

## 2023S – Database Design & SQL

### SQL Project – 8%

---

For the individual assignment I have decided to use the database tables which are from our group project on **Design of Car Rental System**. There are all together eight tables with their respective attributes. Now, first I will write down the queries to create and insert data into the individual table so that we can write other queries later.

#### Create table Customer\_142:

```
CREATE TABLE Customer_142 (
    Driver_License_Number VARCHAR2(20) PRIMARY KEY,
    First_Name VARCHAR2(100) NOT NULL,
    Last_Name VARCHAR2(100) NOT NULL,
    Street VARCHAR2(255) NOT NULL,
    City VARCHAR2(50) NOT NULL,
    Postal_Code VARCHAR2(10) NOT NULL,
    Province VARCHAR2(50) NOT NULL,
    Phone VARCHAR2(15) NOT NULL,
    Email VARCHAR2(150) NOT NULL,
    Member_ID VARCHAR2(20),
    Gender CHAR(1)
);
```

#### Create table Car\_142:

```
CREATE TABLE Car_142 (
    Chassis_Number VARCHAR2(25) PRIMARY KEY,
    Model VARCHAR2(100) NOT NULL,
    Model_Number VARCHAR2(50) NOT NULL,
    Is_Available CHAR(1) NOT NULL,
    Mileage NUMBER(10,2) NOT NULL,
    No_Of_Person INT NOT NULL,
    Price_Per_Day NUMBER(10,2) NOT NULL,
    Late_Fee_Per_Hour NUMBER(10,2) NOT NULL,
    No_Of_Luggage INT NOT NULL,
    Insurance_Code VARCHAR2(20) NOT NULL,
    Make VARCHAR2(100),
    Condition VARCHAR2(50)
);
```

#### Create table Insurance\_142:

```
CREATE TABLE Insurance_142 (
    Insurance_Code VARCHAR2(20) PRIMARY KEY,
    Coverage_Type VARCHAR2(100) NOT NULL,
    Cost_Per_Day NUMBER(10,2) NOT NULL,
    Name VARCHAR2(150)
);
```

**Create table Location \_142:**

```
CREATE TABLE Location_142 (
    Location_ID INT PRIMARY KEY,
    Street VARCHAR2(255) NOT NULL,
    City VARCHAR2(50) NOT NULL,
    Postal_Code VARCHAR2(10) NOT NULL,
    Province VARCHAR2(50) NOT NULL
);
```

**Create table Office \_142:**

```
CREATE TABLE Office_142 (
    Office_ID INT PRIMARY KEY,
    Name VARCHAR2(100) NOT NULL,
    Address VARCHAR2(255) NOT NULL,
    Postal_Code VARCHAR2(10) NOT NULL,
    Province VARCHAR2(50)
);
```

**Create table Employee \_142:**

```
CREATE TABLE Employee_142 (
    Employee_ID INT PRIMARY KEY,
    First_Name VARCHAR2(100) NOT NULL,
    Last_Name VARCHAR2(100) NOT NULL,
    Address VARCHAR2(255) NOT NULL,
    Office_ID INT NOT NULL,
    Gender CHAR(1),
    Age INT
);
```

**Create table Payment \_142:**

```
CREATE TABLE Payment_142 (
    Payment_ID INT PRIMARY KEY,
    Payment_Type VARCHAR2(50) NOT NULL,
    Payment_Due_Date DATE NOT NULL,
    Total_Amount NUMBER(10,2) NOT NULL,
    Tax_Amount NUMBER(10,2) NOT NULL,
    Payment_Status VARCHAR2(50) NOT NULL,
    Driver_License_Number VARCHAR2(20) NOT NULL,
    Contract_ID INT NOT NULL,
    Advance_Amount NUMBER(10,2),
    Cancellation_Charge NUMBER(10,2),
    Late_Fee NUMBER(10,2)
);
```

**Create table Contract \_142:**

```
CREATE TABLE Contract_142 (
    Contract_ID INT PRIMARY KEY,
    Start_Date DATE NOT NULL,
    End_Date DATE NOT NULL,
    Contract_Status VARCHAR2(50) NOT NULL,
    Return_Date DATE NOT NULL,
    Amount NUMBER(10,2) NOT NULL,
    Chassis_Number VARCHAR2(25) NOT NULL,
    Driver_License_Number VARCHAR2(20) NOT NULL,
    Office_ID INT NOT NULL,
    Location_ID INT NOT NULL
);
```

**Insert data into Customer\_142 table:**

```
INSERT INTO Customer_142 (Driver_License_Number, First_Name, Last_Name, Street, City,
Postal_Code, Province, Phone, Email, Member_ID, Gender)
VALUES ('DL12345', 'John', 'Doe', '123 Main St', 'New York', '10001', 'NY', '1234567890',
'john.doe@email.com', 'M123', 'M');
```

**Insert data into Car\_142 table:**

```
INSERT INTO Car_142 (Chassis_Number, Model, Model_Number, Is_Available, Mileage,
No_Of_Person, Price_Per_Day, Late_Fee_Per_Hour, No_Of_Luggage, Insurance_Code, Make,
Condition)
VALUES ('CH12345', 'Sedan', 'S123', 'Y', 50000, 5, 100, 10, 4, 'IC123', 'Toyota', 'New');
```

**Insert data into Insurance\_142 table:**

```
INSERT INTO Insurance_142 (Insurance_Code, Coverage_Type, Cost_Per_Day, Name)
VALUES ('IC123', 'Full', 15, 'Best Insurance');
```

**Insert data into Location\_142 table:**

```
INSERT INTO Location_142 (Location_ID, Street, City, Postal_Code, Province)
VALUES (1, '456 Market St', 'Los Angeles', '90001', 'CA');
```

