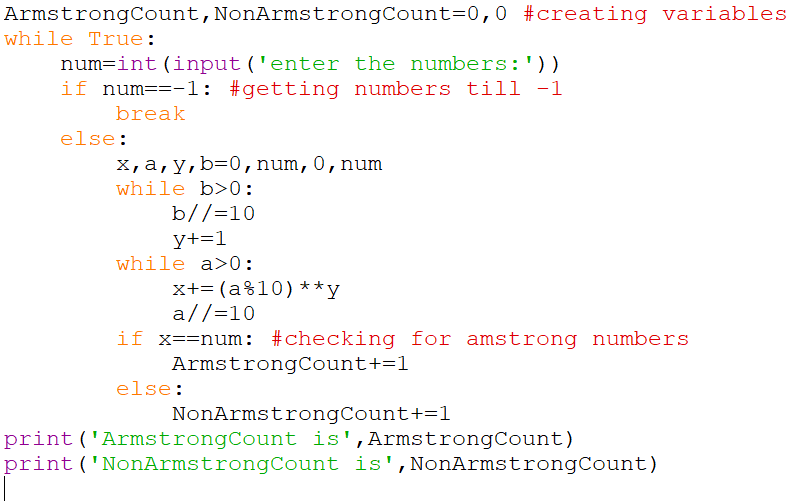
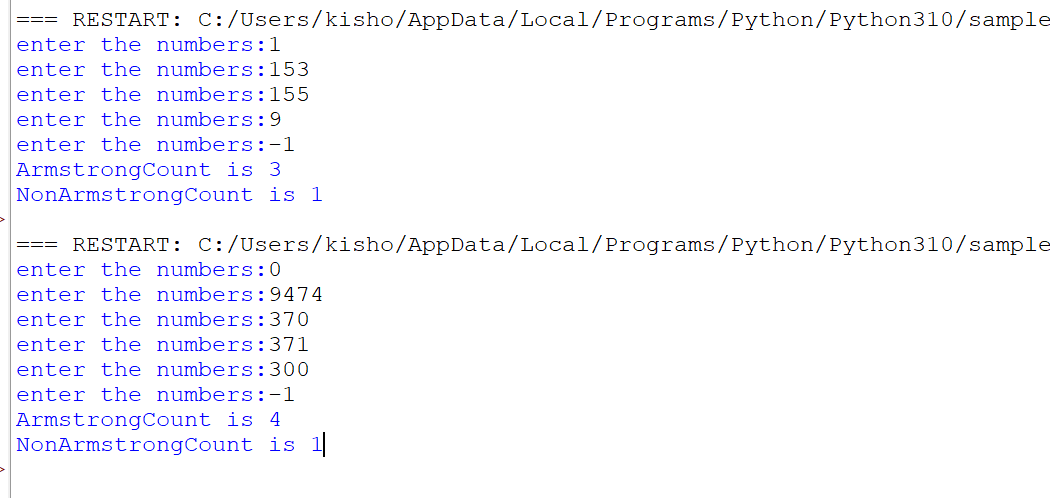
**IPS and PAT codes from Vpropel**

**Write a Python program using while loop to get numbers until -1 is encountered. Find the count of the Amstrong Numbers present in the given set of numbers and display the count for both the cases.**

**Input : Get numbers from the user separated by a space until -1 is encountered.**

**Output: Display the "AmstrongCount" and the "NonAmstrongCount" using the respective varibale names specified.**





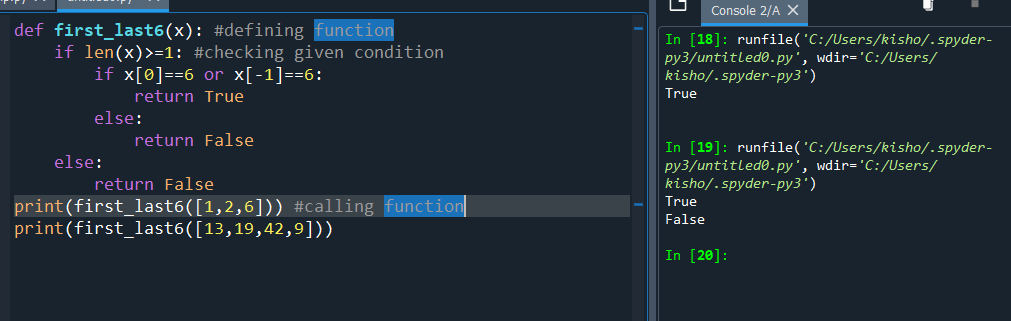
**Given an array of integers in the list data structure, return True from the function, if 6 appears as either the first or last element in the list.**

