

Homework Turnin

Account: 6G_06 (rgalanos@fcps.edu)
Section: 6G
Course: TJHSST APCS 2016-17
Assignment: 08-04
Receipt ID: 288c8babec49002493c3b4b678635020

Turnin Successful!

The following file(s) were received:

Dictionary.java (3566 bytes)

```
1. //Name:    Date:
2. import java.io.*;
3. import java.util.*;
4. public class Dictionary
5. {
6.     public static void main(String[] args)
7.     {
8.         Scanner infile = null;
9.         try
10.        {
11.            infile = new Scanner(new File("spanglish.txt"));
12.            System.setOut(new PrintStream(new FileOutputStream("dictionaryOutput.txt")));
13.        }
14.        catch(Exception e)
15.        {
16.        }
17.
18.        Map<String, Set<String>> eng2spn = makeDictionary( infile );
19.        System.out.println("ENGLISH TO SPANISH");
20.        display(eng2spn);
21.
22.        Map<String, Set<String>> spn2eng = reverse(eng2spn);
23.        System.out.println("SPANISH TO ENGLISH");
24.        display(spn2eng);
25.    }
26.    public static Map<String, Set<String>> makeDictionary(Scanner infile)
27.    {
28.        Map<String, Set<String>> dictionary = new HashMap<String, Set<String>>();
29.
30.        while(infile.hasNext())
31.        {
32.            String english = infile.next();
33.            String spanish = infile.next();
34.            add(dictionary, english, spanish);
35.        }
36.
37.        return dictionary;
38.    }
39.    private static void add(Map<String, Set<String>> dictionary, String word, String translation)
40.    {
41.        Set<String> temp;
42.
43.        if(dictionary.containsKey(word))
44.        {
45.            temp = dictionary.get(word);
46.            temp.add(translation);
47.        }
48.        else
49.        {
50.            temp = new TreeSet<String>();
```

```

51.     temp.add(translation);
52. }
53.
54.     dictionary.put(word,temp);
55. }
56. public static void display(Map<String, Set<String>> m)
57. {
58.     Map<String, Set<String>> m2 = new TreeMap<String, Set<String>>(m);
59.     Set<String> key = m2.keySet();
60.     Iterator it = key.iterator();
61.
62.     while(it.hasNext())
63.     {
64.         Object obj = it.next();
65.         System.out.println("\t" + obj + " " + m2.get(obj));
66.     }
67.     System.out.println();
68. }
69. public static Map<String, Set<String>> reverse(Map<String, Set<String>> dictionary)
70. {
71.     Map<String, Set<String>> reversed = new HashMap<String, Set<String>>();
72.     for(Map.Entry<String, Set<String>> entry: dictionary.entrySet())
73.     {
74.         Set<String> temp = entry.getValue();
75.         Iterator it = temp.iterator();
76.         while(it.hasNext())
77.             add(reversed,it.next().toString(),entry.getKey());
78.     }
79.     return reversed;
80. }
81. }
82.
83. /*****
84. INPUT:
85.     holiday
86.     fiesta
87.     holiday
88.     vacaciones
89.     party
90.     fiesta
91.     celebration
92.     fiesta
93.     <etc.>
94. *****/
95. OUTPUT:
96.     ENGLISH TO SPANISH
97.     banana [banana]
98.     celebration [fiesta]
99.     computer [computadora, ordenador]
100.    double [doblar, doble, duplicar]
101.    father [padre]
102.    feast [fiesta]
103.    good [bueno]
104.    hand [mano]
105.    hello [hola]
106.    holiday [fiesta, vacaciones]
107.    party [fiesta]
108.    plaza [plaza]
109.    priest [padre]
110.    program [programa, programar]
111.    sleep [dormir]
112.    son [hijo]
113.    sun [sol]
114.    vacation [vacaciones]
115.
116.    SPANISH TO ENGLISH
117.    banana [banana]
118.    bueno [good]
119.    computadora [computer]
120.    doblar [double]
121.    doble [double]
122.    dormir [sleep]
123.    duplicar [double]
124.    fiesta [celebration, feast, holiday, party]
125.    hijo [son]
126.    hola [hello]
127.    mano [hand]
128.    ordenador [computer]
129.    padre [father, priest]
130.    plaza [plaza]
131.    programa [program]
132.    programar [program]

```

```
132.      sol [sun]
133.      vacaciones [holiday, vacation]
134.
135. *****/
```