Daily Challenge **Happy Coding from necse** SKILLRACK

IPv4 Address Validity

Given an IPv4 address as a string, check if the address is valid. Print **Valid** if the address is valid else print **Invalid**. IPv4 address has 4 blocks of 8 bit (unsigned) numbers ranging from 0 to 255 separated by a . (dot).

Boundary Condition(s):
1 <= Length of address <= 50

Input Format:
The first line contains the IP address.

Output Format:
The first line contains Valid or Invalid.

Example Input/Output 1:
Input:
127.0.0.1

Output:
Valid

Example Input/Output 2:
Input:
266.2.9.34.12

Output:
Invalid

Explanation:
As 266 is present in the IP address

Max Execution Time Limit: 5000 millisecs

Ambiance

Java (12.0)

```
1  import java.util.*;
2  public class Hello {
3
4      public static void main(String[] args) {
5          Scanner sc = new Scanner(System.in);
6
7          String s[] = sc.nextLine().split("[.]");
8          if(s.length!=4){ failed(); return; }
9
10         for(String x:s){
11             if(x.equals("")) { failed(); return; }
12             for(char i:x.toCharArray()){
13                 if(!(i>='0' && i<='9')){ failed(); return;}
14             }
15             int k=Integer.parseInt(x);
16             if(!(k>=0 && k<=255)) { failed(); return; }
17         }
18
19         System.out.println("Valid");
20
21     }
22     public static void failed(){
23         System.out.println("Invalid");
24     }
25 }
```

1912067@nec

Code did not pass the execution

```
Hello.java:11: error: -> expected
    if(x.equals("")) { failed(); return(); }
                     ^
Hello.java:13: error: -> expected
        if(!(i>='0' && i<='9')){ failed(); return();}
                               ^
2 errors
```

Save Run