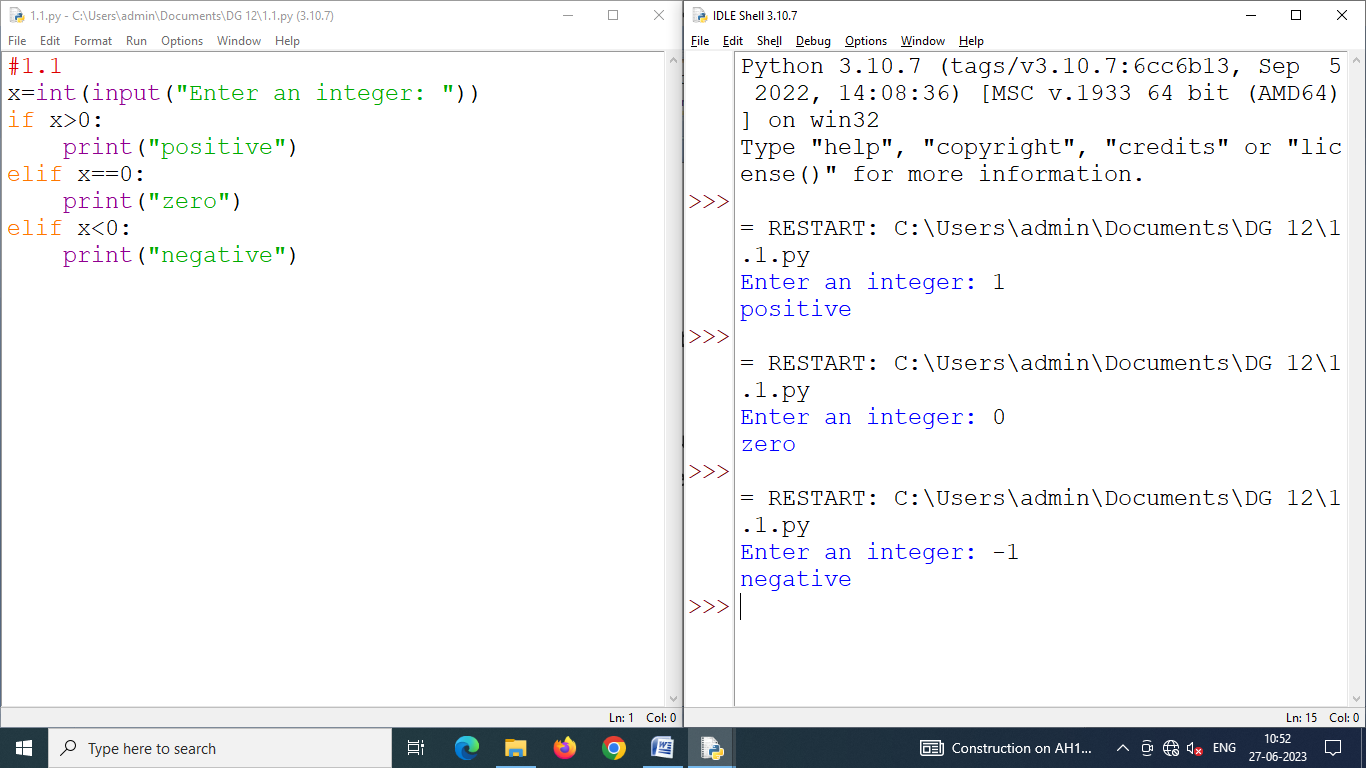
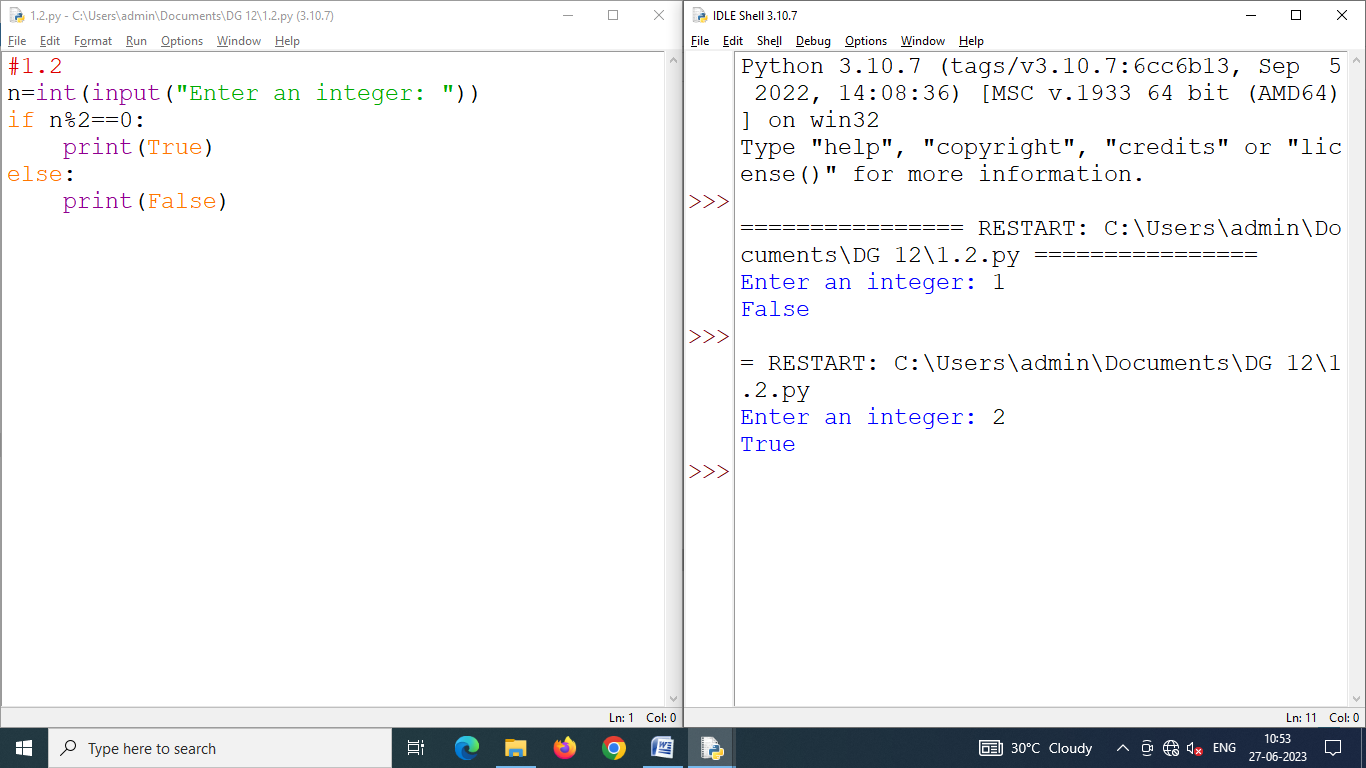
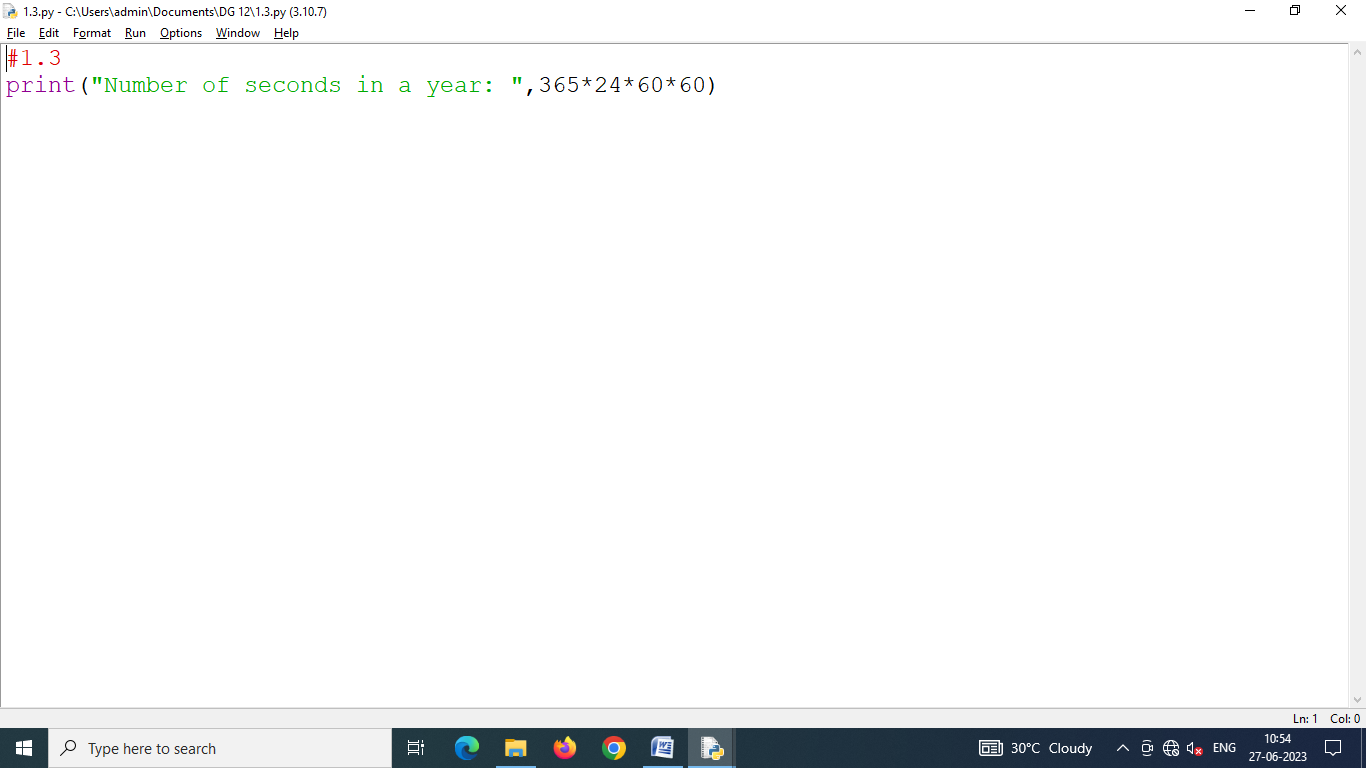
Program 1.1 **Write a program to print one of the words negative, zero, or positive, according to whether variable x is less than zero, zero, or greater than zero, respectively.**

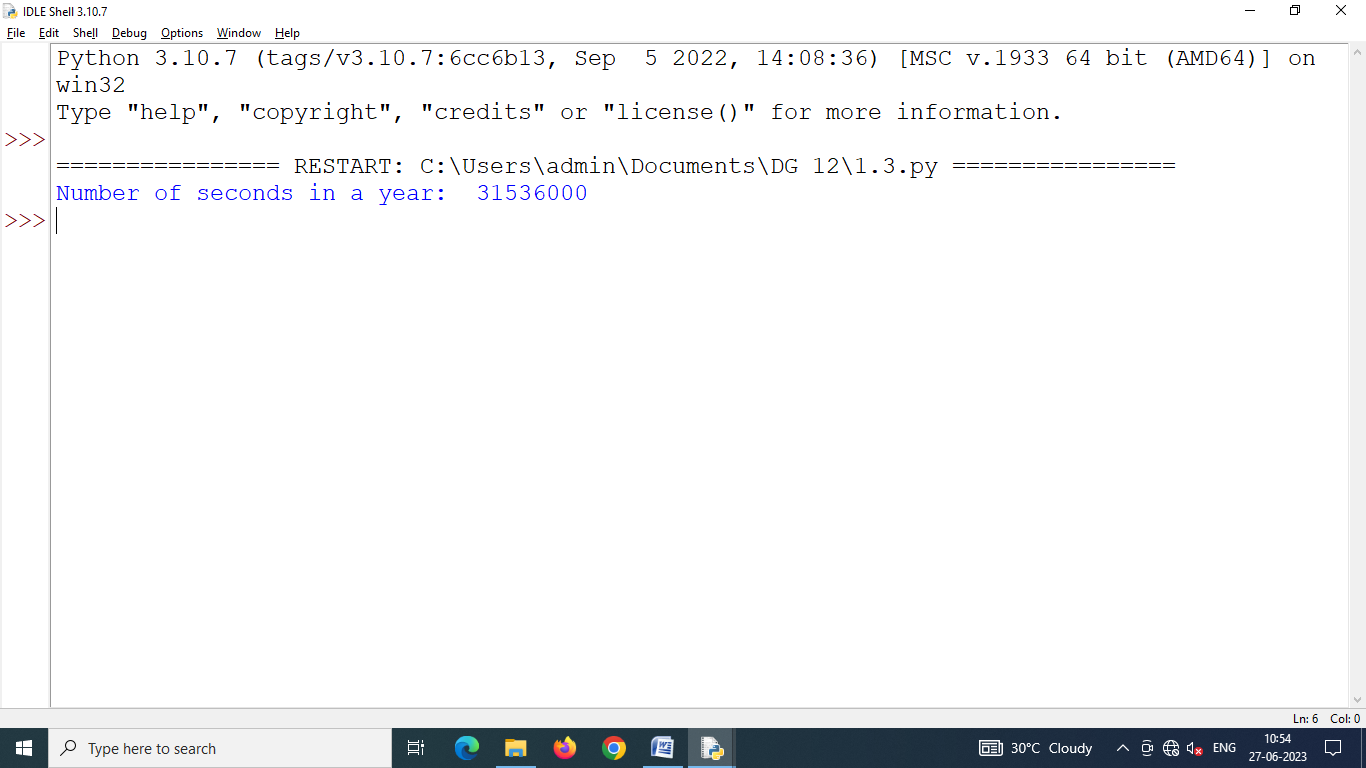
****

Program 1.2 **Write a program that returns True if the input number is an even number, False otherwise.**

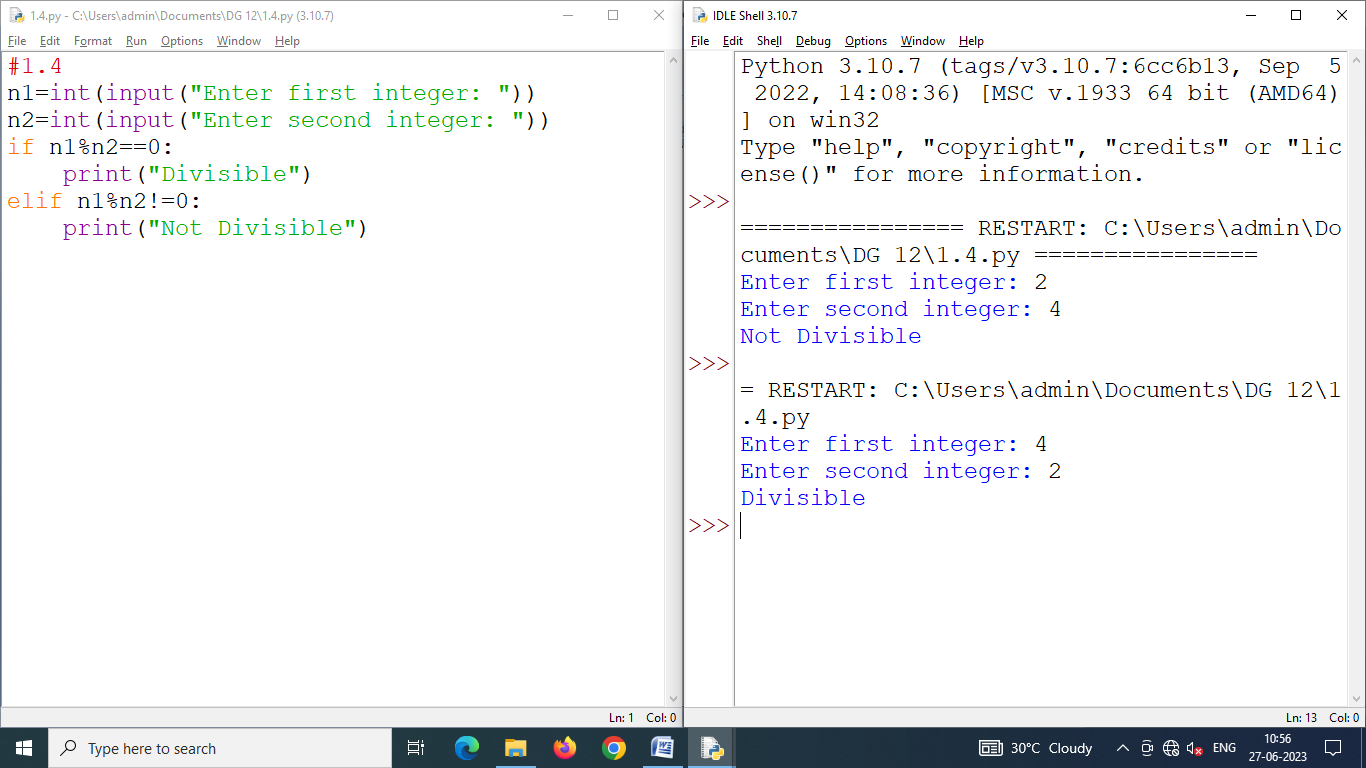
****

Program 1.3 **Write a Python program that calculates and print the number of seconds in a year.**

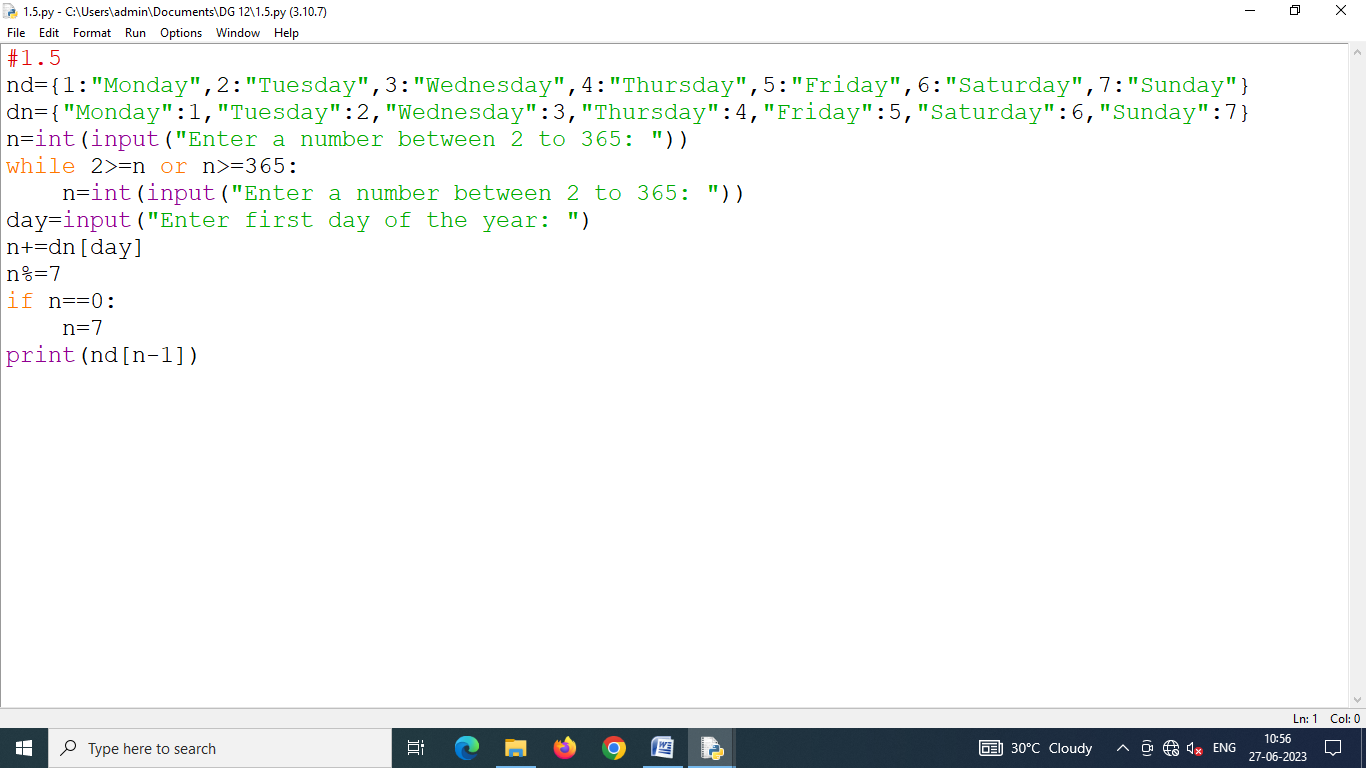


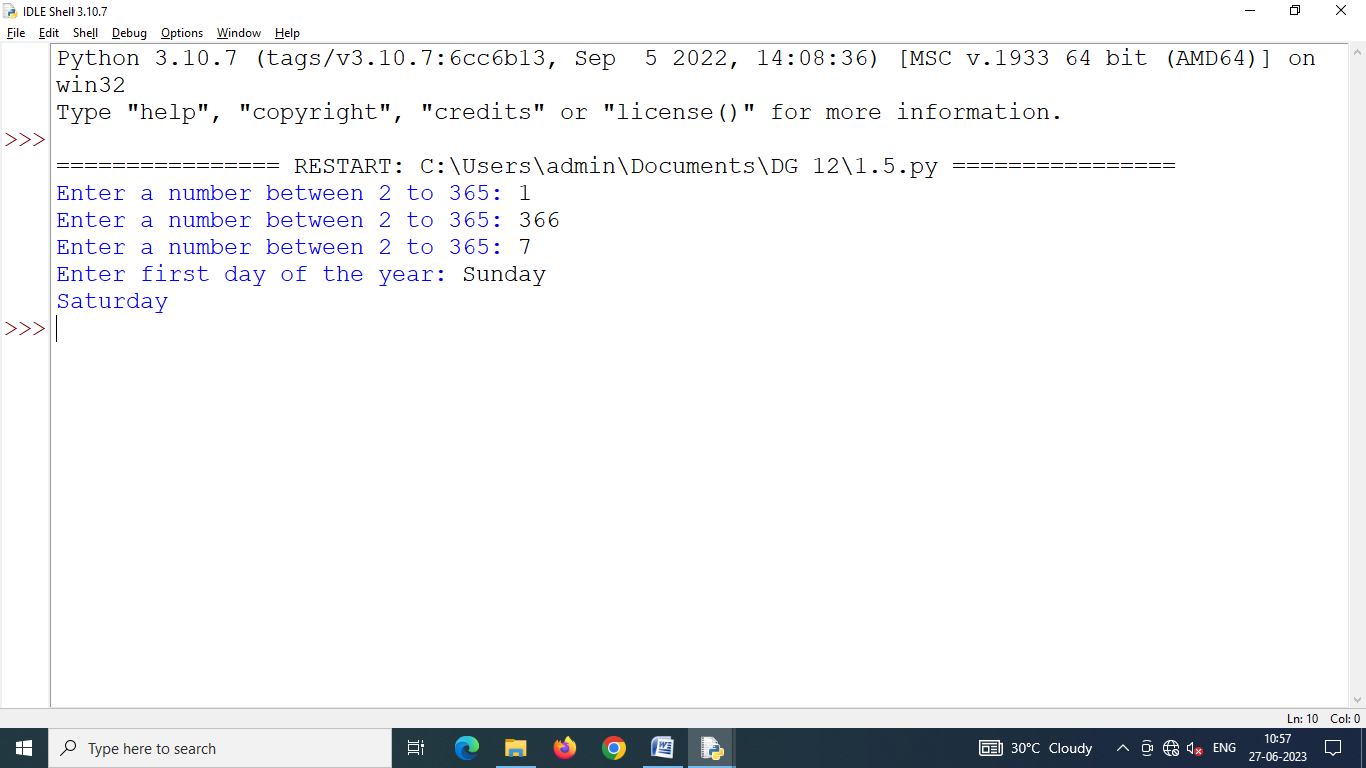


Program 1.4 **Write a Python program that accepts two integers from the user and prints a message saying if first number is divisible by second number or if it is not.**

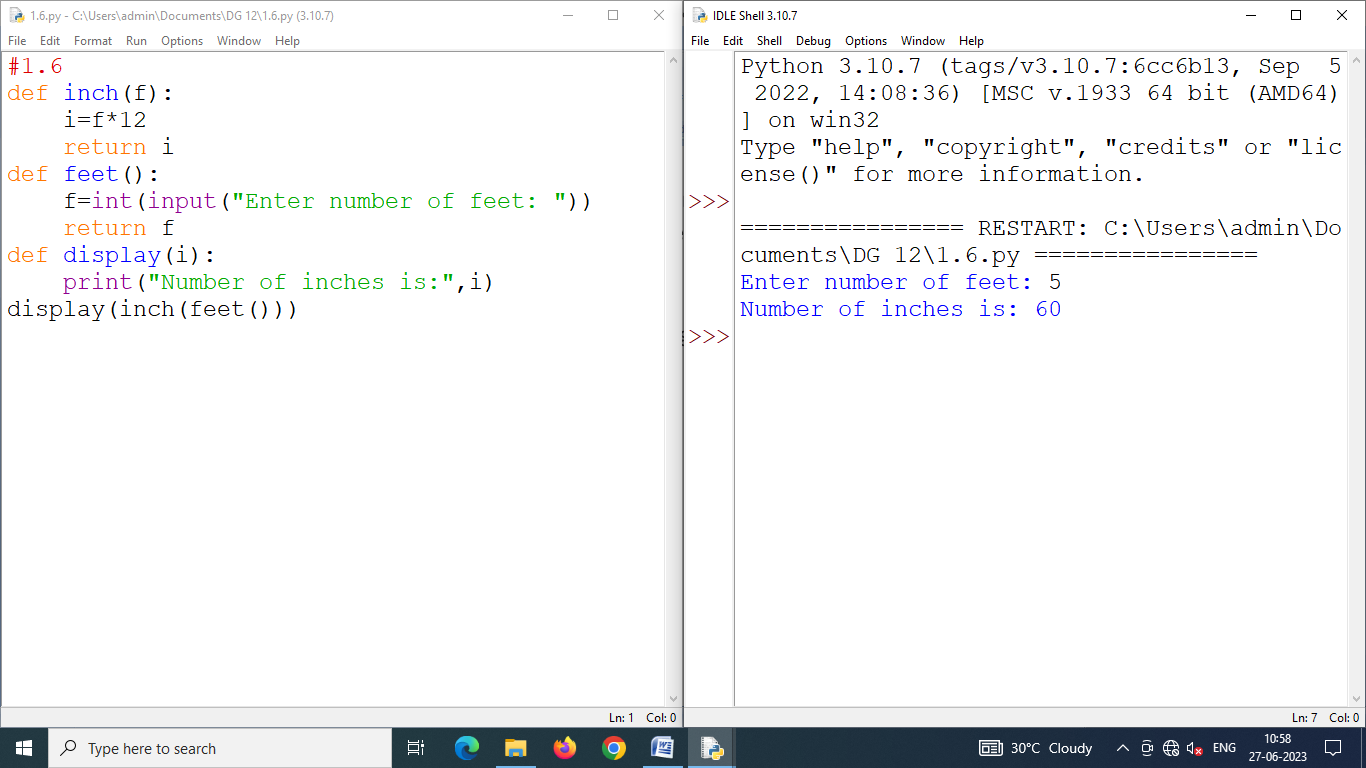
****

Program 1.5 **Write a program that asks the user the day number in a year in the range 2 to 365 and asks the first day of the year – Sunday or Monday or Tuesday etc. Then the program should display the day on the day-number that has been input.**

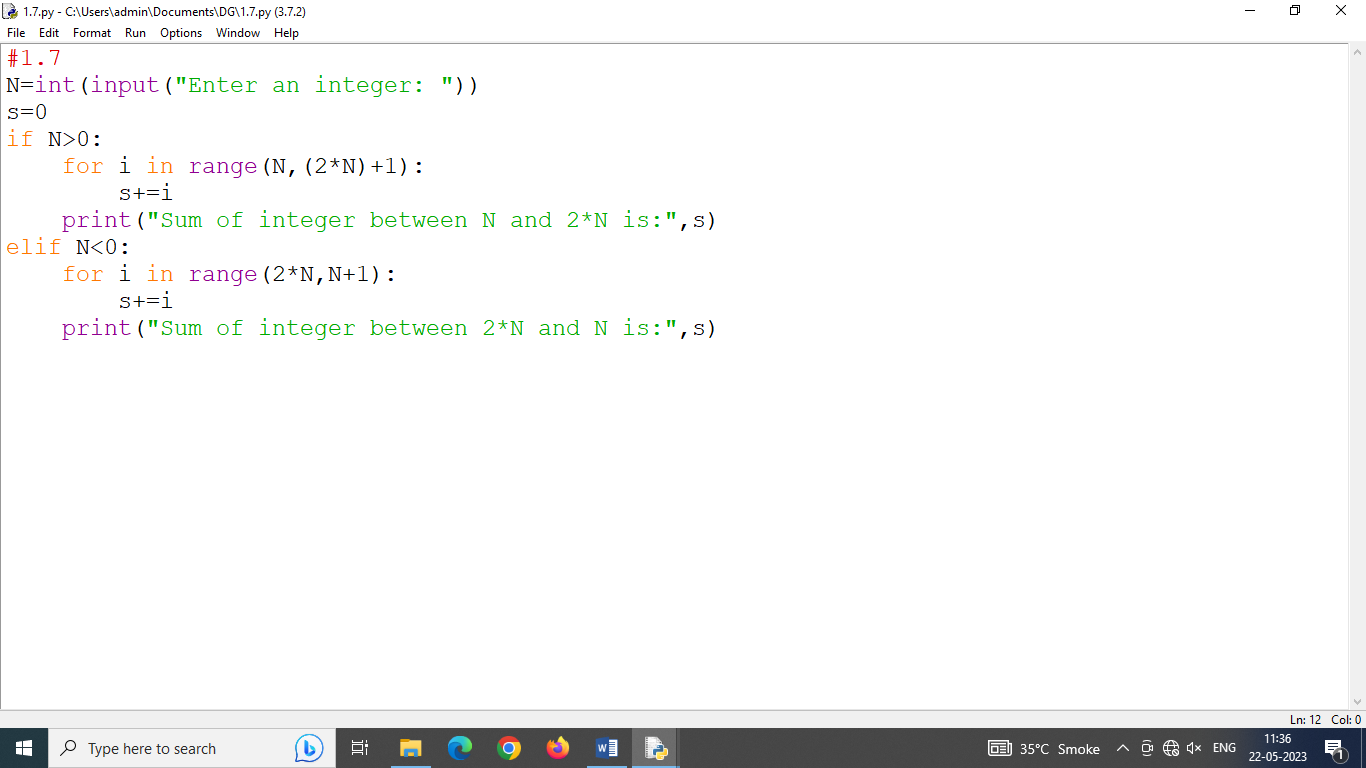
****

****

Program 1.6 **One foot equals 12 inches. Write a function that accepts a length written in feet as an argument and returns this length written in inches. Write a second function that asks the user for a number of feet and returns this value. Write a third function that accepts a number of inches and displays this to the screen. Use these three functions to write a program that asks the user for a number of feet and tells them the corresponding number of inches.**

