

Program No: 1

AIM: Python program to find area of circle.

```
def area(r):
    pi = 3.14
    return pi * (r * r)
num = float(input("Enter the value for:"))
print("Area is %.6f", area(num))
```

RESULT

The program has been executed and the output was verified.

OUTPUT

Enter the value for: 3
Area is 28.26000

PROGRAM NO:2

AIM:

Write a python program to find the square entered by the user.

```
def sq():
```

```
    x = int(input("Enter a number:"))
```

```
    x = x * x
```

```
    print("square: ", x)
```

RESULT

The program has been executed and the output was verified.

OUTPUT

Enter a number of 4 digit: 16
Square: 16

$$4 \times 4 = 16$$

$$(4 \times 4) + 4 = 16$$

((4 * 4) + 4) * 4 = 16
(((4 * 4) + 4) * 4) + 4 = 16

between and and morphing
between and and morphing

PROGRAM NO:3

AIM: Python program to find largest among three numbers.

```
n1=float(input("Enter the 1st number:"))
n2=float(input("Enter the 2nd number:"))
n3=float(input("Enter the 3rd number:"))

if (n1>n2) and (n1>n3):
    print("Largest is ", n1)
elif (n2>n3):
    print("Largest is ", n2)
else:
    print("Largest is ", n3)
```

RESULT

The program has been executed and the output was verified

OUTPUT

Enter the 1st number : 2

Enter the 2nd number : 4

Enter the 3rd number : 5

Largest is 5

$$x * x = x$$

(x * : assume * is +)

has numbers and and comparing info
half less than sum of digits add

PROGRAM NO : 4

INTRODUCTION

AIM:

python program to find area
square of n

```
def sq():
    n = int(input("Enter the total number"))
    list = []
    for i in range(n):
        val = int(input("Enter a number : "))
        list.append(val)
    for i in list:
        print("square of {} is {}".format(i, i*i))
```

RESULT

The program has been executed
and the output was verified.

OUTPUT

Enter the total number: 4

Enter a number: 5

Enter a number: 3

Enter a number: 2

Enter a number: 1

Square of 5 is 25

Square of 3 is 9

Square of 2 is 4

Square of 1 is 1

total is 50

total is 50

total is 50

total is 50

using nested loop output will
be like now square of 5 is 25

PROGRAM NO:5

AIM

Python program to form a list of vowels selected from a given word.

```
def vow():
    word = input("Enter a word")
    vow = "aeiou AEIOU"
    v1 = [each for each in word if
          each in vow]
    print(v1)
```

RESULT

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

OUTPUT

RISHI MANGAL

enter a word : ashish

[a'ɪʃɪʃ]

English meaning of the word : (1) p2 Job

((("ashish" bedeutet sch"nig") twgnt) fnt = o

meaning of the word : (2) p2 Job

meaning of the word : (3) p2 Job

((("ashish" bedeutet sch"nig") twgnt) fnt = o)

(o)vi kroppa, t21

: H2W n2 i rof

((("ashish" bedeutet sch"nig") twgnt) fnt = o)

bedeutet und znd mörpong mit

mehreren new twgnta eff PWD

PROGRAM NO: 6

AIM:

python program to display future leap years from current year to a final year entered by user.

```
x = int(input("Enter a year to check leap year"))
if (x % 400 == 0) or ((x % 4 and x % 100 != 0)):
    print(x, "is a leap year")
else:
    print(x, "is not a leap year")
```

RESULT

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

OUTPUT

Enter a year to check leap year be not 2021
2021 is leap year.

