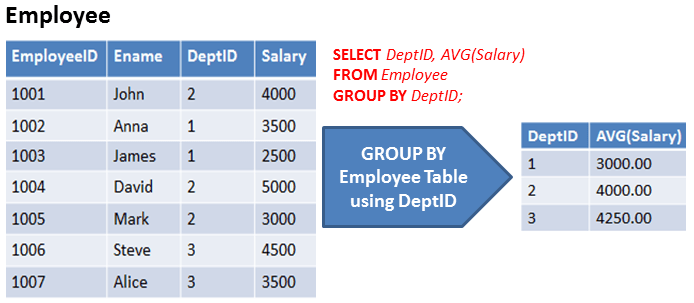
**SQL AGGREGATION KEYWORDS:**

**Group by:**

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.



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **empID** | **empName** | **depID** | **salary** | **Gender** | **Joinyear** |
| **1** | **Sana** | **2** | **3000** | **Female** | **2021** |
| **2** | **Renu** | **1** | **4000** | **Female** | **2020** |
| **3** | **Anuj** | **3** | **6000** | **Male** | **2019** |
| **4** | **Sujit** | **2** | **5500** | **Male** | **2021** |
| **5** | **Debashish** | **2** | **3200** | **Male** | **2022** |
| **6** | **Himanshu** | **3** | **6700** | **Male** | **1990** |
| **7** | **Deepali** | **1** | **7800** | **Female** | **2020** |
| **8** | **Sanjeev** | **3** | **2100** | **Male** | **2021** |
| **9** | **Kisan** | **2** | **2200** | **Male** | **2022** |
| **10** | **Kush** | **1** | **1100** | **Male** | **2022** |

use maulika;

select \* from emp\_details;

**SET GLOBAL sql\_mode=(SELECT REPLACE(@@sql\_mode,'ONLY\_FULL\_GROUP\_BY',''));set @@global.sql\_mode := replace(@@global.sql\_mode, 'ONLY\_FULL\_GROUP\_BY', '');**

**1.Department wise salary**

select sum(salary) from emp\_details group by depID;

select depID,sum(salary) from emp\_details group by depID;

**2.Department wise AVG salary**

select depID,AVG(salary) from emp\_details group by depID;

select depID,sum(salary),AVG(salary) from emp\_details group by depID;

**3.Department wise min salary**

select depID,min(salary) from emp\_details group by depID;

select depID,sum(salary),min(salary) from emp\_details group by depID;

**4.Department wise max salary**

select depID,max(salary) from emp\_details group by depID;

select depID,sum(salary),max(salary) from emp\_details group by depID;

**5.Department wise sum,avg,min,max salary**

select depID,sum(salary), avg(salary) ,max(salary) ,min(salary) from emp\_details group by depID;

**6. count:**

select depID,count(salary) from emp\_details group by depID;

select depID,count(depID) from emp\_details group by depID;

**7.salaries person count according to depid**

select depID,sum(salary), avg(salary) ,max(salary) ,min(salary) ,count(depID) from emp\_details group by depID;

**Multiple column group by**

**8. count the employees :**

select depId,count(depID) as totalemp from emp\_details group by depID;

**without give column name:**

select depId,count(depID) from emp\_details group by depID;

select depId,joinyear,count(depID) as totalemp from emp\_details group by depID,joinyear;

**9. find out total salary, avg salary in dep 1 and 3?**

select depId,avg(salary),sum(salary) from emp\_details

where depID in(1,3)

group by depID;

select depId,avg(salary),sum(salary) from emp\_details

where depID in(2,3)

group by depID;

**10. find out min salary, max salary in dep 1 and 3?**

select depId,min(salary),max(salary) from emp\_details

where depID in(1,3)

group by depID;

select depId,min(salary),max(salary) from emp\_details

where depID in(2,3)

group by depID;

**Having clause:**

The HAVING clause is **used to apply a filter on the result of GROUP BY based on the specified condition**. The conditions are Boolean type i.e. use of logical operators(AND, OR). This clause was included in SQL as the WHERE keyword failed when we use it with aggregate expressions.

Having clause filters the whole group.

Having clause can contain aggregate function.

Having clause selects the data after grouping.

Having clause can not be used without group by clause.

1. **Find total salary in department 1 and 3 and group total salary should be greater than 13000.**

select depID,sum(salary) as TotalSalary from emp\_details

where depId in(1,3)

group by depID

having sum(salary) >13000;

1. **Find total salary in department 2 and 3 and group total salary should be greater than 14000.**

select depID,sum(salary) as TotalSalary from emp\_details

where depId in(2,3)

group by depID

having sum(salary) >14000;

**3.Find avg salary in department 2 and 3 and group AvgSalary should be less than 3500.**

select depID,avg(salary) as AvgSalary from emp\_details

where depId in(2,3)

group by depID

having avg(salary) < 3500;

**4.Find avg salary in department 1 and 3 and group AvgSalary should be less than 4500.**

select depID,avg(salary) as AvgSalary from emp\_details

where depId in(1,3)

group by depID

having avg(salary) < 4500;

**5.Find total salary in department 2 and 3 and group totalSalary should be less than 14000.**

select depID,sum(salary) as totalsalary from emp\_details

where depId in(2,3)

group by depID

having sum(salary ) < 14000;

**6. Find total salary in department 1 and 3 and group totalSalary should be grater than 13000**

select depID,sum(salary) as totalsalary from emp\_details

where depId in(1,3)

group by depID

having sum(salary ) > 13000;

**Find MAX salary in department 1 and 3 and group max\_Salary should be grater than 7000**

**Find avg salary in department 1 ,2,3 and group avg\_Salary should be grater than 4500**