****

Program 1.7 **Write a program that reads an integer N from the keyboard computes and displays the sum of the numbers from N to (2\*N) if N is nonnegative. If N is a negative number, then it’s the sum of the numbers from (2\*N) to N. The starting and ending points are included in the sum.**

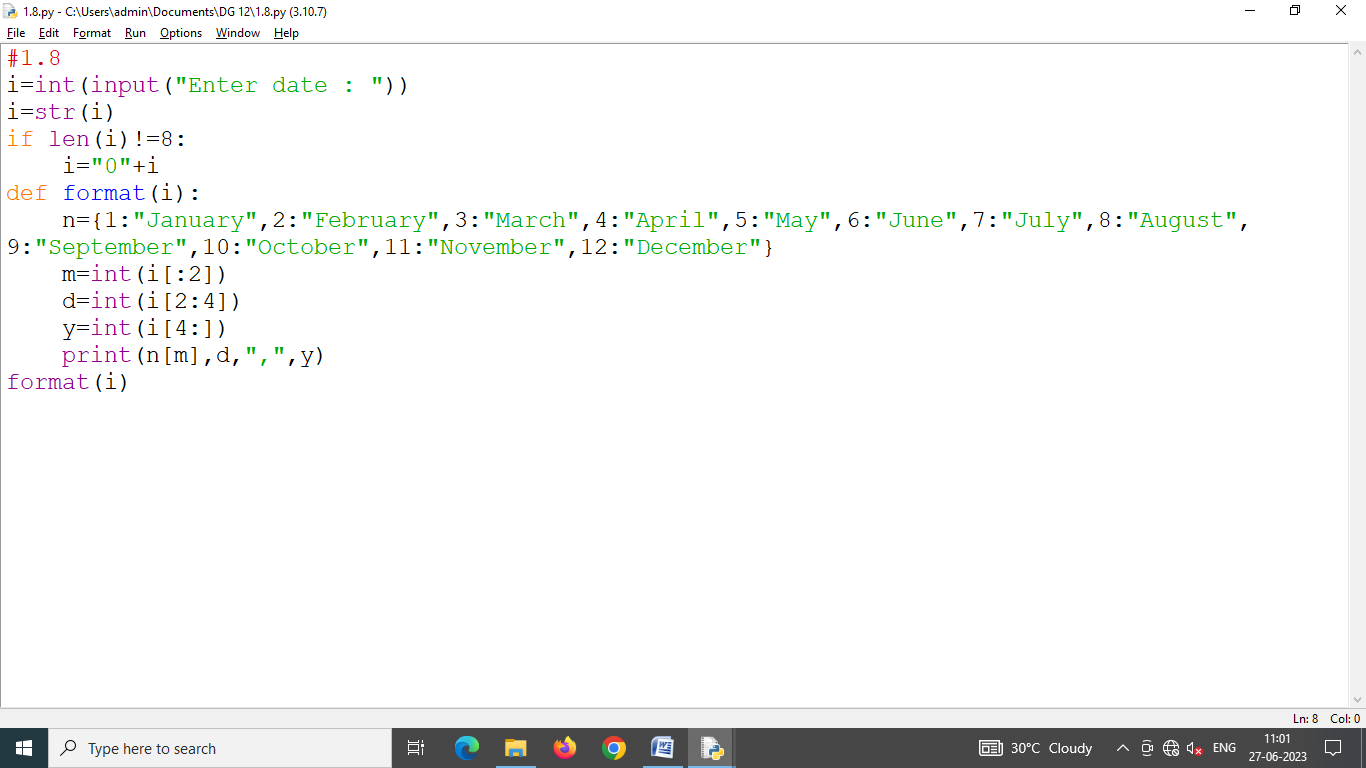


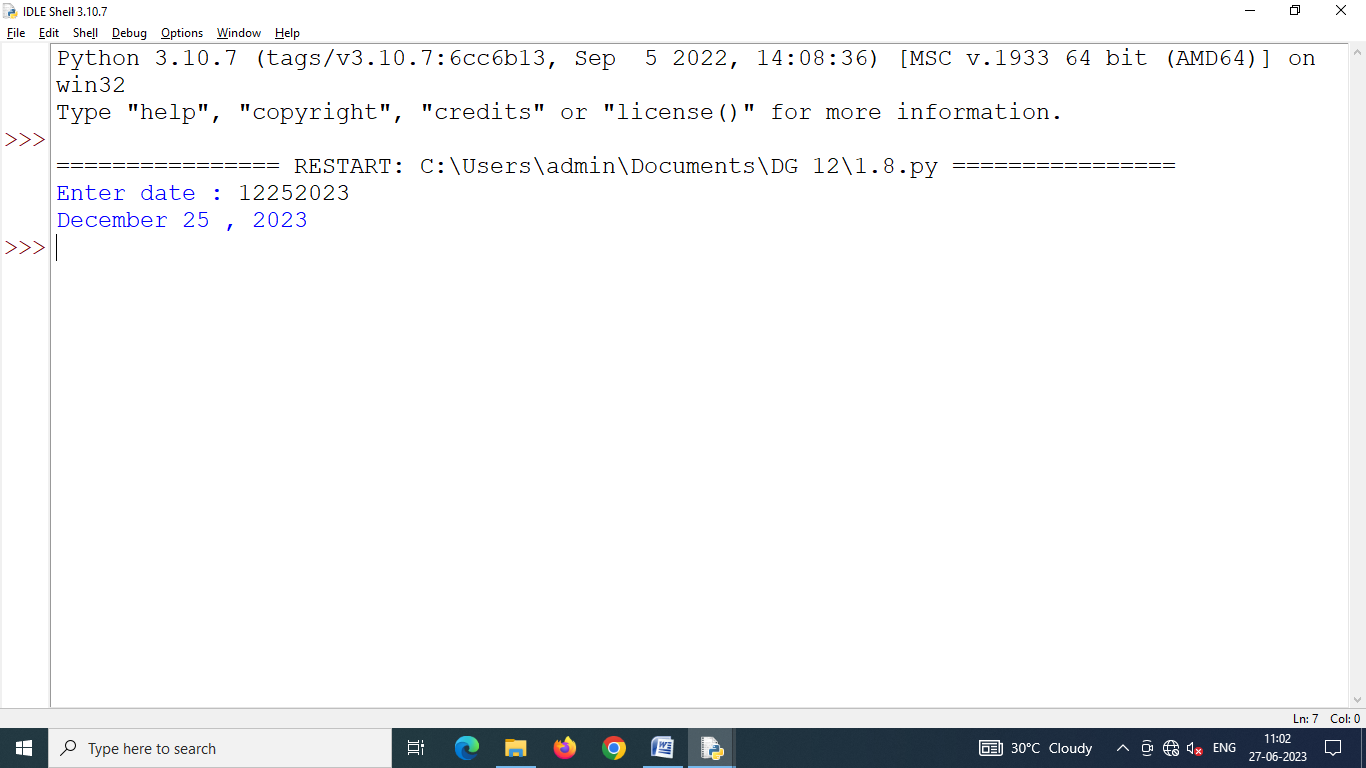


Program 1.8 **Write a program that reads a date as an integer in the format MMDDYYYY. The program will call a function that prints print out the date in the format <Month Name><day>, <year>.**

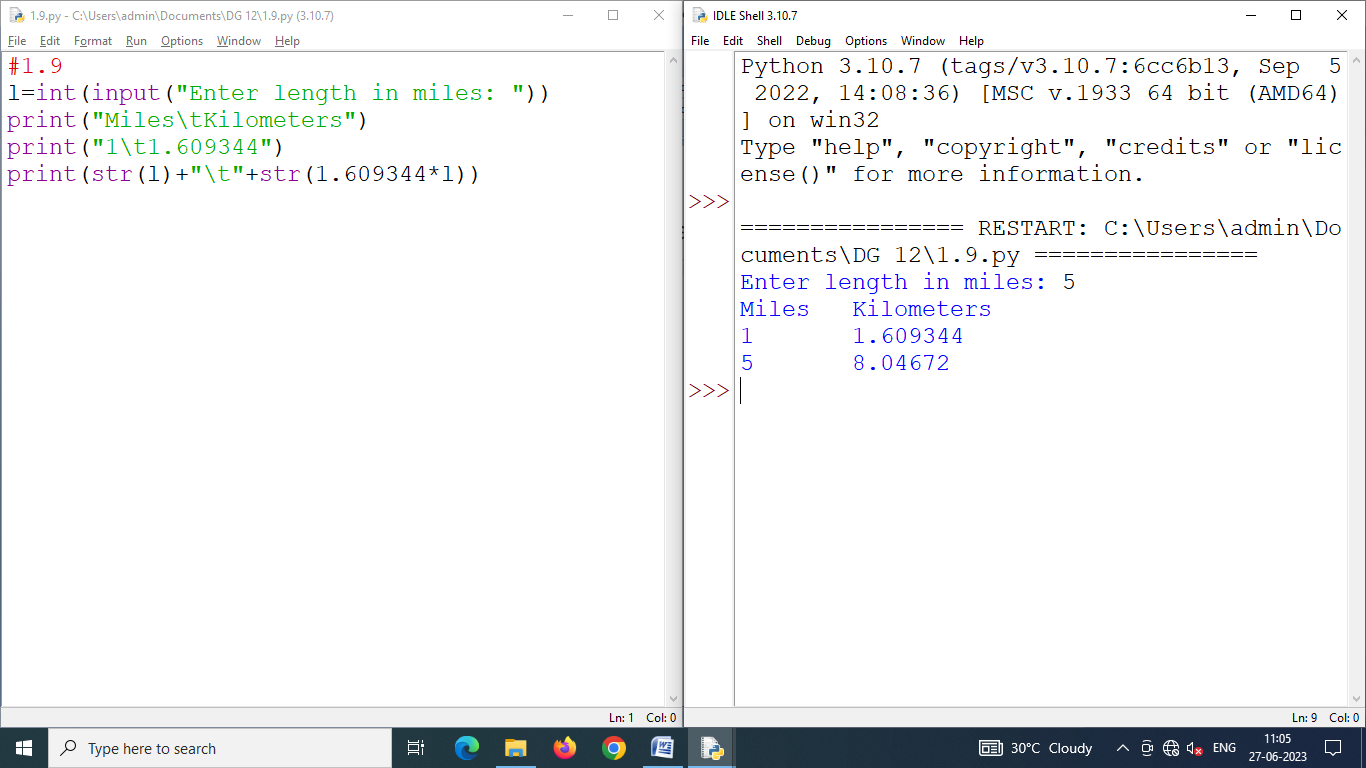
**Sample run: Enter date : 12252023**

**December 25, 2023**

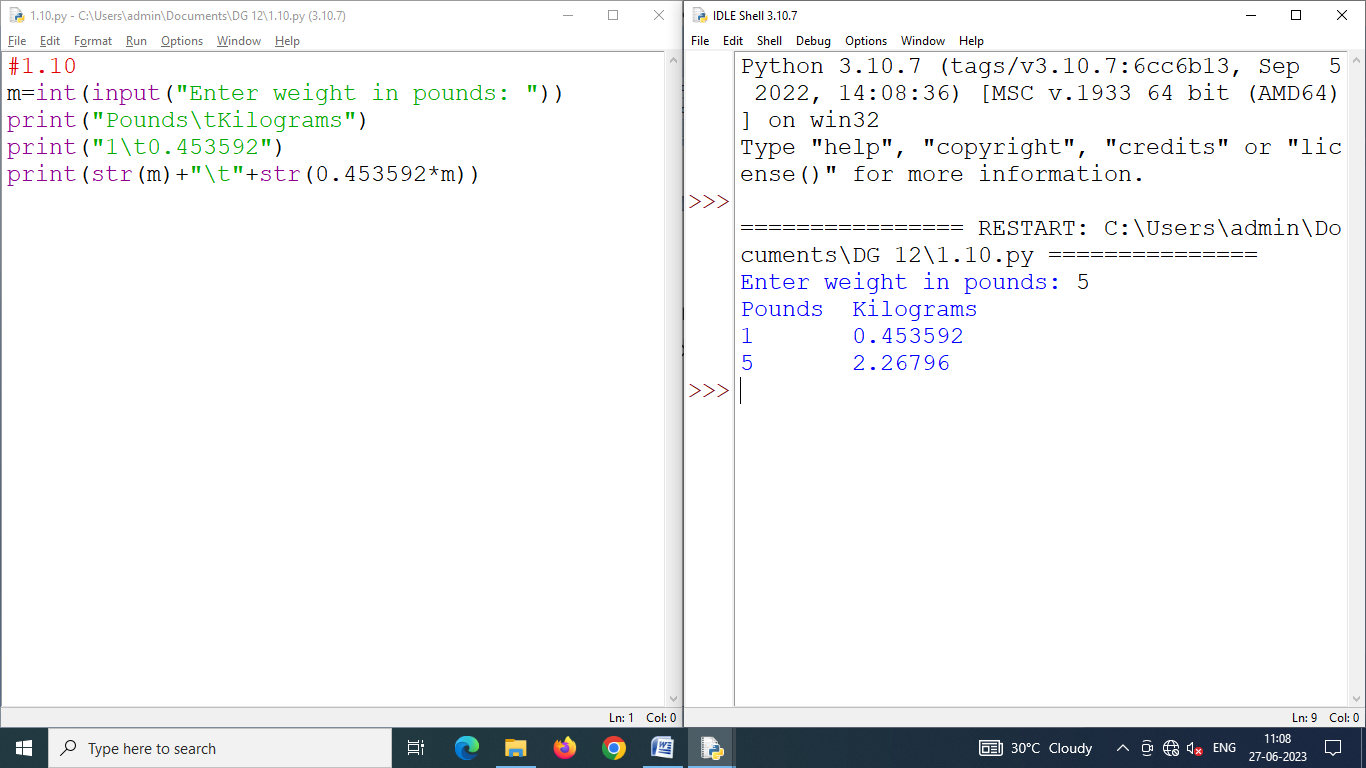
****

****

Program 1.9 **Write a program that prints a table on two columns - table that helps converting miles into kilometres.**

****

Program 1.10 **Write another program printing a table with two columns that helps convert pounds in kilograms.**

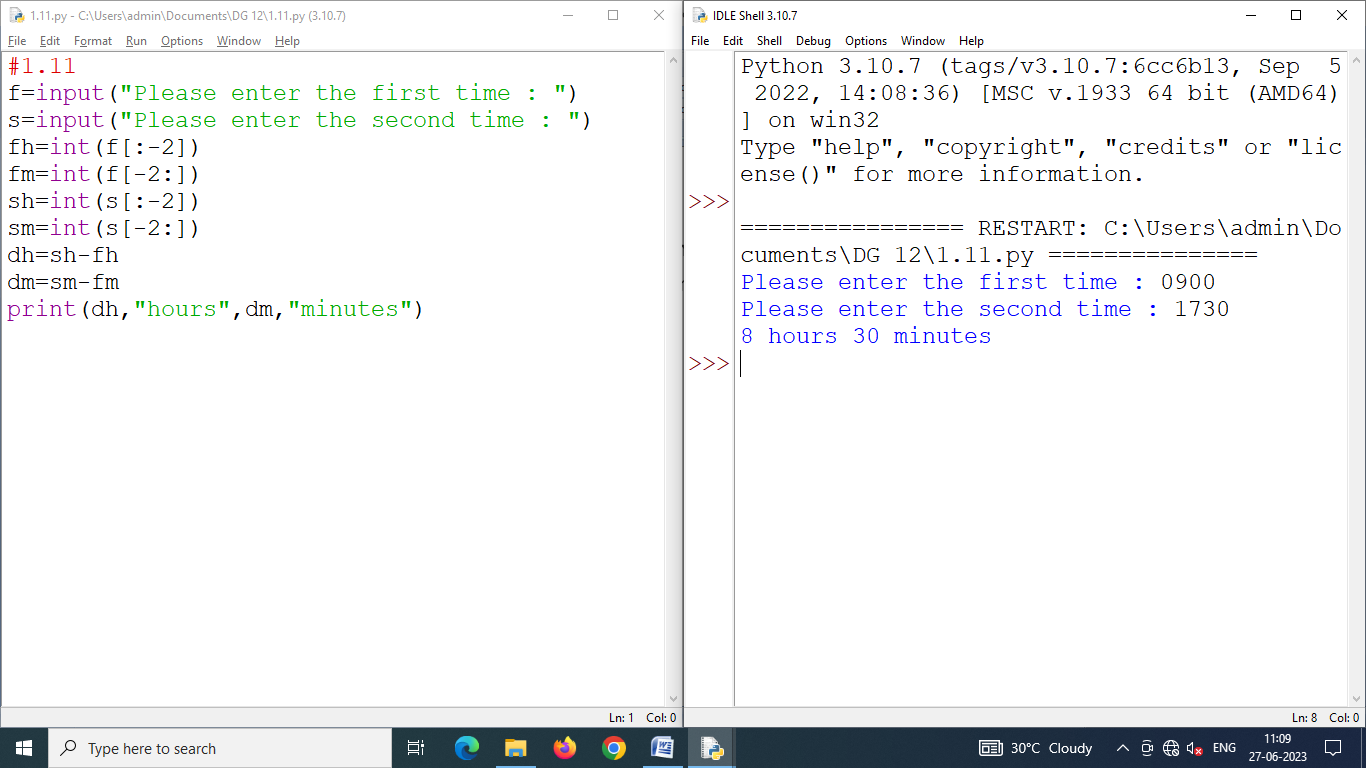
****

Program 1.11 **Write a program that reads two times in military format (0900, 1730) and prints the number of hours and minutes between the two times. A sample run is being given below :**

**Please enter the first** **time : 0900**

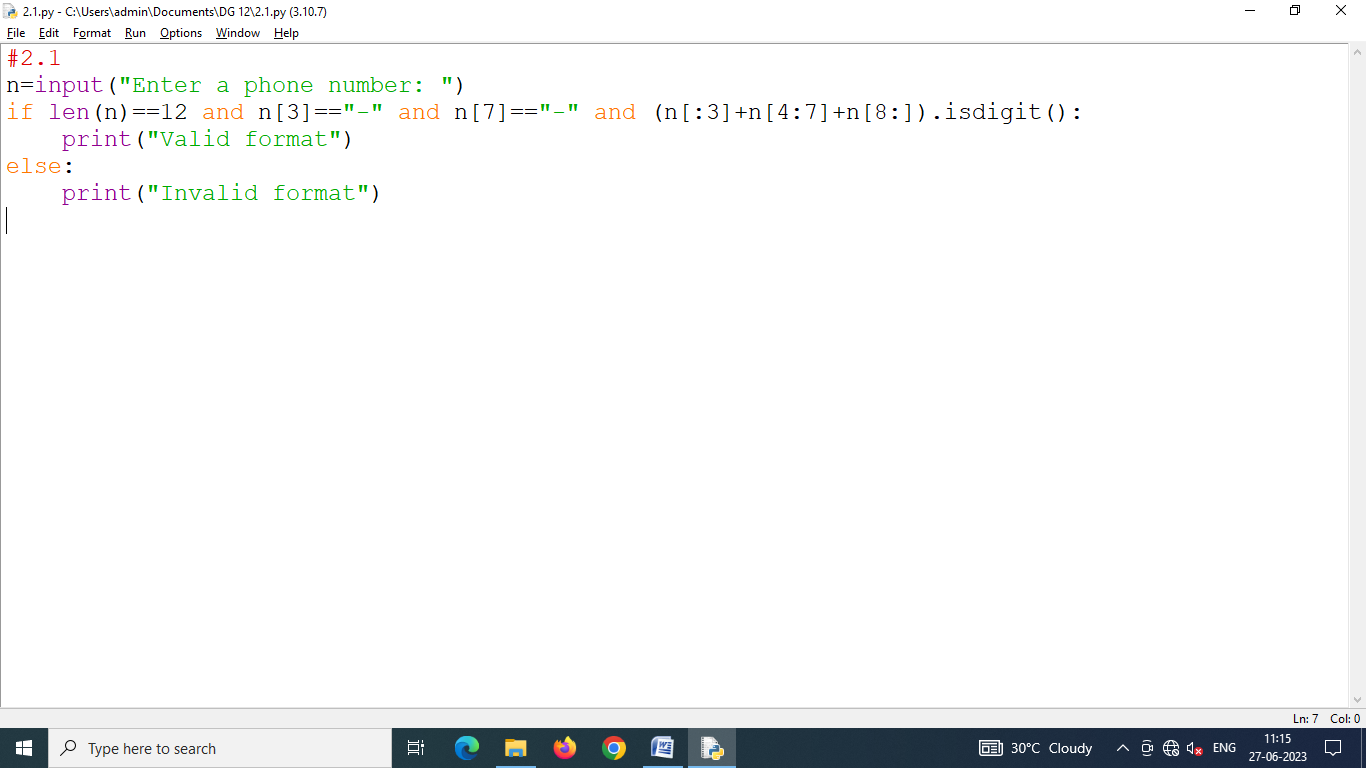
**Please enter the second** **time : 1730**

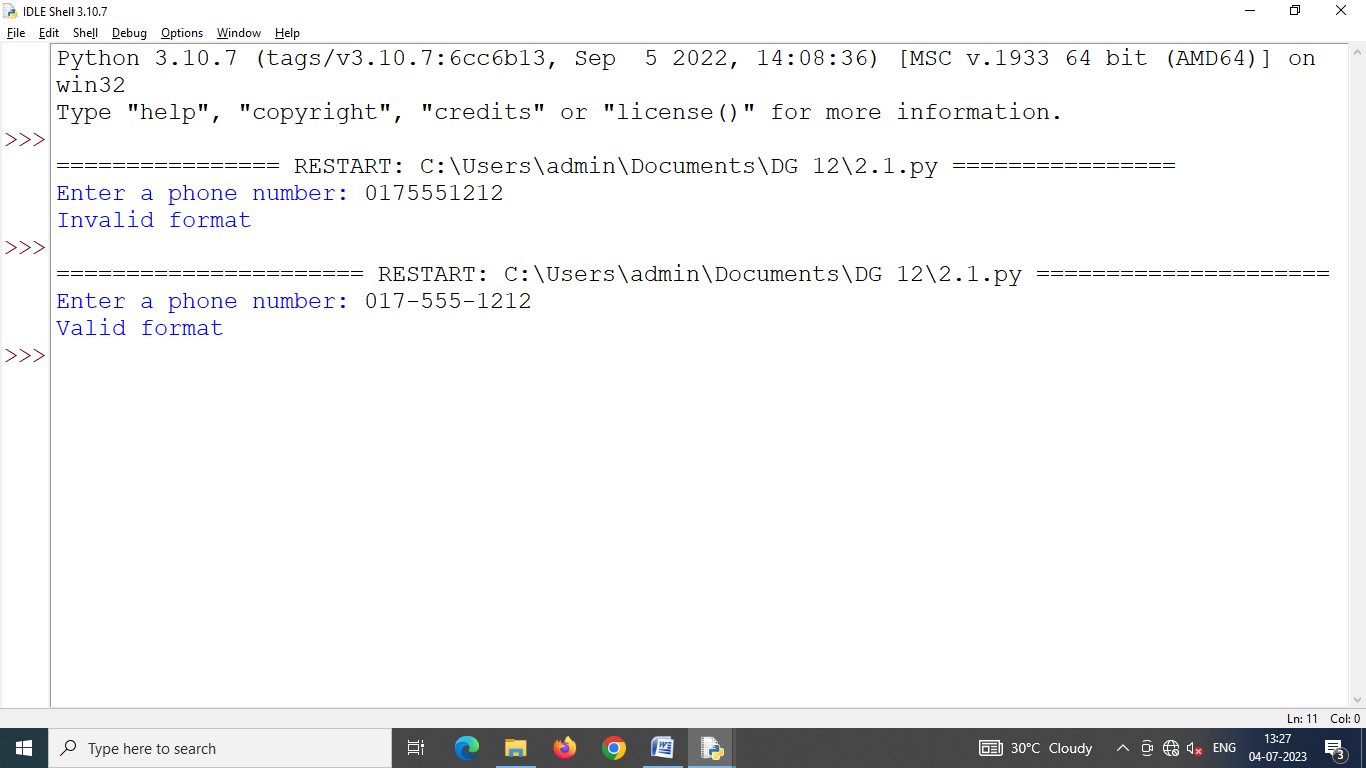
**8 hours 30 minutes**

****

Program 2.1 **Write a program that prompts for a phone number of 10 digits and two dashes, with dashes after the area code and the next three numbers. For example, 017-555-1212 is a legal input.**

**Display if the phone number entered is valid format or not and display if the phone number is valid or not (i.e., contains just the digits and dash at specific places).**



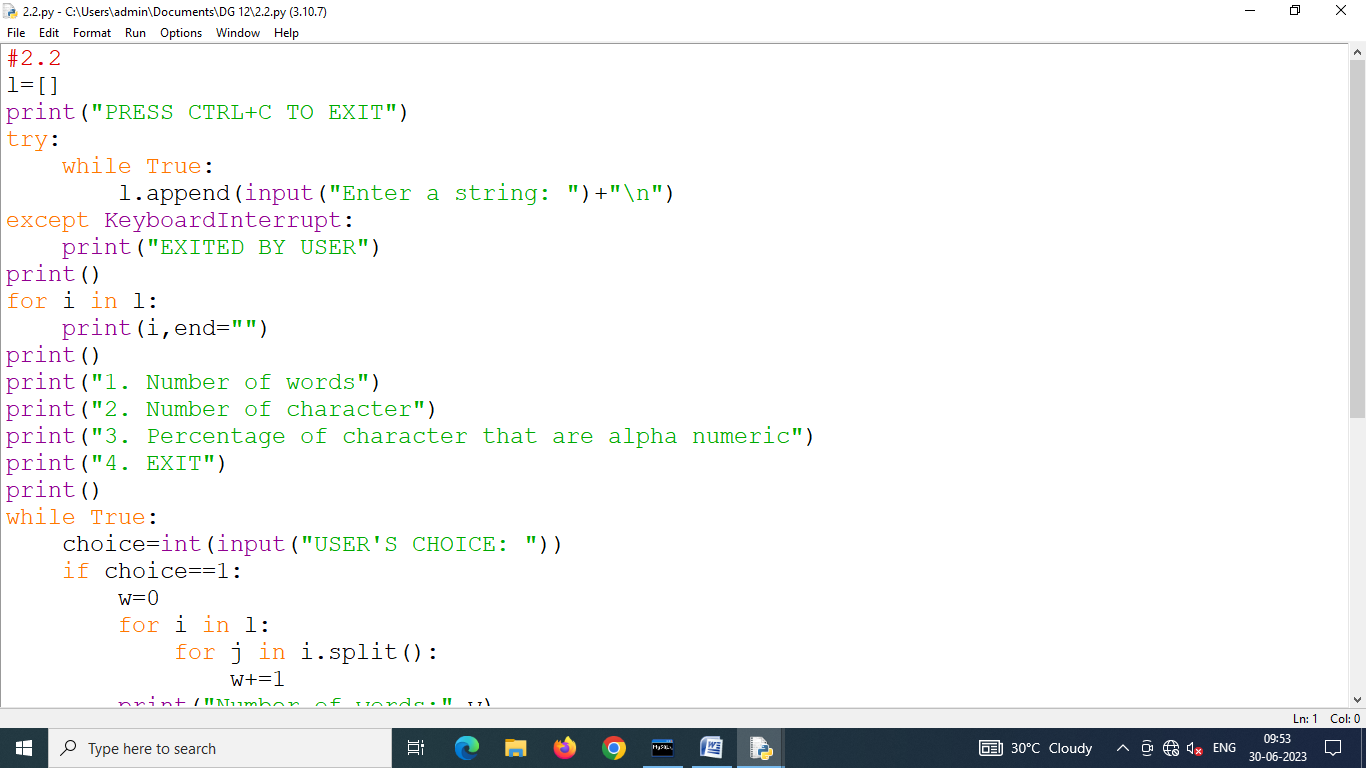


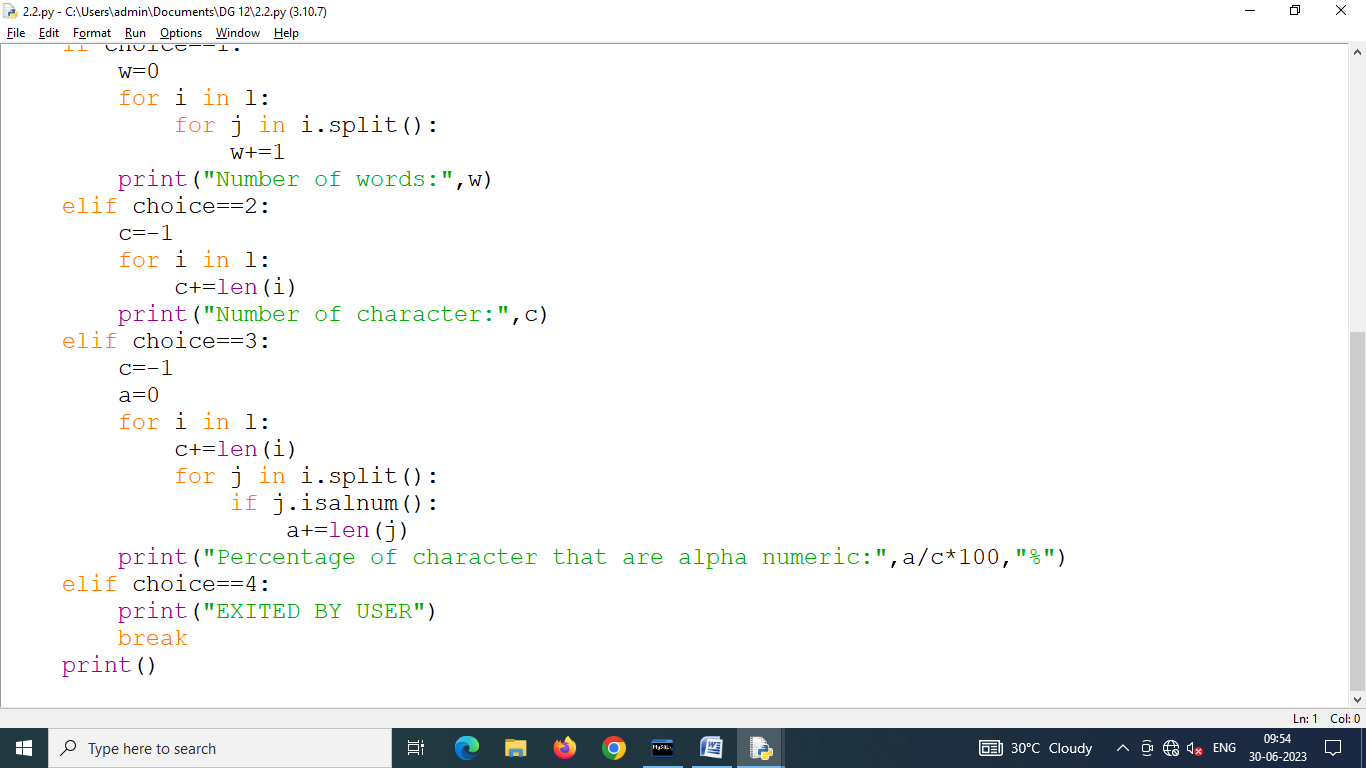
Program 2.2 **Write a program that should prompt the user to type some sentence(s) followed by "enter". It should then print the original sentence(s) and the following statistics relating to the sentence(s****) :**

