**Assignment V**

Name :- Bibek Chand Sah

Roll :- 22054029

Branch :- CSE

Section :- CSE-05

1. Find the average salary of each department.

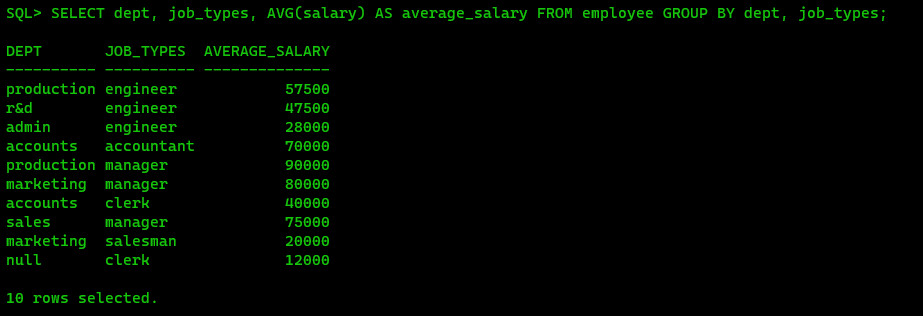
🡺 SELECT dept, AVG(salary) AS average\_salary FROM employee GROUP BY dept;

A screenshot of a computer screen

Description automatically generated

2. Find the average salary for each jobtype according to each department.

🡺 SELECT dept, job\_types, AVG(salary) AS average\_salary FROM employee GROUP BY dept, job\_types;



3. Find the department names and their corresponding average salary where the average

salary is greater than 40000.

🡺 SELECT dept, AVG(salary) AS average\_salary FROM employee GROUP BY dept HAVING AVG(salary) > 40000;

A screen shot of a black background

Description automatically generated

4. Select the departments where the maximum salary is more than 55000.

🡺 SELECT dept FROM employee GROUP BY dept HAVING MAX(salary) > 55000;

A black screen with green text

Description automatically generated

5. Find the department names and their average salary where the maximum salary of the

department is higher than 55000.

🡺 SELECT dept, AVG(salary) AS average\_salary FROM employee GROUP BY dept HAVING MAX(salary) > 55000;

A black screen with green text

Description automatically generated

6. Display the job\_types and the total monthly salary for each jobtypes as “PAYROLL”, where

the total payroll according to jobtypes exceeds 100000/month.

🡺 SELECT job\_types, SUM(salary) AS total\_salary FROM employee GROUP BY job\_types HAVING SUM(salary) > 100000;

A black screen with green text

Description automatically generated

7. Display the job\_types and the total monthly salary for each jobtypes as “PAYROLL”, where

the total payroll according to jobtypes exceeds 100000/month and jobtype is not engineer.

🡺 SELECT job\_types, SUM(salary) AS total\_salary FROM employee WHERE job\_types != 'engineer' GROUP BY job\_types HAVING SUM(salary) > 100000;

A black screen with green text

Description automatically generated

8. Display the job\_types and the total monthly salary for each jobtypes as “PAYROLL”, where

the total payroll according to jobtypes exceeds 60000/month and jobtype is not engineer

and sort the list in ascending order of sum of salary.

🡺 SELECT job\_types, SUM(salary) AS total\_salary FROM employee WHERE job\_types != 'engineer' GROUP BY job\_types HAVING SUM(salary) > 60000 ORDER BY SUM(salary) ASC;

A black screen with green text

Description automatically generated

9. Display the job\_types and the total monthly salary for each jobtypes as “PAYROLL”, where

the total payroll according to jobtypes exceeds 50000/month and jobtype is not engineer

and sort the list in descending order of sum of salary.

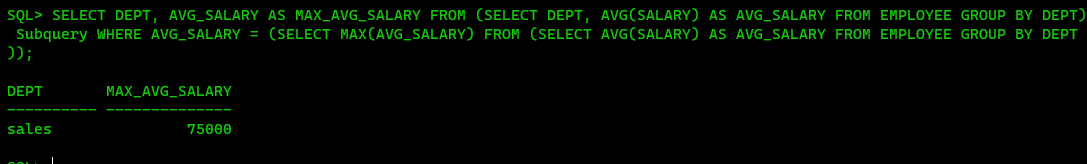
🡺 SELECT job\_types, SUM(salary) AS total\_salary FROM employee WHERE job\_types != 'engineer' GROUP BY job\_types HAVING SUM(salary) > 50000 ORDER BY SUM(salary) DESC;

