PBL: Grade Book Part 6: `class AssignmentScore`

©2026 Chris Nielsen – *www.nielsenedu.com*

This assignment continues the project to create software for a school grade book that will hold the scores that students received for their assignments.

Eventually, each student will have their own list of scores for the various assignments they have completed. The score will be stored in `AssignmentScore` objects.

Study and understand the following class that will test `class StudentLoader`, as well as output that was obtained from running it.

```
public class TestAssignmentScore {

    public static void main(String[] args) {

        Assignment a1 =
            new Assignment("Primitive Types Assignment", 8);

        Assignment a2 =
            new Assignment("Selection (if)", 10, false);

        AssignmentScore a1_s1 = new AssignmentScore(a1, 12.0);
        System.out.println(a1_s1);

        AssignmentScore a1_s2 = new AssignmentScore(a1, 7.445);
        System.out.println(a1_s2);

        AssignmentScore a2_s1 = new AssignmentScore(a2, 12.0);
        System.out.println(a2_s1);

    }
}
```

```
Primitive Types Assignment: 100.0%
Primitive Types Assignment: 74.5%
Selection (if): 120.0%
```

Answer these questions before reading further:

1. Is an `AssignmentScore` object instantiated?
2. What parameters does the `AssignmentScore` constructor take?
3. What `StudentList` methods does the test code invoke? For each:
   - What parameters does the method take?
   - Does the method produce any output to the console?
   - What is the return type of the method?

Part 1:

After studying the test code, you are ready to write `class AssignmentScore`. Here are additional specifications and hints:

- Declare the necessary fields, and write a constructor to set them.
- Write getters methods: one to get the `Assignment` object and another to get the `score`.
- Write a `getPercent` method that returns the assignment score as a percentage of the maximum score possible for the assignment. For the calculation, use the `score` stored in the `AssignmentScore` object, and the `maxScore` stored in the `Assignment` object. If the `maxScore` stored in the `Assignment` object is zero, it is not possible to calculate a percent because the denominator of the division would be zero. For this case, have the method return `100.0`.
- Write a `toString` method that returns a `String` representation of the `AssignmentScore` object that will result in the output shown shown that was produced by running the `TestAssignmentScore` class.

Part 2:

First get the above working, then:

- Update the `Assignment` class to include a private `boolean` field called `strictMaxScore`. This field is to signal, when set to `true`, that a student should not receive any score on the assignment that is higher than the `maxScore` set in the `Assignment` object.
- Update the `Assignment` class to add a second constructor that takes a third parameter, the third parameter a `boolean` value used to set `strictMaxScore`. Have the original constructor set the `strictMaxScore` field to the value `true`.
- Add a method `isStrictMaxScore` that returns the value of `strictMaxScore`.
- Update the `AssignmentScore` constructor so that that it checks if the `Assignment` object's `strictMaxScore` is set to `true`, and if it is, then restricts the `score` field to less than or equal to the `Assignment` object's `maxScore` value.