

### **For Human Resources Data Base**

**15.** Find the average, max, min and sum of the salary's of REPresentatives.

```
SELECT    MIN(salary), MAX(salary), AVG(salary), SUM(salary)
FROM      employees
WHERE     job_id LIKE '%REP5';
```

**16.** Find the minimum and maximum hire dates of employees. (most recently and the oldest hire dates)

```
SELECT    MIN(hire_date), MAX(hire_date)
FROM      employees;
```

**17.** Find the number of employees working in the department 50.

```
SELECT    COUNT(*)
FROM      employees
WHERE     department_id=50;
```

**18.** Find the number of employees taking commissions in department 50. (Take care of null values)

```
SELECT    COUNT(commission_pct)
FROM      employees
WHERE     department_id=50;
```

```
SELECT    COUNT(*)
FROM      employees
WHERE     department_id=50 AND commission_pct IS NOT NULL;
```

**19.** Find the number of departments in those some employees working for.

```
SELECT    COUNT(distinct department_id)
FROM      employees;
```

**20.** Find names, job\_id and salary of employees whose salary is greater than the minimum salary of department 50.

```
SELECT    last_name, job_id, salary
FROM      employees
WHERE     salary =      (SELECT MIN(salary)
                        FROM employees);
```

**21.** Find the department IDS and minimum salaries of those departments which have higher minimum salary than the department 50's minimum salary.

```
SELECT    department_id, MIN(salary)
FROM      employees
GROUP BY  department_id
HAVING    MIN(salary) > (SELECT MIN(salary)
                        FROM employees
                        WHERE department_id=50);
```

**22.** Find the average commission\_pct of employees. (Real average)

```
SELECT    AVG(commission_pct)
FROM      employees;
```

**23.** Find the average commission\_pct of all employees. (General average)

```
SELECT    AVG(NVL(commission_pct,0))
FROM      employees;
```

**24.** Find the average salaries of all departments.

```
SELECT    department_id, AVG(salary)
FROM      employees
GROUP BY  department_id;
```

**25.** For every department those have department\_id greater than 40, find the sum of the salaries of every job.

```
SELECT    department_id, job_id, SUM(salary)
FROM      employees
WHERE     department_id >40
GROUP BY  department_id, job_id
ORDER BY  department_id;
```

**26.** For every department, find the maximum salaries, if it is greater than 10.000.

```
SELECT    department_id, MAX(salary)
FROM      employees
GROUP BY  department_id
HAVING    MAX(salary)>10000;
```

**27.** List the jobs and the sum of the salary's of that job, if it is more than 1300\$ and they are not any kind of representatives.

```
SELECT    job_id, SUM(salary)
FROM      employees
WHERE     job_id NOT LIKE '%REP%'
GROUP BY  job_id
HAVING    SUM(salary)>1300
ORDER BY  SUM(salary);
```

**28.** Display the maximum average salary.

```
SELECT    MAX(AVG(salary))
FROM      employees
GROUP BY  department_id;
```

**29.** Display the details of the employees who are managed by the same manager and work in the same department as employees with the first name of "John".

PAIRWISE SOLUTION:

```
SELECT    employee_id, manager_id, department_id
FROM      employees
WHERE     (manager_id, department_id) IN
        (SELECT manager_id, department_id
         FROM employees
         WHERE first_name= 'John')
```

AND first\_name <> 'John';

NONPAIRWISE SOLUTION:

```
SELECT    employee_id, manager_id, department_id
FROM      employees
WHERE     manager_id IN
                (SELECT manager_id
                 FROM employees
                 WHERE first_name= 'John')
        department_id IN
                (SELECT department_id
                 FROM employees
                 WHERE first_name= 'John')
AND first_name <> 'John';
```

- 30.** List the employee\_id, last name and country information such that country is CANADA for location number 1800 and USA otherwise.

```
SELECT    employee_id, last_name
        ( CASE  WHEN department_id = (SELECT department_id
                                     FROM departments
                                     WHERE location_id= 1800)
                THEN 'CANADA' ELSE 'USA' END) location
FROM      employees;
```

- 31.** Find all employees who earn more than the average salary in their department.

```
SELECT    last_name, salary, department_id
FROM      employees outer_table
WHERE     salary > (SELECT AVG(salary)
                  FROM employees inner_table
                  WHERE inner_table.department_id = outer_table.department_id);
```

- 32.** Display details of those employees who have changed jobs at least twice.

```
SELECT    e.employee_id, last_name, e.job_id
FROM      employees e
WHERE     2 <= (SELECT count(*)
               FROM job_history
               WHERE employee_id = e.employee_id);
```

- 33.** Find the name's, job\_id's and department\_id's of the managers.

```
SELECT    employee_id, last_name, job_id, department_id
FROM      employees outer
WHERE     EXISTS (SELECT 'X'
                 FROM employees
                 WHERE manager_id = outer.employee_id);
```

- 34.** List the names of employees, their commission-pct and salaries and new salaries calculated as %10 rise if they have commission, %20 rise if they do not have commission. (use NVL2)

```
SELECT first_name, last_name, commission_pct, salary,  
       NVL2(commission_pct,salary*1.1,salary*1.2)
```

"new salary"

```
FROM employees;
```

- 35.** Find the names of departments which has higher average salary than the overall average salary of the company.

```
SELECT      department_name, AVG(salary)  
FROM        departments JOIN employees USING (department_id)  
GROUP BY    department_name  
HAVING      AVG(salary) > (SELECT AVG(salary)  
                           FROM employees);
```