TODAYS QUESTION:

EMPNO ENAME JOB MGR HIREDATE SAL COMM DEPTNO

---------- ---------- --------- ---------- --------- ---------- ---------- ----------

7369 SMITH CLERK 7902 17-DEC-80 800 20

7499 ALLEN SALESMAN 7698 20-FEB-81 1600 300 30

7521 WARD SALESMAN 7698 22-FEB-81 1250 500 30

7566 JONES MANAGER 7839 02-APR-81 2975 20

7654 MARTIN SALESMAN 7698 28-SEP-81 1250 1400 30

7698 BLAKE MANAGER 7839 01-MAY-81 2850 30

7782 CLARK MANAGER 7839 09-JUN-81 2450 10

7788 SCOTT ANALYST 7566 19-APR-87 3000 20

7839 KING PRESIDENT 17-NOV-81 5000 10

7844 TURNER SALESMAN 7698 08-SEP-81 1500 0 30

7876 ADAMS CLERK 7788 23-MAY-87 1100 20

7900 JAMES CLERK 7698 03-DEC-81 950 30

7902 FORD ANALYST 7566 03-DEC-81 3000 20

7934 MILLER CLERK 7782 23-JAN-82 1300 10

1. WRITE A QUERY TO DISPLAY ALL THE DETAILS FROM THE EMPLOYEE TABLE;

SQL> SELECT \* FROM EMP;

1. WAQTD NAMES OF ALL THE EMPLOYEES;

SQL> SELECT ENAME FROM EMP;

1. WAQTD NAME AND SALARY GINVEN TO ALL THE EMPLOYEES.

SQL> SELECT ENAME,SAL FROM EMP;

1. WAQTD NAME AND COMMISSION GIVEN TO ALL THE EMPLOYEES;

SQL> SELECT ENAME,COMM FROM EMP;

1. WAQTD EMPLOYEE ID AND DEPARTMENT NUMBER OF ALL THE EMPLOYEES IN EMP TABLE;

SQL> SELECT EMPNO,DEPTNO FROM EMP;

1. WAQTD ENAME AND HIREDATE OF ALL THE EMPLOYEES

SQL> SELECT ENAME,HIREDATE FROM EMP;

1. WAQTD NAME AND DESIGNATION OF ALL THE EMPLOYEES;

SQL> SELECT ENAME,JOB FROM EMP;

1. WAQTD NAME, JOB AND SALARY GIVEN ALL THE EMPLOYEES.

SQL> SELECT ENAME,JOB,SAL FROM EMP;

1. WAQTD DNAMES PRESENT IN DEPARTMENT TABLE;

SQL> SELECT \* FROM DEPT;

DEPTNO DNAME LOC

---------- -------------- -------------

10 ACCOUNTING NEW YORK

20 RESEARCH DALLAS

30 SALES CHICAGO

40 OPERATIONS BOSTON

SQL> SELECT DNAME FROM DEPT;

1. WAQTD DNAME AND LOCATION PRESENT IN DEPT TABLE.

SQL> SELECT DNAME,LOC FROM DEPT;

ANOTHER 25 PRACTICE QUESTIONS

SQL> select \* from tab;

TNAME TABTYPE CLUSTERID

------------------------------ ------- ----------

COUNTRIES TABLE

DEPARTMENTS TABLE

EMPLOYEES TABLE

EMP\_DETAILS\_VIEW VIEW

JOBS TABLE

JOB\_HISTORY TABLE

LOCATIONS TABLE

REGIONS TABLE

8 rows selected.

SQL> desc employees;

Name Null? Type

EMPLOYEE\_ID NOT NULL NUMBER(6)

FIRST\_NAME VARCHAR2(20)

LAST\_NAME NOT NULL VARCHAR2(25)

EMAIL NOT NULL VARCHAR2(25)

PHONE\_NUMBER VARCHAR2(20)

HIRE\_DATE NOT NULL DATE

JOB\_ID NOT NULL VARCHAR2(10)

SALARY NUMBER(8,2)

COMMISSION\_PCT NUMBER(2,2)

MANAGER\_ID NUMBER(6)

DEPARTMENT\_ID NUMBER(4)

SQL> --q1(1. Display the employee ID and first name of all employees.)

SQL> select employee\_id,first\_name from employees;

SQL> --(Q2. Show the last name and hire date of all employees.)

SQL> select last\_name,hire\_date from employees;

SQL> --(Q3. List the job ID and salary for all employees.)

SQL> select job\_id,salary from employees;

SQL> --(Q4. Display the email and phone number of all employees.)

SQL> select email,phone\_number from employees;

SQL> --(Q5. Show the department name and location ID of all departments)

SQL> desc departments; Name Null? Type

DEPARTMENT\_ID NOT NULL NUMBER(4) DEPARTMENT\_NAME NOT NULL VARCHAR2(30) MANAGER\_ID NUMBER(4) LOCATION\_ID NUMBER(4)

SQL> select department\_name,location\_id from departments;

SQL> --(Q6. Retrieve the employee ID, first name, and last name of all employees.)

SQL> select employee\_id,first\_name,last\_name from employees;

SQL> --(Q7. Show the job title and minimum salary from the jobs table.)

SQL> desc jobs;

Name Null? Type

JOB\_ID NOT NULL VARCHAR2(10)

JOB\_TITLE NOT NULL VARCHAR2(35)

MIN\_SALARY NUMBER(6)

MAX\_SALARY NUMBER(6)

SQL> select job\_title,min\_salary from jobs;

SQL> --(Q8. List the commission percentage and salary of all employees)

SQL> select commission\_pct,salary from employees;

SQL> --(Q9. Display the department name and manager ID for all departments.)

SQL> select department\_name,manager\_id from departments;

SQL> --(Q10. Retrieve the first name and job ID of all employees.)

SQL> select first\_name,job\_id from employees;

SQL> Q11. Show the job ID and maximum salary from the jobs table.

SQL> select job\_id,max\_salary from jobs;

SQL> Q12. List the department ID and department name from the departments table.

SQL> select department\_id,department\_name from departments;

SQL> Q13. Display the employee ID and department ID for all employees.

SQL> select employee\_id,department\_id from employees;

SQL> --(Q14. Show the city and state province )

SQL> desc REGION;

Name Null? Type

REGION\_ID NOT NULL NUMBER(2)

REGION\_NAME VARCHAR2(25)

SQL> desc locations; Name Null? Type

LOCATION\_ID NOT NULL NUMBER(4) STREET\_ADDRESS VARCHAR2(40) POSTAL\_CODE VARCHAR2(12) CITY NOT NULL VARCHAR2(30) STATE\_PROVINCE VARCHAR2(25) COUNTRY\_ID CHAR(2)

SQL> select city,state\_province from locations;

SQL> --(Q15. List the employee ID and start date )

SQL> select employee\_id,hire\_date from employees;

SQL> --(Q16. Retrieve the first name, last name, and email of all employees.)

SQL> select first\_name,last\_name,email from employees;

SQL> --(Q17. Display the job title and department ID of all employees.)

