Challenge 2: Implementing a Sports Team!

In this exercise, you have to perform aggregation between 3 classes.

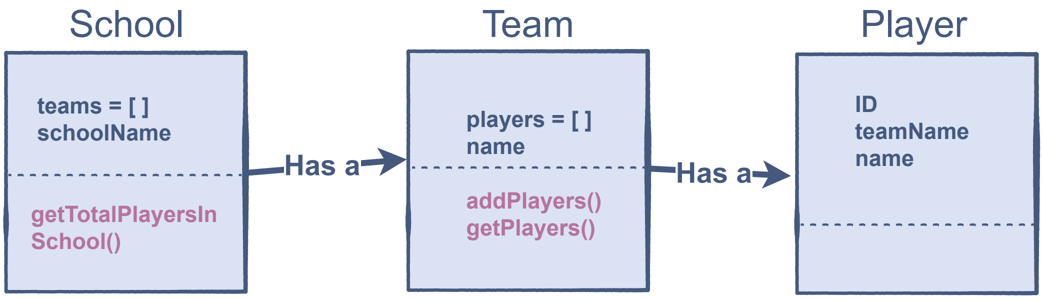
We'll cover the following

- Problem Statement
  - Task 1
  - Task 2
  - Task 3
- Coding Exercise

Problem Statement

You have to implement **3** classes: `School`, `Team`, and `Player`, such that an instance of a `School` should contain instances of `Team` objects. Similarly, a `Team` object can contain instances of `Player` class.

Consider this diagram for clarification:



School, Team, Player: Class Representation

You have to implement a `School` class containing a list of `Team` objects, and a `Team` class comprising of a list of `Player` objects.

Task 1

- `Player` class should have *three* properties that will be set using an initializer:

1. `ID`
2. `name`
3. `teamName`

Task 2

- `Team` class will have *two* properties that will be set using an initializer:

    1. `name`
    2. `players`, a list with `player` class objects in it.

- It will have *two* methods:

    1. `addPlayer()`, which will **add** new `player` objects in the `players` list.

    2. `getNumberOfPlayers()`, which will **return** the total number of players in the `players` list.

Task 3

- The `School` class will contain *two* properties that will be set using an initializer:

    1. `teams`, a list of `team` class objects
    2. `name`

- It will have *two* methods:

    1. `addTeam`, which will **add** new `team` objects in the `teams` list.
    2. `getTotalPlayersInSchool()` method counts the total players in all of the teams in the `School` and **returns** the count!

So, your school should have these players in their respective teams:

Player ID’s	Player Names	Teams
1	Harris	Red
2	Carol	Red

1	Johnny	Blue
2	Sarah	Blue

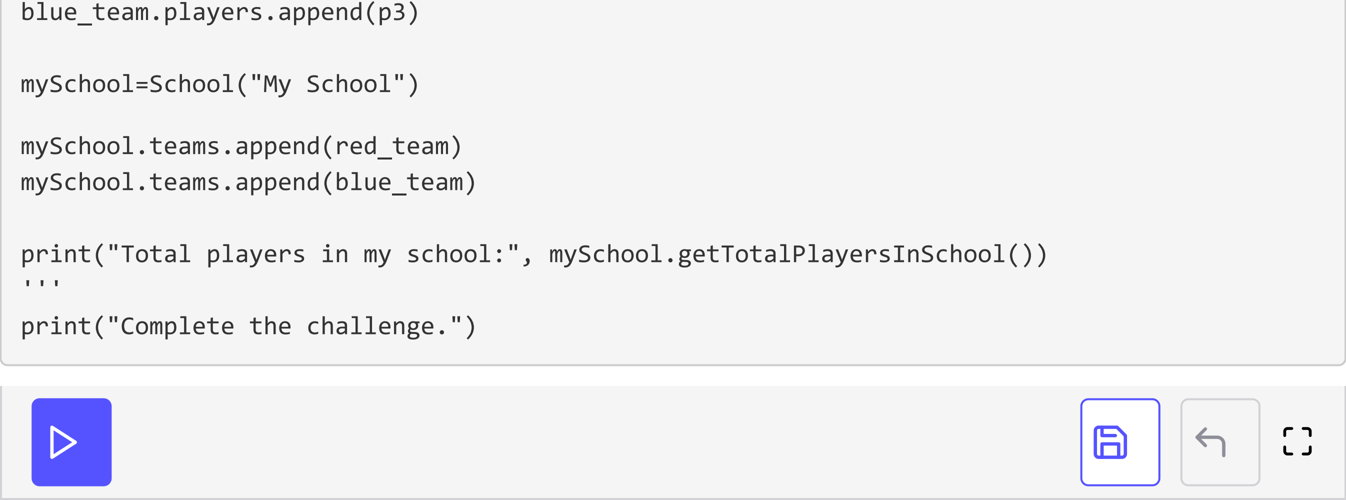
Coding Exercise

First, take a close look and design a step-by-step algorithm before trying the implementation. This problem is designed for your practice, so initially try to solve it on your own. If you get stuck, you can always refer to the solution provided in the solution review.

Good luck!

problem solution

```
# Player class
class Player:
    pass
    # Complete the implementation


# Team class contains a list of Player
# Objects
class Team:
    pass

    # Complete the implementation


# School class contains a list of Team
# objects.
class School:
    pass


# Complete the implementation

# code to test the implementation
# remove backticks when you want to test the implemenation of your code
'''
p1 = Player("Harris", 1, "Red");
p2 = Player("Carol", 2, "Red");

p3 = Player("Johnny", 1, "Blue");
p4 = Player("Sarah", 2, "Blue");

red_team=Team("Red Team")
red_team.players.append(p1)
red_team.players.append(p2)

blue_team=Team("Blue Team")
blue_team.players.append(p2)
```



```
blue_team.players.append(p3)

mySchool=School("My School")

mySchool.teams.append(red_team)
mySchool.teams.append(blue_team)

print("Total players in my school:", mySchool.getTotalPlayersInSchool())
'''
print("Complete the challenge.")
```

The solution is explained in the next lesson!