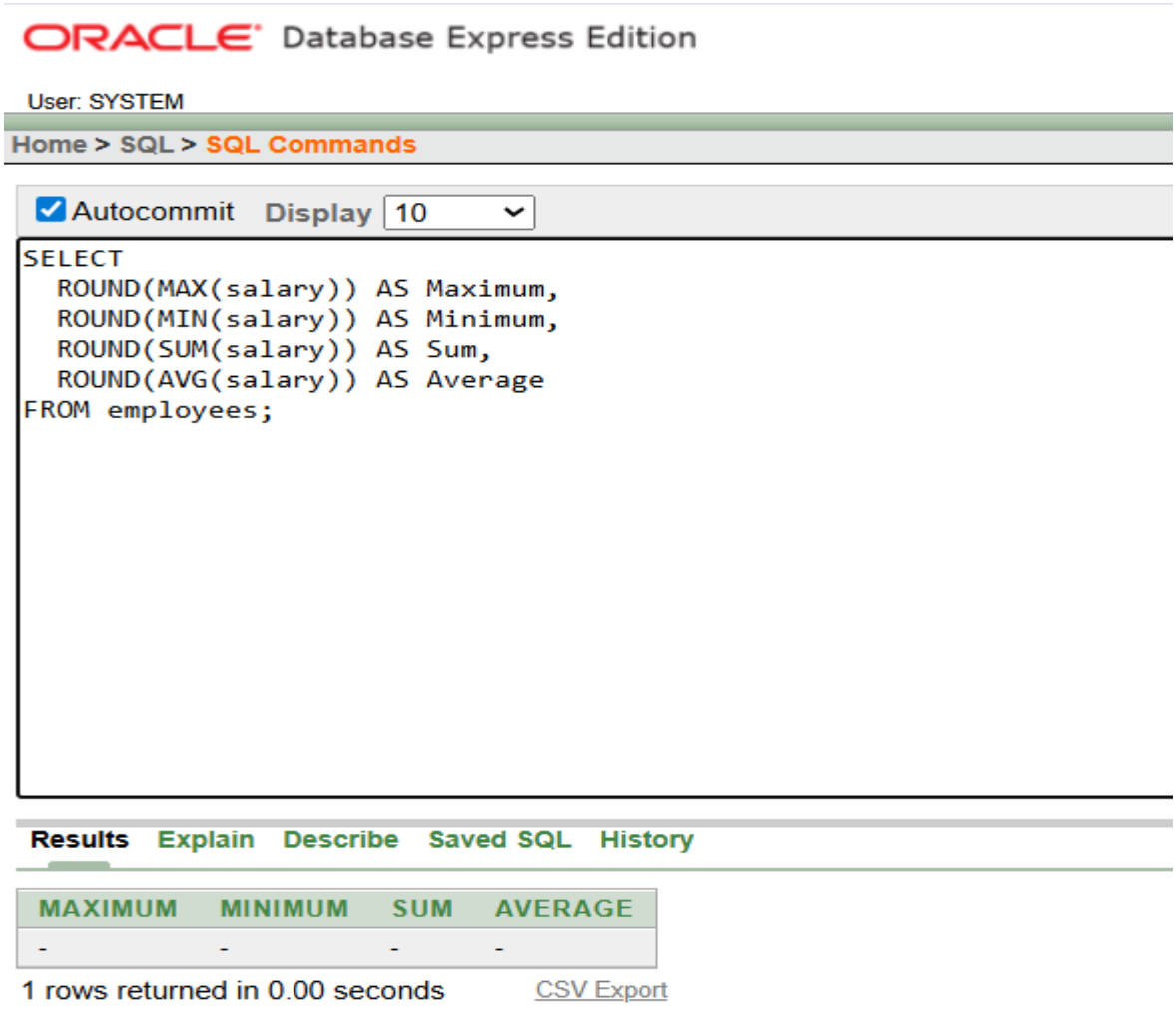
EXERCISE 8

NAME: VIDHYASHREE.J

REG NO: 241801310

1. **Group functions work across many rows to produce one result per group.**
   True
2. **Group functions include nulls in calculations.**
   False
3. **The WHERE clause restricts rows prior to inclusion in a group calculation.**
   True
4. Find the highest, lowest, sum, and average salary of all employees. Label the columns Maximum, Minimum, Sum, and Average, respectively. Round your results to the nearest whole number



5.Modify the above query to display the minimum, maximum, sum, and average salary for each job type.

