

5. FootballLeague

Design a program that simulates a league consisting of teams, players, and coaches. The program should allow you to simulate league games and determine the outcome (win, lose, or draw) of each match.

Human:

- Attributes: name (string) and age (integer).
- Methods: introduce() - prints a brief introduction of the human.

FootballPlayer (inherits from Human):

- Additional Attributes: position (string), number (integer), performance_score (float), and salary (float).
- Additional Methods: score_goal() - simulates the player scoring a goal.

Coach (inherits from Human):

- Additional Attributes: experience (integer) and salary (float).

Team:

- Attributes: name (string), captain (FootballPlayer object), players (list of FootballPlayer objects), coach (Coach object), score (integer), and balance (float).
- Methods:
- assign_captain - assigns a player as the captain of the team.
- add_player - adds a player to the team.
- remove_player - removes a player from the team.
- display_players - displays the details of all players in the team.
- has_enough_players - checks if the team has exactly 11 players.
- buy_player - buys a player from another team by paying the transfer fee.

League:

- Attributes: name (string) and teams (list of Team objects).
- Methods: add_team - adds a team to the league.
- remove_team(team: Team) - removes a team from the league.
- simulate_match - simulates a match between two teams.
- display_standings - displays the current standings of the league.
- select_random_teams - selects two teams randomly from the teams in the league.
- display_teams_by_points - displays the names of the teams in descending order of their points.

In the simulation, you can create multiple instances of teams, players, and coaches and add them to the league. Use the provided methods to assign captains, add and remove players from teams, display the team's details, check if a team has enough players, and buy players from other teams. You can also use the simulate_match() method to simulate matches between teams and determine the outcome (win, lose, or draw) based on some predefined rules. Finally, the display_standings() method will show the current standings of the league based on the match results, and the display_teams_by_points() method will display the names of the teams in descending order of their points.