SQL> select job\_id,department\_id from employees;

18. Show the department ID and location ID

SQL> select department\_id,location\_id from departments;

1. List the job ID and job title

SQL> select job\_id,job\_title from jobs;

1. Retrieve the location ID and city name

SQL> select location\_id,city from locations;

1. Show the employee ID and salary from the employees table.

SQL> select employee\_id,salary from employees;

1. Display the country ID and region ID from the countries table.

SQL> select country\_id,region\_id from countries;

1. List the employee ID, start date, and end date of the job.

SQL> select employee\_id,start\_date,end\_date from job\_history;

1. Show the first name and last name from the employees table.

SQL> select first\_name,last\_name from employees;

1. Retrieve the location ID and postal code from the locations table.

SQL> select location\_id,postal\_code from locations;

**Question Related to Expression[Alias]:**

### ****1. WAQTD NAME OF THE EMPLOYEE ALONG WITH THEIR ANNUAL SALARY.****

SELECT ENAME, (SAL \* 12) AS ANNUAL\_SALARY FROM EMP;

### ****2. WAQTD ENAME AND JOB FOR ALL THE EMPLOYEES WITH THEIR HALF TERM SALARY.****

SELECT ENAME, JOB, (SAL \* 6) AS HALF\_SALARY FROM EMP;

### ****3. WAQTD ALL THE DETAILS OF THE EMPLOYEES ALONG WITH AN ANNUAL BONUS OF 2000.****

SELECT EMP.\*, (SAL + 2000) AS AFTER\_ANNUAL\_BONUS FROM EMP;

### ****4. WAQTD NAME, SALARY, AND SALARY WITH A HIKE OF 10%.****

SELECT ENAME, SAL, (SAL \* 1.1) AS SALARY\_AFTER\_HIKE FROM EMP;

### ****5. WAQTD NAME AND SALARY WITH A DEDUCTION OF 25%.****

SELECT ENAME, (SAL \* 12 \* 0.75) AS SALARY\_AFTER\_DEDUCT FROM EMP;

### ****6. WAQTD NAME AND SALARY WITH A MONTHLY HIKE OF 50.****

SELECT ENAME, (SAL + 12 \* 50) AS SALARY\_WITH\_MONTHLY\_HIKE FROM EMP;

### ****7. WAQTD NAME AND ANNUAL SALARY WITH A DEDUCTION OF 10%.****

SELECT ENAME, (SAL \* 12 \* 0.9) AS SALARY\_AFTER\_DEDUCT FROM EMP;

### ****8. WAQTD TOTAL SALARY GIVEN TO EACH EMPLOYEE (SAL + COMM).****

SELECT ENAME, (SAL + NVL(COMM, 0)) AS TOTAL\_SALARY FROM EMP;

### ****9. WAQTD DETAILS OF ALL THE EMPLOYEES ALONG WITH ANNUAL SALARY.****

SELECT EMP.\*, (SAL \* 12) AS ANNUAL\_SALARY FROM EMP;

### ****10. WAQTD NAME AND DESIGNATION ALONG WITH 100 PENALTY IN SALARY.****

SELECT ENAME, JOB, (SAL - 100) AS SALARY\_AFTER\_PENALTY FROM EMP;

### 1. Employee ID and salary after adding $1000 bonus.

SELECT EMPLOYEE\_ID, (SALARY + 1000) AS BONUS\_AFTER\_SALARY FROM EMPLOYEES;

### 2. Employee ID and annual salary (monthly salary \* 12).

SELECT EMPLOYEE\_ID, (SALARY \* 12) AS ANNUAL\_SALARY FROM EMPLOYEES;

### 3. Employee ID and salary after a 10% hike.

SELECT EMPLOYEE\_ID, (SALARY \* 1.1) AS SALARY\_AFTER\_HIKE FROM EMPLOYEES;

### 4. Employee ID and salary after 5% tax deduction.

SELECT EMPLOYEE\_ID, (SALARY \* 0.95) AS SALARY\_AFTER\_TAX FROM EMPLOYEES;

### 5. Employee ID and salary after a 20% reduction.

SELECT EMPLOYEE\_ID, (SALARY \* 0.8) AS SALARY\_AFTER\_TEMPORARY\_CUT FROM EMPLOYEES;

### 6. Employee ID and salary after a 15% raise.

SELECT EMPLOYEE\_ID, (SALARY \* 1.15) AS SALARY\_RAISE FROM EMPLOYEES;

### 7. Employee ID and half-year salary (monthly salary \* 6).

SELECT EMPLOYEE\_ID, (SALARY \* 6) AS HALF\_SALARY FROM EMPLOYEES;

### 8. Employee ID and salary divided by 2 (part-time role).

SELECT EMPLOYEE\_ID, (SALARY / 2) AS PART\_TIME\_SALARY FROM EMPLOYEES;

### 9. Employee ID and salary after a 15% increase.

SELECT EMPLOYEE\_ID, SALARY, (SALARY \* 1.15) AS INCREASED\_SALARY FROM EMPLOYEES;

### 10. Employee ID and salary after a 25% raise.

SELECT EMPLOYEE\_ID, SALARY, (SALARY \* 1.25) AS INCREASED\_SALARY FROM EMPLOYEES;

### 11. Employee ID and salary after a $500 deduction.

SELECT EMPLOYEE\_ID, SALARY, (SALARY - 500) AS DEDUCTED\_SALARY FROM EMPLOYEES;

### 12. Employee ID and three-month salary.

SELECT EMPLOYEE\_ID, SALARY, (SALARY \* 3) AS THREE\_MONTH\_SALARY FROM EMPLOYEES;

### 13. Employee ID and salary after a 30% reduction.

SELECT EMPLOYEE\_ID, SALARY, (SALARY \* 0.7) AS REDUCED\_SALARY FROM EMPLOYEES;

### 14. Employee ID and salary after a 10% cut.

SELECT EMPLOYEE\_ID, SALARY, (SALARY \* 0.9) AS CUT\_SALARY FROM EMPLOYEES;

### 15. Employee ID and difference between salary and $50,000.

SELECT EMPLOYEE\_ID, SALARY, (50000 - SALARY) AS DIFFERENCE\_SALARY FROM EMPLOYEES;

### 16. Employee ID and total salary over 18 months.

SELECT EMPLOYEE\_ID, SALARY, (SALARY \* 18) AS TOTAL\_SALARY FROM EMPLOYEES;

### 17. Employee ID and monthly salary (salary / 12).

SELECT EMPLOYEE\_ID, SALARY, (SALARY / 12) AS MONTHLY\_SALARY FROM EMPLOYEES;

### 18. Employee ID and salary after $2500 bonus and $500 deduction.

SELECT EMPLOYEE\_ID, SALARY, (SALARY + 2500 - 500) AS BONUS\_SALARY FROM EMPLOYEES;

### 19. Employee ID and salary after 10% hike and $200 deduction.

SELECT EMPLOYEE\_ID, SALARY, (SALARY \* 1.1 - 200) AS CURRENT\_SALARY FROM EMPLOYEES;

### 20. Employee ID and weekly salary (salary / 52).

SELECT EMPLOYEE\_ID, SALARY, (SALARY / 52) AS WEEKLY\_SALARY FROM EMPLOYEES;

### 21. Employee ID and salary after $2000 deduction.

SELECT EMPLOYEE\_ID, SALARY, (SALARY - 2000) AS DIFFERENCE\_SALARY FROM EMPLOYEES;