ORACLE Database Express Edition

User: SYSTEM

Home > SQL > SQL Commands

Autocommit Display 10

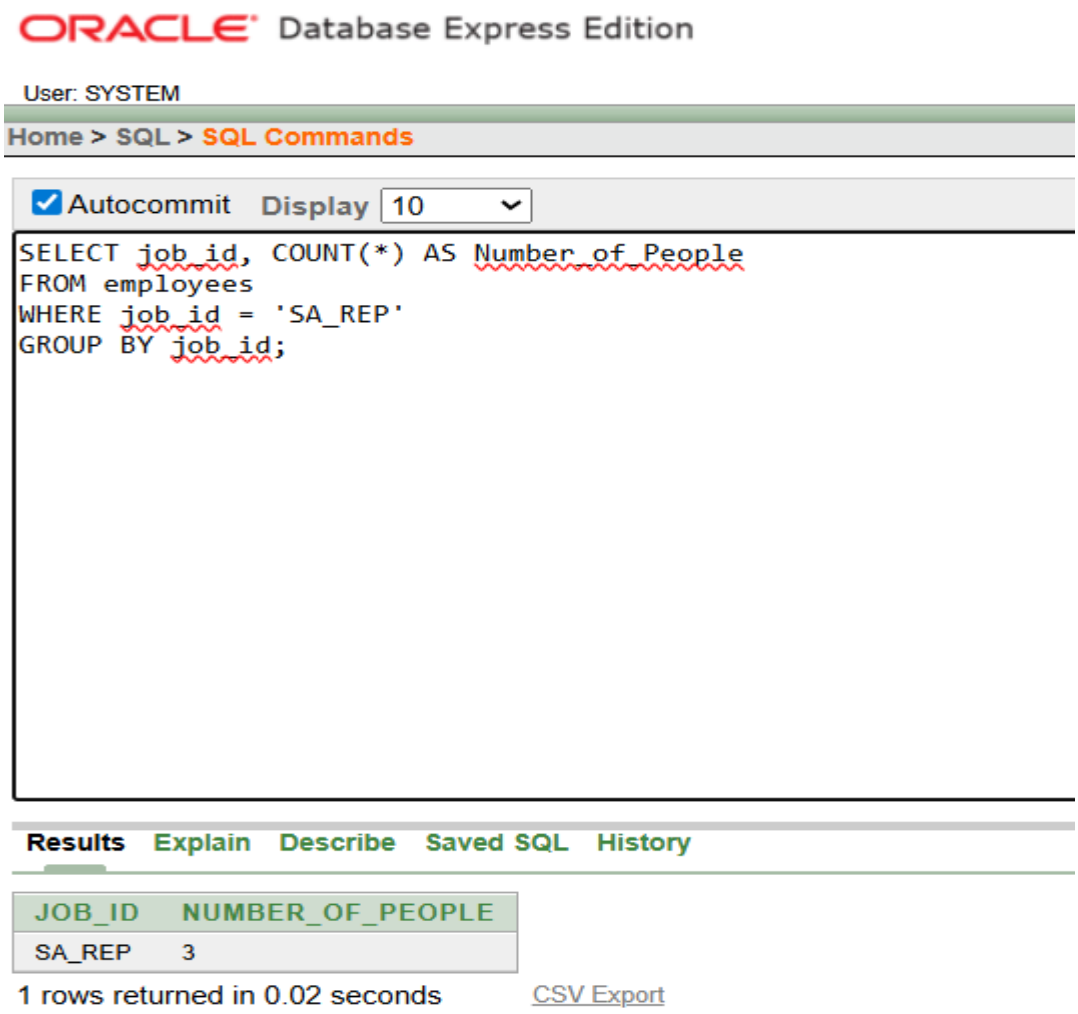
```
SELECT
  job_id,
  ROUND(MIN(salary)) AS Minimum,
  ROUND(MAX(salary)) AS Maximum,
  ROUND(SUM(salary)) AS Sum,
  ROUND(AVG(salary), 2) AS Average
FROM employees
GROUP BY job_id;
```

