Intro to Java Week 5 Coding Assignment

Category	Criteria	% of Grade
Functionality	Does the code work?	25
Organization	Is the code clean and organized? Proper use of white space, syntax, and consistency are utilized. Names and comments are concise and clear.	25
Creativity	Student solved the problems presented in the assignment using creativity and out of the box thinking.	25
Completeness	All requirements of the assignment are complete.	25

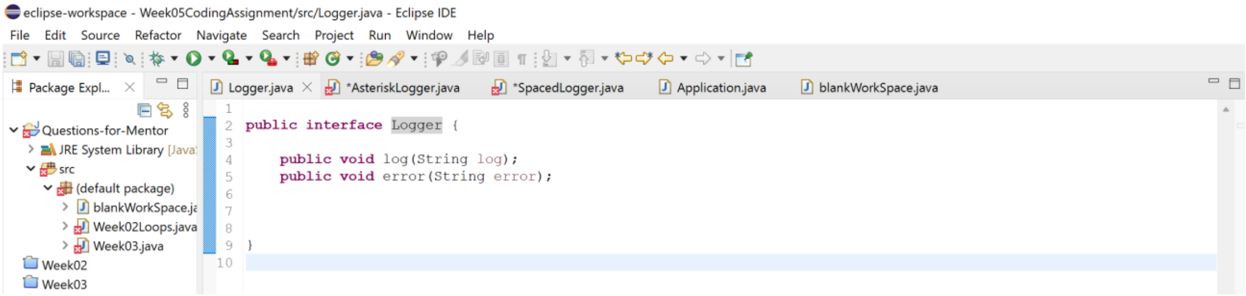
Instructions: In Eclipse, or an IDE of your choice, write the code that accomplishes the objectives listed below. Ensure that the code compiles and runs as directed. Take screenshots of the code and of the running program (make sure to get screenshots of all required functionality) and paste them in this document where instructed below. Create a new repository on GitHub for this week's assignments and push this document, with your Java project code, to the repository. Add the URL for this week's repository to this document where instructed & submit this document to your instructor.

Coding Steps:

1. Create an interface named Logger.

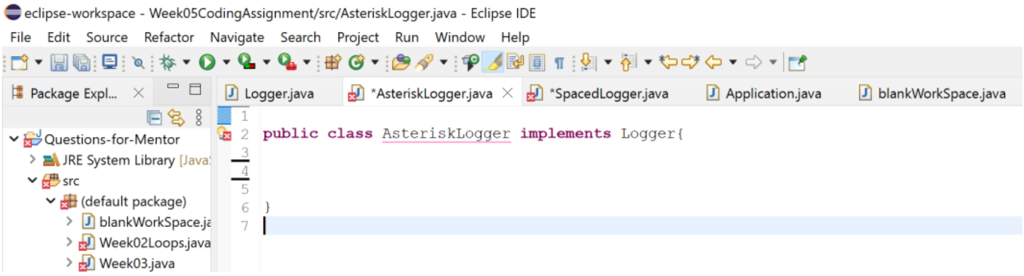
   (see below)

2. Add two void methods to the Logger interface, each should take a String as an argument
   a. Log
   b. Error

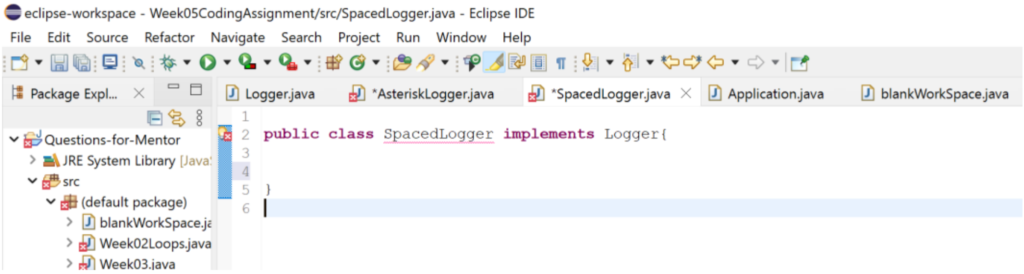


```
public interface Logger {

	public void log(String log);
	public void error(String error);




}
```