### 22. Employee ID and salary multiplied by 1.05, then divided by 2.

SELECT EMPLOYEE\_ID, SALARY, (SALARY \* 1.05 / 2) AS DIFFERENCE\_SALARY FROM EMPLOYEES;

### 23. Employee ID and doubled salary.

SELECT EMPLOYEE\_ID, SALARY, (SALARY \* 2) AS DOUBLE\_SALARY FROM EMPLOYEES;

### 24. Employee ID and quarterly salary (monthly salary \* 3).

SELECT EMPLOYEE\_ID, SALARY, (SALARY \* 3) AS THREE\_MONTH\_SALARY FROM EMPLOYEES;

### 25. Employee ID and salary after $5000 deduction and 15% raise.

SELECT EMPLOYEE\_ID, SALARY, ((SALARY - 5000) \* 1.15) AS ADJUSTED\_SALARY FROM EMPLOYEES;

## Where clause

-- 1. Retrieve employees from department 60

SELECT \* FROM employees WHERE department\_id = 60;

-- 2. Find employees with a job ID 'SA\_REP'

SELECT \* FROM employees WHERE job\_id = 'SA\_REP';

-- 3. Get employees earning more than 80,000

SELECT \* FROM employees WHERE salary > 80000;

-- 4. List employees hired after January 1, 2021

SELECT \* FROM employees WHERE hire\_date > TO\_DATE('2021-01-01', 'YYYY-MM-DD');

-- 5. Retrieve employees with the job ID 'FI\_ACCOUNT'

SELECT \* FROM employees WHERE job\_id = 'FI\_ACCOUNT';

-- 6. Find employees who report to manager ID 100

SELECT \* FROM employees WHERE manager\_id = 100;

-- 7. Get employees in department ID 80

SELECT \* FROM employees WHERE department\_id = 80;

-- 8. Find employees with a salary equal to 10,000

SELECT \* FROM employees WHERE salary = 10000;

-- 9. List employees with a commission percentage equal to 0.10

SELECT \* FROM employees WHERE commission\_pct = 0.10;

-- 10. Retrieve employees with department ID 90

SELECT \* FROM employees WHERE department\_id = 90;

-- 11. Find jobs with a minimum salary greater than 5,000

SELECT \* FROM jobs WHERE min\_salary > 5000;

-- 12. Get employees whose name is William

SELECT \* FROM employees WHERE first\_name = 'William';

-- 13. List departments with location ID 1700

SELECT \* FROM departments WHERE location\_id = 1700;

-- 14. Retrieve employees from department ID 100

SELECT \* FROM employees WHERE department\_id = 100;

-- 15. Find jobs where the maximum salary is less than 20,000

SELECT \* FROM jobs WHERE max\_salary < 20000;

-- 16. Get employees hired on or after January 1, 2022

SELECT \* FROM employees WHERE hire\_date >= TO\_DATE('2022-01-01', 'YYYY-MM-DD');

-- 17. Find employees with job ID 'AD\_VP'

SELECT \* FROM employees WHERE job\_id = 'AD\_VP';

-- 18. Retrieve departments with department ID 50

SELECT \* FROM departments WHERE department\_id = 50;

-- 19. Get employees with manager ID 102

SELECT \* FROM employees WHERE manager\_id = 102;

-- 20. Find locations with country ID 'US'

SELECT \* FROM locations WHERE country\_id = 'US';

**1. Details of employees working as Clerk and earning less than 1500:**

SELECT \* FROM EMP WHERE JOB = 'CLERK' AND SAL < 1500;

**2. Name and hire date of employees working as Manager in Dept 30:**

SELECT ENAME, HIREDATE FROM EMP WHERE JOB = 'MANAGER' AND DEPTNO = 30;

**3. Details of employees along with annual salary in Dept 30 as Salesman with an annual salary > 14000:**

SELECT EMP.\*, SAL\*12 AS ANNUAL\_SAL FROM EMP WHERE DEPTNO = 30 AND JOB = 'SALESMAN' AND SAL\*12 > 14000;

**4. All details of employees in Dept 30 or as Analyst:**

SELECT \* FROM EMP WHERE DEPTNO = 30 OR JOB = 'ANALYST';

**5. Names of employees whose salary is < 1100 and designation is Clerk:**

SELECT ENAME FROM EMP WHERE SAL < 1100 AND JOB = 'CLERK';

**6. Name, salary, annual salary, and DeptNo if DeptNo is 20, salary > 1100, and annual salary > 12000:**

SELECT ENAME, SAL, SAL\*12 AS ANNUAL\_SAL, DEPTNO FROM EMP WHERE DEPTNO = 20 AND SAL > 1100 AND SAL\*12 > 12000;

**7. EmpNo and names of employees working as Manager in Dept 20:**

SELECT EMPNO, ENAME FROM EMP WHERE JOB = 'MANAGER' AND DEPTNO = 20;

**8. Details of employees working in Dept 20 or 30:**

SELECT \* FROM EMP WHERE DEPTNO = 20 OR DEPTNO = 30;

**9. Details of employees working as Analyst in Dept 10:**

SELECT \* FROM EMP WHERE JOB = 'ANALYST' AND DEPTNO = 10;

**10. Details of employees working as President with a salary of 4000:**

SELECT \* FROM EMP WHERE JOB = 'PRESIDENT' AND SAL = 4000;

**11. Names, DeptNo, and job of employees working as Clerk in Dept 10 or 20:**

SELECT ENAME, DEPTNO, JOB FROM EMP WHERE JOB = 'CLERK' AND DEPTNO IN (10, 20);

**12. Details of employees working as Clerk or Manager in Dept 10:**

SELECT \* FROM EMP WHERE JOB IN ('MANAGER', 'CLERK') AND DEPTNO = 10;

**13. Names of employees in Dept 10, 20, 30, or 40:**

SELECT ENAME FROM EMP WHERE DEPTNO IN (10, 20, 30, 40);

**14. Details of employees with EmpNo 7902, 7839:**

SELECT \* FROM EMP WHERE EMPNO IN (7902, 7839);

**15. Details of employees working as Manager, Salesman, or Clerk:**

SELECT \* FROM EMP WHERE JOB IN ('MANAGER', 'SALESMAN', 'CLERK');

**16. Names of employees hired after 1981 and before 1987:**

SELECT ENAME FROM EMP WHERE HIREDATE BETWEEN '31-DEC-81' AND '31-DEC-86';

**17. Details of employees earning > 1250 and < 3000:**

SELECT \* FROM EMP WHERE SAL BETWEEN 1251 AND 2999;

**18. Names of employees hired after 1981 in Dept 10 or 30:**

SELECT ENAME FROM EMP WHERE HIREDATE > '31-DEC-81' AND (DEPTNO = 10 OR DEPTNO = 30);

**19. Names and annual salary of employees working as Manager or Clerk in Dept 10 or 30:**

SELECT ENAME, SAL\*12 AS ANNUAL\_SAL FROM EMP WHERE (JOB = 'MANAGER' OR JOB = 'CLERK') AND DEPTNO IN (10, 30);

**20. All details along with annual salary if salary is between 1000 and 4000 and annual salary > 15000:**

SELECT EMP.\*, SAL\*12 AS ANNUAL\_SAL FROM EMP WHERE SAL BETWEEN 1000 AND 4000 AND SAL\*12 > 15000;

