Homework Turnin

Name:	Xuqing Wu
Account:	xw88 (xw88@uw.edu)
Student ID:	1933202
Section:	AS
Course:	CSE 142 19au
Assignment:	a6
Receipt ID:	bd1b6444b73941f8485442087cd8c976

Turnin Successful!

The following file(s) were received:

MadLibs.java (4164 bytes, sha256: 8bf0e9e8ece7c23436a027e1f85d3281)

```
1.  // Xuqing Wu
2.  // 11/11/2019
3.  // CSE142
4.  // TA: Ethan M Knutson
5.  // Assignment #6
6.  //
7.  // This program will show a game madlib, users can type the name of files for
8.  // input and output. They will then type words to replace placeholders in the
9.  // input file. Users can create, view the madlib or quit the game.
10.
11. import java.io.*;
12. import java.util.*;
13.
14. public class MadLibs{
15.    public static void main(String[] args) throws FileNotFoundException{
16.       Scanner console = new Scanner(System.in);
17.       intro();
18.       String choice = "";
19.       while(!choice.equals("q")){
20.          System.out.print("(C)reate mad-lib, (V)iew mad-lib, (Q)uit? ");
21.          choice = console.nextLine().toLowerCase();
22.          if(choice.equals("c")){
23.             File file = inputFile(console);
24.             PrintStream output = outputFile(console);
25.             replace(file, output, console);
26.          }
27.          else if(choice.equals("v")){
28.             File file = inputFile(console);
29.             view(file, console);
30.          }
31.       }
32.    }
33.
34.    //type the introduction of the game
35.    private static void intro(){
36.       System.out.println("Welcome to the game of Mad Libs.");
37.       System.out.println("I will ask you to provide various words");
38.       System.out.println("and phrases to fill in a story.");
39.       System.out.println("The result will be written to an output file.");
40.       System.out.println();
41.    }
42.
43.    //let user type file names until the file exists and return file
```

```
44.     //Scanner console - Scanner to get the name of input file
45.     public static File inputFile(Scanner console){
46.        System.out.print("Input file name: ");
47.        File file = new File(console.nextLine());
48.        while (!file.exists()){
49.           System.out.print("File not found. Try again: ");
50.           file = new File(console.nextLine());
51.        }
52.        return file;
53.     }
54.
55.     //let user type output file name and return PrintStream output - the output file
56.     //Scanner console - Scanner to get the name of output file
57.     public static PrintStream outputFile(Scanner console) throws FileNotFoundException{
58.        System.out.print("Output file name: ");
59.        String outputFile = console.nextLine();
60.        PrintStream output = new PrintStream(new File(outputFile));
61.        System.out.println();
62.        return output;
63.     }
64.
65.     //let user type words to replace placeholders and saves them to the output file
66.     //File file - input file
67.     //PrintStream output - output file
68.     //Scanner console - Scanner to get the input file content
69.     public static void replace(File file, PrintStream output, Scanner console) throws
70.     FileNotFoundException{
71.        Scanner input = new Scanner(file);
72.        while(input.hasNextLine()){
73.           String line = input.nextLine();
74.           Scanner lineScan = new Scanner(line);
75.           while(lineScan.hasNext()){
76.              String word = lineScan.next();
77.              if(word.startsWith("<") && word.endsWith(">")){
78.                 word = word.substring(1, word.length() - 1);
79.                 word = word.replace("-", " ");
80.                 String first = word.toLowerCase();
81.                 if(first.startsWith("a") || first.startsWith("e") || first.startsWith("i")
82.                    || first.startsWith("o") || first.startsWith("u")){
83.                    System.out.print("Please type an " + word + ": ");
84.                 }
85.                 else{
86.                    System.out.print("Please type a " + word + ": ");
87.                 }
88.                 String newWord = console.nextLine();
89.                 output.print(newWord + " ");
90.              }
91.              else{
92.                 output.print(word + " ");
93.              }
94.           }
95.           output.println();
96.        }
97.        System.out.println("Your mad-lib has been created!");
98.        System.out.println();
99.     }
100.
101.    //show the created mad-lib
102.    //File file - input file
103.    //Scanner console - Scanner to get the viewed file
104.    public static void view(File file, Scanner console) throws FileNotFoundException{
105.       System.out.println();
106.       console = new Scanner(file);
107.       while (console.hasNextLine()){
108.          String row = console.nextLine();
109.          System.out.println(row);
110.       }
111.       System.out.println();
112.    }
113. }
```

mymadlib.txt (579 bytes, sha256: ac119dbf2c29d2edb8f3a55b66b8f2d9)

Once upon a time there was a <adjective> boy called <Male-Name> .
He fell love with a girl called <Female-Name> . He loved her so
much that he would do everything for her. However, the girl only
treated him as a <noun> and did not know his love. On one <adjective>
day, he decided to make an expression. Unfortunately, he was refused
because he was so poor that he can not buy <Property> for the girl.
What a <adjective> feeling. The girl rarely contact him since that day.
His world turns <Color!> and he uses <plural-noun> to replace the girl.
What a <adjective> story.