Homework 8

Use the skeleton code provided on AsULearn to get started. You will have to create a BlueJ project for this homework. Don't forget to add the tests to your project and run them for feedback and insights on errors. Submit your final, compiled code to AsULearn by the due date.

SumOfNumbers class

Write a program in the **SumOfNumbers** class that:

- asks the user for a positive nonzero integer value
- use a loop to get the sum of all the integers from 1 up to the number entered
- Uses the exact prompts found in the example below

Note: Do not accept a number less than one.

For example, if the user enters 50, the loop will find the sum of 1, 2, 3, 4, . . . , 50. A run of this program with valid input would look like this.

Please enter a positive integer:
12
78

A run of this program with invalid input first would look like this.

Please enter a positive integer:
-1
That is not a positive integer!
Please enter a positive integer:
6
21

DistanceTraveled class

Write a program in the **DistanceTraveled** class that:

- stores the speed of a vehicle (in miles per hour) in a variable called **speed**
- stores the number of hours it has traveled in a variable called **hours**
- has a method named **getDistance** that returns the distance, in miles, that the vehicle has traveled

Note: The distance a vehicle travels can be calculated as follows:

Distance = Speed * Time

For example, if a train travels 40 miles per hour (mph) for three hours, the distance traveled is 120 miles.

Driver class

Demonstrate the DistanceTraveled class by writing a program in the **Driver** class that:

- uses a loop to display the distance a vehicle has traveled for each hour of a time period specified by the user
- use the exact prompts found below
    - the header is printed using the following printf statement
      System.out.printf("\n%-8s%-17s\n", "Hour", "Distance Traveled");
    - the rows are printed using the following format string in a printf statement
      "%-8d%-8d\n"

For example, if a vehicle is traveling at 40 mph for a three-hour time period (per user input), it should display a report as shown here.

```
Please enter the speed: 40
Please enter the time in hours: 3
Hour    Distance Traveled
1       40
2       80
3       120
```

A run of this program with invalid input should look like this.

```
Please enter the speed: -40
Cannot enter negative speed.
Please enter the speed: 40
Please enter the time in hours: -3
Cannot enter less than one hour.
Please enter the time in hours: 3
Hour    Distance Traveled
1       40
2       80
3       120
```

Note: Use input validation. Do not accept a negative number for speed, and do not accept any value less than one for the time traveled.

Please complete the Javadoc comments and add any comments you believe are necessary to determine what the program is doing.

Make sure that your code compiles! **If your code does not compile, then you will receive a 0 on this assignment.**