1. **Select Queries**
   1. Display all departments from department table.

Select \* FROM departments;

* 1. Display all employees from employee table.

SELECT \* FROM employees

* 1. Select the employee in department 30.

SELECT \* from employees

WHERE department\_id = 30;

* 1. List the names, numbers and departmentno of all clerks.

SELECT first\_name,phone\_number,department\_id

-> FROM employees

-> WHERE job\_id LIKE '%CLERK';

* 1. Find the depart numbers and the name of employee of all dept with Deptno greater or equal to 20.

SELECT department\_id,first\_name,last\_name

FROM employees

WHERE department\_id >= 20;

* 1. Find the employees whose commission is greater than their salary.
  2. Find the employees whose commission is greater than 60 percent of their salary.
  3. Find the employee whose commission is greater than 50 percent of their salary.
  4. List the name, job and salary of all employees in dept 20 who earn more than 2000.

SELECT first\_name,last\_name,job\_id,salary

FROM employees

WHERE department\_id= 20 AND salary>=2000

* 1. Find all salesmen in dept 30 whose salary is greater than or equal to Rs. 1500.
  2. Find all the employees whose job is either a president or manager.

SELECT \* FROM employees

WHERE job\_id like '%MGR' or '%PRES';

* 1. Find all managers who are not in dept 30.

SELECT \* from employees

WHERE job\_id like '%MGR' AND department\_id <> 30;

* 1. Find the details of all managers and clerks in dept 10.

SELECT \* FROM employees

WHERE job\_id like '%MGR' AND '%CLERK' AND department\_id= 10;

* 1. Find the details of all manager (in any dept) and all clerks in dept 10

SELECT \* FROM employees

WHERE job\_id like '%MGR'

OR job\_id like '%CLERK'

AND department\_id = 10;

* 1. Find the details of all managers in dept 10 and all clerks in dept 20.

SELECT \* FROM employees

WHERE job\_id like '%MGR'

AND department\_id=10

OR job\_id like '%CLERK'

AND department\_id = 20;

* 1. Find all employees who are neither clerks nor manager but whose salary is greater than or equal to Rs. 2000.

SELECT \* FROM employees

-> WHERE job\_id NOT like '%MGR' AND job\_id NOT like '%CLERK' AND salary >=2000;

* 1. Find the employees who earns between Rs. 1200 and Rs.1400.

SELECT \* FROM employees

-> WHERE salary BETWEEN 1200 AND 1400;

* 1. Find the employees who are clerks, analysts or salesman.

SELECT \* FROM employees

WHERE job\_id= like ‘%CLERK’;

* 1. Find the employees who are not clerks, analyst or salesman.

SELECT \* FROM employees

-> WHERE job\_id NOT like '%CLERK';

* 1. Find the employees who do not receive a commission i.e. commission is NULL.

SELECT \* FROM employees

-> WHERE commission\_pct IS NULL;

* 1. Find the employee whose commission is Rs. 0.

SELECT \* FROM employees

-> WHERE commission\_pct = 0;

* 1. Find the different jobs of the employees receiving commission.

SELECT job\_id

-> FROM employees

-> WHERE commission\_pct IS NOT NULL;

* 1. Find all employees who do not receive a commission or whose Commission is less than Rs. 1000.

SELECT commission\_pct FROM employees

WHERE commission\_pct IS NULL

AND commission\_pct\*salary <1000;

* 1. The employees who not receiving commission are entailed to Rs. 250, Show the net earnings of all employees. (find about **nvl() )**

SELECT first\_name, IF(commission\_pct IS NOT NULL,salary+(salary\*commission\_pct),salary+250)

-> AS net\_sal

-> FROM employees;

* 1. Find all employees whose total earnings are greater than Rs. 2000.

SELECT \* FROM employees

WHERE salary\*commission\_pct>2000;

* 1. Find all employees whose names begin with m.

SELECT first\_name FROM employees

-> WHERE first\_name like 'm%';

* 1. Find all employees whose names end with m.

SELECT first\_name FROM employees

-> WHERE first\_name like '%m';

* 1. Find all employees whose names contain the letter m.

SELECT first\_name FROM employees

-> WHERE first\_name like '%m%';

* 1. Find the employees whose names are 5 characters long and end with n.

SELECT first\_name FROM employees

-> WHERE length(first\_name)= 5 AND first\_name like '%n';

* 1. Find the employees who have the letter r as the third letter in their name.

SELECT first\_name FROM employees

-> WHERE first\_name like '\_\_r%';

1. **Numeric, Character & Date Function** 
   1. Find all employees hired in month of February (of any year).

SELECT \* FROM employees

-> WHERE month(hire\_date)=2;

* 1. Find all employees who were hired on the last day of the month.

SELECT first\_name

WHERE LAST\_DAY(hire\_date)=hire\_date

-> FROM employees;

* 1. Find the employees who were hired more than 12 years ago.

SELECT first\_name

-> FROM employees

-> WHERE datediff(now(),hire\_date)>12;

* 1. Find the managers hired in the year 2007.

