Athlete Scores

Prerequisites

Before you begin this exercise, you need mastery of the following:
Unit testing: Knowledge of unit testing

Goal

Reinforce your ability to write test drivers

Outcomes

You will demonstrate mastery of the following:
Producing a test driver that verifies the implementation of a Java class

Background

In this assignment, you will write a test driver to verify that an untested piece of code is working correctly. We will provide the untested code --- a class called AthleteScores.

Description

Class AthleteScores
AthleteScores contains four instance variable: the name of an athlete and the three scores. A complete implementation of AthleteScores is included in the student archive student-files.zip. AthleteScores contains the following constructor and methods:

- public AthleteScores(String initialName, double initialScoreOne, double initialScoreTwo, double initialScoreThree). Constructor that initializes the instance variables name, scoreOne, scoreTwo, and scoreThree.

- String getName(). Returns the name of the athlete.

- double getScoreOne(). Returns the value of the athlete's first score.

- double getScoreTwo(). Returns the value of the athlete's second score.

- double getScoreThree(). Returns the value of the athlete's third score.

- void setScoreOne(double newScore). Modifies the value of the athlete's first score.

- void setScoreTwo(double newScore). Modifies the value of the athlete's second score.

- void setScoreThree(double newScore). Modifies the value of the athlete's third score.

- double getMinimum(). Returns the smallest of the three scores.

- boolean equals(Object object). Overrides the method Object.equals. Two AthleteScores objects are equal if their names are equal.

- String toString(). Overrides the method Object.toString. Returns the string representation of an AthleteScores object. The String returned has the following format:

name,scoreOne,scoreTwo,scoreThree

The fields are separated by a comma (,). You can assume that the fields themselves do not contain any commas.

Class TestAthleteScores

The file TestAthleteScores.java is an incomplete test driver. This file is included in the student archive student-files.zip. Class TestAthleteScores contains the following methods:

- public static void assertTrue(String message, boolean condition). This method is complete and should not be modified. It displays a message, in the standard error stream, if the value specified by parameter condition is false. For example, if a designer want to verify that two integers are equal and display an error message if they are not, then the designer could write:

  assertTrue("1: integers are not equal", x == y);

  If integers x and y are not equal, then assertTrue will display the error message specified by the first parameter. The error message begins with a number because it is often convenient to number error messages.

  public static void main(String[] args). This method is incomplete. It should test each method in the class AthleteScores. If an error is found, it should display an error message.

Files

The following files are needed to complete this assignment.

- Student-files.zip. Download this file. This archive contains the following:
  - AthleteScores.java. A complete implementation
  - TestAthleteScores.java. An incomplete implementation

Task

Complete the implementation of the test driver. Use the method assertTrue to display an error message when a problem is found. Each error message should contain a unique number. Document using Javadoc and follow Sun's code conventions.

The following steps will guide you through this assignment. Work incrementally and test each increment. Save often.

1. write the code to test the accessors: getName, getScoreOne, getScoreTwo, and getScoreThree.

2. Next, write the code to test the mutators: setScoreOne, setScoreTwo, and setScoreThree.
3. Then, write the code to test the method getMinimum. Include the following test cases:
   - Verify that it returns the first score when the first score is the smallest score.
   - Verify that it returns the second score when the second score is the smallest score.
   - Verify that it returns the third score when the third score is the smallest score.
   - Next, write the code to test the method equals. Include the following test cases:
     - Verify that it returns true when objects being compared have the same name.
     - Verify that it returns false when objects being compared do not have the same name.
     - Verify that it returns false when objects being compared are not instances of AthleteScores.
4. Finally, write the code to test the method toString.