# SQL LAB 3

NAME : YASH SHRIVASTAVA

**------------------SECTION : 1----------------**

1. Write a query in SQL to display the full name (first and last name), and salary for those employees who earn below 6000. Table: employees.

SELECT CONCAT(FIRST\_NAME, ' ', LAST\_NAME) EMPLOYEE\_NAME, SALARY FROM EMPLOYEES WHERE SALARY<6000;

1. Write a query in SQL to display the first and last\_name, department number and salary for those employees who earn more than 8000. Table: employees.

SELECT FIRST\_NAME, LAST\_NAME, DEPARTMENT\_ID, SALARY FROM EMPLOYEES WHERE SALARY>8000;

1. Write a query in SQL to display the first and last name, and department number for all employees whose last name is “McEwen”. Table: employees

SELECT FIRST\_NAME, LAST\_NAME, DEPARTMENT\_ID FROM EMPLOYEES WHERE LAST\_NAME = 'McEwen';

1. Write a query in SQL to display all the information for all employees without any department number. Table: employees.

SELECT \* FROM EMPLOYEES WHERE DEPARTMENT\_ID IS NULL;

1. Write a query in SQL to display all the information about the department Marketing. Table: departments

SELECT \* FROM DEPARTMENTS WHERE DEPARTMENT\_NAME ='Marketing';

1. Write a query in SQL to display the full name (first and last), hire date, salary, and department number for those employees whose first name does not containing the letter M and make the result set in ascending order by department number. Table: employees.

SELECT CONCAT(FIRST\_NAME,' ' ,LAST\_NAME) EMP\_NAME, HIRE\_DATE, SALARY, DEPARTMENT\_ID FROM EMPLOYEES WHERE FIRST\_NAME NOT SIMILAR TO '%M%' ORDER BY DEPARTMENT\_ID;

SELECT CONCAT(FIRST\_NAME,' ' ,LAST\_NAME) EMP\_NAME, HIRE\_DATE, SALARY, DEPARTMENT\_ID FROM EMPLOYEES WHERE FIRST\_NAME NOT LIKE '%M%' ORDER BY 4;

1. Write a query in SQL to display all the information of employees whose salary is in the range of 8000 and 12000 and commission is not null or department number is except the number 40, 120 and 70 and they have been hired before June 5th, 1987. Table: employees.

SELECT \* FROM EMPLOYEES WHERE SALARY BETWEEN 8000 AND 12000 AND COMMISSION\_PCT IS NOT NULL OR DEPARTMENT\_ID NOT IN (40,70,120) AND HIRE\_DATE<DATE '1987-06-05';

1. Write a query in SQL to display the full name (first and last name), and salary for all employees who does not earn any commission. Table: employees.

SELECT CONCAT(FIRST\_NAME,' ' ,LAST\_NAME) EMP\_NAME, SALARY FROM EMPLOYEES WHERE COMMISSION\_PCT=0.00 OR COMMISSION\_PCT IS NULL;

1. Write a query in SQL to display the full name (first and last), the phone number and email separated by hyphen, and salary, for those employees whose salary is within the range of 9000 and 17000. The column headings assign with Full\_Name, Contact\_Details and Remuneration respectively. Table: employees

SELECT CONCAT(FIRST\_NAME,' ' ,LAST\_NAME) Full\_Name, PHONE\_NUMBER|| '-' || EMAIL Contact\_Details, SALARY Remuneration FROM EMPLOYEES WHERE SALARY BETWEEN 9000 AND 17000;

1. Write a query in SQL to display the first and last name, and salary for those employees whose first name is ending with the letter m. Table: employees.

SELECT FIRST\_NAME, LAST\_NAME, SALARY FROM EMPLOYEES WHERE FIRST\_NAME LIKE '%m';

1. Write a query in SQL to display the full name (first and last) name, and salary, for all employees whose salary is out of the range 7000 and 15000 and make the result set in ascending order by the full name. Table: employees.

SELECT CONCAT(FIRST\_NAME,' ' ,LAST\_NAME) FULL\_NAME, SALARY FROM EMPLOYEES WHERE SALARY NOT BETWEEN 7000 AND 15000 ORDER BY FULL\_NAME;

1. Write a query in SQL to display the full name (first and last), job id and date of hire for those employees who was hired during November 5th, 2007 and July 5th, 2009. Table: employees.

SELECT CONCAT(FIRST\_NAME,' ' ,LAST\_NAME) FULL\_NAME, JOB\_ID, HIRE\_DATE FROM EMPLOYEES WHERE HIRE\_DATE BETWEEN '2007-11-05' AND '2009-06-05';

1. Write a query in SQL to display the the full name (first and last name), and department number for those employees who works either in department 70 or 90. Table: employees.

SELECT CONCAT(FIRST\_NAME,' ' ,LAST\_NAME) FULL\_NAME, DEPARTMENT\_ID FROM EMPLOYEES WHERE DEPARTMENT\_ID =70 OR DEPARTMENT\_ID=90;

SELECT CONCAT(FIRST\_NAME,' ' ,LAST\_NAME) FULL\_NAME, DEPARTMENT\_ID FROM EMPLOYEES WHERE DEPARTMENT\_ID IN (70,90);

1. Write a query in SQL to display the full name (first and last name), salary, and manager number for those employees who is working under a manager. Table: employees.

SELECT FIRST\_NAME || ' ' || LAST\_NAME FULL\_NAME, SALARY, MANAGER\_ID FROM EMPLOYEES WHERE MANAGER\_ID IS NOT NULL;

1. Write a query in SQL to display all the information from Employees table for those employees who was hired before June 21st, 2002. Table: employees

SELECT \* FROM EMPLOYEES WHERE HIRE\_DATE< DATE '2002-06-21';

SELECT \* FROM EMPLOYEES WHERE HIRE\_DATE<TO\_DATE('June 21st, 2002','FMMONTH DDST,YYYY');

1. Write a query in SQL to display the first and last name, email, salary and manager ID,for those employees whose managers are hold the ID 120, 103 or 145. Table: employees.

SELECT FIRST\_NAME, LAST\_NAME, EMAIL, SALARY, MANAGER\_ID FROM EMPLOYEES WHERE MANAGER\_ID IN (103,120,145) ORDER BY 5;

1. Write a query in SQL to display all the information from Employees table for those employees who was hired before June 21st, 2002. Table: employees

SELECT \* FROM EMPLOYEES WHERE HIRE\_DATE< DATE '2002-06-21';

1. Write a query in SQL to display the first and last name, email, salary and manager ID for those employees whose managers are hold the ID 120, 103 or 145. Table: employees.