3. Create two classes that implement the Logger interface
   a. AsteriskLogger
   b. SpacedLogger



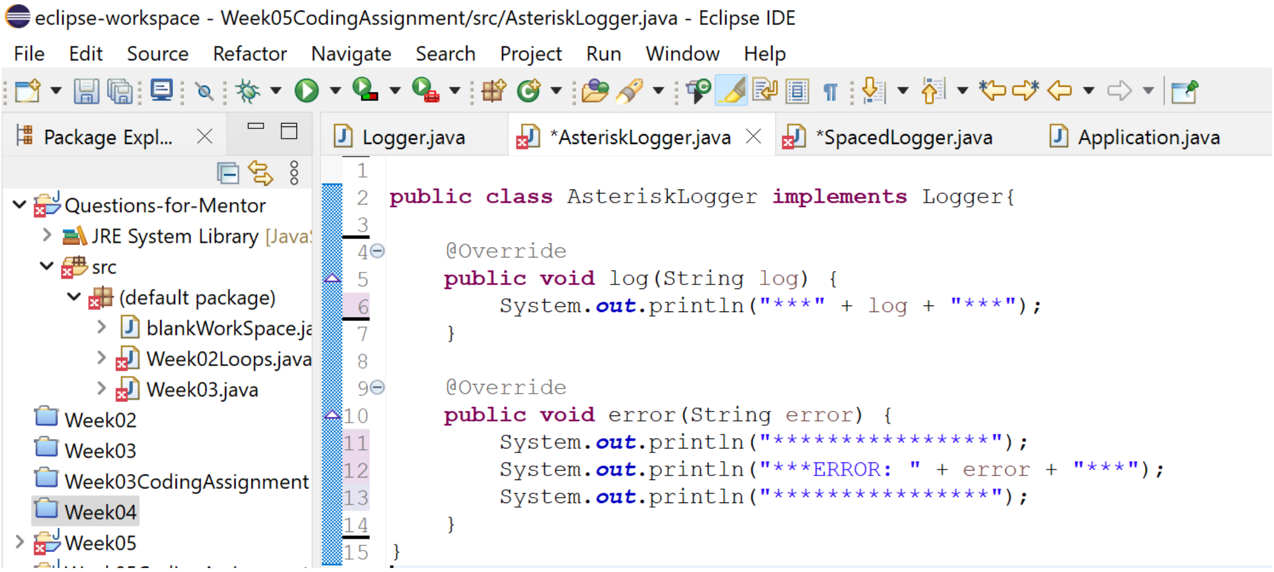
```
public class AsteriskLogger implements Logger{



}
```



```
public class SpacedLogger implements Logger{


}
```

4. The log method on the AsteriskLogger should print out the String it receives between 3 asterisks on either side of the String. (see below)

5. The error method on the AsteriskLogger should print the String it receives inside a box of asterisks, with the String preceded by the word "ERROR:".

