**CS3733-D23 Wong**

**Assignment 3 – Models to Code**

Due F 3-31 at 11:59pm. Late submission policy: 10% off if submitted by Sa 4-1 at noon. This assignment is to be done in teams of two with the same partner as in Assignment 2. Zip your single IntelliJ project and upload it to Canvas using the Assignment 3 link.

This fictitious exercise is to ensure that all team members are able to take a simple class diagram and transform it into Java code. The International Olympic Committee (IOC) recognizes a number of International Sports Federations (ISFs). Each of the ISFs contain National Sports Federations (NSFs). Players become registered members of the NSFs.

Create a new project named OlympicSports in IntelliJ with main() in a public OlympicSports class. Be careful to use the exact naming of the classes, attributes, and methods in the diagram or your assignment may not pass the auto-grader! Please note that the different visibility specified in the class diagram is just for this exercise and should not be regarded as a good design.

1. All classes in the diagram below are package private. Create the getters and setters using IntelliJ’s code generator and make sure the visibility is correctly implemented. All collections are to be coded to a supertype of List<>. For example, if there is one person to many credit cards, you would create List<CreditCard> as an attribute in the Person class. The visibility of the attribute should match that of the other attributes in that same class.
2. IOC (International Olympic Committee) class. There are IOCs for each continent. An IOC contains many ISFs. For example, the African IOC might contain the ISFs of soccer, tennis, and surfing.
   1. The constructor creates a empty List of ISFs.
   2. IOC.countNSFs() determines how many NSFs are in all of the IOC’s ISFs.
   3. IOC.countPlayersinISFNSFs() determines how many players are registered across all of this IOC’s ISF’s NSFs. Note: players listed in more than one NSF will be counted multiple times.
3. ISF (International Sports Federation) class. This class represents one of the international sports federations within an IOC.
   1. The constructor creates a empty List of NSFs.
   2. countNSFs() determines how many NSFs the ISF has.
   3. countPlayersInNSFs() determines how many players are registered in all of its National Sport Federations (NSFs). Players are counted multiple times if they are registered under more than one NSF.
4. NSF (National Sports Federations) class.
   1. The constructor creates a empty List of Players.
   2. Create the toString() method for just the country and sport attributes. For example, “United States Basketball” or “Indonesia Badminton”.
   3. You do not have to write removePlayer().
   4. listNSFPlayers() displays the NSF’s list of players in text in the following format:

NSF country sport’s players: player1\_name, player2\_name, player3\_name  
For example: United States Tennis players: Serena Williams, Roger Federer, Martina Navratilova

Do not include a comma at the end of the lists.

1. Player class.
   1. The Player() constructor creates a empty ArrayList of NSFs.
   2. Create the toString() method for just the player’s name (first name and last name with a space in between).
   3. You do not have to write removeNSF().
   4. listPlayerNSFs() displays the player’s list of NSFs in text in the following format:

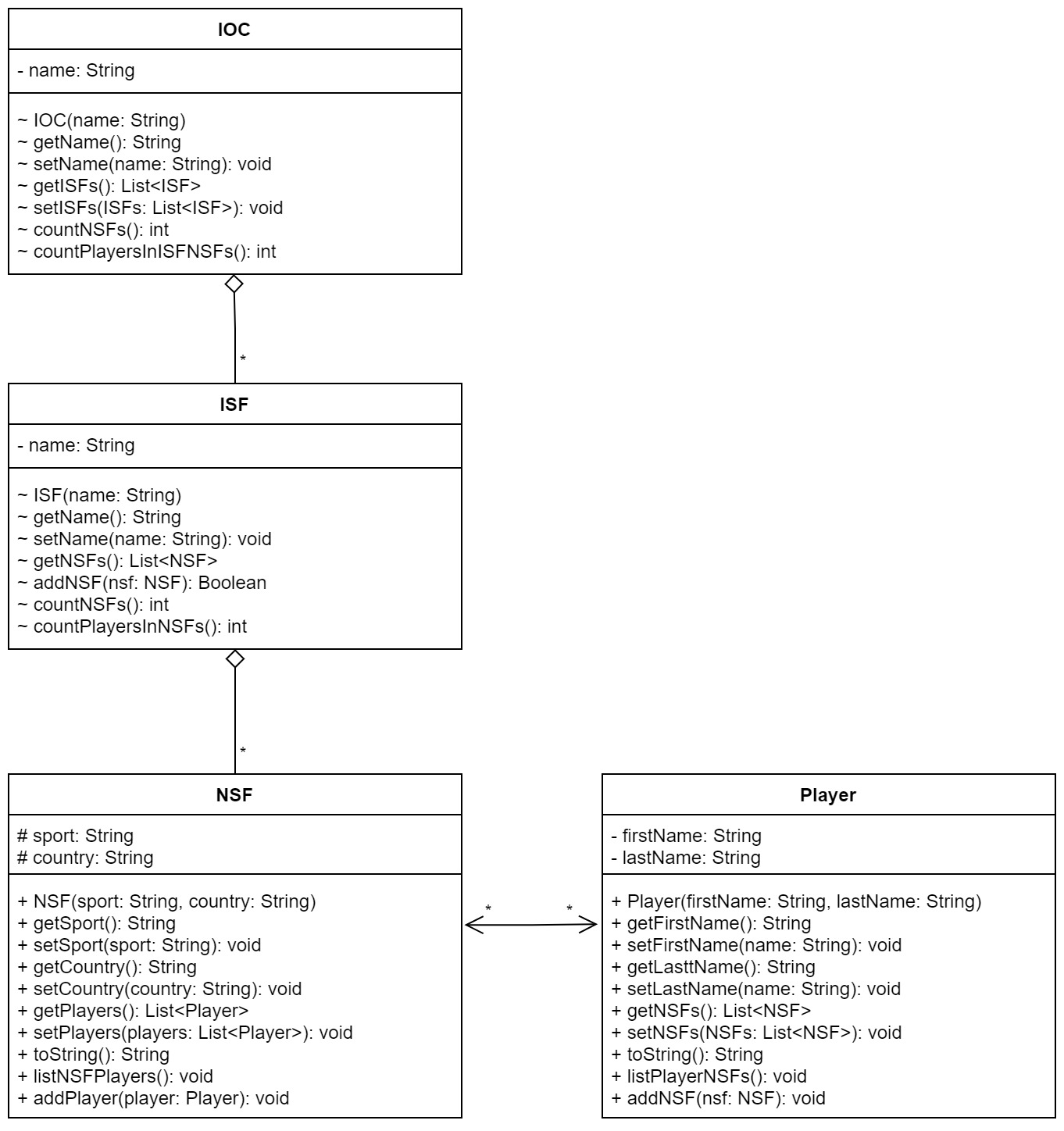
Player name: NSF1, NSF2, NSF3

For example, Amy Tam: Spain Soccer, Spain Swimming, Spain Curling

Do not include a comma at the end of the lists.

1. main(). Write the following to test your code.
   1. Create 5 NSFs and 15 players.
   2. For each of the 5 NSFs, add at least 3 players to them.
   3. Take 3 players and add at least 2 national sports federations to each of them.
   4. Create 2 ISFs. The first ISF has 3 national sports federations and the second ISF has the remaining 2 national sports federations.
   5. Create an IOC that contains the 2 ISFs.
   6. Display the players for the NSFs.
   7. For two players, display the NSFs they are in. Make sure at they are each registered for more than one NSF.
   8. Display the total number of players registered with NSFs in each ISF, and in the IOC
   9. Display the total number of NSFs in each ISF and and in the IOC.

See next page for the UML class diagram that you will be implementing.



Note: You will want to refer to the updated **Designs to Code** PowerPoint slides for the many-to-many bidirectional association.