**Special Operators:**

**1. Employees whose commission is NULL:**

SELECT \* FROM EMP WHERE COMM IS NULL;

**2. Employees without a reporting manager:**

SELECT \* FROM EMP WHERE MGR IS NULL;

**3. Salesmen in Dept 30:**

SELECT \* FROM EMP WHERE JOB = 'SALESMAN' AND DEPTNO = 30;

**4. Salesmen in Dept 30 earning > 1500:**

SELECT \* FROM EMP WHERE JOB = 'SALESMAN' AND DEPTNO = 30 AND SAL > 1500;

**5. Employees whose name starts with 'S' or 'A':**

SELECT \* FROM EMP WHERE ENAME LIKE 'S%' OR ENAME LIKE 'A%';

**6. Employees not in Dept 10 and 20:**

SELECT \* FROM EMP WHERE DEPTNO NOT IN (10, 20);

**7. Employees whose name does not start with 'S':**

SELECT \* FROM EMP WHERE ENAME NOT LIKE 'S%';

**8. Employees with reporting managers in Dept 10:**

SELECT \* FROM EMP WHERE MGR IS NOT NULL AND DEPTNO = 10;

**9. Employees whose commission is NULL and working as Clerk:**

SELECT \* FROM EMP WHERE COMM IS NULL AND JOB = 'CLERK';

**10. Employees without a reporting manager in Dept 10 or 30:**

SELECT \* FROM EMP WHERE MGR IS NULL AND DEPTNO IN (10, 30);

**11. Salesmen in Dept 30 earning > 2450:**

SELECT \* FROM EMP WHERE JOB = 'SALESMAN' AND DEPTNO = 30 AND SAL > 2450;

**12. Analysts in Dept 20 earning > 2500:**

SELECT \* FROM EMP WHERE JOB = 'ANALYST' AND DEPTNO = 20 AND SAL > 2500;

**13. Employees whose name starts with 'M' or 'J':**

SELECT \* FROM EMP WHERE ENAME LIKE 'M%' OR ENAME LIKE 'J%';

**14. Employees with annual salary except those in Dept 30:**

SELECT EMP.\*, SAL\*12 AS ANNUAL\_SAL FROM EMP WHERE DEPTNO <> 30;

**15. Employees whose name does not end with 'ES' or 'R':**

SELECT \* FROM EMP WHERE ENAME NOT LIKE '%ES' AND ENAME NOT LIKE '%R';

**16. Employees with a 10% hike in salary if they have reporting managers:**

SELECT EMP.\*, SAL+SAL\*0.1 AS HIKE\_SAL FROM EMP WHERE MGR IS NOT NULL;

**17. Salesmen with 'E' as the last but one character and 4-character salary:**

SELECT \* FROM EMP WHERE JOB = 'SALESMAN' AND ENAME LIKE '%E\_' AND SAL LIKE '\_\_\_\_';

**18. Employees hired after 1981:**

SELECT \* FROM EMP WHERE HIREDATE > '31-DEC-81';

**19. Employees hired in February:**

SELECT \* FROM EMP WHERE HIREDATE LIKE '%FEB%';

**20. Employees not working as Managers and Clerks in Dept 10 and 20 with salary between 1000 and 3000:**

SELECT \* FROM EMP WHERE JOB NOT IN ('MANAGER', 'CLERK') AND DEPTNO IN (10, 20) AND SAL BETWEEN 1000 AND 3000;

SQL> select max(sal), min(sal), count(\*), avg(sal) from emp where job = 'PRESIDENT';

SQL> //WAQTD NO OF EMP TOTAL SAL TO EMP AND WHOSE NAME IS CONSECUTIVE 'L' IN THEIR NAME

SQL> SELECT COUNT(\*), SUM(SAL) AS TOTAL\_SAL FROM EMP WHERE ENAME LIKE '%LL%';

SQL> //WAQTD TOTAL SAL AND AVG SAL WHO ARE HIRED IN MONTH OF DEC

SQL> SELECT AVG(SAL), SUM(SAL) AS TOTAL\_SAL FROM EMP WHERE HIREDATE LIKE '%DEC%';

SQL> / WAQTD MAX SAL AND MIN SAL GIVE TO EMP WHO WORK AS 'MANAGER'

SQL> SELECT MIN(SAL), MAX(SAL) FROM EMP WHERE JOB = 'MANAGER';

SQL> /WAQTD NO OF DISTINCT JOB PRESENT IN EMP TABLE

SQL> SELECT COUNT(DISTINCT JOB) FROM EMP;

SQL> //WAQTD NO OF EMP WHO HIRE AFTER 1981 IN DEPT NO 20 OR 30

SQL> SELECT COUNT(\*) FROM EMP WHERE HIREDATE > '31-DEC-81' AND DEPTNO IN(20,30);

SQL> SELECT \* FROM EMP;

SQL> SELECT \* FROM EMP;

SQL> SELECT SUM(SAL) FROM EMP WHERE JOB='CLERK' AND DEPTNO = 30;

SQL> SELECT MAX(SAL) FROM EMP WHERE JOB = 'ANALYST';

SQL> SELECT COUNT(DISTINCT SAL) FROM EMP;

SQL> SELECT AVG(SAL) FROM EMP WHERE JOB = 'CLERK';

SQL> SELECT MIN(SAL) FROM EMP WHERE DEPTNO = 10 AND JOB IN ('MANAGER', 'CLERK');

SQL> SELECT COUNT(\*), JOB FROM EMP GROUP BY JOB;

SQL> //DISPLAY MAX,MIN SAL GIVEN TO EACH JOB IF THE ANNUAL SALARY OF EMP IS MORE THAN 12000;

SQL> SELECT MAX(SAL),MIN(SAL), JOB FROM EMP WHERE SAL\*12>12000 GROUP BY JOB;

SQL> SELECT MAX(SAL) AS MAX\_SAL,MIN(SAL) AS MIN\_SAL, JOB FROM EMP WHERE SAL\*12>12000 GROUP BY JOB;

SQL> SELECT COUNT(\*),DEPTNO FROM EMP WHERE JOB NOT IN 'PRESIDENT' GROUP BY DEPTNO;

SQL> SELECT SUM(SAL),JOB FROM EMP GROUP BY JOB;

SQL> SELECT COUNT(\*), DEPTNO FROM EMP WHERE JOB = 'MANAGER' GROUP BY DEPTNO;

SQL> SELECT AVG(SAL),DEPTNO FROM EMP WHERE DEPTNO NOT IN(20) GROUP BY DEPTNO;

SQL> SELECT COUNT(\*),JOB FROM EMP WHERE ENAME LIKE '%A%' GROUP BY JOB;

SQL> SELECT COUNT(\*),AVG(SAL),DEPTNO FROM EMP WHERE SAL > 2000 GROUP BY DEPTNO;

SQL> SELECT SUM(SAL), JOB FROM EMP WHERE JOB='SALESMAN' GROUP BY JOB;

SQL> SELECT COUNT(\*),MAX(SAL),JOB FROM EMP GROUP BY JOB;

SQL> SELECT MAX(SAL),DEPTNO FROM EMP GROUP BY DEPTNO;