SELECT first\_name

-> FROM employees

-> WHERE job\_id like '%MGR' AND YEAR(hire\_date)=2007;

* 1. Display the names and the jobs of all employees, separated by ','(comma). For example (smith, clerk).

SELECT CONCAT(first\_name,',',job\_id)

-> FROM employees;

* 1. Display the names of all employees with the initial letter only in capitals.

select left(first\_name,13)

-> FROM employees;

* 1. Display the names of all employees, right aligning them to 15 characters.
  2. Display the names of all employees, padding them to right up-to 15 characters with '-'.
  3. Display the length of the name of all employees.

SELECT first\_name,last\_name, LENGTH(first\_name)+ LENGTH(last\_name) ‘length of names’

FROM employees;

* 1. Display the names of all employees centering them with 20 characters.
  2. Display the names of all employees without any leading 'a'.

SELECT \* FROM employees

WHERE first\_name NOT like 'a%

* 1. Display the names of all employees without any trailing 'r'.

SELECT \* FROM employees

WHERE first\_name NOT like '%r’;

* 1. Show the first three characters of the names of all employees.

SELECT left(first\_name,3)

FROM employees;

* 1. Show the last three characters of the names of all employees.

SELECT right(first\_name,3)

FROM employees;

* 1. Display the names of all employees replacing any 'a' with 'e'.

SELECT REPLACE(first\_name,'a','e') FROM employees;

* 1. Display the names of all employees and the position at which the string 'ar' occurs in the name.

SELECT first\_name,POSITION('ar' IN first\_name)

-> FROM employees;

* 1. Show the salary of all employees rounding it to the nearest Rs. 1000. For example (3790 will be 4000)

SELECT first\_name,FLOOR((salary+999)/1000)\*1000 as totalSAL

FROM employees;

* 1. Show the daily salary of all employees assuming a month has 30 days.

> SELECT first\_name,salary , salary/30

-> FROM employees;

* 1. Display the name of all employees, and their bonus. Assume each Employee gets a bonus of 20 percent of his salary subject to the Maximum of Rs. 500.

SELECT first\_name, salary+(salary\*0.2) as bonus

-> FROM employees

-> WHERE salary\*0.2<=500;

* 1. Display the name of all employees, and their bonus. Assume each employee gets a bonus of 20 percent of his salary subject to the Maximum of Rs. 200.

SELECT first\_name, salary+(salary\*0.2) as bonus

-> FROM employees

-> WHERE salary\*0.2<=200;

* 1. For each employee display the number of days passed since the employee joined the company.

SELECT datediff(now(),hire\_date) as no\_of\_days

-> FROM employees;

* 1. For each employee display the number of months passed since the Employee joined the company.

SELECT TIMESTAMPDIFF(month,hire\_date,now())

-> FROM employees;

* 1. Display the tenure of service in the years, months and days for all Employees in character format. Assume every month has 30 days.

SELECT first\_name,TIMESTAMPDIFF(YEAR,hire\_date,now()) as years,TIMESTAMPDIFF(MONTH,hire\_date,now()) as months,TIMESTAMPDIFF(month,hire\_date,now())\*30 as dates

-> FROM employees;

* 1. Display the employee details in the following manner. 'Miler joined on the twenty-third of January of the year nineteen hundred and eighty Two'.

1. **Ordering by Queries**
   1. Display the details of all employees, sorted on the names.

SELECT \* from employees group by first\_name order by first\_name asc;

* 1. Display the name of all employees, based on their tenure, with the oldest employee coming first.

SELECT first\_name,TIMESTAMPDIFF(year,hire\_date,now()) as tenure

-> FROM employees

-> ORDER BY tenure DESC;

* 1. Display the names, job and salary of all employees sorted on jobs and Salary.

SELECT first\_name,job\_id,salary

-> FROM employees

-> ORDER BY job\_id,salary;

* 1. Display the names, job and salary of all employees, sorted on jobs and within job, sorted on the descending order of salary.

SELECT first\_name,job\_id,salary

-> FROM employees

-> GROUP BY job\_id

-> ORDER BY salary DESC;

* 1. Display the names, job and salary of all employees, sorted on Descending order of job and within job, sorted on the descending order of salary.

SELECT first\_name,job\_id,salary

-> FROM employees

-> ORDER BY job\_id DESC,salary DESC;

* 1. Display the name, month and year of all employees, sorted on the month of their hire date irrespective of the year.

SELECT first\_name,EXTRACT(YEAR\_MONTH FROM hire\_date) AS Month\_year

-> FROM employees

-> ORDER BY MONTH(hire\_date);

* 1. Display the name, month and year of joining of all employees, sorted on the month of their hire date, and within that on the year, with the earliest year appearing first.

SELECT first\_name,EXTRACT(YEAR\_MONTH FROM hire\_date) AS datea

-> FROM employees

-> ORDER BY YEAR(hire\_date) ASC, MONTH(hire\_date) ASC;