(i) know P file + 1 input = know

"00131 nois" = know

If know ai does not done 7 = IV

[know] ai does

(IV) failing

knows and not ongoing int.

hitting and 2nd trying with bus

PROGRAM NO:7

TOPPER

AIM

Python program to count the occurrences of each word in a line of text

```
def count():
    txt = input("Enter a string")
    count = 1
    for i in txt:
        if (i == " "):
            count += 1
    print("The count of word is", count)
```

RESULT

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

OUTPUT

Enter a string how are you

The count of words is : 3

Count of words in the string is 3

(Input string a - sleep is intelligen) total = 3
(a = sleep is intelligen) p.
(sleep is intelligen) total = 3

(sleep is intelligen) total = 3

Ans 3 strings and last character is
newline and last character is

PROGRAM NO : 8AIM

Python program from input string
where all occurrence of first character
replaced with '\$' except first character

```
def string():
    string = input("Enter a string : ")
    char = string[0]
    string = string.replace(char, '$')
    print(char + string[1:])
```

RESULT

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

OUTPUT

None

Enter a string: onion

onion

is an a vowel does it

is it was

(empty is null) thing = true

t = true

it is not

(r k = i) fi

t = true

(true) if draw {t true str } thing

None

but works and and mapping all

keflel and and output all

PROGRAM NO - 9

TUTORIAL

AIM

python program to create a string where first and last characters exchanged.

```
def reverse(getstr):  
    return getstr[::-1]
```

RESULT

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

OUTPUT

reverse ("python")

'nohtyP' most common method

reverse() most common method

: O print len

(len(s) - i) : print s[i]

[0] print = end

(i+1, end) : print print = print

([0:i] print + end) : print

basics and print mapping int
basics and print strings with know

PROGRAM NO :- 10AIM

python program to accept an integer n and compute $n+nn+nnn$

```
def nsum():
    n = int(input("Enter a number:"))
    a = n
    b = n + n
    c = n + n + n
    nsum = 0
    nsum = int(a) + int(b) + int(c)
    print(nsum)
```

RESULT

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

OUTPUT

Enter a number : 5

: (top) 523456789

[E-1] top answer

PROGRAM No: 11

AIM

python programs to merge two dictionaries

```
def merge():
```

```
    x = {'a': 1, 'b': 2}
```

```
    y = {'b': 10, 'c': 11}
```

```
    z = x.update(y)
```

```
    print(z)
```

```
    print(x)
```

RESULT

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

OUTPUT

None

{ 'a': 1, 'b': 20, 'c': 11 }

addition of numbers

: (sum of job

((a + b) + c) + d = a + b + c + d

$$a+b+d$$

$$a+b+c = d$$

$$a+a+a = d$$

$$a+m+n = d$$

$$(a+m)+n+(d-n) = m+n+d$$

(constant) taking

RESULTS

hence add and compare it

thus add and see output of program

PROGRAM NO: 12

AIM

python program to find gcd of two numbers

```
def gcd():
    a = int(input("Enter first number"))
    b = int(input("Enter second number"))
    while b != 0:
        r = a % b
        a = b
        b = r
    print("The GCD of the numbers is", a)
```

RESULT

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

OUTPUT

Enter first number 40

Enter second number 50

The GCD of the numbers is 10

: (0) sum - 1 do

Factor 1 : $x^2 + 3x + 2 = 0$

Factor 2 : $x^2 + 2x + 1 = 0$

(x+1)(x+2) = 0

(x)twins

(x)twins

Two factors add up to zero

Halfway add and bring it

PROGRAM NO: 13

TUTORIAL

AIM

python program to from a list of integers, create a list removing even numbers

```
def list1():
    list1 = []
    odd = []
    num = int(input("Total number of elements:"))
    for i in range(num):
        list1.append(int(input("Enter a number")))
    odd = [each for each in list1 if each % 2 == 0]
    print(odd)
```

RESULT

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

OUTPUT

Total number of elements : 5

Enter a number : 1

Enter a number : 2

Enter a number : 3

Enter a number : 4

Enter a number : 5

[1, 3, 5]

d = p

d = s

r = d

(a) if medium set p (and INT) + temp

PROGRAM NO: 14AIM

python program to find the factorial of a number

```
def fac(n):
    fact = 1
    if n == 0:
        print(n, "!", "=", fact)
    for i in range(1, n+1):
        fact = i * fact
    print("n !")
    print(n, "!", "=", fact)
n = int("Enter a number : n"))
fac(n)
```