SELECT FIRST\_NAME, LAST\_NAME, EMAIL, SALARY, MANAGER\_ID FROM EMPLOYEES WHERE MANAGER\_ID IN (103,120,145);

1. Write a query in SQL to display all the information for all employees who have the letters D, S, or N in their first name and also arrange the result in descending order bysalary. Table: employees

SELECT \* FROM EMPLOYEES WHERE FIRST\_NAME SIMILAR TO '%[DSN]%' ORDER BY SALARY DESC;

1. Write a query in SQL to display the full name (first name and last name), hire date, commission percentage, email and telephone separated by '-', and salary for those employees who earn the salary above 11000 or the seventh digit in their phone number equals 3 and make the result set in a descending order by the first name. Table: employees

SELECT FIRST\_NAME || ' '|| LAST\_NAME FULL\_NAME, HIRE\_DATE, COMMISSION\_PCT, EMAIL||'-'||PHONE\_NUMBER CONTACT, SALARY FROM EMPLOYEES WHERE SALARY>11000 OR SUBSTR(REPLACE(PHONE\_NUMBER,'.',''),7,1)='3' ORDER BY FIRST\_NAME DESC;

SELECT FIRST\_NAME || ' '|| LAST\_NAME FULL\_NAME, HIRE\_DATE, COMMISSION\_PCT, EMAIL||'-'||PHONE\_NUMBER CONTACT, SALARY FROM EMPLOYEES WHERE SALARY>11000 OR SUBSTR(,7,1)='3' ORDER BY FIRST\_NAME DESC;

1. Write a query in SQL to display the first and last name, and department number for those employees who holds a letter s as a 3rd character in their first name. Table : employees

SELECT FIRST\_NAME, LAST\_NAME, DEPARTMENT\_ID FROM EMPLOYEES WHERE FIRST\_NAME SIMILAR TO '\_\_s%';

1. Write a query in SQL to display the employee ID, first name, job id, and department number for those employees who is working except the departments 50,30 and 80.Table : employees

SELECT EMPLOYEE\_ID, FIRST\_NAME, JOB\_ID, DEPARTMENT\_ID FROM EMPLOYEES WHERE DEPARTMENT\_ID NOT IN (30,50,80);

1. Write a query in SQL to display the employee Id, first name, job id, and department number for those employees whose department number equals 30, 40 or 90.Table : employees

SELECT EMPLOYEE\_ID, FIRST\_NAME, JOB\_ID, DEPARTMENT\_ID FROM EMPLOYEES WHERE DEPARTMENT\_ID IN (30,40,90);

1. Write a query in SQL to display the ID for those employees who did two or more jobs in the past. Table : job\_history

SELECT EMPLOYEE\_ID FROM JOB\_HISTORY GROUP BY EMPLOYEE\_ID HAVING COUNT(\*)>=2;

1. Write a query in SQL to display job ID, number of employees, sum of salary, and difference between highest salary and lowest salary for a job. Table : employees

SELECT JOB\_ID, COUNT(EMPLOYEE\_ID) TOTAL\_EMPLOYEES, SUM(SALARY) TOTAL\_SALARY, MAX(SALARY)-MIN(SALARY) "HIGHEST-LOWEST" FROM EMPLOYEES GROUP BY JOB\_ID;

1. Write a query in SQL to display job ID for those jobs that were done by two or more for more than 300 days. Table : job\_history

SELECT JOB\_ID FROM JOB\_HISTORY WHERE END\_DATE-START\_DATE>'300 DAYS' GROUP BY JOB\_ID HAVING COUNT(\*)>=2;

1. Write a query in SQL to display the country ID and number of cities in that country we have. Table : loations

SELECT COUNTRY\_ID, COUNT(CITY) CITIES FROM LOCATIONS GROUP BY COUNTRY\_ID;

1. Write a query in SQL to display the manager ID and number of employees managed by the manager. Table : employees

SELECT MANAGER\_ID, COUNT(\*) FROM EMPLOYEES WHERE MANAGER\_ID IS NOT NULL GROUP BY MANAGER\_ID;

1. Write a query in SQL to display the details of jobs in descending sequence on job title. Table : jobs

SELECT \* FROM JOBS ORDER BY JOB\_TITLE DESC;

1. Write a query in SQL to display the first and last name and date of joining of the employees who is either Sales Representative or Sales Man. Table : employees, jobs

SELECT FIRST\_NAME, LAST\_NAME, HIRE\_DATE FROM EMPLOYEES WHERE JOB\_ID IN (SELECT JOB\_ID FROM JOBS WHERE JOB\_TITLE IN ('Sales Representative','Sales Man'));

1. Write a query in SQL to display the average salary of employees for each department who gets a commission percentage. Table : employees

SELECT DEPARTMENT\_ID, AVG(SALARY) FROM EMPLOYEES WHERE COMMISSION\_PCT IS NOT NULL GROUP BY DEPARTMENT\_ID HAVING DEPARTMENT\_ID IS NOT NULL;

1. Write a query in SQL to display those departments where any manager is managing 4 or more employees. Table : employees

SELECT DISTINCT DEPARTMENT\_ID FROM EMPLOYEES WHERE MANAGER\_ID IS NOT NULL GROUP BY DEPARTMENT\_ID, MANAGER\_ID HAVING COUNT(\*)>=4 AND DEPARTMENT\_ID IS NOT NULL;

1. Write a query in SQL to display those departments where more than ten employees work who got a commission percentage. Table : employees

SELECT DISTINCT DEPARTMENT\_ID FROM EMPLOYEES WHERE COMMISSION\_PCT IS NOT NULL GROUP BY DEPARTMENT\_ID HAVING COUNT(COMMISSION\_PCT)>10;

1. Write a query in SQL to display the employee ID and the date on which he ended his previous job. Table : job\_history

SELECT EMPLOYEE\_ID, MAX(END\_DATE) END\_DATE\_PREV\_JOB FROM JOB\_HISTORY GROUP BY EMPLOYEE\_ID ;

1. Write a query in SQL to display the details of the employees who have no commission percentage and salary within the range 7000 to 12000 and works in that department which number is

50.Table : employees

SELECT \* FROM EMPLOYEES WHERE COMMISSION\_PCT IS NULL AND SALARY BETWEEN 7000 AND 12000 AND DEPARTMENT\_ID =50;

1. Write a query in SQL to display the job ID for those jobs which average salary is above8000.Table : employees

SELECT JOB\_ID FROM EMPLOYEES GROUP BY JOB\_ID HAVING AVG(SALARY)>8000;

1. Write a query in SQL to display job Title, the difference between minimum and maximum salaries for those jobs which max salary within the range 12000 to 18000.Table : jobs

SELECT JOB\_TITLE, MAX\_SALARY-MIN\_SALARY "MAX-MIN\_SALARY" FROM JOBS WHERE MAX\_SALARY BETWEEN 12000 AND 18000;

1. Write a query in SQL to display all those employees whose first name or last name starts with the letter D.Table : employees

SELECT \* FROM EMPLOYEES WHERE FIRST\_NAME SIMILAR TO 'D%' OR LAST\_NAME SIMILAR TO 'D%';

1. Write a query in SQL to display the details of jobs which minimum salary is greater than 9000.Table : jobs

SELECT \* FROM JOBS WHERE MIN\_SALARY >9000;

1. Write a query in SQL to display those employees who joined after 7th September,1987.Table : employees

SELECT \* FROM EMPLOYEES WHERE HIRE\_DATE > '1987-09-07';

**------------------SECTION : 2----------------**

1. Write a query in SQL to display the first name, last name, department number, and department name for each employee. Table: departments, employees

--I use ‘LEFT JOIN’ because ‘Department\_Id’ column in ‘Employees’ table can be NULL.

