

Exercise Description: FrequencyTable

The class `FrequencyTable` represents a table that tracks the number of repeated occurrences of items in a list of `Strings`. It has the API

```
public FrequencyTable()           // new table
public void click(String word)    // add 1 to frequency of this word
public int count(String word)     // what is frequency of this word?
public void show()                // print out all words and frequencies
```

For example,

```
FrequencyTable ft = new FrequencyTable();
ft.click("duck");
ft.click("duck");
ft.click("goose");
StdOut.println(ft.count("duck")); // should print 2
StdOut.println(ft.count("goose")); // should print 1
StdOut.println(ft.count("horse")); // should print 0
```

In addition, create a `main` that takes no arguments, reads words from standard input (using `StdIn.readString()`) until none are left, and then prints out the frequencies of all words using `show()`. Here are the relevant components of the ST API:

```
public class ST<Key, Value>           // Note: Key must be Comparable
-----
ST<Key, Value>()                     // create a symbol table
void put(Key key, Value v)           // put key-value pair into the table
Value get(Key key)                   // return value paired with key
                                     // or null if no such value
boolean contains(Key key)            // is there a value paired with key?
// Allows iteration with enhanced for loops:
for (Key key : st) {...}            // executes body once for each key
```

```

1 // Dependencies: ST.java (available on precepts page), StdIn, StdOut
2 public class FrequencyTable {
3     // maintain counts of all words seen so far
4     // the key is the word and the value is the count
5     private ST< _____ > st = _____ ;
6
7     // remark: we have not declared a constructor! but Java lets every class
8     // have a no-argument constructor by default. It only runs the line of
9     // code above (instance variable initialization).
10
11    // add 1 to the frequency of this word
12    public void click(String word) {
13        int count = count(word);
14        st.put(word, count + 1);
15    }
16
17    // what is the frequency of this word?
18    public int count(String word) {
19        if (!st. _____ ( _____ )) return 0; // if word is not in ST
20        else return _____ . _____ ( _____ ); // get word's count
21    }
22
23    // print out all words and frequencies
24    public void show() {
25        // foreach loop. goes through all keys in alphabetical order
26        for (String word : st) {
27            // print out frequency and word, separated by a space
28            StdOut.println( _____ ( _____ ) + " " + _____ );
29        }
30    }
31
32    // method used by client to count all words in StdIn
33    public static void main(String[] args) {
34
35        // build frequency table from words on standard input
36        FrequencyTable freq = new FrequencyTable();
37        while (!StdIn.isEmpty()) {
38            String word = StdIn.readString();
39            freq. _____ (word);
40        }
41
42        // print frequency table to standard output
43        freq.show();
44    }
45 }

```