

Answer Sheet - SQL

Name : Vishal Shastri

Phone No. : 7030153436

Email : vishalshastri1999@gmail.com

Major Question 1

A) List the salary of all the employees.

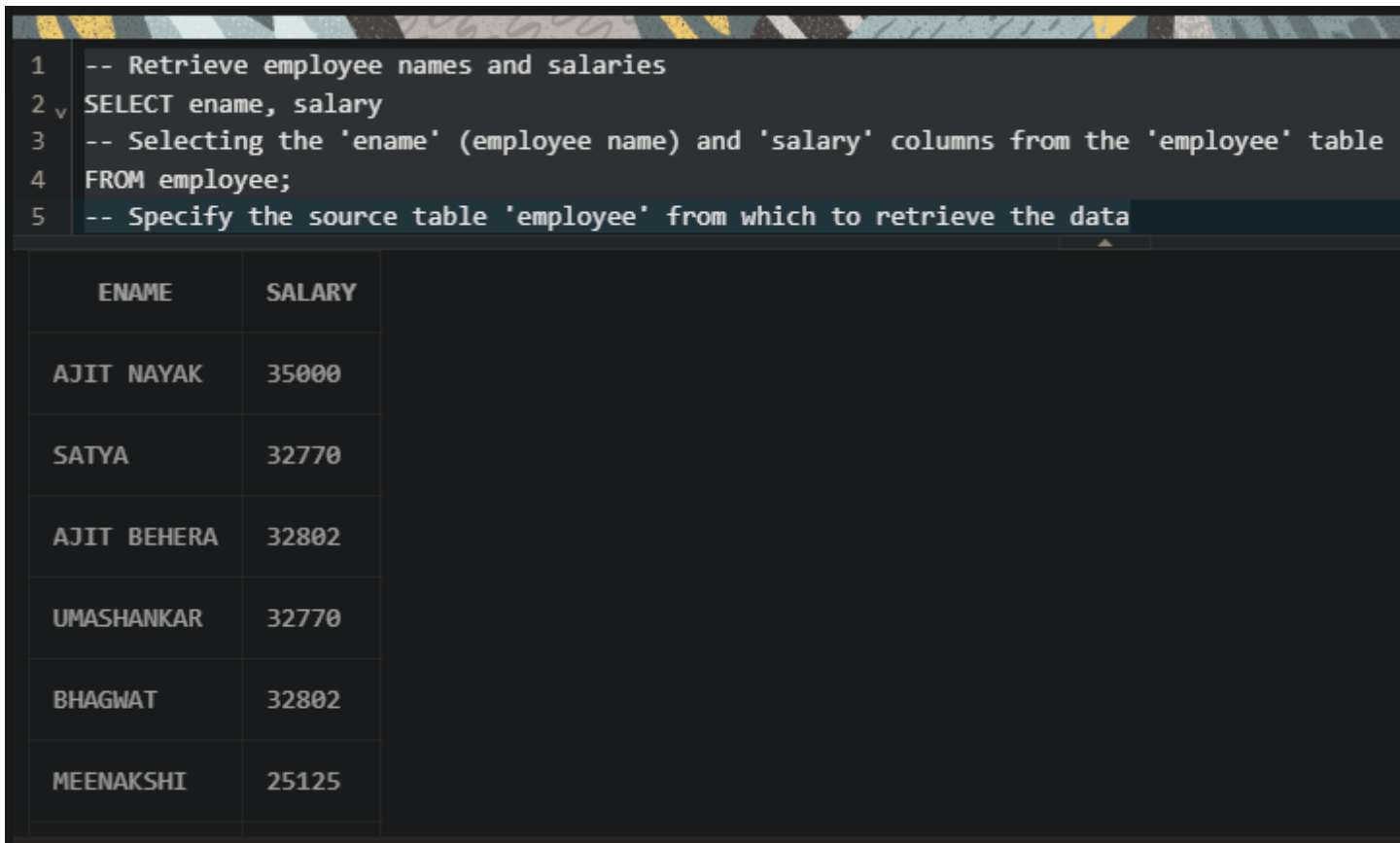
```
-- Retrieve employee names and salaries
SELECT ename, salary

-- Selecting the 'ename' (employee name) and 'salary'
columns from the 'employee' table

FROM employee;

-- Specify the source table 'employee' from which to
retrieve the data
```

