```java
import components.simplereader.SimpleReader;

/**
 * A program that computes the square root of a number using Newton Iteration
 *
 * @author VishalKumar
 *
 */
public final class Newton4 {

    /**
     * Private constructor so this utility class cannot be instantiated.
     */
    private Newton4() {
    }

    /**
     * Computes estimate of square root of x to within relative error 0.01%.
     *
     * @param x
     *            positive number (or 0) to compute square root of
     * @return estimate of square root
     */
    private static double sqrt(double x, double epsilon) {
        double r = x;
        // exception for when user wants to calculate the root of 0
        if (x == 0.0) {
            return 0.0;
        } else {
            while (!(((r * r) - x) / x < (epsilon * epsilon))) {
                r = ((r + (x / r)) / 2);
            }
        }
        return r;
    }

    /**
     * Main method.
     *
     * @param args
     *            the command line arguments
     */
    public static void main(String[] args) {
        SimpleReader in = new SimpleReader1L();
        SimpleWriter out = new SimpleWriter1L();

        // boolean to store whether or not user wants to do another round
        boolean another = false;

        // compute if user would like to calculate another square root
        out.print("Enter positive double (#.##): ");
        double num = in.nextDouble();

        if (num > 0.0) {
            another = !another;
        }
        // loop until user wants to calculate the root of a negative number
```

```
61          while (another) {
62              // get desired epsilon from user
63              out.print("Enter desired epsilon: ");
64              double epsilon = in.nextDouble();
65              double rootNum = sqrt(num, epsilon);
66              out.println("The square root of " + num + " is " + rootNum);
67
68              out.print("Enter positive double (#.##): ");
69              num = in.nextDouble();
70
71              if (num < 0.0) {
72                  another = !another;
73              }
74
75          }
76          out.println("Goodbye!");
77          /*
78           * Close input and output streams
79           */
80          in.close();
81          out.close();
82      }
83
84  }
85
```