4. Grouping and Sorting
Ex. 4.

i) Group By and having ii) Order By

```
--First Create required table.

CREATE TABLE employees

( emp_no NUMBER(5), name VARCHAR2(50), age NUMBER(3), department
VARCHAR2(50), salary NUMBER(10,2) );


  -- Insert sample records

INSERT INTO employees VALUES (115,'Rajani',27,'Accounts',46000.00);

INSERT INTO employees VALUES (115,'Rajani',29,'Accounts',55000.00);

INSERT INTO employees VALUES (115,'Rajani',32,'Sales',43000.00) ;

INSERT INTO employees VALUES (115,'Rajani',30,'Computer',65000.00);

INSERT INTO employees VALUES (115,'Rajani',31,'Sales',58000.00);
```

i)

The GROUP BY clause is used in SQL to group the results of a query by one or more columns. It is typically used in conjunction with aggregate functions like COUNT, SUM, AVG, etc. to group the data based on specific criteria.

```
SELECT department, COUNT(*) as num_employees

FROM employees

GROUP BY department;
```

In this example, we are selecting the department column and counting the number of employees in each department using the COUNT function. We are then grouping the results by the department column using the GROUP BY clause.

The HAVING clause is used in SQL to filter the results of a GROUP BY query based on a condition. It is similar to the WHERE clause, but is

used to filter groups of rows rather than individual rows. Here is an example of how the HAVING clause can be used in SQL:

```
SELECT department, AVG(salary) as avg_salary

FROM employees

GROUP BY department

HAVING AVG(salary) > 50000;
```

In this example, we are selecting the department column and calculating the average salary of employees in each department with average salary more that 50000 (using the AVG function).

ii) Using ORDER BY

The ORDER BY clause in SQL is used to sort the results of a query in ascending or descending order based on one or more columns.

```
SELECT name, age, salary

FROM employees

ORDER BY salary DESC;
```

In this example, we are selecting the name, age, and salary columns from the employees table. We are then using the ORDER BY clause to sort the results in descending order based on the salary column.

It is also possible to sort the results based on multiple columns using the ORDER BY clause. example:

```
SELECT name, age, salary

FROM employees

ORDER BY age DESC, salary ASC;
```

In this example, we are selecting the name, age, and salary columns from the employees table. We are then using the ORDER BY clause to sort the results in descending order based on the age column and in ascending order based on the salary column. The output will be a table with three columns, where the rows are displayed in descending order based on the age column and in ascending order based on the salary column.