

Queries

1. Display details of jobs where the minimum salary is greater than 10000.

```
SELECT * FROM JOBS WHERE MIN_SALARY > 10000
```

2. Display the first name and join date of the employees who joined between 2002 and 2005.

```
SELECT FIRST_NAME, HIRE_DATE FROM EMPLOYEES  
WHERE TO_CHAR(HIRE_DATE, 'YYYY') BETWEEN 2002 AND 2005 ORDER BY  
HIRE_DATE
```

3. Display first name and join date of the employees who is either IT Programmer or Sales Man.

```
SELECT FIRST_NAME, HIRE_DATE  
FROM EMPLOYEES WHERE JOB_ID IN ('IT_PROG', 'SA_MAN')
```

4. Display employees who joined after 1st January 2008.

```
SELECT * FROM EMPLOYEES where hire_date > '01-jan-2008'
```

5. Display details of employee with ID 150 or 160.

```
SELECT * FROM EMPLOYEES WHERE EMPLOYEE_ID in (150,160)
```

6. Display first name, salary, commission pct, and hire date for employees with salary less than 10000.

```
SELECT FIRST_NAME, SALARY, COMMISSION_PCT, HIRE_DATE FROM EMPLOYEES  
WHERE SALARY < 10000
```

7. Display job Title, the difference between minimum and maximum salaries for jobs with max salary in the range 10000 to 20000.

```
SELECT JOB_TITLE, MAX_SALARY-MIN_SALARY DIFFERENCE FROM JOBS WHERE  
MAX_SALARY BETWEEN 10000 AND 20000
```

8. Display first name, salary, and round the salary to thousands.

```
SELECT FIRST_NAME, SALARY, ROUND(SALARY, -3) FROM EMPLOYEES
```

9. Display details of jobs in the descending order of the title.

```
SELECT * FROM JOBS ORDER BY JOB_TITLE
```

10. Display employees where the first name or last name starts with S.

```
SELECT FIRST_NAME, LAST_NAME FROM EMPLOYEES WHERE FIRST_NAME LIKE 'S%'
OR LAST_NAME LIKE 'S%'
```

11. Display employees who joined in the month of May.

```
SELECT * FROM EMPLOYEES WHERE TO_CHAR(HIRE_DATE, 'MON') = 'MAY'
```

12. Display details of the employees where commission percentage is null and salary in the range 5000 to 10000 and department is 30.

```
SELECT * FROM EMPLOYEES WHERE COMMISSION_PCT IS NULL AND SALARY BETWEEN
5000 AND 10000 AND DEPARTMENT_ID=30
```

13. Display first name and date of first salary of the employees.

```
SELECT FIRST_NAME, HIRE_DATE, LAST_DAY(HIRE_DATE)+1 FROM EMPLOYEES
```

14. Display first name and experience of the employees.

```
SELECT FIRST_NAME, HIRE_DATE, FLOOR((SYSDATE-HIRE_DATE)/365) FROM
EMPLOYEES
```

15. Display first name of employees who joined in 2001.

```
SELECT FIRST_NAME, HIRE_DATE FROM EMPLOYEES WHERE TO_CHAR(HIRE_DATE,
'YYYY')=2001
```

16. Display first name and last name after converting the first letter of each name to upper case and the rest to lower case.

```
SELECT INITCAP(FIRST_NAME), INITCAP(LAST_NAME) FROM EMPLOYEES
```

17. Display the first word in job title.

```
SELECT JOB_TITLE, SUBSTR(JOB_TITLE,1, INSTR(JOB_TITLE, ' ')-1) FROM
JOBS
```

18. Display the length of first name for employees where last name contain character 'b' after 3rd position.

```
SELECT FIRST_NAME, LAST_NAME FROM EMPLOYEES WHERE INSTR(LAST_NAME, 'B') >
3
```

19. Display first name in upper case and email address in lower case for employees where the first name and email address are same irrespective of the case.

```
SELECT UPPER(FIRST_NAME), LOWER(EMAIL) FROM EMPLOYEES WHERE
UPPER(FIRST_NAME)=UPPER(EMAIL)
```

20. Display employees who joined in the current year.

```
SELECT * FROM EMPLOYEES WHERE TO_CHAR(HIRE_DATE, 'YYYY')=TO_CHAR(SYSDATE, 'YYYY')
```

21. Display the number of days between system date and 1st January 2011.

```
SELECT SYSDATE - to_date('01-jan-2011') FROM DUAL
```

22. Display how many employees joined in each month of the current year.

```
SELECT TO_CHAR(HIRE_DATE, 'MM'), COUNT (*) FROM EMPLOYEES  
WHERE TO_CHAR(HIRE_DATE, 'YYYY')= TO_CHAR(SYSDATE, 'YYYY') GROUP BY  
TO_CHAR(HIRE_DATE, 'MM')
```

