DBMS ASSIGNMENT-5
GROUP - 7

1.INNER JOIN

Query 1

```
SELECT Employee.Employee_id, employee.`First Name`, projects.Project_id
FROM projects
INNER JOIN employee ON employee.Employee_id = projects.Employee_id
ORDER BY employee.Employee_id;
```

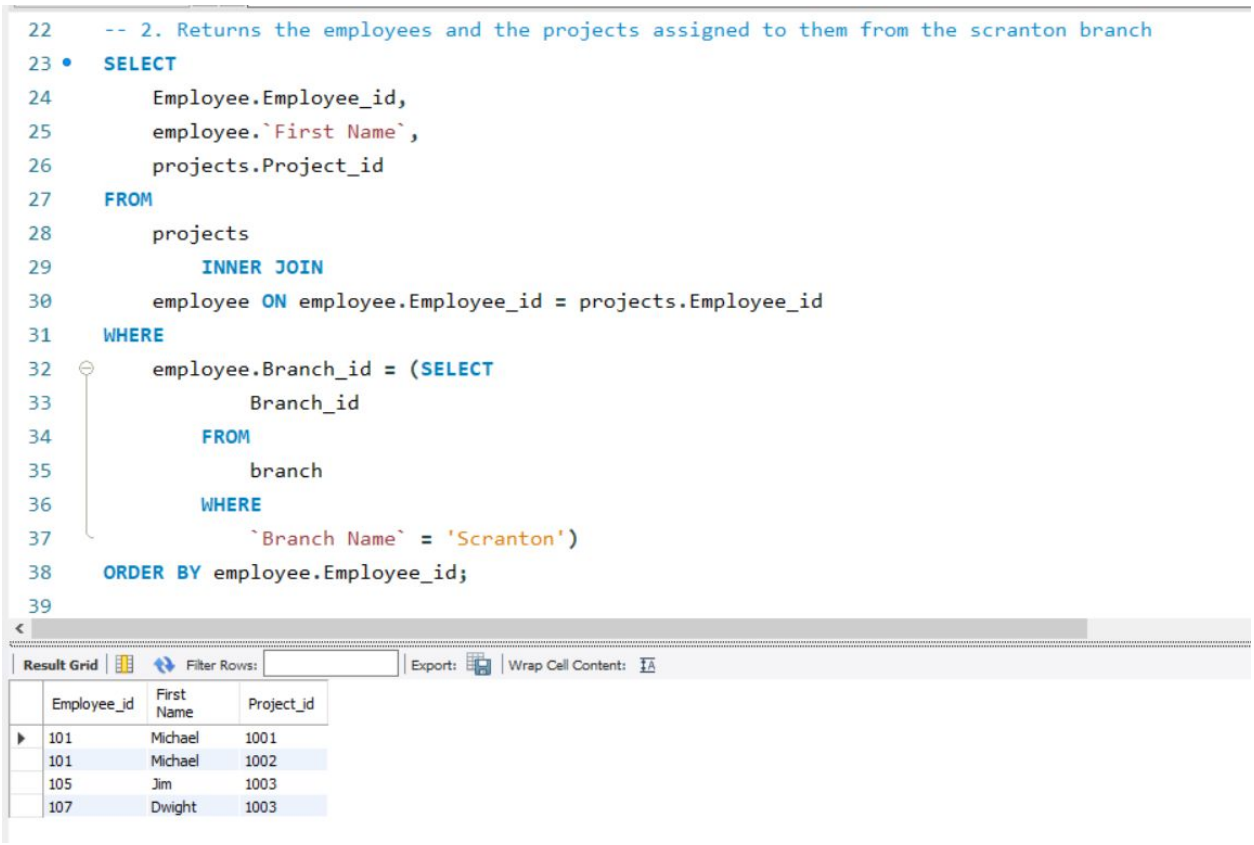
```
11    -- 1. Returns the employees that are assigned to a project
12 •  SELECT
13        Employee.Employee_id,
14        employee.`First Name`,
15        projects.Project_id
16    FROM
17        projects
18            INNER JOIN
19        employee ON employee.Employee_id = projects.Employee_id
20    ORDER BY employee.Employee_id;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Employee_id	First Name	Project_id
101	Michael	1001
101	Michael	1002
102	Jan	1001
103	Andy	1002
104	Karen	1002
105	Jim	1003
107	Dwight	1003

Query 2

```
SELECT Employee.Employee_id, employee.`First Name`, projects.Project_id
FROM projects
INNER JOIN employee ON employee.Employee_id = projects.Employee_id
WHERE employee.Branch_id =(
                    SELECT Branch_id FROM branch
                                WHERE `Branch Name`= 'Scranton')
                                    ORDER BY employee.Employee_id;
```



Query 3

```
SELECT Employee.Employee_id, employee.`First Name`, projects.Project_id,
branch.`Branch Name` FROM projects
        INNER JOIN employee ON employee.Employee_id =
projects.Employee_id
                INNER JOIN branch ON branch.Branch_id = employee.Branch_id
```

ORDER BY employee.Employee_id;

```
40    -- 3. Returns the project ids that have been assigned to employees along with their branch
41 •  SELECT
42        employee.Employee_id,
43        employee.`First Name`,
44        projects.Project_id,
45        branch.`Branch Name`
46    FROM
47        projects
48            INNER JOIN
49        employee ON employee.Employee_id = projects.Employee_id
50            INNER JOIN
51        branch ON branch.Branch_id = employee.Branch_id
52    ORDER BY employee.Employee_id;
53
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Employee_id	First Name	Project_id	Branch Name
101	Michael	1001	Scranton
101	Michael	1002	Scranton
102	Jan	1001	Corporate
103	Andy	1002	Stamford
104	Karen	1002	Utica
105	Jim	1003	Scranton
107	Dwight	1003	Scranton