**Insert data into Office\_142 table:**

```
INSERT INTO Office_142 (Office_ID, Name, Address, Postal_Code, Province)
VALUES (1, 'Main Office', '789 Central St', '20001', 'DC');
```

**Insert data into Employee\_142 table:**

```
INSERT INTO Employee_142 (Employee_ID, First_Name, Last_Name, Address, Office_ID,
Gender, Age)
VALUES (1, 'Jane', 'Smith', '101 Central St', 1, 'F', 30);
```

**Insert data into Payment\_142 table:**

```
INSERT INTO Payment_142 (Payment_ID, Payment_Type, Payment_Due_Date,  
Total_Amount, Tax_Amount, Payment_Status, Driver_License_Number, Contract_ID,  
Advance_Amount, Cancelation_Charge, Late_Fee)  
VALUES (1, 'Credit Card', TO_DATE('2023-08-10', 'YYYY-MM-DD'), 500, 50, 'Pending',  
'DL12345', 1, 100, 20, 10);
```

**Insert data into Contract\_142 table:**

```
INSERT INTO Contract_142 (Contract_ID, Start_Date, End_Date, Contract_Status,  
Return_Date, Amount, Chassis_Number, Driver_License_Number, Office_ID, Location_ID)  
VALUES (1, TO_DATE('2023-08-01', 'YYYY-MM-DD'), TO_DATE('2023-08-10', 'YYYY-  
MM-DD'), 'Active', TO_DATE('2023-08-10', 'YYYY-MM-DD'), 450, 'CH12345', 'DL12345', 1,  
1);
```

Now, let's begin with writing the 50 SQL queries based on the categories provided on assignment:

**1. Select columns from a table:**

```
SELECT First_Name, Last_Name, Email FROM Customer_142;
```

The screenshot shows a 'Live SQL' interface with a 'SQL Worksheet' tab. The worksheet contains the following code:

```
1 | SELECT First_Name, Last_Name, Email FROM Customer_142;
```

Below the worksheet, the results are displayed in a table:

FIRST_NAME	LAST_NAME	EMAIL
John	Doe	john.doe@email.com
Roshan	Shrestha	roshan@email.com
Albert	Adison	albert@email.com
Alice	Brown	alice.brown@email.com

## 2. Computed Columns:

```
SELECT Model, Price_Per_Day * 7 AS Weekly_Price FROM Car_142;
```

The screenshot shows a SQL worksheet interface with the following details:

- Header:** Live SQL, Feedback, Help, shrestharoshan776@gmail.com, a toggle switch.
- Toolbar:** SQL Worksheet, Clear, Find, Actions, Save, Run.
- Query:** SELECT Model, Price\_Per\_Day \* 7 AS Weekly\_Price FROM Car\_142;
- Result:** A table with two columns: MODEL and WEEKLY\_PRICE. The data is as follows:

MODEL	WEEKLY_PRICE
Sedan	700
Maybach	2100
MMY	346.5

## 3. IS NULL:

```
SELECT First_Name, Member_ID FROM Customer_142 WHERE Member_ID IS NULL;
```

The screenshot shows a SQL worksheet interface with the following details:

- Header:** Live SQL, Feedback, Help, shrestharoshan776@gmail.com, a toggle switch.
- Toolbar:** SQL Worksheet, Clear, Find, Actions, Save, Run.
- Query:** SELECT First\_Name, Member\_ID FROM Customer\_142 WHERE Member\_ID IS NULL;
- Result:** A table with two columns: FIRST\_NAME and MEMBER\_ID. The data is as follows:

FIRST_NAME	MEMBER_ID
Roshan	-
Albert	-
Alice	-

#### 4. IS NOT NULL:

```
SELECT First_Name, Member_ID FROM Customer_142 WHERE Member_ID IS NOT NULL;
```

The screenshot shows the 'Live SQL' interface with the following details:

- Header:** Includes 'Live SQL' logo, 'Feedback', 'Help', user email 'shrestharoshan776@gmail.com', and a dark mode toggle.
- Toolbar:** Contains 'Clear', 'Find', 'Actions', 'Save', and a 'Run' button.
- SQL Worksheet:** Displays the query: `SELECT First_Name, Member_ID FROM Customer_142 WHERE Member_ID IS NOT NULL;`.
- Results:** A table with columns 'FIRST\_NAME' and 'MEMBER\_ID'. The data row is: John | M123.

#### 5. String concatenation:

```
SELECT First_Name || ' ' || Last_Name AS Full_Name FROM Customer_142;
```

The screenshot shows the 'Live SQL' interface with the following details:

- Header:** Includes 'Live SQL' logo, 'Feedback', 'Help', user email 'shrestharoshan776@gmail.com', and a dark mode toggle.
- Toolbar:** Contains 'Clear', 'Find', 'Actions', 'Save', and a 'Run' button.
- SQL Worksheet:** Displays the query: `SELECT First_Name || ' ' || Last_Name AS Full_Name FROM Customer_142;`.
- Results:** A table with a single column 'FULL\_NAME'. The data rows are: John Doe, Roshan Shrestha, Albert Adison, and Alice Brown.