SQL> SELECT COUNT(\*),SAL FROM EMP GROUP BY SAL;

SELECT DEPTNO, COUNT(\*) AS NUM\_OF\_EMPLOYEES FROM EMPLOYEE\_TABLE WHERE DEPTNO IN (SELECT DEPTNO FROM EMPLOYEE\_TABLE WHERE JOB = 'CLERK’ GROUP BY DEPTNO HAVING COUNT(\*) >= 2) GROUP BY DEPTNO;

1. Find the position of '@' in email addresses from the CUSTOMER table.

Select instr(email,’@’,1) from customers;

2. Identify customers whose emails contain 'gmail'.

Select full\_name from customers where instr(email, ‘gmail’, 1) != 0;

3. Find the position of the last occurrence of 'a' in customer names.

Select instr(full\_name, ‘a’, 1, length(full\_name) - length(replace(fullname,’a’))) from customers;

4. Check if phone numbers contain a specific digit sequence (e.g., '123').

Select \* from customer where instr(phno,’123’, 1) != 0;

5. Find customers whose first names contain the letter 'e'.

X -> Select full\_name from customers where substr(full\_name,1,instr(full\_name,’e’,1)) != 0;

Select instr(substr(full\_name, 1, instr(full\_name,’ ‘, 1)-1), ‘e’, 1) != 0;

6. Extract the domain name from email addresses.

Select substr(email,instr(email,’@’,1)+1, instr(email,’com’,1)-1) from customers;

7. Get the first 3 characters of customer names.

Select substr(full\_name, 1, 3) from customers;

8. Extract the last 4 digits of phone numbers.

Select substr(phone,instr(phno,’ ‘,1,2)+1 from customers;

Select substr(phno,instr(phno,’-’, 1,2)+1) from customers;

9. Extract the first name from the full name.

Select substr(full\_name, 1, instr(full\_name, ‘ ‘, 1) - 1) from customers;

10. Extract the last name from the full name.

Select substr(full\_name, instr(full\_name, ‘ ‘, 1)+1) from customers;

11. Extract text before '@' from email addresses.

Select substr(email\_id, 1, instr(email\_id, [‘@’,](mailto:‘@gmail.com’,) 1)-1) from customers;

12. Extract the middle name from full names (if it exists).

Select substr(full\_name,instr(full\_name, ‘ ’, 1)+1, instr(full\_name,’ ’, 1,2)-instr(full\_name,’’, 1,1)) from customers;

13. Swap first and last names in the full name.

Select substr(full\_name, instr(full\_name, ‘’, 1)+1) || ‘ ‘ || substr(full\_name, 1, instr(full\_name, ‘ ‘, 1) - 1) from customers;

14. Extract the area code from phone numbers (assuming format '123-456-7890').

15. Extract the second-level domain from email addresses.

16. Find customers with .com domains in their email.

Select full\_name from customers where instr(email,’.com,1) != 0;

17. Extract the initials of customer names.

Select upper(substr(full\_name,1,1)) || ‘ ‘ || upper(substr(full\_name, instr(full\_name,’ ‘, 1)+1) from customers;

1. Check if the last name contains 'son'.

Select full\_name from customers where substr(full\_name,instr(full\_name,’son, 1)+1) != 0;

1. Extract email usernames shorter than 6 characters.

Select email from customers where length(substr(email,1,(instr(email,’@’,1)-1) < 6;

1. Find customers whose phone numbers end with '999'.

Select full\_name from customers where instr(phno, ‘999’, 1) != 0;

-- 1. Display full names in uppercase and their email in lowercase.

select upper(first\_name) || ' ' || upper(last\_name), lower(email) from employees;

-- 2. Format hire dates as "Month, Year" and show the day of the week.

SELECT TO\_CHAR(HIRE\_DATE, 'Month, YYYY') AS HIRE\_MONTH\_YEAR, TO\_CHAR(HIRE\_DATE, 'Day') AS HIRE\_DAY

FROM EMPLOYEES;

-- 3. Combine employee ID and last name into a string, padded with leading zeros for the ID.

SELECT LPAD(EMPLOYEE\_ID, 6, '0') || '\_' || LAST\_NAME AS ID\_LASTNAME

FROM EMPLOYEES;

-- 4. Calculate the yearly salary of employees and round it to the nearest thousand.

SELECT EMPLOYEE\_ID, ROUND(SALARY \* 12, -3) AS YEARLY\_SALARY

FROM EMPLOYEES;

-- 5. Find employees whose email addresses contain the letter "A" (case-insensitive).

SELECT \*

FROM EMPLOYEES

WHERE LOWER(EMAIL) LIKE '%a%';

-- 6. Extract the first three letters of the first name and concatenate them with the department ID.

SELECT SUBSTR(FIRST\_NAME, 1, 3) || DEPARTMENT\_ID AS FIRST3NAME\_DEPTID

FROM EMPLOYEES;

-- 7. Display the number of years and months an employee has worked (from hire date).

SELECT EMPLOYEE\_ID, TRUNC(MONTHS\_BETWEEN(SYSDATE, HIRE\_DATE) / 12) AS YEARS\_WORKED,TRUNC(MONTHS\_BETWEEN(SYSDATE, HIRE\_DATE) ) AS MONTH\_WORKED, FROM EMPLOYEES;

-- 8. Replace any digits in the phone number with 'X'.

-- 9. Format the salary to include commas as thousand separators and a dollar sign.

SELECT EMPLOYEE\_ID, TO\_CHAR(SALARY, '$99,999.99') AS FORMATTED\_SALARY

FROM EMPLOYEES;

-- 10. Calculate the date 100 days after hire and format it as "Day, DD-Month-YYYY".

SELECT EMPLOYEE\_ID, TO\_CHAR(HIRE\_DATE + 100, 'Day, DD-Month-YYYY') AS FUTURE\_DATE

FROM EMPLOYEES;

-- 11. Convert last names to uppercase and calculate the length of each.

SELECT UPPER(LAST\_NAME) AS LAST\_NAME\_UPPER, LENGTH(LAST\_NAME) AS LAST\_NAME\_LENGTH

FROM EMPLOYEES;

-- 12. Extract the first five characters from email and pad it with asterisks.

SELECT RPAD(SUBSTR(EMAIL, 1, 5), 8, '\*') AS PADDED\_EMAIL

FROM EMPLOYEES;

-- 13. Round the commission percentage to two decimal places.

SELECT EMPLOYEE\_ID, ROUND(SALARY+COMMISSION\_PCT, 2) AS ROUNDED\_COMMISSION FROM EMPLOYEES WHERE COMMISSION\_PCT IS NOT NULL;

-- 14. Calculate the difference between the current date and the hire date in months.

SELECT EMPLOYEE\_ID, MONTHS\_BETWEEN(SYSDATE, HIRE\_DATE) AS MONTHS\_WORKED

FROM EMPLOYEES;

-- 15. Replace all vowels in first names with asterisks.

-- 16. Display the first and last names concatenated, separated by a comma.

SELECT FIRST\_NAME || ', ' || LAST\_NAME AS FULL\_NAME

FROM EMPLOYEES;

-- 17. Format phone numbers to remove special characters and spaces.