**The list will be length 1 or more.**

**first\_last6([1, 2, 6]) should return True**

**first\_last6([6, 1, 2, 3])** **should return** **True**

**first\_last6([13, 6, 1, 2, 3])** **should return** **False**

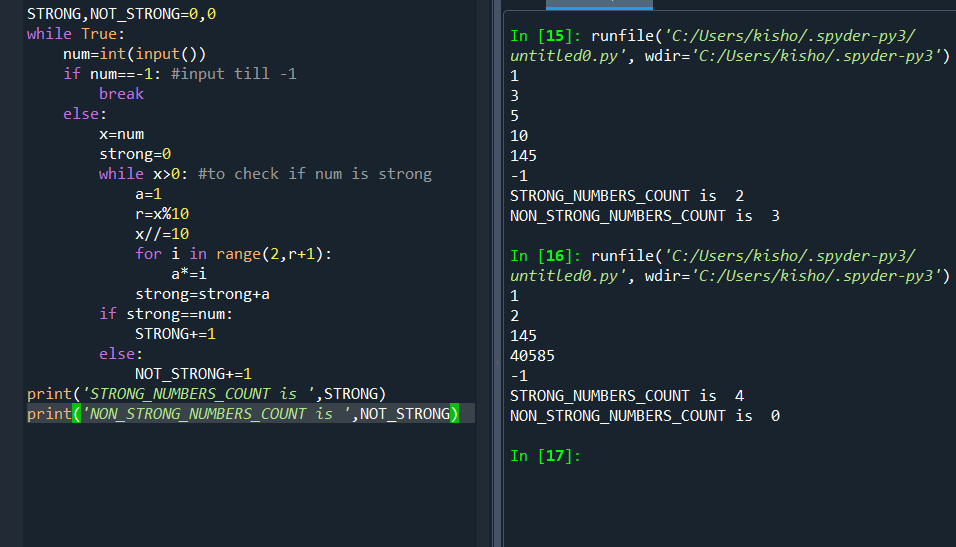


**Write a Python Code to check whether the number that is given by the user is a STRONG number or not.**

**Continue getting the numbers till -1 is pressed by the user. Display the count of both the cases.**

**Input: Get values one after the other from the user till the user enters -1.**

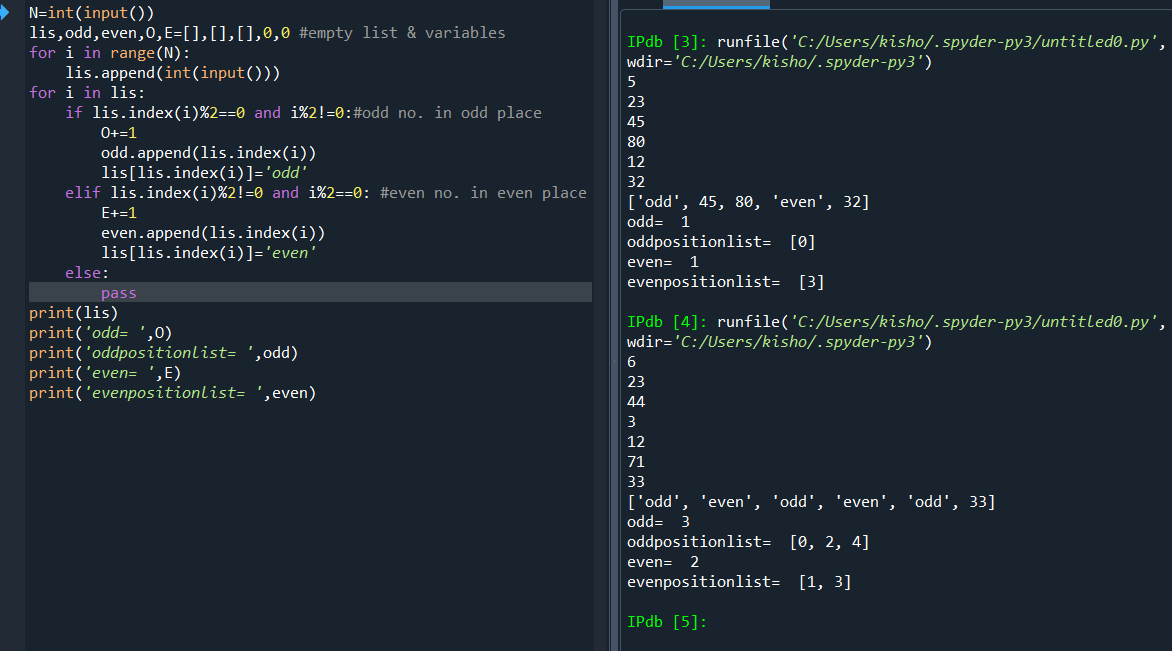
**Output: Display the count of STRONG and NOT\_STRONG numbers**



**Get a Python list of N integer values.  
Find the presence of ODD and EVEN numbers in the list.  
If the ODD number occurs at the ODD position replace the ODD number with 'odd'.  
If the EVEN number occurs at the EVEN position replace the EVEN number with 'even'.**

**Find the number of odd places that were changed and also the number of even places that were changed**

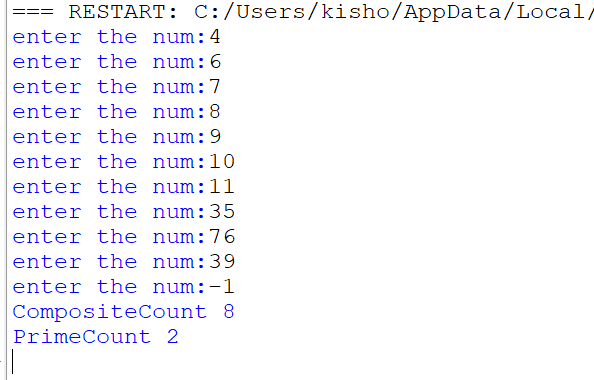
**Prepare a new list for the odd and even positions.**



**Write a Python program using while loop to read the numbers until -1 is encountered. Count the number of prime and composite numbers entered by the user.**