## 6. Distinct Keyword:

```
SELECT DISTINCT City FROM Customer_142;
```

The screenshot shows a SQL worksheet interface with the following details:

- Header:** Live SQL, Feedback, Help, shrestharoshan776@gmail.com, Dark Mode switch.
- Toolbar:** SQL Worksheet, Clear, Find, Actions, Save, Run.
- SQL Worksheet Area:** Contains the query: `SELECT DISTINCT City FROM Customer_142;`
- Result Area:** A table titled "CITY" with five rows:
  - Boston
  - New York
  - New Jersey
  - New Buffalo

## 7. NOT operator:

```
SELECT Model FROM Car_142 WHERE NOT Model = 'Sedan';
```

The screenshot shows a SQL worksheet interface with the following details:

- Header:** Live SQL, Feedback, Help, shrestharoshan776@gmail.com, Dark Mode switch.
- Toolbar:** SQL Worksheet, Clear, Find, Actions, Save, Run.
- SQL Worksheet Area:** Contains the query: `SELECT Model FROM Car_142 WHERE NOT Model = 'Sedan';`
- Result Area:** A table titled "MODEL" with two rows:
  - Maybach
  - MMY

## 8. BETWEEN Operator:

```
SELECT First_Name, Last_Name FROM Customer_142 WHERE Postal_Code  
BETWEEN '10000' AND '20000';
```

The screenshot shows the Live SQL interface with the following details:

- Toolbar:** Includes "Live SQL" icon, "Feedback" button, "Help" button, user info "shrestharoshan776@gmail.com", and a dark mode switch.
- SQL Worksheet:** Labeled "SQL Worksheet". Contains the SQL query: `SELECT First_Name, Last_Name FROM Customer_142 WHERE Postal_Code BETWEEN '10000' AND '20000';`.
- Result Area:** A table with columns "FIRST\_NAME" and "LAST\_NAME". The data row is John Doe.

## 9. IN Operator:

```
SELECT Model FROM Car_142 WHERE Make IN ('Toyota', 'Honda');
```

The screenshot shows the Live SQL interface with the following details:

- Toolbar:** Includes "Live SQL" icon, "Feedback" button, "Help" button, user info "shrestharoshan776@gmail.com", and a dark mode switch.
- SQL Worksheet:** Labeled "SQL Worksheet". Contains the SQL query: `SELECT Model FROM Car_142 WHERE Make IN ('Toyota', 'Honda');`.
- Result Area:** A table with column "MODEL". The data row is Sedan.

## 10. NOT IN:

```
SELECT Model FROM Car_142 WHERE Make NOT IN ('Toyota', 'Honda');
```

The screenshot shows a SQL worksheet interface. At the top, there are navigation links for 'Live SQL', 'Feedback', 'Help', and a user account. Below the header, the title 'SQL Worksheet' is displayed along with buttons for 'Clear', 'Find', 'Actions', 'Save', and a green 'Run' button. The main area contains the following code:

```
1 SELECT Model FROM Car_142 WHERE Make NOT IN ('Toyota', 'Honda');
2
```

On the right side, there is a vertical list labeled 'MODEL' containing the results: 'Maybach' and 'MMY'. The background features a decorative pattern.

## 11. LIKE Operator:

```
SELECT First_Name FROM Customer_142 WHERE First_Name LIKE 'J%';
```

The screenshot shows a SQL worksheet interface. At the top, there are navigation links for 'Live SQL', 'Feedback', 'Help', and a user account. Below the header, the title 'SQL Worksheet' is displayed along with buttons for 'Clear', 'Find', 'Actions', 'Save', and a green 'Run' button. The main area contains the following code:

```
1 SELECT First_Name FROM Customer_142 WHERE First_Name LIKE 'J%';
2
```

On the right side, there is a vertical list labeled 'FIRST\_NAME' containing the result: 'John'. The background features a decorative pattern.

## 12. Comparison Operator (<,<=,=):

```
SELECT Model, Mileage FROM Car_142 WHERE Mileage > 30000 AND Price_Per_Day <= 100;
```

The screenshot shows a "Live SQL" interface with a "SQL Worksheet" tab. The query entered is:

```
1 SELECT Model, Mileage FROM Car_142 WHERE Mileage > 30000 AND Price_Per_Day <= 100;
```

The results are displayed in a table:

MODEL	MILEAGE
Sedan	50000

### 13. Comparision Operator (=):

```
SELECT First_Name, Last_Name FROM Customer_142 WHERE Gender = 'M';
```

The screenshot shows a "Live SQL" interface with a "SQL Worksheet" tab. The query entered is:

```
1 SELECT First_Name, Last_Name FROM Customer_142 WHERE Gender = 'M';
```

The results are displayed in a table:

FIRST_NAME	LAST_NAME
John	Doe
Roshan	Shrestha
Albert	Adison

### 14. LIKE and IN operator:

```
SELECT First_Name, Last_Name FROM Customer_142 WHERE Last_Name LIKE 'A%' AND First_Name IN ('Alice', 'Albert');
```

The screenshot shows a SQL worksheet interface. At the top, there are navigation links for 'Live SQL', 'Feedback', 'Help', and a user profile. Below that is a toolbar with 'Clear', 'Find', 'Actions', 'Save', and a 'Run' button. The main area contains the following SQL code:

```
1 SELECT First_Name, Last_Name FROM Customer_142 WHERE Last_Name LIKE 'A%' AND First_Name IN ('Alice', 'Albert')
```

Below the code, the results are displayed in a table:

FIRST_NAME	LAST_NAME
Albert	Adison

### 15. AND NOT Operator:

```
SELECT First_Name, Last_Name FROM Customer_142 WHERE City = 'New York'  
AND NOT Gender = 'F';
```

The screenshot shows a SQL worksheet interface. At the top, there are navigation links for 'Live SQL', 'Feedback', 'Help', and a user profile. Below that is a toolbar with 'Clear', 'Find', 'Actions', 'Save', and a 'Run' button. The main area contains the following SQL code:

```
1 SELECT First_Name, Last_Name FROM Customer_142 WHERE City = 'New York' AND NOT Gender = 'F';
```

Below the code, the results are displayed in a table:

FIRST_NAME	LAST_NAME
John	Doe

### 16. AND Operator:

```
SELECT Model FROM Car_142 WHERE (Make = 'Toyota' OR Make = 'Honda') AND Mileage > 30000;
```

The screenshot shows the 'Live SQL' interface. At the top, there are navigation links for 'Feedback', 'Help', and a user account (shrestharoshan776@gmail.com). Below that is a toolbar with 'Clear', 'Find', 'Actions', 'Save', and a 'Run' button. The main area contains the SQL query:

```
1 SELECT Model FROM Car_142 WHERE (Make = 'Toyota' OR Make = 'Honda') AND Mileage > 30000;
```

Below the query results, a table named 'MODEL' is displayed with one row: 'Sedan'.