-- 18. Display hire dates in the format "YYYY/MM/DD".

SELECT TO\_CHAR(HIRE\_DATE, 'YYYY/MM/DD') AS FORMATTED\_HIRE\_DATE

FROM EMPLOYEES;

-- 19. Replace spaces in job IDs with underscores and convert them to lowercase.

SELECT LOWER(REPLACE(JOB\_ID, ' ', '\_')) AS FORMATTED\_JOB\_ID

FROM EMPLOYEES;

-- 20. Extract the numeric portion of the department ID and double its value.

SUBQUERY:

SQL> select \* from emp where deptno = 20 and job = (select job from emp where ename = 'SMITH');

SQL> select \* from emp where job = 'MANAGER' and deptno = (select deptno from emp where ename = 'TURNER');

SQL> SELECT ENAME,SAL\*12 AS ANNUAL\_SAL FROM EMP WHERE SAL\*12>(SELECT SAL\*12 FROM EMP WHERE ENAME = 'SMITH') AND SAL\*12 < (SELECT SAL\*12 FROM EMP WHERE ENAME = 'KING');

SQL> SELECT ENAME,COMM FROM EMP WHERE COMM IS NOT NULL AND DEPTNO = (SELECT DEPTNO FROM EMP WHERE ENAME = 'TURNER');

SQL> SELECT ENAME FROM EMP WHERE ENAME LIKE 'A%' AND DEPTNO = (SELECT DEPTNO FROM EMP WHERE ENAME = 'BLAKE');

SQL> SELECT ENAME,JOB FROM EMP WHERE JOB='CLERK' AND DEPTNO = (SELECT DEPTNO FROM EMP WHERE ENAME = 'TURNER');

SQL> SELECT \* FROM EMP WHERE SAL > (SELECT SAL FROM EMP WHERE EMANE = 'SCOTT') AND SAL < (SELECT SAL FROM EMP WHERE ENAME = 'KING);

SQL> SELECT \* FROM EMP WHERE SAL > (SELECT SAL FROM EMP WHERE EMANE = 'SCOTT') AND SAL < (SELECT SAL FROM EMP WHERE ENAME = 'KING');

SELECT \* FROM EMP WHERE SAL > (SELECT SAL FROM EMP WHERE EMANE = 'SCOTT') AND SAL < (SELECT SAL FROM EMP WHERE ENAME = 'KING')

SQL> SELECT \* FROM EMP WHERE SAL > (SELECT SAL FROM EMP WHERE ENAME = 'SCOTT') AND SAL < (SELECT SAL FROM EMP WHERE ENAME = 'KING');

no rows selected

SQL> SELECT ENAME,HIREDATE FROM EMP WHERE HIREDATE < '31-DEC-1980' AND HIREDATE > (SELECT HIREDATE FROM EMP WHERE ENAME = 'KING');

no rows selected

4 / 1 / 25

1. Find employees earning more than the average salary.

select \* from employees where salary > (select avg(salary) from employees);

2. Get employees from the same department as 'John'.

SELECT \* FROM EMPLOYEES WHERE DEPARTMENT\_ID IN (SELECT DEPARTMENT\_ID FROM EMPLOYEES WHERE FIRST\_NAME = 'John');

1. List employees who have the same job as 'Alice'.

select \* from employees where job\_id in (select job\_id from employees where first\_name = 'Alice');

1. Get employees whose salary is equal to the highest salary.

select \* from employees where salary = (select max(salary) from employees);

1. Find employees working in department 10 with the highest salary.

select \* from employees where department\_id = 10 and salary = (select max(salary) from employees where department\_id = 10);

7. List employees working in a department with at least 5 employees.

select \* from employees where department\_id in (select department\_id from employees group by department\_id having count(\*) >= 5);

9. Get jobs where employees earn more than 10,000.

select job\_id from employees where salary > 10000;

10. Find employees who have worked in at least 2 jobs.

select \* from employees where employee\_id in (select employee\_id from job\_history group by employee\_id having count(\*)>=2);

1. List employees earning more than anyone in department 50.

select \* from employees where salary > (select max(salary) from employees where department\_id = 50);

1. Retrieve employees from departments where the average salary is above 6000

select \* from employees where department\_id in (select department\_id from employees group by department\_id having avg(salary) > 6000);

14. Find employees whose job title is similar to 'Developer'.

select \* from employees where job\_id in (select job\_id from jobs where job\_title = 'Programmer');

15. Get employees who have worked in different departments.

Right ans- select \* from employees where employee\_id in (select employee\_id from job\_history group by employee\_id having count(distinct department\_id) > 1);

Wrong ans- select \* from employees where department\_id in (select department\_id from employees group by department\_id having count(\*) > 1);

retrieves all employees who work in departments where there is more than one employee

16. List employees whose salary is higher than the highest salary in department 30.

select \* from employees where salary > (select max(salary) from employees where department\_id = 30);

1. Find employees who have the highest salary in their job role.

SELECT \*

FROM employees

WHERE (job\_id, salary) IN (

SELECT job\_id, MAX(salary)

FROM employees

GROUP BY job\_id

);

select \* from employees where salary in (select max(salary) from employees where job\_id in (select job\_id from jobs));

1. List employees working in the same department as an employee earning more than 12,000.

select \* from employees where department\_id in (select department\_id from employees where salary > 12000);

1. WAQTD NAME OF THE EMPLOYEES EARNING S MORE THAN THE SALESMAN

select ename from emp where sal >ALL(select sal from emp where job = 'SALESMAN');

1. WAQTD DETAILS OF THE EMPLOYEES HIRED AFT THE CLERKS

SELECT \* FROM EMP WHERE HIREDATE >ALL(SELECT HIREDATE FROM EMP WHERE JOB = 'CLERK') AND JOB != 'CLERK';

1. WAQTD NAME AND SALARY FOR ALL THE EMPLOYEES IF THEY ARE EARNING LESS THAN AT LEAST A MANAGER

SELECT ENAME,SAL FROM EMP WHERE SAL <ANY(SELECT SAL FROM EMP WHERE JOB = 'MANAGER');

1. WAQTD NAME AND HIREDATE OF EMPLOYEES HIRED BEFORE ALL THE MANAGERS

SELECT ENAME,HIREDATE FROM EMP WHERE HIREDATE <ALL(SELECT HIREDATE FROM EMP WHERE JOB = 'MANAGER');

1. WAQTD NAMES OF THE EMPLOYEES HIRED AFTER ALL THE MANAGERS AND EARNING SALARY MORE THAN ALL THE CLERKS

SELECT ENAME,HIREDATE FROM EMP WHERE HIREDATE >ALL(SELECT HIREDATE FROM EMP WHERE JOB = 'MANAGER') AND SAL >ALL(SELECT SAL FROM EMP WHERE JOB = 'CLERK');

1. WAQTD DETAILS OF THE EMPLOYEES WORKING AS CLERK AND HIRED BEFORE AT LEAST A SALESMAN

SELECT \* FROM EMP WHERE JOB = 'CLERK' AND HIREDATE >ANY(SELECT HIREDATE FROM EMP WHERE JOB = 'SALESMAN');