Results Explain Describe Saved SQL History

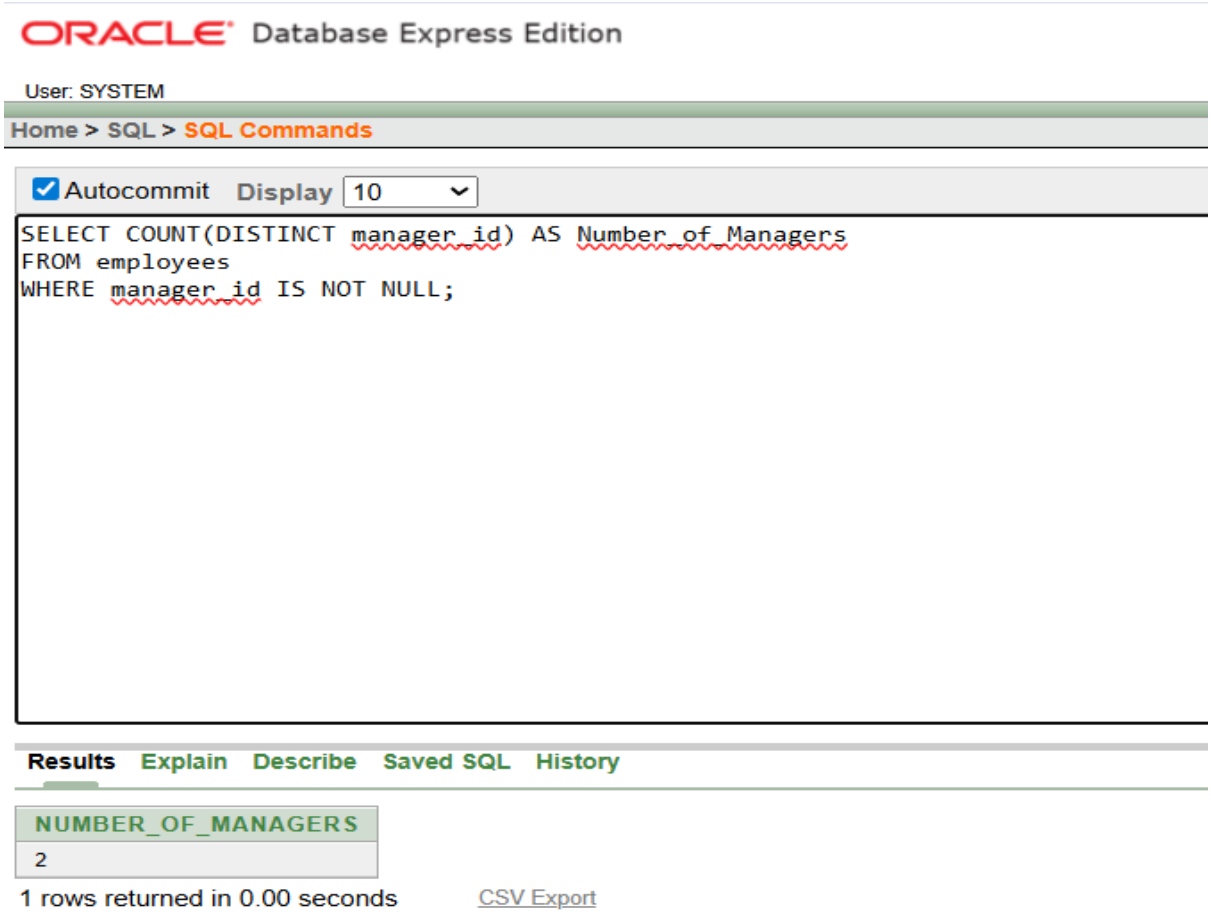
JOB_ID	MINIMUM	MAXIMUM	SUM	AVERAGE
AC_MGR	9000	9000	9000	9000
IT_PROG	6000	6200	12200	6100
AC_ACCOUNT	4000	4000	4000	4000
SA_REP	6800	7500	21300	7100
HR_REP	4000	4000	4000	4000