### 17. ORDER BY operator:

```
SELECT First_Name, Last_Name FROM Customer_142 ORDER BY Last_Name ASC;
```

The screenshot shows the 'Live SQL' interface. At the top, there are navigation links for 'Feedback', 'Help', and a user account (shrestharoshan776@gmail.com). Below that is a toolbar with 'Clear', 'Find', 'Actions', 'Save', and a 'Run' button. The main area contains the SQL query:

```
1 SELECT First_Name, Last_Name FROM Customer_142 ORDER BY Last_Name ASC;
```

Below the query results, a table named 'FIRST\_NAME LAST\_NAME' is displayed with four rows:

FIRST_NAME	LAST_NAME
Albert	Adison
Alice	Brown
John	Doe
Roshan	Shrestha

### 18. AVG function:

```
SELECT AVG(Price_Per_Day) AS Avg_Price FROM Car_142 WHERE Make = 'Toyota';
```

The screenshot shows a 'Live SQL' interface. At the top, there are navigation links for Feedback, Help, and a user account (shrestharoshan776@gmail.com). Below that is a toolbar with 'Clear', 'Find', 'Actions', 'Save', and a 'Run' button. The main area is titled 'SQL Worksheet'. A code editor contains the following SQL query:

```
1 SELECT AVG(Price_Per_Day) AS Avg_Price FROM Car_142 WHERE Make = 'Toyota';
2
```

Below the code editor is a results table with one row:

AVG_PRICE
100

### 19. COUNT function:

```
SELECT COUNT(DISTINCT Make) as Unique_Makes, SUM(Price_Per_Day) as Total_Daily_Price, AVG(Mileage) as Average_Mileage FROM Car_142;
```

The screenshot shows a 'Live SQL' interface. At the top, there are navigation links for Feedback, Help, and a user account (shrestharoshan776@gmail.com). Below that is a toolbar with 'Clear', 'Find', 'Actions', 'Save', and a 'Run' button. The main area is titled 'SQL Worksheet'. A code editor contains the following SQL query:

```
1 v SELECT COUNT(DISTINCT Make) as Unique_Makes, SUM(Price_Per_Day)
2      as Total_Daily_Price, AVG(Mileage)
3      as Average_Mileage
4  FROM Car_142;
5
```

Below the code editor is a results table with three columns:

UNIQUE_MAKES	TOTAL_DAILY_PRICE	AVERAGE_MILEAGE
2	449.5	26333.5

## 20. GROUP BY operator:

```
SELECT Province, COUNT(*) AS Customer_Count FROM Customer_142 GROUP BY Province;
```

The screenshot shows a SQL worksheet interface with the following details:

- Toolbar:** Includes "Live SQL" icon, "Feedback", "Help", user info "shrestharoshan776@gmail.com", and a dark mode switch.
- SQL Worksheet:** Shows the executed SQL code:

```
1 v SELECT Province, COUNT(*) AS Customer_Count
2   FROM Customer_142
3     GROUP BY Province;
```
- Results:** A table titled "SQL Results" displays the output:

PROVINCE	CUSTOMER_COUNT
BC	1
MA	1
VC	1
NY	1

## 21. Group By Having:

```
SELECT Office_ID, AVG(Age) AS Avg_Age FROM Employee_142 GROUP BY Office_ID HAVING AVG(Age) > 30;
```

The screenshot shows a SQL worksheet interface with the following details:

- Toolbar:** Includes "Live SQL" icon, "Feedback", "Help", user info "shrestharoshan776@gmail.com", and a dark mode switch.
- SQL Worksheet:** Shows the executed SQL code:

```
1 v SELECT Office_ID, AVG(Age) AS Avg_Age
2   FROM Employee_142
3     GROUP BY Office_ID
4       HAVING AVG(Age) > 30;
```
- Results:** A table titled "SQL Results" displays the output:

OFFICE_ID	AVG_AGE
3	35

## 22. TO\_CHAR() function:

```
SELECT TO_CHAR(Start_Date, 'DD-MON-YYYY') AS Formatted_Start_Date FROM Contract_142 WHERE ROWNUM = 1;
```

The screenshot shows the Oracle SQL Developer interface. At the top, there's a navigation bar with 'Live SQL' selected. Below it is a 'SQL Worksheet' tab. The code area contains the following SQL statement:

```
1 v SELECT TO_CHAR(Start_Date, 'DD-MON-YYYY') AS Formatted_Start_Date
2   FROM Contract_142
3 WHERE ROWNUM = 1;
```

Below the code, the results are displayed in a table with one row:

FORMATTED_START_DATE
01-AUG-2023

## 23. LOWER function:

```
SELECT LOWER(First_Name) AS Lowercase_Name FROM Customer_142 WHERE ROWNUM = 1;
```

The screenshot shows the Oracle SQL Developer interface. At the top, there's a navigation bar with 'Live SQL' selected. Below it is a 'SQL Worksheet' tab. The code area contains the following SQL statement:

```
1 v SELECT LOWER(First_Name) AS Lowercase_Name
2   FROM Customer_142
3 WHERE ROWNUM = 1;
```

Below the code, the results are displayed in a table with one row:

LOWERCASE_NAME
john

#### 24. UPPER function:

```
SELECT UPPER(Last_Name) AS Uppercase_Last_Name FROM Customer_142  
WHERE ROWNUM = 1;
```

The screenshot shows the 'Live SQL' interface with the following details:

- Header:** Shows 'Live SQL' with a red circle icon, 'Feedback', 'Help', and a user email 'shrestharoshan776@gmail.com'. A dark mode switch is also present.
- Toolbar:** Includes 'Clear', 'Find', 'Actions', 'Save', and a 'Run' button.
- SQL Worksheet:** Displays the SQL query:

```
1 SELECT UPPER(Last_Name) AS Uppercase_Last_Name  
2 FROM Customer_142  
3 WHERE ROWNUM = 1;
```
- Results:** A table with one row labeled 'UPPERCASE\_LAST\_NAME' containing the value 'DOE'.

