Harithakrishnan
Ist year MCA A
Roll NO: 40
Amal Jyothi college of
Engineering

Python Lab Record

Date : 12-01-2021

program NO : 1

AIM : Program to check if else condition.

```
a = 12
b = 17
if (a > b):
    Print (" the value of a is greater than b")
else:
    Print (" the value of b is greater than a")
```

Result :- The program has been executed and the output was verified.

Output-

the value of b is greater than a.

Date : 06-01-2021

Program NO : 2

AIM : Program to find the square of a number entered by user.

```
num = int(input("enter a number: "))
square = num * num
print(f"square of {num} is {square}")
```

Result : The program has been executed and the output was verified.

Output

Enter a number 9

Square of 9 is 81

Date : 16-01-2021

Program No : 3

AIM : Program to find area of a circle using a function.

```
def circleArea(r):
    PI = 3.14
    return PI * (r*r)

n = float(input("enter the value"))
print("area is %.6f", %.circleArea(n))
```

Result : The program has been executed and the output was verified.

Output-

enter r value : 7

area is 153.860000

Date : 16-01-2021

Program NO : 3

AIM : program to find the biggest of three numbers entered by the user.

```
num1 = float(input("enter a number:"))
num2 = float(input("enter a number:"))
num3 = float(input("enter a number:"))

If (num1 > num2) and (num1 > num3):
        largest = num1

elif (num2 > num1) and (num2 > num3):
        largest = num3

Print("The largest number is", largest).
```

Result : The program has been executed and the output was verified.

Output

enter a number : 3
enter a number : 5
enter a number : 8
the largest number is 8.

Date : 27-01-2021

Program NO : 4

AIM : count the occurrences of each word in a line of text.

```
def word(str):
    counts = dict()
    words = str.split()
    for i in words:
        if i in counts:
            counts[i] += 1
        else:
            counts[i] = 1
    return counts
print(word('have a nice day'))
```

Result : The program has been executed and the output was verified.

output

{'have': 1, 'a': 1, 'nice': 1 'day': 1}

Date : 27-01-2021

Program No : 5

AIM : Store a list of first names, count the occurrences of 'a' within the list

```
newline = "say hai anna"
count = 0
for i in newline:
    if i == 'a':
        count = count + 1
Print("count of a in say hai anna : " + str(count))
```

Result : The program has been executed and the output was verified.

Output-

count of a in say hai anna : 4

Date: 27-01-2021

Program No: 6

AIM: Get a string from an input string where all occurrences of first character replaced with '$' except first character.

```
def change_char(str1):
  char = str1[0]
 str1 = str1.replace(char, '$')
 str1 = char + str1[1:]

 return str1

print(change_char('restart'))
```

Result: The program has been executed and the output was verified.

Output

```
restart
```

Date : 27-01-2021

Program NO : 7

AIM : Create a string from given string where first and last characters exchanged.

```
def change_string(str1):
    return str1[-1:] + str1[1:-1] + str1[:1]
print(change_string('abcd'))
```

Result : The program has been executed and the output was verified.

Output

dbca

Date : 27-01-2021

Program No : 8

AIM : Accept an integer n and compute n+nn+nnn.

```
n = int(input("enter a number"))
tp = str(n)
t1 = tp + tp
t2 = tp + tp + tp
comp = n + int(t1) + int(t2)
print("value", comp)
```

Result : The program has been executed and the output was verified.

Output

Enter a number : 4

Value 999

Date: 27-01-2021

Program NO: 9

AIM: Merge two dictionaries

```
dict1 = {'a': 10, 'b': 8}
dict2 = {'d': 6, 'c': 4}
def merge(dict1, dict2):
    return(dict2.update(dict1))
print(merge(dict1, dict2))
print(dict2)
```

Result: The program has been executed and the output was verified.

Output

{'d': 6, 'c': 4, 'a': 10, 'b': 8}

Date : 03-02-2021

Program No : 10

AIM : program to find the factorial of a number

```
num = int(input("enter a number"))
factorial = 1
If num < 0:
        Print("factorial does not exist for negative
              numbers")
elif num == 0:
        Print("the factorial of 0 is 1")
else:
    for i in range(1, num+1):
        factorial = factorial * i
Print("The factorial of", num "is", factorial)
```

Result : The program has been executed and the output verified.

Output

Enter a number: 7

The factorial of 7 is 5040.

Date : 03-02-2021

Program No : 11

AIM : Generate fibonacci series of N terms

```
n = int(input("enter the value of 'n' : "))
n = int(input("enter the
a = 0
b = 1
sum = 0
count = 1
print("fibonacci series:" end = " ")
while(count <= n):
    print(sum, end = " ")
    count += 1
    a = b
    b = sum
    sum = a+b
```

Result : The program has been executed and the output was verified.

