```
1   /*-------------------------------------------------------------------------------------------------------
2    * Author:      Dan Cassidy
3    * Date:        2015-07-10
4    * Assignment:  HW3-2
5    * Source File: CalculateAverage.java
6    * Language:    Java
7    * Course:      CSCI-C 490, Android Programming, MoWe 08:00
8   -------------------------------------------------------------------------------------------------------*/
9   import java.util.Scanner;
10
11  /**
12   * A small class to calculate the average of a given number of integers.
13   *
14   * @author Dan Cassidy
15   */
16  public class CalculateAverage
17  {
18
19      public static void main(String[] args)
20      {
21          int numberOfNumbers = 0;
22          boolean valid = false;
23          Scanner consoleInput = new Scanner(System.in);
24
25          // Loop while input is not valid.
26          while (!valid)
27          {
28              try
29              {
30                  numberOfNumbers = readInt("Please enter the number of numbers to average: ");
31                  if (numberOfNumbers <= 0)
32                      throw new Exception("Number must be greater than 0.");
33                  valid = true;
34              }
35              catch (Exception ex)
36              {
37                  System.out.println(ex.getMessage());
38              }
39          }
40
41          // Declare an array of the specified size and then ask for input for all elements.
42          int[] numbers = new int[numberOfNumbers];
43          for (int counter = 0; counter < numbers.length; counter++)
44              numbers[counter] = readInt("Please input a number for entry " + (counter + 1) + ": ");
45
46          System.out.println("Average of all entries: " + average(numbers));
47      }
48
49      /**
50       * Computes the average (arithmetic mean) of an array of numbers. If <b>numbers</b> is null or
51       * an empty array, 0 is returned.
52       *
53       * @param numbers An array of integers, from which their average will be computed.
54       * @return double, representing the average of the elements contained in <b>numbers</b>.
55       */
56      public static double average(int[] numbers)
57      {
58          // No need to throw an error, just return 0 if the argument is no good.
59          if (numbers == null || numbers.length == 0)
60              return 0;
```

```
61
62          // Compute and return the average.
63          double sum = 0;
64          for (int number : numbers)
65              sum += number;
66          return sum / numbers.length;
67      }
68
69      /**
70       * Reads an integer from the console.
71       *
72       * @param prompt A String object containing the prompt text for a user entering a number.
73       * @return int, holding the integer read from the console.
74       */
75      public static int readInt(String prompt)
76      {
77          int number = 0;
78          boolean valid = false;
79          Scanner consoleInput = new Scanner(System.in);
80
81          // Loop while input is not valid.
82          while (!valid)
83          {
84              try
85              {
86                  System.out.print(prompt);
87                  number = Integer.parseInt(consoleInput.nextLine());
88                  valid = true;
89              }
90              catch (NumberFormatException ex)
91              {
92                  System.out.println("Invalid input, please try again.");
93              }
94          }
95
96          return number;
97      }
98  }
99
```