#### 25. INITCAP function:

```
SELECT INITCAP(Street) AS Proper_Street FROM Customer_142 WHERE ROWNUM  
= 1;=
```

The screenshot shows the 'Live SQL' interface with the following details:

- Header:** Shows 'Live SQL' with a red circle icon, 'Feedback', 'Help', and a user email 'shrestharoshan776@gmail.com'. A dark mode switch is also present.
- Toolbar:** Includes 'Clear', 'Find', 'Actions', 'Save', and a 'Run' button.
- SQL Worksheet:** Displays the SQL query:

```
1 SELECT INITCAP(Street) AS Proper_Street  
2 FROM Customer_142  
3 WHERE ROWNUM = 1;=
```
- Results:** A table with one row labeled 'PROPER\_STREET' containing the value '123 Main St'.

## 26. CONCAT function:

```
SELECT CONCAT(First_Name, CONCAT(' ', Last_Name)) AS Full_Name FROM Customer_142 WHERE ROWNUM = 1;
```

The screenshot shows a SQL worksheet interface with the following details:

- Header:** Live SQL, Feedback, Help, shrestharoshan776@gmail.com, Run button.
- Toolbar:** Clear, Find, Actions, Save, Run.
- SQL Worksheet Area:** Contains the following SQL code:

```
1 SELECT CONCAT(First_Name, CONCAT(' ', Last_Name)) AS Full_Name
2 FROM Customer_142
3 WHERE ROWNUM = 1;
```
- Output Area:** Shows the results of the query in a table format:

FULL_NAME
John Doe

## 27. SUBSTR function:

```
SELECT SUBSTR(Phone, 1, 3) AS Area_Code FROM Customer_142 WHERE ROWNUM = 1;
```

The screenshot shows a SQL worksheet interface with the following details:

- Header:** Live SQL, Feedback, Help, shrestharoshan776@gmail.com, Run button.
- Toolbar:** Clear, Find, Actions, Save, Run.
- SQL Worksheet Area:** Contains the following SQL code:

```
1 SELECT SUBSTR(Phone, 1, 3) AS Area_Code
2 FROM Customer_142
3 WHERE ROWNUM = 1;
```
- Output Area:** Shows the results of the query in a table format:

AREA_CODE
123

## 28. INSTR function:

```
SELECT INSTR(Email, '@') AS At_Symbol_Position FROM Customer_142 WHERE ROWNUM = 1;
```

The screenshot shows the Oracle SQL Developer interface with a "Live SQL" session. The SQL Worksheet contains the following query:

```
1 SELECT INSTR(Email, '@') AS At_Symbol_Position
2 FROM Customer_142
3 WHERE ROWNUM = 1;
```

The results pane below shows a single row with the column name "AT\_SYMBOL\_POSITION" and the value "9".

## 29. TRIM function:

```
SELECT TRIM('A' FROM 'AABBBA') AS Trimmed_String FROM Customer_142 WHERE ROWNUM = 1;
```

The screenshot shows the Oracle SQL Developer interface with a "Live SQL" session. The SQL Worksheet contains the following query:

```
1 SELECT TRIM('A' FROM 'AABBBA') AS Trimmed_String
2 FROM Customer_142
3 WHERE ROWNUM = 1;
```

The results pane below shows a single row with the column name "TRIMMED\_STRING" and the value "BBB".

### 30. RPAD function:

```
SELECT RPAD(First_Name, 10, '*') AS Padded_Name FROM Customer_142 WHERE ROWNUM = 1;
```

The screenshot shows the Oracle Live SQL interface. At the top, there are navigation links for Feedback, Help, and a user account (shrestharoshan776@gmail.com). On the right, there is a dark mode toggle switch. Below the header is a toolbar with buttons for Clear, Find, Actions (dropdown), Save, and Run. The main area is titled "SQL Worksheet". The SQL code entered is:

```
1 v SELECT RPAD(First_Name, 10, '*') AS Padded_Name
2 FROM Customer_142
3 WHERE ROWNUM = 1;
```

Below the code, the results are displayed in a table with one row:

PADDDED_NAME
John*****

### 31. ROUND function:

```
SELECT ROUND(Price_Per_Day, 1) AS Rounded_Price FROM Car_142 WHERE ROWNUM = 1;
```

The screenshot shows the Oracle Live SQL interface. At the top, there are navigation links for Feedback, Help, and a user account (shrestharoshan776@gmail.com). On the right, there is a dark mode toggle switch. Below the header is a toolbar with buttons for Clear, Find, Actions (dropdown), Save, and Run. The main area is titled "SQL Worksheet". The SQL code entered is:

```
1 v SELECT ROUND(Price_Per_Day, 1) AS Rounded_Price
2 FROM Car_142
3 WHERE ROWNUM = 1;
```

Below the code, the results are displayed in a table with one row:

ROUNDED_PRICE
100

### 32. CELI function:

```
SELECT CEIL(Mileage) AS Rounded_Mileage FROM Car_142 WHERE ROWNUM = 1;
```

The screenshot shows a SQL worksheet interface. At the top, there are navigation links for 'Feedback', 'Help', and a user account (shrestharoshan776@gmail.com). Below that is a toolbar with 'Clear', 'Find', 'Actions', 'Save', and a 'Run' button. The main area is titled 'SQL Worksheet' and contains the following SQL code:

```
1 v SELECT CEIL(Mileage) AS Rounded_Mileage
2   FROM Car_142
3 WHERE ROWNUM = 1;
```

Below the code, the results are displayed in a table:

ROUNDED_MILEAGE
50000

### 33. NVL function:

```
SELECT NVL(Member_ID, 'No Member') AS Membership_Status FROM Customer_142 WHERE ROWNUM = 1;
```

The screenshot shows a SQL worksheet interface. At the top, there are navigation links for 'Feedback', 'Help', and a user account (shrestharoshan776@gmail.com). Below that is a toolbar with 'Clear', 'Find', 'Actions', 'Save', and a 'Run' button. The main area is titled 'SQL Worksheet' and contains the following SQL code:

```
1 v SELECT NVL(Member_ID, 'No Member') AS Membership_Status
2   FROM Customer_142
3 WHERE ROWNUM = 1;
```

Below the code, the results are displayed in a table:

MEMBERSHIP_STATUS
M123

### 34. Inner Join:

```
SELECT c.First_Name, c.Last_Name, car.Model FROM Customer_142 c JOIN  
Contract_142 con ON c.Driver_License_Number = con.Driver_License_Number JOIN  
Car_142 car ON con.Chassis_Number = car.Chassis_Number WHERE ROWNUM = 1;
```

The screenshot shows the 'Live SQL' interface with the following details:

- Toolbar:** Includes 'Feedback', 'Help', a user profile for 'shrestharoshan776@gmail.com', and a toggle switch.
- SQL Worksheet:** Labeled 'SQL Worksheet' with a 'Run' button. It contains the SQL code for the inner join query.
- Results:** A table with three columns: FIRST\_NAME, LAST\_NAME, and MODEL. The data row is John Doe Sedan.

### 35. Left Outer Join:

```
SELECT c.First_Name, c.Last_Name, con.Contract_ID FROM Customer_142 c LEFT  
JOIN Contract_142 con ON c.Driver_License_Number = con.Driver_License_Number  
WHERE ROWNUM = 1;
```

The screenshot shows the 'Live SQL' interface with the following details:

- Toolbar:** Includes 'Feedback', 'Help', a user profile for 'shrestharoshan776@gmail.com', and a toggle switch.
- SQL Worksheet:** Labeled 'SQL Worksheet' with a 'Run' button. It contains the SQL code for the left outer join query.
- Results:** A table with three columns: FIRST\_NAME, LAST\_NAME, and CONTRACT\_ID. The data row is John Doe 1.

### 36. Right Outer Join:

```
SELECT c.First_Name, con.Contract_ID, con.Start_Date FROM Customer_142 c  
RIGHT JOIN Contract_142 con ON c.Driver_License_Number =  
con.Driver_License_Number WHERE ROWNUM = 1;
```

The screenshot shows the Oracle SQL Developer Live SQL interface. The top navigation bar includes 'Live SQL' (with a red circle icon), 'Feedback', 'Help', and a user email 'shrestharoshan776@gmail.com'. A dark mode switch is also present. Below the header is a toolbar with 'Clear', 'Find', 'Actions', 'Save', and a green 'Run' button with a play icon. The main area is titled 'SQL Worksheet' and contains the following SQL code:

```
1 v SELECT c.First_Name, con.Contract_ID, con.Start_Date  
2 FROM Customer_142 c  
3 RIGHT JOIN Contract_142 con ON c.Driver_License_Number = con.Driver_License_Number  
4 WHERE ROWNUM = 1;
```

Below the code, the results are displayed in a table:

FIRST_NAME	CONTRACT_ID	START_DATE
John	1	01-AUG-23

### 37. Cross Join:

```
SELECT c.First_Name, o.Name FROM Customer_142 c CROSS JOIN Office_142 o  
WHERE ROWNUM = 1;
```

The screenshot shows the Oracle SQL Developer Live SQL interface. The top navigation bar includes 'Live SQL' (with a red circle icon), 'Feedback', 'Help', and a user email 'shrestharoshan776@gmail.com'. A dark mode switch is also present. Below the header is a toolbar with 'Clear', 'Find', 'Actions', 'Save', and a green 'Run' button with a play icon. The main area is titled 'SQL Worksheet' and contains the following SQL code:

```
1 v SELECT c.First_Name, o.Name  
2 FROM Customer_142 c  
3 CROSS JOIN Office_142 o  
4 WHERE ROWNUM = 1;
```

Below the code, the results are displayed in a table:

FIRST_NAME	NAME
John	Main Office

### 38. Self-Join:

```
SELECT e1.First_Name AS Employee1, e2.First_Name AS Employee2 FROM  
Employee_142 e1 JOIN Employee_142 e2 ON e1.Office_ID = e2.Office_ID WHERE  
ROWNUM <= 1;
```

The screenshot shows a "Live SQL" interface with a "SQL Worksheet" tab. The code input area contains the following SQL query:

```
1 v SELECT e1.First_Name AS Employee1, e2.First_Name AS Employee2  
2   FROM Employee_142 e1  
3   JOIN Employee_142 e2 ON e1.Office_ID = e2.Office_ID  
4  WHERE ROWNUM <= 1;
```

The results pane displays a table with two columns: "EMPLOYEE1" and "EMPLOYEE2". Both columns contain the value "Jane".

