Assessment python

Day1-5

Practical 1: palindrome checker

Code:

#palindrome checker

# to check if the given word is same in reverse or not

#first we will create a function to reverse the string and match with given string

def palindromecheck(s):

    return s == s[::-1]

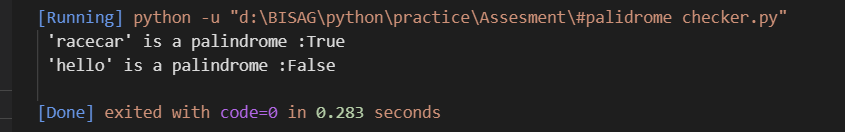
string1 = "racecar"

string2 = "hello"

print(f" '{string1}' is a palindrome :{palindromecheck(string1)}")

print(f" '{string2}' is a palindrome :{palindromecheck(string2)}")

output:



Practical 2: String anagram checker

Code:

#String anagram

# check if two words are made of same alphabets

#we will first make inout lower case then will use " Sort"  function to sort word alphabaticaly

def anagram(w1,w2):

    return sorted(w1.lower()) == sorted(w2.lower())

#### input words

word1 = input("Please enter first word:")

word2 = input("Please enter second word:")

##### now we will call function with inputs

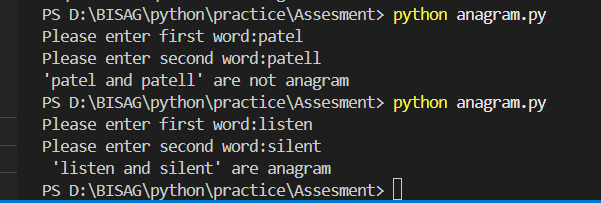
if anagram(word1,word2):

    print(f" '{word1} and {word2}' are anagram")

else:

    print(f"'{word1} and {word2}' are not anagram")

Output:



Practical 3: Factorial calculator

Code:

num = int(input("Enter a number: "))

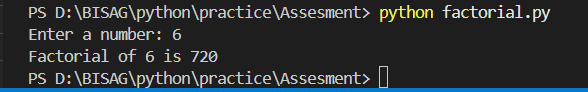
factorial = 1

for i in range(1, num + 1):

    factorial \*= i

print("Factorial of", num, "is", factorial)

output:



Practical 4: Fizzbuzz

#### Fizz Buzzz########

#we will pprint number 1 to 100

# print (replace)with  "Fizzbuzz" if n is divided by 3 and 5 ###

# print (replace)with "Fizz" if n is divided only by 3

# print (replace) with "buzz" if n is divided only by 5

for i in range(1,101):

    if i % 3 == 0 and i % 5 == 0:

        print("FizzBuzz")

    elif i % 3 == 0 :

        print("Fizz")

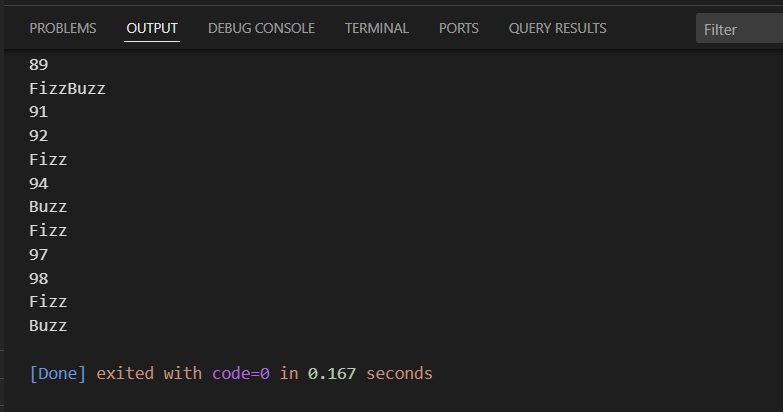
    elif i % 5 == 0 :

        print("Buzz")

    else:

        print(i)

output:



Practical 5: leap year checker

Code:

#leapyearchecker

#logic############################3

###year enterned divided by 4  is a leap year ###

### if year is divided by 100 it should also be divided by 400

####year is not divisible by 100, but was divisible by 4  then it's a leap year.