RESULT

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

OUTPUT

Enter a number : 4

$$4! = 24$$

: 0.000 sec

□ = 1.000

□ = 0.000

: count space is 3 more

: ("address of ans") + 4 * 4 = 16

: ("address of ans") + 4 * 4 + 1 = 17

(char) saving

RESULT

ans becomes 0 and memory unit
contains 0 and ans contains 16

PROGRAM NO : 15

AIM

Python program to generate Fibonacci series of N terms.

```
def fib(n):
    a, b = 0, 1
    while a < n:
        print(a)
        a, b = b, a+b
n = int(input("ENTER a number :"))
fib(n)
```

RESULT

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

OUTPUT

Enter the number : 5

0

1

1

2

43

(a) sqf. b.

B = 3.142

: 0 = 0.14

(sqf. = 1" a) find

: (Circles) area of is 0.14

Sqf. R = Sqf

+ missing

(sqf. = 1" a) find

((* : admiral's 3.142) find = 1

(a) last

between and and overlap set
between and and within set and

PROGRAM NO: 16AIM

Python program to display the given pyramid with step number accepted from user.

```
x = int(input("Enter a number"))
j, i = 1, 1
for i in range(1, x+1):
    p = i
    for j in range(1, i+1)
        p = i*j
        print(p, end=" ", flush=True)
        print(" ", end=" ")
    print("\n")
```

RESULT
@VAPR27

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

OUTPUT

Enter a number 5

1,

2, 4,

3, 6, 9,

4, 8, 12, 16

5, 10, 15, 20, 25

PROGRAM NO:17AIM

Add 'ing' at the end of given string.
If it already ends with 'ing', Then add 'ly'

```
def adstr(s1):
    length = len(s1)
    if length > 2:
        if s1[-3:] == 'ing':
            s1 += 'ly'
        else:
            s1 += 'ing'
    return s1
```

```
print(adstr('happy'))
print(adstr('playing'))
```

RESULT

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

OUTPUT

happening

playfully

("admire a film") taking photo = x

: (admire a film) taking photo = x

(take a photo of it) taking photo = q

[$x = q$]

(watch a movie, a video, a film) taking

(watch a movie, a film) taking

(watch a film) taking

प्रज्ञान
प्रसवा

and between and among up
between and two. at

PROGRAM 18AIM

Python program to generate all factors of a number.

```
def fact(n):
    print("The factors of", n, "are:")
    for i in range(1, n+1):
        if n % i == 0:
            print(i)
```

n=100

fact(n)

RESULT

The program has been executed and output was verified.

OUTPUT

the factors of 100 are:

1

2

4

5

10

20

25

50

100

((program) starts) doing
((program) does) thing

Program 19

AIM

HTML Lambda functions to find area of square, rectangle and triangle.

$s = \text{int}(\text{input}("Length of square:"))$

$l = \text{int}(\text{input}("Length of rectangle:"))$

$b = \text{int}(\text{input}("Length of one the breadth of rectangle:"))$

$b = \text{int}(\text{input}("Enter base of triangle:"))$

$d = \text{int}(\text{input}("Enter height of triangle:"))$

$x = \lambda s: s * s$

$y = \lambda l, b: l * b$

$t = 0.5$

$z = \lambda h, d, t: h * d * t$

$\text{print}("Area of Square is:", x(s))$

$\text{print}("Area of rectangle is:", y(l, b))$

$\text{print}("Area of triangle is:", z(h, d, t))$

RESULT

The program has been executed and the output was verified.

INPUT

length of square : 2

length of rectangle : 2

Enter the breadth of rectangle : 2

Enter base of triangle : 3

Enter height of triangle : 2

Area of square is 4

Area of rectangle is 4

Area of triangle is 3.0

Program 20

AIM

Python programs to perform the sum of given items

```
numbers = [1, 2, 3, 4, 5, 2, 5]
```

```
sum = sum(numbers)
```

```
print(sum)
```

RESULT

The program has been executed and the output was verified.

