828. Word Pattern

* [Description](http://lintcode.com/en/problem/word-pattern/" \l "description)
* [Notes](http://lintcode.com/en/problem/word-pattern/#note)
* [Testcase](http://lintcode.com/en/problem/word-pattern/#testcase)
* [Judge](http://lintcode.com/en/problem/word-pattern/#judge)

Given a pattern and a string str, find if strfollows the same pattern.

Here **follow** means a full match, such that there is a bijection between a letter in pattern and a **non-empty** word in str.

 Notice

You may assume pattern contains only lowercase letters, and str contains lowercase letters separated by a single space.

Have you met this question in a real interview?

Yes

**Example**

Given pattern = "abba", str = "dog cat cat dog", return true.  
Given pattern = "abba", str = "dog cat cat fish", return false.  
Given pattern = "aaaa", str = "dog cat cat dog", return false.  
Given pattern = "abba", str = "dog dog dog dog", return false.

<http://lintcode.com/en/problem/word-pattern/#>

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package javaapplication1;

import java.util.\*;

/\*\*

\*

\* @author Usuario

\*/

public class JavaApplication1 {

public static boolean wordPattern(String pattern, String str) {

// write your code here

String[] sp = str.split(" ");

if(pattern.length() != sp.length) {

return false;

}

HashMap <Character, String> hm = new HashMap();

for (int i = 0; i < pattern.length(); i++)

{

char key = pattern.charAt(i);

String val = sp[i];

if (hm.containsKey(key))

{

if (!hm.get(key).equals(val))

{

return false;

}

}

else

{

//if (!diccio.ContainsValue(val))

if(!hm.containsValue(val))

{

//diccio[key] = val;

hm.put(key, val);

}

else

{

return false;

}

}

}

return true;

}

public static void main(String[] args) {

// TODO code application logic here

String pattern = "abba";

String str = "dog cat cat dog";

//String pattern ="abba";

//String str = "dog cat cat fish";

//String pattern = "abba";

// String str = "dog dog dog dog";

System.out.println(wordPattern(pattern, str));

}

}