CS 5200 Database Management Systems

SQL Workshop 9: Window Functions

Prof. Jeongkyu Lee

Northeastern University
**Khoury College of
Computer Sciences**

Northeastern University
Silicon Valley

WS9-1: Project Employees II

- Write an SQL query to find the team size of each of the employees.

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

- The query result format is in the following example.

```
Employee Table:
+-------------+-----------+
| employee_id | team_id   |
+-------------+-----------+
|     1       |     8     |
|     2       |     8     |
|     3       |     8     |
|     4       |     7     |
|     5       |     9     |
|     6       |     9     |
+-------------+-----------+
```

```
+-------------+-----------+
| employee_id | team_size |
+-------------+-----------+
|     1       |     3     |
|     2       |     3     |
|     3       |     3     |
|     4       |     1     |
|     5       |     2     |
|     6       |     2     |
+-------------+-----------+
```

From https://leetcode.com/problems/project-employees-ii/

SQL Answer

```sql
select employee_id,
    count(*) over(partition
        by team_id) as team_size
from employee
;
```

Comments

- Decide if "Group By" OR

    "Window Function"

- Group By ☐ Aggregation, i.e., one value for each group

- Window Function ☐ Each row

- This case is for ?

WS9-2: Running Total for Different Genders

- Write an SQL query to find the total score for each gender on each day.

- Return the result table ordered by gender and day in **ascending order**.

- The query result format is in the following example.

```
+--------+------------+-------+
| gender | day        | total |
+--------+------------+-------+
| F      | 2019-12-30 | 17    |
| F      | 2019-12-31 | 40    |
| F      | 2020-01-01 | 57    |
| F      | 2020-01-07 | 80    |
| M      | 2019-12-18 | 2     |
| M      | 2019-12-25 | 13    |
| M      | 2019-12-30 | 26    |
| M      | 2019-12-31 | 29    |
| M      | 2020-01-07 | 36    |
+--------+------------+-------+
```

Scores table:
```
+-------------+--------+------------+--------------+
| player_name | gender | day        | score_points |
+-------------+--------+------------+--------------+
| Aron        | F      | 2020-01-01 | 17           |
| Alice       | F      | 2020-01-07 | 23           |
| Bajrang     | M      | 2020-01-07 | 7            |
| Khali       | M      | 2019-12-25 | 11           |
| Slaman      | M      | 2019-12-30 | 13           |
| Joe         | M      | 2019-12-31 | 3            |
| Jose        | M      | 2019-12-18 | 2            |
| Priya       | F      | 2019-12-31 | 23           |
| Priyanka    | F      | 2019-12-30 | 17           |
+-------------+--------+------------+--------------+
```

From https://leetcode.com/problems/running-total-for-different-genders/

SQL Answer

```
SELECT gender,
    day,
    SUM(score_points) OVER(PARTITION BY
        gender ORDER BY day) AS total
FROM Scores
;
```

Comments

- Is it Group BY or Window Function?
- Think about using self-join (sub query)

WS9-3: Homework (Same as WS2-3)

- Write a SQL query to find the highest sales with its corresponding product for each employee. In case of a tie, you should find the product with the smallest product_id.

- Return the result table ordered by employee_id in **ascending order**.

- The query result format is in the following example.

- This is the same as WS2 homework, but your answer must use "Window Function"

```
Sales
+------------+-----------+-------+
| employee_id | product_id | sales |
+------------+-----------+-------+
|          2 |          2 |    95 |
|          2 |          3 |    95 |
|          1 |          1 |    90 |
|          1 |          2 |    99 |
|          3 |          1 |    80 |
|          3 |          2 |    82 |
|          3 |          3 |    82 |
+------------+-----------+-------+
```

```
+-------------+--------------------+
| employee_id | product_id | sales |
+-------------+--------------------+
| 1           | 2          | 99    |
| 2           | 2          | 95    |
| 3           | 2          | 82    |
+-------------+--------------------+
```