Output

Enter the value of n : 5

0 1 1 2 3

Date : 03-02-2021

Program NO : 12

AIM : Find the sum of all items in a list.

```
sum = 0
li = [12, 134, 5, 4, 21]
for val in range[0, len(li)):
        sum = sum + li[val]
Print ("sum of all values in a given list",
        sum)
```

Result : The program has been executed and the output was verified.

Output

Sum of all values in a given list : 76

Date: 03-02-2021

Program No : 13

AIM : Count the number of characters in a string.

```
def char_freq(str1):
    dict = {}
    for n in str1:
        keys = dict.keys()
        If n in keys:
            dict[n] += 1
        else:
            dict[n] = 1
    return dict
print( char_freq('pathanamthitta')
```

Result : The program has been executed and the output was verified.

Output

{'p':1, 'a':4, 't':4, 'b':2, 'n':1, 'm':1
'i':1}

Date: 03-02-2021

Program No: 14

AIM: Adding 'ing' at the end of a given string. If it already ends with 'ing' then add 'ly'

```
def add_string(str1):
    length = len(str1)
        if length > 2:
            if str1[-3:] == 'ing':
            else:
                str1 += 'ing'
            return str1
print(add-string('play'))
print(add-string('playing'))
```

Result: The program has been executed and the output was verified.

Output

playing

playingly

Date : 03-02-2021

Program No : 15

AIM : Accept a list of words and return length of longest word.

```
def longest_word(words_li):
    word_len = []
    for n in words_li:
        word_len.append((len(n), n))
        word_len.sort()
    return word_len[-1][0], word_len[-1][1]
result = longest_word(["rose", "Jasmine", "lotus"])
print("Longest word :", result[1])
print("length of the longest word :", result[0])
```

Result : The program has been executed and the output was verified.

Output-

Longest word : Jasmine

Length of the longest word : 7

Date : 03-02-2021

Program No : 16

AIM : Generate all factors of a number.

```
n = int(input("enter n"))
factors = [1]
for i in range(2, n):
    if(n % i == 0):
        factors.append(i)
        factors.append(n)
print(factors)
```

Result : The program has been executed and the output was verified.

Output:

enter n 4

[1, 2, 4]

Date: 03-02-2021

Program No: 17

AIM: Lambda functions to find area of a square, rectangle and triangle.

```
s = int(input("enter the sides : "))
x = lambda a: a*a
print(x(s))
l = int(input("enter length : "))
b = int(input("enter breadth : "))
y = lambda l, b: l*b
print(y(l, b))
h = int(input("enter the height of triangle : "))
b1 = int(input("enter the base of triangle : "))
z = lambda h, b1: (h*b1)/2
print(z(h, b1))
```

Result: The program has been executed and the output was verified.

output

enter the sides : 4

16

enter the length : 6

enter breadth : 3

18

enter height of triangle : 7

enter base of triangle : 2

6.0

Date : 03-02-2021

Program No : 18

AIM : From a list of integers, create a list removing even numbers.

```
num = [5, 7, 20, 11, 18, 25, 4]
num = [x for x in num if x%2!=0]
print(num)
```

Result : The program has been executed and the output was verified.

Output

[5, 7, 11, 25]

Date: 03-02-2021

Program NO: 19

AIM : Display the given pyramid with step number accepted from user.

N = 4

```
1
2 4
3 6 9
4 8 12 16
```

```
def num():
    n = int(input("enter the number:"))
    i = 1
    for i in range(1, n+1):
        j = 1
    for j in range(1, i+1):
            temp = i * j
        print(temp, end = " ")
        print(" ")
num()
```

Result : The program has been executed and the output was verified.

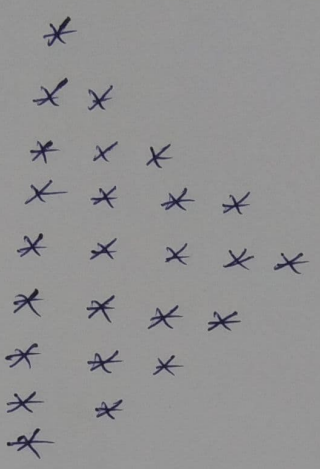
Output

```
Enter the number : 4
1
2 4
3 6 9
4 8 12 16
```

Date : 03-02-2021

Program No : 20

AIM : construct following pattern using nested loop.



```
n = 5:
for i in range(n):
    for j in range(i):
        print('*', end(" "))
    print(')
for i in range(n, 0, -1):
  for j in range(i):
      print('*', end = ' ')
  print(' ')
```

Result : The Program has been executed and the output was verified.

Output