Insert Image-



```
1  -- Retrieve employee names and salaries
2  v SELECT ename, salary
3  -- Selecting the 'ename' (employee name) and 'salary' columns from the 'employee' table
4  FROM employee;
5  -- Specify the source table 'employee' from which to retrieve the data
```

ENAME	SALARY
AJIT NAYAK	35000
SATYA	32770
AJIT BEHERA	32802
UMASHANKAR	32770
BHAGWAT	32802
MEENAKSHI	25125

B) Display the names of all employees with any “A” at any place of the name.

```
-- Selects the "ENAME" column, which likely represents employee names.
```

```
SELECT ENAME
```

```
-- Specifies the source table for data retrieval, which is the "EMPLOYEE" table.
```

```
FROM EMPLOYEE
```

```
-- It only includes rows where the "ENAME" column contains the letter 'A' at any position.
```

```
WHERE ENAME LIKE '%A%';
```

Insert Image –

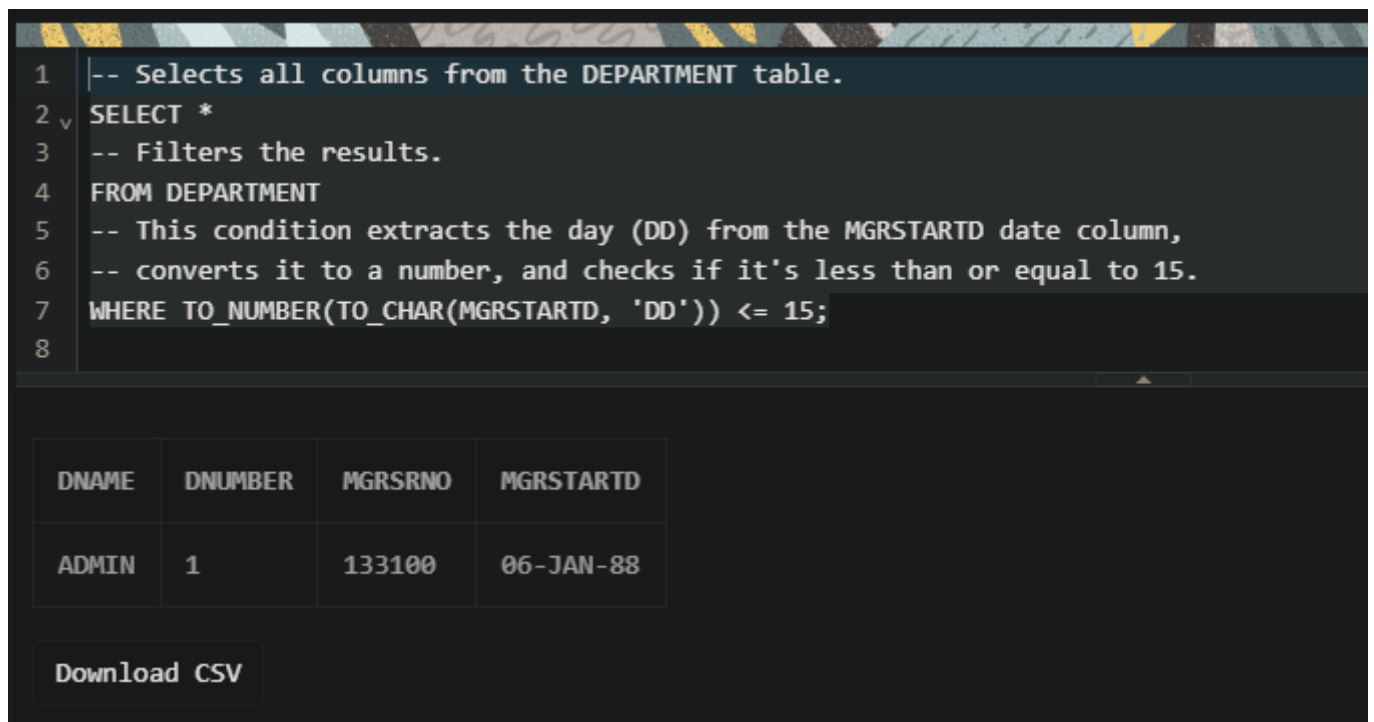
```
1 -- Selects the "ENAME" column, which likely represents employee names.
2 v SELECT ENAME
3 -- Specifies the source table for data retrieval, which is the "EMPLOYEE" table.
4 FROM EMPLOYEE
5 -- It only includes rows where the "ENAME" column contains the letter 'A' at any position.
6 WHERE ENAME LIKE '%A%';
```