5 rows returned in 0.00 seconds CSV Export

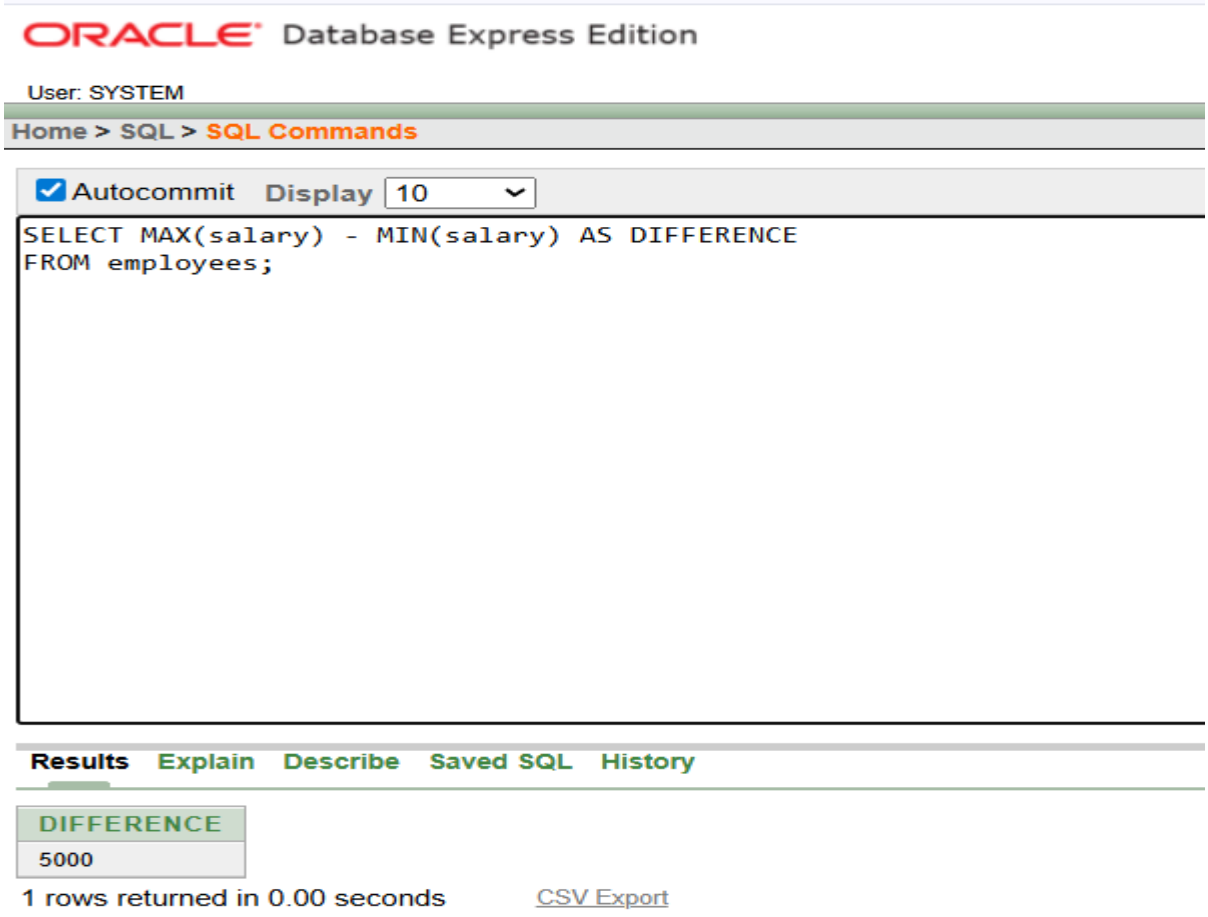
6.Write a query to display the number of people with the same job. Generalize the query so that the user in the HR department is prompted for a job title.