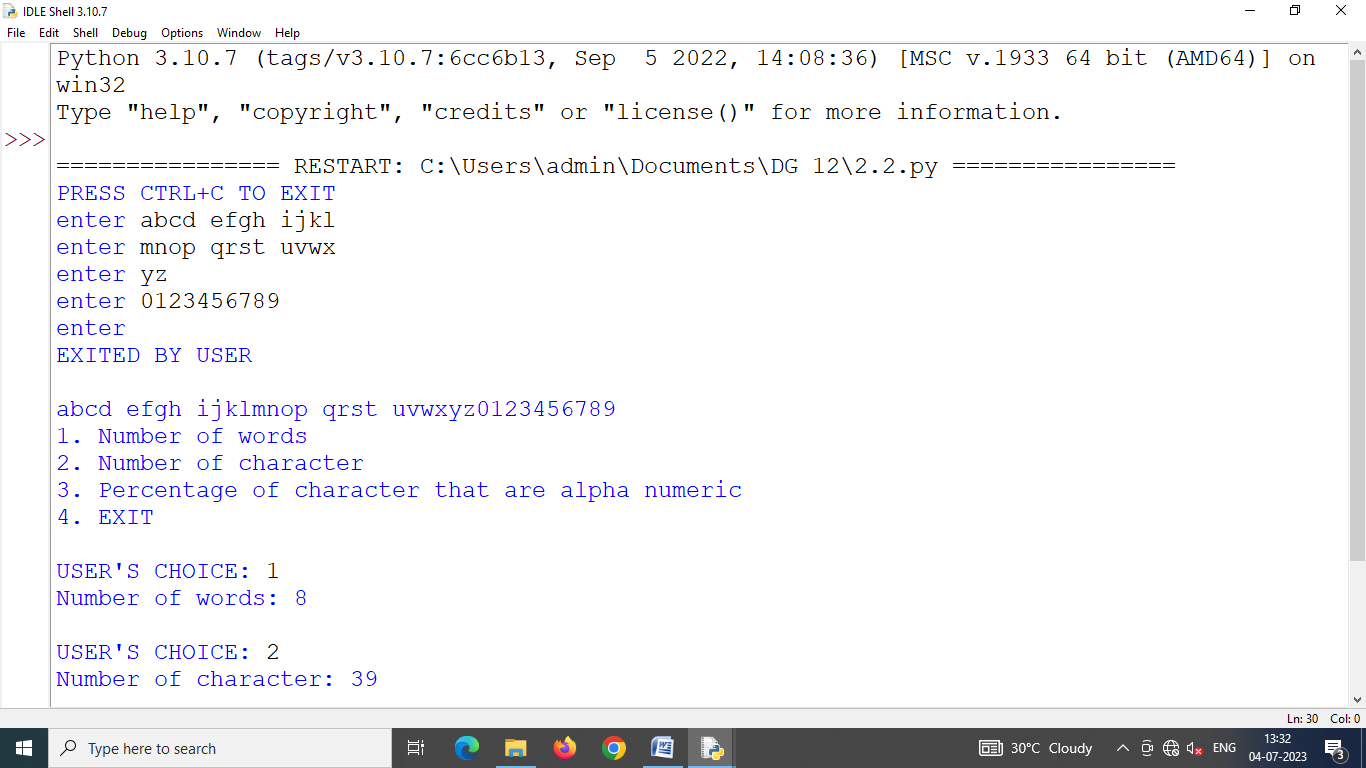
* **Number of words**
* **Number of characters (including white-space and punctuation)**
* **Percentage of characters that are alpha numeric**

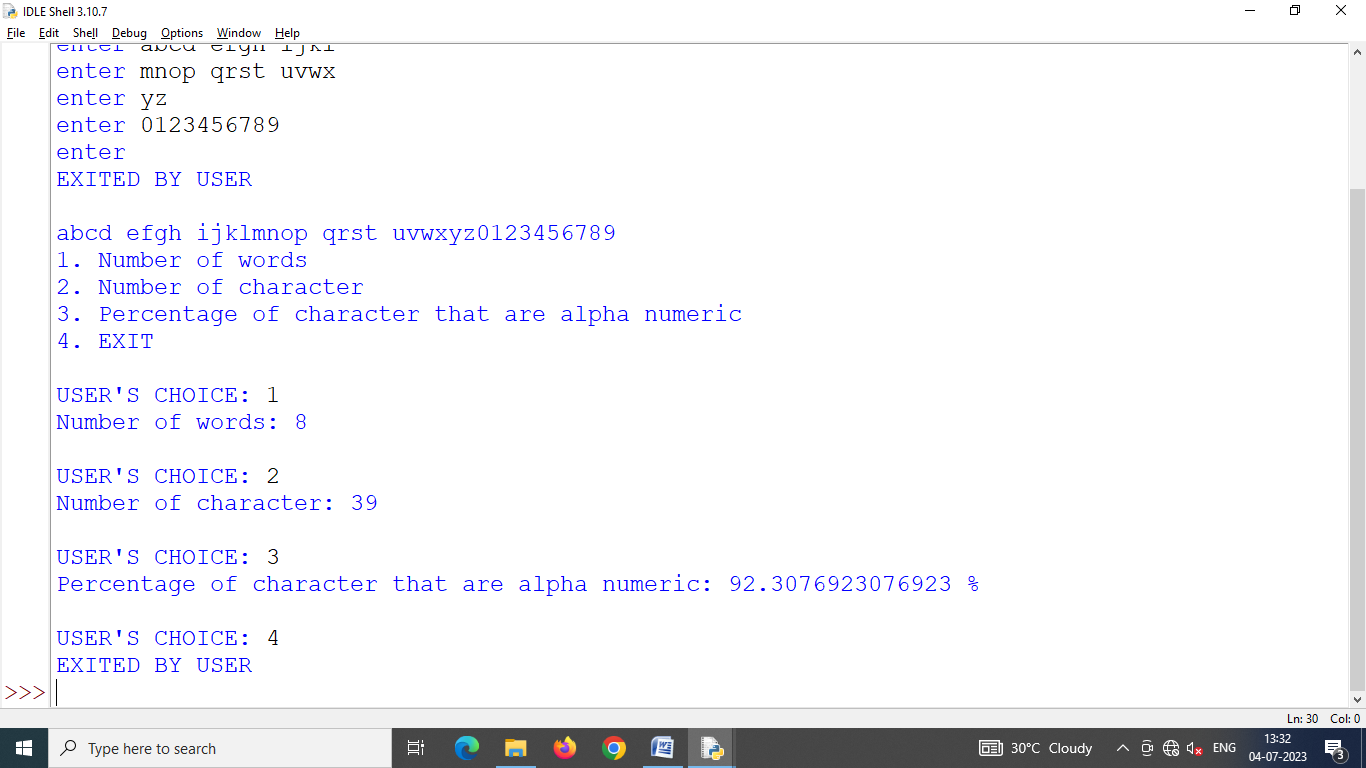
**Hints**

* **Assume any consecutive sequence of non-blank characters in a word.**

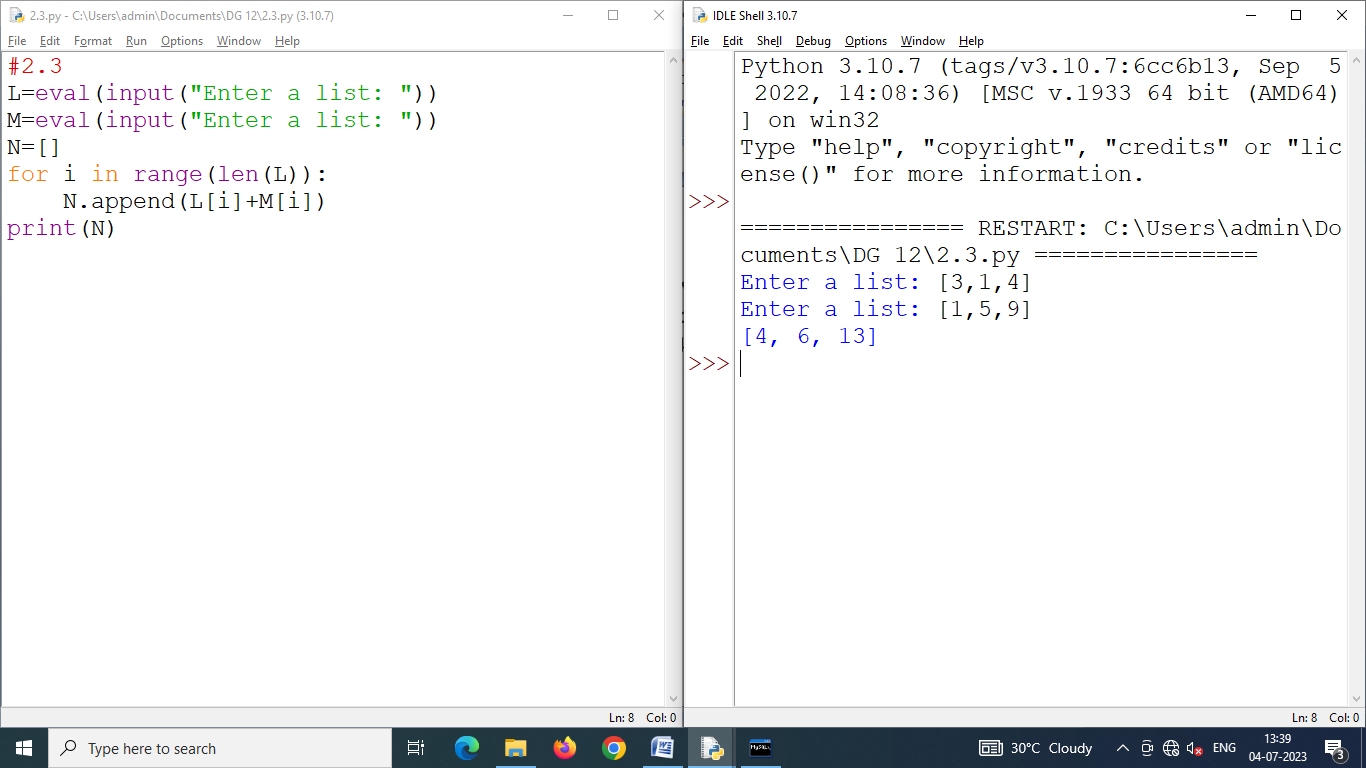
****

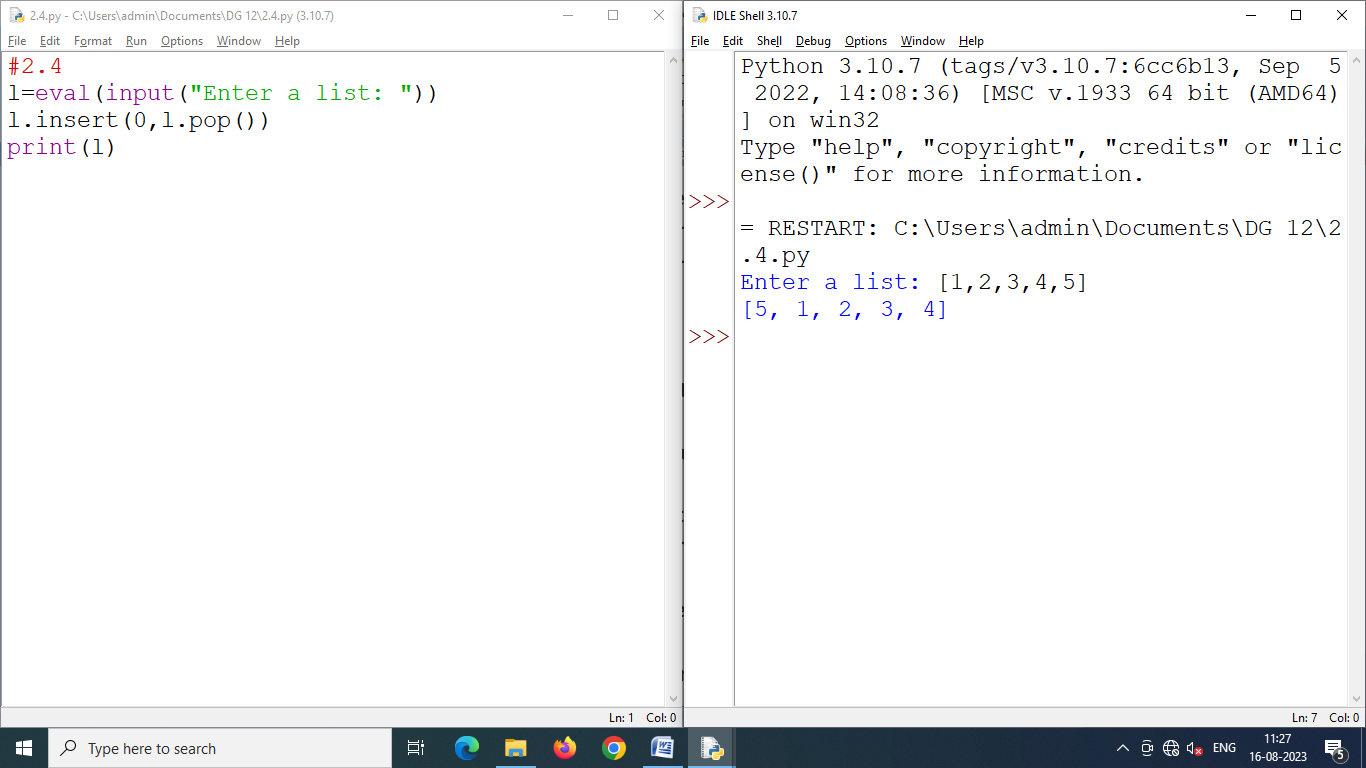
****

****

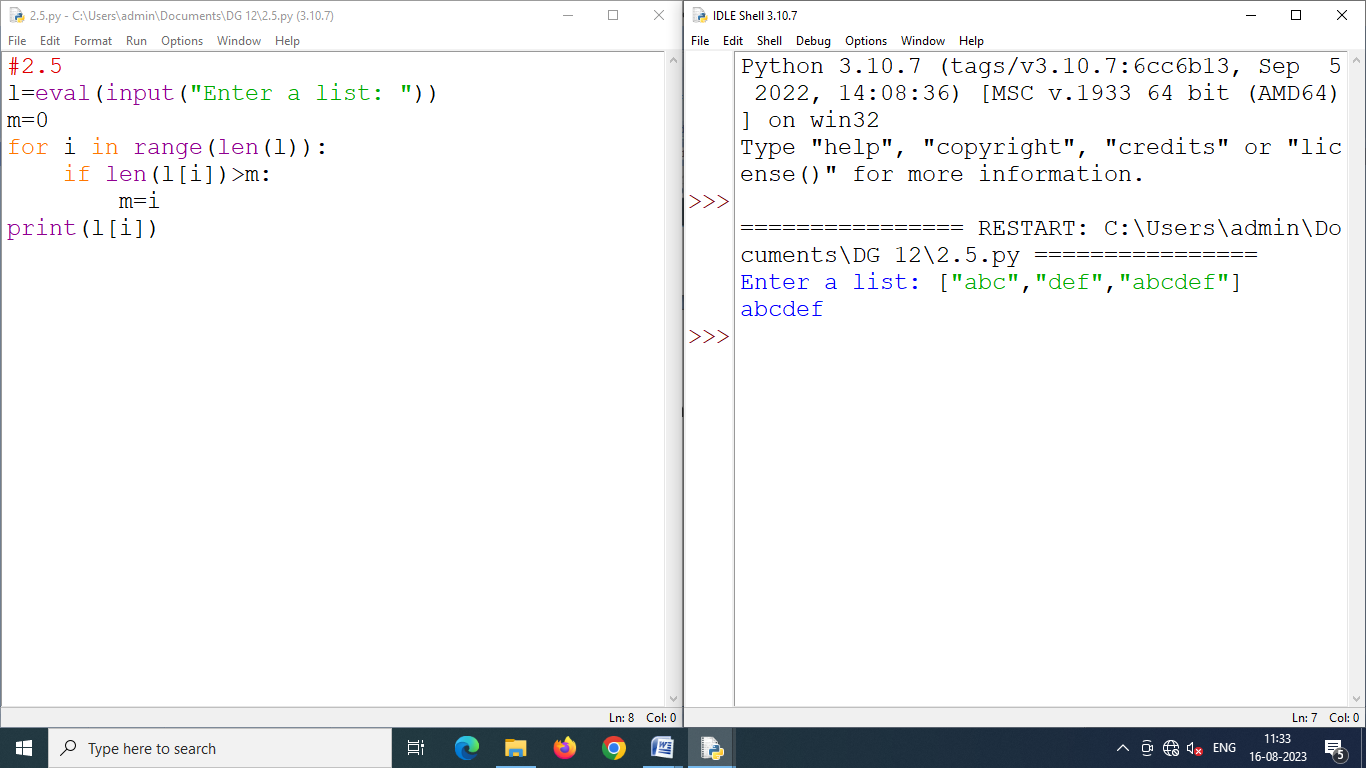
****

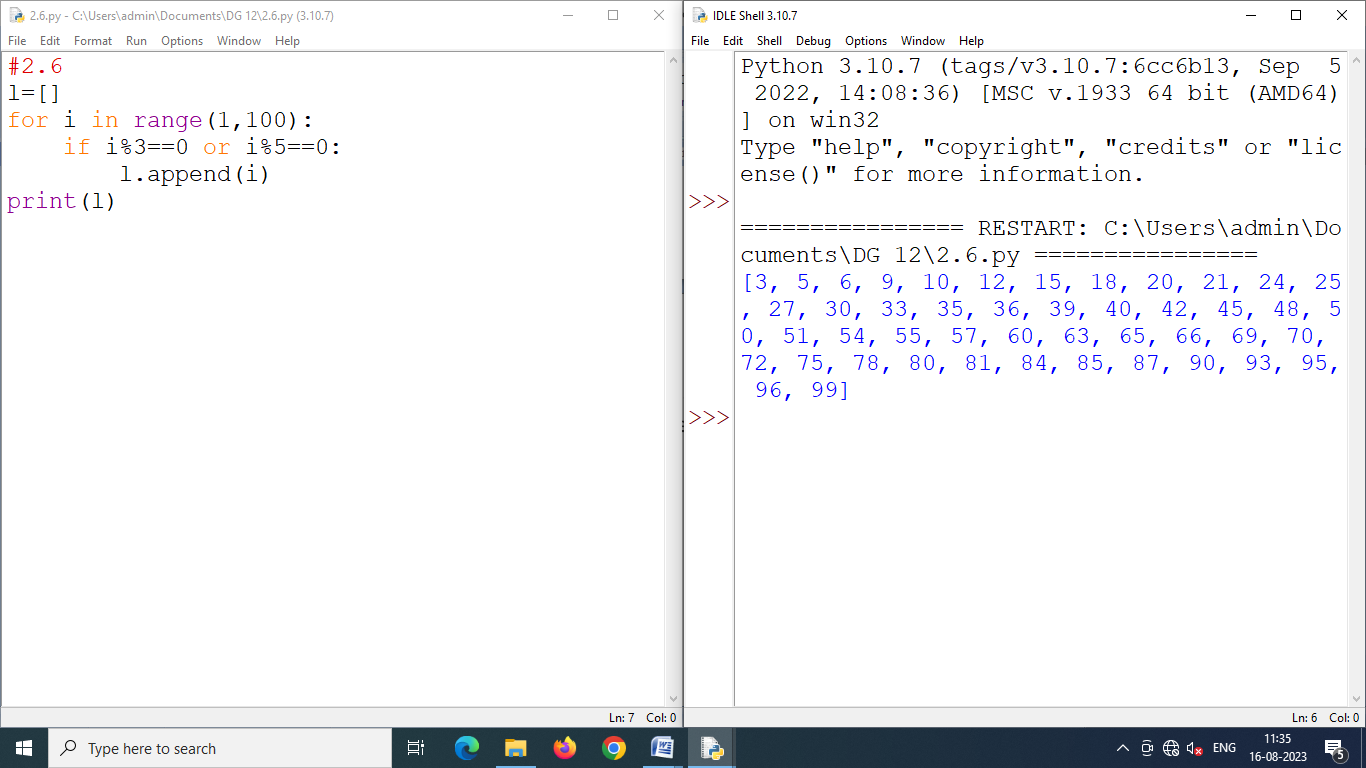
Program 2.3 **Write a program that takes any two lists L and M of the same size and adds their elements together to form a new list N whose elements are sums of the corresponding elements in L and M. For instance, if L****=[3,1,4] and M****=[1,5,9] then N should equal [4,6,13].**

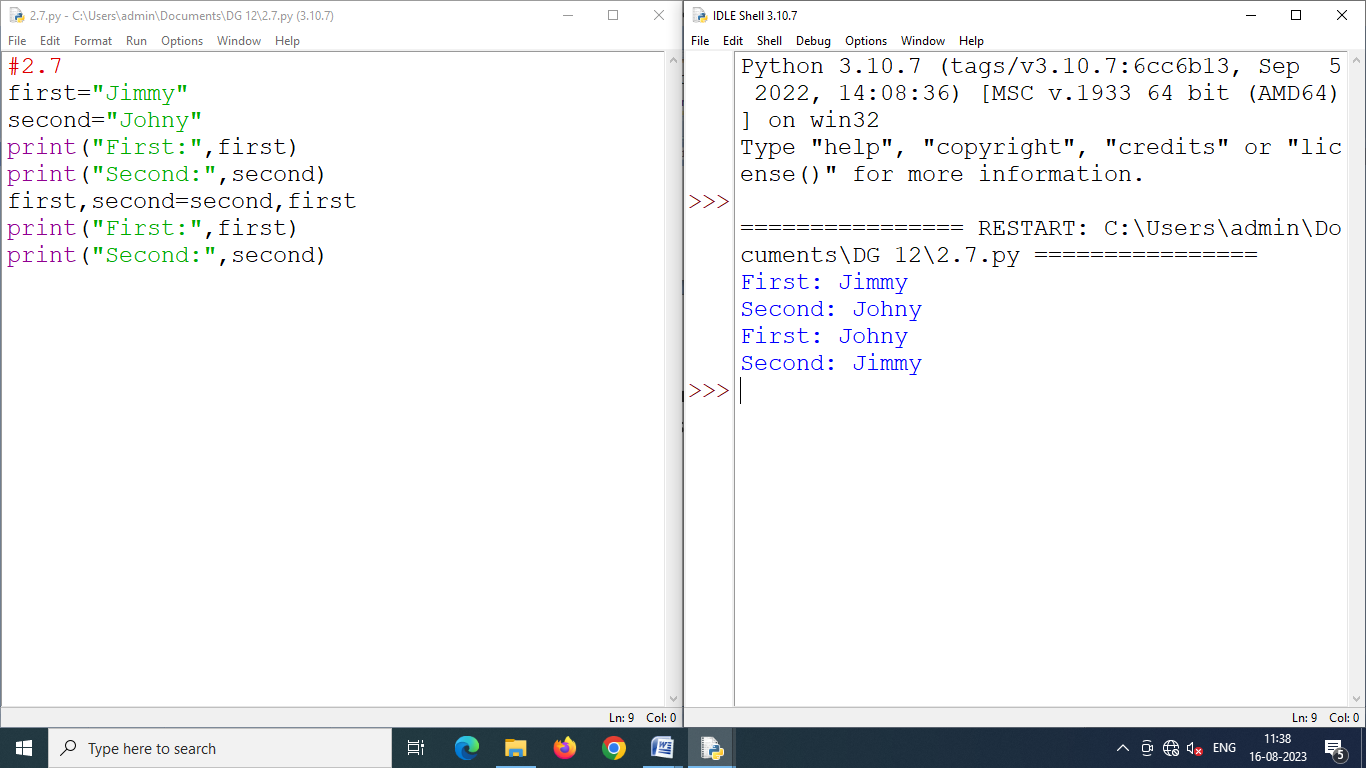
****Program 2.4 **Write a program that rotates the elements of a list so that the element at the first index moves to the second index, the element in the second index moves to the third index, etc., and the element in the last index moves to the first index.**

****

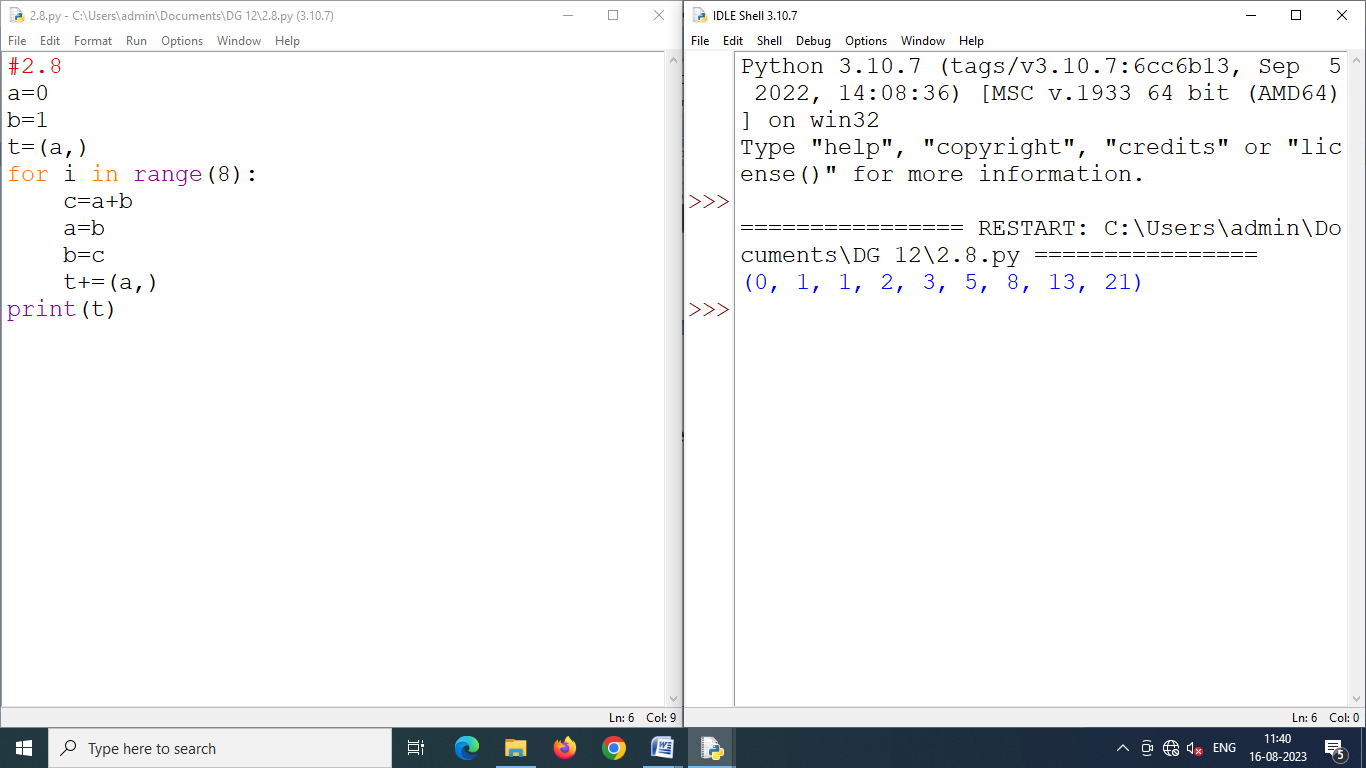
Program 2.5 **Write a short Python code segment that prints the longest word in a list of words.**

Program 2.6 **Write a program that creates a list of all the integers less than 100 that are multiples of 3 or 5.**

Program 2.7 **Define two variables first and second so that first = "Jimmy" and second = "Johny". Write a short Python code segment that swaps the values assigned to these two variables and prints the results.**

