Practical - 02

Title: - Implementation of Analytical queries

Aim: - Implementation of Analytical queries like Roll_UP, CUBE, First, Last, Lead ,Lag,Rank AND Dense Rank.

Q.1. Create tables Employee and Department as per the given schema and insert data into them. dept(deptno number(2,0),dname varchar2(14),loc varchar2(13),constraint pk_dept primary key (deptno)); emp(empno number(4,0),ename varchar2(10), job varchar2(9), mgr number(4,0),hiredate date,sal number(7,2),comm number(7,2),deptno number(2,0), constraint pk_emp primary key (empno),constraint fk_deptno foreign key (deptno) references dept (deptno));

Execute the following queries.

Create a table employee with attribute empid, name, deptid, deptname, salary and joining date.

Write the queries -

1. To return the first salary reported in each department.
2. To show us how the average salary has changed over the years
3.To display the salary of each employee, along with the lowest and highest within their department
4. To divide the whole result set into five buckets based on salary
5. To display for each employee in Department 30 in the employees table, the hire date of the employee hired just after

SQL Query:

```
CREATE TABLE department (
  deptno NUMBER(2,0),
  dname VARCHAR2(14),
  loc VARCHAR2(13),
  CONSTRAINT pk_dept PRIMARY KEY (deptno)
);

CREATE TABLE employee (
  empno NUMBER(4,0),
  ename VARCHAR2(10),
  job VARCHAR2(9),
  mgr NUMBER(4,0),
  hiredate DATE,
  sal NUMBER(7,2),
  comm NUMBER(7,2),
  deptno NUMBER(2,0),
  CONSTRAINT pk_emp PRIMARY KEY (empno),
  CONSTRAINT fk_deptno FOREIGN KEY (deptno) REFERENCES department (deptno)
);
```

1. To return the first salary reported in each department.

SQL Query:

```
SELECT deptno, MIN(sal) AS first_salary
FROM employee GROUP BY deptno;
```

DEPTNO	FIRST_SALARY
12	10200
11	10300
10	10400
13	10000

2. To show us how the average salary has changed over the years

SQL Query:

SELECT hiredate, AVG(sal) AS average_salary
FROM employee GROUP BY hiredate ORDER BY hiredate ASC;

HIREDATE	AVERAGE_SALARY
01-FEB-23	10400
10-FEB-23	10300
13-FEB-23	10030
01-AUG-23	10200
10-SEP-23	10430
10-NOV-23	30000
10-DEC-23	10000
30-DEC-23	20000

SELECT hiredate, AVG(sal) AS average_salary
FROM employee GROUP BY hiredate ORDER BY hiredate DESC;

HIREDATE	AVERAGE_SALARY
30-DEC-23	20000
10-DEC-23	10000
10-NOV-23	30000
10-SEP-23	10430
01-AUG-23	10200
13-FEB-23	10030
10-FEB-23	10300
01-FEB-23	10400

3. To display the salary of each employee, along with the lowest and highest within their department

SQL Query:

SELECT empid, name, salary, MIN(salary) OVER (PARTITION BY deptid) AS lowest_salary, MAX(salary) OVER (PARTITION BY deptid) AS highest_salary FROM employee;

EMPNO	ENAME	SAL	LOWEST_SALARY	HIGHEST_SALARY
105	Suyash	10400	10400	10430
108	Amey	10430	10400	10430
104	Rahul	10300	10300	10300
102	Vishal	20000	10200	20000
107	Aniket	10200	10200	20000
101	Onkar	10000	10000	30000
106	Yash	10030	10000	30000
103	Aditya	30000	10000	30000

4. To divide the whole result set into five buckets based on salary

SQL Query:

```
SELECT empno, ename, sal,
  NTILE(5) OVER (ORDER BY sal) AS salary_bucket
FROM employee;
```

EMPNO	ENAME	SAL	SALARY_BUCKET
101	Onkar	10000	1
106	Yash	10030	1
107	Aniket	10200	2
104	Rahul	10300	2
105	Suyash	10400	3
108	Amey	10430	3
102	Vishal	20000	4
103	Aditya	30000	5

5. To display for each employee in Department 13 in the employees table, the hire date of the employee hired just after

SQL Query:

```
SELECT empno, ename, hiredate,
  LAG(hiredate) OVER (PARTITION BY deptno ORDER BY hiredate) AS next_hire_date
FROM employee WHERE deptno = 13;
```

EMPNO	ENAME	HIREDATE	NEXT_HIRE_DATE
106	Yash	13-FEB-23	-
103	Aditya	10-NOV-23	13-FEB-23
101	Onkar	10-DEC-23	10-NOV-23