**Input: Numbers are given one after the other in separate lines. Give the last number as -1.**

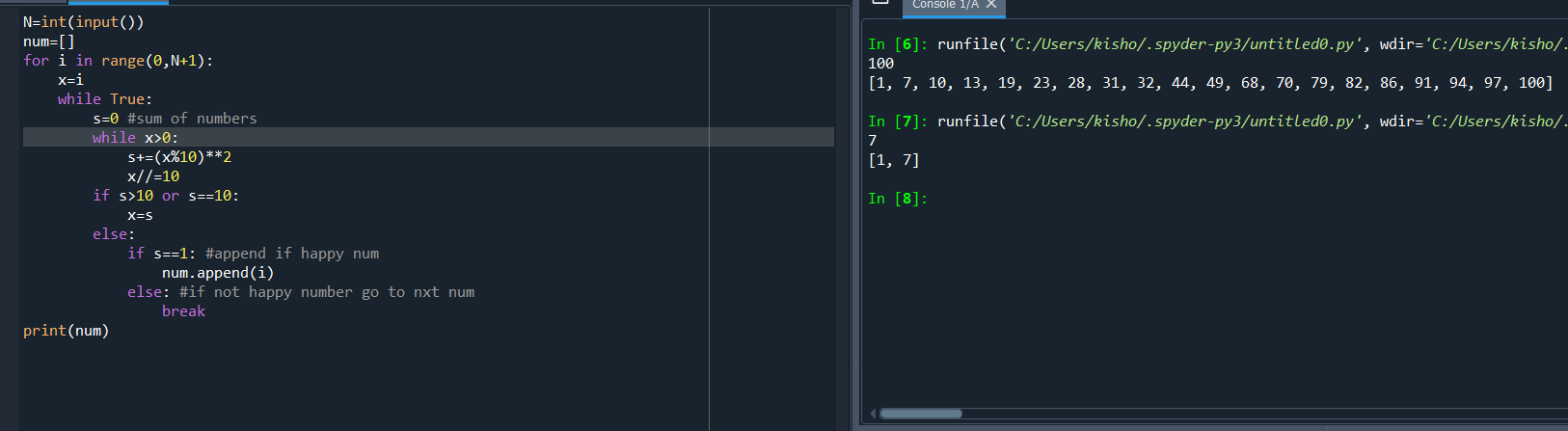
**Output: Print the count of the prime numbers using the variable name PrimeCount and the count of composite numbers using the variable name CompositeCount.**



**Write a Python code to display all the happy numbers in a list data structure from 0 till the given number N which is got as input**

**Input: Getthe value of N from the user**

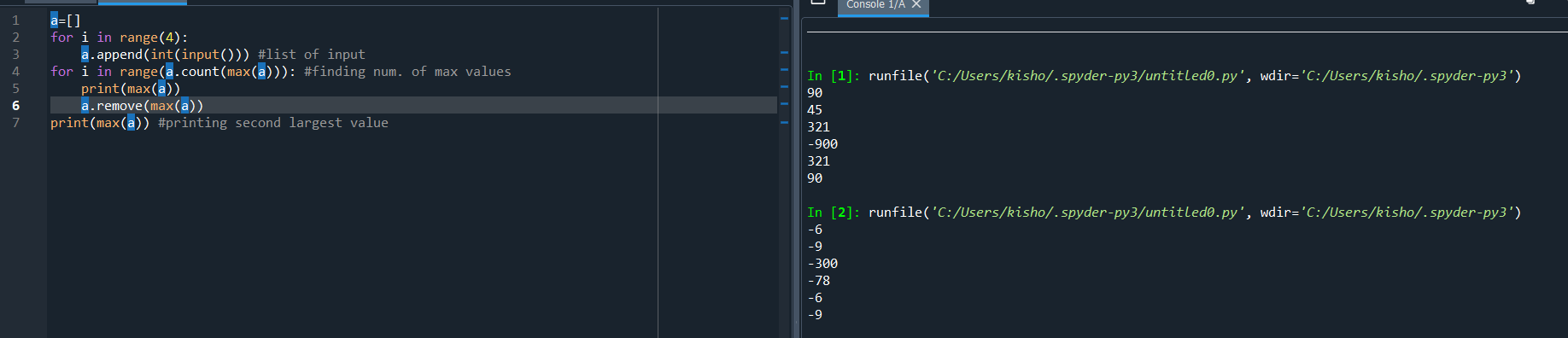
**Output: Display all the Happy Numbers in a list from 0 till that number N**