1. WAQTD DETAILS OF EMPLOYEES WORKING IN ACCOUTING OR SALES DEPT

SELECT \* FROM EMP WHERE DEPTNO IN (SELECT DEPTNO FROM DEPT WHERE DNAME IN ('ACCOUNTING','SALES'));

1. WAQTD DNAME OF THE EMPLOYEES WITH NAME SMITH,KING,MILLER.

SELECT DNAME FROM DEPT WHERE DEPTNO IN (SELECT DEPTNO FROM EMP WHERE ENAME IN('SMITH','KING','MILLER'));

1. WAQTD DETAIS OF EMPLOYEES WORKING IN NEWYORK OR CHICHGO

SELECT \* FROM EMP WHERE DEPTNO IN (SELECT DEPTNO FROM DEPT WHERE LOC IN ('NEW YORK','CHICAGO'));

1. WAQTD EMP NAMES IF EMPLOYEES ARE HIRED AFTER ALL THE EMPLOYEES OF DEPT 10.

SELECT ENAME FROM EMP WHERE HIREDATE >ALL(SELECT HIREDATE FROM EMP WHERE DEPTNO IN (SELECT DEPTNO FROM DEPT WHERE DEPTNO = 10));

**ASSIGNMENT ON INNER JOIN:**

1. Name of the employee and his location of all the employees.

select ename,loc from emp,dept where emp.deptno = dept.deptno;

1. WAQTD DNAME and salary for all the employees working in Accounting.

select dname,sal from emp e, dept d where e.deptno = d.deptno and dname = 'ACCOUNTING';

1. WAQTD DNAME and annual salary for all employees whose salary is more than 2340.

SELECT DNAME,SAL\*12 AS ANNUAL\_SALARY FROM EMP E,DEPT D WHERE E.DEPTNO = D.DEPTNO AND SAL > 2340;

1. WAQTD ENAME and DNAME for employees having character 'A' in their DNAME.

SELECT ENAME,DNAME FROM EMP E, DEPT D WHERE E.DEPTNO = D.DEPTNO AND DNAME LIKE '%A%';

1. WAQTD ENAME and DNAME for all the employees working as Salesman.

SELECT ENAME,DNAME FROM EMP E,DEPT D WHERE E.DEPTNO = D.DEPTNO AND JOB = 'SALESMAN';

1. WAQTD DNAME and job for all the employees whose job and DNAME start with character 'S'.

SELECT DNAME,JOB FROM EMP,DEPT WHERE EMP.DEPTNO = DEPT.DEPTNO AND JOB LIKE 'S%' AND DNAME LIKE 'S%';

1. WAQTD DNAME and MGR\_NO for employees reporting to 7839.

SELECT DNAME,MGR FROM EMP,DEPT WHERE EMP.DEPTNO = DEPT.DEPTNO AND EMPNO IN (SELECT EMPNO FROM EMP WHERE MGR = 7839);

1. WAQTD DNAME and hire date for employees hired after '83 into Accounting or Research departments.

SELECT DNAME,HIREDATE FROM EMP,DEPT WHERE EMP.DEPTNO = DEPT.DEPTNO AND HIREDATE >'31-DEC-1983' AND DNAME IN ('ACCOUNTING','RESEARCH');

1. WAQTD ENAME and DNAME of the employees who are getting commission in DEPT 10 or 30.

SELECT ENAME,DNAME FROM EMP,DEPT WHERE EMP.DEPTNO = DEPT.DEPTNO AND COMM IS NOT NULL AND EMP.DEPTNO IN (10,30);

1. WAQTD DNAME and EMPNO for all the employees whose EMPNO are (7839, 7902) and are working in LOC New York.

SELECT EMPNO,ENAME,DNAME FROM EMP,DEPT WHERE EMP.DEPTNO = DEPT.DEPTNO AND EMPNO IN (7839,7902) AND LOC = 'NEW YORK';

1. Write a query to get employee names along with their department names.

select first\_name,department\_name from employees,departments where employees.department\_id = departments.department\_id;

2. Write a query to get all employees and their job titles.

select first\_name,job\_title from employees,jobs where employees.job\_id = jobs.job\_id;

3. Write a query to find all employees with their location city.

select employees.\*,city from employees,departments,locations where employees.department\_id = departments.department\_id and departments.location\_id = locations.location\_id;

4. Write a query to find all departments and their corresponding location cities.

select departments.\*,city from departments,locations where departments.location\_id = locations.location\_id;

5. Write a query to find all employees, and if they don't have a department, still list them.

select \* from employees inner join departments on employees.department\_id = departments.department\_id;

6. Write a query to get the department names and the employees working in each department, including departments with no employees.

SELECT D.DEPARTMENT\_NAME, E.EMPLOYEE\_ID, E.FIRST\_NAME, E.LAST\_NAME FROM DEPARTMENTS D INNER JOIN EMPLOYEES E ON D.DEPARTMENT\_ID = E.DEPARTMENT\_ID ORDER BY D.DEPARTMENT\_NAME, E.EMPLOYEE\_ID;

7. Write a query to find the employees along with the country they work in.

SELECT FIRST\_NAME,DEPARTMENT\_NAME,COUNTRY\_ID FROM EMPLOYEES,DEPARTMENTS,LOCATIONS WHERE EMPLOYEES.DEPARTMENT\_ID = DEPARTMENTS.DEPARTMENT\_ID AND DEPARTMENTS.LOCATION\_ID = LOCATIONS.LOCATION\_ID;

8. Write a query to list job titles and their minimum and maximum salaries, and any job that doesn't have employees.

SELECT FIRST\_NAME,JOB\_TITLE,MIN\_SALARY,MAX\_SALARY FROM EMPLOYEES INNER JOIN JOBS ON EMPLOYEES.JOB\_ID = JOBS.JOB\_ID;

9. Write a query to find all employees and their region name.

SELECT

E.FIRST\_NAME,

D.DEPARTMENT\_NAME,

C.COUNTRY\_NAME,

R.REGION\_NAME,

L.CITY

FROM

EMPLOYEES E

INNER JOIN

DEPARTMENTS D

ON

E.DEPARTMENT\_ID = D.DEPARTMENT\_ID

INNER JOIN

LOCATIONS L

ON

D.LOCATION\_ID = L.LOCATION\_ID

INNER JOIN

COUNTRIES C

ON

L.COUNTRY\_ID = C.COUNTRY\_ID

INNER JOIN

REGIONS R

ON

C.REGION\_ID = R.REGION\_ID;

