

CSCE 145 Lab 13

Objectives

The goal of this lab assignment is to introduce you to *inheritance* in Java.

Instructions

1. You will write 4 Java class files - `FootballPlayer.java`, `OffensivePlayer.java`, `DefensivePlayer.java`, and `QuarterBack.java`.
2. You will also be using a `TestFootballPlayer.java` file that you can download from <http://www.cse.sc.edu/~carrollh/CSCE145/lib/TestFootballPlayer.java>.
3. The following are the specifications for the *FootballPlayer* class:
 - Instance variables *name(String)*, *gamesPlayed(int)*, and *numberOfInjuries(int)*
 - A constructor that takes 3 arguments (*String*, *int*, and *int*) and sets the instance variables accordingly
 - Methods include all getters and setters (accessors and mutators) as well as a *display()* method with no return type that outputs the *FootballPlayer's* information
4. The following are the specifications for the *OffensivePlayer* class:
 - This class should inherit from the *FootballPlayer* class
 - Instance variables are *totalYards(int)*, *numberOfTDs(int)*
 - A constructor that takes 5 arguments (*String*, *int*, *int*, *int*, and *int*) and sets the instance variables accordingly
 - Methods include all getters and setters as well as a *display()* method with no return type that outputs the *OffensivePlayer's* information (should call the super display method as well)
5. The following are the specifications for the *DefensivePlayer* class:
 - This class should inherit from the *FootballPlayer* class
 - Instance variables are *numberOfInterceptions(int)*, *numberOfTackles(int)*
 - A constructor that takes 5 arguments (*String*, *int*, *int*, *int*, and *int*) and sets the instance variables accordingly
 - Methods include all getters and setters as well as a *display()* method with no return type that outputs the *DefensivePlayer's* information (should call the super display method as well)
6. The following are the specifications for the *QuarterBack* class:
 - This class should inherit from the *OffensivePlayer* class
 - Instance variables are *interceptionsThrown(int)*, *completionPercentage(double)*
 - A constructor that takes 7 arguments (*String*, *int*, *int*, *int*, *int*, *int*, *double*), i.e., the five arguments for *OffensivePlayer* plus two more for the *QuarterBack's* instance variables, and sets the instance variables accordingly
 - Methods include all getters and setters as well as a *display()* method with no return type that outputs the *QuarterBack's* information (should call the super display method as well)
7. Once all of these classes have been written you should download `TestFootballPlayer.java` and test that all of your classes have been written correctly.

Assuming you have written everything correctly, your output should look like this (if your software “fails” anything, or the output at the bottom is different, something is incorrect in your code and you must fix it):

Welcome to the Football Player Program.

Creating FootballPlayer...Passed
Creating OffensivePlayer...Passed
Creating DefensivePlayer...Passed
Creating QuarterBack...Passed

Changing name of player1...Passed
Changing name of player2...Passed
Changing name of player3...Passed
Changing name of player4...Passed

Changing games played of player1...Passed
Changing total yards of player2...Passed
Changing number of tackles of player3...Passed
Changing number of Interceptions thrown of player4...Passed

Printing Player Information:

Name: Jim
Games Played: 234
Number Of Injuries: 8

Name: John
Games Played: 128
Number Of Injuries: 3
Total Yards: 1300
Number Of Touchdowns: 14

Name: Jack
Games Played: 56
Number Of Injuries: 2
Number Of Interceptions: 3
Number Of Tackles: 71

Name: Bert
Games Played: 336
Number Of Injuries: 13
Total Yards: 1300
Number Of Touchdowns: 14
Interceptions Thrown: 29

Completion Percentage: 54.5

Thank you for using this program. Goodbye.

Zip and upload your classes to the CSE Dropbox.