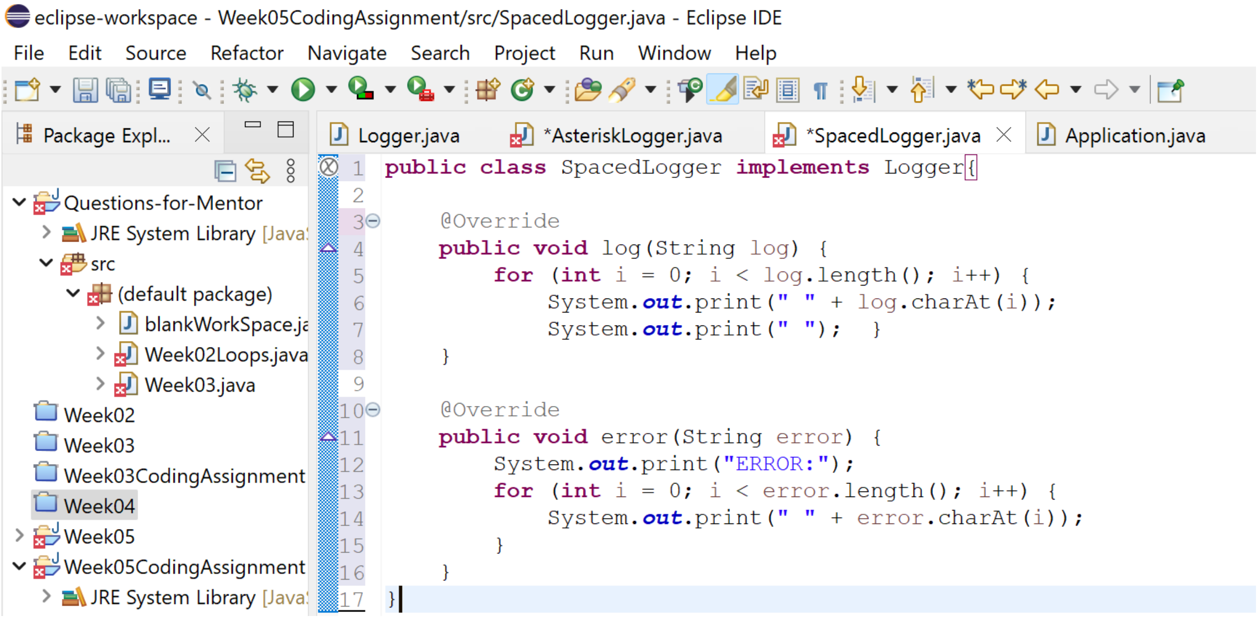


```java

public class AsteriskLogger implements Logger{

	@Override
	public void log(String log) {
		System.out.println("***" + log + "***");
	}

	@Override
	public void error(String error) {
		System.out.println("****************");
		System.out.println("***ERROR: " + error + "***");
		System.out.println("****************");
	}
}
```

6. The SpacedLogger should add spaces between each character of the String argument passed into its methods. (see below)

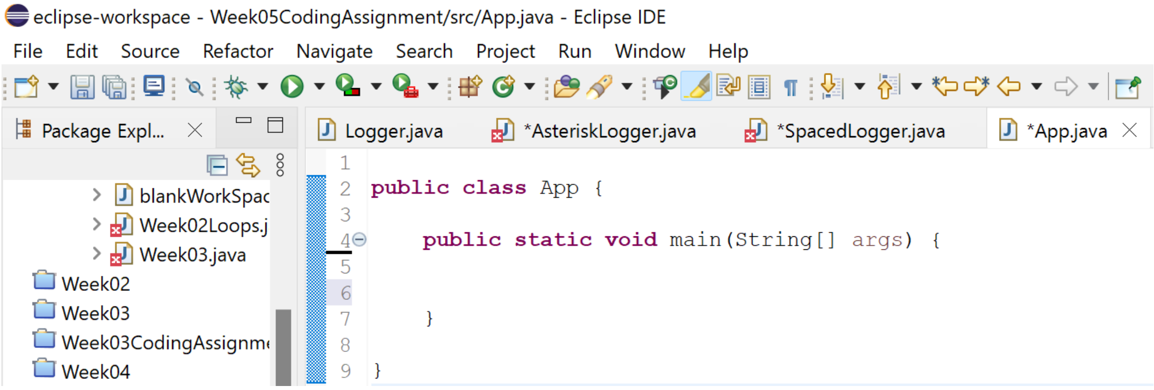
7. If the log method received "Hello" as an argument, it should print H e l l o. (see below)

8. The error method should do the same, but with "ERROR:" preceding the spaced-out input.



```java
public class SpacedLogger implements Logger{

	@Override
	public void log(String log) {
		for (int i = 0; i < log.length(); i++) {
			System.out.print(" " + log.charAt(i));
			System.out.print(" ");   }
	}

	@Override
	public void error(String error) {
		System.out.print("ERROR:");
		for (int i = 0; i < error.length(); i++) {
			System.out.print(" " + error.charAt(i));
		}
	}
}
```