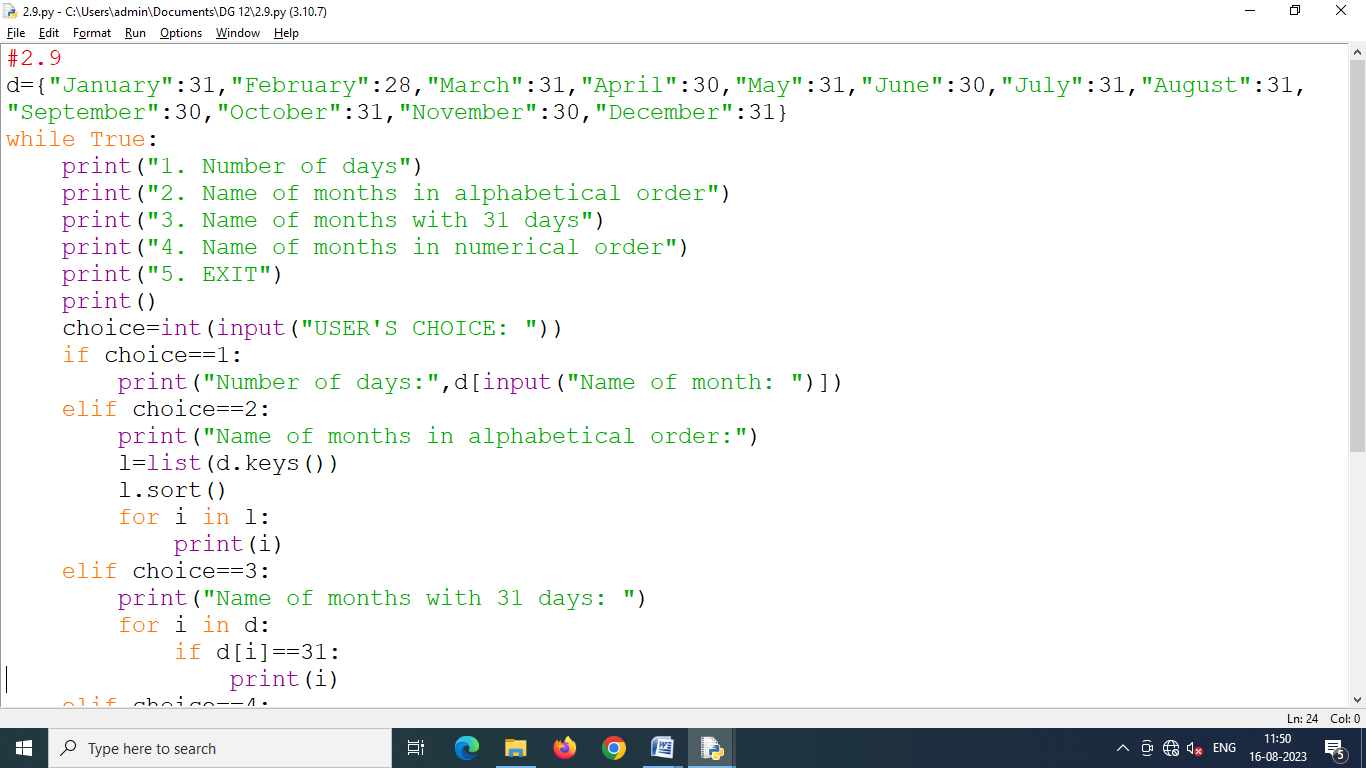
****

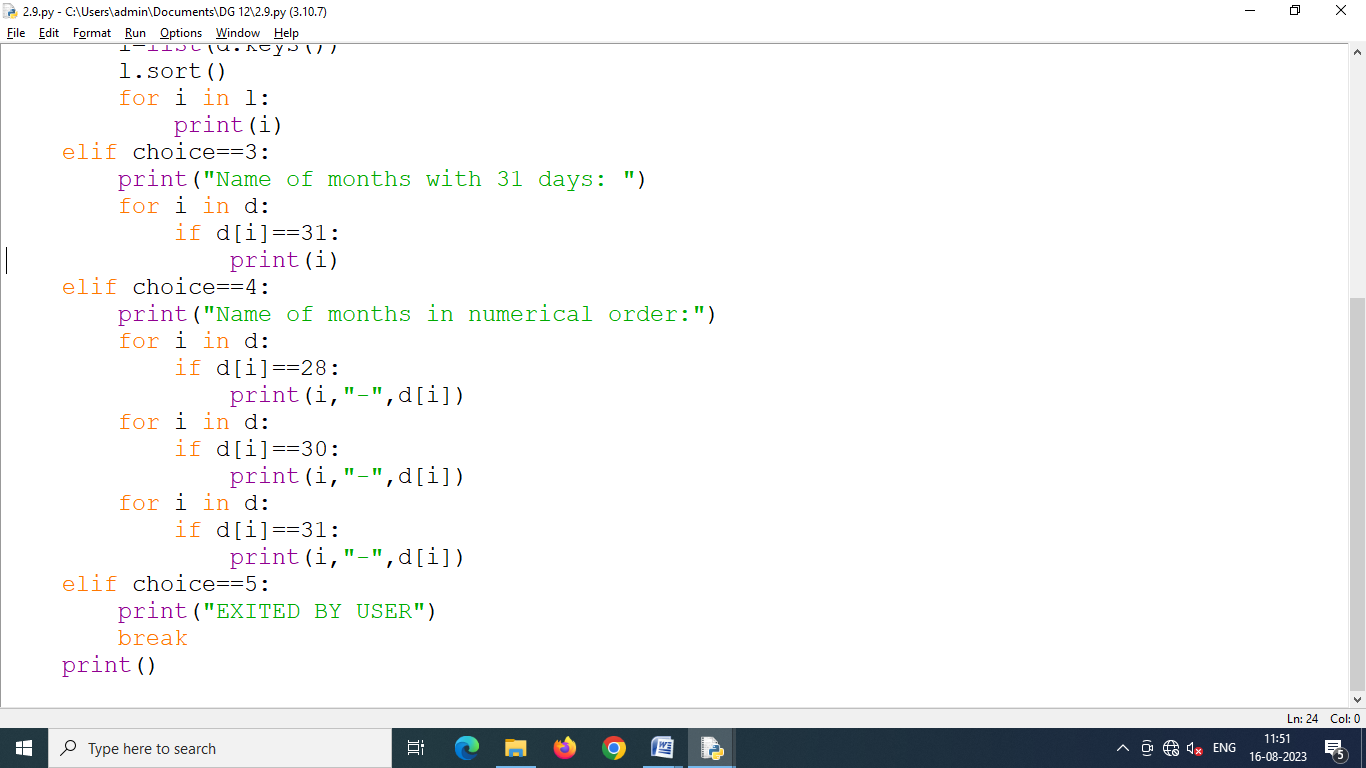
Program 2.8 **Write a Python program that creates a tuple storing first 9 terms of Fibonacci series.**

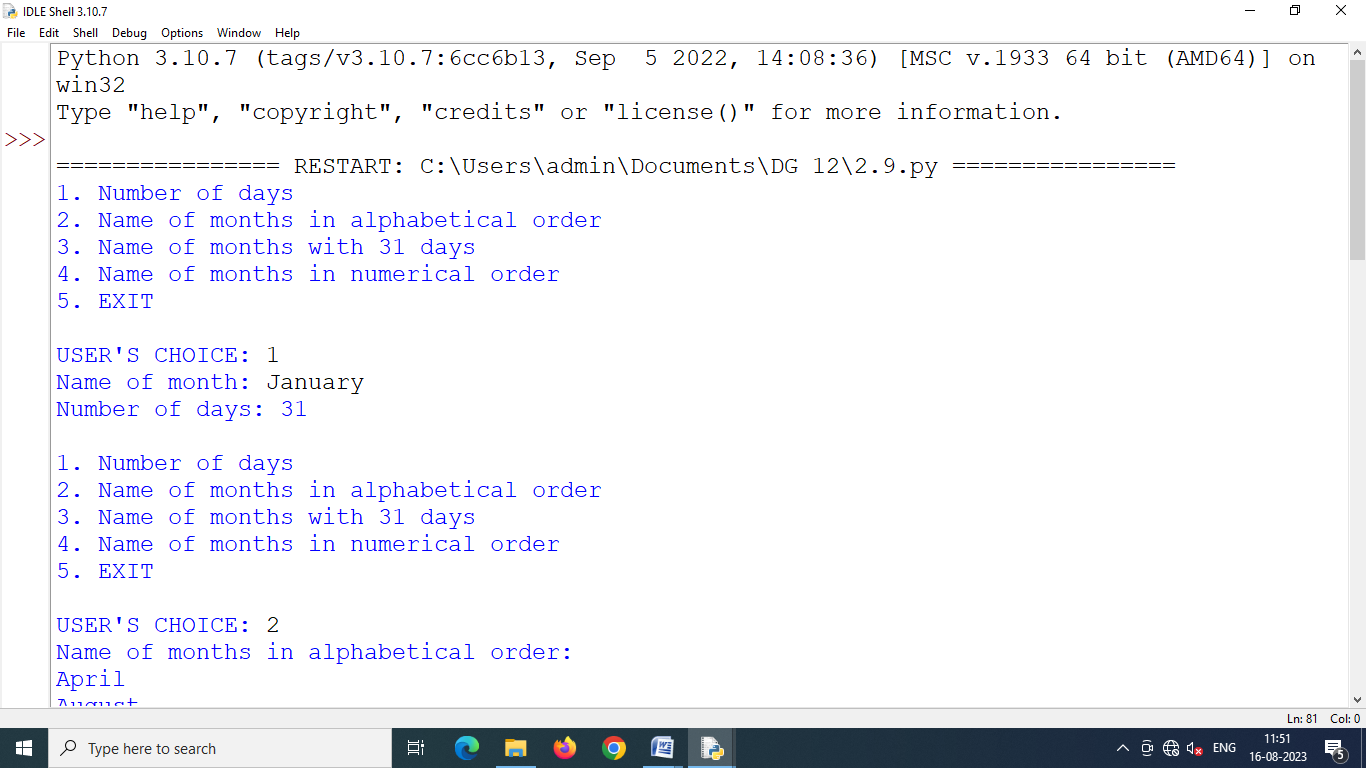
****

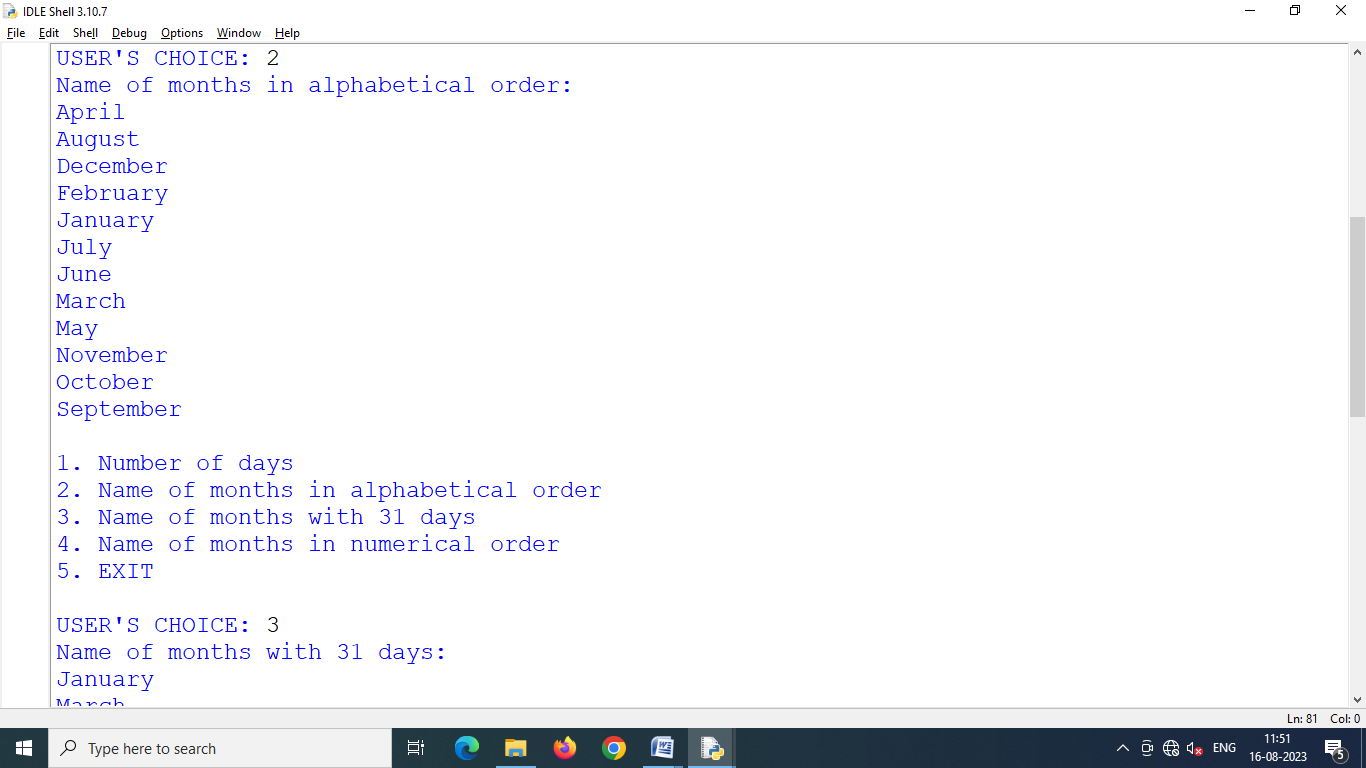
Program 2.9 **Create a dictionary whose keys are month names and whose values are the number of days in the corresponding months.**

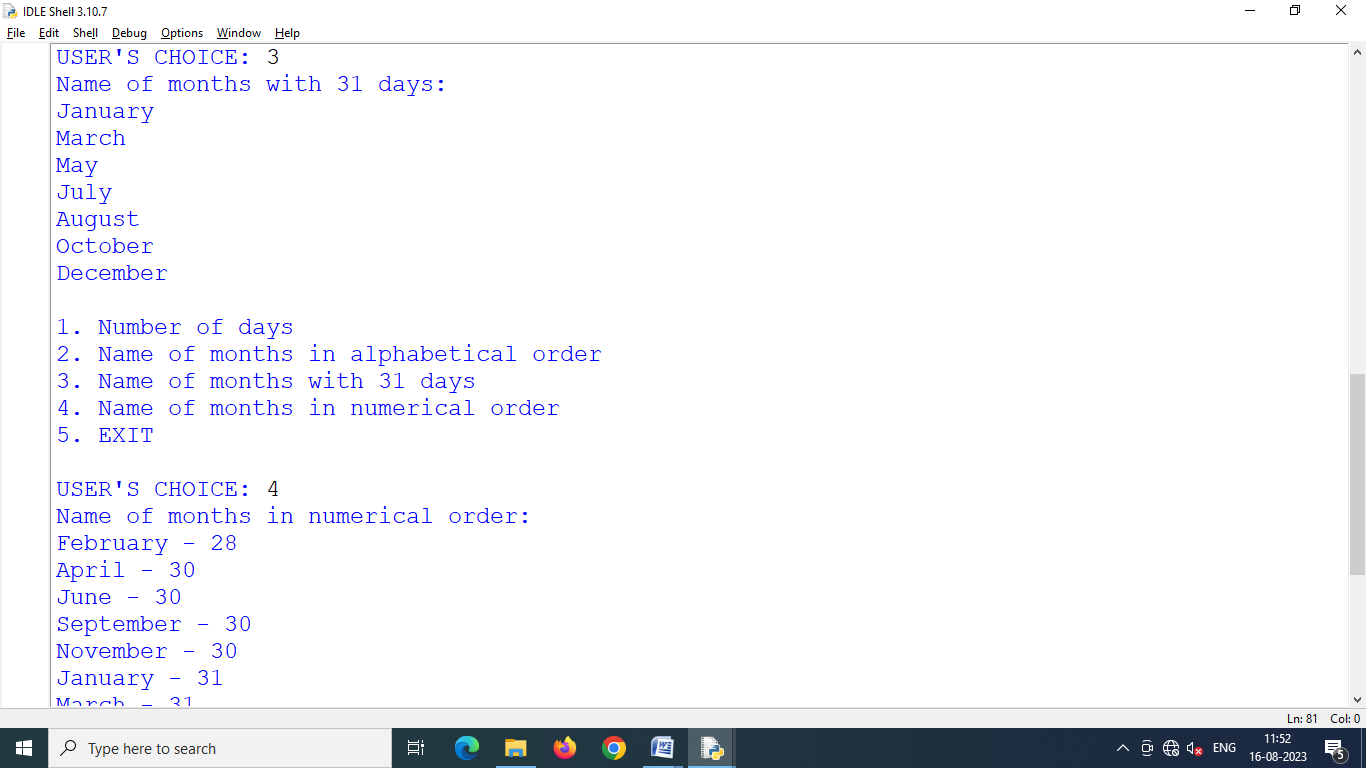
1. **Ask the user to enter a month name and use the dictionary to tell them how many days are in the month.**
2. **Print out all of the keys in alphabetical order.**
3. **Print out all of the months with 31 days.**
4. **Print out the (key-value) pairs sorted by the number of days in each month.**

****

****

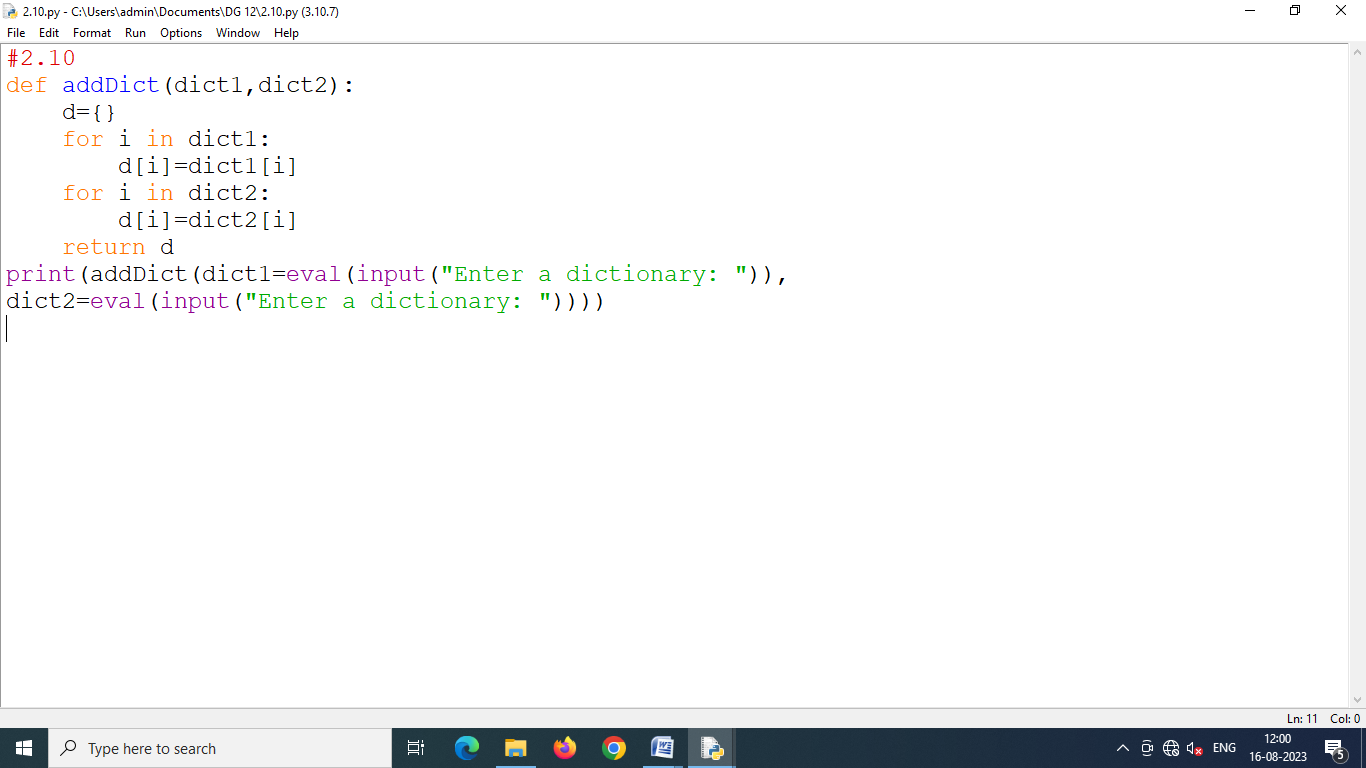
****

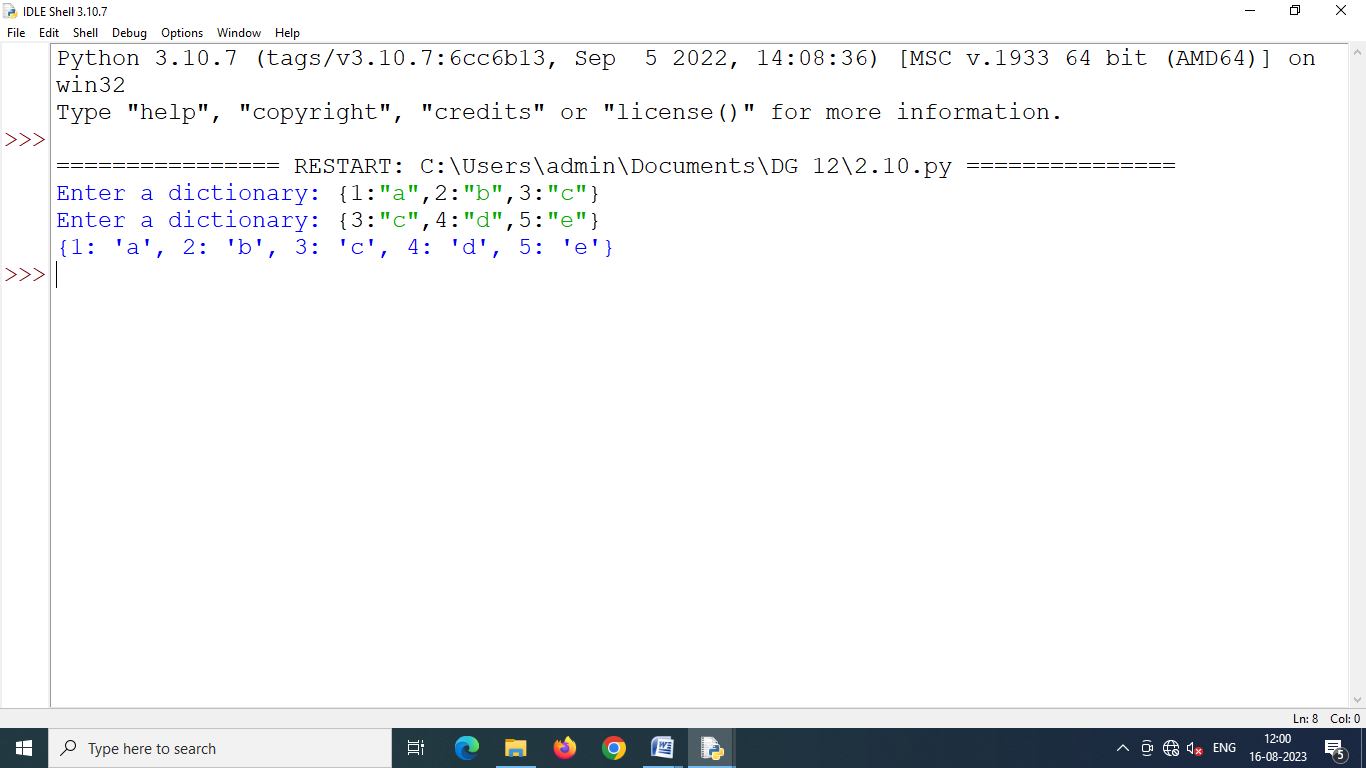
****

****

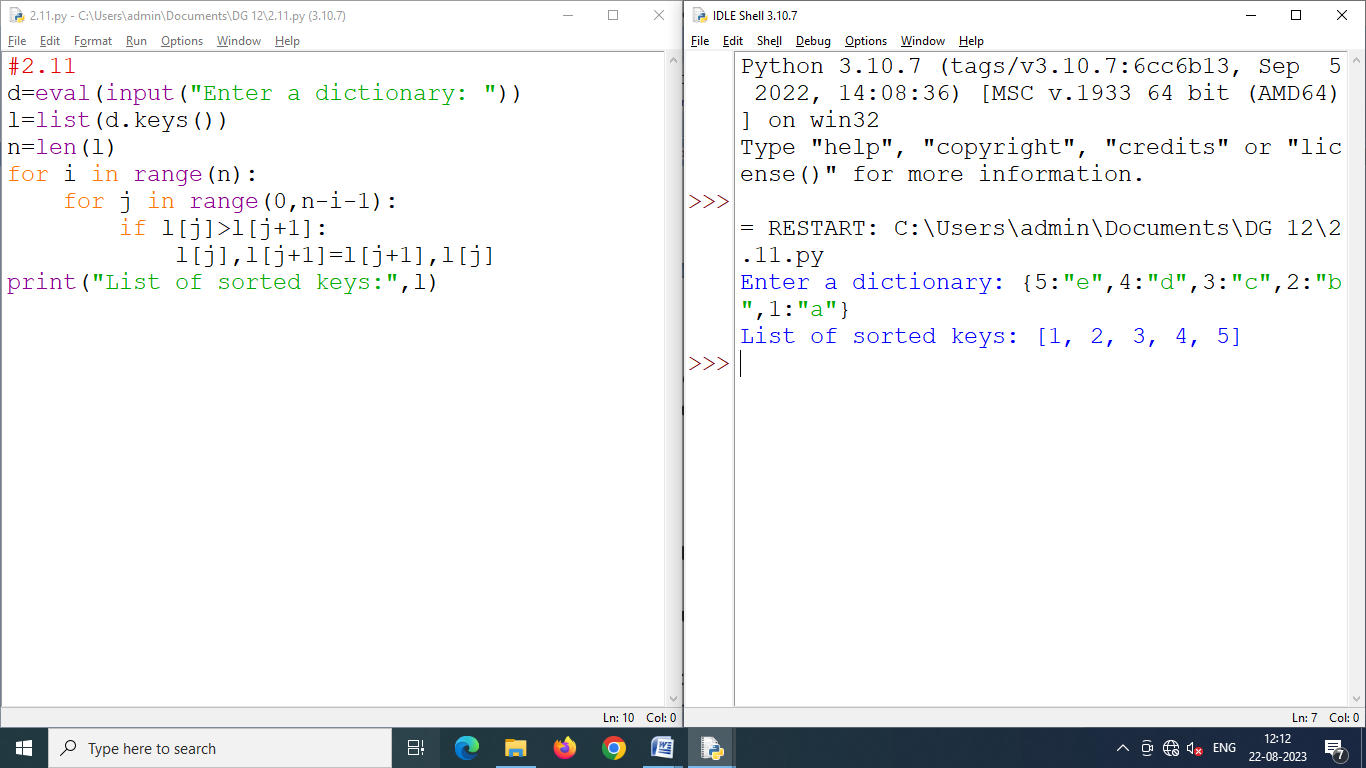
****

Program 2.10 **Write a function called addDict(dict1, dict2) which computes the union of two dictionaries. It should return a new dictionary, with all the items in both its arguments (assumed to be dictionaries). If the same key appears in both arguments, feel free to pick a value from either.**

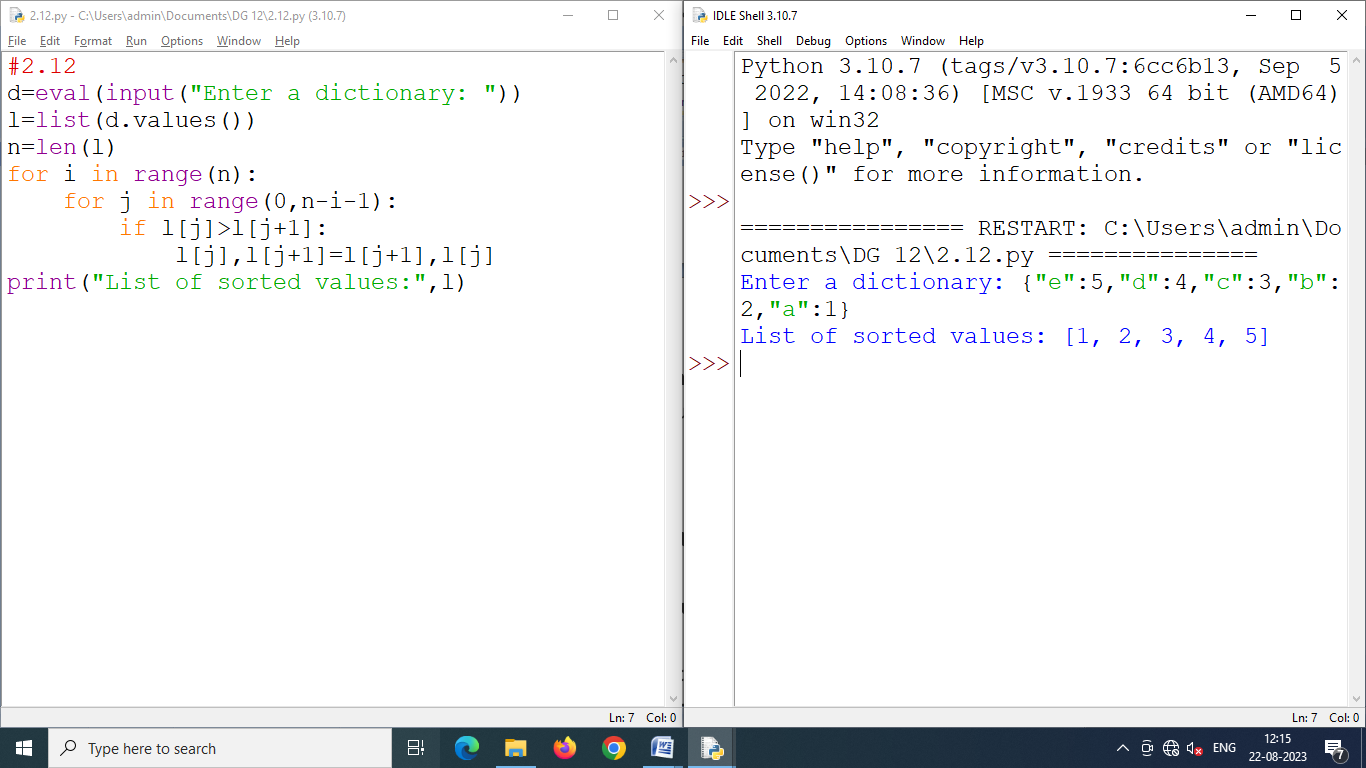
****

****

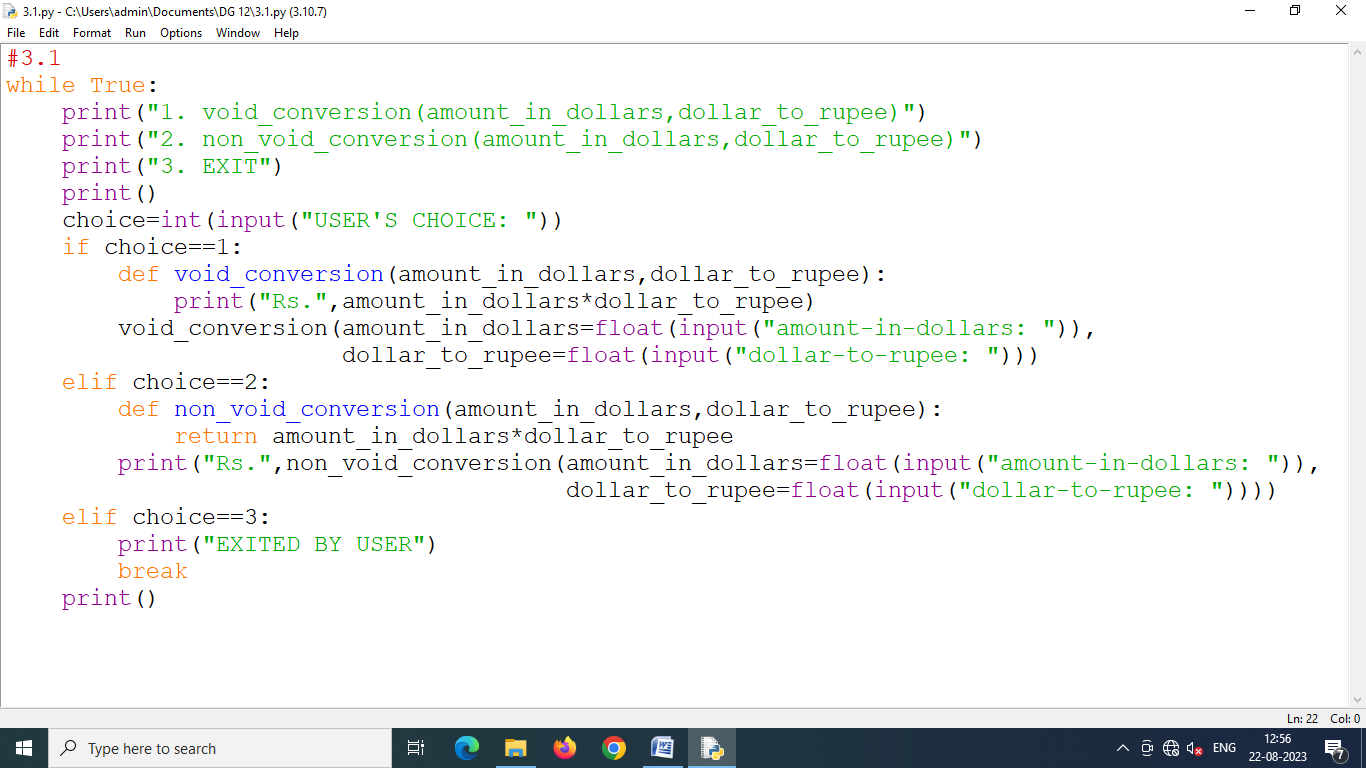
Program 2.11 **Write a program to sort a dictionary's keys using Bubble sort and produce the sorted keys as a list.**

