

```
1 import java.io.BufferedReader;
2 import java.io.BufferedWriter;
3 import java.io.FileReader;
4 import java.io.FileWriter;
5 import java.io.IOException;
6 import java.io.PrintWriter;
7
8 import components.simplereader.SimpleReader;
9 import components.simplereader.SimpleReader1L;
10 import components.simplewriter.SimpleWriter;
11 import components.simplewriter.SimpleWriter1L;
12
13 /**
14  * {@code Queue} represented as a {@code Sequence} of entries,
15  * with
16  * implementations of primary methods.
17  *
18  * @param <T>
19  *         type of {@code Queue} entries
20  * @correspondence this = $this.entries
21  */
22
23 public class HelloWorld {
24
25     //1.
26     public static void main(String[] args) {
27
28         // TODO - fill in body
29         SimpleReader in = new SimpleReader1L(args[0]);
30         SimpleWriter out = new SimpleWriter1L(args[1]);
31
32         while (!in.atEOS()) {
33             String temp = in.nextLine();
34             out.println();
35         }
36
37         in.close();
38         out.close();
39     }
40
41     //2.
42     public static void main(String[] args) throws IOException {
43
44         // TODO - fill in body
45         BufferedReader input = new BufferedReader(new
```

```
    FileReader(args[0]));
44    PrintWriter output = new PrintWriter(
45        new BufferedWriter(new FileWriter(args[1])));
46
47    String s = input.readLine();
48    while (s != null) {
49        output.println(s);
50        s = input.readLine();
51    }
52
53    input.close();
54    output.close();
55 }
56
57 //3.
58 public static void main(String[] args) {
59
60     // TODO - fill in body
61     BufferedReader input;
62     PrintWriter output;
63     try {
64         input = new BufferedReader(new FileReader(args[0]));
65         output = new PrintWriter(
66             new BufferedWriter(new
67                 FileWriter(args[1])));
68         catch (IOException e) {
69             System.err.println("Error when opening the file");
70             return;
71         }
72         try {
73             String s = input.readLine();
74             while (s != null) {
75                 output.println(s);
76                 s = input.readLine();
77             }
78         } catch (IOException e) {
79             System.err.println(
80                 "Error when reading from the file or writing
81                 to the file");
82         }
83         try {
84             input.close();
```

```
85         output.close();
86     } catch (IOException e) {
87         System.err.println("Error when closing the file");
88     }
89 }
90
91 }
```