SELECT E.FIRST\_NAME, E.LAST\_NAME, E.DEPARTMENT\_ID, D.DEPARTMENT\_NAME FROM EMPLOYEES E LEFT JOIN DEPARTMENTS D ON E.DEPARTMENT\_ID = D. DEPARTMENT\_ID;

1. Write a query in SQL to display the first and last name, department, city, and state province for each employee. Table:departments,employees,locations

--I use ‘LEFT JOIN’ because ‘Department\_Id’ column in ‘Employees’ table and ‘Location\_Id’ cloumn in ‘Departments’ table can be NULL.

SELECT E.FIRST\_NAME, E.LAST\_NAME, D.DEPARTMENT\_NAME, L.CITY, L.STATE\_PROVINCE FROM EMPLOYEES E LEFT JOIN DEPARTMENTS D ON E.DEPARTMENT\_ID = D.DEPARTMENT\_ID LEFT JOIN LOCATIONS L ON D.LOCATION\_ID = L.LOCATION\_ID;

1. Write a query in SQL to display the first name, last name, salary, and job grade for allemployees. Table:employees, job\_grades

-- job\_grades table is not present in the schema.

1. Write a query in SQL to display the first name, last name, department number and department name, for all employees for departments 80 or 40. Table: departments, employees

SELECT E.FIRST\_NAME, E.LAST\_NAME, E.DEPARTMENT\_ID, D.DEPARTMENT\_NAME FROM EMPLOYEES E JOIN DEPARTMENTS D ON E.DEPARTMENT\_ID = D.DEPARTMENT\_ID WHERE E.DEPARTMENT\_ID IN (40,80);

1. Write a query in SQL to display those employees who contain a letter z to their firstname and also display their last name, department, city, and state province. Table:departments, employees, locations

SELECT E.FIRST\_NAME, E.LAST\_NAME, D.DEPARTMENT\_NAME, L.CITY, L.STATE\_PROVINCE FROM (SELECT \* FROM EMPLOYEES WHERE FIRST\_NAME SIMILAR TO '%z%') E LEFT JOIN DEPARTMENTS D USING(DEPARTMENT\_ID) LEFT JOIN LOCATIONS L USING(LOCATION\_ID);

1. Write a query in SQL to display all departments including those where does not have any employee. Table: departments, employees

SELECT D.DEPARTMENT\_ID, D.DEPARTMENT\_NAME , E.FIRST\_NAME, E.LAST\_NAME FROM EMPLOYEES E RIGHT JOIN DEPARTMENTS D USING (DEPARTMENT\_ID);

1. Write a query in SQL to display the first and last name and salary for those employeeswho earn less than the employee earn whose number is 182. Table:employees

SELECT FIRST\_NAME, LAST\_NAME, SALARY FROM EMPLOYEES WHERE SALARY < (SELECT SALARY FROM EMPLOYEES WHERE EMPLOYEE\_ID =182);

1. Write a query in SQL to display the first name of all employees including the first name of their manager. Table:employees

SELECT FIRST\_NAME EMP\_NAME, (SELECT FIRST\_NAME FROM EMPLOYEES E1 WHERE E1.EMPLOYEE\_ID=E.MANAGER\_ID) MANAGER\_NAME FROM EMPLOYEES E WHERE MANAGER\_ID IS NOT NULL;

SELECT E.FIRST\_NAME EMPLOYEE\_NAME, M.FIRST\_NAME MANAGER\_NAME FROM EMPLOYEES E JOIN EMPLOYEES M ON E.MANAGER\_ID=M.EMPLOYEE\_ID;

1. Write a query in SQL to display the department name, city, and state province for each department. Table: departments, locations

SELECT D.DEPARTMENT\_NAME, L.CITY, L.STATE\_PROVINCE FROM DEPARTMENTS D LEFT JOIN LOCATIONS L ON D.LOCATION\_ID = L.LOCATION\_ID;

1. Write a query in SQL to display the first name, last name, department number andname, for all employees who have or have not any department. Table:departments, employees

SELECT E.FIRST\_NAME, E.LAST\_NAME, E.DEPARTMENT\_ID, D.DEPARTMENT\_NAME FROM EMPLOYEES E LEFT OUTER JOIN DEPARTMENTS D ON E.DEPARTMENT\_ID=D.DEPARTMENT\_ID;

1. Write a query in SQL to display the first name of all employees and the first name of their manager including those who does not working under any manager. Table: employees.

SELECT E.FIRST\_NAME EMPLOYEE\_NAME, M.FIRST\_NAME FROM EMPLOYEES E LEFT OUTER JOIN EMPLOYEES M ON E.MANAGER\_ID=M.EMPLOYEE\_ID;

12. Write a query in SQL to display the first name, last name, and department number forthose employees who works in the same department as the employee who holds the lastna

me as Taylor. Table:employees

SELECT FIRST\_NAME, LAST\_NAME, DEPARTMENT\_ID FROM EMPLOYEES WHERE DEPARTMENT\_ID IN (SELECT DEPARTMENT\_ID FROM EMPLOYEES WHERE LAST\_NAME='Taylor');

13.Write a query in SQL to display the job title, department name, full name (first and lastname ) of employee, and starting date for all the jobs which started on or after 1st January,1993 and ending with on or before 31 August, 1997. Table: job\_history, employees, jobs,DEPARTMENTS

SELECT J.JOB\_TITLE, D.DEPARTMENT\_NAME, E.FIRST\_NAME || ' ' || E.LAST\_NAME EMPLOYEE\_NAME, JH.START\_DATE FROM EMPLOYEES E JOIN JOB\_HISTORY JH ON E.EMPLOYEE\_ID =JH.EMPLOYEE\_ID JOIN JOBS J ON JH.JOB\_ID = J.JOB\_ID JOIN DEPARTMENTS D ON D.DEPARTMENT\_ID = JH.DEPARTMENT\_ID WHERE JH.START\_DATE >= DATE '1993-01-01' AND JH.END\_DATE <= DATE '1997-07-31';

1. Write a query in SQL to display job title, full name (first and last name ) of employee,and the difference between maximum salary for the job and salary of the employee. Table: employees, jobs

SELECT J.JOB\_TITLE, E.FIRST\_NAME || ' ' || E.LAST\_NAME EMPLOYEE\_ID, J.MAX\_SALARY-E.SALARY DIFF\_FROM\_MAX\_SALARY, E.SALARY FROM EMPLOYEES E JOIN JOBS J ON E.JOB\_ID=J.JOB\_ID;

