```java
import java.util.*;
public class ValidParentheses {
    public static boolean isVaild(String s){
      Stack<Character> st=new Stack<>();

      for(int i=0;i<s.length();i++){
            if(s.charAt(i)=='{' ||
               s.charAt(i)=='[' ||
               s.charAt(i)=='('  ){
                  st.push(s.charAt(i));
            }
            else if(!st.empty() && (
                        (s.charAt(i)==']' && st.peek()=='[')||
                        (s.charAt(i)=='}' && st.peek()=='{')||
                        (s.charAt(i)==')' && st.peek()=='('))){
                  st.pop();

            }
            else{
                  st.push(s.charAt(i));
            }
      }
            return st.empty()?true:false;

    }
      public static void main(String[] args) {
            // TODO Auto-generated method stub
            String s="{()}";
            boolean res=isVaild(s);
            if(res)
                  System.out.print("yes");
            else
                  System.out.print("no");

      }

}
//yes


import java.util.*;
public class GroupAnagramTogether {
      //Anagram--same char order diff
    public static List<List<String>> groupAnagrams(String[] strs){
      List<List<String>> res=new ArrayList<>();

      HashMap<String,List<String>> map=new HashMap<>();//List of
String
```

```
      for(String str:strs){//traverse
            char[] chArr=str.toCharArray();//eat arr
            Arrays.sort(chArr);//sort   aet   //nlogn
            String key=new String(chArr);//string

            if(map.containsKey(key)){
                  map.get(key).add(str);//tea
            }
            else{
                  List<String> strList=new ArrayList<>();//create
                  strList.add(str);//add
                  map.put(key, strList);//aet k,eat v
            }
            }
      res.addAll(map.values());
      return res;
    }
      public static void main(String[] args) {
            // TODO Auto-generated method stub
            String[] strs={"eat","tea","tan","ate","nat","bat"};
            System.out.println(groupAnagrams(strs));

      }

}
//[[eat, tea, ate], [bat], [tan, nat]]



import java.util.*;

public class RomanToInteger {
      //12 -XII larger 1st
      //7- VII

      //4  -IV(SUBTRACTION)
      //9  -IX ''
      //40 -XL
      //90 -XC

      //---
      //400 -CD
      //900 -CM
    public static int  convertRomanToInteger(String str){


      Map<Character,Integer> map=new HashMap<>();
```

```
		map.put('I', 1);//		(7)
		map.put('V', 5);
		map.put('X', 10);
		map.put('L', 50);
		map.put('C', 100);
		map.put('D', 500);
		map.put('M', 1000);

		int res=0;
		for(int i=0;i<str.length()-1;i++){//traverse
			if(map.get(str.charAt(i))>=map.get(str.charAt(i+1))){//
X>=I
				res+=map.get(str.charAt(i));//ADDTION
			}
			else{
				res-=map.get(str.charAt(i));// IV IX SUBTRACTION
			}
		}
		res+=map.get(str.charAt(str.length()-1));
			return res;

	}
		public static void main(String[] args) {
			// TODO Auto-generated method stub
			String str="XII";
			System.out.println(convertRomanToInteger(str));

		}

}
//12



import java.util.*;
public class VowelConsonantCount {

		public static void main(String[] args) {
			// TODO Auto-generated method stub

			String str="Sneha";

			countVowels(str);

		}

		private static void countVowels(String str) {
			// TODO Auto-generated method stub
```

```
		int vowelCount=0;
		int consonantCount=0;

		for(int i=0;i<str.length();i++){
			if(isVowel(str.charAt(i))){
				vowelCount++;
			}
			else{
				consonantCount++;
			}
		}
		System.out.println("vowelCount "+ vowelCount);
		System.out.print("consonantCount "+consonantCount);
	}

	private static boolean isVowel(char ch) {
		// TODO Auto-generated method stub
		ch=Character.toUpperCase(ch);

		return (ch=='A' ||ch=='E' || ch=='I' || ch=='O' ||
ch=='U');
	}

}
//vowelCount 2
//consonantCount 3



import java.util.*;
public class RotatedString {

	public static void main(String[] args) {
		// TODO Auto-generated method stub
		System.out.print(isRotatedVersion("abcd","bcda"));

	}
	public static boolean isRotatedVersion(String str,String
rotatedString)
	{
		boolean isRotated=false;

		if(str.length()!=rotatedString.length()){
			return false;
		}else{
			String concatenated=str+str;
			return concatenated.contains(rotatedString);
		}
```

```
	}

}
//true



import java.util.*;
public class EvenWord {

	public static void main(String[] args) {
		// TODO Auto-generated method stub

		String str="Hell World We Lol";

		for(String s:str.split(" "))
		{
			if(s.length()%2==0){
				System.out.println(s);
			}
		}

	}

}
//Hell
//We




public class SumStrings {

	public static void main(String[] args) {
		// TODO Auto-generated method stub
	   String s1="111";
	   String s2="222";

	   int sum=Integer.parseInt(s1)+Integer.parseInt(s2);
	   System.out.print(sum);
	   System.out.println(String.valueOf(sum));
	}

}
```

```java
//333



public class StringEquals {

	public static void main(String[] args) {
		// TODO Auto-generated method stub
		 String s1=new String("sneha");
		 String s2=new String("Sneha");
		 System.out.println(s1.equals(s2));
		 System.out.println(s1.equalsIgnoreCase(s2));
		 System.out.println(s1.compareTo(s2));
		 System.out.println(s1.compareToIgnoreCase(s2));

	}

}
//false
//true
//32
//0
```