23. Display manager ID and number of employees managed by the manager.

```
SELECT MANAGER_ID, COUNT(*) FROM EMPLOYEES GROUP BY MANAGER_ID
```

24. Display employee ID and the date on which he ended his previous job.

```
SELECT EMPLOYEE_ID, MAX(END_DATE) FROM JOB_HISTORY GROUP BY EMPLOYEE_ID
```

25. Display number of employees joined after 15th of the month.

```
SELECT COUNT(*) FROM EMPLOYEES WHERE TO_CHAR(HIRE_DATE, 'DD') > 15
```

26. Display the country ID and number of cities we have in the country.

```
SELECT COUNTRY_ID, COUNT(*) FROM LOCATIONS GROUP BY COUNTRY_ID
```

27. Display average salary of employees in each department who have commission percentage.

```
SELECT DEPARTMENT_ID, AVG(SALARY) FROM EMPLOYEES  
WHERE COMMISSION_PCT IS NOT NULL GROUP BY DEPARTMENT_ID
```

28. 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(*), SUM(SALARY), MAX(SALARY)-MIN(SALARY) SALARY  
FROM EMPLOYEES GROUP BY JOB_ID
```

29. Display job ID for jobs with average salary more than 10000.

```
SELECT JOB_ID, AVG(SALARY) FROM EMPLOYEES  
GROUP BY JOB_ID  
HAVING AVG(SALARY)>10000
```

30. Display years in which more than 10 employees joined.

```
SELECT TO_CHAR(HIRE_DATE, 'YYYY') FROM EMPLOYEES  
GROUP BY TO_CHAR(HIRE_DATE, 'YYYY')
```

```
HAVING COUNT(EMPLOYEE_ID) > 10
```

31. 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(COMMISSION_PCT) > 5
```

32. Display employee ID for employees who did more than one job in the past.

```
SELECT EMPLOYEE_ID FROM JOB_HISTORY GROUP BY EMPLOYEE_ID HAVING COUNT(*)  
> 1
```

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

```
SELECT JOB_ID FROM JOB_HISTORY  
WHERE END_DATE - START_DATE > 100  
GROUP BY JOB_ID  
HAVING COUNT(*) > 3
```

34. Display department ID, year, and Number of employees joined.

```
SELECT DEPARTMENT_ID, TO_CHAR(HIRE_DATE, 'YYYY'), COUNT(EMPLOYEE_ID)  
FROM EMPLOYEES  
GROUP BY DEPARTMENT_ID, TO_CHAR(HIRE_DATE, 'YYYY')  
ORDER BY DEPARTMENT_ID
```

35. Display departments where any manager is managing more than 5 employees.

```
SELECT DISTINCT DEPARTMENT_ID  
FROM EMPLOYEES  
GROUP BY DEPARTMENT_ID, MANAGER_ID  
HAVING COUNT(EMPLOYEE_ID) > 5
```

36. Change salary of employee 115 to 8000 if the existing salary is less than 6000.

```
UPDATE EMPLOYEES SET SALARY = 8000 WHERE EMPLOYEE_ID = 115 AND SALARY <  
6000
```

37. Insert a new employee into employees with all the required details.

```
INSERT INTO EMPLOYEES (EMPLOYEE_ID, FIRST_NAME, LAST_NAME, EMAIL,  
PHONE_NUMBER, HIRE_DATE, JOB_ID, SALARY, DEPARTMENT_ID)  
VALUES (207, 'ANGELA', 'SNYDER', 'ANGELA', '215 253 4737',  
  
SYSDATE, 'SA_MAN', 12000, 80)
```

38. Delete department 20.

```
DELETE FROM DEPARTMENTS WHERE DEPARTMENT_ID=20
```

39. Change job ID of employee 110 to IT_PROG if the employee belongs to department 10 and the existing job ID does not start with IT.

```
UPDATE EMPLOYEES SET JOB_ID= 'IT_PROG'  
WHERE EMPLOYEE_ID=110 AND DEPARTMENT_ID=10 AND NOT JOB_ID LIKE 'IT%'
```

40. Insert a row into departments table with manager ID 120 and location ID in any location ID for city Tokyo.

```
INSERT INTO DEPARTMENTS (150, 'SPORTS', 120, 1200)
```

41. Display department name and number of employees in the department.

```
SELECT DEPARTMENT_NAME, COUNT(*) FROM EMPLOYEES NATURAL JOIN DEPARTMENTS  
GROUP BY DEPARTMENT_NAME
```

42. Display job title, employee ID, number of days between ending date and starting date for all jobs in department 30 from job history.

```
SELECT EMPLOYEE_ID, JOB_TITLE, END_DATE-START_DATE DAYS  
FROM JOB_HISTORY NATURAL JOIN JOBS  
WHERE DEPARTMENT_ID=30
```

