SQL ASSIGNMENT

Database Management System

Submitted By -
Mohd Shoaib Rayeen – 21
Jatin Rohilla – 15

Q1
Write an SQL query to determine the 5th highest salary from employee table without using TOP or limit method.

Query -

```
SELECT DISTINCT e.Salary
  FROM employee e
  WHERE 5 =
        (
          SELECT COUNT(DISTINCT( e1.Salary )) FROM employee e1 WHERE e.SALARY <=
e1.SALARY
        )
;
```

Output –

SALARY
30000

Q2
Retrieve the first and last names of employees with the same salary.

Query –

```
SELECT e.Fname, e.Lname
  FROM employee e
  WHERE Salary IN
        (
          SELECT Salary FROM employee GROUP BY Salary HAVING COUNT(*) > 1
        )
;
```

Output –

FNAME	LNAME
Joyce	English
Ahmad	Jabbar
Alicia	Zelaya

Q3
Retrieve department number of departments that have less than five employees in it.

Query –

```
SELECT Dnumber
  FROM department
  WHERE Dnumber IN ( SELECT DNO FROM employee GROUP BY Dno HAVING COUNT(*) < 5 )
     OR Dnumber NOT IN ( SELECT Dno FROM employee );
```

Output –

DNUMBER
5
1
4

Q4

Retrieve the names of employees who make at least $10,000 more than the employee who is paid the least in the company.

Query –

```
SELECT e.Fname, e.Lname
  FROM employee e
  WHERE ( Salary - 10000 ) >=
        (
          SELECT min(Salary) FROM employee
        )
;
```

Output –

FNAME	MINIT	LNAME	SALARY
Franklin	T	Wong	40000
Ramesh	K	Narayan	38000
James	E	Borg	55000
Jennifer	S	Wallace	43000

Q5

Retrieve the number of male employees in each department.

Query -

```
SELECT DNO, COUNT(*)
  FROM EMPLOYEE
  WHERE SEX = 'M'
  GROUP BY DNO
;
```

Output –

DNO	COUNT(*)
1	1
4	1
5	3

Q6

Retrieve the first and last names and department number and name of all employees directly supervised by James Borg. Show results in ascending alpha order (by last name and then first name).

Query –

With JOIN -

```
SELECT E.FNAME, E.LNAME, E.DNO
  FROM EMPLOYEE      E
       JOIN EMPLOYEE S
           ON E.SUPER_SSN = S.SSN
  WHERE S.FNAME = "James" AND S.LNAME = "Borg"
  ORDER BY E.LNAME, E.FNAME
;
```

Without JOIN -

```
SELECT E.FNAME, E.LNAME, E.DNO
  FROM EMPLOYEE E
  WHERE E.Super_ssn =
        (
          SELECT S.Ssn FROM EMPLOYEE S WHERE S.FNAME = "James" AND S.LNAME = "Borg"
        )
  ORDER BY E.LNAME, E.FNAME
;
```

Output –

FNAME	LNAME	DNO
Jennifer	Wallace	4
Franklin	Wong	5

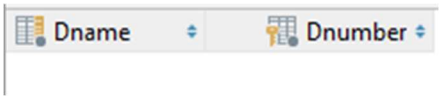
Q7

Retrieve the name and number of departments which have employees who do not work on at least one project. Show results in ascending alpha order. (NOTE: a department should appear on this list if it has an employee who does not work on any project at all.)

Query -

```
SELECT d.Dname, d.Dnumber
  FROM department               d,
       employee                 e
       LEFT OUTER JOIN works_on wo
           ON e.Ssn = wo.Essn
  WHERE wo.Essn IS NULL AND d.Dnumber = e.Dno
;
```

Output –



Q8

For each department, list the department name and the total number of hours assigned to projects controlled by the department (irrespective of the employee to whom they are assigned) and the total number of hours assigned to employees of the department (irrespective of the project involved). Show results in ascending alpha order.

Query –

```
SELECT department.Dname,
       hours_assigned_to_projects_of_dept.project_hour_count,
       hours_assigned_to_employees_of_dept.emp_hour_count
  FROM department
       JOIN
           (
             SELECT p.Dnum, SUM(wo.Hours) AS project_hour_count
               FROM works_on     wo
                    JOIN project p
                        ON wo.Pno = p.Pnumber
               GROUP BY p.Dnum
           )
               AS hours_assigned_to_projects_of_dept
           ON hours_assigned_to_projects_of_dept.Dnum = department.Dnumber
       JOIN
           (
             SELECT employee.Dno, sum(works_on.Hours) AS emp_hour_count
               FROM works_on
                    JOIN employee
                        ON works_on.Essn = employee.Ssn
               GROUP BY employee.Dno
           ) AS   hours_assigned_to_employees_of_dept
           ON hours_assigned_to_employees_of_dept.Dno = department.Dnumber
  ORDER BY department.Dname
;
```

Output –

Dname	project_hour_count	emp_hour_count
Administration	110.0	115.0
Headquarters	25.0	NULL
Research	140.0	160.0

Q9

Retrieve the names of departments which have at least one project which employs every one of the employees of the department that controls the project. Also show the name of the project. Show results in ascending alpha order.

Query –

```
SELECT d.Dnumber, p.Pname
  FROM department d,
       project    p
  WHERE p.Dnum = d.Dnumber AND NOT EXISTS(
           #  employees of a department
              SELECT e1.Ssn
                FROM employee e1
                WHERE e1.Dno = d.Dnumber AND NOT EXISTS(
                  #  should cancel out employees of a project of that department
                                        SELECT wo2.Essn
                                          FROM project       p2
                                         JOIN works_on wo2 ON p2.Pnumber = wo2.Pno
                                          WHERE p2.Dnum = d.Dnumber
                                            AND p2.Pnumber = p.Pnumber
                                            AND e1.Ssn = wo2.Essn
                                )
      )
  ORDER BY p.Pname
;
```

Output –

	Dnumber	Pname
1	4	Newbenefits

Q10

Retrieve the first and last names of employees who work on projects which are not controlled by their departments. Also show the names of the projects, the employee's department number, and the number of the project's controlling department. (All of this should be shown in the same result table.) Show results in ascending alpha order (by last name and then first name and then project name).