**Write a Python code to display the second largest number in the given list of numbers without sorting the list.**

**Input: Set of numbers are given in the Python data structure named list.**

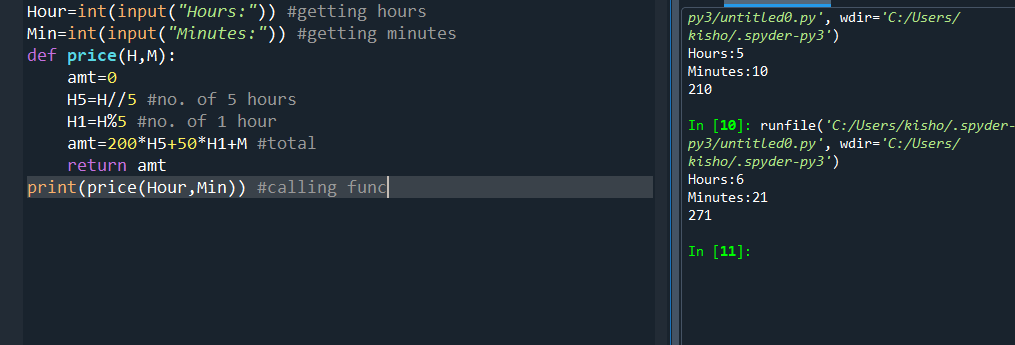
**Output: Second largest number in that list is display.**



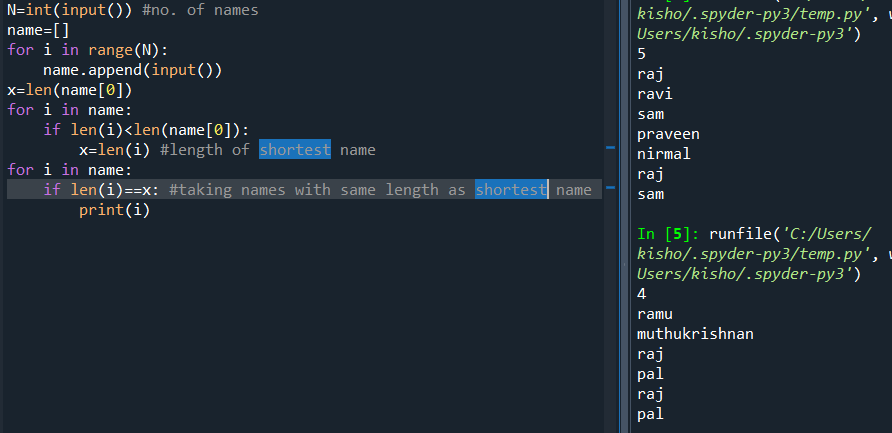
**Given the number of hours and minutes browsed, write a program using functions to calculate bills for Internet Browsing in a browsing centre. The conditions are given below.  
(a) 1 Hour - Rs.50  
(b) 1 minute - Re. 1  
(c) Five hours - Rs. 200**

**Pass the number of hours and minutes browsed to the function and display the result in the driver program from where the function was called.**

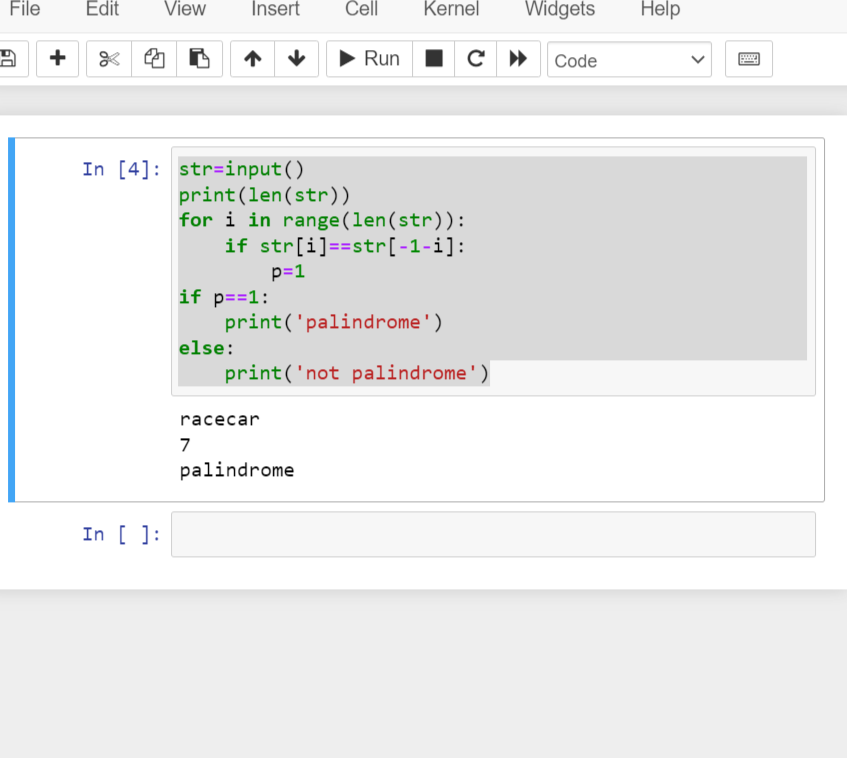
**Use default values in case the number of hours or minutes are not being passed.**