A black screen with green text

Description automatically generated

10. Find the maximum average salary according to departments.

🡺 SELECT DEPT, AVG\_SALARY AS MAX\_AVG\_SALARY FROM (SELECT DEPT, AVG(SALARY) AS AVG\_SALARY FROM EMPLOYEE GROUP BY DEPT) Subquery WHERE AVG\_SALARY = (SELECT MAX(AVG\_SALARY) FROM (SELECT AVG(SALARY) AS AVG\_SALARY FROM EMPLOYEE GROUP BY DEPT ));



11. Find the minimum average salary according to jobtypes.

🡺 SELECT MIN(AVG(SALARY)) AS MIN\_AVG\_SALARY FROM EMPLOYEE GROUP BY JOB\_TYPES;

A black screen with green text

Description automatically generated

12. Find the employee name and date of joining who are working in delhi.

🡺 SELECT e.FIRST\_NAME || ' ' || e.LAST\_NAME AS EMPLOYEE\_NAME, e.DOJ FROM EMPLOYEE e JOIN DEPARTMENT d ON e.DEPT = d.D\_NAME WHERE d.D\_LOC = 'delhi';

A black screen with green text

Description automatically generated

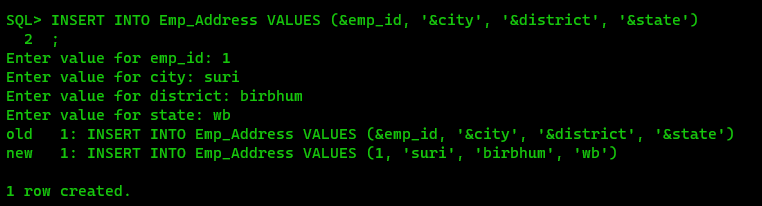
13. Create the table ‘Emp\_Address’ for storing the permanent address of the employees and

insert the values.



🡺 CREATE TABLE Emp\_Address (emp\_id INT, city VARCHAR(15), district VARCHAR(15), state VARCHAR(15));

INSERT INTO Emp\_Address VALUES (&emp\_id, '&city', '&district', '&state');



14. Display the employee name with their home city and the city they work in.

🡺 SELECT e.FIRST\_NAME || ' ' || e.LAST\_NAME AS Employee\_Name, a\_home.City AS Home\_City, d.D\_LOC AS Working\_City FROM EMPLOYEE e JOIN EMP\_ADDRESS a\_home ON e.EMP\_ID = a\_home.EMP\_ID JOIN DEPARTMENT d ON e.DEPT = d.D\_NAME;

15. Create the following Job\_Grades table.

A screenshot of a computer

Description automatically generated

🡺 CREATE TABLE JOB\_GRADES (GRADE VARCHAR2(5), LOWEST\_SAL NUMBER(10), HIGHEST\_SAL NUMBER(10));

INSERT INTO JOB\_GRADES VALUES('A',10000,24999);

INSERT INTO JOB\_GRADES VALUES('B',25000,49999);

INSERT INTO JOB\_GRADES VALUES('C',50000,100000);

A computer screen with green text

Description automatically generated

16. Display the employee names along with their salary and job\_grade.

🡺 SELECT e.FIRST\_NAME || ' ' || e.LAST\_NAME AS EMPLOYEE\_NAME, e.SALARY, j.GRADE FROM EMPLOYEE e JOIN JOB\_GRADES j ON e.SALARY BETWEEN j.LOWEST\_SAL AND j.HIGHEST\_SAL;

A computer screen with green text

Description automatically generated

17. Display the employees name along with their manager’s name. (use SELF JOIN)

🡺 SELECT e1.FIRST\_NAME AS Employee\_Name, e2.FIRST\_NAME AS Manager\_Name FROM EMPLOYEE e1 JOIN EMPLOYEE e2 ON e1.MANAGER\_ID = e2.EMPLOYEE\_ID;

A screen shot of a computer

Description automatically generated

18. Display emp\_id, f\_name, d\_loc, and hod\_id (using natural join).

🡺 SELECT EMPLOYEE.EMPLOYEE\_ID, EMPLOYEE.FIRST\_NAME, DEPARTMENT.D\_LOC, DEPARTMENT.HOD\_ID FROM EMPLOYEE NATURAL JOIN DEPARTMENT;

A screenshot of a computer

Description automatically generated

19. Display the employees f\_name, city and state in which they live (using natural join).

🡺 SELECT EMPLOYEE.FIRST\_NAME, EMP\_ADDRESS.CITY, EMP\_ADDRESS.STATE FROM EMPLOYEE NATURAL JOIN EMP\_ADDRESS;