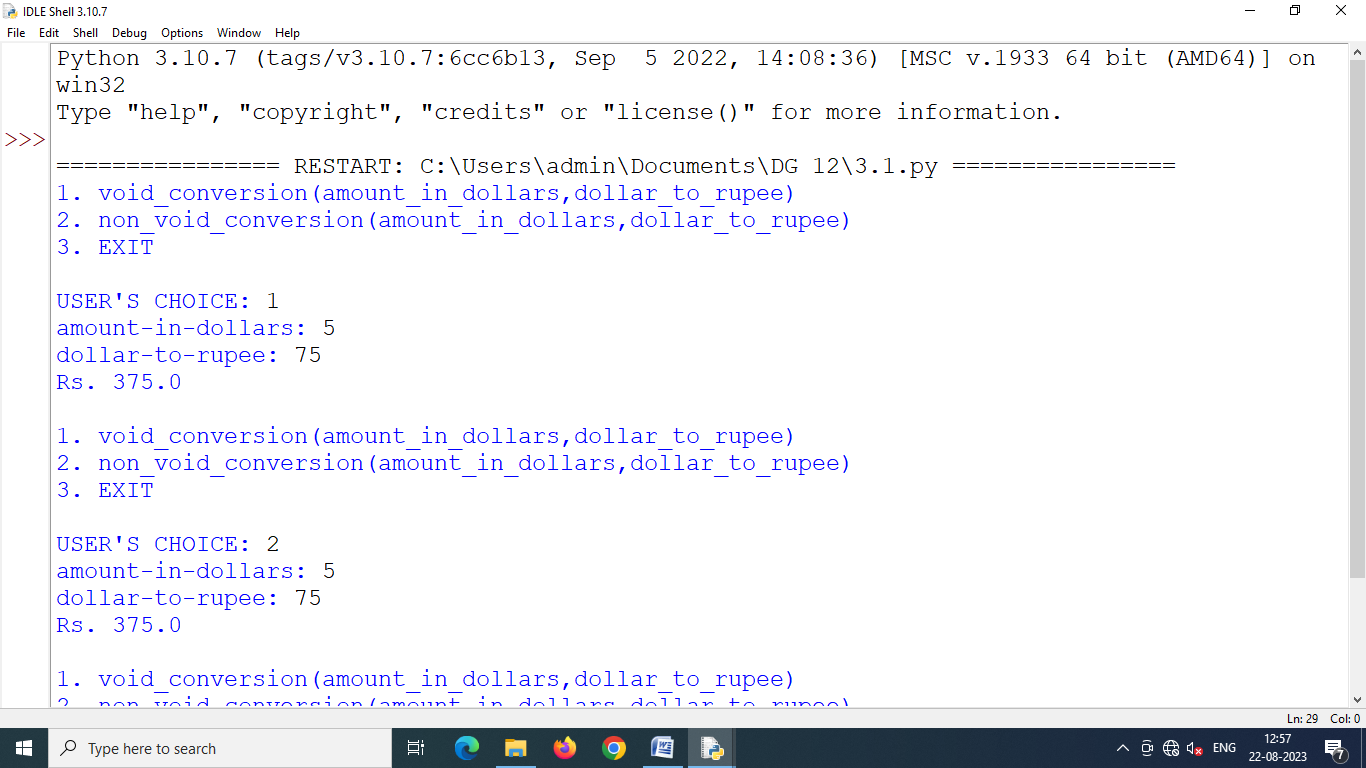


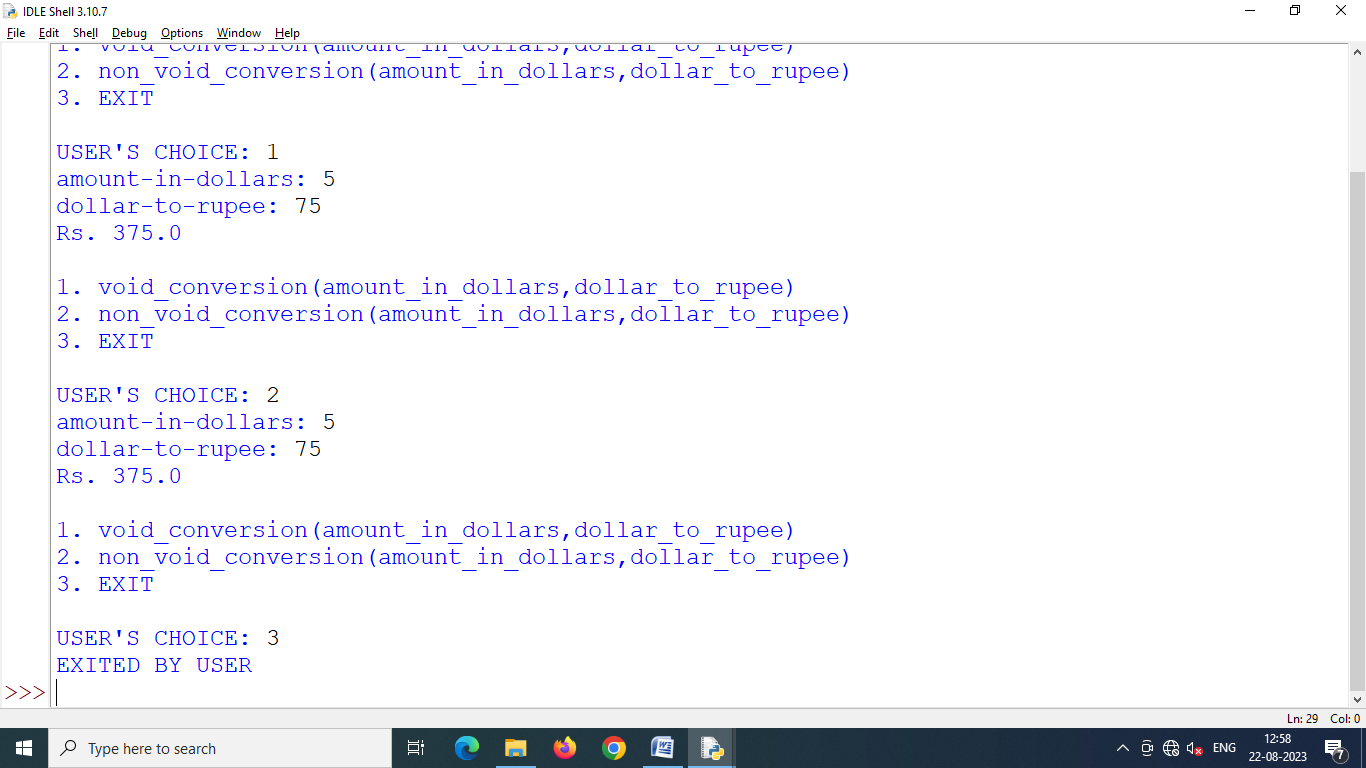
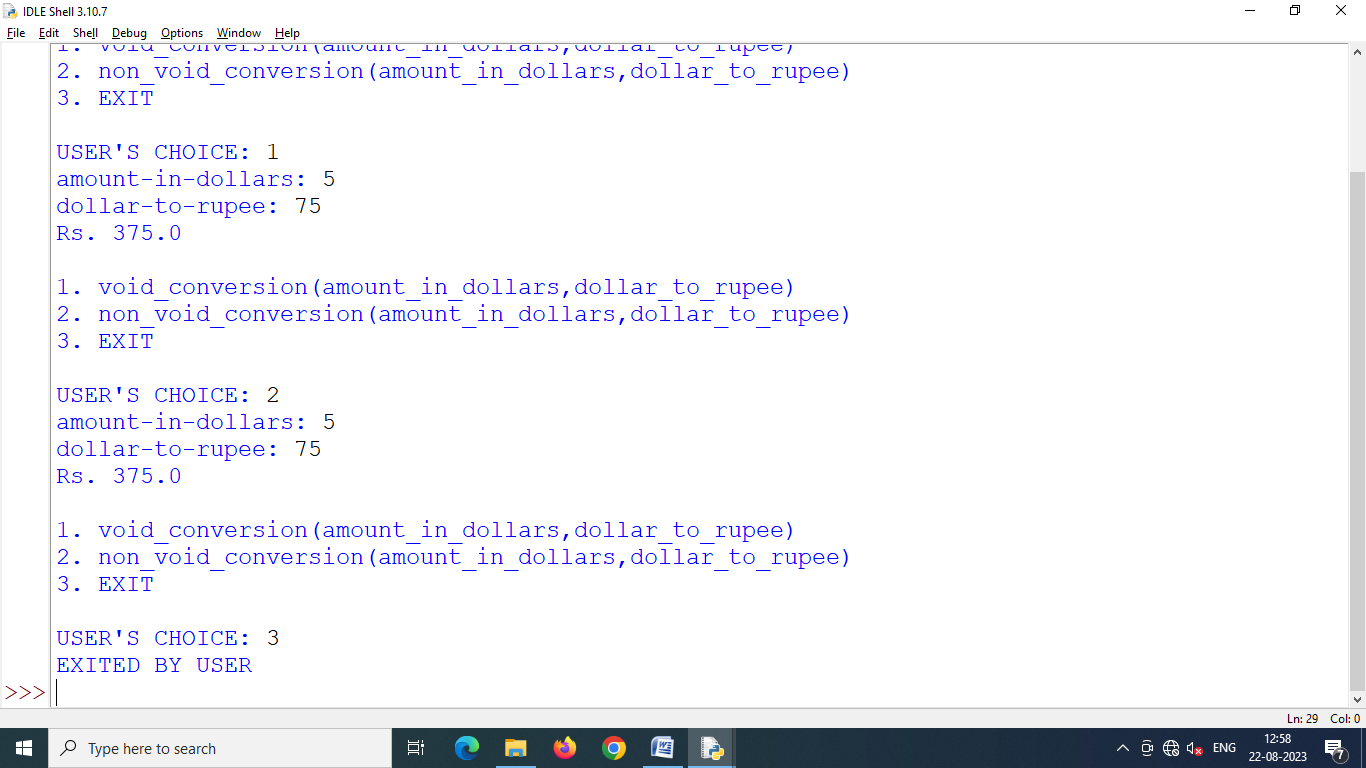
Program 2.12 **Write a program to sort a dictionary's values using Bubble sort and produce the sorted values as a list.**

****

Program 3.1 **Write a function that takes amount-in-dollars and dollar-to-rupee conversion price; it then returns the amount converted to rupees. Create the function in both void and non-void forms.**



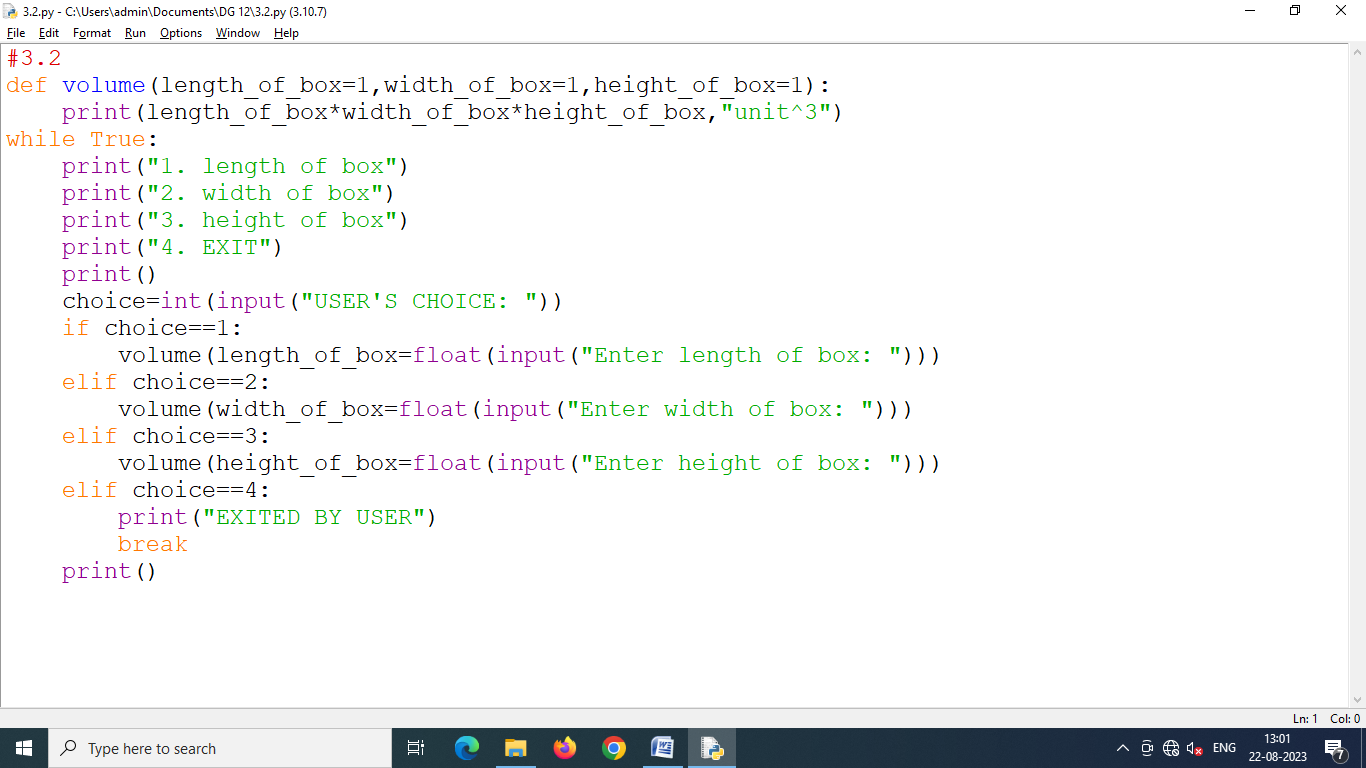


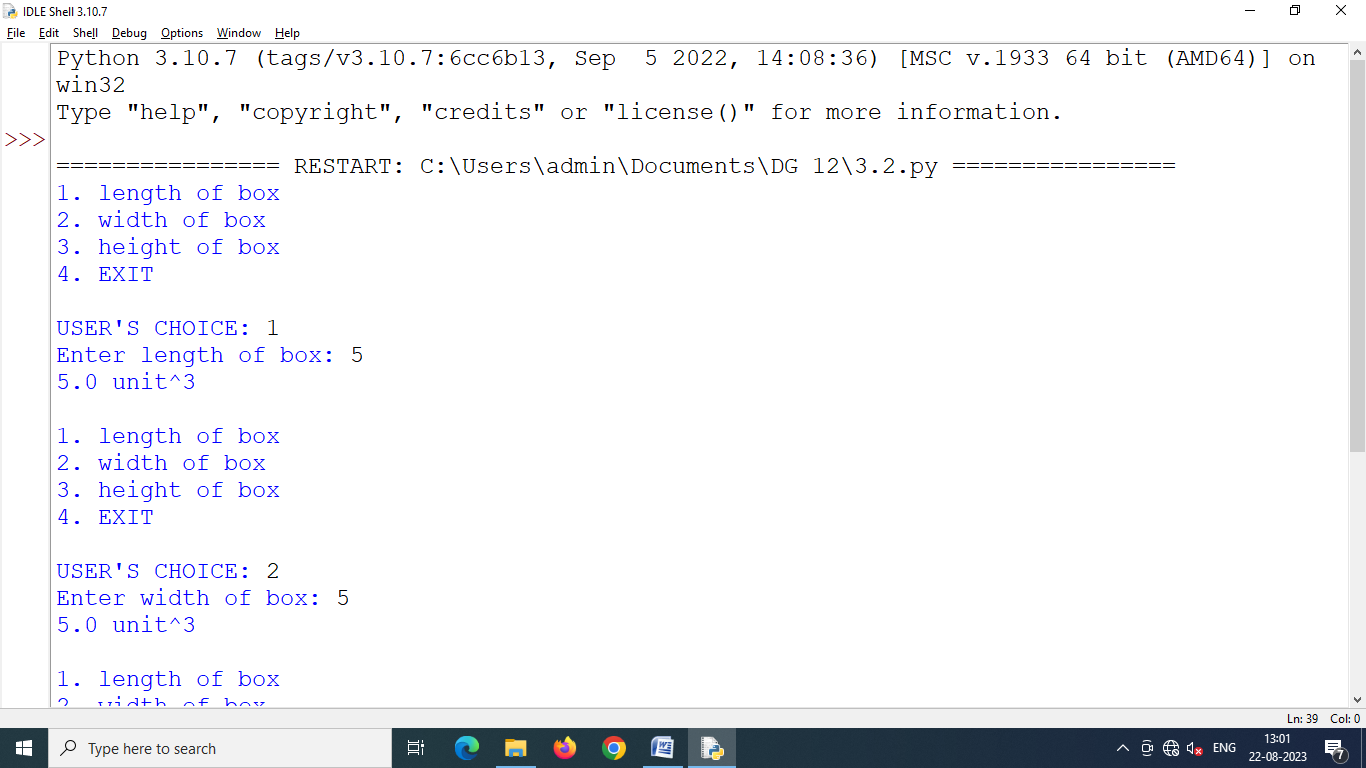


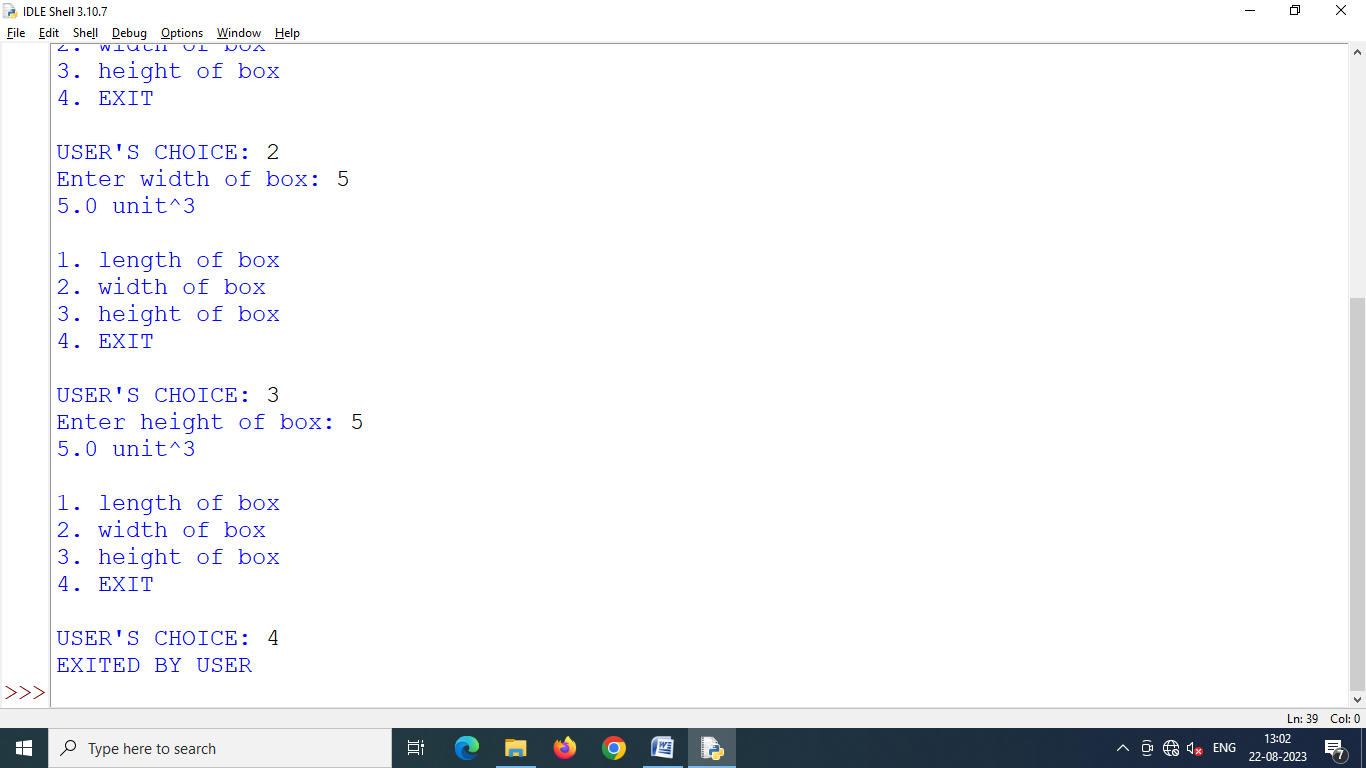
Program 3.2 **Write a function to calculate volume of a box with appropriate default values for its parameters. Your function should have the following input parameters :**

1. **length of box ;**
2. **width of box ;**
3. **height of box.**

**Test it by writing complete program to invoke it.**

****

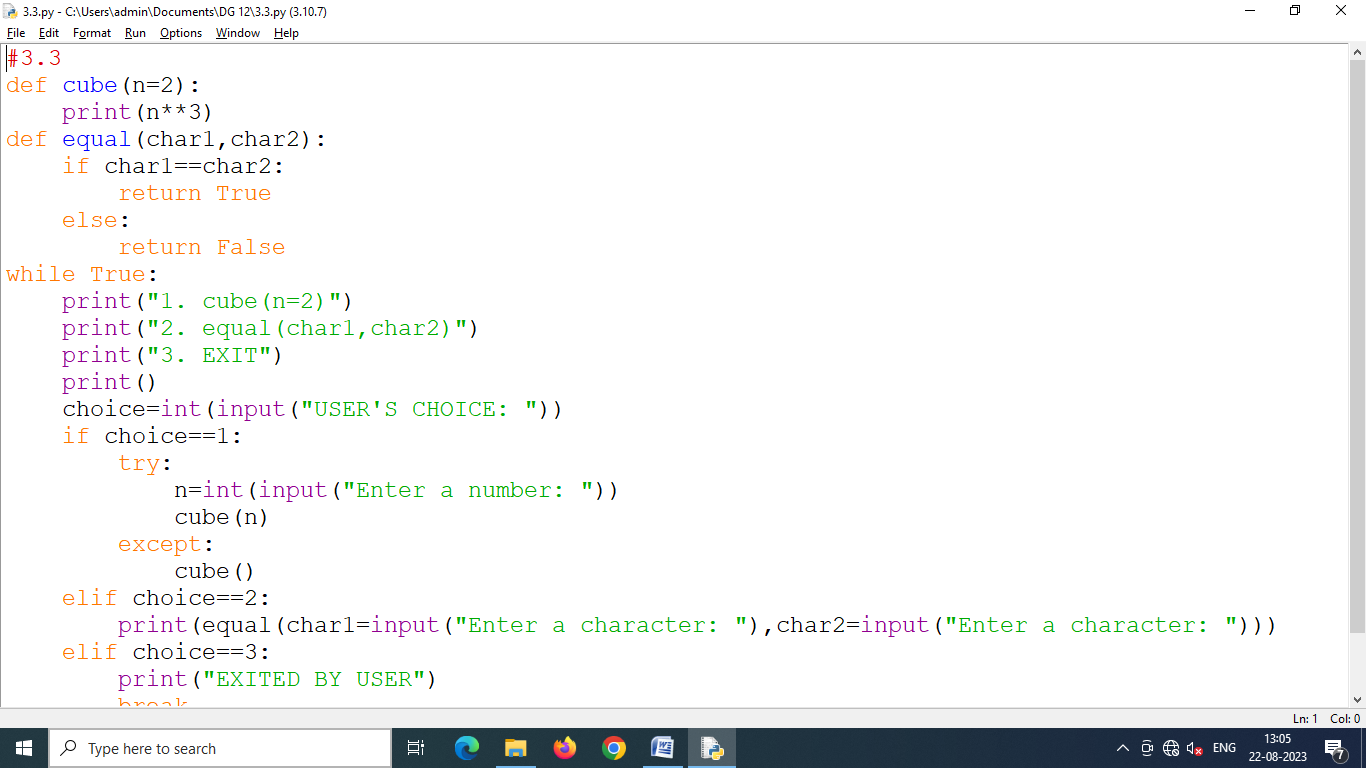
****

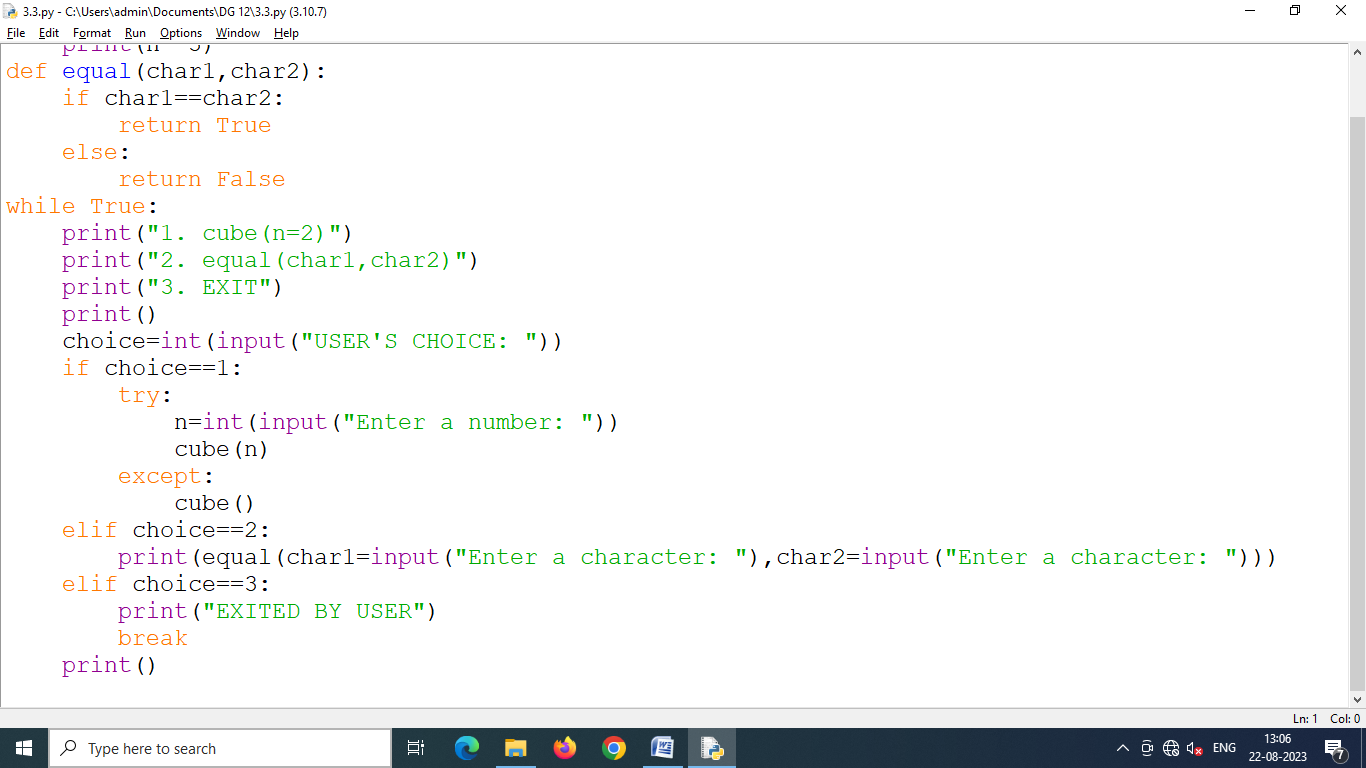
****

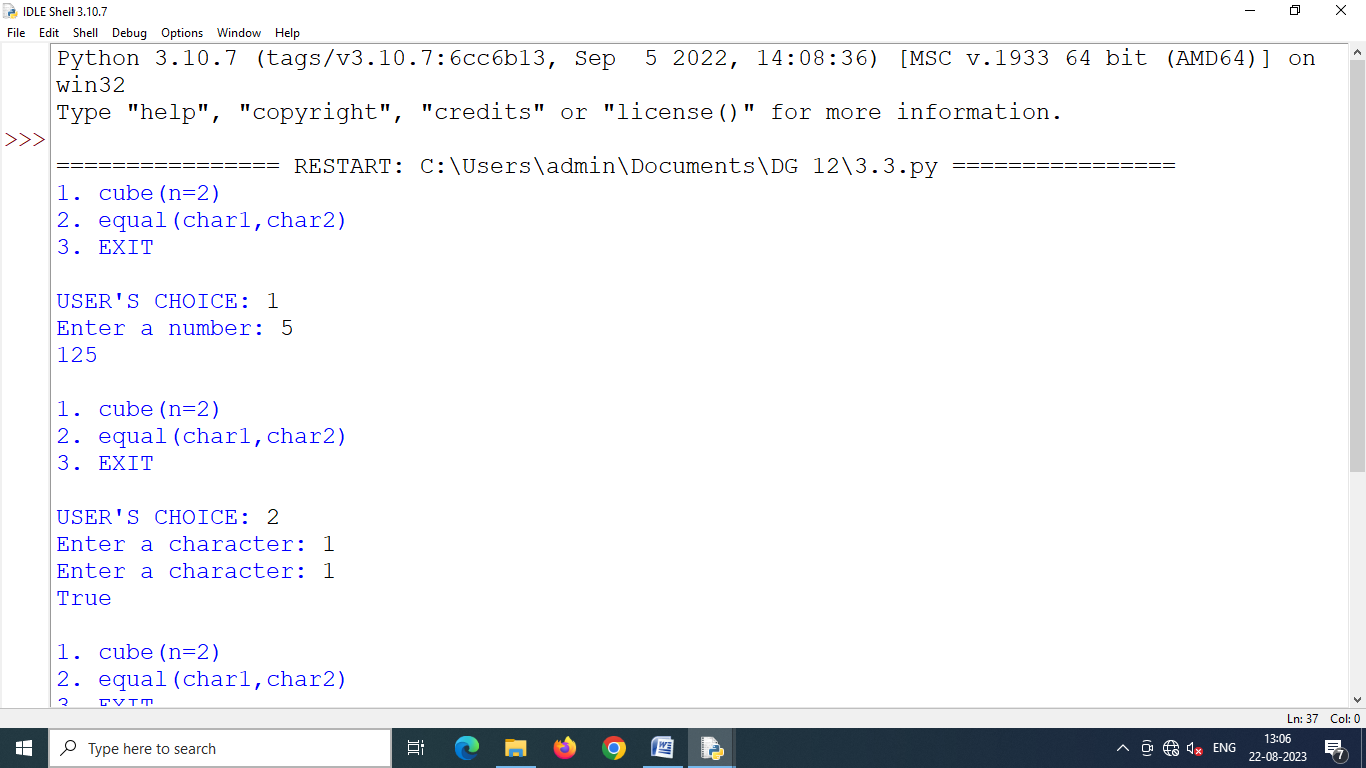
Program 3.3 **Write a program to have following functions :**