**Get N names in a list and prints the shortest names in the list, one on each line. There is more than one shortest name, so both need to be printed.**



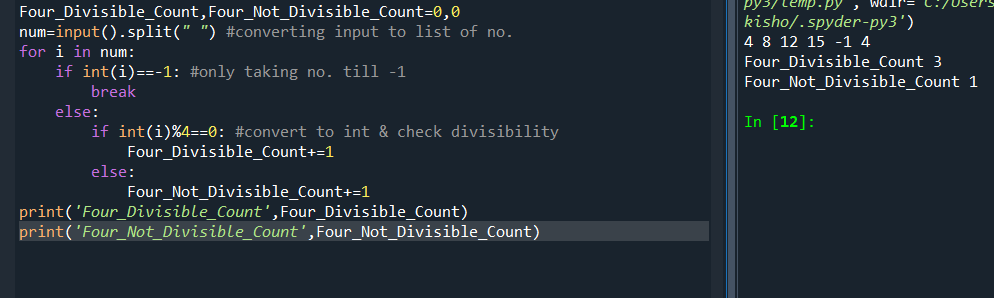
**From the given string, find the string length and check whether the string is palindrome or not.**



**Write a Python program using while loop to read numbers until -1 is encountered. Count the numbers divisible by 4 and not divisible by 4 and display them.**

**Input: Get numbers using while loop until -1 is encountered separated by a space and not tab**

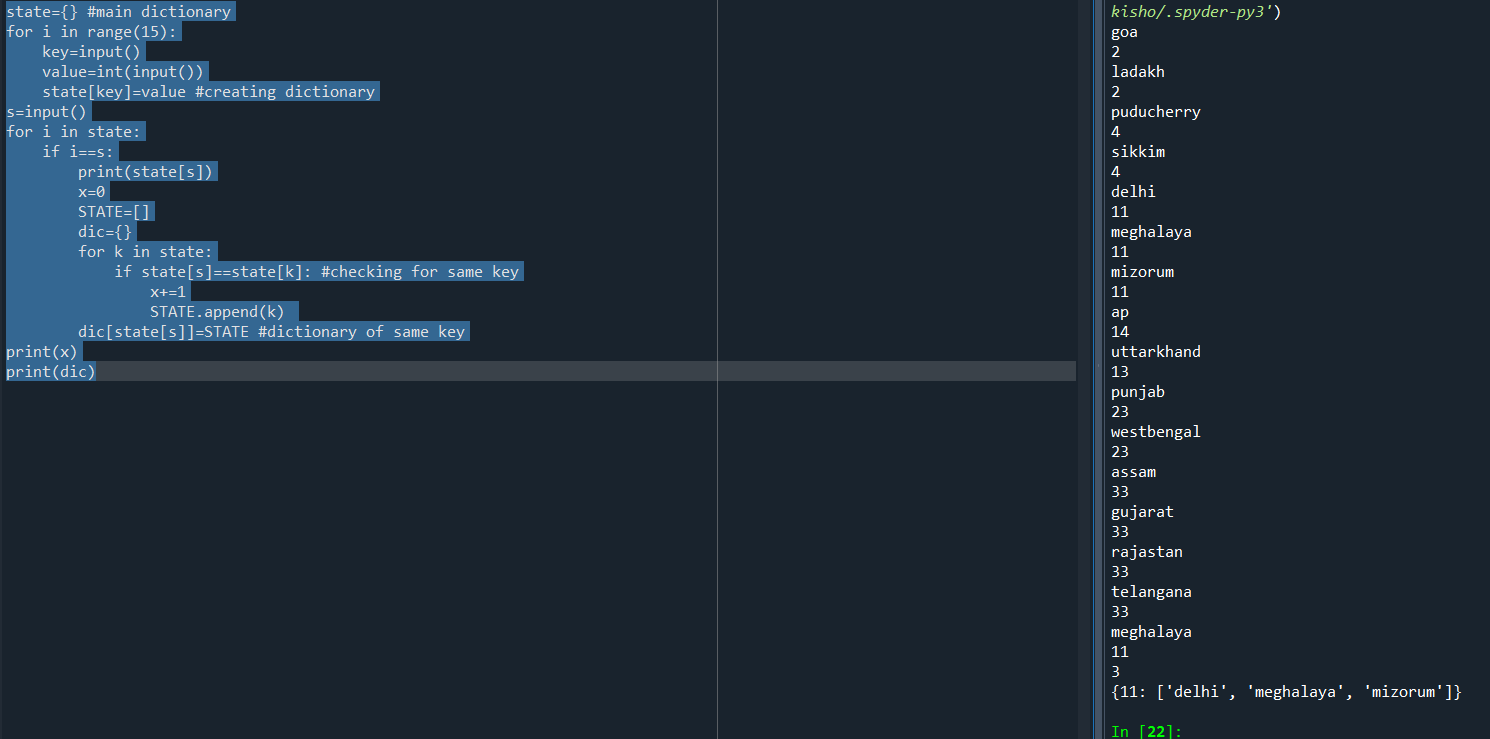
**Output: Display the count of the numbers divisible by 4 using the variable named Four\_Divisible\_Count and the count of the numbers not divisible by 4 using the variable named Four\_Not\_Divisible\_Count.**