ENAME
AJIT NAYAK
SATYA
AJIT BEHERA
UMASHANKAR
BHAGWAT

C) Show all employees who were hired in the first half of the month (Before the 16th of the month).

```
-- Selects all columns from the DEPARTMENT table.  
SELECT *  
  
-- Filters the results.  
  
FROM DEPARTMENT  
  
-- This condition extracts the day (DD) from the  
MGRSTARTD date column,  
  
-- converts it to a number, and checks if it's less than  
or equal to 15.  
  
WHERE TO_NUMBER(TO_CHAR(MGRSTARTD, 'DD')) <= 15;
```

Insert Image –



The screenshot shows a SQL query editor with a dark background. The query is as follows:

```
1 | -- Selects all columns from the DEPARTMENT table.  
2 | SELECT *  
3 | -- Filters the results.  
4 | FROM DEPARTMENT  
5 | -- This condition extracts the day (DD) from the MGRSTARTD date column,  
6 | -- converts it to a number, and checks if it's less than or equal to 15.  
7 | WHERE TO_NUMBER(TO_CHAR(MGRSTARTD, 'DD')) <= 15;  
8 |
```

Below the query editor, the results are displayed in a table with the following columns: DNAME, DNUMBER, MGRSRNO, and MGRSTARTD. The table contains one row of data:

DNAME	DNUMBER	MGRSRNO	MGRSTARTD
ADMIN	1	133100	06-JAN-88

At the bottom of the results section, there is a button labeled "Download CSV".

D) Display the name of all female employees.

```
SELECT ENAME, SEX
```

```
-- Select the columns for employee name and gender.
```

```
FROM EMPLOYEE
```

```
-- The data is being retrieved from the EMPLOYEE table.
```

```
WHERE SEX='F';
```

```
-- Filters the results to include only records where the  
gender is 'F' (Female).
```

Insert Image –



The screenshot shows a SQL IDE with a dark theme. The top part displays a SQL query with line numbers 1 through 8. The query is: `SELECT ENAME, SEX` (line 1), `-- Select the columns for employee name and gender.` (line 2), `FROM EMPLOYEE` (line 3), `-- The data is being retrieved from the EMPLOYEE table.` (line 4), `WHERE SEX='F';` (line 5), and `-- Filters the results to include only records where the gender is 'F' (Female).` (line 6). The bottom part shows the results of the query in a table with two columns: ENAME and SEX. The results are: MEENAKSHI (F) and DEBASMITA (F).

```
1 SELECT ENAME, SEX
2 -- Select the columns for employee name and gender.
3 FROM EMPLOYEE
4 -- The data is being retrieved from the EMPLOYEE table.
5 WHERE SEX='F';
6 -- Filters the results to include only records where the gender is 'F' (Female).
7
8
```

