

| Column Name | Type |
|-------------|------|
| project_id | int |
| employee_id | int |

(project_id, employee_id) is the primary key of this table.
 employee_id is a foreign key to Employee table.
 Each row of this table indicates that the employee with employee_id is working on the project with project_id.

Table: Employee

| Column Name | Type |
|------------------|---------|
| employee_id | int |
| name | varchar |
| experience_years | int |

employee_id is the primary key of this table. It's guaranteed that employee_id is not NULL.
 experience_years is not NULL.
 Each row of this table contains information about one employee.

Question ;1075

Write an sql query that reports the average experience years of all the employees for each project, rounded to 2 digits.

```
SELECT
    p.project_id,
    ROUND(AVG(e.experience_years),
    AS average_experience
FROM
    Project p
JOIN
    Employee e
ON
    p.employee_id = e.employee_id
GROUP BY
    p.project_id;
```

| project_id | average_years |
|------------|---------------|
| 1 | 2.00 |
| 2 | 2.50 |

Question-577

Example 1:

Input:

Employee table:

| empId | name | supervisor | salary |
|-------|--------|------------|--------|
| 3 | Brad | null | 4000 |
| 1 | John | 3 | 1000 |
| 2 | Dan | 3 | 2000 |
| 4 | Thomas | 3 | 4000 |

Bonus table:

| empId | bonus |
|-------|-------|
| 2 | 500 |
| 4 | 2000 |

Output:

| name | bonus |
|------|-------|
| Brad | null |
| John | null |
| Dan | 500 |

Write a solution to report the name and bonus of each employee with a bonus less than 1000

```
SELECT
e.name,
b.bonus
FROM
Employee e
LEFT JOIN
Bonus b
ON
e.empId = b.empId
WHERE
b.bonus < 1000 OR b.bonus IS NULL;
```

| empId | name | supervisor |
|-------|--------|------------|
| 3 | Brad | null |
| 1 | John | 3 |
| 2 | Dan | 3 |
| 4 | Thomas | 3 |

Bonus Table:

| empId | bonus |
|-------|-------|
| 2 | 500 |
| 4 | 2000 |

Output:

| name | bonus |
|------|-------|
| Brad | null |
| John | null |
| Dan | 500 |

