Homework: Abstraction, interfaces, polymorphism **Date:** 02/15/2023

Follow the directions step-by-step, so you can see some errors that Java will throw when the methods are not defined correctly in your classes. Add to GitHub all your java files and a document (doc, txt) with the answer to questions 3, 4, 5, 7 and 8.

1. Convert the superclass Shape from yesterday's homework to an abstract class.
2. Use polymorphism to create your objects. You may have a Driver class to create your objects and call the methods.
   ```
   Shape s = new Rectangle(2, 4)
   ```
3. You have the method getArea() in your Rectangle class, try to call the method to calculate the area:
   ```
   s.getArea();
   ```
   Did you get an error? Why?
4. Create an abstract method getArea() in your Shape class and try again.
   What happened?
5. Try to create an instance from the abstract class Shape.
   Did you get an error? Why?
6. Let's add an interface on top of the abstract Shape and let's implement ShapeInterface in our abstract Shape class

   ```
   // interface
   interface ShapeInterface {
      double getHeight();
      double getWidth();
      void info();
      double getArea();
   }

   // abstract class implements your interface
   abstract class Shape implements ShapeInterface{
   }
   ```

7. Now with the abstract class implementing the interface, try to remove the methods getArea() from the classes Shape and Rectangle, and try the following:
   ```
   Rectangle r = new Rectangle(2, 4)
   r.getArea()
   ```
   What happened?

8. Let's restore the getArea() method in your Rectangle class (do not restore the method in your Shape class yet).

Does it work? Why?