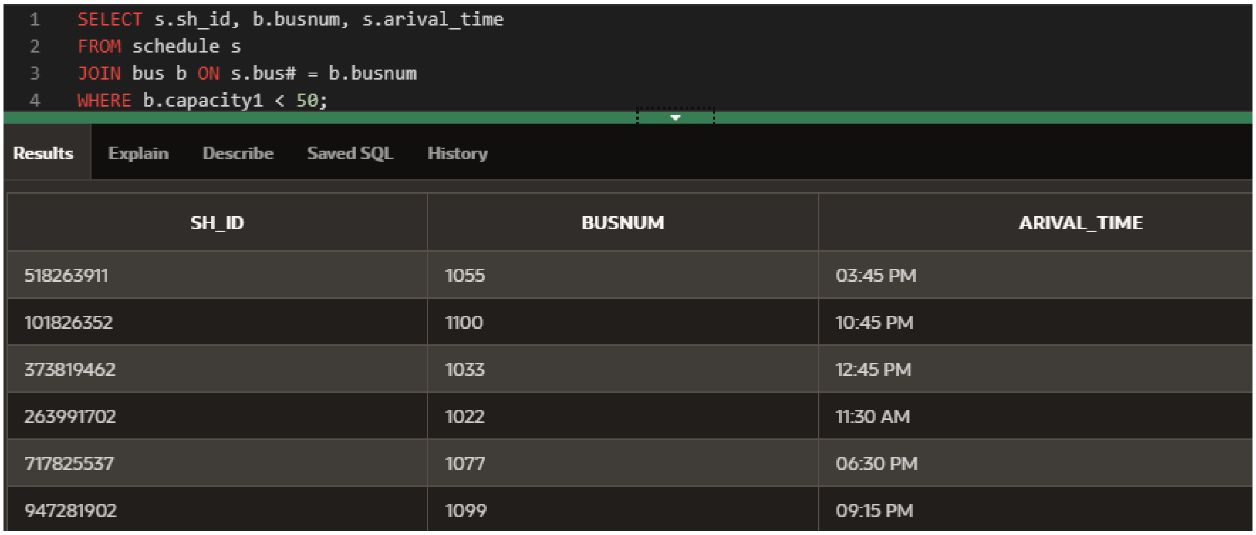
6- Display schedule id, bus number, and arrival time for every bus with capacity of <50

SELECT s.sh_id, b.busnum, s.arival_time

FROM schedule s

JOIN bus b ON s.bus# = b.busnum

WHERE b.capacity1 < 50;



```
SELECT s.sh_id, b.busnum, s.arival_time
FROM schedule s
JOIN bus b ON s.bus# = b.busnum
WHERE b.capacity1 < 50;
```

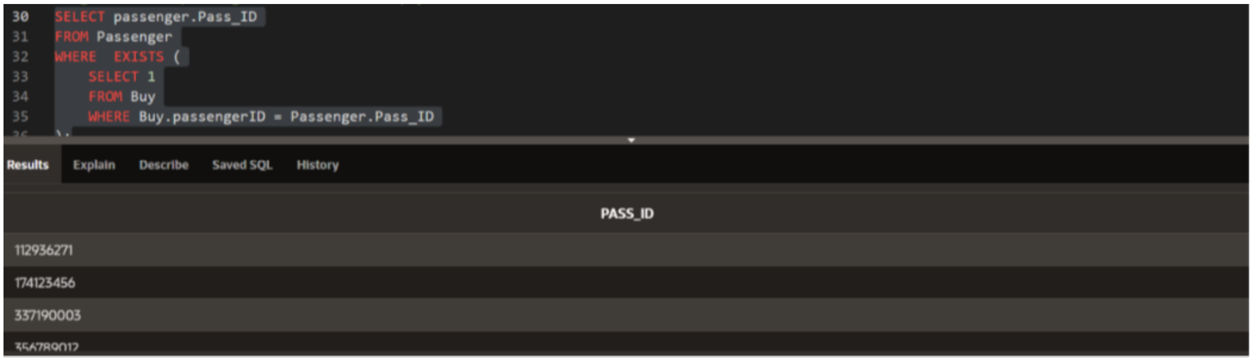
SH_ID	BUSNUM	ARIVAL_TIME
518263911	1055	03:45 PM
101826352	1100	10:45 PM
373819462	1033	12:45 PM
263991702	1022	11:30 AM
717825537	1077	06:30 PM
947281902	1099	09:15 PM

7- Display the passengers who didn't pay for their tickets.