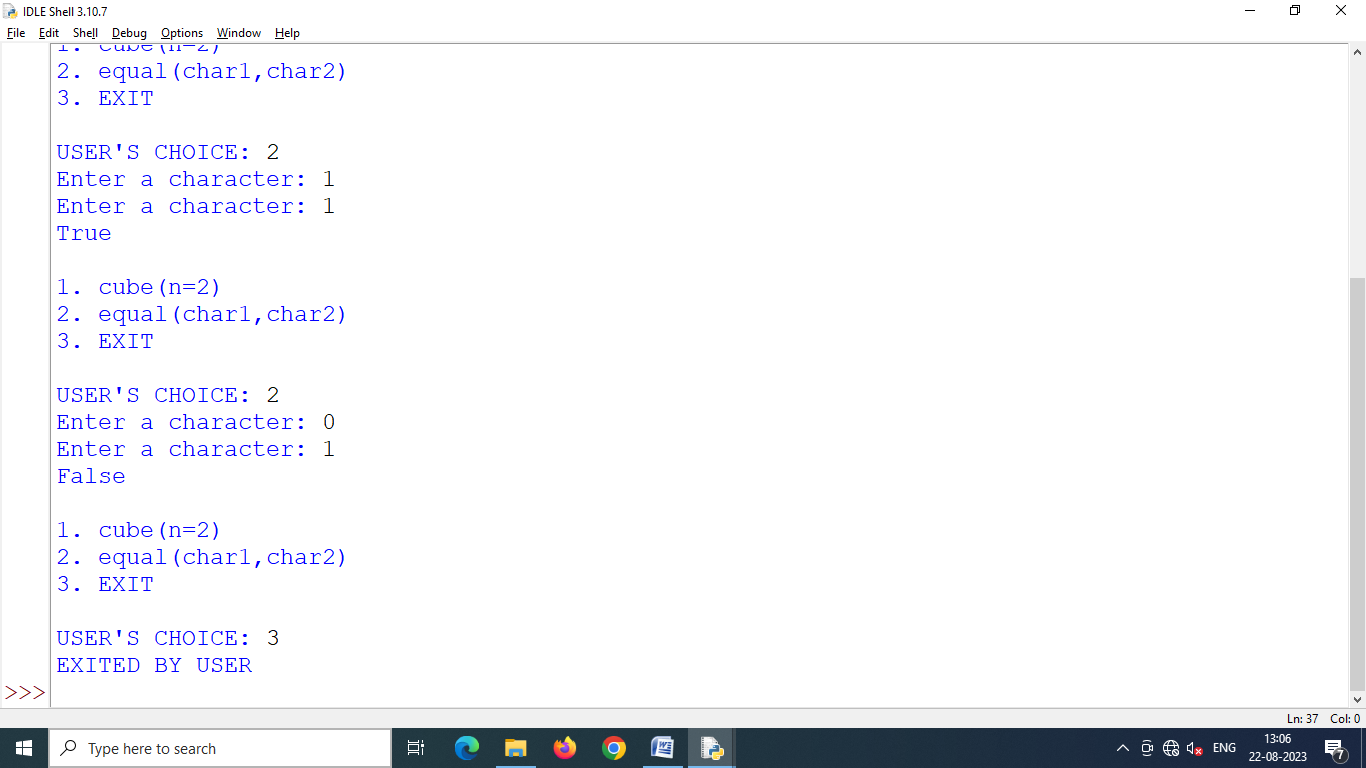
1. **a function that takes a number as argument and calculates cube for it. The function does not return a value. If there is no value to the function in function call, the function should calculate cube of 2.**
2. **a function that takes two char arguments and returns True if both the arguments are equal otherwise False.**

**Test both these functions by giving appropriate function call statements.**

****

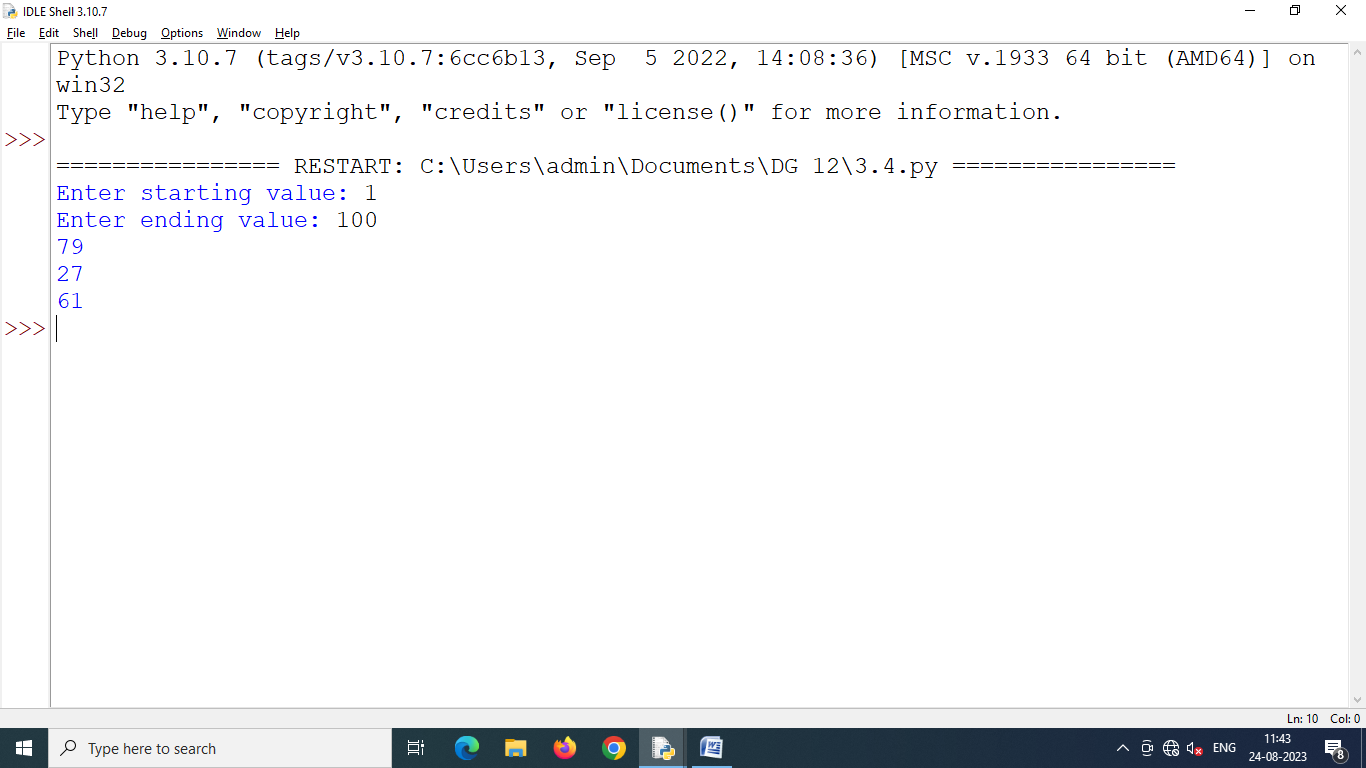
****

****

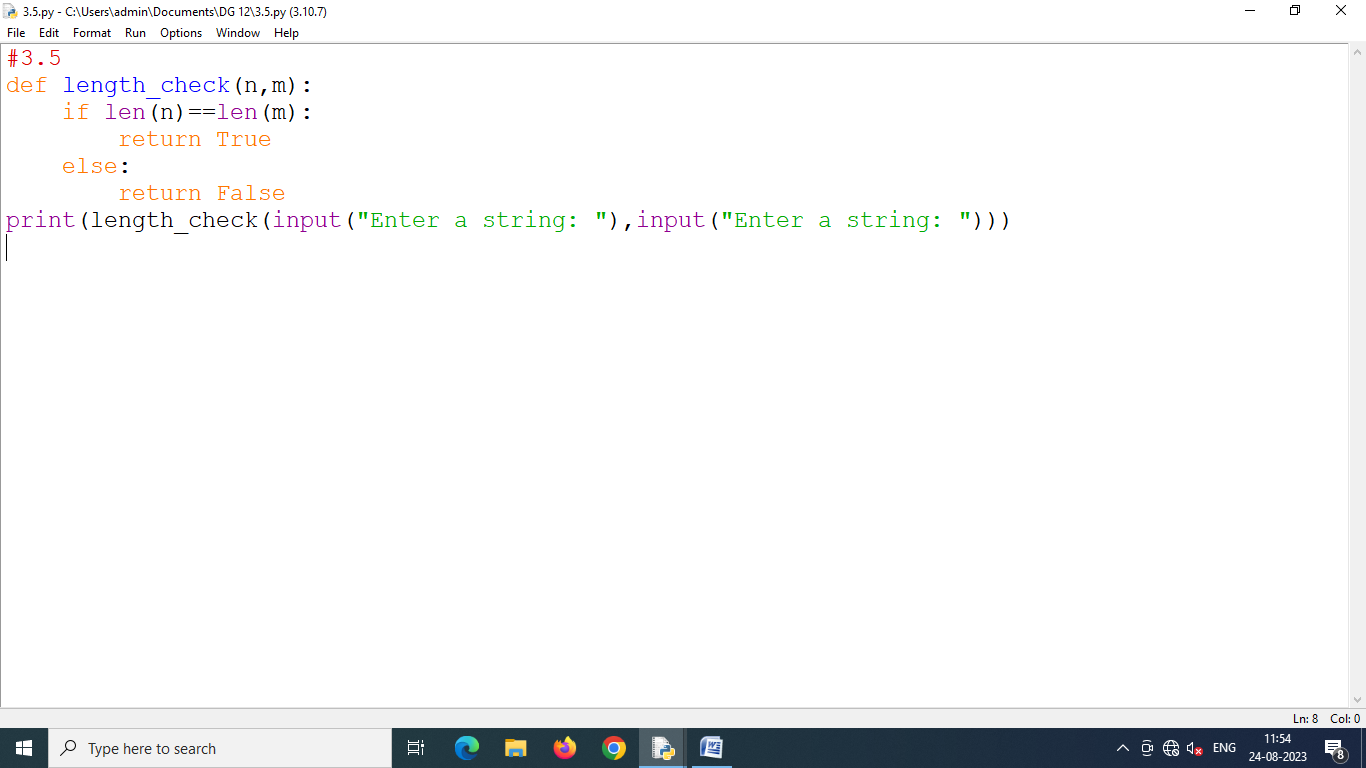
****

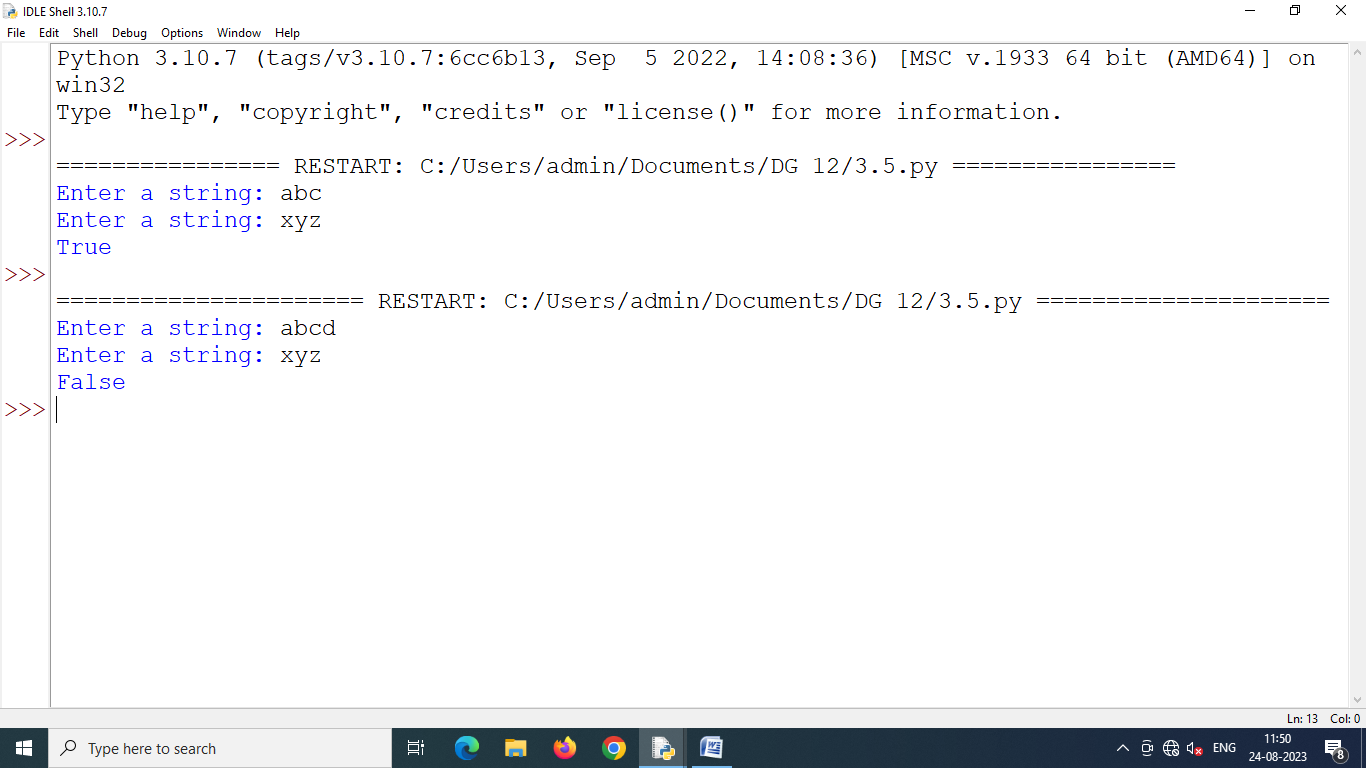
Program 3.4 **Write a function that receives two numberg.s and generates a random number from that range. Using this function, the main program should able to print three numbers randomly.**

****

****

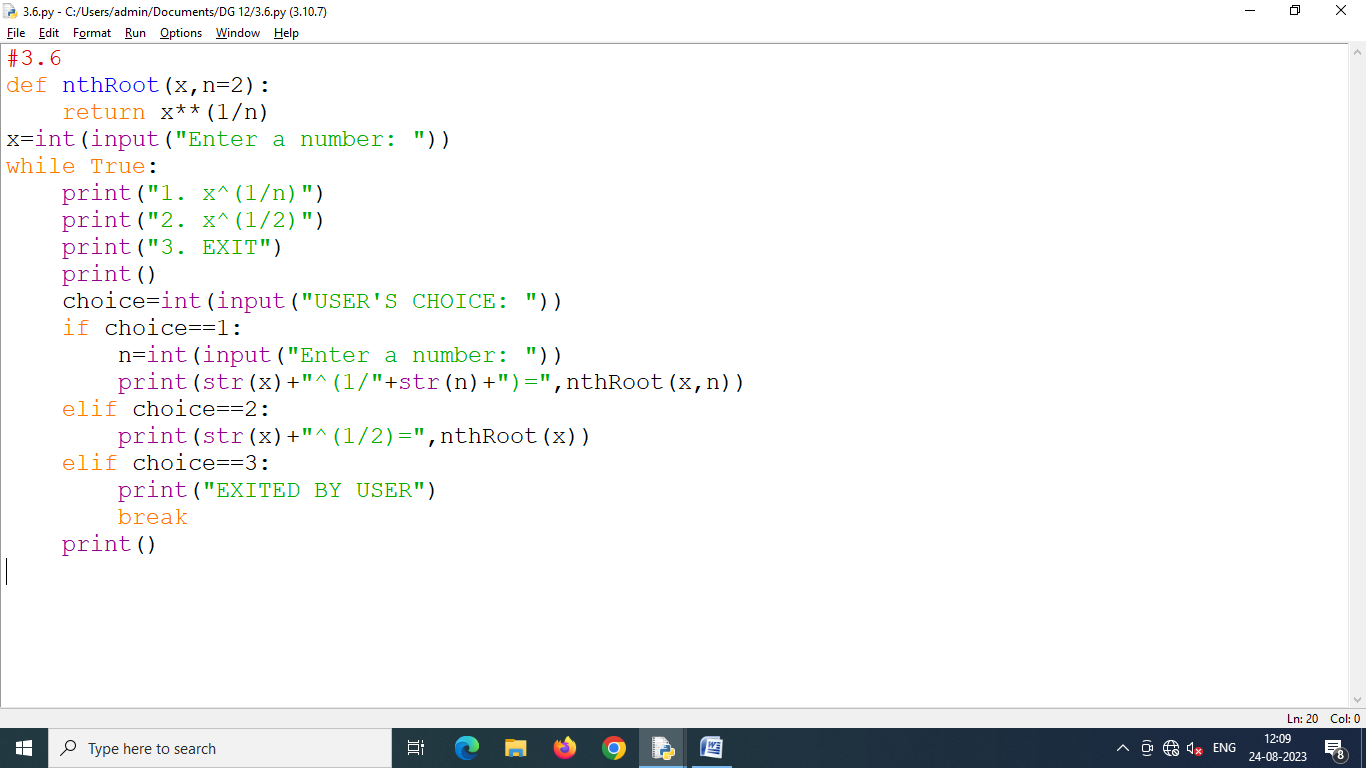
Program 3.5 **Write a function that receives two string arguments and checks whether they are same-length strings (returns True in this case otherwise False).**

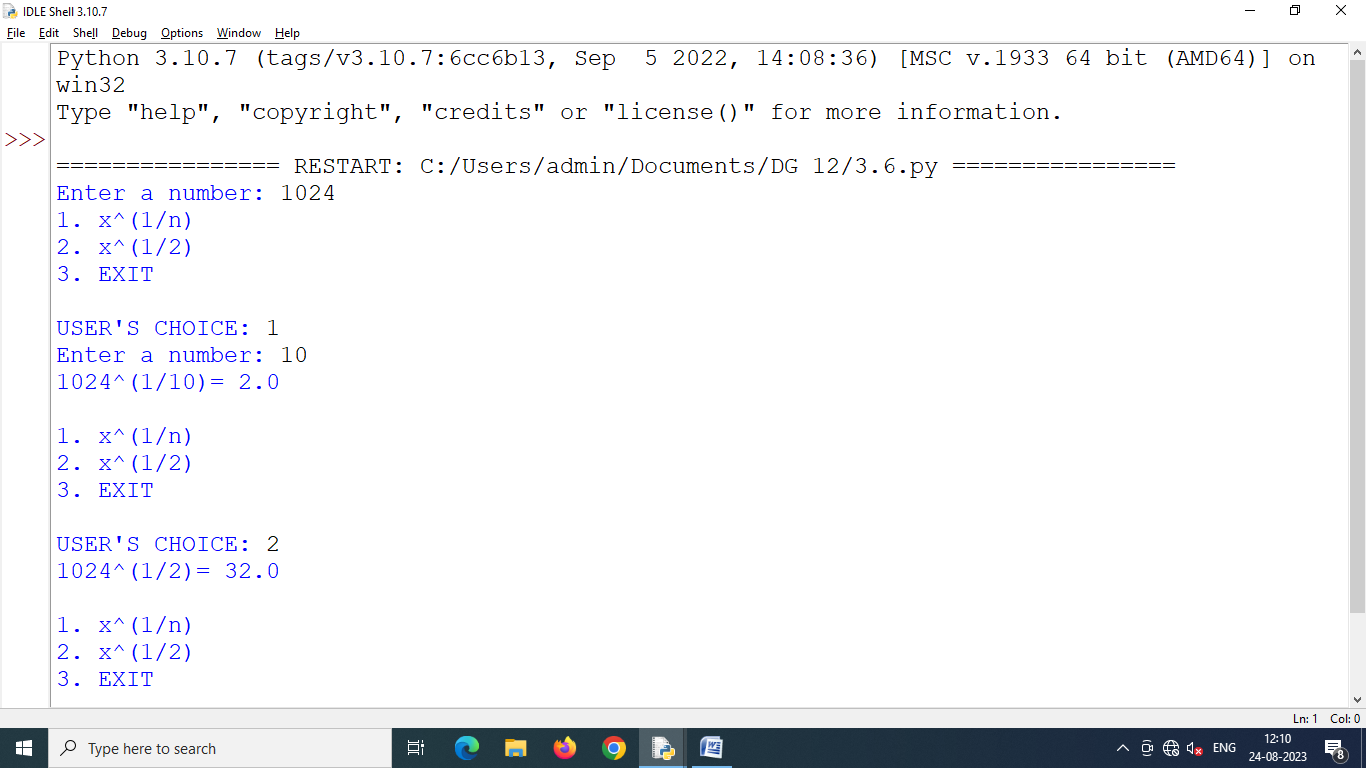
****

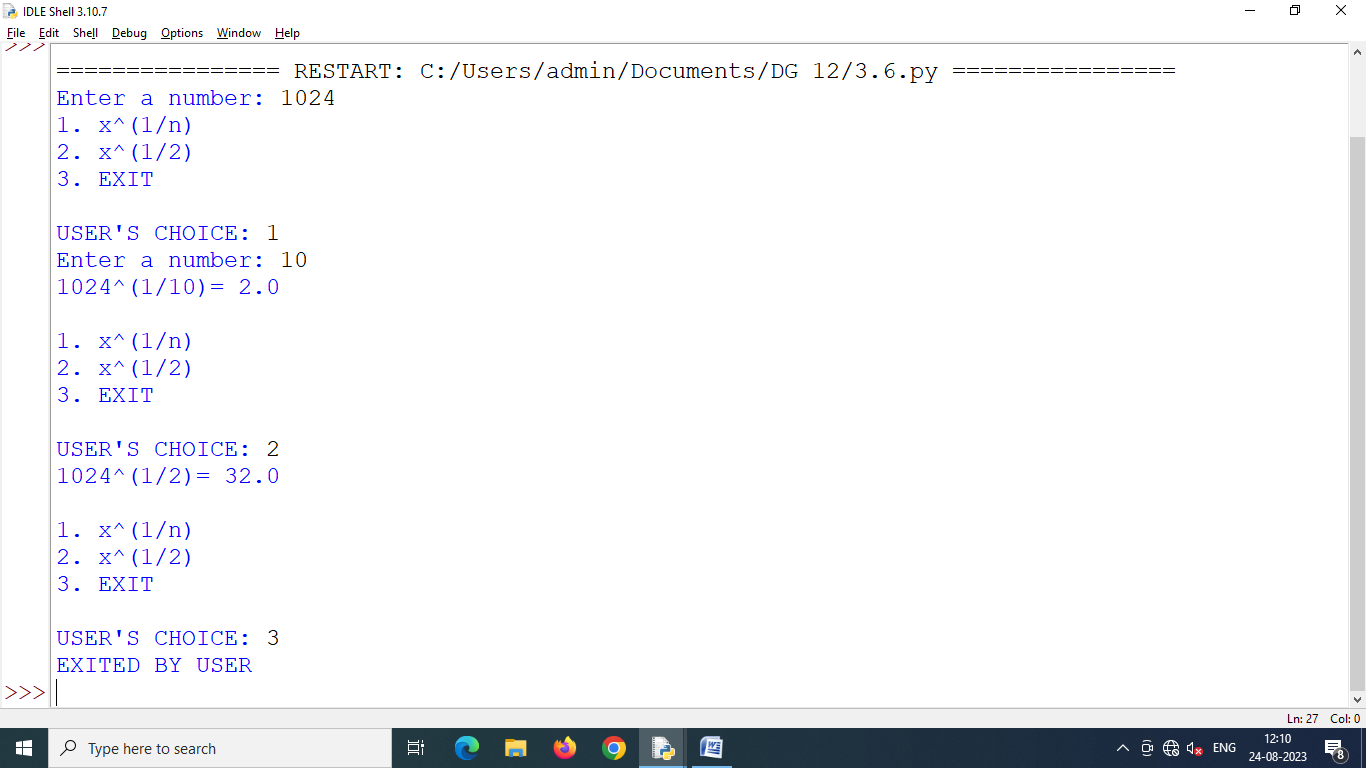
****

Program 3.6 **Write a function namely nthRoot() that receives two parameters x and n and returns nth root of x i.e.,**

