

## Question #1

```
There's a service that on start loads a preexisting collection of words.
It exposes two APIs:

Put(string s) - Adds string s to the collection of strings.
Get(string s) - Returns all of the permutations of the given string s that
are stored in the service.

Implement the above.

Example: Given an initial list of strings: "acb", "bac", "bbcc", "cba"
Get("bac") --> returns "acb", "bac", "cba"
Get("bbc") --> return empty collection
Put("bac") --> Adds "bac" to the words collection
```

Solution:

```
import java.util.*;

public class Solution {

    public static void main(String [] args) {

        System.out.println("This is a debug message");
        WordServices wordService = new WordServices();
        wordService.putWordIntoCollection("");
    }
}

class WordServices {
    public static Set<String> wordsCollection = new HashSet<String>();
    Map<String , Set<String>> wordPermatMap = new HashMap <String , Set<String>>();

    public void putWordIntoCollection (String s)
    {
        Set<String> allPermSet = new HashSet<String>();
        //the words collection
        if(wordPermatMap.containsKey(sortWord(s)))
        {
            allPermSet = new HashSet<String>(wordPermatMap.get(sortWord(s)));
        }
    }
}
```

```
        allPermSet.add(s);
        wordPermatMap.put(sortWord(s) , allPermSet);
    }

    public Set<String> getAllPerm( String word )
    {
        return wordPermatMap.get(sortWord(word));
    }

    public String sortWord(String word)
    {
        char[] wordArr = word.toCharArray();
        Arrays.sort(wordArr);
        word = new String(wordArr);
        return word;
    }
}
```