1. Write a query to list all departments, their locations, and the countries they belong to.

SELECT

E.FIRST\_NAME,

D.DEPARTMENT\_NAME,

C.COUNTRY\_NAME,

R.REGION\_NAME,

L.CITY

FROM

EMPLOYEES E

INNER JOIN

DEPARTMENTS D

ON

E.DEPARTMENT\_ID = D.DEPARTMENT\_ID

INNER JOIN

LOCATIONS L

ON

D.LOCATION\_ID = L.LOCATION\_ID

INNER JOIN

COUNTRIES C

ON

L.COUNTRY\_ID = C.COUNTRY\_ID

INNER JOIN

REGIONS R

ON

C.REGION\_ID = R.REGION\_ID;

select e1.ename,d1.dname,e2.ename,d2.dname

from emp e1 join emp e2 on e1.mgr = e2.empno

inner join dept d1 on e1.deptno = d1.deptno

inner join dept d2 on e2.deptno = d2.deptno

where d1.loc = d2.loc;

select e1.ename,e1.hiredate,e2.ename,e2.hiredate

from emp e1 join emp e2 on e1.mgr = e2.empno

inner join dept d2 on e2.deptno = d2.deptno

where e2.hiredate < e1.hiredate and d2.dname = 'ACCOUNTING';

select e1.ename,e2.ename,e3.ename

from emp e1 join emp e2 on e1.mgr = e2.empno

join emp e3 on e2.mgr = e3.empno;

select e1.ename,d1.dname,e2.ename,d2.dname

from emp e1 join emp e2 on e1.mgr = e2.empno

inner join dept d1 on e1.deptno = d1.deptno

inner join dept d2 on e2.deptno = d2.deptno

where e2.hiredate < '01-JAN-82';

select e1.ename,d1.dname,e2.ename,d2.loc,e3.ename,d3.loc

from emp e1 join emp e2 on e1.mgr = e2.empno

join emp e3 on e2.mgr = e3.empno

inner join dept d1 on e1.deptno = d1.deptno

inner join dept d2 on e2.deptno = d2.deptno

inner join dept d3 on e3.deptno = d3.deptno;

1. Retrieve the names of all employees along with their manager's name.

select e1.ename as emp\_name, e2.ename as manager\_name

from emp e1 join emp e2 on e1.mgr = e2.empno;

1. List the employees who do not have a manager.

select ename from emp where mgr is null;

1. Display the details of employees whose manager's name is "BLAKE."

select e1.ename

from emp e1 join emp e2

on e1.mgr = e2.empno

where e2.ename = 'BLAKE';

1. Find the total salary of employees working under each manager.

select e2.ename,sum(e1.sal)

from emp e1 join emp e2

on e1.mgr = e2.empno

group by e2.ename;

1. Retrieve the names of employees who have the same job as their manager.

select e1.ename,e1.job,e2.ename,e2.job

from emp e1 join emp e2

on e1.mgr = e2.empno

where e1.job = e2.job;

1. Find the name of the manager for the employee "SCOTT."

select e2.ename

from emp e1 join emp e2

on e1.mgr = e2.empno

where e1.ename = 'SCOTT';

1. Display the employees and their manager details who work in the same department.

select e1.ename,e1.deptno,e2.ename,e2.deptno

from emp e1 join emp e2

on e1.mgr = e2.empno

where e1.deptno = e2.deptno;

1. List the employees who have "CLERK" as a job and their respective manager details.

select e1.ename,e1.job,e2.\*

from emp e1 join emp e2

on e1.mgr = e2.empno

where e1.job = 'CLERK';

1. Retrieve the names of all managers along with their respective employee count.

SELECT E2.ENAME,COUNT(E1.ENAME)

FROM EMP E1 JOIN EMP E2

ON E1.MGR = E2.EMPNO

GROUP BY E2.ENAME;

1. Find employees who have a higher salary than their manager.

SELECT E1.ENAME,E1.SAL,E2.ENAME,E2.SAL

FROM EMP E1 JOIN EMP E2

ON E1.MGR = E2.EMPNO

WHERE E1.SAL > E2.SAL;

1. Display the department details along with the names of employees and their managers.
2. Retrieve the employees whose manager belongs to the "ACCOUNTING" department.
3. Find the employee-manager pairs where the manager's salary is greater than twice the employee's salary.
4. Display the list of employees who have "MANAGER" as a manager’s job.
5. Find the manager and employees whose hire dates fall in the same year.
6. Retrieve employees and their manager details who work in the "SALES" department.
7. Find employees who joined before their manager.
8. Retrieve the name and job of employees whose manager is in the "RESEARCH" department.
9. List all employees who do not report to "KING."
10. Find employees who are in the same department as their manager.
11. Retrieve employees and their manager details who work in different departments.
12. List all employees whose manager’s job title is "PRESIDENT."
13. Display employees and their managers who belong to departments located in "DALLAS."
14. Find employees who share the same manager and list them in groups.
15. Retrieve employees whose managers earn less than 2000.
16. List employees and managers who joined within six months of each other.
17. Display employees with their manager and department details.
18. Retrieve employees whose managers are from the "OPERATIONS" department.
19. Find all managers who have more than three employees reporting to them.
20. Display employees who report directly or indirectly to "KING."
21. Find the employees whose manager’s hire date is after theirs.
22. Retrieve employees who have no one reporting to them.
23. Display employees who have the same manager and the same job.
24. Find the managers of employees whose salaries are more than 1500.
25. Retrieve employees whose managers are in a different department from theirs.
26. Display all employee-manager pairs where the employee's name starts with "M."
27. Find the names of employees and managers who work in "CHICAGO."
28. Retrieve the managers whose employees’ average salary is above 2000.
29. Find employees who do not report to a manager in the same department.
30. List employees whose manager has "MANAGER" as their job and is located in "NEW YORK."

1. Employees, Departments, and Locations

Write a query to display the employee\_id, first\_name, department\_name, and city where the employee works.

select e.employee\_id,e.first\_name,d.department\_name,l.city

from employees e inner join departments d on e.department\_id = d.department\_id

inner join locations l on d.location\_id = l.location\_id;

2. Employees, Jobs, and Job History

Retrieve the employee\_id, first\_name, job\_title (current job), start\_date, and end\_date .

select e.employee\_id,e.first\_name,j.job\_title,jh.start\_date,jh.end\_date

from employees e inner join jobs j on e.job\_id = j.job\_id

inner join job\_history jh on e.employee\_id = jh.employee\_id;

SELECT e.employee\_id, e.first\_name, j.job\_title, jh.start\_date, jh.end\_date

FROM employees e

INNER JOIN jobs j ON e.job\_id = j.job\_id

LEFT JOIN job\_history jh ON e.employee\_id = jh.employee\_id;

3. Find Countries Without Employees

List the country\_name and region\_name for countries that do not have any employees working in them.

select e.first\_name,c.country\_name,r.region\_name

from employees e inner join departments d on e.department\_id = d.department\_id

inner join locations l on d.location\_id = l.location\_id

right join countries c on l.country\_id = c.country\_id

inner join regions r on c.region\_id = r.region\_id

Where e.employee\_id is null;

4. Departments, Managers, and Locations

Write a query to display the department\_id, department\_name, the manager\_name (manager’s full name), and the city where the department is located.

select distinct e2.first\_name || ' ' || e2.last\_name as manager\_name,d2.department\_id,d2.department\_name,l2.city

from employees e1 join employees e2 on e1.manager\_id = e2.employee\_id

inner join departments d2 on e2.department\_id = d2.department\_id

inner join locations l2 on d2.location\_id = l2.location\_id;

5. Employees’ Salary Details Across Regions

Write a query to list the region\_name, country\_name, department\_name, first\_name, last\_name, and salary of employees.

select e.first\_name || ' ' || e.last\_name,e.salary,r.region\_name,c.country\_name,d.department\_name

from employees e inner join departments d on e.department\_id = d.department\_id

inner join locations l on d.location\_id = l.location\_id

inner join countries c on l.country\_id = c.country\_id

inner join regions r on c.region\_id = r.region\_id;