Codility_

CodeCheck Report: trainingMR57W6-NS4

Test Name:

Check out Codility training tasks

Summary Timeline

Tasks summary

Task	Time spent	Score
SqlWorldCup SQL (PostgreSQL) ⚠	26 min	100%

Total score

Tasks Details

1. SqlWorldCup

Medium

Given a list of matches in a group stage of the soccer World Cup, compute the number of points each team currently has.

Task Score	Correctness	Performance
100%	100%	Not assessed

Task description

You are given two tables, `teams` and `matches`, with the following structures:

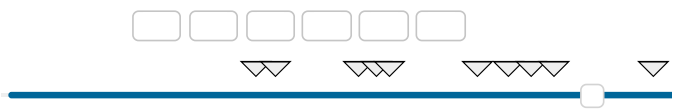
```
create table teams (
    team_id integer not null,
    team_name varchar(30) not null,
    unique(team_id)
);

create table matches (
    match_id integer not null,
    host_team integer not null,
    guest_team integer not null,
    host_goals integer not null,
    guest_goals integer not null,
    unique(match_id)
);
```

Each record in the table `teams` represents a single soccer team. Each record in the table `matches` represents a finished match between two teams. Teams (`host_team`, `guest_team`) are

Solution

Programming language used:	SQL (PostgreSQL)
Total time used:	26 minutes
Effective time used:	26 minutes
Notes:	*not defined yet*

Task timeline



18:40:50 19:06:42

Code: 19:06:42 UTC, sql-postgres, final, score: **100** show code in pop-up

represented by their IDs in the `teams` table (`team_id`). No team plays a match against itself. You know the result of each match (that is, the number of goals scored by each team).

You would like to compute the total number of points each team has scored after all the matches described in the table. The scoring rules are as follows:

- If a team wins a match (scores strictly more goals than the other team), it receives three points.
- If a team draws a match (scores exactly the same number of goals as the opponent), it receives one point.
- If a team loses a match (scores fewer goals than the opponent), it receives no points.

Write an SQL query that returns a ranking of all teams (`team_id`) described in the table `teams`. For each team you should provide its name and the number of points it received after all described matches (`num_points`). The table should be ordered by `num_points` (in decreasing order). In case of a tie, order the rows by `team_id` (in increasing order).

For example, for:

```
  teams:

   team_id | team_name
  ---------+---------------
   10      | Give
   20      | Never
   30      | You
   40      | Up
   50      | Gonna


  matches:

   match_id | host_team | guest_team |
host_goals | guest_goals
  ----------+-----------+------------+----------
--+-------------
   1        | 30        | 20         | 1
| 0
   2        | 10        | 20         | 1
| 2
   3        | 20        | 50         | 2
| 2
   4        | 10        | 30         | 1
| 0
   5        | 30        | 50         | 0
| 1
```

your query should return:

```
   team_id | team_name | num_points
  ---------+-----------+------------
   20      | Never     | 4
   50      | Gonna     | 4
   10      | Give      | 3
   30      | You       | 3
   40      | Up        | 0
```



```sql
SELECT
    t.team_id
    , t.team_name
    , SUM (CASE
        -- won a game
        WHEN (
            (t.team_id = m.host_team AND host_
            OR (t.team_id = m.guest_team AND h
        ) THEN 3
        -- draw
        WHEN host_goals = guest_goals THEN 1
        -- no match played or lost a game
        ELSE 0
    END) AS points
FROM teams t
LEFT JOIN matches m
    ON t.team_id = m.host_team OR t.team_id =
GROUP BY
    team_id
    , team_name
ORDER BY
    points DESC
    , team_id
```

Analysis summary

The solution obtained perfect score.

Analysis

collapse all — Example tests

▼ example — ✔ OK

example test

1. 0.040 s OK

function result:

```
+----+-------+---+
| 20 | Never | 4 |
| 50 | Gonna | 4 |
| 10 |  Give | 3 |
| 30 |   You | 3 |
| 40 |    Up | 0 |
+----+-------+---+
```

collapse all — Correctness tests

▼ both_tables_empty — ✔ OK

no teams nor matches

1. 0.040 s OK
2. 0.040 s OK

▼ no_matches — ✔ OK

some teams, but no matches

1. 0.040 s OK
2. 0.040 s OK

▼ one_match — ✔ OK

many teams, only one match

1. 0.040 s OK

▼ simple_only_draws — ✔ OK

all teams drawed their matches

1. 0.040 s OK

▼ simple_no_draws no draw in any match	✔ OK
1. 0.040 s OK	
▼ simple_distinct_scores all teams have different score at the end, also no need to reorder anything	✔ OK
1. 0.040 s OK	
▼ many_draws test where teams with many draws are better than a single win	✔ OK
1. 0.040 s OK	
▼ same_scores test where many matches have been conducted and all teams tie	✔ OK
1. 0.040 s OK	
▼ only_two_teams_playing many teams, only two of them playing 50 matches	✔ OK
1. 0.040 s OK	
▼ random totally random test, 8 teams, 100 matches	✔ OK
1. 0.040 s OK	
▼ random_some_teams totally random test, 20 teams, 100 matches between 8 first teams	✔ OK
1. 0.040 s OK	