Group By Clause

The GROUP BY statement groups rows that have the same values into summary rows, like "find the number of customers in each country".

The GROUP BY statement is often used with aggregate functions (COUNT(), MAX(), MIN(), SUM(), AVG()) to group the result-set by one or more columns.

Syntax:

```
SELECT
  column_name(s)
FROM
  table_name
WHERE
  condition
GROUP BY column_name(s)
ORDER BY column_name(s);
```

1. Write a query to display the dept_id and average salary of all employees in each department

```
SELECT
  dept_id, AVG(salary)
FROM
  employee
GROUP BY dept_id;
```

Output:

dept_id	AVG(salary)
20	104200.0000
40	53200.0000
50	98200.0000

70	84200.0000
80	60100.0000

2. Query the different department in the employee table

```
SELECT
  dept_id
FROM
  employee
GROUP BY
  dept_id;
```

Output:

dept_id
20
40
50
70
80

3. Write a query to retrieve the dept_id, salary of employees having different salaries in each department

```
SELECT
  dept_id, salary
FROM
  employee
GROUP BY
  dept_id, salary;
```

Output:

dept_id	salary
80	78000
70	84200
50	98200
80	42200
40	42200
40	64200
20	84200
20	124200

Here unique value with the combination of dept_id and salary is retrieved.

4. Query the number of employees in each department

```
SELECT
   count(*)
FROM
   employee
GROUP BY
   dept_id;
```

Output:

count(*)
2
2

1
1
2

5. Write a query to display department id and least salary of all employees in each department

```
SELECT
  dept_id, min(salary)
FROM
  employee
GROUP BY
  dept_id;
```

Output:

dept_id	min(salary)
20	84200
40	42200
50	98200
70	84200
80	42200

6. Query the total salary paid to the employees of each department whose id is greater than 20

```
SELECT
  SUM(salary)
FROM
  employee
WHERE
  dept_id > 20
GROUP BY
  dept_id ;
```

Output:

SUM(salary)
106400
98200
84200
120200

7. Query the number of employees, min salary of all employees in each department except sales department
   Dept_id of sales is 20.

```
SELECT
  COUNT(*), MIN(salary)
FROM
  employee
WHERE
  dept_id != 20
GROUP BY
  dept_id ;
```

Output:

COUNT(*)	salary

2	42200
1	98200
1	84200
2	42200

8. Query the number of employees hired in each year

```
SELECT
  COUNT(*), YEAR(hire_date)
FROM
  employee
GROUP BY
  YEAR(hire_date);
```

Output:

COUNT(*)	YEAR(hire_date)
5	2021
1	2020
1	2019
1	2018

9. Display the dept_id, number of employees in each department in descending order

```
SELECT
  dept_id, COUNT(emp_id) as c
FROM
  employee
GROUP BY
```

```
  dept_id
ORDER by c DESC ;
```

Output:

dept_id	c
20	2
40	2
80	2
50	1
70	1

As you can see from the above output count of emp_id of each department is in decreased order