Assignment #3 (4%)
Posted on Tuesday February 16, 2023
Due Date: Monday February 20, 2023; 11:59 PM
Submit to the Slate Drop box for Assignment#3 - no email submission
Late submissions 10% penalty per day up to 3 days until Thursday February 23, 2023, 11:59 PM
Then no marks after late submission deadline
Related Topics: Interfaces, the Comparable Interface, The Cloneable Interface

Design an interface named **`Colorable`** with a **`void`** method named **`howToColor()`**. Every class of a colorable object must implement the **`Colorable`** interface. Design a class named **`Square`** that extends **`GeometricObject`** and implements **`Colorable`**. Implement **`howToColor`** to display the message **`Color all four sides`**. The **`Square`** class contains a data field **`side`** with getter and setter methods, and a constructor for constructing a **`Square`** with a specified side. The **`Square`** class has a private double data field named `side` with its getter and setter methods. It has a no-arg constructor to create a **`Square`** with side 0, and another constructor that creates a **`Square`** with the specified side.

Draw a UML diagram that involves **`Colorable`**, **`Square`**, and **`GeometricObject`**. Write a test program that creates an array of five **`GeometricObject`**s. For each object in the array, display its area and invoke its **`howToColor`** method if it is colorable.

IMPORTANT HINT:
You may use following codes from the ICE_Ch13:
GeometricObject (ICE# 13.1), Circle (ICE# 13.2), Rectangle (ICE# 13.3). If needed modify their codes.
Make your test program to create an array of five GeometricObjects to be:

1. A square with side 3.5
2. A circle with radius 4.5
3. A square with side 6
4. A rectangle with width 2 and height 5
5. A square with side 5.5

Deliverable & Evaluation:
Please have your submission in one MSWord File.
2. (1 Marks) Full UML diagram for each class and display classes relationships
3. (2 Marks) Program Code, observing principles of object-oriented programming (i.e. abstraction, encapsulation, inheritance, polymorphism, interfaces, etc.). codes must have indentation and single spaced and follow submission standards, coding standard and quality.
4. (1 Marks) Program output (provide screen shot of your output). your output screen shot should provide the information required above with numerical values of 2 decimal points.

Penalty marks:
1. Penalty mark for Late submission 10% per day up to 3 days then mark of 0
2. Penalty mark for failing to follow submission standard 50%
3. submissions is electronically examined for similarity and will not be accepted in cases of high similarity.
4. No color background. Black text on while background
5. Submission order; UML, Colorable, Square, driver class

Sample Program output: