

```
1 import components.simplereader.SimpleReader;
5
6 /**
7  * In this project, I ask the user if they want to calculate the
   square root of
8  * a number and calculate the number within an error of 0.01.
   After some
9  * calculations, the result is printed out excluding the number
   0.
10 *
11 * @author Mohamed Jama
12 *
13 */
14 public final class Newton3 {
15
16     /**
17      * No argument constructor--private to prevent
   instantiation.
18      */
19     private Newton3() {
20     }
21
22     /**
23      * Put a short phrase describing the static method Newton1
   here.
24      */
25     /**
26      * Computes estimate of square root of x to within relative
   error number
27      * asked by the user.
28      *
29      * @param x
30      * @param epsilon
31      *         positive number to compute square root of
32      * @return estimate of square root with epsilon given by
   user.
33      */
34     private static double sqrt(double x, double epsilon) {
35         double r = x;
```

```

36     double e = epsilon;
37     if (r != 0) {
38         while (Math.abs(r * r - x) / x - (e * e) > 0) {
39             r = (r + x / r) / 2;
40         }
41     }
42     return r;
43
44 }
45
46 /**
47  * Main method.
48  *
49  * @param args
50  *         the command line arguments
51  */
52 public static void main(String[] args) {
53     SimpleReader in = new SimpleReader1L();
54     SimpleWriter out = new SimpleWriter1L();
55     double number;
56     double epsilon;
57     String con;
58     out.println("Would you like to proceed and calculate?(y/
n): ");
59     con = in.nextLine();
60
61     while (con.equals("y")) {
62         out.println("Enter a positive double number: ");
63         number = in.nextDouble();
64         out.println("Enter a value of  $\epsilon$ : ");
65         epsilon = in.nextDouble();
66         if ((number - 0) <= 0) {
67             out.println("Undefined");
68         }
69         double result = sqrt(number, epsilon);
70         if ((result - 0) > 0) {
71             out.println("The square root of the number " +
number
72                 + " within a relative error of " +

```

```
    epsilon + " is "
73         + result);
74         out.println("Goodbye");
75     }
76 }
77 /*
78  * Close input and output streams
79  */
80
81 out.println("Goodbye");
82 in.close();
83 out.close();
84 }
85 }
86
```