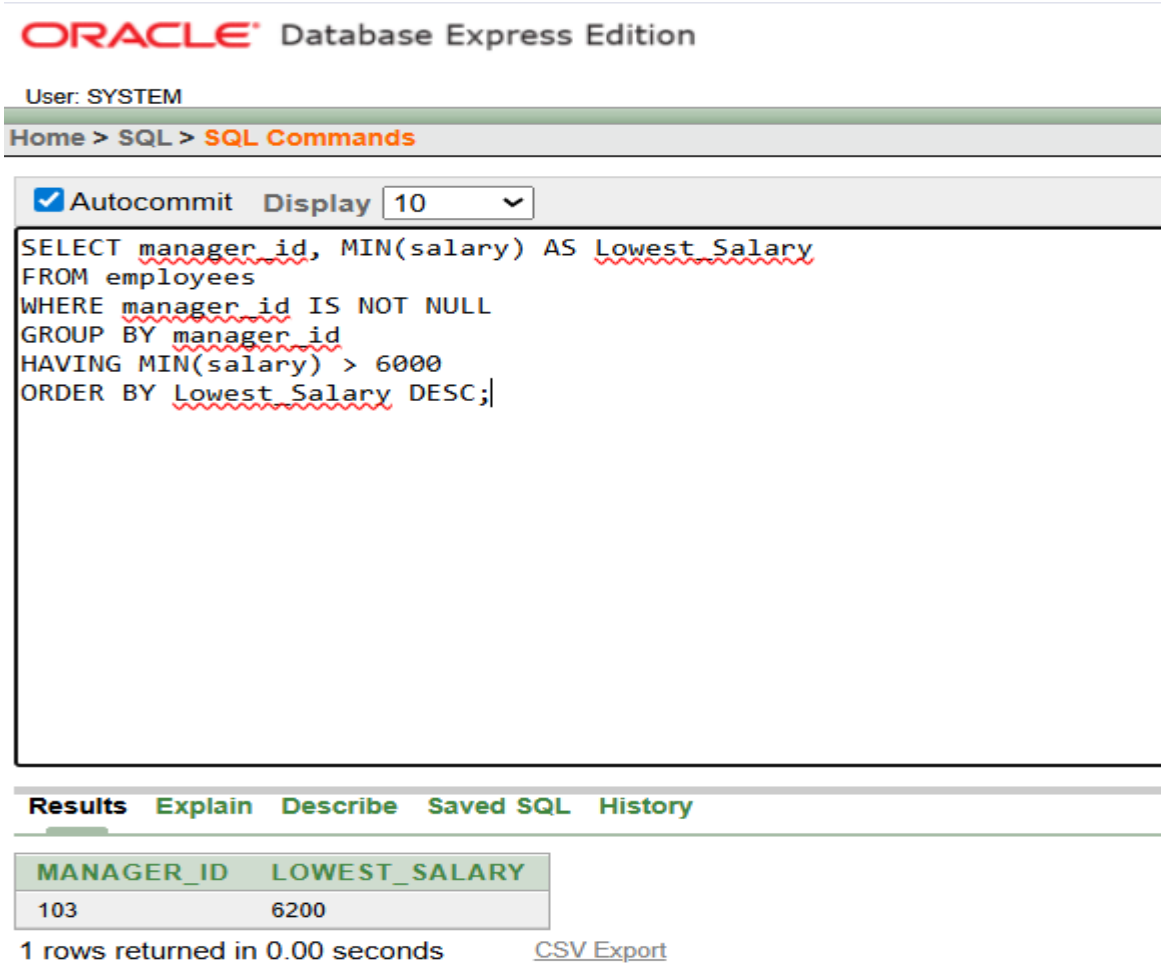
7. Determine the number of managers without listing them. Label the column Number of Managers. Hint: Use the MANAGER_ID column to determine the number of managers



8. Find the difference between the highest and lowest salaries. Label the column DIFFERENCE.



9. Create a report to display the manager number and the salary of the lowest-paid employee for that manager. Exclude anyone whose manager is not known. Exclude any groups where the minimum salary is $6,000 or less. Sort the output in descending order of salary.



10. Create a query to display the total number of employees and, of that total, the number of employees hired in 1995, 1996, 1997, and 1998. Create appropriate column headings.