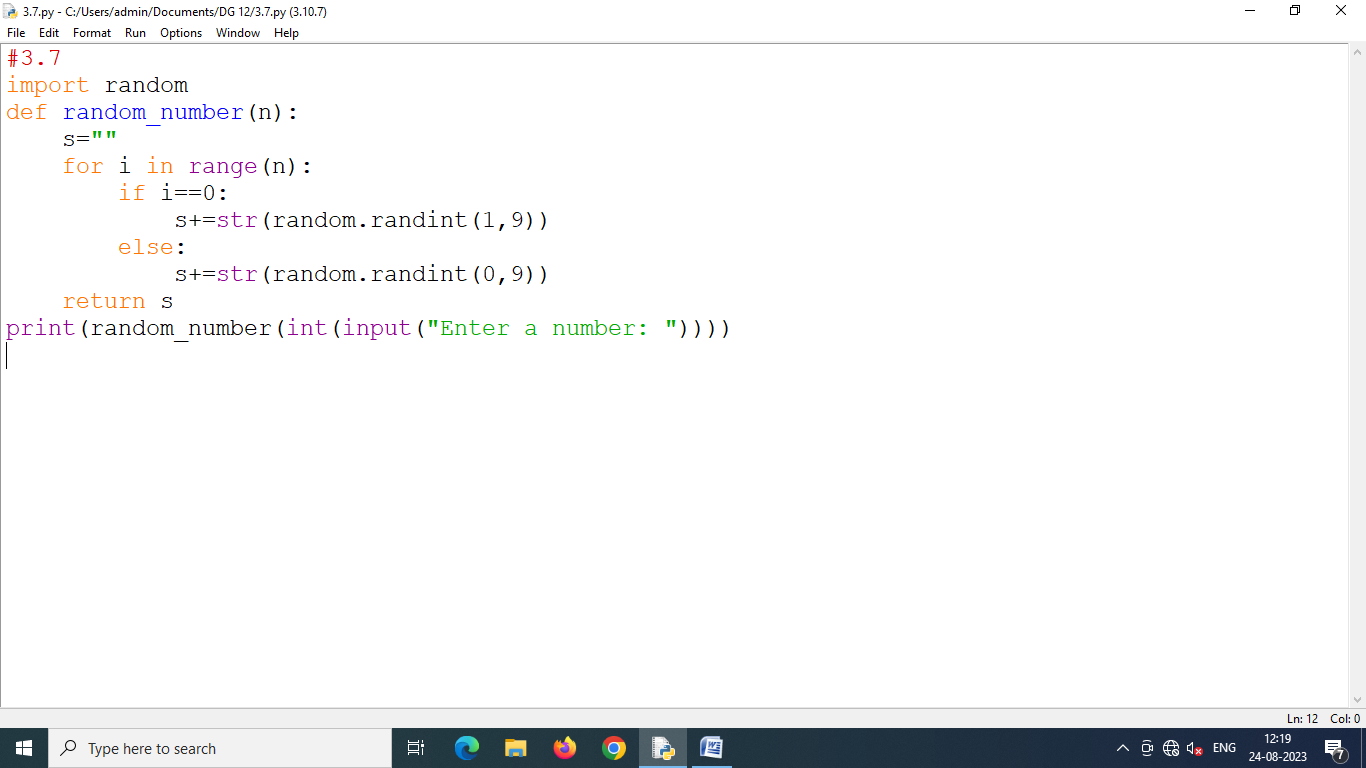
**The default value n is 2.**

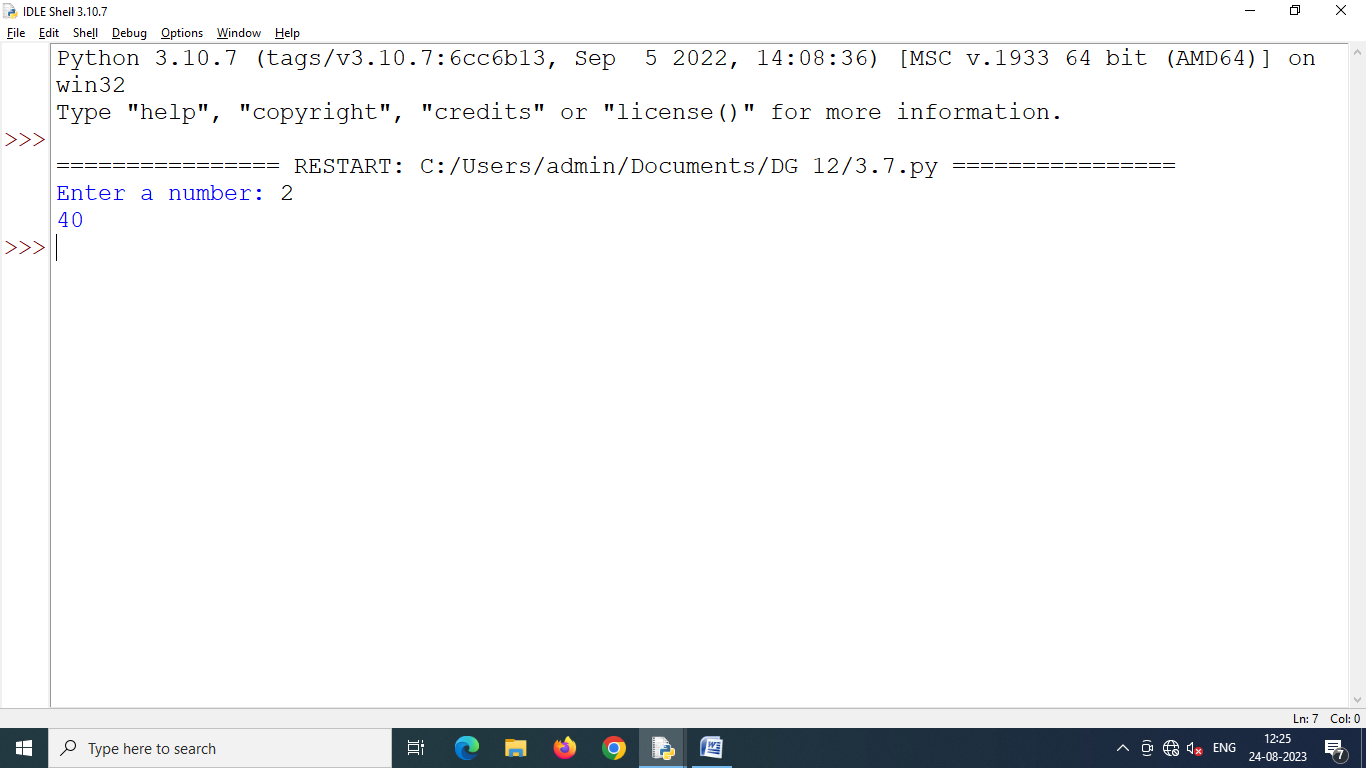
****

****

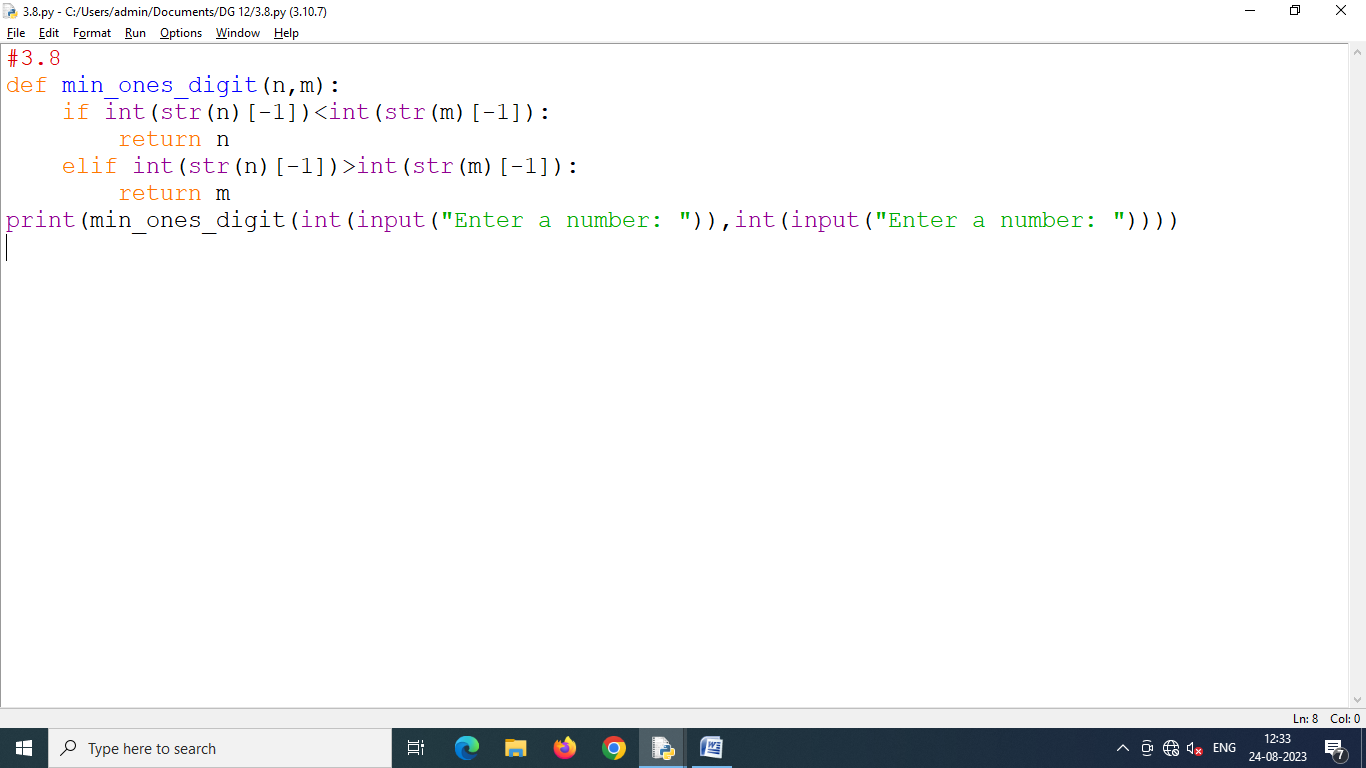
****

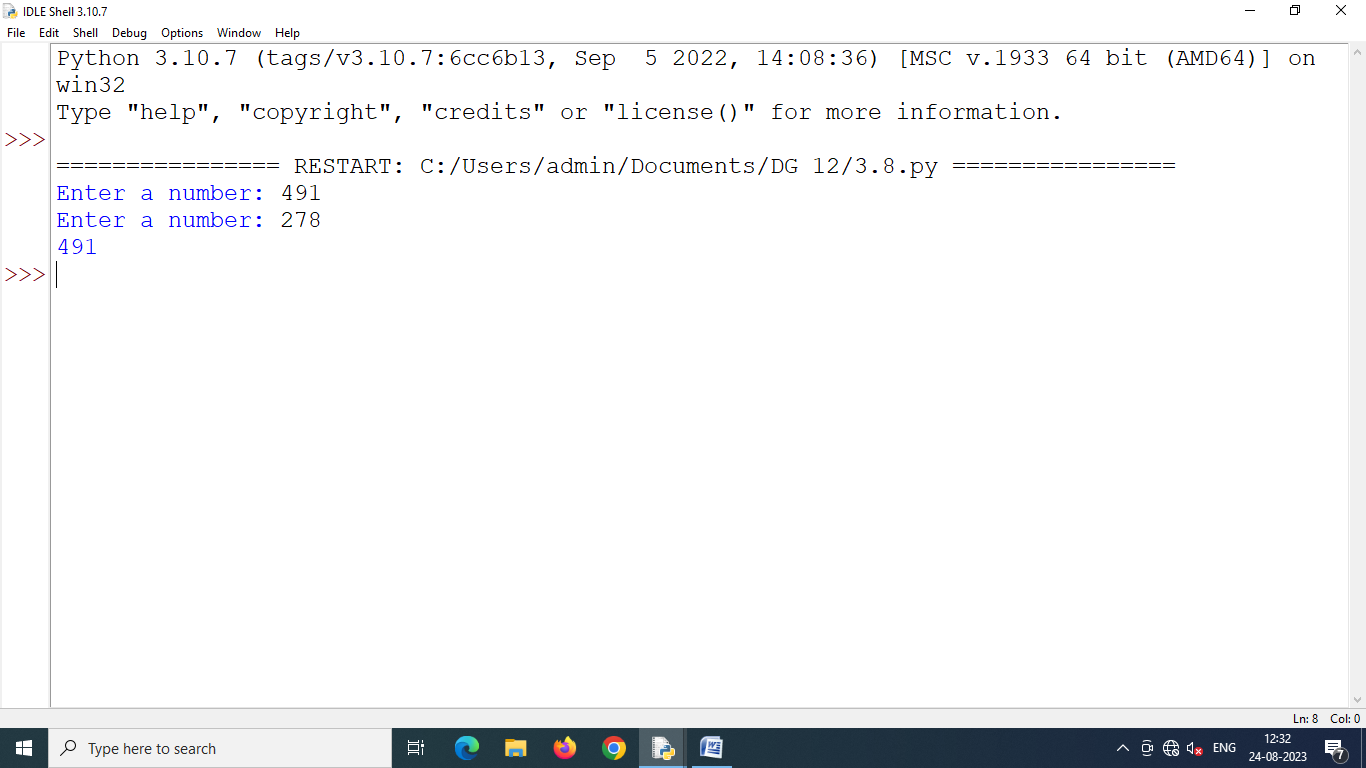
Program 3.7 **Write a function that takes a number n and then returns a randomly generated number having exactly n digits (not starting with zero) e.g., if n is 2 then function can randomly return a number 10-99 but 07, 02 etc. are not valid two digit numbers.**

****

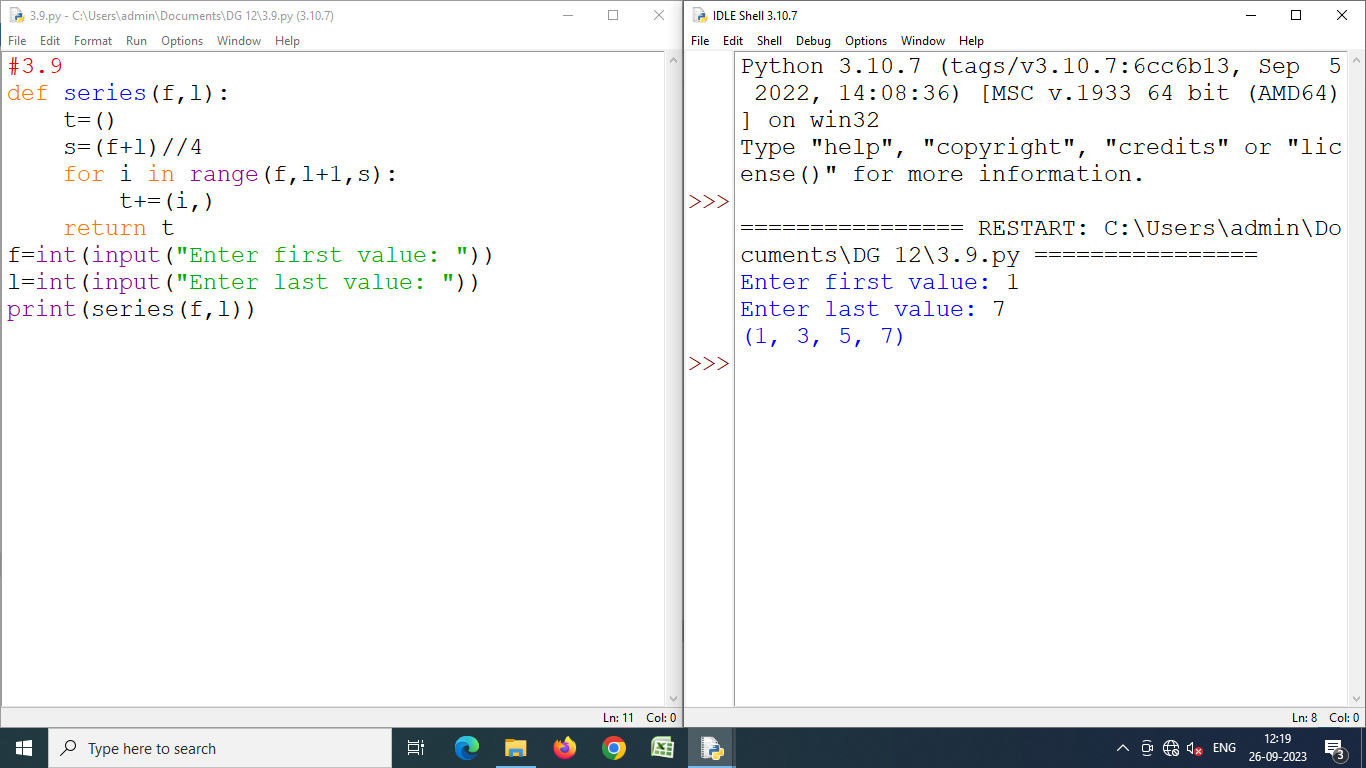
****

Program 3.8 **Write a function that takes two numbers and returns the number that has minimum one's digit. [For example, if numbers passed are 491 and 278, then the function will return 491 because it has got minimum one's digit out of two given numbers (491's 1 is < 278's 8)].**

****

****

Program 3.9 **Write a program that generates a series using a function which takes first and last values of the series and then generates four terms that are equidistant e.g., if two numbers passed are 1 and 7 then function returns 1 3 5 7.**

****

Program 4.1 **Write a program having following functions :**

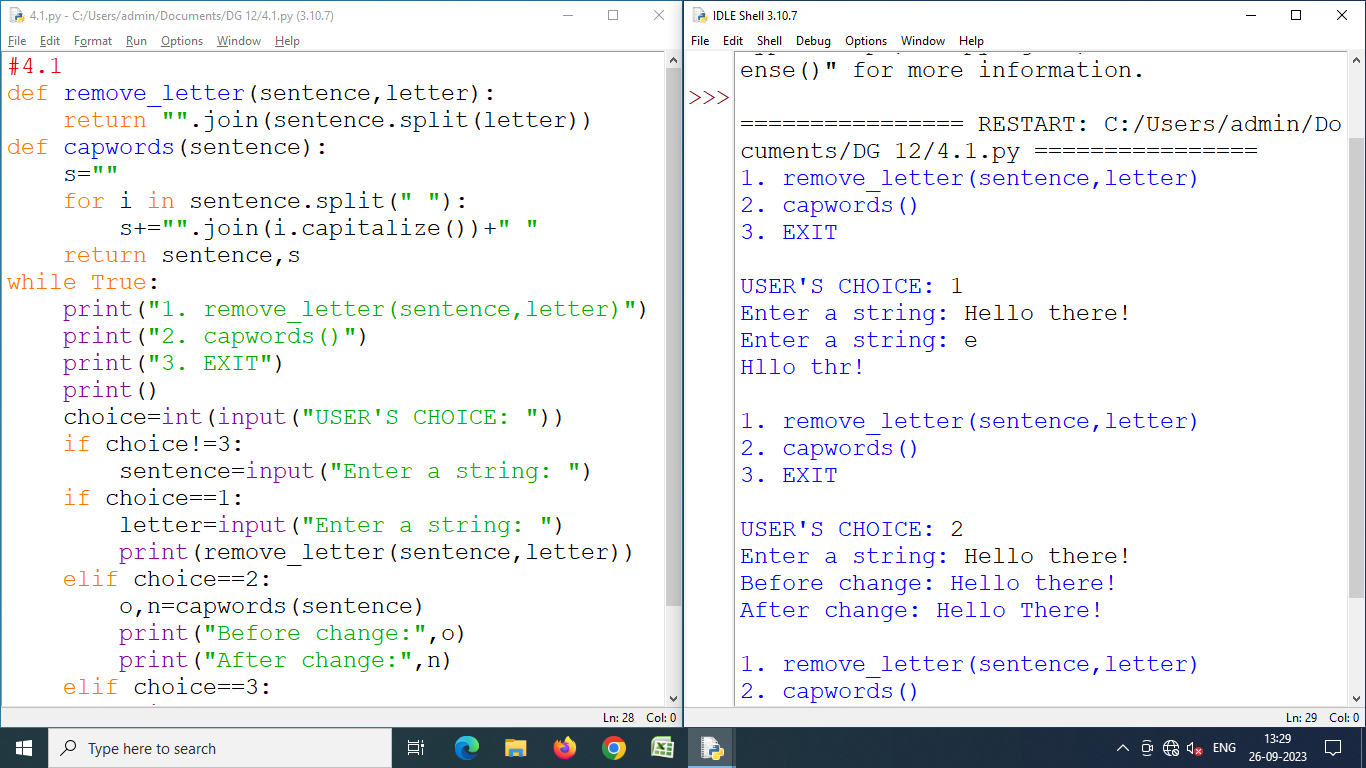
1. **A function with the following signature : remove\_letter(sentence, letter)**

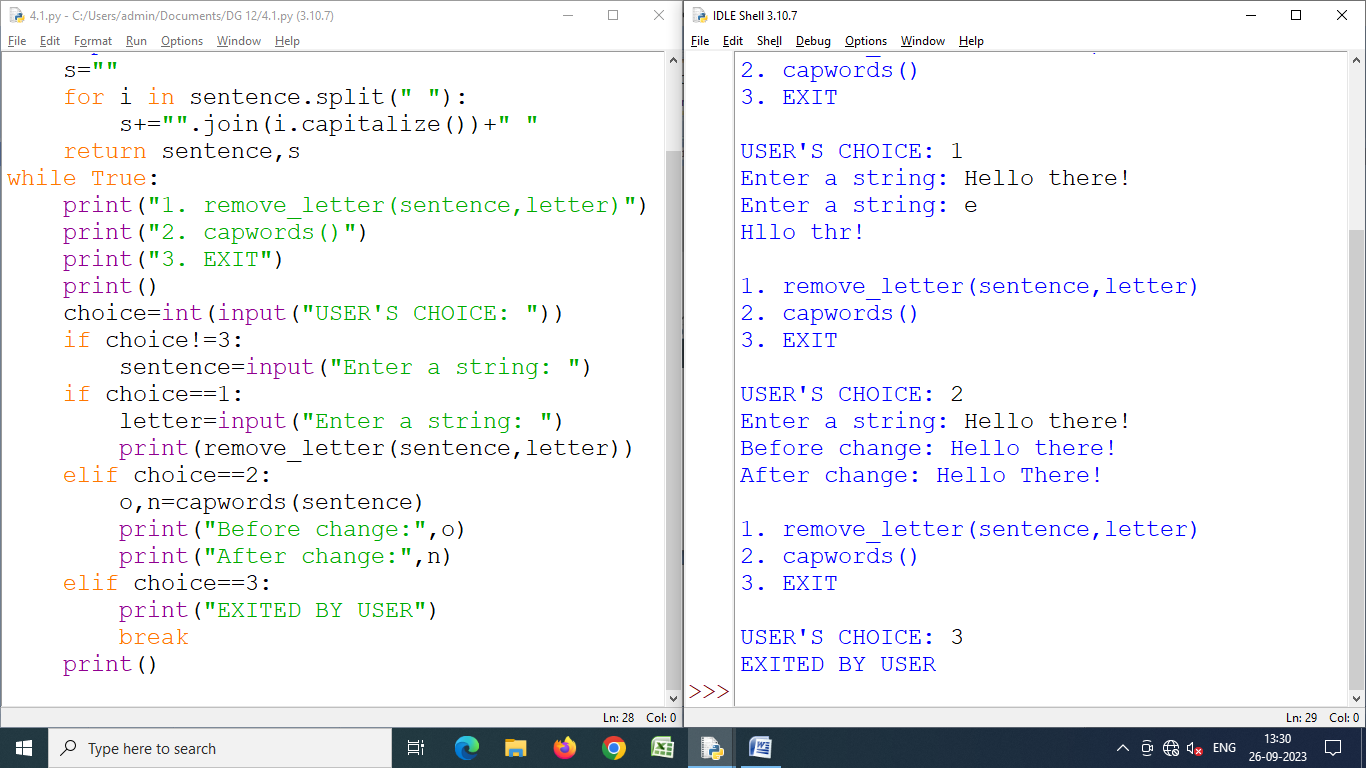
**This function should take a string and a letter (as a single-character string) as arguments, returning a copy of that string with every instance of the indicated letter removed. For example, remove\_letter("Hello there!","e") should return the string "Hllo thr!".**

**Try implementing it using <str>.split( ) function.**

1. **Write a function to do the following :**

**Try implementing the capwords( ) functionality using other functions, i.e., split( ), capitalize() and join( ). Compare the result with the capwords( ) function's result.**

****

****

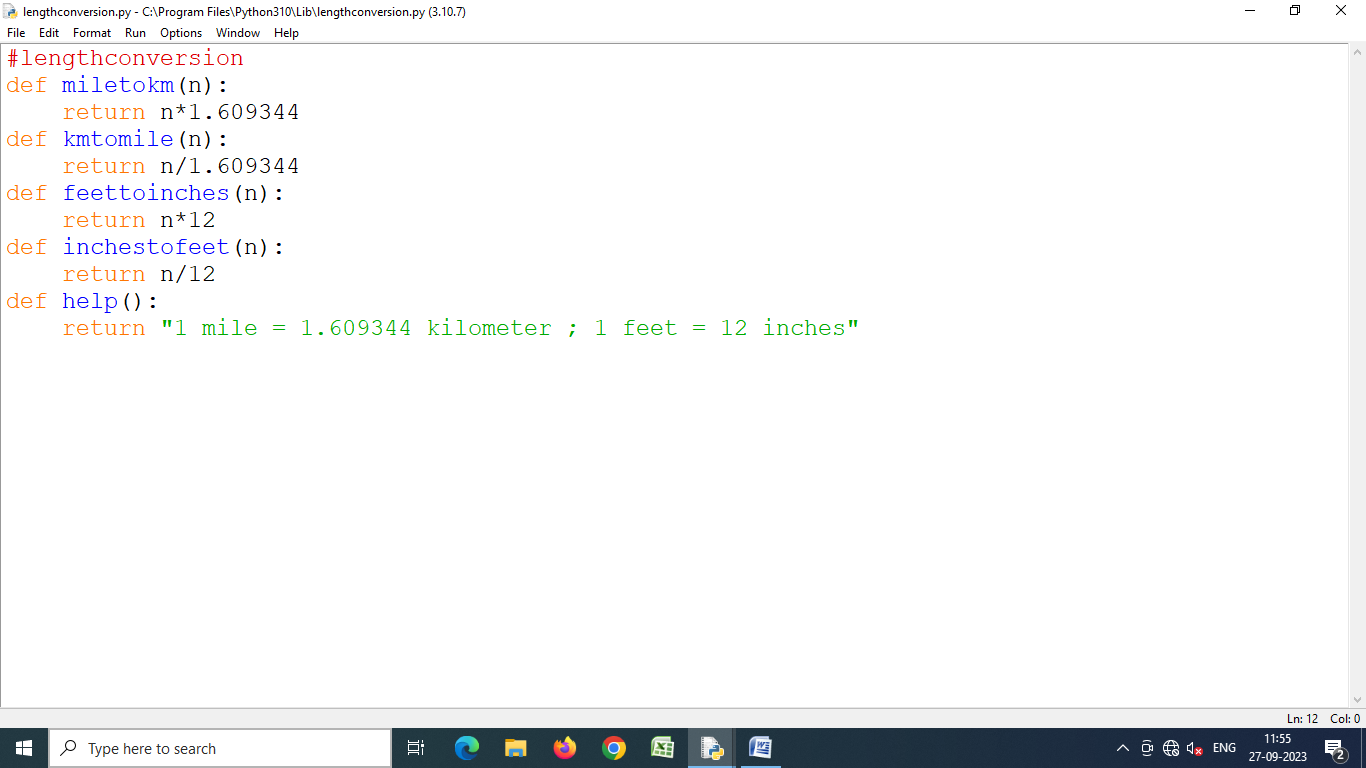
Program 4.2 **Create a module lengthconversion.py that stores functions for various lengths conversion e.g.,**

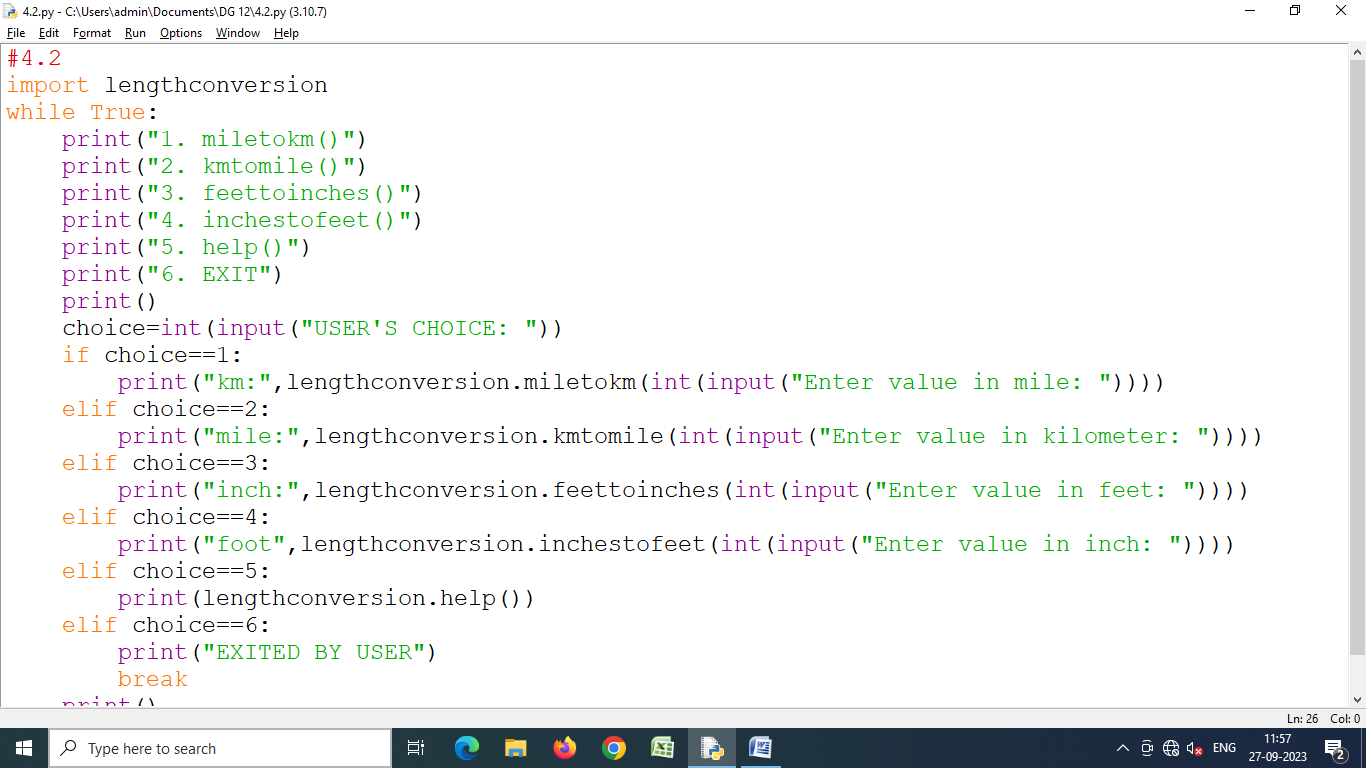
* **miletokm( ) to convert miles to kilometer**
* **kmtomile( ) to convert kilometers to miles**
* **feettoinches( ) • inchestofeet( )**

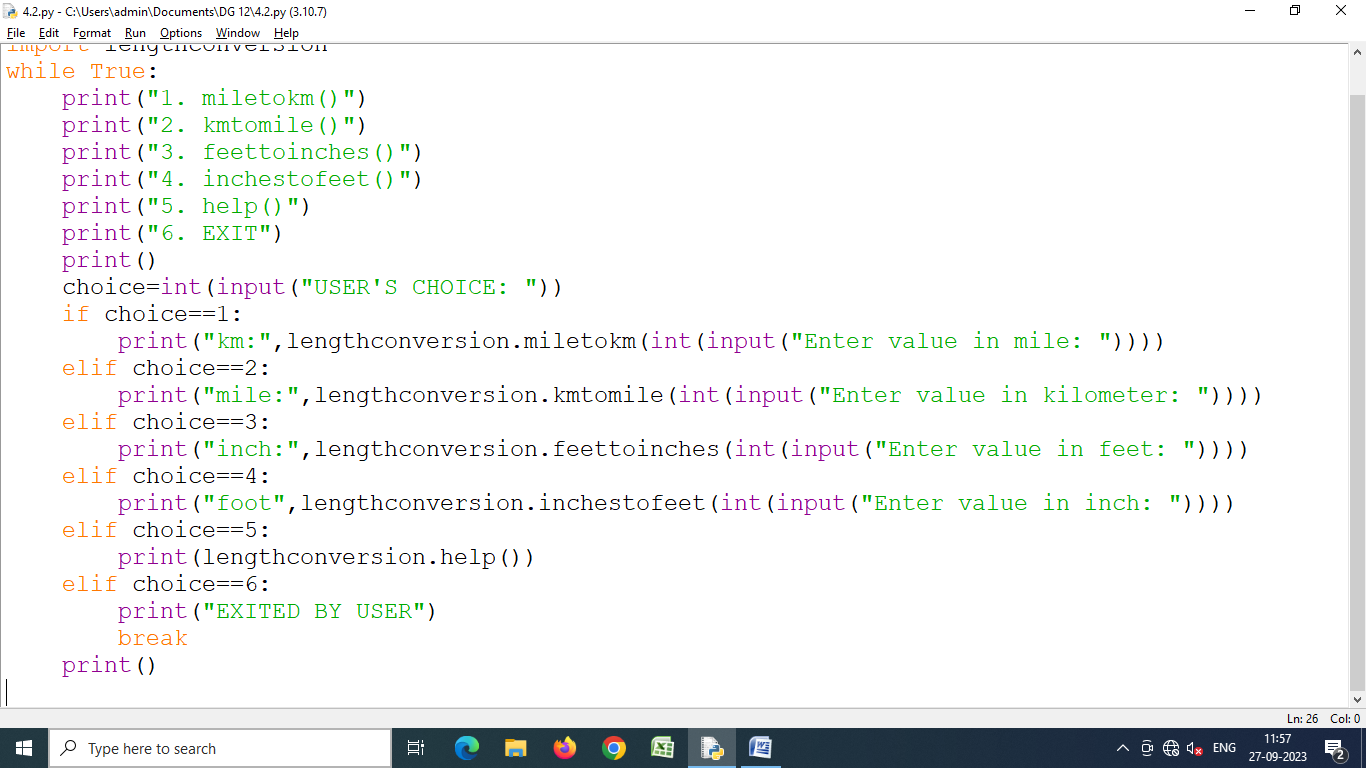
**It should also store constant values such as value of (mile in kilometers and vice versa).**

