Diagram

team_manager
- 🔑 team_manager_id
- team_id
- manager_id
- start_date
- end_date

team
- 🔑 team_id
- team_name

match
- 🔑 match_id
- match_date
- home_team_id
- away_team_id
- home_score
- away_score

manager
- 🔑 manager_id
- manager_name
- nation_id

player
- 🔑 player_id
- player_name
- nation_id
- team_id

goal
- 🔑 goal_id
- match_id
- player_id
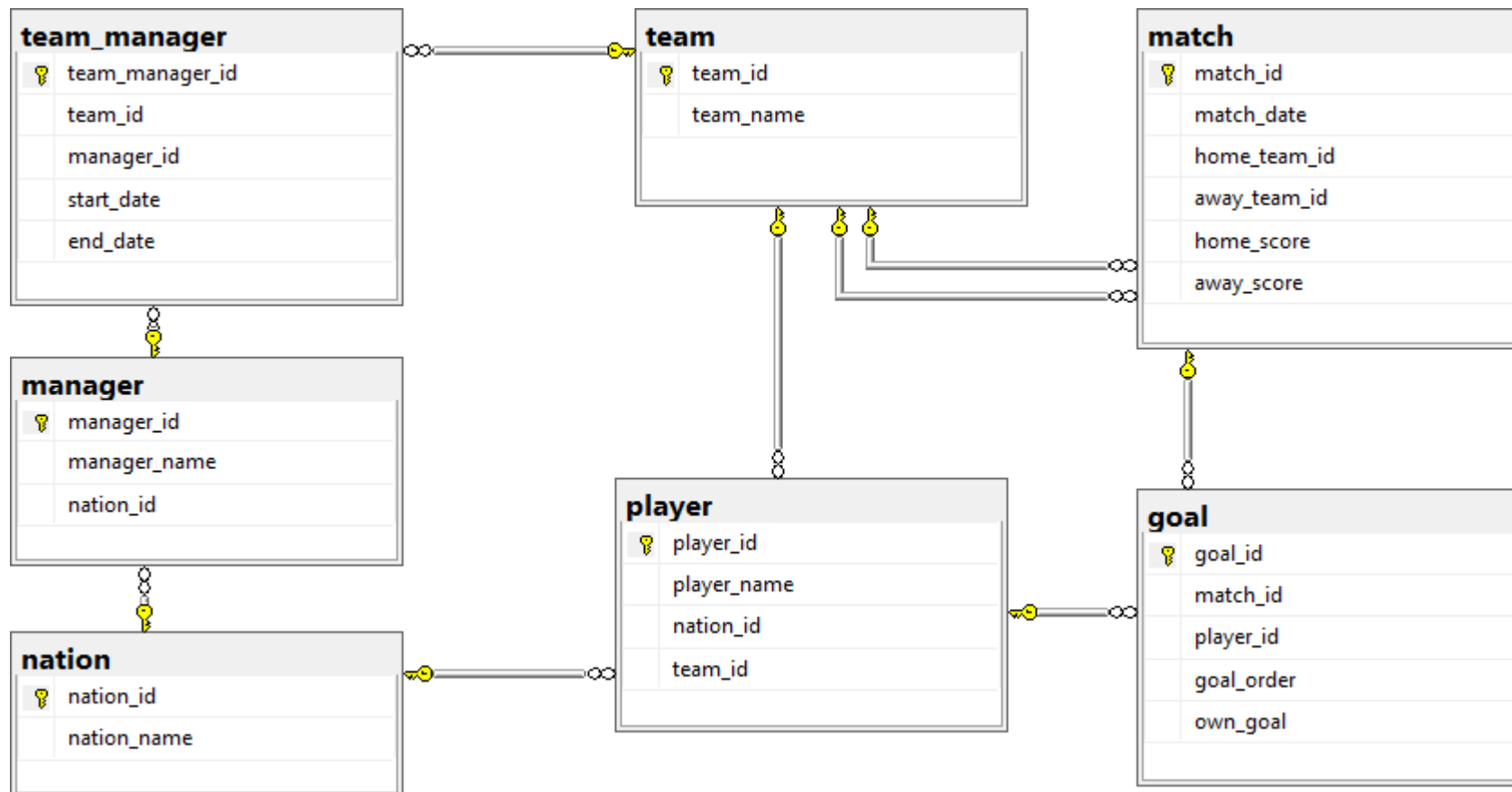- goal_order
- own_goal

nation
- 🔑 nation_id
- nation_name

Explain the diagram:

- `team`: football team.
- `match`: game (`home_team_id`: home team code, `away_team_id`: away team code, `home_score`: home team's goals, `away_score`: away team's goals).
- `goal`: goal (if it's an own goal then own_goal = 1)
- `manager`: coach.
- `nation`: country.
- `player`: player.

- `team_manager`: coaching information.

Students run the premiership.sql file to create the database.

Students write the following queries:

1. Query the names of the teams with the most own goals. Tip: use the **WITH TIES** keyword.
2. Query the name of the top scorer (player who scores the most goals).
3. Query the number of drawn matches with goals.
4. Query the name of the team that scores the least away goals.
5. Query the name of the team that concedes the most goals.
6. Query the average number of goals scored.
7. Query the name of the country with the most players currently playing in the league.
8. Query the name of the football team that has changed coaches the most.
9. Query the name of the coach who has led for the longest and the corresponding duration of leadership.
   Note: if `end_date` is `NULL`, then take `end_date` as the maximum value present in the `team_manager` table.
10. Query the names of the matches with the most goals scored and the corresponding number of goals.
    Note: match names should be in the format of home team v. away team, for example, Chelsea v. Arsenal.