

MySQL Aggregate Functions and Group by- Exercises, Practice, Solution

1.] Write a query to list the number of jobs available in the employees table.

⇒ Output:

```
SELECT COUNT(DISTINCT JOB_ID ) FROM employees
```

COUNT(DISTINCT JOB_ID)
19

2.] Write a query to get the total salaries payable to employees.

⇒ Output:

```
SELECT SUM(SALARY) FROM employees;
```

SUM(SALARY)
691400.00

3.] Write a query to get the minimum salary from employees table.

⇒ Output:

```
SELECT MIN(SALARY) FROM employees;
```

MIN(SALARY)
2100.00

4.] Write a query to get the maximum salary of an employee working as a Programmer.

⇒ Output:

```
SELECT MAX(SALARY) FROM employees;
```

MAX(SALARY)
24000.00

5.] Write a query to get the average salary and number of employees working the department 90.

⇒ Output:

```
SELECT AVG(SALARY),COUNT(*) FROM employees WHERE DEPARTMENT_ID;
```

AVG(SALARY)	COUNT(*)
6456.603774	106

6.] Write a query to get the highest, lowest, sum, and average salary of all employees.

⇒ Output:

```
SELECT ROUND(MAX(SALARY))AS MAXIMUM,ROUND(MIN(SALARY)) AS  
MINIMUM,ROUND(SUM(SALARY)) AS SUM_OF_SALARY,ROUND(AVG(SALARY)) AS  
AVGRAGE_OF_SALARY FROM employees;
```

	MAXIMUM	MINIMUM	SUM_OF_SALARY	AVGRAGE_OF_SALARY
	24000	2100	691400	6462

7.] Write a query to get the number of employees with the same job.

⇒ Output:

```
SELECT JOB_ID, COUNT(*) AS SAME_JOB FROM employees GROUP BY JOB_ID;
```

JOB_ID	SAME_JOB
SA_MAN	5
SA_REP	30
SH_CLERK	20
ST_CLERK	20
ST_MAN	5

8.] Write a query to get the difference between the highest and lowest salaries.

⇒ Output:

```
SELECT MAX(SALARY) - MIN(SALARY) AS DIFF FROM employees;
```

DIFF
21900.00

9.] Write a query to find the manager ID and the salary of the lowest-paid employee for that manager.

⇒ Output:

```
SELECT MANAGER_ID, MIN(SALARY) FROM employees WHERE MANAGER_ID IS NOT NULL GROUP BY MANAGER_ID ORDER BY MANAGER_ID;
```

MANAGER_ID	MIN(SALARY)
0	24000.00
100	5800.00
101	4400.00
102	9000.00
103	4200.00

10.] Write a query to get the department ID and the total salary payable in each department.

⇒ Output:

```
SELECT DEPARTMENT_ID, SUM(SALARY) FROM employees GROUP BY DEPARTMENT_ID;
```

DEPARTMENT_ID	SUM(SALARY)
0	7000.00
10	4400.00
20	19000.00
30	24900.00
40	6500.00

11.] Write a query to get the average salary for each job ID excluding programmer.

⇒ Output:

```
SELECT JOB_ID, AVG(SALARY) FROM employees WHERE JOB_ID <> 'IT_PROG' GROUP BY JOB_ID;
```

JOB_ID	AVG(SALARY)
AC_ACCOUNT	8300.000000
AC_MGR	12000.000000
AD_ASST	4400.000000
AD_PRES	24000.000000
AD_VP	17000.000000

12.] Write a query to get the total salary, maximum, minimum, average salary of employees (job ID wise), for department ID 90 only.

⇒ Output:

```
SELECT JOB_ID , SUM(SALARY)AS TOTAL, AVG(SALARY) AS AVERAGE, MAX(SALARY) AS MAXIMUM,  
MIN(SALARY) AS MINIMUM FROM employees WHERE DEPARTMENT_ID GROUP BY JOB_ID;
```

	JOB_ID	TOTAL	AVERAGE	MAXIMUM	MINIMUM
▶	AC_ACCOUNT	8300.00	8300.000000	8300.00	8300.00
	AC_MGR	12000.00	12000.000000	12000.00	12000.00
	AD_ASST	4400.00	4400.000000	4400.00	4400.00
	AD_PRES	24000.00	24000.000000	24000.00	24000.00
	AD_VP	34000.00	17000.000000	17000.00	17000.00

13.] Write a query to get the job ID and maximum salary of the employees where maximum salary is greater than or equal to \$4000.

⇒ Output:

```
SELECT JOB_ID , MAX(SALARY) FROM employees GROUP BY JOB_ID HAVING MAX(SALARY) >= 4000;
```

	JOB_ID	MAX(SALARY)
▶	AC_ACCOUNT	8300.00
	AC_MGR	12000.00
	AD_ASST	4400.00
	AD_PRES	24000.00
	AD_VP	17000.00

14.] Write a query to get the average salary for all departments employing more than 10 employees.

⇒ Output:

```
SELECT DEPARTMENT_ID, AVG(SALARY),COUNT(*) FROM employees GROUP BY DEPARTMENT_ID  
HAVING COUNT(*) > 10;
```

	DEPARTMENT_ID	AVG(SALARY)	COUNT(*)
▶	50	3475.555556	45
	80	8955.882353	34