****

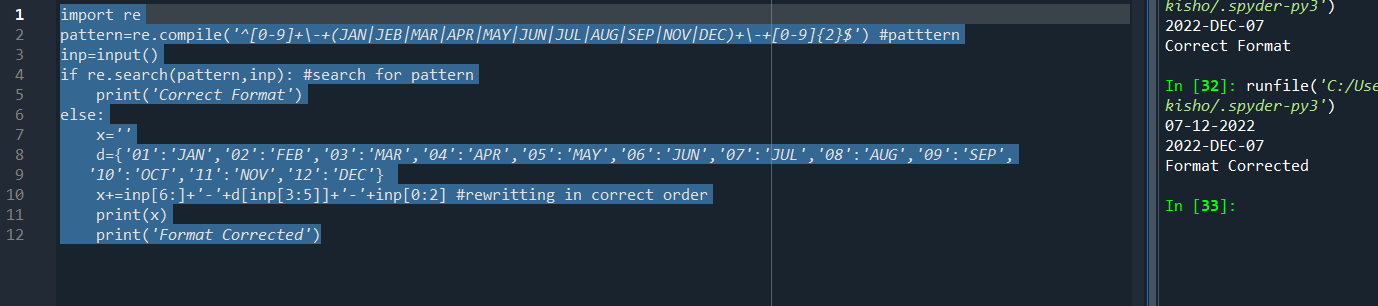
**Get the following states and their number of districts as key and value pairs in a dictionary.**

**Example: {'GOA':2, 'LADAKH':2, 'PUDUCHERRY':4, 'SIKKIM':4, 'DELHI':11, 'MEGHALAYA':11, 'MIZORAM':11, 'ANDHRA PRADESH':13, 'UTTARAKHAND':13, 'PUNJAB':23, 'WEST BENGAL':23, 'ASSAM':33, 'GUJARAT':33, 'RAJASTHAN':33, 'TELANGANA':33}  
For every entered state,  
- find its number of districts   
- count the number of states that has the same count of districts  
- Form a new dictionary from the above-founded number of districts as key and the list of states that have the same number of districts as its value.**

