

AP Computer Science

Unit 9 Test Project

Implement a superclass **BaseballPlayer**. Make three classes, **Hitter**, **Pitcher**, and **Fielder** that inherit from **BaseballPlayer**. A **BaseballPlayer** has a name and a number of games played. A **Hitter** has at bats, hits, and a batting average. The batting average is determined by this formula: $\text{ave} = \text{hits}/\text{at_bats}$. A **Pitcher** has innings pitched, earned runs, and an earned run average. The earned run average is determined by this formula: $\text{era} = (9 * \text{earned_runs})/\text{innings_pitched}$. A **Fielder** has assists, putouts, errors, and a fielding percentage. The fielding percentage is determined by this formula: $\text{pct} = (\text{assists} + \text{putouts})/(\text{assists} + \text{putouts} + \text{errors})$.

BaseballPlayer should be an abstract class. It should have a parametric constructor, a `toString` method, and an abstract method: `public abstract void setStat();`. The `setStat()` method should determine batting average, earned run average, and fielding percentage in the **Hitter**, **Pitcher**, and **Fielder** classes respectively. Each of these stats should be of type `String` so that trailing zeros can be displayed. Abstract classes are not allowed to be declared as `private`.

The three subclasses, in addition to the `setStat()` method, should have a parametric constructor and a `toString()` method that will call the `toString()` method of the superclass.

The driver class should fill an `ArrayList` of **BaseballPlayers** with the following data:

```

                Name           Games  At Bats Hits
new Hitter("Tyler Lawson", 29, 78, 25);
new Hitter("Jason Brown", 16, 54, 13);
                Name           Games  Innings  Earned Runs
new Pitcher("Tom Jensen", 9, 50.7, 18);
new Pitcher("Jason Brown", 10, 72.3, 15);
                Name           Games Assists Putouts Errors
new Fielder("Willie Starks", 30, 11, 97, 4);
new Fielder("Jason Brown", 16, 23, 6, 3);
```

The driver program should then print out the contents of the ArrayList using an implicit call to the appropriate toString() method (polymorphism).

The output should look like this...

Name: Tyler Lawson
Games: 29
Batting Average: 0.321

Name: Jason Brown
Games: 16
Batting Average: 0.241

Name: Tom Jensen
Games: 9
Earned Run Average: 3.20

Name: Jason Brown
Games: 10
Earned Run Average: 1.87

Name: Willie Starks
Games: 30
Fielding Percentage: 0.964

Name: Jason Brown
Games: 16
Fielding Percentage: 0.906

To complete this project, turn in this output with BaseballPlayer.java, Hitter.java, Pitcher.java, Fielder.java, and BaseballDriver.java.