Given Schema

- EMPLOYEES (employee_id, first_name, last_name, email, phone_number, hire_date, job_id, salary, commission_pct, manager_id, department_id)
- DEPARTMENTS (department_id, department_name, manager_id, location_id)
- LOCATIONS (location_id, street_address, postal_code, city, state_province, country_id)
- COUNTRIES (country_id, country_name, region_id)

1. Write a query to display the names (first_name, last_name) using an alias name “First Name", "Last Name"

Solution:

SELECT first_name "First Name", last_name "Last Name" FROM employees;

2. Write a query to get unique department ID from employee table.

Solution:

SELECT DISTINCT department_id FROM employees;

3. Write a query to get the details of all employees according to first name in descending order.

Solution:

SELECT * FROM employees ORDER BY first_name DESC;

4. Write a query to get the names (first_name, last_name), salary, PF of all the employees (PF is calculated as 15% of salary).

Solution:

SELECT first_name, last_name, salary, salary*.15 PF FROM employees;

5. Write a query to get the total salaries payable to employees.

Solution:

SELECT SUM(salary) FROM employees;

6. Write a query to get the number of employees working with the company.

Solution:

SELECT COUNT(*) FROM employees;

7. Write a query get all first names from the employees table in upper case.

Solution:

SELECT UPPER(first_name) FROM employees;

8. Write a query to get the first three characters of the first name of all employees.

Solution:

SELECT SUBSTRING(first_name,1,3) FROM employees;

9. Write a query to get the first name of all employees table after removing white spaces from both sides.

Solution:

SELECT TRIM(first_name) FROM employees;

10. Write a query to select first 10 records from a table.

Solution:

SELECT employee_id, first_name FROM employees LIMIT 10;

11. Write a query to display the names (first_name, last_name) and salary for all employees whose salary is not in the range $10,000 through $15,000.

Solution:

SELECT first_name, last_name, salary FROM employees WHERE salary NOT BETWEEN 10000 AND 15000;

12. Write a query to display the name (first_name, last_name) and department ID of all employees in departments 30 or 100 in ascending order.

Solution:

SELECT first_name, last_name, department_id FROM employees WHERE department_id IN (30, 100) ORDER BY department_id ASC;

13. Write a query to display the name (first_name, last_name) and hire date for all employees who were hired in 1987.

Solution:

SELECT first_name, last_name, hire_date FROM employees WHERE YEAR(hire_date) LIKE '1987%';

14. Write a query to display the last name, job, and salary for all employees whose job is that of a Programmer or a Shipping Clerk, and salary is not equal to $4,500, $10,000, or $15,000.

Solution:

SELECT last_name, job_id, salary FROM employees

WHERE job_id IN ('IT_PROG', 'SH_CLERK') AND salary NOT IN (4500,10000, 15000);

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

Solution:

SELECT COUNT(DISTINCT job_id) FROM employees;

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

Solution:

SELECT MAX(salary) FROM employees WHERE job_id = 'IT_PROG';

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

Solution:

```
SELECT ROUND(MAX(salary),0) 'Maximum', ROUND(MIN(salary),0) 'Minimum',
ROUND(SUM(salary),0) 'Sum', ROUND(AVG(salary),0) 'Average' FROM employees;
```

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

Solution:

```
SELECT job_id, COUNT(*) FROM employees GROUP BY job_id;
```

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

Solution:

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

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

Solution:

```
SELECT job_id, MAX(salary) FROM employees
GROUP BY job_id HAVING MAX(salary) >=4000;
```

21. Write a query to find the name (first_name, last_name) of all employees who works in the IT department.

Solution:

```
SELECT first_name, last_name FROM employees WHERE department_id IN
(SELECT department_id FROM departments WHERE department_name='IT');
```

22. Write a query to select last 10 records from a table.

Solution:

```
SELECT * FROM ( SELECT * FROM employees ORDER BY employee_id DESC LIMIT 10)
sub ORDER BY employee_id ASC;
```

23. Write a query to find the name (first_name, last_name) of the employees who have a manager and worked in a USA based department.

Solution:

```
SELECT first_name, last_name FROM employees WHERE manager_id IN
(SELECT employee_id FROM employees WHERE department_id IN
(SELECT department_id FROM departments WHERE location_id IN
(SELECT location_id from locations where country_id='US')));
```

24. Write a query to find the name (first_name, last_name), and salary of the employees whose salary is greater than the average salary.

Solution:

```
SELECT first_name, last_name, salary FROM employees
WHERE salary > (SELECT AVG(salary) FROM employees);
```

25. Write a query to display the employee ID, first name, last name, and department names of all employees.

Solution:

```
SELECT employee_id, first_name, last_name,
(SELECT department_name FROM departments d WHERE e.department_id = d.department_id)
department  FROM employees e ORDER BY department;
```

26. Write a query to find the addresses (location_id, street_address, city, state_province, country_name) of all the departments.

Solution:

```
SELECT location_id, street_address, city, state_province, country_name
FROM locations NATURAL JOIN countries;
```

27. Write a query to find the name (first_name, last_name), job, department ID and name of the employees who works in London.

Solution:

```
SELECT e.first_name, e.last_name, e.job_id, e.department_id, d.department_name
FROM employees e JOIN departments d ON (e.department_id = d.department_id)
JOIN locations l ON (d.location_id = l.location_id)
WHERE LOWER(l.city) = 'London';
```

28. Write a query to find the name (first_name, last_name) and hire date of the employees who was hired after 'Jones'.

Solution:

```
SELECT e.first_name, e.last_name, e.hire_date FROM employees e
JOIN employees davies ON (davies.last_name = 'Jones')
WHERE davies.hire_date < e.hire_date;
```

29. Write a query to display the first day of the month (in datetime format) three months before the current month.

Solution:

```
SELECT date(((PERIOD_ADD (EXTRACT(YEAR_MONTH
FROM CURDATE()),-3)*100)+1));
```

****