SELECT passenger.Pass_ID
FROM Passenger
WHERE EXISTS (

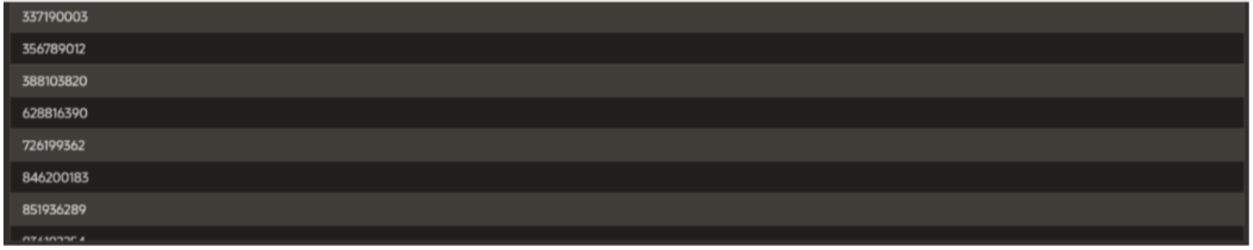
SELECT 1
FROM Buy
WHERE Buy.passengerID = Passenger.Pass_ID

);



```sql
SELECT passenger.Pass_ID
FROM Passenger
WHERE  EXISTS (
    SELECT 1
    FROM Buy
    WHERE Buy.passengerID = Passenger.Pass_ID
);
```

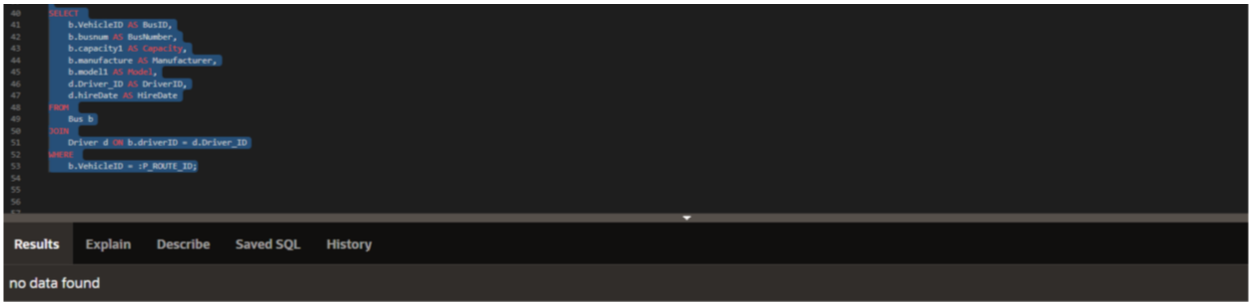
```
PASS_ID
112936271
174123456
337190003
356789012
```



```
337190003
356789012
388103820
628816390
726199362
846200183
851936289
[illegible]
```

8-This query selects specific columns from the "Bus" and "Driver" tables, uses the (join) to join them based on the "driverID" and "Driver_ID" columns, and filters the result based on the "VehicleID" matching the provided ":P_ROUTE_ID"

```sql
SELECT b.VehicleID AS BusID, b.busnum AS BusNumber, b.capacity1 AS Capacity,
b.manufacture AS Manufacturer, b.model1 AS Model,
 d.Driver_ID AS DriverID, d.hireDate AS HireDate

FROM Bus b

JOIN
 Driver d ON b.driverID = d.Driver_ID

WHERE
 b.VehicleID = :P_ROUTE_ID;
```



```
SELECT
    b.VehicleID AS BusID,
    b.busnum AS BusNumber,
    b.capacity1 AS Capacity,
    b.manufacture AS Manufacturer,
    b.model1 AS Model,
    d.Driver_ID AS DriverID,
    d.hireDate AS HireDate
FROM
    Bus b
JOIN
    Driver d ON b.driverID = d.Driver_ID
WHERE
    b.VehicleID = :P_ROUTE_ID;
```

```
no data found
```

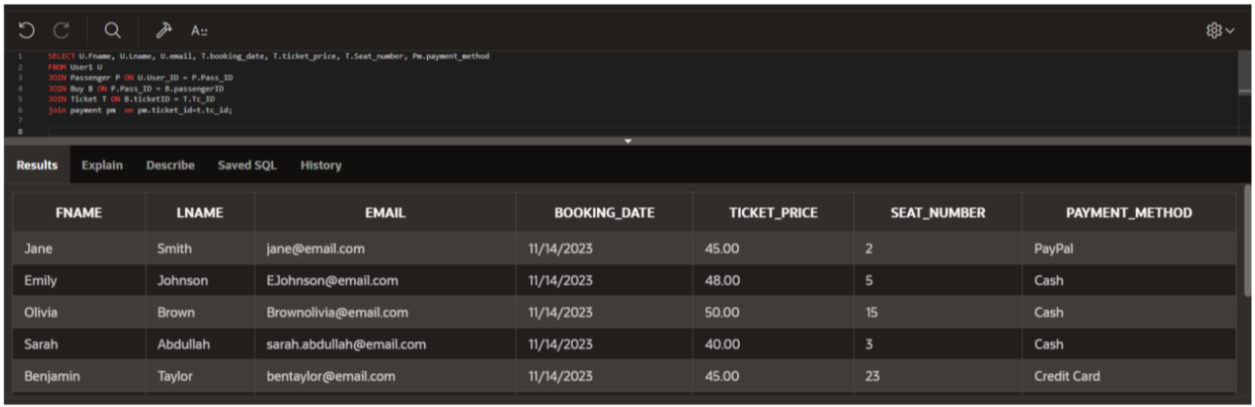
Note

User input is received during runtime. The result I attached had an ID that didn't have any rows in it.

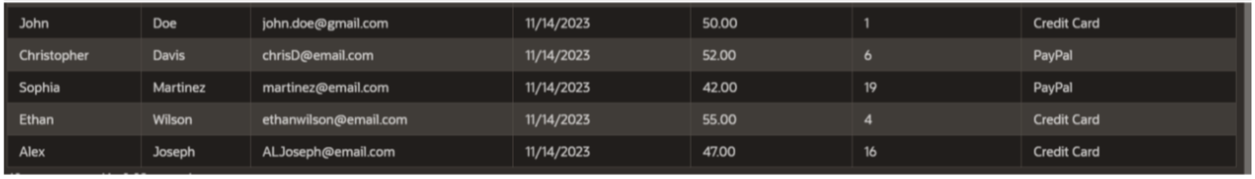
9-Display the first and last name of passengers along with their email who have booked tickets. Include the ticket details such as booking date, price, seat number, and payment method

SELECT U.Fname, U.Lname, U.email, T.booking_date, T.ticket_price, T.Seat_number, Pm.payment_method

FROM User1 U
 JOIN Passenger P ON U.User_ID = P.Pass_ID JOIN Buy B ON P.Pass_ID = B.passengerID
JOIN Ticket T ON B.ticketID = T.Tc_ID
 join payment pm on pm.ticket_id=t.tc_id;



```
SELECT U.Fname, U.Lname, U.email, T.booking_date, T.ticket_price, T.Seat_number, Pm.payment_method
FROM User1 U
JOIN Passenger P ON U.User_ID = P.Pass_ID
JOIN Buy B ON P.Pass_ID = B.passengerID
JOIN Ticket T ON B.ticketID = T.Tc_ID
join payment pm  on pm.ticket_id=t.tc_id;
```