OUTPUT

22

half of zirconium obtained XMA
done it and obtained sample to area

(C) sample to obtain weight tri = 2

(E) sample to obtain weight tri = 1

(L) sample to obtain weight tri = 1

((O) reaction for standard

(C) sample to obtain weight tri = 4

(E) sample to obtain weight tri = 4

2x3 = 6 obtained = X

2x1 = 2 obtained = Y

2.0 = Z

standard obtained = Z

((O) x 2) sample to area 1) tri = 1

((E) sample to area 1) tri = 1

((L) sample to area 1) tri = 1

bottom used and removing all
material now trying at best

Program No: 21

AIM:

Python program to find perfect even square number in Range

```
num1 = int(input("Enter two Numbers"))
num2 = int(input())
for i in range (num1, num2+1):
    for j in range (32, 100+1):
        if i == j*j:
            string = str(i)
            if int(string[0])%2 == 0 and
               int(string[1])%2 == 0 and
               int(string[2])%2 == 0 and
               int(string[3])%2 == 0
            print(i)
```

RESULT

The program has been executed
and output was verified.

OUTPUT

Enter two Numbers: 4444

9999

4624

6084

6400

2464

[2,5,2,48,8,8] = random
random() and rand
rand() func.

program No: 22AIM:

A python program to count the number of character in a string

```
def chars(str1):
    dict = {}
    for n in str1:
        keys = dict.keys()
        if n in keys:
            dict[n] += 1
        else:
            dict[n] = 1
    return dict
```

```
print(chars('hello how are you'))
```

RESULT

The program has been executed
and output was verified.

OUTPUT

$\{k: 2, 'e': 2, 't': 2, 'o': 3, 'w': 1, 'a': 1, 'r': 1,$
 $'y': 1, 'v': 1\}$

Constants and their frequency
(0.0001) to 0.0003
: (0.0003, 0.0001) goes to 0.0001
0.0001 to 0.0001
; 0.0001 to 0.0001
. (0.0001 to 0.0001)

0.0001 to 0.0001 (0.0001 to 0.0001)
0.0001 to 0.0001 (0.0001 to 0.0001)
0.0001 to 0.0001 (0.0001 to 0.0001)
0.0001 to 0.0001 (0.0001 to 0.0001)
. (0.0001 to 0.0001)

numbers and add memory int.
. before now the two bus

Program No: 23

AIM:

Python program to accept a list of words and return of longest word

```
def find(word):
```

```
    wI = []
```

```
    for n in word:
```

```
        wI.append(len(n))
```

```
    wI.sort()
```

```
result = wI[-1][0], wI[-1][1]
```

```
print("longest word:", result[0])
```

```
print("length of longest word:", result[1])
```

```
find(["hello", "morning", "hi"])
```

RESULT:

The program has been executed
and output was verified.

OUTPUT

longest word: morning
length of the longest word: 7

program No: 2ndAIM:

python program to construct pattern
using nested loop

```
def star():
```

```
n=5
```

```
for i in range(n):
```

```
    for j in range(i):
```

```
        print("*", end=" ")
```

```
    print("\n")
```

```
    for i in range(n, 0, -1):
```

```
        for j in range(i):
```

```
            print("*", end=" ")
```

```
        print("\n")
```

```
star()
```

RESULT:

The program has been executed and output is verified.

OUTPUT

A grid of 20 hand-drawn asterisks arranged in four rows of five on lined paper. The asterisks are drawn in blue ink and are of varying sizes and orientations, some pointing up and some pointing down. They are spaced evenly apart both horizontally and vertically.

Program No: 25

AIM:

python program to find factors of a Number.

```
def pfact(x):
```

```
    print("The factors of ", x, " are : ")
```

```
    for i in range(1, x+1):
```

```
        if x % i == 0
```

```
            print(i)
```

```
p-fact(232)
```

Result:

The program has been executed and output was verified.