1. Write a query in SQL to display the name of the department, average salary and number of employees working in that department who got commission. Table: employees, departments

SELECT D.DEPARTMENT\_NAME, AVG(E.SALARY) AVERAGE\_SALARY, COUNT(E.EMPLOYEE\_ID) TOTAL\_EMPLOYEES FROM EMPLOYEES E JOIN DEPARTMENTS D ON E.DEPARTMENT\_ID=D.DEPARTMENT\_ID WHERE E.COMMISSION\_PCT IS NOT NULL GROUP BY D.DEPARTMENT\_NAME;

1. Write a query in SQL to display the full name (first and last name ) of employee, and job title of those employees who is working in the department which ID is 80. Table: employees, jobs

SELECT E.FIRST\_NAME || ' ' || E.LAST\_NAME EMPLOYEE\_NAME, J.JOB\_TITLE FROM EMPLOYEES E JOIN JOBS J ON E.JOB\_ID=J.JOB\_ID WHERE E.DEPARTMENT\_ID =80;

SELECT E.FIRST\_NAME || ' ' || E.LAST\_NAME FULL\_NAME, J.JOB\_TITLE FROM (SELECT \* FROM EMPLOYEES WHERE DEPARTMENT\_ID=80) E JOIN JOBS J ON E.JOB\_ID=J.JOB\_ID;

1. Write a query in SQL to display the name of the country, city, and the departments which are running there. Table: countries, locations

SELECT C.COUNTRY\_NAME, L.CITY, D.DEPARTMENT\_NAME FROM COUNTRIES C INNER JOIN LOCATIONS L ON C.COUNTRY\_ID=L.COUNTRY\_ID INNER JOIN DEPARTMENTS D ON L.LOCATION\_ID = D.LOCATION\_ID;

SELECT country\_name, city, department\_name FROM countries NATURAL JOIN locations NATURAL JOIN departments;

1. Write a query in SQL to display department name and the full name (first and lastname) of the manager. Table:departments, employees

SELECT D.DEPARTMENT\_NAME, E.FIRST\_NAME || ' ' ||E.LAST\_NAME MANAGER\_NAME FROM DEPARTMENTS D JOIN EMPLOYEES E ON D.MANAGER\_ID=E.EMPLOYEE\_ID;

1. Write a query in SQL to display job title and average salary of employees. Table:employees, jobs

SELECT J.JOB\_TITLE, AVG(E.SALARY) AVERAGE\_SALARY FROM EMPLOYEES E NATURAL JOIN JOBS J GROUP BY J.JOB\_TITLE;

SELECT J.JOB\_TITLE, AVG(E.SALARY) AVERAGE\_SALARY FROM EMPLOYEES E JOIN JOBS J ON E.JOB\_ID=J.JOB\_ID GROUP BY J.JOB\_TITLE;

1. Write a query in SQL to display the details of jobs which was done by any of the employees who is presently earning a salary on and above 12000. Table: employees, job\_history

SELECT JH.\* FROM EMPLOYEES E JOIN JOB\_HISTORY JH ON E.EMPLOYEE\_ID = JH.EMPLOYEE\_ID WHERE E.SALARY>=12000;

1. Write a query in SQL to display the country name, city, and number of those departments where at least 2 employees are working. Table: countries, locations, employees, departments

SELECT C.COUNTRY\_NAME, L.CITY, COUNT(D.DEPARTMENT\_ID) FROM DEPARTMENTS D JOIN LOCATIONS L ON D.LOCATION\_ID=L.LOCATION\_ID JOIN COUNTRIES C ON L.COUNTRY\_ID=C.COUNTRY\_ID WHERE D.DEPARTMENT\_ID IN (SELECT DEPARTMENT\_ID FROM EMPLOYEES GROUP BY DEPARTMENT\_ID HAVING COUNT(\*)>=2) GROUP BY C.COUNTRY\_NAME, L.CITY ;

1. Write a query in SQL to display the department name, full name (first and last name) of manager, and their city. Table: employees, departments, locations

SELECT D.DEPARTMENT\_ID, E.FIRST\_NAME || ' ' || E.LAST\_NAME MANAGER\_NAME, L.CITY FROM DEPARTMENTS D LEFT JOIN LOCATIONS L ON D.LOCATION\_ID=L.LOCATION\_ID JOIN EMPLOYEES E ON D.MANAGER\_ID=E.EMPLOYEE\_ID;

1. Write a query in SQL to display the employee ID, job name, number of days worked in for all those jobs in department 80. Table: jobs, job\_history

SELECT JH.EMPLOYEE\_ID, J.JOB\_TITLE, END\_DATE-START\_DATE DAYS\_WORKED FROM JOB\_HISTORY JH NATURAL JOIN JOBS J WHERE JH.DEPARTMENT\_ID =80 ;

SELECT EMPLOYEE\_ID, (SELECT JOB\_TITLE FROM JOBS J WHERE J.JOB\_ID = JH.JOB\_ID), END\_DATE-START\_DATE WORKED FROM JOB\_HISTORY JH WHERE DEPARTMENT\_ID=80;

1. Write a query in SQL to display the full name (first and last name), and salary of those employees who working in any department located in London. Table: departments, locations, employees

SELECT E.FIRST\_NAME || ' ' || E.LAST\_NAME FULL\_NAME, E.SALARY FROM EMPLOYEES E JOIN DEPARTMENTS D ON E.DEPARTMENT\_ID=D.DEPARTMENT\_ID JOIN LOCATIONS L ON D.LOCATION\_ID =L.LOCATION\_ID WHERE L.CITY='London';

1. Write a query in SQL to display full name(first and last name), job title, starting and ending date of last jobs for those employees with worked without a commission percentage. Table: jobs, job\_history, employees

SELECT CONCAT(E.FIRST\_NAME,' ', E.LAST\_NAME) FULL\_NAME, J.JOB\_TITLE JOB\_NAME, JH.START\_DATE START\_ON, JH.END\_DATE ENDED FROM EMPLOYEES E JOIN (SELECT EMPLOYEE\_ID, MAX(START\_DATE) "start\_date", MAX(END\_DATE) "end\_date" FROM JOB\_HISTORY GROUP BY EMPLOYEE\_ID) JH ON E.EMPLOYEE\_ID =JH.EMPLOYEE\_ID AND E.COMMISSION\_PCT IS NULL JOIN JOBS J ON E.JOB\_ID = J.JOB\_ID ;

1. Write a query in SQL to display the department name and number of employees in each of the department. Table: departments, employees

SELECT D.DEPARTMENT\_NAME, COUNT(EMPLOYEE\_ID) FROM DEPARTMENTS D LEFT JOIN EMPLOYEES E ON D.DEPARTMENT\_ID = E.DEPARTMENT\_ID GROUP BY D.DEPARTMENT\_NAME;