EMPLOYEE1	EMPLOYEE2
Jane	Jane

### 39. Single row subqueries:

```
SELECT Chassis_Number, Model, Price_Per_Day FROM Car_142 WHERE  
Price_Per_Day < (SELECT AVG(Price_Per_Day) FROM Car_142);
```

The screenshot shows a "Live SQL" interface with a "SQL Worksheet" tab. The code input area contains the following SQL query:

```
1 v SELECT Chassis_Number, Model, Price_Per_Day  
2   FROM Car_142  
3  WHERE Price_Per_Day < (SELECT AVG(Price_Per_Day) FROM Car_142);
```

The results pane displays a table with three columns: "CHASSIS\_NUMBER", "MODEL", and "PRICE\_PER\_DAY". The data is as follows:

CHASSIS_NUMBER	MODEL	PRICE_PER_DAY
CH12345	Sedan	100
CH12005	MMY	49.5

#### 40. EXISTS operator:

```
SELECT c.First_Name, c.Last_Name FROM Customer_142 c WHERE EXISTS  
(SELECT 1 FROM Contract_142 con WHERE con.Driver_License_Number =  
c.Driver_License_Number);
```

The screenshot shows a SQL worksheet interface with the following details:

- Header:** Live SQL, Feedback, Help, shrestharoshan776@gmail.com, Run button.
- SQL Worksheet:** Contains the SQL query from above.
- Results:** A table titled "Customer\_142" with columns FIRST\_NAME and LAST\_NAME, showing three rows: John Doe, Roshan Shrestha, and Alice Brown.

#### 41. NOT EXISTS operator:

```
SELECT c.First_Name, c.Last_Name FROM Customer_142 c WHERE NOT EXISTS  
(SELECT 1 FROM Contract_142 con WHERE con.Driver_License_Number =  
c.Driver_License_Number);
```

The screenshot shows a SQL worksheet interface with the following details:

- Header:** Live SQL, Feedback, Help, shrestharoshan776@gmail.com, Run button.
- SQL Worksheet:** Contains the SQL query from above.
- Results:** A table titled "Customer\_142" with columns FIRST\_NAME and LAST\_NAME, showing one row: Albert Adison.

## 42. UNION operator:

```
SELECT First_Name, Last_Name FROM Customer_142 UNION SELECT First_Name, Last_Name FROM Employee_142;
```

The screenshot shows a SQL worksheet interface with the title "Live SQL". The code entered is:

```
1 v SELECT First_Name, Last_Name FROM Customer_142
2 UNION
3 SELECT First_Name, Last_Name FROM Employee_142;
```

The results are displayed in a table:

FIRST_NAME	LAST_NAME
Albert	Adison
Alice	Brown
David	Miller
Jane	Smith
John	Doe
Michael	Johnson
Roshan	Shrestha

## 43. INTERSECT operator:

```
SELECT First_Name FROM Customer_142 WHERE Gender = 'M' INTERSECT
SELECT First_Name FROM Customer_142 WHERE Last_Name LIKE '%son%';
```

The screenshot shows a SQL worksheet interface with the title "Live SQL". The code entered is:

```
1 v SELECT First_Name
2   FROM Customer_142
3 WHERE Gender = 'M'
4 INTERSECT
5 SELECT First_Name
6   FROM Customer_142
7 WHERE Last_Name LIKE '%son%';
```

The results are displayed in a table:

FIRST_NAME
Albert

#### 44. MINUS operator:

```
SELECT First_Name FROM Customer_142 WHERE Gender = 'M' MINUS SELECT First_Name FROM Customer_142 WHERE Last_Name LIKE '%son%';
```

The screenshot shows the Live SQL interface with the following details:

- Header:** Live SQL, Feedback, Help, shrestharoshan776@gmail.com, a toggle switch.
- Toolbar:** SQL Worksheet, Clear, Find, Actions, Save, Run.
- SQL Worksheet:** The code is:

```
1 v SELECT First_Name
2   FROM Customer_142
3   WHERE Gender = 'M'
4 MINUS
5   SELECT First_Name
6   FROM Customer_142
7   WHERE Last_Name LIKE '%son%';
```
- Results:** A table titled "FIRST\_NAME" with two rows:

FIRST_NAME
John
Roshan

#### 45. Query to List Cars with Most Rentals:

```
SELECT Car.Model FROM Car_142 Car WHERE Car.Chassis_Number IN ( SELECT Chassis_Number FROM Contract_142 GROUP BY Chassis_Number HAVING COUNT() = ( SELECT MAX(Rentals) FROM ( SELECT Chassis_Number, COUNT() AS Rentals FROM Contract_142 GROUP BY Chassis_Number ) ));
```

The screenshot shows the Live SQL interface with the following details:

- Header:** Live SQL, Feedback, Help, shrestharoshan776@gmail.com, a toggle switch.
- Toolbar:** SQL Worksheet, Clear, Find, Actions, Save, Run.
- SQL Worksheet:** The code is:

```
3 WHERE Car.Chassis_Number IN (
4   SELECT Chassis_Number
5   FROM Contract_142
6   GROUP BY Chassis_Number
7   HAVING COUNT(*) = (
8     SELECT MAX(Rentals)
9     FROM (
10       SELECT Chassis_Number, COUNT(*) AS Rentals
11       FROM Contract_142
12       GROUP BY Chassis_Number
13     )
14   )
15 );
16
```
- Results:** A table titled "MODEL" with three rows:

MODEL
MMY
Maybach
Sedan

#### 46. Query to Show Total Payments for Each Customer:

```
SELECT C.First_Name, C.Last_Name, P.Total_Amount FROM Customer_142 C JOIN  
Payment_142 P ON C.Driver_License_Number = P.Driver_License_Number WHERE  
P.Total_Amount >= ALL ( SELECT MAX(Total_Amount) FROM Payment_142 GROUP  
BY Driver_License_Number );
```

The screenshot shows a SQL worksheet interface with the following elements:

