Knight Foundation School of Computing and Information Sciences

COP 3337: Computer Programming II
Assignment – V: Interfaces
Due Date & Time: Sunday, October 16, 2022, 11:59 pm
Instructor: Ahmad Waqas, PhD

Instructions:

- Create a separate java class file for each class.
- Put all the .java files in one folder (the folder name must be your name) and create the .zip file for submission.
- The general rubrics are given below:

Description	Max. Marks
Coding style Use the best practices for writing the code. The code is well organized and very easy to follow.	20%
Logic The student has used effective programming logic for solutions and demonstrates the appropriate concept in the respective task.	60%
Results The program is error-free and generates the expected results as per the specifications.	20%

Task 1:

Define an interface named `Shape` with a single method named `area` that calculates the area of the geometric shape:

```
public double area();
```

Next, define a class named `Circle` that implements `Shape`. The `Circle` class should have an instance variable for the radius, a constructor that sets the radius, accessor/mutator methods for the radius, and an implementation of the `area` method. Also, define a class named `Rectangle` that implements `Shape`. The `Rectangle` class should have instance variables for the height and width, a constructor that sets the height and width, accessor and mutator methods for the height and width, and an implementation of the `area` method.

The following test code should then output the area of the `Circle` and `Rectangle` objects:

```
public static void main(String[] args){
      Circle c = new Circle(4); // Radius of 4
      Rectangle r = new Rectangle(4,3); // Height = 4, Width = 3
      ShowArea(c);
      ShowArea(r);
}
public static void ShowArea(Shape s){
      double area = s.area();
      System.out.println("The area of the shape is " + area);}
```

The sample output is shown in the given screenshot.

```
The area of the shape is 50.26544
The area of the shape is 12.0
```

Task 2:

Define a class named `Author` that implements the `Comparable` interface and contains the following instance variables:

```
String firstName, lastName, bookTitle;
```

Since the `Author` class implements the `Comparable` interface, it will override the `CompareTo` method that will sort the authors by `lastName`, and if the two authors have the same last name, they will be sorted by `firstName`. And, if they have the same last and first name, the list will be sorted by `bookTitle`.

Test the author class in the main program and create an array or array list of at least 5 authors and display them first without sorting and then display the list after calling the Collection.sort method.

The sample output is shown in the given screenshot.

```
Henry, Miller; Tropic of Cancer
Nalo, Hopkinson; Brown Girl in the Ring
Frank, Miller; 300
Deborah, Hopkinson; Sky Boys
George R. R., Martin; Song of Ice and Fire

*** After sorting ***

Deborah, Hopkinson; Sky Boys
Nalo, Hopkinson; Brown Girl in the Ring
George R. R., Martin; Song of Ice and Fire
Frank, Miller; 300
Henry, Miller; Tropic of Cancer
```