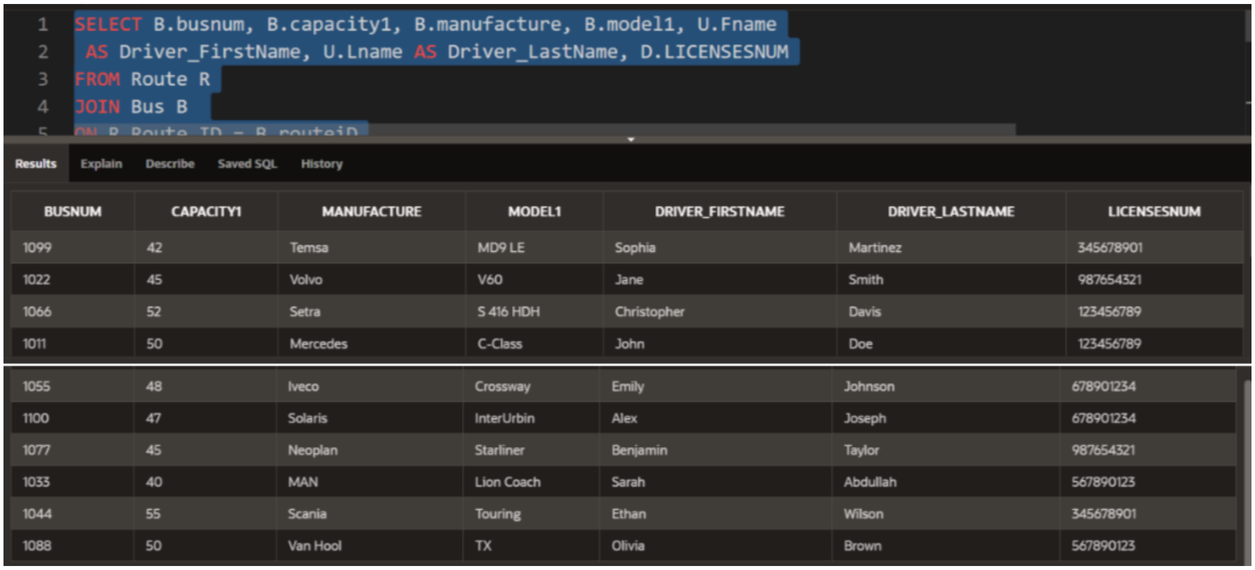


FNAME	LNAME	EMAIL	BOOKING_DATE	TICKET_PRICE	SEAT_NUMBER	PAYMENT_METHOD
Jane	Smith	jane@email.com	11/14/2023	45.00	2	PayPal
Emily	Johnson	EJohnson@email.com	11/14/2023	48.00	5	Cash
Olivia	Brown	Brownolivia@email.com	11/14/2023	50.00	15	Cash
Sarah	Abdullah	sarah.abdullah@email.com	11/14/2023	40.00	3	Cash
Benjamin	Taylor	bentaylor@email.com	11/14/2023	45.00	23	Credit Card
John	Doe	john.doe@gmail.com	11/14/2023	50.00	1	Credit Card
Christopher	Davis	chrisD@email.com	11/14/2023	52.00	6	PayPal
Sophia	Martinez	martinez@email.com	11/14/2023	42.00	19	PayPal
Ethan	Wilson	ethanwilson@email.com	11/14/2023	55.00	4	Credit Card
Alex	Joseph	ALJoseph@email.com	11/14/2023	47.00	16	Credit Card

10- Display the bus details (bus number, capacity, manufacturer, model) and driver details (name, license number) for a specific route (R56789012) on a given date

SELECT B.busnum, B.capacity1, B.manufacture, B.model1, U.Fname AS Driver_FirstName, U.Lname AS Driver_LastName, D.LICENSESNUM

FROM Route R
 JOIN Bus B
 ON R.Route_ID = B.routeiD
 JOIN Schedule S
 ON B.busnum = S.Bus#
 JOIN Driver D
 ON B.driverID = D.Driver_ID
 JOIN User1 U ON D.Driver_ID = U.User_ID
 WHERE R.Route_ID = 'R56789012' or S.departure_time >= '2023-01-01' or S.departure_time <= '2023-01-02';



```sql
SELECT B.busnum, B.capacity1, B.manufacture, B.model1, U.Fname
 AS Driver_FirstName, U.Lname AS Driver_LastName, D.LICENSESNUM
FROM Route R
JOIN Bus B
ON R.Route_ID = B.routeiD
```

BUSNUM	CAPACITY1	MANUFACTURE	MODEL1	DRIVER_FIRSTNAME	DRIVER_LASTNAME	LICENSESNUM
1099	42	Temsa	MD9 LE	Sophia	Martinez	345678901
1022	45	Volvo	V60	Jane	Smith	987654321
1066	52	Setra	S 416 HDH	Christopher	Davis	123456789
1011	50	Mercedes	C-Class	John	Doe	123456789
1055	48	Iveco	Crossway	Emily	Johnson	678901234
1100	47	Solaris	InterUrbin	Alex	Joseph	678901234
1077	45	Neoplan	Starliner	Benjamin	Taylor	987654321
1033	40	MAN	Lion Coach	Sarah	Abdullah	567890123
1044	55	Scania	Touring	Ethan	Wilson	345678901
1088	50	Van Hool	TX	Olivia	Brown	567890123

14.2. ACM Code of Ethics

14.2.1. General Ethical Principles

1.1Contribute to society and human well-being.

1.2 Avoid harm in all forms.

1.3 Be honest and trustworthy.

1.4 Be fair and do not discriminate.

1.5 Respect privacy.

1.6 Achieve confidentiality.

14.2.2. Professional Responsibilities

2.1. strive to achieve high quality in the process and product of professional work.

2.2. Maintain high standards of competence in competence, conduct, and ethical practices.

2.3. Respect and follow existing rules in your work.

2.4. Accept and provide the appropriate professional review.

2.5. Perform work only in the area of competence.

2.6 Design systems that are secure in both usability and robustness.

14.2.3. Professional Leadership Principles

3.1. Ensure that the public good is the central concern.

3.2. Articulate, accept, and evaluate social responsibilities of group members.

3.3. Manage resources and personnel to enhance the quality of working life.

3.4. Create opportunities for members of the organization.

3.5. Use care when changing and modifying the system.

3.6. Apply and support policies that reflect the code principle.

14.2.4. Compliance with the Code

4.1. Promote and respect the principles of the code.

4.2. Treat any violation of the code as inconsistent with membership in the ACM.

Strategy:

One of the most important strategies for demonstrating teamwork is to assign specific tasks and responsibilities to each team member. The team feels more responsibility as a consequence, and everyone is certain to understand their own roles and specialties. Each participant was in charge of her own task throughout the first phase.

Each team member engaged in active communication with the others, sharing insights, details, and critiques. Through regular meetings and instant chat, each participant tracked their progress and provided real-time updates.

Participation needs to be active in order to show teamwork. Each team member contributed ideas, expertise, counsel, and constructive criticism to our project while actively participating in it. Throughout the project, team members encourage and support one another.