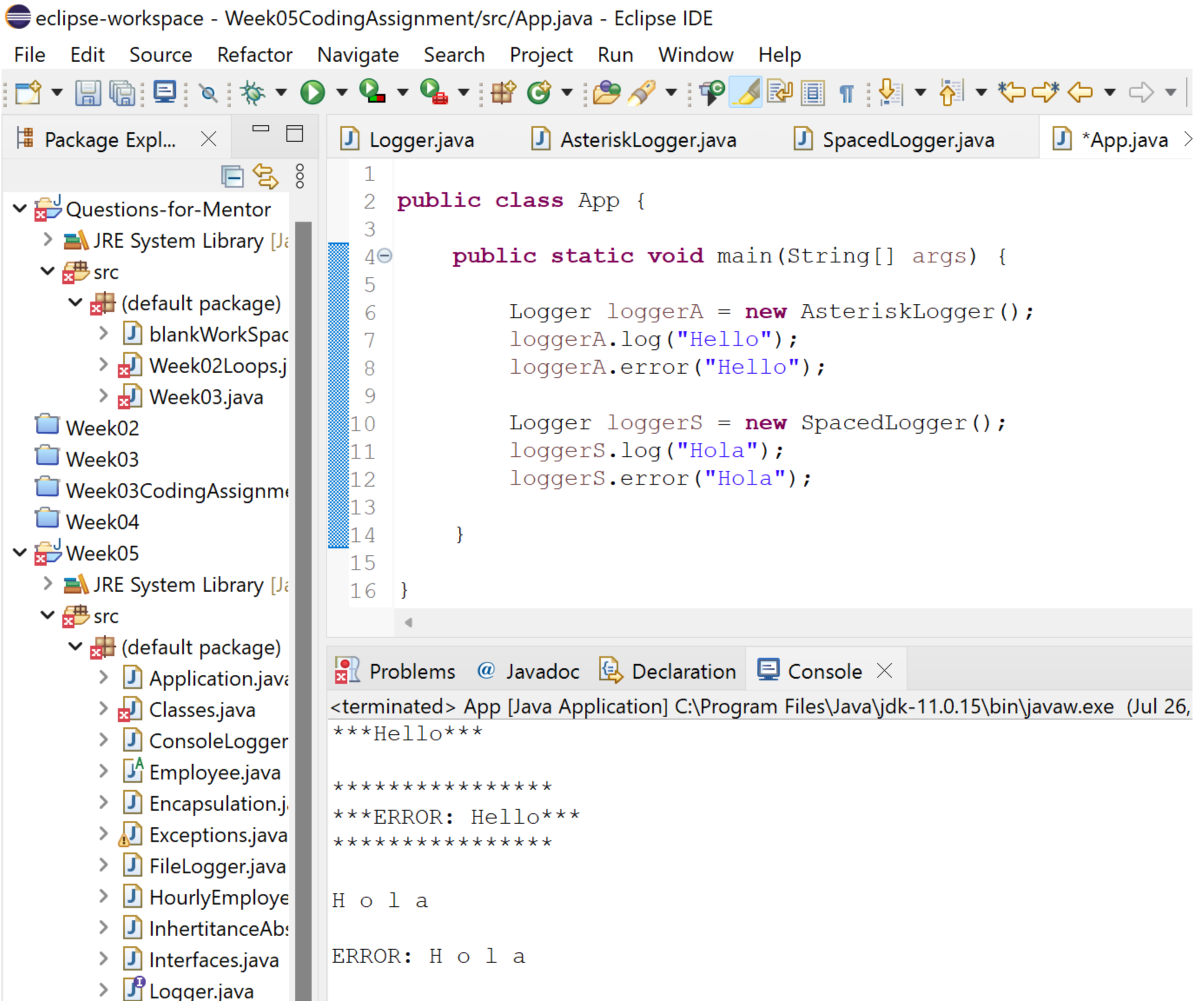
9. Create a class named App that has a main method.



```java

public class App {

    public static void main(String[] args) {


    }
}
```

10. In this class instantiate an instance of each of your logger classes that implement the Logger interface.
(see below)

11. Test both methods on both instances, passing in Strings of your choice.



```java

public class App {

    public static void main(String[] args) {

        Logger loggerA = new AsteriskLogger();
        loggerA.log("Hello");
        loggerA.error("Hello");

        Logger loggerS = new SpacedLogger();
        loggerS.log("Hola");
        loggerS.error("Hola");

    }
}
```

<terminated> App [Java Application] C:\Program Files\Java\jdk-11.0.15\bin\javaw.exe (Jul 26,

```
***Hello***

****************
***ERROR: Hello***
****************

H o l a

ERROR: H o l a
```

URL to GitHub Repository:

https://github.com/JaxYoungblood/Week05CodingAssignment.git