OUTPUT

The factors of 232 are:

1

2

4

8

29

58

116

232

program 26:

AIM:

Python program to display future leap year from current year to final year entered by user.

```
import datetime
```

```
a = datetime.datetime.now()
```

```
a = int(a.year)
```

```
b = int(input("Enter final year"))
```

```
print("Leap Year :")
```

```
for i in range(a, b+1):
```

```
    if (i%4 == 0):
```

```
        print(i)
```

RESULT:

The output

program has been created and
was verified.

OUTPUT

Enter final year 2040
Leap Years:

2024

2028

2032

2036

2040

Program 27

Aim:

python program to generate positive list of number from a given list of integers.

```
list1 = [1, -2, 3, -4, 5, -6, -7, -8]
```

```
pos = list()
```

```
for i in list1
```

```
    if i > 0:
```

```
        pos.append(i)
```

```
print ('original list:', list1)
```

```
print ('positive integers list:', pos)
```

RESULT

The program has been executed and the output was verified.

Output

Original List : [1, -2, 3, -4, 5, -6, -7, 8]

positive List : [1, 3, 5, 8]

Program 28

AIM

python program to create a list of colors from comma-separated colors names entered by user display first and last colors.

```
colors = input("Enter colors separated by commas : '").split(',')
print ("First color : ", colors[0])
print ("Last color : ", [len(colors)-1])
```

RESULT

The program has been executed and output was verified.

Output

Enter colors separated by commas : red, black,

yellow

First color: Red

Last color: Yellow

Program 29:

AIM:

Python program to print out all colors from color_list1 not contained in color_list2

```
print("Enter colors separated by commas : ")
color1 = set(input().split(','))
color2 = set(input().split(','))
```

```
print('color in color_list1 not contained  
color_list2 are : ', list(color1.difference  
(color2)))
```

RESULT

The program has been executed and the output was verified.

Output

Enter colors separated by commas

red, yellow, brown
black, white

Colors in ColorList1 not contained in
ColorList2: ['brown', 'red', 'yellow']

Input boxes and sort compare with
different colors together

Program No: 80

AIM

Python program to create a single string separated with space from two string by swapping the character at position 3.

```
string1 = input("Enter a String")
```

```
str2 = input("Enter another string")
```

```
str3 = str2[0] + string1[1:] + string1[0] + str2[1:]
```

```
print(str3)
```

RESULT

The program has been executed and output is verified.

OUTPUT

Enter a string: Ashish
Enter another string: Wilson
Ashish's Wilson
ashish wilson

Program 31

Ques:

Create a Bank account with members account name, type of account and balance write constructor methods to deposit at bank.

Class Bank :

```
def __init__(self, a, n, t, b):
```

```
    self.acno = a
```

```
    self.name = n
```

```
    self.type = t
```

```
    self.bal = b
```

```
def deposit(self, a):
```

```
    self.bal += a
```

```
print('Rs.', 'a', 'deposited! current
```

```
balance is Rs.', self.bal)
```

```
def withdraw(self, a):
```

```
    if self.bal >= a
```

```
        self.bal -= a
```

```
    print('Rs.', 'a' 'withdrawn! current
```

```
balance Rs.', self.bal)
```

```
else:
```

```
    print('Insufficient balance')
```

```
a = int (Input ('Enter account number'))  
n = input ('Enter name of the account holder')  
t = input ('Enter account type : ')  
b = float (input ('Enter a balance: '))  
act = Bank Account (a, n, t, b)  
act.deposit (float (input ('Enter account to  
deposit : ')))  
act.withdraw (float (input ('Enter account to  
withdraw : ')))
```

RESULT

The program has been executed and
output is verified.

OUTPUT

Enter account number : 101

Enter name of the account holder : Ashish

Enter account type : Savings

Enter your balance : 1052

Enter amount to deposit : 500

Rs 500 deposited : current balance Rs : 1552

Enter amount to withdraw : 200

Rs 200 withdraw current balance is Rs 1352.0

Bank savings and withdraw bank
balance is updated

Program No: 32.