1. Write a query in SQL to display the full name (firt and last name ) of employee with IDand name of the country presently where (s)he is working. Table: countries, locations, employees, departments

SELECT E.EMPLOYEE\_ID, E.FIRST\_NAME || ' ' || E.LAST\_NAME EMPLOYEE\_NAME, C.COUNTRY\_NAME FROM EMPLOYEES E JOIN DEPARTMENTS D ON E.DEPARTMENT\_ID=D.DEPARTMENT\_ID JOIN LOCATIONS L ON D.LOCATION\_ID=L.LOCATION\_ID JOIN COUNTRIES C ON L.COUNTRY\_ID=C.COUNTRY\_ID;

**------------------SECTION : 3----------------**

1. Write a query to display the name ( first name and last name ) for those employees who gets more salary than the employee whose ID is 163. Table: employees

SELECT FIRST\_NAME || ' ' || LAST\_NAME FULL\_NAME FROM EMPLOYEES WHERE SALARY >(SELECT SALARY FROM EMPLOYEES WHERE EMPLOYEE\_ID =163);

1. Write a query to display the name ( first name and last name ), salary, department id, jobid for those employees who works in the same designation as the employee works whoseid is 169. Table: employees

SELECT FIRST\_NAME || ' ' || LAST\_NAME FULL\_NAME, SALARY, DEPARTMENT\_ID, JOB\_ID FROM EMPLOYEES WHERE JOB\_ID = (SELECT JOB\_ID FROM EMPLOYEES WHERE EMPLOYEE\_ID=169);

1. Write a query to display the name ( first name and last name ), salary, department id for those employees who earn such amount of salary which is the smallest salary of any of thedepartments. Table: employees

SELECT CONCAT(FIRST\_NAME, ' ' , LAST\_NAME) EMPLOYEE\_NAME , SALARY, DEPARTMENT\_ID FROM EMPLOYEES WHERE SALARY =ANY(SELECT MIN(SALARY) FROM EMPLOYEES GROUP BY DEPARTMENT\_ID);

1. Write a query to display the employee id, employee name (first name and last name ) for all employees who earn more than the average salary. Table: employees

SELECT EMPLOYEE\_ID, CONCAT(FIRST\_NAME,' ' ,LAST\_NAME) EMPLOYEE\_NAME FROM EMPLOYEES WHERE SALARY> (SELECT AVG(SALARY) FROM EMPLOYEES);

1. Write a query to display the employee name ( first name and last name ), employee id and salary of all employees who report to Payam. Table: employees

SELECT EMPLOYEE\_ID, CONCAT(FIRST\_NAME, ' ' , LAST\_NAME) EMPLOYEE\_NAME, SALARY FROM EMPLOYEES WHERE MANAGER\_ID =ANY (SELECT EMPLOYEE\_ID FROM EMPLOYEES WHERE UPPER(FIRST\_NAME) ='PAYAM' OR UPPER(LAST\_NAME) ='PAYAM');

1. Write a query to display the department number, name ( first name and last name ), job and department name for all employees in the Finance department. Table: employees, departments

SELECT D.DEPARTMENT\_ID, D.DEPARTMENT\_NAME , CONCAT(E.FIRST\_NAME,' ',E.LAST\_NAME) EMPLOYEE\_NAME, E.JOB\_ID FROM EMPLOYEES E INNER JOIN DEPARTMENTS D ON E.DEPARTMENT\_ID = D.DEPARTMENT\_ID AND D.DEPARTMENT\_NAME ='Finance';

SELECT D.DEPARTMENT\_ID, D.DEPARTMENT\_NAME , CONCAT(E.FIRST\_NAME,' ',E.LAST\_NAME) EMPLOYEE\_NAME, E.JOB\_ID FROM EMPLOYEES E INNER JOIN DEPARTMENTS D ON E.DEPARTMENT\_ID = D.DEPARTMENT\_ID WHERE D.DEPARTMENT\_NAME='Finance';

1. Write a query to display all the information of an employee whose salary and reporting person id is 3000 and 121 respectively. Table : employees

SELECT \* FROM EMPLOYEES WHERE (SALARY,MANAGER\_ID)=(3000,121);

1. Display all the information of an employee whose id is any of the number 134, 159 and183. Table: employees

SELECT \* FROM EMPLOYEES WHERE EMPLOYEE\_ID IN (134,159,183);

1. Write a query to display all the information of the employees whose salary is within the range 1000 and 3000. Table: employees

SELECT \* FROM EMPLOYEES WHERE SALARY BETWEEN 1000 AND 3000;

1. Write a query to display all the information of the employees whose salary is within the range of smallest salary and 2500. Table: employees

SELECT \* FROM EMPLOYEES WHERE SALARY BETWEEN (SELECT MIN(SALARY) FROM EMPLOYEES) AND 2500;

1. Write a query to display all the information of the employees who does not work in those departments where some employees works whose id within the range 100 and 200. Table: employees, departments.

SELECT \* FROM EMPLOYEES WHERE DEPARTMENT\_ID IN (SELECT DEPARTMENT\_ID FROM EMPLOYEES WHERE EMPLOYEE\_ID NOT BETWEEN 100 AND 200);

1. Write a query to display all the information for those employees whose id is any id who earn the second highest salary. Table: employees

SELECT \* FROM EMPLOYEES WHERE SALARY =(SELECT DISTINCT SALARY FROM EMPLOYEES ORDER BY SALARY DESC OFFSET 1 ROW LIMIT 1);

1. Write a query to display the employee name( first name and last name ) and hiredatefor all employees in the same department as Clara. Exclude Clara. Table: employees

-- Using ‘IN’

SELECT FIRST\_NAME, LAST\_NAME, HIRE\_DATE FROM EMPLOYEES WHERE DEPARTMENT\_ID IN (SELECT DEPARTMENT\_ID FROM EMPLOYEES WHERE FIRST\_NAME SIMILAR TO 'Clara') AND FIRST\_NAME <> 'Clara';

-- Using ‘ANY’

SELECT FIRST\_NAME, LAST\_NAME, HIRE\_DATE FROM EMPLOYEES WHERE DEPARTMENT\_ID =ANY (SELECT DEPARTMENT\_ID FROM EMPLOYEES WHERE FIRST\_NAME SIMILAR TO 'Clara') AND FIRST\_NAME<> 'Clara';

1. Write a query to display the employee number and name( first name and last name )for all employees who work in a department with any employee whose name contains a T. Table: employees

-- Using ‘ANY’

SELECT EMPLOYEE\_ID, FIRST\_NAME, LAST\_NAME FROM EMPLOYEES WHERE DEPARTMENT\_ID =ANY (SELECT DEPARTMENT\_ID FROM EMPLOYEES WHERE FIRST\_NAME||LAST\_NAME SIMILAR TO '%T%');

-- Using ‘IN’