ORACLE Database Express Edition

User: SYSTEM

Home > SQL > SQL Commands

Autocommit Display 10

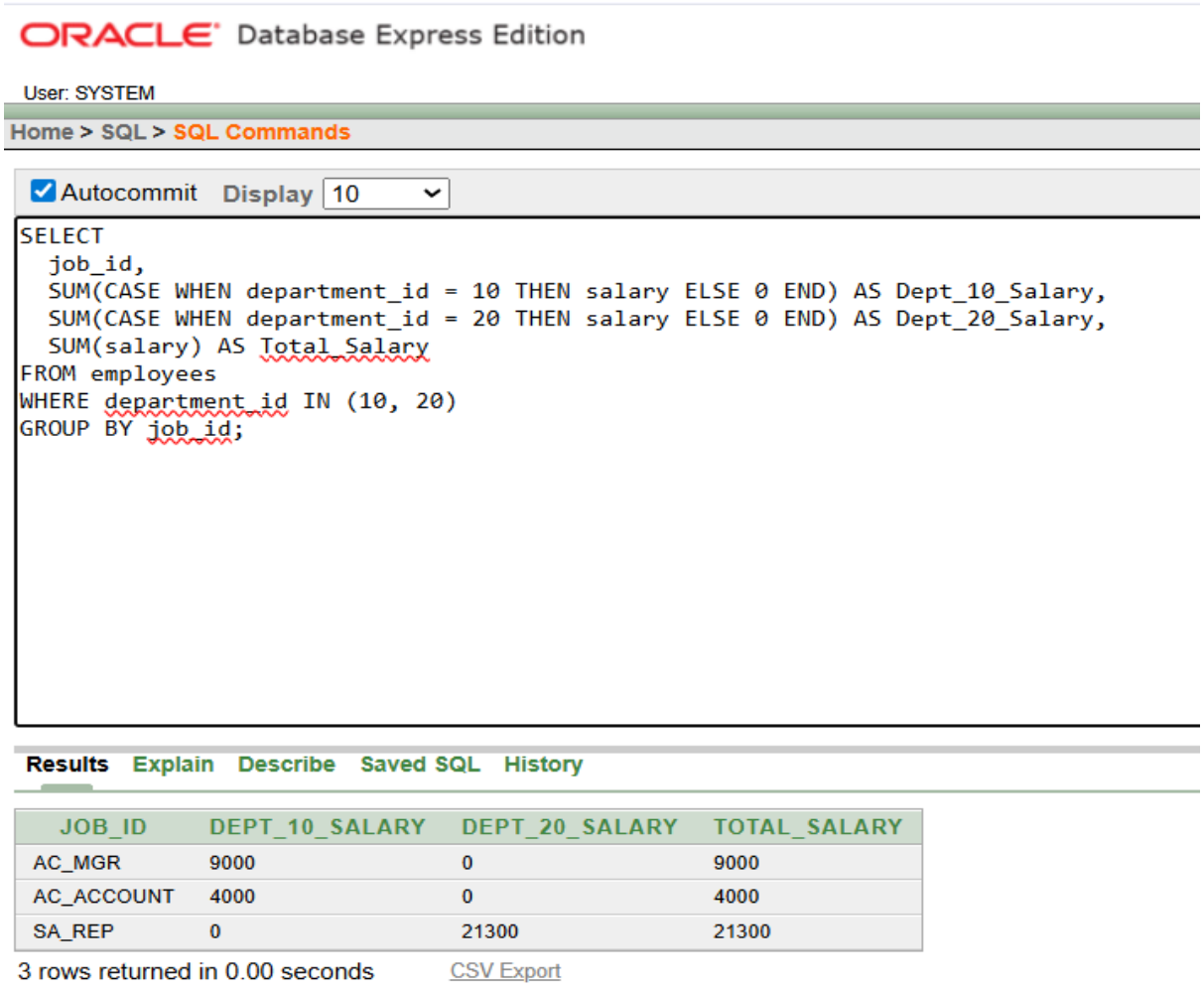
```
SELECT
  COUNT(*) AS Total_Employees,
  SUM(CASE WHEN TO_CHAR(hire_date, 'YYYY') = '1995' THEN 1 ELSE 0 END) AS Hired_1995,
  SUM(CASE WHEN TO_CHAR(hire_date, 'YYYY') = '1996' THEN 1 ELSE 0 END) AS Hired_1996,
  SUM(CASE WHEN TO_CHAR(hire_date, 'YYYY') = '1997' THEN 1 ELSE 0 END) AS Hired_1997,
  SUM(CASE WHEN TO_CHAR(hire_date, 'YYYY') = '1998' THEN 1 ELSE 0 END) AS Hired_1998
FROM employees;
```

Results Explain Describe Saved SQL History

TOTAL_EMPLOYEES	HIRED_1995	HIRED_1996	HIRED_1997	HIRED_1998
8	2	2	2	2

1 rows returned in 0.00 seconds CSV Export

11. Create a matrix query to display the job, the salary for that job based on department number, and the total salary for that job, for departments 20, 50, 80, and 90, giving each column an appropriate heading.



12.Write a query to display each department's name, location, number of employees, and the average salary for all the employees in that department. Label the column name-Location, Number of people, and salary respectively. Round the average salary to two decimal places.

ORACLE Database Express Edition

User: SYSTEM

Home > SQL > SQL Commands

Autocommit Display 10

```
SELECT
  d.department_name AS name_Location,
  l.location AS Location,
  COUNT(e.employee_id) AS Number_of_people,
  ROUND(AVG(e.salary), 2) AS salary
FROM departments d
JOIN employees e ON d.department_id = e.department_id
JOIN locations l ON d.location_id = l.location_id
GROUP BY d.department_name, l.location;
```

Results Explain Describe Saved SQL History

NAME_LOCATION	LOCATION	NUMBER_OF_PEOPLE	SALARY
HR	New York	1	4000
Accounting	New York	2	6500
Sales	San Francisco	3	7100
IT	Chicago	2	6100

4 rows returned in 0.00 seconds CSV Export