43. Display department name and manager first name.

```
SELECT DEPARTMENT_NAME, FIRST_NAME FROM DEPARTMENTS D JOIN EMPLOYEES E  
ON (D.MANAGER_ID=E.EMPLOYEE_ID)
```

44. Display department name, manager name, and city.

```
SELECT DEPARTMENT_NAME, FIRST_NAME, CITY FROM DEPARTMENTS D JOIN  
EMPLOYEES E ON (D.MANAGER_ID=E.EMPLOYEE_ID) JOIN LOCATIONS L USING  
(LOCATION_ID)
```

45. Display country name, city, and department name.

```
SELECT COUNTRY_NAME, CITY, DEPARTMENT_NAME  
FROM COUNTRIES JOIN LOCATIONS USING (COUNTRY_ID)  
JOIN DEPARTMENTS USING (LOCATION_ID)
```

46. Display job title, department name, employee last name, starting date for all jobs from 2000 to 2005.

```
SELECT JOB_TITLE, DEPARTMENT_NAME, LAST_NAME, START_DATE  
FROM JOB_HISTORY JOIN JOBS USING (JOB_ID) JOIN DEPARTMENTS  
USING (DEPARTMENT_ID) JOIN EMPLOYEES USING (EMPLOYEE_ID)  
WHERE TO_CHAR(START_DATE, 'YYYY') BETWEEN 2000 AND 2005
```

47. Display job title and average salary of employees

```
SELECT JOB_TITLE, AVG(SALARY) FROM EMPLOYEES
NATURAL JOIN JOBS GROUP BY JOB_TITLE
```

48. Display job title, employee name, and the difference between maximum salary for the job and salary of the employee.

```
SELECT JOB_TITLE, FIRST_NAME, MAX_SALARY-SALARY DIFFERENCE FROM
EMPLOYEES NATURAL JOIN JOBS
```

49. Display last name, job title of employees who have commission percentage and belongs to department 30.

```
SELECT JOB_TITLE, FIRST_NAME, MAX_SALARY-SALARY DIFFERENCE FROM
EMPLOYEES NATURAL JOIN JOBS WHERE DEPARTMENT_ID = 30
```

50. Display details of jobs that were done by any employee who is currently drawing more than 15000 of salary.

```
SELECT JH.*
FROM JOB_HISTORY JH JOIN EMPLOYEES E ON (JH.EMPLOYEE_ID =
E.EMPLOYEE_ID)
WHERE SALARY > 15000
```

51. Display department name, manager name, and salary of the manager for all managers whose experience is more than 5 years.

```
SELECT DEPARTMENT_NAME, FIRST_NAME, SALARY
FROM DEPARTMENTS D JOIN EMPLOYEES E ON (D.MANAGER_ID=E.MANAGER_ID)
WHERE (SYSDATE-HIRE_DATE) / 365 > 5
```

52. Display employee name if the employee joined before his manager.

```
SELECT FIRST_NAME FROM EMPLOYEES E1 JOIN EMPLOYEES E2 ON
(E1.MANAGER_ID=E2.EMPLOYEE_ID)
WHERE E1.HIRE_DATE < E2.HIRE_DATE
```

53. Display employee name, job title for the jobs employee did in the past where the job was done less than six months.

```
SELECT FIRST_NAME, JOB_TITLE FROM EMPLOYEES E JOIN JOB_HISTORY JH ON
(JH.EMPLOYEE_ID = E.EMPLOYEE_ID) JOIN JOBS J ON( JH.JOB_ID = J.JOB_ID)
WHERE MONTHS_BETWEEN(END_DATE,START_DATE) < 6
```

54. Display employee name and country in which he is working.

```
SELECT FIRST_NAME, COUNTRY_NAME FROM EMPLOYEES JOIN DEPARTMENTS
USING(DEPARTMENT_ID)
JOIN LOCATIONS USING( LOCATION_ID)
JOIN COUNTRIES USING ( COUNTRY_ID)
```

55. Display department name, average salary and number of employees with commission within the department.

```
SELECT DEPARTMENT_NAME, AVG(SALARY), COUNT(COMMISSION_PCT)
FROM DEPARTMENTS JOIN EMPLOYEES USING (DEPARTMENT_ID)
GROUP BY DEPARTMENT_NAME
```

56. Display the month in which more than 5 employees joined in any department located in Sydney.

```
SELECT TO_CHAR(HIRE_DATE, 'MON-YY')
FROM EMPLOYEES JOIN DEPARTMENTS USING (DEPARTMENT_ID) JOIN LOCATIONS
USING (LOCATION_ID)
WHERE CITY = 'Seattle'
GROUP BY TO_CHAR(HIRE_DATE, 'MON-YY')
HAVING COUNT(*) > 5
```