ENAME	SEX
MEENAKSHI	F
DEBASMITA	F

E) Display the employee who is paid most in the company.

```
-- Select all columns from the "EMPLOYEE" table
SELECT * FROM EMPLOYEE

-- Filter the results using a subquery
WHERE SALARY IN

(

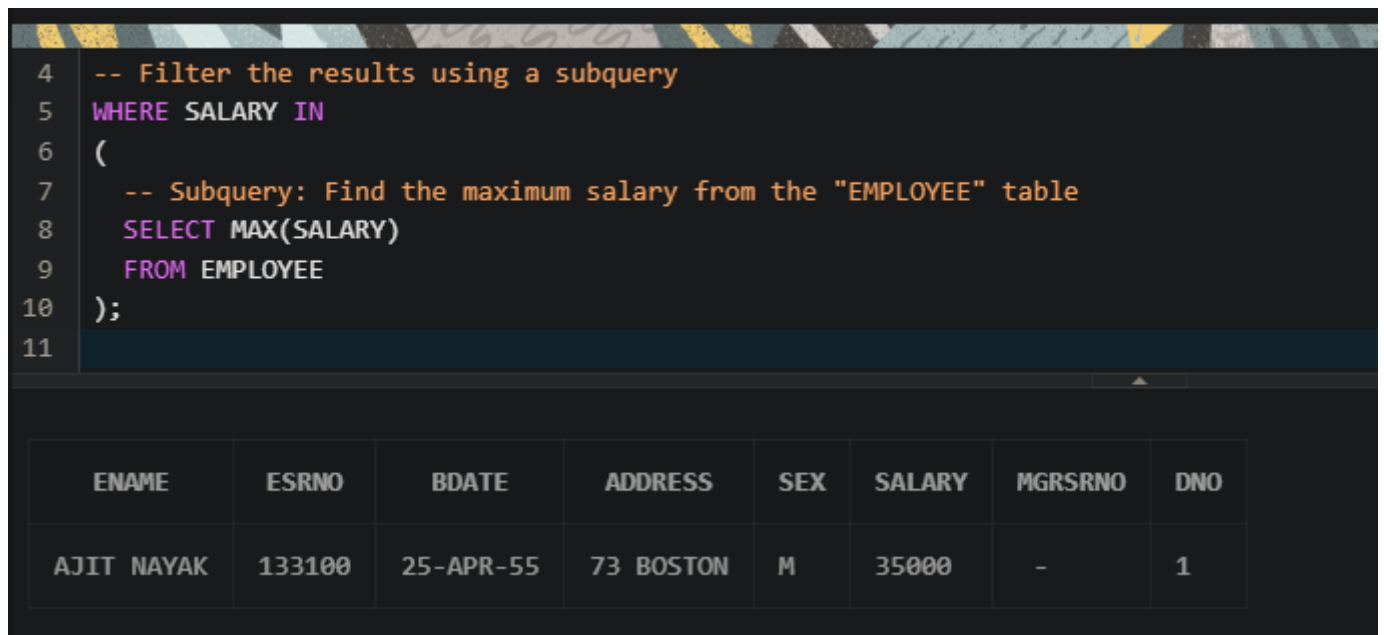
-- Subquery: Find the maximum salary from the "EMPLOYEE"
table

SELECT MAX(SALARY)

FROM EMPLOYEE

);
```

Insert Image –



The screenshot shows a terminal window with a dark background. The top part displays a SQL query with line numbers 4 through 11. The query is designed to select all columns from the EMPLOYEE table where the salary is equal to the maximum salary in the table. Below the query, a table of results is displayed with 8 columns: ENAME, ESRNO, BDATE, ADDRESS, SEX, SALARY, MGRSRNO, and DNO. The result shows a single row for the employee AJIT NAYAK.

```
4  -- Filter the results using a subquery
5  WHERE SALARY IN
6  (
7      -- Subquery: Find the maximum salary from the "EMPLOYEE" table
8      SELECT MAX(SALARY)
9      FROM EMPLOYEE
10 );
11
```

ENAME	ESRNO	BDATE	ADDRESS	SEX	SALARY	MGRSRNO	DNO
AJIT NAYAK	133100	25-APR-55	73 BOSTON	M	35000	-	1

Major Question 2

A) Display employee name, address, department no and department name.

```
-- Selects employee name, address, department number, and
department name

SELECT ENAME, ADDRESS, DNO, DNAME

-- Specifies the source tables to retrieve data from

FROM EMPLOYEE

-- Performs an INNER JOIN with the DEPARTMENT table

INNER JOIN DEPARTMENT

-- Specifies the join condition based on employee's serial
number and manager's serial number

ON EMPLOYEE.ESRNO = DEPARTMENT.MGRSRNO;
```

Insert Image –

```
1  |-- Selects employee name, address, department number, and department name
2  v SELECT ENAME, ADDRESS, DNO, DNAME
3  |-- Specifies the source tables to retrieve data from
4  |FROM EMPLOYEE
5  |-- Performs an INNER JOIN with the DEPARTMENT table
6  |INNER JOIN DEPARTMENT
7  |-- Specifies the join condition based on employee's serial number and manager's serial number
8  |ON EMPLOYEE.ESRNO = DEPARTMENT.MGRSRNO;
```

ENAME	ADDRESS	DNO	DNAME
AJIT NAYAK	73 BOSTON	1	ADMIN
SATYA	26 FINE OAK	4	RESEARCH
AJIT BEHERA	10 KALINGA	3	ACADEMIC
UMASHANKAR	26 FINE OAK	2	PROJECT

