

```
1
2 import java.io.IOException;
3 import java.io.BufferedReader;
4 import java.io.InputStreamReader;
5 import java.text.NumberFormat;
6
7 /**
8  * Calculates the mean of a set of non-negative marks.
9  *
10 * @author M. Arkin
11 * @version 1.0 2012-10-14
12 */
13 public class MarksMeanWithUtilityMethods
14 {
15     // global variables
16     private static BufferedReader keyboard;
17
18     public static void main(String[] argument) throws IOException
19     {
20         // Establish a connection to the keyboard.
21         keyboard = new BufferedReader(new InputStreamReader(System.in));
22
23         // Welcome and instruct user.
24         System.out.println("Welcome to the Marks Mean Machine!"
25             + "\n\nEnter as many non-negative marks as you wish."
26             + "\nWhen you're done, enter any negative number,"
27             + "\nthen I'll calculate and display the mean of "
28             + "\nthe marks you entered.");
29
30         // Initialize the accumulator and counter.
31         double marksSum = 0;
32         int marksCount = 0;
33
34         // Get the first mark.
35         int markNumber = 1;
36         double mark = validDouble(markNumber);
37
38         // Accumulate and count all valid marks until
39         // a negative mark is entered.
40         while (mark >= 0)
41         {
42             // If we got this far, the mark entered was
43             // both valid and non-negative.
44             marksSum = marksSum + mark;
45             marksCount = marksCount + 1;
46
47             // Get the next mark.
48             markNumber = markNumber + 1;
49             mark = validDouble(markNumber);
50         } // end of while (mark >= 0)
51
52         // Calculate the mean.
53         double mean = 0;
54         if (marksCount > 0 )
55         {
56             mean = marksSum / marksCount;
57         } // end of if (marksCount > 0)
58
59         // Establish a number formatter and set its precision.
60         NumberFormat formatter;
61         formatter = NumberFormat.getNumberInstance();
```

```
62     formatter.setMaximumFractionDigits(1);
63     formatter.setMinimumFractionDigits(1);
64
65     // Report results.
66     System.out.println("\nnumber of marks entered: "
67         + marksCount
68         + "\n mean of marks entered: "
69         + formatter.format(mean));
70
71     // Thank user.
72     System.out.println("\nThanks for using the Marks Mean "
73         + "Machine!");
74
75 } // end of method main(String[] argument) throws IOException
76
77 private static double validDouble(int promptInteger) throws IOException
78 {
79     // Initialize to satisfy the compiler.
80     double validMark = Double.MIN_VALUE;
81
82     boolean inputWasValidDouble = false;
83
84     do
85     {
86         System.out.print("\nMark #" + promptInteger + ": ");
87         String input = keyboard.readLine();
88         try
89         {
90             validMark = Double.parseDouble(input);
91             inputWasValidDouble = true;
92         }
93         catch (NumberFormatException exception)
94         {
95             System.out.println("\nError! Invalid input!"
96                 + "\"" + input + "\""
97                 + "is not a valid mark."
98                 + "\nPlease retry.");
99         } // end of catch (NumberFormatException exception)
100     } while (!inputWasValidDouble);
101
102     return validMark;
103 } // end of method validDouble() throws IOException
104
105 } // end of class MarksMeanWithUtilityMethods
```