57. Display details of departments in which the maximum salary is more than 10000.

```
SELECT * FROM DEPARTMENTS WHERE DEPARTMENT_ID IN
( SELECT DEPARTMENT_ID FROM EMPLOYEES
GROUP BY DEPARTMENT_ID
HAVING MAX(SALARY)>10000)
```

58. Display details of departments managed by 'Smith'.

```
SELECT * FROM DEPARTMENTS WHERE MANAGER_ID IN
(SELECT EMPLOYEE_ID FROM EMPLOYEES WHERE FIRST_NAME='SMITH')
```

59. Display jobs into which employees joined in the current year.

```
SELECT * FROM JOBS WHERE JOB_ID IN
(SELECT JOB_ID FROM EMPLOYEES WHERE
TO_CHAR(HIRE_DATE, 'YYYY')=TO_CHAR(SYSDATE, 'YYYY'))
```

60. Display employees who did not do any job in the past.

```
SELECT * FROM EMPLOYEES WHERE EMPLOYEE_ID NOT IN
(SELECT EMPLOYEE_ID FROM JOB_HISTORY)
```

61. Display job title and average salary for employees who did a job in the past.

```
SELECT JOB_TITLE, AVG(SALARY) FROM JOBS NATURAL JOIN EMPLOYEES
GROUP BY JOB_TITLE
WHERE EMPLOYEE_ID IN
(SELECT EMPLOYEE_ID FROM JOB_HISTORY)
```

62. Display country name, city, and number of departments where department has more than 5 employees.

```
SELECT COUNTRY_NAME, CITY, COUNT(DEPARTMENT_ID)
FROM COUNTRIES JOIN LOCATIONS USING (COUNTRY_ID) JOIN DEPARTMENTS USING
(LOCATION_ID)
WHERE DEPARTMENT_ID IN
(SELECT DEPARTMENT_ID FROM EMPLOYEES
GROUP BY DEPARTMENT_ID
HAVING COUNT(*) > 5)
```

```

GROUP BY DEPARTMENT_ID
HAVING COUNT(DEPARTMENT_ID)>5)
GROUP BY COUNTRY_NAME, CITY;

```

63. Display details of manager who manages more than 5 employees.

```

SELECT FIRST_NAME FROM EMPLOYEES
WHERE EMPLOYEE_ID IN
(SELECT MANAGER_ID FROM EMPLOYEES
GROUP BY MANAGER_ID
HAVING COUNT(*)>5)

```

64. Display employee name, job title, start date, and end date of past jobs of all employees with commission percentage null.

```

SELECT FIRST_NAME, JOB_TITLE, START_DATE, END_DATE
FROM JOB_HISTORY JH JOIN JOBS J USING (JOB_ID) JOIN EMPLOYEES E ON
( JH.EMPLOYEE_ID = E.EMPLOYEE_ID)
WHERE COMMISSION_PCT IS NULL

```

65. Display the departments into which no employee joined in last two years.

```

SELECT * FROM DEPARTMENTS
WHERE DEPARTMENT_ID NOT IN
( SELECT DEPARTMENT_ID FROM EMPLOYEES WHERE FLOOR((SYSDATE-
HIRE_DATE)/365) < 2)

```

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

```

SELECT * FROM DEPARTMENTS
WHERE DEPARTMENT_ID IN
(SELECT DEPARTMENT_ID FROM EMPLOYEES
WHERE EMPLOYEE_ID IN (SELECT EMPLOYEE_ID FROM JOB_HISTORY)
GROUP BY DEPARTMENT_ID
HAVING MAX(SALARY) >10000)

```

67. Display details of current job for employees who worked as IT Programmers in the past.

```

SELECT * FROM JOBS
WHERE JOB_ID IN
(SELECT JOB_ID FROM EMPLOYEES WHERE EMPLOYEE_ID IN
(SELECT EMPLOYEE_ID FROM JOB_HISTORY WHERE JOB_ID='IT_PROG'))

```

68. Display the details of employees drawing the highest salary in the department.

```

SELECT DEPARTMENT_ID, FIRST_NAME, SALARY FROM EMPLOYEES OUTER WHERE
SALARY =
(SELECT MAX(SALARY) FROM EMPLOYEES WHERE DEPARTMENT_ID =
OUTER.DEPARTMENT_ID)

```


69. Display the city of employee whose employee ID is 105.

```
SELECT CITY FROM LOCATIONS WHERE LOCATION_ID =  
      (SELECT LOCATION_ID FROM DEPARTMENTS WHERE DEPARTMENT_ID =  
        (SELECT DEPARTMENT_ID FROM EMPLOYEES WHERE  
EMPLOYEE_ID=105)  
      )
```

70. Display third highest salary of all employees

```
select salary  
from employees main  
where 2 = (select count( distinct salary )  
          from employees  
          where salary > main.salary)
```