AIM:

Create a class Rectangle with private attribute length and width 'c' operator to compare the area of two rectangle.

Class Rectangle:

```
def __init__(self, l, w):
```

```
    self.__length = l
```

```
    self.__width = w
```

```
    self.area = self.width * self.length
```

```
def __init__(self, other):
```

```
    if self.area < other.area:
```

```
        print("Rectangle with length =", self.length  
              and width = "self.__width "has less area")
```

```
    elif other.area < self.area
```

```
        print("Rectangle with length =", other.length  
              and width = "other.width "has less area")
```

```
else
```

```
    print("They are equal")
```

```
l = float(input("Enter length of 1st rectangle"))
```

```
w = float(input("Enter width of 1st rectangle"))
```

```
R1 = Rectangle(l, w)
```

```
L = float (input ("Enter length of 2nd rectangle"))
w = float (input ("Enter width of 2nd rectangle"))
R2 = Rectangle (L,w)
R1 < R2
```

RESULT

The program has been executed and
output is verified

OUTPUT.

Enter length of 1st Rectangle : 5

Enter width of 1st Rectangle : 3

Enter length of 2nd Rectangle : 9

Enter width of 2nd Rectangle : 6

Rectangle with length = 5 and width 30
has the lesser area.

Program 33

AIM

Create a class Time with private attribute hour, minute and second. overload '+' operator to find sum of 2 time.

Class Time :

def __init__(self, hh=0, mm=0, ss=0)

self.hour = hh

self.minute = mm

self.second = ss

def __add__(self, other)

second = int((self.second + other.second) % 60)

minute = int((self.minute + other.minute) % 60 +

((self.second + other.second) // 60))

hour = int((self.hour + other.hour) % 24 +
self.minute + other.minute) // 60

print('Time[hh:mm:ss]', hour, ':', minute, ':', second)

T1 = Time(11, 35, 15)

T2 = Time(10, 15, 46)

T1 + T2

RESULT

The program has been executed and output is verified.

OUTPUT.

Time[hh:mm:ss] 21:51:1

Program No: 34.

AIM:

Create a string & repeated with space from two string by swapping character at position

String = Input ["Enter a string separated by commas :"]

String = String . split (',')

print (String[1][0] + String[0][1:] + " " +
String[0][0] + String[1][1:])

Output

The program has been executed
and output is verified.

OUTPUT

Enter → String by comma : hi, hello
i hello

Program No: 35

AIM:

Create a class publisher (name). Derive class book from publisher with attribute title and author. Derive class python from Book with attributes price and no-of-pages. Write program that display information about a python book.

class publisher:

```
def __init__(self, name):  
    self.name = name
```

class Book(publisher):

```
def __init__(self, title, author, name):  
    self.title = title  
    self.author = author  
    publisher.__init__(self, name)
```

class Python(Book):

```
def __init__(self, p, no, title, author):  
    self.price = p
```

self.no_of_pages = no.

```
Book.__init__(self, title, aut)
```

```
def show(self):  
    print("BOOK Title:", self.title)  
    print("Author:", self.author)  
    print("Name:", self.name)  
    print("Publisher:", self.price)  
    print("No.of.pages:", self.no.of.pages)
```

P1 = Python(144.67, 300, 'Python', 'Py',
 'PyPub')

P1.show()

RESULT

The program has been executed
and output was verified.

OUTPUT.

Book Title: Python

Author : Py

Publisher : py pub

Price : Rs. 144.67

No. of pages: 300

Program No:36

AIM

write a python program to read a file line by line and store it into a list?

```
def fileread(fname):
    with open (fname) as f:
        c = f.readline()
        print(c)
    file-read("file2.txt")
```

RESULT

The program has been executed and output is verified

Output

[41. Hi how are you An 2. This is
world An 3; It is the introduction