A computer screen with green text

Description automatically generated

20. Display the employees emp\_id, f\_name, d\_loc, hod\_id using inner join.

🡺 SELECT EMPLOYEE.EMPLOYEE\_ID, EMPLOYEE.FIRST\_NAME, DEPARTMENT.D\_LOC, DEPARTMENT.HOD\_ID FROM EMPLOYEE INNER JOIN DEPARTMENT ON EMPLOYEE.DEPT = DEPARTMENT.D\_NAME;

A screen shot of a computer

Description automatically generated

21. Display the employees f\_name, city and state in which they live (using inner join).

🡺 SELECT EMPLOYEE.FIRST\_NAME, EMP\_ADDRESS.CITY, EMP\_ADDRESS.STATE FROM EMPLOYEE INNER JOIN EMP\_ADDRESS ON EMPLOYEE.EMP\_ID = EMP\_ADDRESS.EMP\_ID;

22. Display the employees f\_name, city and state in which they live (using join keyword).

🡺 SELECT EMPLOYEE.FIRST\_NAME, EMP\_ADDRESS.CITY, EMP\_ADDRESS.STATE FROM EMPLOYEE JOIN EMP\_ADDRESS ON EMPLOYEE.EMP\_ID = EMP\_ADDRESS.EMP\_ID;

23. Insert the following two rows in the employee table without inserting any value in the

department field.

A number of numbers and a few words

Description automatically generated with medium confidence

🡺 INSERT INTO EMPLOYEE VALUES(20,'ALEX',NULL,'ENGINEER',28000,2000,NULL,1,TO\_DATE('31-JAN-2017','DD-MM-YYYY'));

INSERT INTO EMPLOYEE VALUES(21,'PRIYA','PATEL','CLERK',12000,500,NULL,1,TO\_DATE('01-APR-2017','DD-MM-YYYY'));

A black screen with green text

Description automatically generated

24. Insert the following two rows into the department table.

A group of black text

Description automatically generated

🡺 INSERT INTO DEPARTMENT VALUES('TRAINING','MUMBAI',1);

INSERT INTO DEPARTMENT VALUES('PLACEMENT','MUMBAI',1);

A black background with green text

Description automatically generated

25. Display the employees f\_name, city and state in which they live after joining employee

and employee\_address table using left outer join.

🡺 SELECT EMPLOYEE.FIRST\_NAME, EMP\_ADDRESS.CITY, EMP\_ADDRESS.STATE FROM EMPLOYEE LEFT JOIN EMP\_ADDRESS ON EMPLOYEE.EMP\_ID = EMP\_ADDRESS.EMP\_ID;

26. Display the employees f\_name and their work location after joining employee and

department table using left join.

🡺 SELECT EMPLOYEE.FIRST\_NAME, DEPARTMENT.D\_LOC AS WORK\_LOCATION FROM EMPLOYEE LEFT JOIN DEPARTMENT ON EMPLOYEE.DEPT = DEPARTMENT.D\_NAME;

A computer screen with green text

Description automatically generated

27. Display the employees f\_name and their work location after joining employee and

department table using right join.

🡺 SELECT EMPLOYEE.FIRST\_NAME, DEPARTMENT.D\_LOC AS WORK\_LOCATION FROM EMPLOYEE RIGHT JOIN DEPARTMENT ON EMPLOYEE.DEPT = DEPARTMENT.D\_NAME;

A black screen with green text

Description automatically generated

28. Display the employees f\_name and their work location after joining employee and

department table using full join/full outer join.

🡺 SELECT EMPLOYEE.FIRST\_NAME, DEPARTMENT.D\_LOC AS WORK\_LOCATION FROM EMPLOYEE FULL JOIN DEPARTMENT ON EMPLOYEE.DEPT = DEPARTMENT.D\_NAME;

A black screen with green text

Description automatically generated

29. Find the employees who are working in their home city.

🡺 SELECT e.EMP\_ID, e.FIRST\_NAME, e.LAST\_NAME FROM EMPLOYEE e JOIN DEPARTMENT d ON e.DEPT = d.D\_NAME JOIN EMP\_ADDRESS ea ON e.EMP\_ID = ea.EMP\_ID WHERE d.D\_LOC = ea.CITY;

30. Find the job type having the minimum average salary according to jobtypes.

🡺 SELECT JOB\_TYPE, AVG(SALARY) AS AVERAGE\_SALARY FROM EMPLOYEE GROUP BY JOB\_TYPE ORDER BY AVG(SALARY) ASC FETCH FIRST 1 ROW ONLY;