Query –

```
SELECT e.Fname, e.Lname,
       e.Dno AS emp_dept_no,
       p.Pname AS project_name,
       p.Dnum AS projects_controlling_dept
  FROM employee      e
       JOIN works_on wo
           ON e.Ssn = wo.Essn
       JOIN project  p
           ON wo.Pno = p.Pnumber
  WHERE p.Dnum != e.Dno
  ORDER BY e.Lname, e.Fname, p.Pname
;
```

Output –

Fname	Lname	emp_dept_no	project_name	projects_controlling_dept
Jennifer	Wallace	4	Reorganization	1
Franklin	Wong	5	Computerization	4
Franklin	Wong	5	Reorganization	1

Q11

Retrieve the first and last names of employees who work on more than the average number of projects. (Note: employees who do not work on any project are to be included in the average.) Display their names, the number of projects they work on, and the average number of projects. (The same average should be repeated in each row.) Show results in ascending alpha order (by last name and then first name). [The average number of projects is the average number of projects worked on per employee.]

Query –

```
SELECT employee.Fname           AS first_name,
       employee.Lname           AS last_name,
       count(works_on.Pno)      AS no_of_projects,
       avg_projects_count.value AS average_no_of_projects
  FROM employee,
       works_on,
       (
         SELECT AVG(project_count_table.no_of_projects) AS value
           FROM
                (
                  SELECT employee.Ssn, count(works_on.Pno) AS no_of_projects
                    FROM employee
                         LEFT OUTER JOIN works_on
                             ON employee.Ssn = works_on.Essn
                    GROUP BY employee.Ssn
                ) AS project_count_table
       ) AS avg_projects_count

  WHERE employee.Ssn = works_on.Essn
  GROUP BY works_on.Essn
  HAVING count(works_on.Pno) > avg_projects_count.value
  ORDER BY employee.Lname, employee.Fname
;
```

Output –

	first_name	last_name	no_of_projects	average_no_of_projects
1	Franklin	Wong	4	2.0000

Q12

Retrieve the name and number of the project which uses the most employees. Also show the total number of employees for that project. If there is more than one project that has attained that maximum, list them all. Show results in ascending alpha order.

Query –

```
SELECT project.Pnumber AS project_number, project.Pname AS project_name,
count(Essn) AS no_of_emps
  FROM works_on
       JOIN project ON project.Pnumber = works_on.Pno
  GROUP BY Pno
  HAVING count(Essn) =
          (
            SELECT max(emp_count_table.emp_count)
              FROM
                    (
                      SELECT count(essn) AS emp_count FROM works_on GROUP BY Pno
                    ) AS emp_count_table
          )
  ORDER BY project.Pname
;
```

Output –

	project_number	project_name	no_of_emps
▶	10	Computerization	3
	30	Newbenefits	3
	2	ProductY	3
	20	Reorganization	3

Q13

Do any departments have a location in which they have no projects? Retrieve the names of departments which have at least one location which is not the same as any of the locations of the department's projects. Show results in ascending alpha order. [This means that one department location is different from every location of every project of that department.]

Answer – NO

Query –

```
SELECT DISTINCT department.Dname
  FROM department,
       dept_locations
       LEFT OUTER JOIN project
           ON dept_locations.Dlocation = project.Plocation
  WHERE project.Plocation IS NULL
    AND department.Dnumber = dept_locations.Dnumber
  ORDER BY department.Dname
```

Output –

Dname

Q14

List the names of dependents that have the same first name as an employee of whom they are not the dependent. Also show the ssn of the employee with the same first name and the ssn of the employee on whom the dependent is dependent (dependent.essn). (All of this should be shown in the same table.) Show results in ascending alpha order.

Query –

```
SELECT dependent.Dependent_name AS dependent_name,
       dependent.Essn AS dependent_on,
       employee.Ssn AS first_name_alike_ssn
  FROM dependent,
       employee
  WHERE dependent.Dependent_name = employee.Fname
    AND dependent.Essn != employee.Ssn
  ORDER BY dependent.Dependent_name
;
```

Output –

dependent_name	dependent_on	first_name_alike_ssn

Q15

Retrieve the first and last names of employees whose supervisor works on any project outside the employee's department. Show results in ascending alpha order (by last name and then first name). [Note that you are to retrieve the employee's name, not the supervisor's.]

Query –

```
SELECT Fname, Lname
  FROM employee e
  WHERE Ssn IN
        (
          SELECT e1.Ssn
            FROM employee      e1
                 JOIN works_on wo1
                     ON e1.Super_ssn = wo1.Essn
                      WHERE e1.Ssn = e.Ssn
                        AND NOT EXISTS(
                          SELECT e2.Ssn, p2.Pnumber
                            FROM employee     e2
                                 JOIN project p2
                                     ON e2.Dno = p2.Dnum
                            WHERE e2.Ssn = e.Ssn AND wo1.Pno = p2.Pnumber
                )
        )
  ORDER BY Lname, Fname
;
```

Output –

	Fname	Lname
1	Joyce	English
2	Ahmad	Jabbar
3	Ramesh	Narayan
4	John	Smith
5	Jennifer	Wallace
6	Franklin	Wong
7	Alicia	Zelaya