

```
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 MarksMean
14 {
15     public static void main(String[] argument) throws IOException
16     {
17         // Establish a connection to the keyboard.
18         BufferedReader keyboard =
19             new BufferedReader(new InputStreamReader(System.in));
20
21         // Welcome and instruct user.
22         System.out.println("Welcome to the Marks Mean Machine!"
23             + "\n\nEnter as many non-negative marks as you wish."
24             + "\nWhen you're done, enter any negative number,"
25             + "\nthen I'll calculate and display the mean of "
26             + "\nthe marks you entered.");
27
28         // Initialize the accumulator and counter.
29         // Initialize the mark to satisfy the compiler.
30         double marksSum = 0;
31         int marksCount = 0;
32         double mark = Double.MIN_VALUE;
33
34         // Get the first mark.
35         int markNumber = 1;
36         boolean inputWasValidDouble = false;
37
38         do
39         {
40             System.out.print("\nMark #" + markNumber + ": ");
41             String input = keyboard.readLine();
42             try
43             {
44                 mark = Double.parseDouble(input);
45                 inputWasValidDouble = true;
46             }
47             catch (NumberFormatException exception)
48             {
49                 System.out.println("\nError! Invalid input!"
50                     + "\"" + input + "\""
51                     + "is not a valid mark."
52                     + "\nPlease retry.");
53             } // end of catch (NumberFormatException exception)
54         } while (!inputWasValidDouble);
55
56         // Accumulate and count all valid marks until
57         // a negative mark is entered.
58         while (mark >= 0)
59         {
60             // If we got this far, the mark entered was
61             // both valid and non-negative.
```

```
62         marksSum = marksSum + mark;
63         marksCount = marksCount + 1;
64
65         // Get the next mark.
66         markNumber = markNumber + 1;
67         inputWasValidDouble = false;
68
69         do
70         {
71             System.out.print("\nMark #" + markNumber + ": ");
72             String input = keyboard.readLine();
73
74             try
75             {
76                 mark = Double.parseDouble(input);
77                 inputWasValidDouble = true;
78             }
79             catch (NumberFormatException exception)
80             {
81                 System.out.println("\nError! Invalid input!"
82                                     + "\"" + input + "\""
83                                     + "is not a valid integer."
84                                     + "\nPlease retry.");
85             } // end of catch (NumberFormatException exception)
86         } while (!inputWasValidDouble);
87     } // end of while (mark >= 0)
88
89     // Calculate the mean.
90     double mean = 0;
91     if (marksCount > 0 )
92     {
93         mean = marksSum / marksCount;
94     } // end of if (marksCount > 0)
95
96     // Establish a number formatter and set its precision.
97     NumberFormat formatter;
98     formatter = NumberFormat.getNumberInstance();
99     formatter.setMaximumFractionDigits(1);
100    formatter.setMinimumFractionDigits(1);
101
102    // Report results.
103    System.out.println("\nnumber of marks entered: "
104                      + marksCount
105                      + "\n mean of marks entered: "
106                      + formatter.format(mean));
107
108    // Thank user.
109    System.out.println("\nThanks for using the Marks Mean "
110                      + "Machine!");
111
112    } // end of method main(String[] argument) throws IOException
113 } // end of class MarksMean
```