B) Display all the employees who are *not in* ACADEMIC department

-- Selects employee names and department names

SELECT EMPLOYEE.ENAME, DEPARTMENT.DNAME

-- Retrieves data from the EMPLOYEE and DEPARTMENT tables

FROM EMPLOYEE

-- Performs an inner join between EMPLOYEE and DEPARTMENT on the manager's serial number

INNER JOIN DEPARTMENT ON EMPLOYEE.ESRNO = DEPARTMENT.MGRSRNO

-- Filters the results to exclude departments with the name 'ACADEMIC'

WHERE DNAME != 'ACADEMIC';

Insert Image –

```
1  -- Selects employee name, address, department number, and department name
2  SELECT ENAME, ADDRESS, DNO, DNAME
3  -- Specifies the source tables to retrieve data from
4  FROM EMPLOYEE
5  -- Performs an INNER JOIN with the DEPARTMENT table
6  INNER JOIN DEPARTMENT
7  -- Specifies the join condition based on employee's serial number and manager's serial number
8  ON EMPLOYEE.ESRNO = DEPARTMENT.MGRSRNO;
```

ENAME	ADDRESS	DNO	DNAME
AJIT NAYAK	73 BOSTON	1	ADMIN
SATYA	26 FINE OAK	4	RESEARCH
AJIT BEHERA	10 KALINGA	3	ACADEMIC
UMASHANKAR	26 FINE OAK	2	PROJECT

C) Display SATYAS' project location.

```
-- Selects distinct employee names (ENAME) and project
locations (PLOCATION)

SELECT DISTINCT EMPLOYEE.ENAME, PROJECT.PLOCATION

-- Specifies the source tables for the query

FROM EMPLOYEE

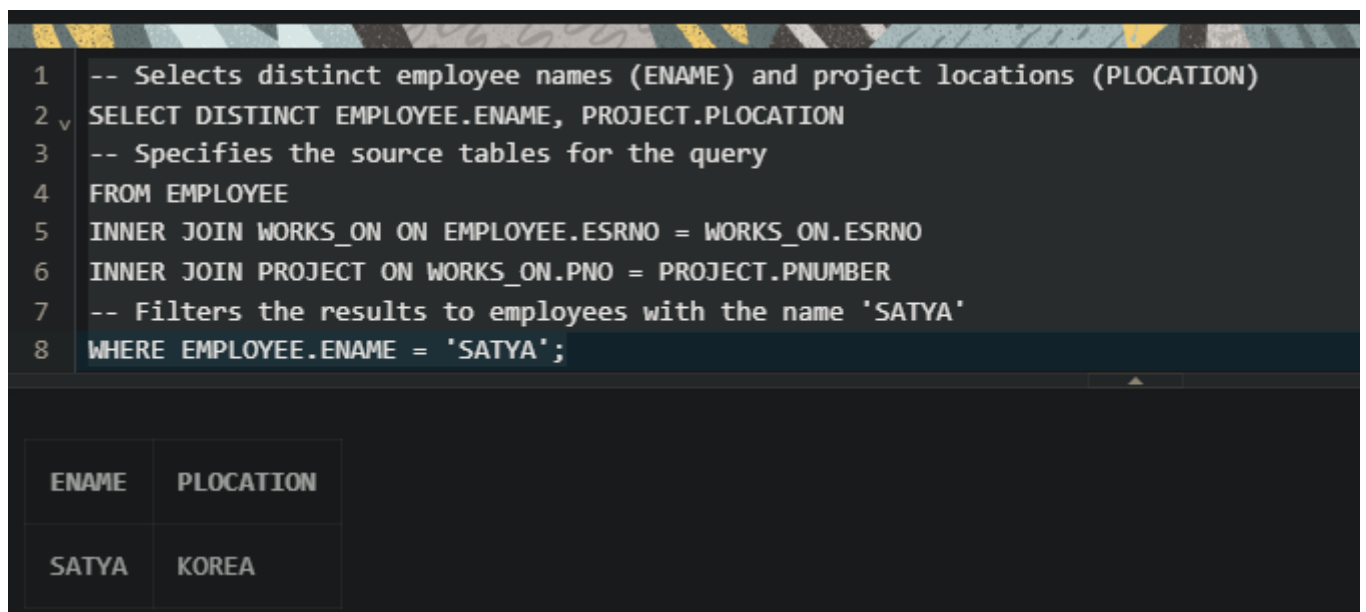
INNER JOIN WORKS_ON ON EMPLOYEE.ESRNO = WORKS_ON.ESRNO

INNER JOIN PROJECT ON WORKS_ON.PNO = PROJECT.PNUMBER

-- Filters the results to employees with the name 'SATYA'

WHERE EMPLOYEE.ENAME = 'SATYA';
```