SELECT EMPLOYEE\_ID, FIRST\_NAME, LAST\_NAME FROM EMPLOYEES WHERE DEPARTMENT\_ID IN (SELECT DEPARTMENT\_ID FROM EMPLOYEES WHERE FIRST\_NAME||LAST\_NAME SIMILAR TO '%T%');

1. Write a query to display the employee number, name( first name and last name ), andsalary for all employees who earn more than the average salary and who work in a department with any employee with a J in their name. Table: employees

-- Using ‘IN’

SELECT EMPLOYEE\_ID, FIRST\_NAME, LAST\_NAME, SALARY FROM EMPLOYEES WHERE SALARY >(SELECT AVG(SALARY) FROM EMPLOYEES) AND DEPARTMENT\_ID IN (SELECT DEPARTMENT\_ID FROM EMPLOYEES WHERE FIRST\_NAME||LAST\_NAME SIMILAR TO '%J%');

-- Using ‘ANY’

SELECT EMPLOYEE\_ID, FIRST\_NAME, LAST\_NAME, SALARY FROM EMPLOYEES WHERE SALARY >(SELECT AVG(SALARY) FROM EMPLOYEES) AND DEPARTMENT\_ID = ANY (SELECT DEPARTMENT\_ID FROM EMPLOYEES WHERE FIRST\_NAME||LAST\_NAME SIMILAR TO '%J%');

1. Display the employee name( first name and last name ), employee id, and job title for all employees whose department location is Toronto. Table: employees, departments.

SELECT E.FIRST\_NAME || ' ' || E.LAST\_NAME EMPLOYEE\_NAME, E.EMPLOYEE\_ID,(SELECT JOB\_TITLE FROM JOBS J WHERE J.JOB\_ID=E.JOB\_ID) FROM EMPLOYEES E JOIN DEPARTMENTS D ON E.DEPARTMENT\_ID = D.DEPARTMENT\_ID JOIN LOCATIONS L ON D.LOCATION\_ID=L.LOCATION\_ID WHERE L.CITY= 'Toronto';

1. Write a query to display the employee number, name( first name and last name ) and job title for all employees whose salary is smaller than any salary of those employees whose job title is MK\_MAN. Table: employees, jobs.

SELECT EMPLOYEE\_ID, CONCAT(FIRST\_NAME, ' ', LAST\_NAME) EMPLOYEE\_NAME, (SELECT JOB\_TITLE FROM JOBS J WHERE J.JOB\_ID=E.JOB\_ID ) FROM EMPLOYEES E WHERE SALARY <ANY (SELECT SALARY FROM EMPLOYEES WHERE JOB\_ID='MK\_MAN');

1. Write a query to display the employee number, name( first name and last name ) and job title for all employees whose salary is smaller than any salary of those employees whose job ID is MK\_MAN. Exclude Job title MK\_MAN. Table: employees, jobs.

SELECT EMPLOYEE\_ID, CONCAT(FIRST\_NAME, ' ',LAST\_NAME) EMPLOYEE\_NAME, (SELECT JOB\_TITLE FROM JOBS J WHERE J.JOB\_ID=E.JOB\_ID ) FROM EMPLOYEES E WHERE SALARY <ANY(SELECT SALARY FROM EMPLOYEES WHERE JOB\_ID='MK\_MAN');

1. Write a query to display the employee number, name( first name and last name ) and job title for all employees whose salary is more than any salary of those employees whose job title is PU\_MAN. Exclude job title PU\_MAN. Table: employees, jobs.

SELECT EMPLOYEE\_ID, CONCAT(FIRST\_NAME,' ',LAST\_NAME) EMPLOYEE\_NAME, (SELECT JOB\_TITLE FROM JOBS J WHERE J.JOB\_ID=E.JOB\_ID ) FROM EMPLOYEES E WHERE SALARY > ANY (SELECT SALARY FROM EMPLOYEES WHERE JOB\_ID='PU\_MAN');

1. Write a query to display the employee number, name( first name and last name ) and job title for all employees whose salary is more than any average salary of any department. Table: employees, jobs.

SELECT EMPLOYEE\_ID, CONCAT(FIRST\_NAME,' ',LAST\_NAME) EMPLOYEE\_NAME, (SELECT JOB\_TITLE FROM JOBS J WHERE J.JOB\_ID=E.JOB\_ID ) FROM EMPLOYEES E WHERE SALARY > ANY(SELECT AVG(SALARY) FROM EMPLOYEES GROUP BY DEPARTMENT\_ID);

1. Write a query to display the employee name( first name and last name ) and department for all employees for any existence of those employees whose salary is more than 3700. Table: employees

SELECT FIRST\_NAME, LAST\_NAME, DEPARTMENT\_ID FROM EMPLOYEES E WHERE EXISTS (SELECT \* FROM EMPLOYEES WHERE SALARY>3700 AND DEPARTMENT\_ID=E.DEPARTMENT\_ID);

1. Write a query to display the department id and the total salary for those departments which contains at least one salaried employee. Table: employees, departments

SELECT D.DEPARTMENT\_ID, E.TOTAL\_AMT FROM DEPARTMENTS D JOIN (SELECT DEPARTMENT\_ID, SUM(SALARY) TOTAL\_AMT FROM EMPLOYEES GROUP BY DEPARTMENT\_ID) E ON D.DEPARTMENT\_ID =E.DEPARTMENT\_ID;

1. Write a query to display the employee id, name ( first name and last name ) and the job\_id column with a modified title SALESMAN for those employees whose job title is ST\_MANand DEVELOPER for whose job title is IT\_PROG. Table: employees

SELECT EMPLOYEE\_ID, CONCAT(FIRST\_NAME,' ',LAST\_NAME) EMPLOYEE\_NAME, CASE WHEN JOB\_ID='ST\_MAN' THEN 'SALESMAN' WHEN JOB\_ID='IT\_PROG' THEN 'DEVELOPER' ELSE JOB\_ID END JOB FROM EMPLOYEES;

1. Write a query to display the employee id, name ( first name and last name ), salary and the SalaryStatus column with a title HIGH and LOW respectively for those employeeswhose salary is more than and less than the average salary of all employees. Table: employees

SELECT EMPLOYEE\_ID, FIRST\_NAME, LAST\_NAME, SALARY , CASE WHEN SALARY >(SELECT AVG(SALARY) FROM EMPLOYEES ) THEN 'HIGH' ELSE 'LOW' END "SalaryStatus" FROM EMPLOYEES;

1. Write a query to display the employee id, name ( first name and last name ),SalaryDrawn, AvgCompare (salary - the average salary of all employees) and theSalaryStatus column with a title HIGH and LOW respectively for those employees whosesalary is more than and less than the average salary of all employees. Table: employees

