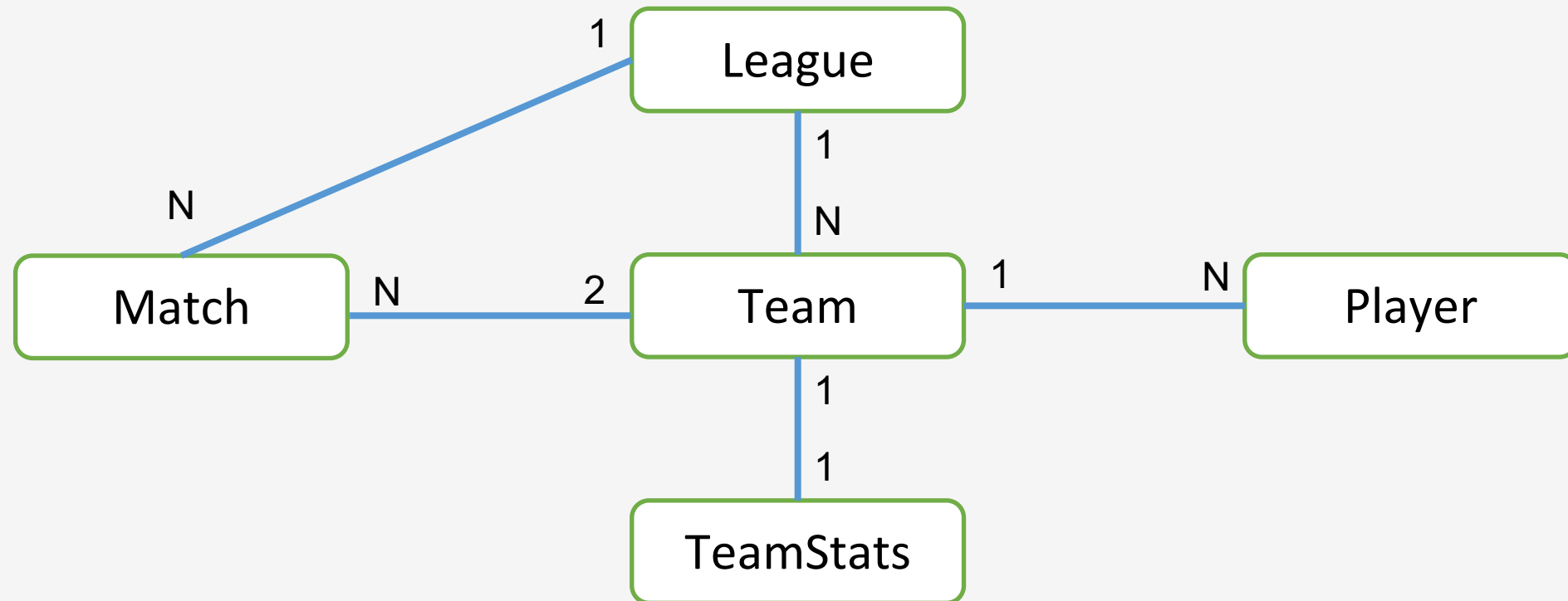




Liga Piłkarska



# Klasa Player



- Atrybuty funkcji inicjującej: name, age, monthly\_salary.
- Atrybuty: name, age, monthly\_salary, team początkowo ustawiony na None.
- Uwaga: atrybut **team** będzie obiektem klasy Team, nie napisem przedstawiającym nazwę zespołu!!!
- Funkcje:
  - Introduce(), która zwróci przywitanie o następującej logice:
    - Jeżeli zawodnik ma przypisany klub
      - My name is **name** I play for **team\_name**.
    - Jeżeli zawodnik nie ma przypisanego klubu:
      - My name is **name** I don't have a team.
  - \_\_str\_\_, która zwróci wywołanie funkcji introduce()



- Atrybuty funkcji inicjalizującej: name, abbreviation
- Atrybuty: name, abbreviation, players (lista przechowująca zawodników)
- Funkcje:
  - hire\_player, która przyjmuje jako argument zawodnika, dodaje go do listy zawodników oraz ustawi wartość atrybutu zawodnika **team** na zespół, do którego został zatrudniony. (Nie napis tylko Obiekt klasy Team!!!)
  - fire\_player, która przyjmuje nazwisko zawodnika a wykonuje proces zwolnienia zawodnika z klubu. Żeby to uczynić należy odnaleźć zawodnika z takim nazwiskiem na liście zawodników, usunąć go stamtąd, oraz ustawić wartość atrybutu **team** tego zawodnika na **None**



- Funkcje:
  - `__str__`, która zwróci nazwę klubu a za nią w nawiasie jej skrót:  
**Manchester City (MNC)**
  - Property `size` zwracającą liczbę zawodników w klubie
  - Property `monthly_cost` zwracającą całkowity miesięczny koszt utrzymania wszystkich zawodników
  - Property `yearly_cost` zwracającą całkowity roczny koszt utrzymania wszystkich zawodników



- Argumenty funkcji inicjalizującej: drużyny **host** oraz **guest**
- Atrybuty: host, guest, result początkowo ustawiony na None.
- Funkcje:
  - play(), która utworzy krotkę zawierającą dwie losowe liczby z przedziału od 0 do 5 i przypisze je do atrybutu **result**.
  - Property goals, która zwróci sumę goli w meczu
  - \_\_str\_\_, która zwróci napis w postaci:  
host.name vs guest.name



- Argumenty funkcji inicjującej: Brak
- Atrybuty: points, wins, loses, ustawione początkowo na **0 zero**. Będzie to statystyka początkowa w nowym sezonie, po jednej dla każdego zespołu.
- Funkcje:
  - `add_win()`, która zwiększa liczbę punktów o 3 oraz liczbę wygranych o 1
  - `add_loss()`, która zwiększa liczbę przegranych o 1
  - `add_draw()`, która zwiększa liczbę punktów o 1 oraz liczbę remisów o 1



- nadpiz funkcję `__str__`, która zwraca napis w postaci:  
points wins draws loses  
12 3 3 2
- jakby był to wiersz w tabeli końcowej. Zwróć uwagę na formatowanie. Niech każda wartość będzie wyrukowana na szerokości dwóch znaków. (Tutaj pomogę I krótko dodam rzeczy do formatowania napisów)





- Atrybuty funkcji inicjującej: year, table, matches
- Atrybuty: year, table, matches
- Funkcje:
  - Później



- `generate_matches`: funkcja która przyjmuje listę zespołów a zwraca listę meczów jakie będą rozegrane w lidze. Wszystkie możliwe pary z rozróżnieniem na pierwszy mecz i rewanż.
- funkcja `build_league`, która przyjmuje rok oraz listę utworzonych zespołów a następnie:
  - Wygeneruje wszystkie możliwe mecze korzystając z funkcji `generate_matches`
  - Zbuduje tabelę jako słownik w formacie `{Team: TeamStats}`
  - Utworzy i zwróci obiekt typu `League`



- `hire_players_in_teams` przyjmująca listę zawodników oraz listę zespołów i zatrudniającą zawodników po kolei do zespołów według zasady: 1szy zawodnik do 1szego zespołu, 2gi do 2go itd. aż 11 zawodnik na liście ponownie ma być zatrudniony do 1go zespołu zakładając, że w naszej lidze będzie grało 20 zespołów.



- Atrybuty funkcji inicjalizującej: year, table, matches
- Atrybuty: year, table, matches
- Funkcje:
  - `update_statistics` przyjmującą mecz a która będzie aktualizować statystyki zespołów które brały udział w meczu
  - `play_season()` za pomocą której rozegrasz wszystkie mecze do końca sezonu oraz zaaktualizujesz po każdym rozegranym meczu statystyki.
  - nadpisz funkcję `__str__`, która zwróci tabelę w formacie

1. Liverpool FC	35	11	2	5
2. Wolverhampton Wanderers	34	10	4	4
3. West Ham United	32	10	2	6

# Blok main:



- Wczytaj zawodników `load_players()`
- wczytaj drużyny: `load_teams()`
- Zatrudnij zawodników w klubach: `hire_players_in_teams()`
- Zbuduj ligę: `build_league()`
- Rozegraj sezon: `play_season()`
- Wydrukuj końcowe wyniki ligi `print(league)`