

EXPRIMENT – 3

SQL query based on Group by clause

1. **Display job ID of jobs that were done by more than 3 employees for more than 100 days.**

```
SELECT job_id
FROM job_history
GROUP BY job_id
HAVING COUNT(DISTINCT employee_id) > 3
AND COUNT(DISTINCT start_date) > 100;
```

no data found

2. **Display departments in which more than five employees have commission percentage.**

```
SELECT department_id
FROM employees
WHERE commission_pct IS NOT NULL
GROUP BY department_id
HAVING COUNT(*) > 5;
```

DEPARTMENT_ID
80

1 rows returned in 0.00 seconds

[CSV Export](#)

3. **Display job ID, number of employees, sum of salary, and difference between highest salary and lowest salary of the employees of the job**

```
SELECT job_id, COUNT(employee_id) AS num_employees,
SUM(salary) AS sum_salary,
MAX(salary) - MIN(salary) AS salary_range
FROM employees
GROUP BY job_id;
```

JOB_ID	NUM_EMPLOYEES	SUM_SALARY	SALARY_RANGE
IT_PROG	5	28800	4800
AC_MGR	1	12000	0
AC_ACCOUNT	1	8300	0
ST_MAN	5	36400	2400
PU_MAN	1	11000	0
AD_ASST	1	4400	0
AD_VP	2	34000	0
SH_CLERK	20	64300	1700
FI_ACCOUNT	5	39600	2100
FI_MGR	1	12000	0
PU_CLERK	5	13900	600
SA_MAN	5	61000	3500
MK_MAN	1	13000	0
PR_REP	1	10000	0
AD_PRES	1	24000	0
SA_REP	30	250500	5400
MK_REP	1	6000	0
ST_CLERK	20	55700	1500
HR_REP	1	6500	0

19 rows returned in 0.02 seconds

[CSV Export](#)

4. **Display the details of departments in which the max salary is greater than 10000 for employees who did a job in the past.**

```
SELECT department_id, MAX(salary) AS max_salary
FROM employees
WHERE job_id IN (SELECT DISTINCT job_id FROM job_history)
GROUP BY department_id
HAVING MAX(salary) > 10000;
```

DEPARTMENT_ID	MAX_SALARY
110	12000
80	14000

2 rows returned in 0.02 seconds

[CSV Export](#)

5. **Display details of manager who manages more than 5 employee**

```
SELECT manager_id, COUNT(employee_id) AS num_employees_managed
FROM employees
WHERE manager_id IS NOT NULL
GROUP BY manager_id
HAVING COUNT(employee_id) > 5;
```

MANAGER_ID	NUM_EMPLOYEES_MANAGED
100	14
120	8
121	8
122	8
123	8
124	8
145	6
146	6
147	6
148	6
149	6

11 rows returned in 0.02 seconds

[CSV Export](#)