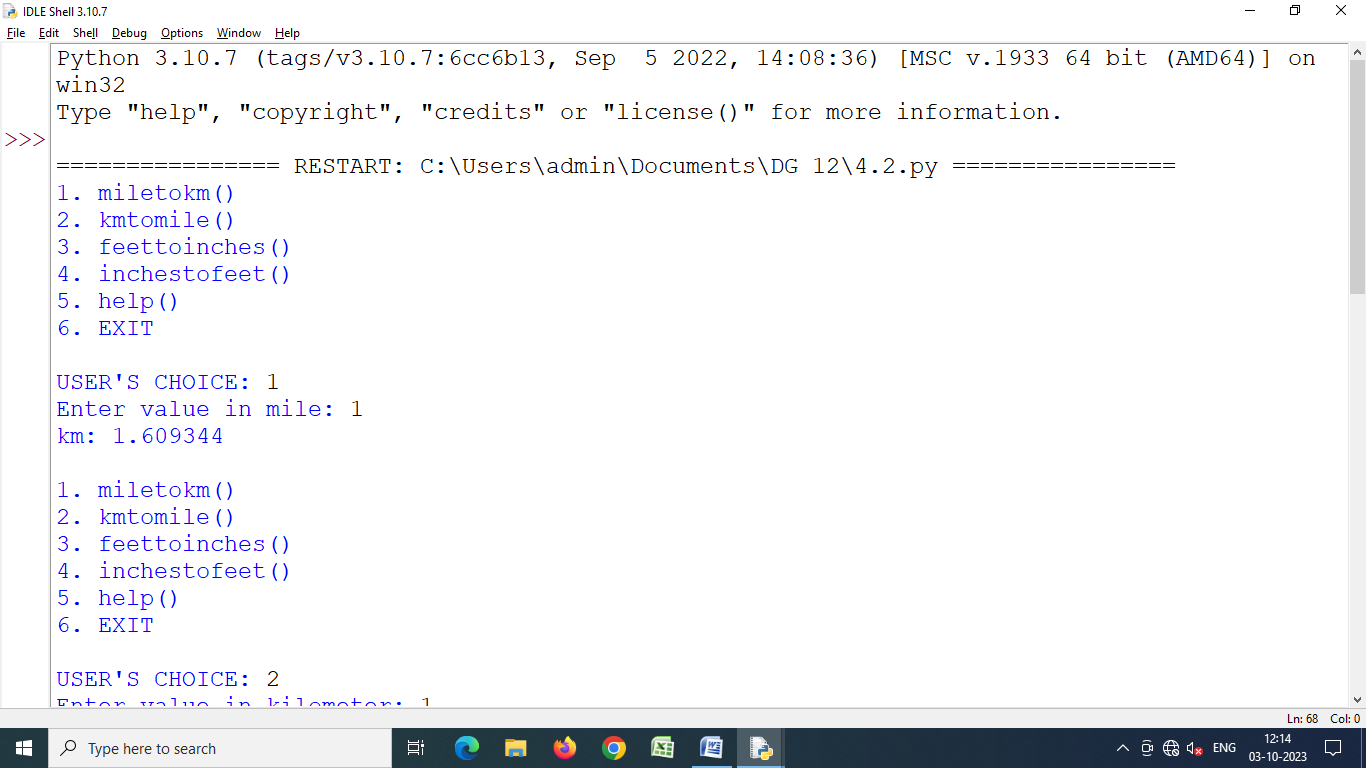
**[1 mile = 1.609344 kilometer ; 1 feet = 12 inches]**

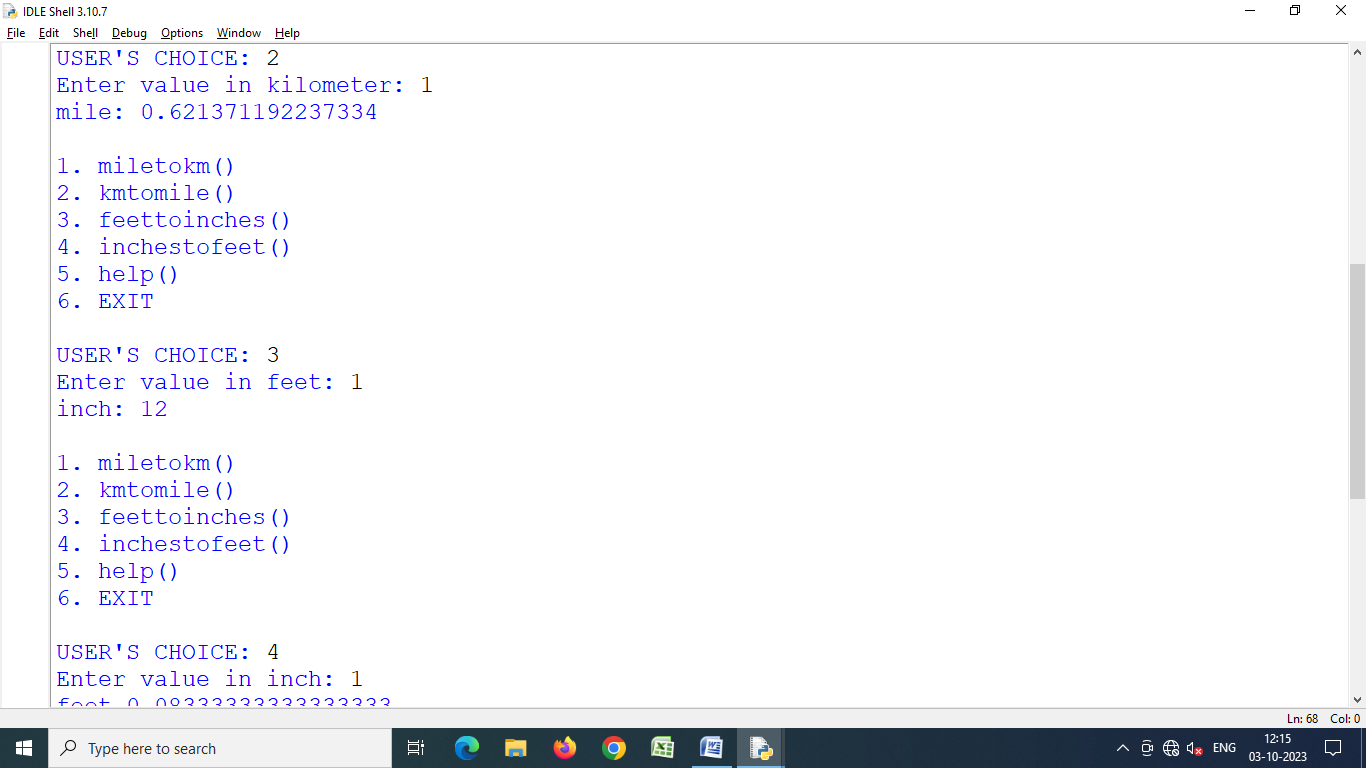
**Help( ) function should display proper information.**

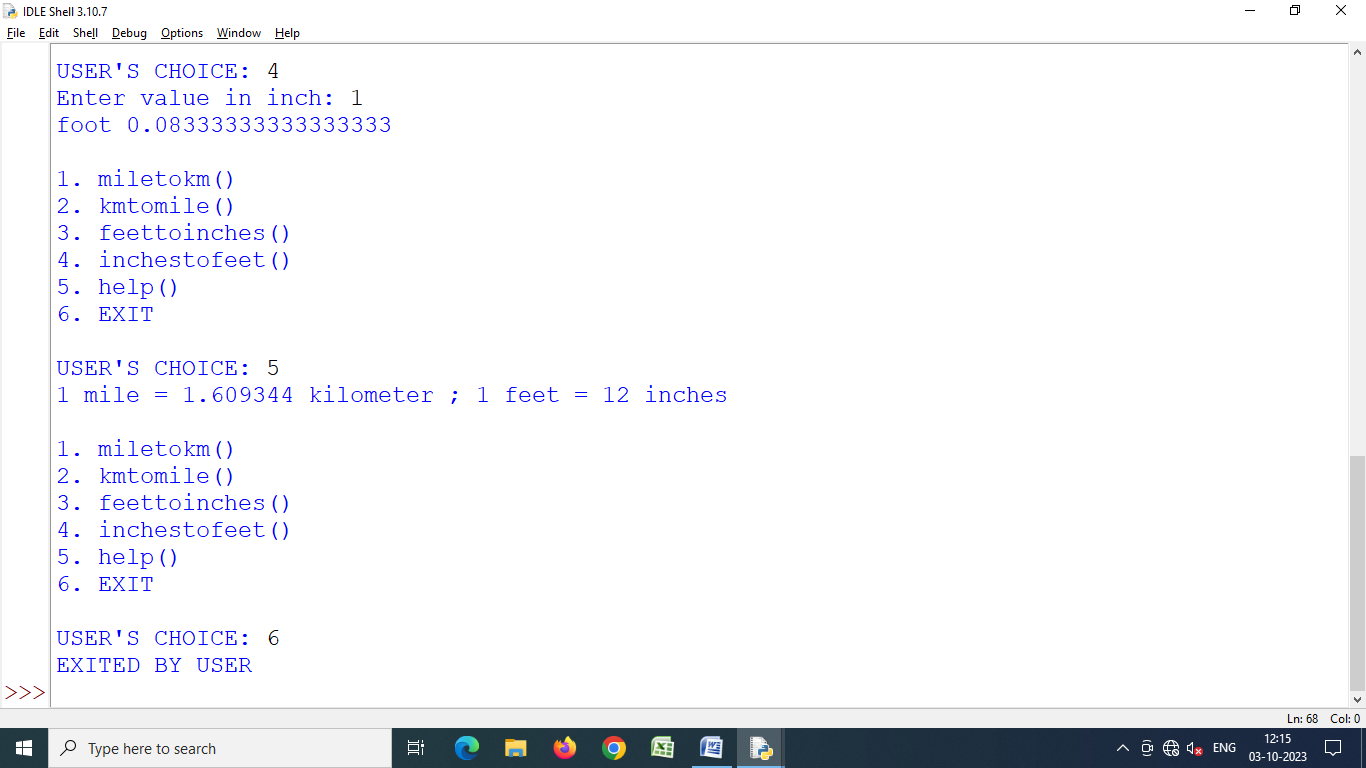
****

****

****

****

****

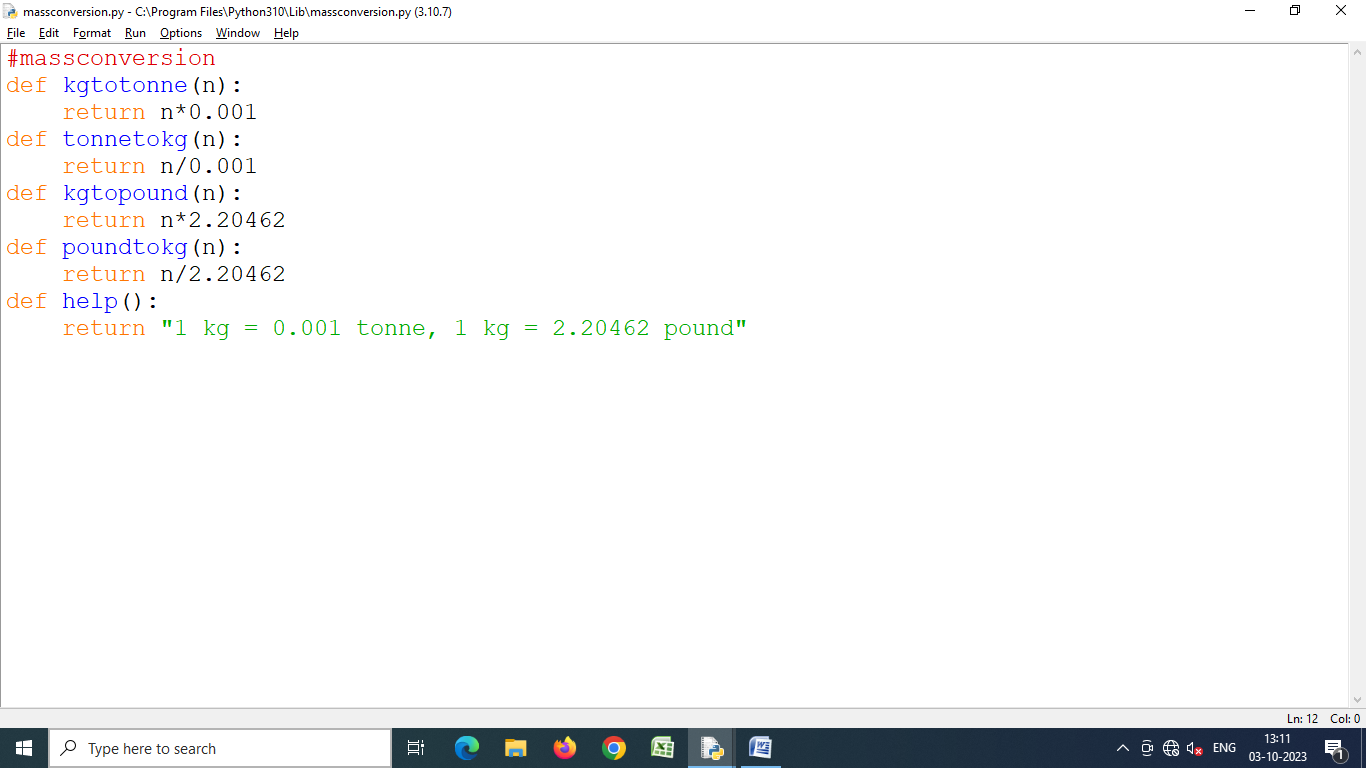
****

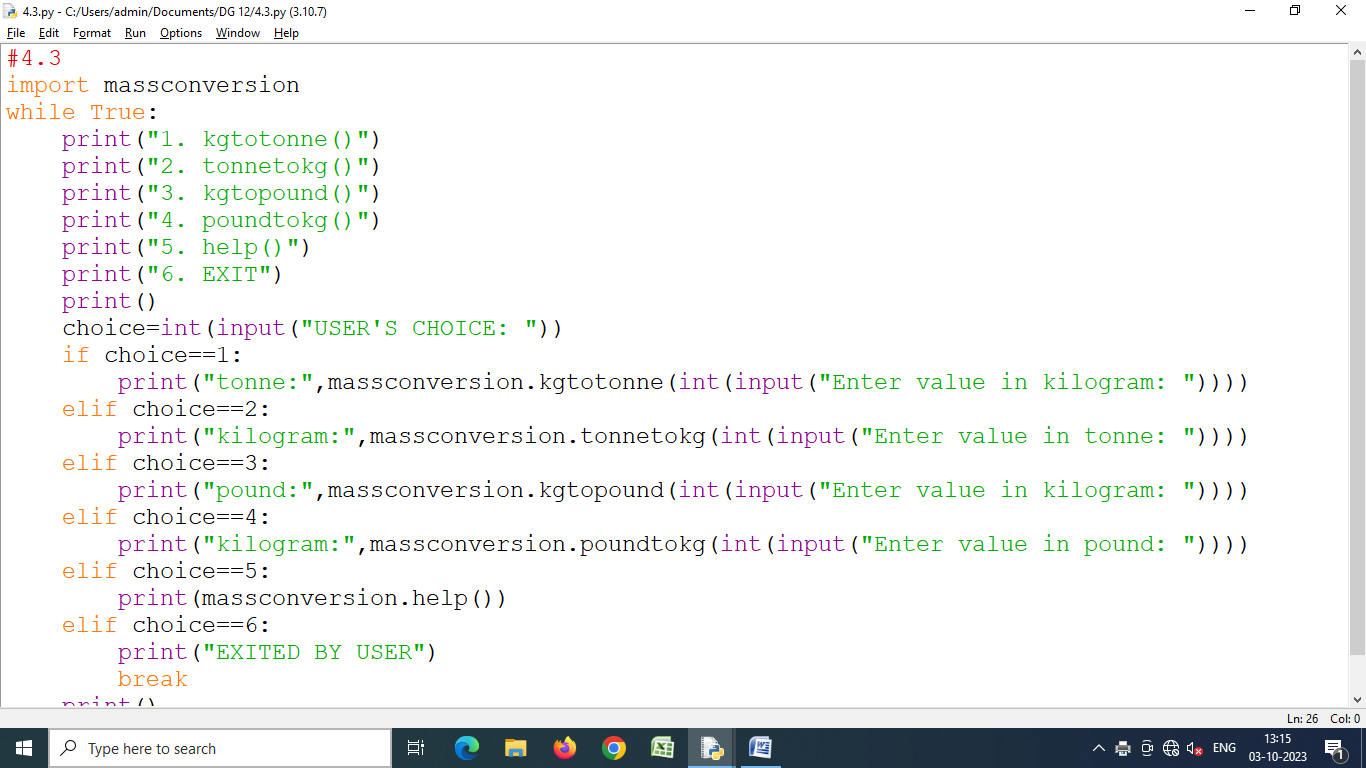
Program 4.3 **Create a module MassConversion.py that stores function for mass conversion e.g.,**

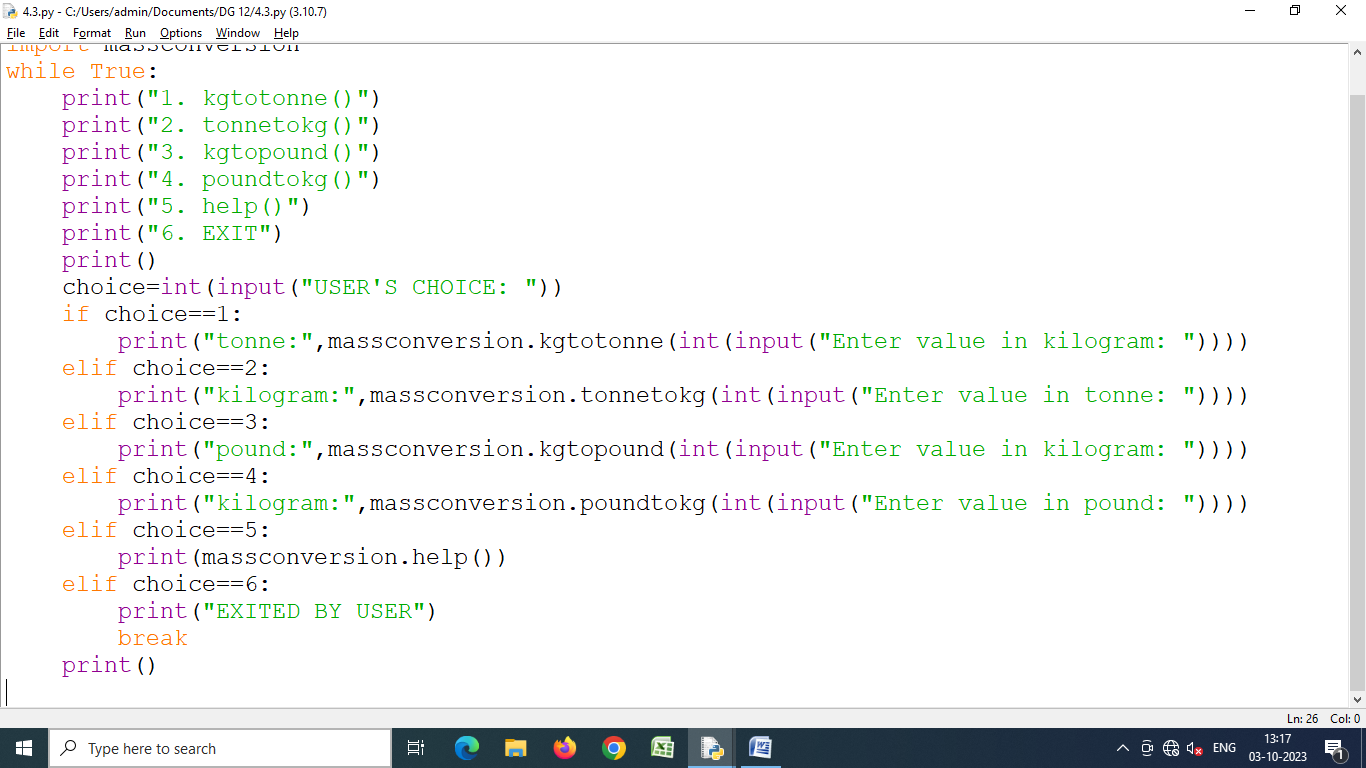
* **kgtotonne( ) to convert kg to tones • tonnetokg( ) to convert tonne to kg**
* **kgtopound( ) to convert kg to pound • poundtokg( ) to convert pound to kg**

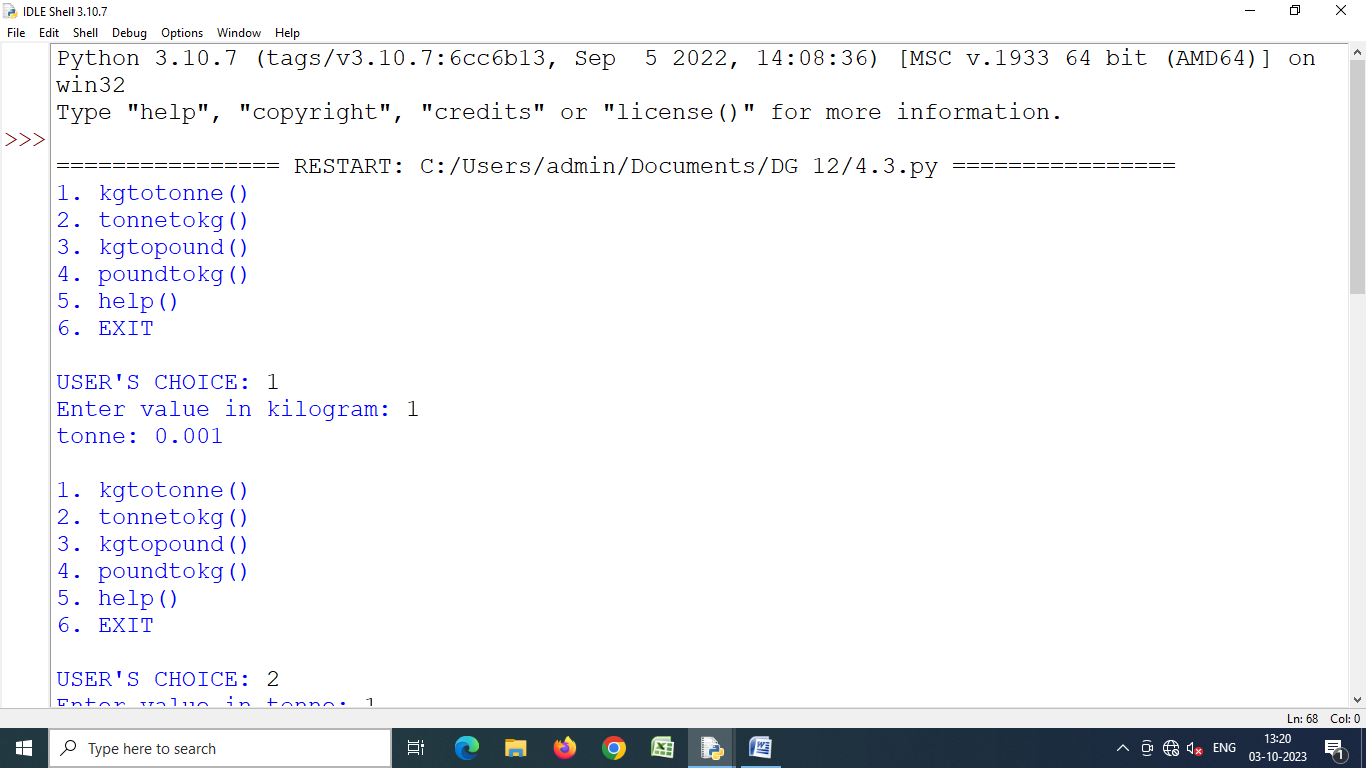
**(Also store constants 1 kg = 0.001 tonne, 1 kg = 2.20462 pound)**

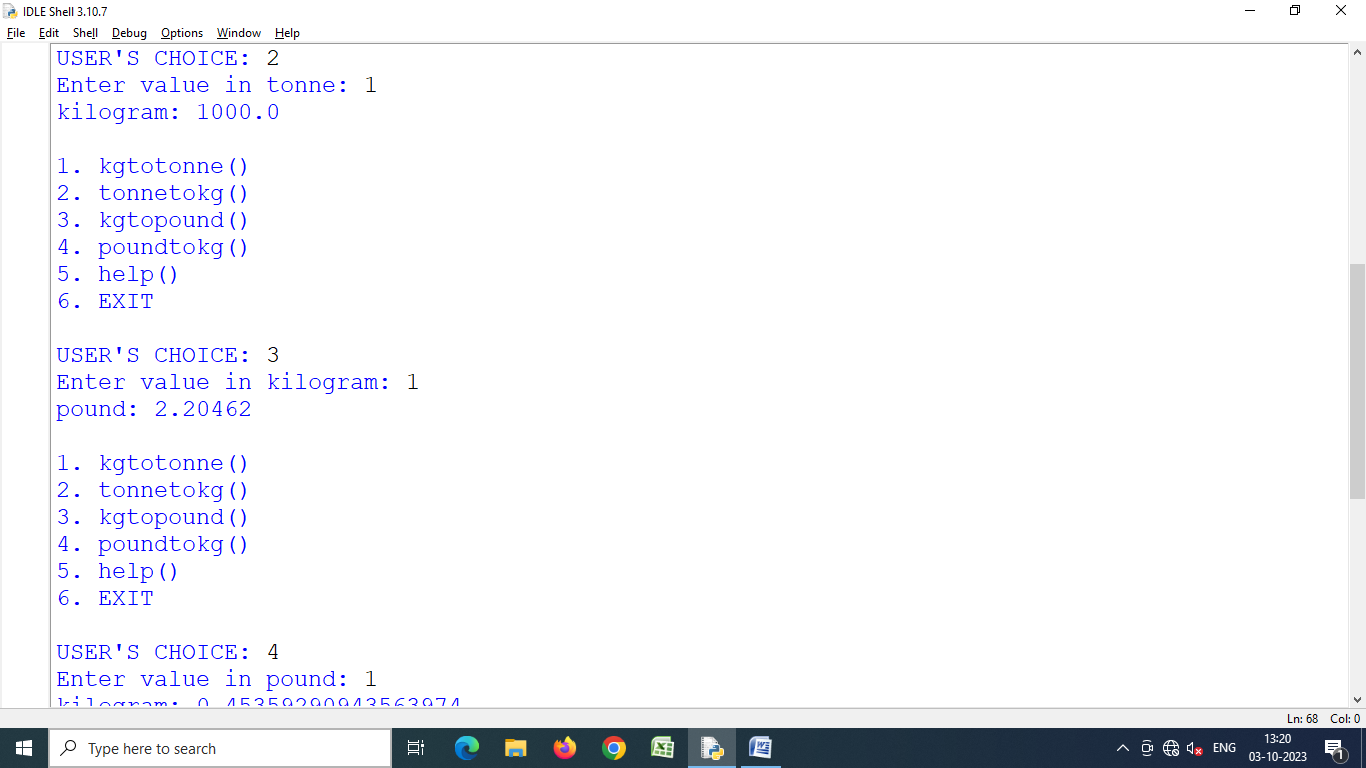
**Help( ) function should give proper information about the module.**

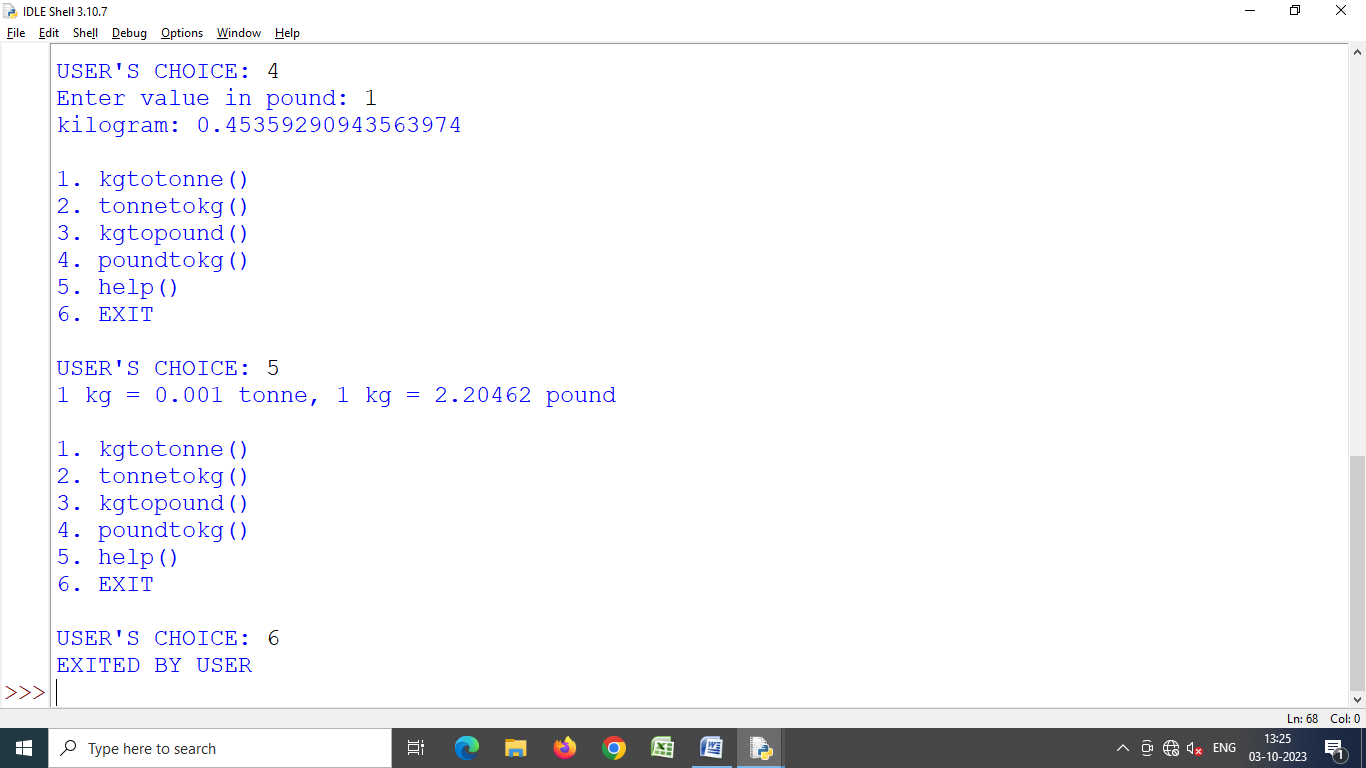
****

****

****

****

****

****

Program 4.4 **Create a package from above two modules as this :**

**Conversion**

**Length**

**Lengthconversion. py**

**Mass**

**Massconversion. py**

**Make sure that above package meets the requirements of being a Python package. Also, you should be able to import above package and/or its modules using import command.**

****

****