def is\_leap\_year(year):

    if (year % 4 == 0):

        if (year % 100 == 0):

            if (year % 400 == 0):

                return True

            else:

                return False

        else:

            return True

    else:

        return False

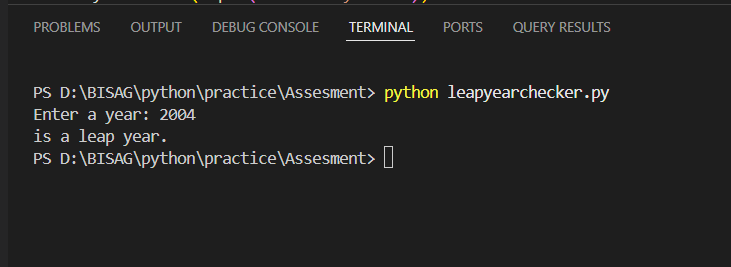
year = int(input("Enter a year: "))

if is\_leap\_year(year):

    print("is a leap year.")

else:

    print("is not a leap year.")

output:

Practical 6: List intersection

Code:

###### finding same value in  2 diffrent lists

list1 = [ 1, 2 , 3, 3 , 4 , 5]

list2 = [ 6, 7 , 8 , 2, 4 ]

intersection = []

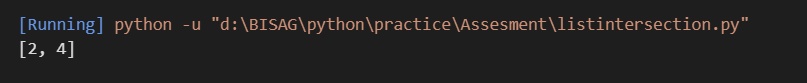
for i in list1:

    if i in list2:

        intersection.append(i)

print(intersection)

output:



Practical 7: Prime number

Code:

# A prime number is a number that is Greater than 1 Can only be divided by 1 and itself

num = 5

for i in range(2,num):

    if num % i == 0:

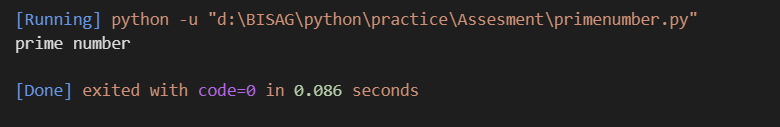
        print("not a prime number")

        break

else:

    print("prime number")

output:



Practical 8: word count

Code:

s = "Python is a snake not a language"

###split and lowercase the letters in given string

def wordcount(s):

    s = s.lower()

    words\_split = s.split()

    empty = {}

    for i in words\_split:

        if i in empty:

            empty[i] += 1

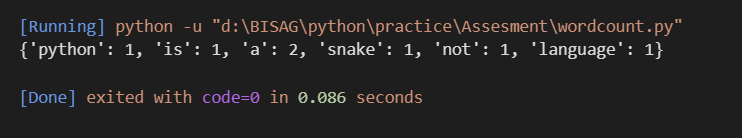
        else:

            empty[i] = 1

    return empty

print(wordcount(s))

Output:



Practical 9: Prime Factorization

Code:

def prime\_factors(n):

factors = []

i = 2

while i \* i <= n:

if n % i:

i += 1

else:

n //= i

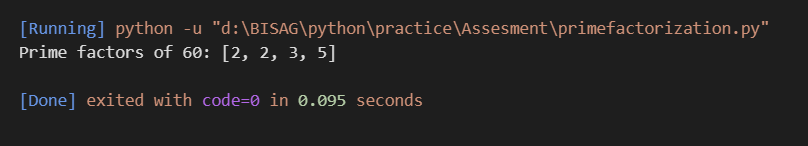
factors.append(i)

if n > 1:

factors.append(n)

return factors

Output:



Practical 10: Caeser cipher

Code:

def caesar\_cipher(text, key):

result = ''

for char in text:

if char.isalpha():

offset = 65 if char.isupper() else 97

result += chr((ord(char) - offset + key) % 26 + offset)

else:

result += char

return result

# Example:

print("Encrypted 'Hello World' with key 3:", caesar\_cipher("Hello World", 3))

# Output: Khoor Zruog

Output:

