Replace Employee ID With The Unique Identifier

replace_empid_problem.sql

```sql
SELECT
    eu.unique_id, em.name
FROM
    employees em
        LEFT JOIN
    employeeuni eu ON em.id = eu.id;
```

Query Breakdown:

- The query selects two columns:
  - **eu.unique_id** from the employeeuni table.
  - **em.name** from the employees table.
- The query uses a **LEFT JOIN**, which means:
  - All rows from the **employees** table will be included.
  - If a matching id is found in the **employeeuni** table, it will pull the corresponding unique_id.
  - If no match is found in employeeuni, the unique_id will be **NULL**.

Purpose of the Query:

The query retrieves a list of employees along with their **unique identifiers**, if available. It ensures that **all employees are included**, even if they don't have a unique ID in the employeeuni table.

1. Product Sales Analysis I
2. Invalid Tweets
3. Customer Who Visited but Did Not Make Any Transactions
4. Rising Temperature
5. Not Boring Movies
6. Average Selling Price
7. Number of unique subjects taught by each teacher
8. Project Employees 1
9. Employees Whose Manager Left the Company
10. Students and Examinations
11. Employee Bonus
12. Percentage of Users attended a contest
13. Monthly Transactions 1
14. Queries Quality and Percentagepp
15. Recyclable and Low Fat Product
16. Find Customer Referee
17. Big Countries
18. Article Views 1
19. User activity for the past 30 days 1
20. Average Time Processed Per Machine
21. Game Play Analysis 1
22. Biggest Single Number
23. Classes More Than 5 Students
24. Find Followers Count
25. Product Sales Analysis 3
26. Customers who bought all products

27. Managers with at Least 5 Direct Reports

28. The Number of Employees Which Report to Each Employee

29. Primary Department for Each Employee

30. Triangle Judgement

31. Consecutive Numbers

32. Confirmation Rate

33. Immediate Food Delivery

34. Exchange Seats