```
import components.simplereader.SimpleReader;
import components.simplereader.SimpleReader1L;
import components.simplewriter.SimpleWriter;
import components.simplewriter.SimpleWriter1L;

/**
 * Project #2: Compute Roots Using Newton's Iteration.
 *
 * @author Danny Kan (kan.74@osu.edu)
 * @version 01272022
 */
public final class Newton1 {

  /**
   * Private constructor so this utility class cannot be instantiated.
   */
  private Newton1() {
  }

  /**
   * Computes estimate of square root of x to within relative error 0.01%.
   *
   * @param x
   *          a positive number to compute the square root of.
   * @return the estimate of the square root.
   */
  private static double sqrt(double x) {
    double r = x; // estimate of the square root of x.
    final double ERROR = 0.0001; // relative error: 0.01%
    while ((Math.abs(Math.pow(r, 2) - x)) / x >= Math.pow(ERROR, 2)) {
      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();
```

```
        out.println("Project #2: Compute Roots Using Newton Iteration");

        // assume the user DOES NOT enter a negative value.
        out.print(
              "Enter a positive value of type double to calculate the square root: ");
        double userVal = Double.parseDouble(in.nextLine());

        // method call:
        out.print(sqrt(userVal) + "\n");

        boolean valid = true;
        while (valid) {
           out.print(
                 "Type \"Y\" or \"y\" if you wish to calculate another square root: ");
           String userInput = in.nextLine();
           if (userInput.equals("Y") || userInput.equals("y")) {
              out.print(
                    "Enter a positive value of type double to calculate the square root: ");
              userVal = Double.parseDouble(in.nextLine());

              // method call:
              out.print(sqrt(userVal) + "\n");
           } else {
              valid = false;
           }
        }

        // close input and output streams.
        in.close();
        out.close();
     }
}
```