

Project 2: Final Submission

Purpose and Goal: As a basketball fan and data scientist, I want a program that allows me to create instances of players and set and interact with their statistics.

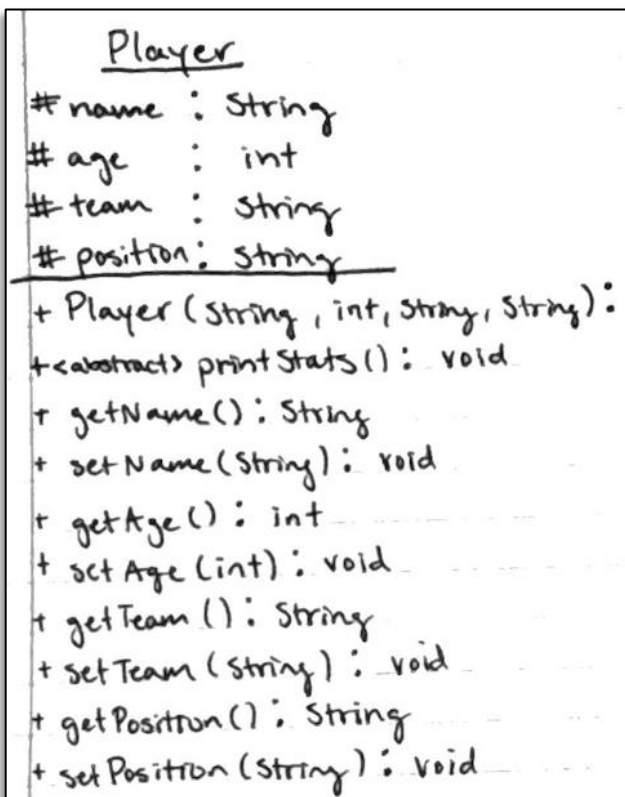
Use of an Interface: In the context of this program, using an interface is useful for defining a common set of behaviors that any type of player must have (displaying statistics in this scenario). This makes the program more flexible and easier to expand in the future. Basically, an interface helps make different parts of the program would together smoothly as new features/player types are added.

Future Expansion: Future expansions of this program could include creating additional methods that compare players to averages or create graphs. This would enhance analytic application and provide deeper insights. Additionally, integrating functionality to write player statistics to a CSV could facilitate data sharing.

Polymorphism: The use of Polymorphism in the 'StatsDisplayable' interface and abstract methods in the 'Player' class ensures that all players can display their stats consistently.

White Box Testing: Completed in WhiteBoxTesting.java

UML of Each Class:



BasketballPlayer

```
- points : int  
- rebounds : int  
- assists : int  
+ BasketballPlayer (String, int, String, String, int, int, int) :  
+ printStats () : void
```

FootballPlayer

```
- totalYards : int  
- touchdowns : int  
+ FootballPlayer (String, int, String, String, int, int) :  
+ printStats () : void
```

StatsDisplayable

```
+ printStats () : void
```

Driver

```
- static myScanner : Scanner  
- static players : ArrayList < Player >  
+ static main (String []) : void  
- static addBasketballPlayer () : void  
- static addFootballPlayer () : void  
- static displayAllPlayers () : void
```

UML of Relationships (IS-A / HAS-A):