2. LEFT JOIN

Query 1

SELECT Employee.Employee_id, employee.`First Name`, projects.Project_id
FROM employee
LEFT JOIN projects ON employee.Employee_id = projects.Employee_id
ORDER BY employee.Employee_id;

```
54    -- 4. Returns all employees and their project ids
55 •  SELECT
56        Employee.Employee_id,
57        employee.`First Name`,
58        projects.Project_id
59    FROM
60        employee
61            LEFT JOIN
62        projects ON employee.Employee_id = projects.Employee_id
63    ORDER BY employee.Employee_id;
64
65
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Employee_id	First Name	Project_id
101	Michael	1001
101	Michael	1002
102	Jan	1001
103	Andy	1002
104	Karen	1002
105	Jim	1003
106	Pam	NULL
107	Dwight	1003
108	Creed	NULL
109	David	NULL
110	AJ	NULL

Query 2

```
SELECT employee.Employee_id, employee.`First Name`, branch.`Branch
Name` FROM employee
LEFT JOIN branch ON employee.Branch_id = branch.Branch_id
        WHERE employee.Employee_id IN(
                SELECT Employee_id FROM projects WHERE Project_id = 1002
AND 1003)
                        ORDER BY Employee_id;
```

```sql
-- 5. Returns the employees and their branch names who have been assigned to projects 1002 and 1003
SELECT
    employee.Employee_id,
    employee.`First Name`,
    branch.`Branch Name`
FROM
    employee
        LEFT JOIN
    branch ON employee.Branch_id = branch.Branch_id
WHERE
    employee.Employee_id IN (SELECT
            Employee_id
        FROM
            projects
        WHERE
            Project_id = 1002 AND 1003)
ORDER BY Employee_id;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Employee_id	First Name	Branch Name
101	Michael	Scranton
103	Andy	Stamford
104	Karen	Utica

Query 3

```sql
SELECT employee.Employee_id, employee.`First Name`, projects.Project_id
FROM projects
LEFT JOIN employee ON employee.Employee_id = projects.Employee_id
WHERE employee.Salary>90000
ORDER BY `First Name`;
```

```
84    -- 6. Returns employees with salary greater than 90000 who have been assigned a project
85 •  SELECT
86        employee.Employee_id,
87        employee.`First Name`,
88        projects.Project_id
89    FROM
90        projects
91            LEFT JOIN
92        employee ON employee.Employee_id = projects.Employee_id
93    WHERE
94        employee.Salary > 90000
95    ORDER BY `First Name`;
96
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Employee_id	First Name	Project_id
102	Jan	1001
104	Karen	1002
101	Michael	1001
101	Michael	1002

3. RIGHT JOIN

Query 1

```
SELECT
   Employee.Employee_id,
   employee.`First Name`,
   projects.Project_id
FROM
   employee
      RIGHT JOIN
   projects ON employee.Employee_id = projects.Employee_id
ORDER BY projects.Project_id;
```

```
29      -- Returns all the projects and the employees assigned to them
30  •   SELECT
31          Employee.Employee_id,
32          employee.`First Name`,
33          projects.Project_id
34      FROM
35          employee
36              RIGHT JOIN
37          projects ON employee.Employee_id = projects.Employee_id
38      ORDER BY projects.Project_id;
39
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Employee_id	First Name	Project_id
101	Michael	1001
102	Jan	1001
103	Andy	1002
101	Michael	1002
104	Karen	1002
107	Dwight	1003
105	Jim	1003
NULL	NULL	1004
NULL	NULL	1005

Query 2

```
SELECT
   Employee.Employee_id,
   employee.`First Name`,
   projects.Project_id,
   YEAR(CURDATE()) - YEAR(DoB) AS Age
FROM
   employee
      RIGHT JOIN
   projects ON employee.Employee_id = projects.Employee_id
WHERE
   YEAR(CURDATE()) - YEAR(DoB) < (SELECT
         AVG(YEAR(CURDATE()) - YEAR(DoB))
```

```
    FROM
        Employee)
ORDER BY projects.Project_id;
```

```
93    -- returns projects assigned to those employees with below average age
94    SELECT
95        Employee.Employee_id,
96        employee.`First Name`,
97        projects.Project_id,
98        year(curdate())-year(DoB) AS Age
99    FROM
100       employee
101           RIGHT JOIN
102       projects ON employee.Employee_id = projects.Employee_id
103       WHERE year(curdate())-year(DoB)   < (
104                       SELECT AVG(year(curdate())-year(DoB)) FROM Employee)
105   ORDER BY projects.Project_id;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Employee_id	First Name	Project_id	Age
103	Andy	1002	45
104	Karen	1002	40
107	Dwight	1003	46
105	Jim	1003	42

Query 3

```
SELECT
   branch.Branch_id, branch.`Branch Name`
FROM
   projects
      INNER JOIN
   employee ON employee.Employee_id = projects.Employee_id
      RIGHT JOIN
   branch ON employee.Branch_id = branch.Branch_id
WHERE
   projects.Employee_id IS NOT NULL
GROUP BY `Branch Name`
ORDER BY projects.Project_id;
```

```
123
124     -- 9. Returns all the branches that have been assigned projects
125 •   SELECT
126         branch.Branch_id, branch.`Branch Name`
127     FROM
128         projects
129             INNER JOIN
130         employee ON employee.Employee_id = projects.Employee_id
131             RIGHT JOIN
132         branch ON employee.Branch_id = branch.Branch_id
133     WHERE
134         projects.Employee_id IS NOT NULL
135     GROUP BY `Branch Name`;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

Branch_id	Branch Name
1	Scranton
2	Corporate
4	Stamford
5	Utica