Insert Image –



The image shows a terminal window with a SQL query and its result. The query is as follows:

```
1  -- Selects distinct employee names (ENAME) and project locations (PLOCATION)
2  v SELECT DISTINCT EMPLOYEE.ENAME, PROJECT.PLOCATION
3  -- Specifies the source tables for the query
4  FROM EMPLOYEE
5  INNER JOIN WORKS_ON ON EMPLOYEE.ESRNO = WORKS_ON.ESRNO
6  INNER JOIN PROJECT ON WORKS_ON.PNO = PROJECT.PNUMBER
7  -- Filters the results to employees with the name 'SATYA'
8  WHERE EMPLOYEE.ENAME = 'SATYA';
```

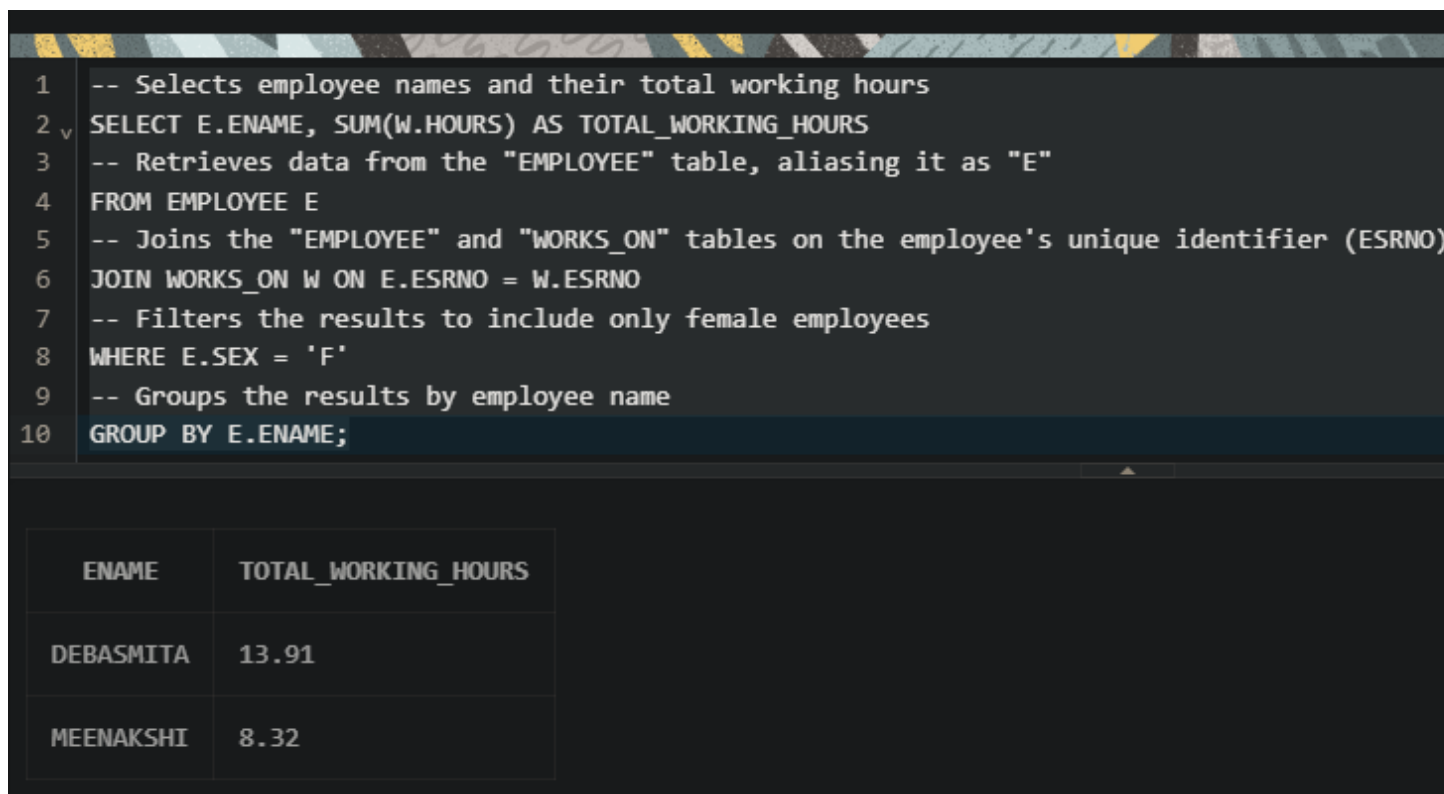
The result of the query is displayed in a table below the query:

ENAME	PLOCATION
SATYA	KOREA

D) Find the total working hours of *each* female employee.

```
-- Selects employee names and their total working hours
SELECT E.ENAME, SUM(W.HOURS) AS TOTAL_WORKING_HOURS
-- Retrieves data from the "EMPLOYEE" table, aliasing it as
"E"
FROM EMPLOYEE E
-- Joins the "EMPLOYEE" and "WORKS_ON" tables on the
employee's unique identifier (ESRNO)
JOIN WORKS_ON W ON E.ESRNO = W.ESRNO
-- Filters the results to include only female employees
WHERE E.SEX = 'F'
-- Groups the results by employee name
GROUP BY E.ENAME;
```

Insert Image –



```
1  -- Selects employee names and their total working hours
2  v SELECT E.ENAME, SUM(W.HOURS) AS TOTAL_WORKING_HOURS
3  -- Retrieves data from the "EMPLOYEE" table, aliasing it as "E"
4  FROM EMPLOYEE E
5  -- Joins the "EMPLOYEE" and "WORKS_ON" tables on the employee's unique identifier (ESRNO)
6  JOIN WORKS_ON W ON E.ESRNO = W.ESRNO
7  -- Filters the results to include only female employees
8  WHERE E.SEX = 'F'
9  -- Groups the results by employee name
10 GROUP BY E.ENAME;
```

ENAME	TOTAL_WORKING_HOURS
DEBASMITA	13.91
MEENAKSHI	8.32

E) Display the details of the people whose projects are located at SOUTH AFRICA.

```
-- Selects distinct employee names, project names, project
locations, and project numbers

SELECT DISTINCT EMPLOYEE.ENAME, PROJECT.PNAME,
PROJECT.PLOCATION, PROJECT.PNUMBER

-- Retrieves data from the "PROJECT" table

FROM PROJECT

-- Joins the "PROJECT" and "WORKS_ON" tables on the project
number (PNUMBER)

INNER JOIN WORKS_ON ON PROJECT.PNUMBER = WORKS_ON.PNO

-- Further joins the "WORKS_ON" and "EMPLOYEE" tables on
the employee's unique identifier (ESRNO)

INNER JOIN EMPLOYEE ON WORKS_ON.ESRNO = EMPLOYEE.ESRNO

-- Filters the results to include only projects located in
'SOUTH AFRICA'

WHERE PROJECT.PLOCATION = 'SOUTH AFRICA'

-- Orders the results by employee name

ORDER BY ENAME;
```

Insert Image –

```
1 -- Selects distinct employee names, project names, project locations, and project numbers
2 v SELECT DISTINCT EMPLOYEE.ENAME, PROJECT.PNAME, PROJECT.PLOCATION, PROJECT.PNUMBER
3 -- Retrieves data from the "PROJECT" table
4 FROM PROJECT
5 -- Joins the "PROJECT" and "WORKS_ON" tables on the project number (PNUMBER)
6 INNER JOIN WORKS_ON ON PROJECT.PNUMBER = WORKS_ON.PNO
7 -- Further joins the "WORKS_ON" and "EMPLOYEE" tables on the employee's unique identifier (ESRNO)
8 INNER JOIN EMPLOYEE ON WORKS_ON.ESRNO = EMPLOYEE.ESRNO
9 -- Filters the results to include only projects located in 'SOUTH AFRICA'
10 WHERE PROJECT.PLOCATION = 'SOUTH AFRICA'
11 -- Orders the results by employee name
```

ENAME	PNAME	PLOCATION	PNUMBER
AJIT BEHERA	BIO INFORMATICS	SOUTH AFRICA	19
MEENAKSHI	BIO INFORMATICS	SOUTH AFRICA	19
NIHAR NAYAK	BIO INFORMATICS	SOUTH AFRICA	19