```
 1: #!/bin/bash
 2:
 3: # AUTHOR: Ravi S. Ramphal
 4: # CLASS:  CCSF CS111B
 5: # DATE:   2017.06.29
 6: #
 7: # These lists were created from scraping the lists of palindromes
 8: # available at http://www.palindromelist.net/. It was manually sanitized
 9: # to remove the comments (Ex.: "Malayalam (language)" --> "Malayalam").
10: #
11: # The two files were created using the same ‘Palindrome‘ program tested below.
12: # ‘shouldPass‘ were all the lines that passed, and
13: # ‘shouldFail‘ were all the lines that were presented as valid palidromes
14: #     on the website, but are in fact erroneous.
15: # The validity of these two lists can be manually inspected to verify accuracy.
16: #
17: # This script below confirms the findings yielded from the above process.
18: # In two loops, the respective files are passed in. If the results are as
19: # expected, then a period (‘.‘) is printed. If they are not as expected, then
20: # an ‘F‘ is printed to indicate an error.
21:
22: ##############################################
23: ################## CONFIG ####################
24: ##############################################
25:
26: shouldPass="palindromes.txt"
27: shouldFail="fake_palindromes.txt"
28:
29: ##############################################
30:
31: echo;
32: echo "SHOULD PASS";
33:
34: while read -r line
35: do
36:   if java Palindrome "$line" | grep -q ’IS a palindrome’; then
37:         echo -n ’.’
38:     else
39:         echo -n ’F’
40:     fi
41: done < "$shouldPass"
42:
43: ##############################################
44:
45: echo;
46: echo;
47: echo "SHOULD FAIL";
48:
49: while read -r line
50: do
51:     if java Palindrome "$line" | grep -q ’IS a palindrome’; then
52:         echo -n ’.’
53:     else
54:         echo -n ’F’
55:     fi
56: done < "$shouldFail"
```