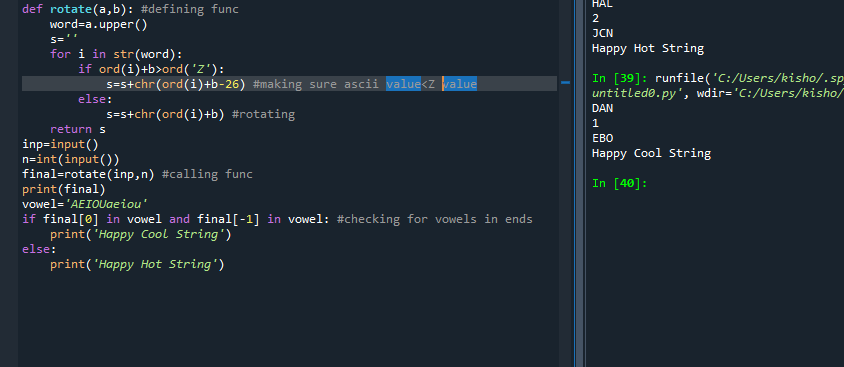
**Sample i/p:  
State1  
2(No. of districts)  
State2  
2(No. of districts)  
State3  
4(No. of districts)  
State4  
4(No. of districts)  
State5  
11(No. of districts)  
State6  
11(No. of districts)  
State7  
11(No. of districts)  
State8  
13(No. of districts)  
State9  
13(No. of districts)  
State10  
23(No. of districts)  
State11  
23(No. of districts)  
State12  
33(No. of districts)  
State13  
33(No. of districts)  
State14  
33(No. of districts)  
State15  
33(No. of districts)  
Sate12  [Search Key]  
Sample o/p:  
33  
4  
{33: ['Sate12', 'State13', 'State14', 'State15']}**

****

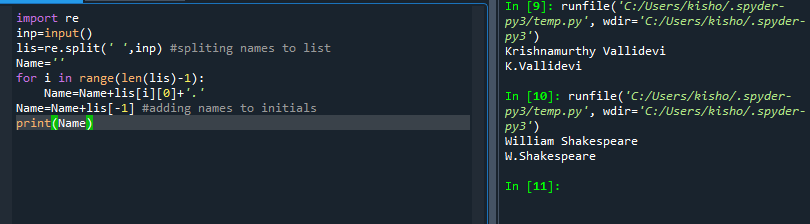
**Write a Python program to check whether the date of birth that is given as input is in the format YYYY-MMM-DD. If it is available leave it as such displaying the message, “Correct Format”. Else, Convert the given input (DD-MM-YYYY) into the required format and display the result along with the message “Format Corrected”. Use regular expression for this program to check the format of the given string.**

****

**Encrypt a given message by “rotating” each letter by a fixed number of places. To rotate a letter means to shift it through the alphabet, wrapping around to the beginning if necessary, so ‘A’ rotated by 3 is ‘D’ and ‘Z’ rotated by 1 is ‘A’. Write a function called rotate\_word that takes a string and an integer as parameters, and returns a new string that contains the letters from the original string rotated by the given amount. Check whether the output string has the vowels in the beginning and at the end of the string. If the condition satisfies, display the message “Happy Cool String” else display “Happy Hot String”.**



**Write a program that takes your full name as input and displays the abbreviations of  
the first and middle names except the last name which is displayed as it is. For example, if your name is Robert Brett Roser, then the output should be R.B.Roser .**



**Given two strings, s1 (all Upper Case) and s2 (All lowerCase), create a mixed String  
Instructions:  
Create a third-string made of the first char of s1 then the last char of s2, Next,  
the second char of s1 and second last char of s2, and so on. Any leftover chars  
go at the end of the result.**

**Check whether the output string follows the pattern of Upper and lower case alphabets being available in the alternate position.**

**If not in the expected pattern, Please display "Does Not Follow Any Pattern"**

**If it follows the pattern, then display, "Follows Pattern"**