- Header:** Includes "Live SQL" logo, "Feedback" link, "Help" link, user email "shrestharoshan776@gmail.com", and a dark mode toggle.
- Toolbar:** Buttons for "Clear", "Find", "Actions", "Save", and a large green "Run" button with a play icon.
- SQL Worksheet Area:** Displays the SQL query code:

```
1 v SELECT C.First_Name, C.Last_Name, P.Total_Amount  
2   FROM Customer_142 C  
3   JOIN Payment_142 P ON C.Driver_License_Number = P.Driver_License_Number  
4 WHERE P.Total_Amount >= ALL (  
5     SELECT MAX(Total_Amount)  
6     FROM Payment_142  
7     GROUP BY Driver_License_Number  
8 );  
9
```
- Results Area:** A table showing the output of the query:

FIRST_NAME	LAST_NAME	TOTAL_AMOUNT
John	Doe	500

**47. Query to show customer's first and last name who has at least one contract:**

```
SELECT C.First_Name, C.Last_Name FROM Customer_142 C WHERE NOT EXISTS (
    SELECT 1 FROM Contract_142 CO WHERE CO.Driver_License_Number =
        C.Driver_License_Number);
```

The screenshot shows a SQL worksheet interface with the following elements:

- Header:** "Live SQL" button, "Feedback" link, "Help" link, user email "shrestharoshan776@gmail.com", and a dark mode toggle switch.
- Toolbar:** "SQL Worksheet" tab, "Clear" button, "Find" button, "Actions" dropdown, "Save" button, and a green "Run" button with a play icon.
- Code Area:** The query code is displayed in a syntax-highlighted text area:

```
1 v SELECT C.First_Name, C.Last_Name
2   FROM Customer_142 C
3 WHERE NOT EXISTS (
4     SELECT 1
5       FROM Contract_142 CO
6      WHERE CO.Driver_License_Number = C.Driver_License_Number
7 );
8
```
- Results Area:** A table showing the results of the query:

FIRST_NAME	LAST_NAME
Roshan	Shrestha
Albert	Adison

#### 48. Query to Show Contracts with Insurance Information:

```
SELECT CO.Contract_ID, CO.Start_Date, I.Coverage_Type FROM Contract_142 CO  
JOIN Car_142 Car ON CO.Chassis_Number = Car.Chassis_Number JOIN  
Insurance_142 I ON Car.Insurance_Code = I.Insurance_Code WHERE  
CO.Chassis_Number IS NOT NULL;
```

The screenshot shows a SQL worksheet interface with the following elements:

- Header:** Includes "Live SQL" (with a red circle icon), "Feedback", "Help", user info "shrestharoshan776@gmail.com", and a dark mode switch.
- Toolbar:** Includes "Clear", "Find", "Actions", "Save", and a "Run" button.
- Code Area:** Displays the SQL query:

```
1 v SELECT CO.Contract_ID, CO.Start_Date, I.Coverage_Type  
2 FROM Contract_142 CO  
3 JOIN Car_142 Car ON CO.Chassis_Number = Car.Chassis_Number  
4 JOIN Insurance_142 I ON Car.Insurance_Code = I.Insurance_Code  
5 WHERE CO.Chassis_Number IS NOT NULL;  
6
```

- Result Area:** A table showing the results of the query:

CONTRACT_ID	START_DATE	COVERAGE_TYPE
1	01-AUG-23	Full

**49. Query to get employee who manage the greatest number of offices based on the number of locations associated with each office:**

```
SELECT E.First_Name FROM Employee_142 E WHERE E.Employee_ID IN (
    SELECT Office_ID FROM ( SELECT Office_ID, COUNT(*) AS Office_Count FROM Location_142 GROUP BY Office_ID ORDER BY Office_Count DESC ) WHERE ROWNUM <= 1 );
```

Live SQL

Feedback Help shrestharoshan776@gmail.com

SQL Worksheet

Clear Find Actions Save Run

```
1 v SELECT E.First_Name
2   FROM Employee_142 E
3 WHERE E.Employee_ID IN (
4     SELECT Office_ID
5       FROM (
6         SELECT Office_ID, COUNT(*) AS Office_Count
7           FROM Location_142
8             GROUP BY Office_ID
9               ORDER BY Office_Count DESC
10      )
11 WHERE ROWNUM <= 1
12 );
13
```

FIRST_NAME
Jane
Michael
David

**50. Query to find the customers who have rented cars with the highest mileage among all the cars available for rent:**

```
SELECT First_Name, Last_Name FROM Customer_142 WHERE
Driver_License_Number IN ( SELECT Driver_License_Number FROM Contract_142
WHERE Chassis_Number IN ( SELECT Chassis_Number FROM Car_142 WHERE
Mileage = ( SELECT MAX(Mileage) FROM Car_142 ) ) );
```

The screenshot shows a SQL worksheet interface with the following elements:

- Header:** Includes "Live SQL" (with a red circle icon), "Feedback", "Help", and a user account section.
- Toolbar:** Buttons for "Clear", "Find", "Actions", "Save", and a green "Run" button with a play icon.
- SQL Editor:** A code editor containing the provided SQL query, numbered from 1 to 15.
- Results:** A table showing the output of the query, with columns "FIRST\_NAME" and "LAST\_NAME". The result is "John Doe".

```
1 v SELECT First_Name, Last_Name
2 FROM Customer_142
3 WHERE Driver_License_Number IN (
4     SELECT Driver_License_Number
5     FROM Contract_142
6     WHERE Chassis_Number IN (
7         SELECT Chassis_Number
8         FROM Car_142
9         WHERE Mileage = (
10             SELECT MAX(Mileage)
11             FROM Car_142
12         )
13     )
14 );
15
```

FIRST_NAME	LAST_NAME
John	Doe