SELECT EMPLOYEE\_ID, CONCAT(FIRST\_NAME,' ',LAST\_NAME) EMPLOYEE\_NAME, SALARY, SALARY-(SELECT AVG(SALARY) FROM EMPLOYEES ) "AvgCompare", CASE WHEN SALARY >(SELECT AVG(SALARY) FROM EMPLOYEES ) THEN 'HIGH' ELSE 'LOW' END "SalaryStatus" FROM EMPLOYEES;

1. Write a subquery that returns a set of rows to find all departments that do actually have one or more employees assigned to them. Table: employees, departments

Subquery : SELECT DISTINCT DEPARTMENT\_ID FROM EMPLOYEES;

SELECT \* FROM DEPARTMENTS WHERE DEPARTMENT\_ID IN (SELECT DEPARTMENT\_ID FROM EMPLOYEES);

SELECT \* FROM DEPARTMENTS D WHERE EXISTS (SELECT 'X' FROM EMPLOYEES E WHERE E.DEPARTMENT\_ID=D.DEPARTMENT\_ID);

1. Write a query that will identify all employees who work in departments located in the United Kingdom. Table: employees, departments, locations, countries

SELECT \* FROM EMPLOYEES WHERE DEPARTMENT\_ID IN (SELECT DEPARTMENT\_ID FROM DEPARTMENTS WHERE LOCATION\_ID IN (SELECT LOCATION\_ID FROM LOCATIONS WHERE COUNTRY\_ID = (SELECT COUNTRY\_ID FROM COUNTRIES WHERE COUNTRY\_NAME='United Kingdom')));

1. Write a query to identify all the employees who earn more than the average and who work in any of the IT departments. Table: employees, departments

SELECT \* FROM EMPLOYEES WHERE SALARY >(SELECT AVG(SALARY) FROM EMPLOYEES) AND DEPARTMENT\_ID IN (SELECT DEPARTMENT\_ID FROM DEPARTMENTS WHERE DEPARTMENT\_NAME='IT' OR DEPARTMENT\_NAME SIMILAR TO 'IT %');

1. Write a query to determine who earns more than Mr. Ozer. Table: employees

--I use ‘ALL’ because it is possible to have more than a person name is Mr. Ozer

SELECT \* FROM EMPLOYEES WHERE SALARY>(SELECT MAX(SALARY) FROM EMPLOYEES WHERE LAST\_NAME='Ozer');

1. Write a query to find out which employees have a manager who works for a department based in the US. Table: employees, departments, locations

SELECT \* FROM EMPLOYEES WHERE MANAGER\_ID IN (SELECT EMPLOYEE\_ID FROM EMPLOYEES WHERE DEPARTMENT\_ID IN (SELECT DEPARTMENT\_ID FROM DEPARTMENTS WHERE LOCATION\_ID IN (SELECT LOCATION\_ID FROM LOCATIONS WHERE COUNTRY\_ID ='US')));

1. Write a query which is looking for the names of all employees whose salary is greater than 50% of their department’s total salary bill. Table : employees

select first\_name, last\_name from employees e where salary >(select sum(salary)\*0.50 from employees d where d.department\_id=e.department\_id group by department\_id);

1. Write a query to get the details of employees who are managers. Table: employees, departments

SELECT \* FROM EMPLOYEES E WHERE EXISTS (SELECT 'X' FROM EMPLOYEES M WHERE E.EMPLOYEE\_ID = M.MANAGER\_ID);

1. Write a query to get the details of employees who manage a department. Table: employees, departments

SELECT \* FROM EMPLOYEES WHERE EMPLOYEE\_ID IN (SELECT MANAGER\_ID FROM DEPARTMENTS);

SELECT \* FROM EMPLOYEES E WHERE EXISTS (SELECT 'X' FROM DEPARTMENTS D WHERE E.EMPLOYEE\_ID = D.MANAGER\_ID);

1. Write a query to display the employee id, name ( first name and last name ), salary, department name and city for all the employees who gets the salary as the salary earn bythe employee which is maximum within the joining person January 1st, 2002 and December 31st, 2003. Table: employees, departments, locations

SELECT E.EMPLOYEE\_ID EMP\_ID, E.FIRST\_NAME || ' ' || E.LAST\_NAME EMPLOYEE\_NAME, E.SALARY EMP\_SAL, D.DEPARTMENT\_NAME DEPT\_NAME, L.CITY EMP\_CITY FROM EMPLOYEES E INNER JOIN DEPARTMENTS D ON E.DEPARTMENT\_ID = D.DEPARTMENT\_ID JOIN LOCATIONS L ON D.LOCATION\_ID=L.LOCATION\_ID WHERE E.SALARY=(SELECT MAX(SALARY) FROM EMPLOYEES WHERE HIRE\_DATE BETWEEN '2002-01-01' AND '2003-12-31');

1. Write a query in SQL to display the department code and name for all departments which located in the city London. Table: departments, locations

SELECT D.DEPARTMENT\_ID, D.DEPARTMENT\_NAME FROM DEPARTMENTS D JOIN LOCATIONS L USING(LOCATION\_ID) WHERE L.CITY='London';

SELECT DEPARTMENT\_ID, DEPARTMENT\_NAME FROM DEPARTMENTS WHERE LOCATION\_ID = (SELECT LOCATION\_ID FROM LOCATIONS WHERE CITY='London');

SELECT DEPARTMENT\_ID, DEPARTMENT\_NAME FROM DEPARTMENTS D WHERE EXISTS (SELECT 'X' FROM LOCATIONS L WHERE L.LOCATION\_ID=D.LOCATION\_ID AND L.CITY='London');

1. Write a query in SQL to display the first and last name, salary, and department ID for all those employees who earn more than the average salary and arrange the list in descending order on salary. Table: employees

SELECT FIRST\_NAME, LAST\_NAME, SALARY, DEPARTMENT\_ID FROM EMPLOYEES WHERE SALARY >(SELECT AVG(SALARY) FROM EMPLOYEES) ORDER BY SALARY DESC;

1. Write a query in SQL to display the first and last name, salary, and department ID for those employees who earn more than the maximum salary of a department which ID is40. Table: employees

SELECT FIRST\_NAME, LAST\_NAME, SALARY, DEPARTMENT\_ID FROM EMPLOYEES WHERE SALARY >(SELECT MAX(SALARY) FROM EMPLOYEES WHERE DEPARTMENT\_ID=40);

1. Write a query in SQL to display the department name and Id for all departments where they located, that Id is equal to the Id for the location where department number 30 is located. Table: departments

SELECT DEPARTMENT\_ID, DEPARTMENT\_NAME FROM DEPARTMENTS WHERE LOCATION\_ID =(SELECT LOCATION\_ID FROM DEPARTMENTS WHERE DEPARTMENT\_ID=30);

1. Write a query in SQL to display the first and last name, salary, and department ID for all those employees who work in that department where the employee works who hold the ID 201. Table: employees

SELECT FIRST\_NAME, LAST\_NAME, SALARY, DEPARTMENT\_ID FROM EMPLOYEES WHERE DEPARTMENT\_ID =(SELECT DEPARTMENT\_ID FROM EMPLOYEES WHERE EMPLOYEE\_ID =201);

