181. Employees Earning More Than Their Managers

Easy | Topics | Companies

SQL Schema > Pandas Schema >

Table: `Employee`

```
+-------------+---------+
| Column Name | Type    |
+-------------+---------+
| id          | int     |
| name        | varchar |
| salary      | int     |
| managerId   | int     |
+-------------+---------+
id is the primary key (column with unique values) for this
table.
Each row of this table indicates the ID of an employee, their
name, salary, and the ID of their manager.
```

Write a solution to find the employees who earn more than their managers.

Return the result table in **any order**.

The result format is in the following example.

Example 1:

Input:
Employee table:

id	name	salary	managerId
1	Joe	70000	3
2	Henry	80000	4
3	Sam	60000	Null
4	Max	90000	Null

Output:

Employee
Joe

Explanation: Joe is the only employee who earns more than his manager.

MySQL:

```
# Write your MySQL query statement below
SELECT e.name AS Employee
FROM Employee e
JOIN Employee m
  ON e.managerId = m.id
WHERE e.salary > m.salary;
```

Pandas:

```
import pandas as pd

def find_employees(employee: pd.DataFrame) -> pd.DataFrame:
    # Perform self-join: employee joins with manager (on employee.managerId == manager.id)
    merged = employee.merge(
        employee,
        left_on="managerId",
        right_on="id",
        suffixes=("", "_manager")
    )
```

```
    # Filter where employee's salary is greater than manager's salary
    result = merged[merged["salary"] > merged["salary_manager"]]

    # Return only the employee names
    return result[["name"]].rename(columns={"name": "Employee"})
```

PostgreSQL:

```
-- Write your PostgreSQL query statement below
SELECT e.name AS Employee
FROM Employee e
JOIN Employee m
  ON e.managerId = m.id
WHERE e.salary > m.salary;
```