

General guidelines:

- 1- This assignment can be done individually or in groups. Maximum group size is 3 persons per group.
- 2- Due date is February 7, 2021.
- 3- You have to upload the source files and screenshots of the outputs to the blackboard.
- 4- I prefer that you use Java to solve the assignments. If you do not like Java, then you can use C#..

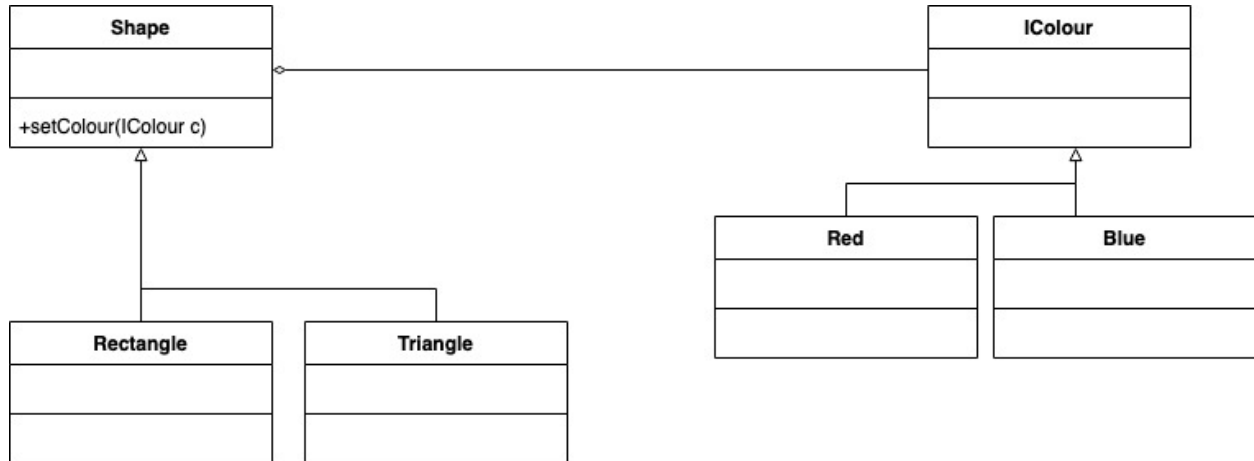
Question#1:

In the GoF book, List interface is defined as follows:

```
interface List {  
    int count(); //return the current number of elements in the list  
    Object get(int index); //return the object at the index in the list  
    Object first(); //return the first object in the list  
    Object last(); //return the last object in the list  
    boolean include(Object obj); //return true is the object in the list  
    void append(Object obj); //append the object to the end of the list  
    void prepend(Object obj); //insert the object to the front of the list  
    void delete(Object obj); //remove the object from the list  
    void deleteLast(); //remove the last element of the list  
    void deleteFirst(); //remove the first element of the list  
    void deleteAll(); //remove all elements of the list  
}
```

- (a) Write a class adapter to adapt Java ArrayList to GoF List interface.
- (b) Write a main program to test your adapters through List interface.
- (c) Same requirement as (a) and (b), but write an object adapter to adapt Java ArrayList to GoF List interface.

Question#2: In class, we studied that Bridge pattern and discussed the Shape example below. Implement this example in Java. Assume that we want to control the thickness of Shape, i.e., to set the thickness to either *Thick* or *Thin*. Add a getter method for Colour and getters and setters to control the thickness. Then write a main method, in which you instantiate four instances of Shape, 2 Rectangle and 2 Triangle: a red thick and a blue thin triangle, and a red thin and a blue thick rectangle. Call the toString() method on all instances to print their colour and thickness.



Question#3: Design and implement a Java program using Abstract Factory and Singleton design patterns.

The program displays date and time in one of the following two formats:

Format 1:

Date: MM/DD/YYYY

Time: HH:MM:SS

Format 2:

Date: DD-MM-YYYY

Time: SS,MM,HH

The following is how the program works. In the beginning, the program asks the user what display format that she wants. Then the program continuously asks the user to give one of the following commands, and performs the corresponding task. Note that the program gets the current date and time from the system clock (use the appropriate Java date and time operations for this).

'd' : display current date

't': display current time

'q': quit the program.

- In the program, there should be 2 product hierarchies: “DateObject” and “TimeObject”. Each hierarchy should have format1 and format2 described above.
- Implement the concrete factories as singleton classes.
- Draw a UML class diagram for the program.