1. Write a query in SQL to display the first and last name, salary, and department ID for those employees whose salary is equal to the salary of the employee who works in that department which ID is 40. Table: employees

SELECT FIRST\_NAME, LAST\_NAME, SALARY, DEPARTMENT\_ID FROM EMPLOYEES WHERE SALARY = ANY (SELECT SALARY FROM EMPLOYEES WHERE DEPARTMENT\_ID =40);

1. Write a query in SQL to display the first and last name, and department code for allemployees who work in the department Marketing. Table: employees, departments

SELECT FIRST\_NAME, LAST\_NAME, DEPARTMENT\_ID FROM EMPLOYEES WHERE DEPARTMENT\_ID = (SELECT DEPARTMENT\_ID FROM DEPARTMENTS WHERE DEPARTMENT\_NAME ='Marketing');

42.Write a query in SQL to display the first and last name, salary, and department ID forthose employees who earn more than the minimum salary of a department which ID is40. Table:

Employees

SELECT FIRST\_NAME, LAST\_NAME SALARY, DEPARTMENT\_ID FROM EMPLOYEES WHERE SALARY >(SELECT MIN(SALARY) FROM EMPLOYEES WHERE DEPARTMENT\_ID=40);

43. Write a query in SQL to display the full name,email, and designation for all those employees who was hired after the employee whose ID is 165. Table: employees

SELECT CONCAT(FIRST\_NAME,' ',LAST\_NAME) FULL\_NAME, EMAIL, JOB\_ID DESIGNATION FROM EMPLOYEES WHERE HIRE\_DATE> (SELECT HIRE\_DATE FROM EMPLOYEES WHERE EMPLOYEE\_ID =165);

44. Write a query in SQL to display the first and last name, salary, and department ID forthose employees who earn less than the minimum salary of a department which ID is 70. Table:

Employees

SELECT FIRST\_NAME, LAST\_NAME, SALARY, DEPARTMENT\_ID FROM EMPLOYEES WHERE SALARY <(SELECT MIN(SALARY) FROM EMPLOYEES WHERE DEPARTMENT\_ID=70);

1. Write a query in SQL to display the first and last name, salary, and department ID forthose employees who earn less than the average salary, and also work at the departmentwhere the employee Laura is working as a first name holder. Table: employees

-- I use ‘IN’ clause because it is possible then there is more than one person having name ‘Laura’

SELECT FIRST\_NAME, LAST\_NAME, SALARY, DEPARTMENT\_ID FROM EMPLOYEES WHERE SALARY <(SELECT AVG(SALARY) FROM EMPLOYEES ) AND DEPARTMENT\_ID IN (SELECT DEPARTMENT\_ID FROM EMPLOYEES WHERE FIRST\_NAME='Laura');

46. Write a query in SQL to display the first and last name, salary and department ID forthose employees whose department is located in the city London. Table: employees

SELECT FIRST\_NAME, LAST\_NAME, SALARY, DEPARTMENT\_ID FROM EMPLOYEES WHERE DEPARTMENT\_ID IN (SELECT DEPARTMENT\_ID FROM DEPARTMENTS WHERE LOCATION\_ID =(SELECT LOCATION\_ID FROM LOCATIONS WHERE CITY='London') );

47. Write a query in SQL to display the city of the employee whose ID 134 and works there. Table: locations, departments, employees

SELECT CITY FROM LOCATIONS WHERE LOCATION\_ID=(SELECT LOCATION\_ID FROM DEPARTMENTS WHERE DEPARTMENT\_ID=(SELECT DEPARTMENT\_ID FROM EMPLOYEES WHERE EMPLOYEE\_ID=134));

48. Write a query in SQL to display the the details of those departments which max salaryis 7000 or above for those employees who already done one or more jobs. Table: departments, employees, job\_history

SELECT \* FROM DEPARTMENTS WHERE DEPARTMENT\_ID IN (SELECT DEPARTMENT\_ID FROM EMPLOYEES WHERE EMPLOYEE\_ID IN (SELECT EMPLOYEE\_ID FROM JOB\_HISTORY GROUP BY EMPLOYEE\_ID HAVING COUNT(\*)>=1) GROUP BY DEPARTMENT\_ID HAVING MAX(SALARY)>=7000);

49. Write a query in SQL to display the detail information of those departments which starting salary is at least 8000. Table: departments, employees

SELECT \* FROM DEPARTMENTS WHERE DEPARTMENT\_ID IN (SELECT DEPARTMENT\_ID FROM EMPLOYEES GROUP BY DEPARTMENT\_ID HAVING MIN(SALARY) >=8000);

50.Write a query in SQL to display the full name (first and last name) of manager who is supervising 4 or more employees. Table : employees

SELECT CONCAT(FIRST\_NAME, ' ', LAST\_NAME) MANAGER FROM EMPLOYEES WHERE EMPLOYEE\_ID IN (SELECT MANAGER\_ID FROM EMPLOYEES GROUP BY MANAGER\_ID HAVING COUNT(\*)>=4);

51. Write a query in SQL to display the details of the current job for those employees who worked as a Sales Representative in the past. Table: jobs, employees, job\_history

(SA\_REP = Sales Representative)

SELECT EMPLOYEE\_ID, JOB\_ID FROM EMPLOYEES WHERE EMPLOYEE\_ID IN (SELECT DISTINCT EMPLOYEE\_ID FROM JOB\_HISTORY WHERE JOB\_ID='SA\_REP');

52. Write a query in SQL to display all the information about those employees who earn second lowest salary of all the employees. Table : employees

SELECT \* FROM EMPLOYEES WHERE SALARY =(SELECT SALARY FROM EMPLOYEES ORDER BY SALARY OFFSET 1 ROW LIMIT 1);

53. Write a query in SQL to display the details of departments managed by Susan. Table: departments, employees

SELECT \* FROM DEPARTMENTS WHERE MANAGER\_ID =(SELECT EMPLOYEE\_ID FROM EMPLOYEES WHERE FIRST\_NAME ='Susan');

54. Write a query in SQL to display the department ID, full name (first and last name),salary for those employees who is highest salary drawar in a department. Table: employees

SELECT DEPARTMENT\_ID, FIRST\_NAME || ' ' || LAST\_NAME FULL\_NAME, SALARY FROM EMPLOYEES WHERE (DEPARTMENT\_ID,SALARY) IN (SELECT DEPARTMENT\_ID, MAX(SALARY) FROM EMPLOYEES GROUP BY DEPARTMENT\_ID);

55. Write a query in SQL to display all the information of those employees who did not have any job in the past. Table: employees, job\_history

SELECT \* FROM EMPLOYEES E WHERE NOT EXISTS (SELECT EMPLOYEE\_ID FROM JOB\_HISTORY JH WHERE E.EMPLOYEE\_ID = JH.EMPLOYEE\_ID);