

## Program No: 26

Aim: Program to Find Area Of Square, Triangle And Rectangle Using Lambda Functions.

```
s = int(input("Enter the length of a side of square:"))
l = int(input("Enter the length of rectangle:"))
b = int(input("Enter the breadth of rectangle:"))

print('Enter the base and height of triangle')
h = int(input("Enter the base of triangle:"))
d = int(input("Enter the height of triangle:"))

x = lambda s: s*s
y = lambda l,b: l+b
t = 0.5

g = lambda h,d,t: h*d*t

print("Area of square is:", x(s))
print("Area of rectangle", y(l,b))
print("Area of triangle", g(h,d,t))
```

Output

Enter the length of a side of a square  
Enter your value : 2

Enter the length and breadth of rectangle.  
Enter your value : 4  
Enter your value : 2

Enter the base and height of triangle

Enter your value : 3  
Enter your value : 2  
Area of Square : 4  
Area of Rectangle : 8  
Area of Triangle : 3.0

## Program No. 27

Aim: Python program to create a class Rectangle with private attributes length and width. Overload 'c' operator to compare the area of rectangle.

class Rectangle:

def \_\_init\_\_(self, l, w):

self.length = l

self.width = w

self.area = self.width \* self.length

def \_\_lt\_\_(self, other):

if self.area < other.area:

print("Rectangle with length =", self.width  
= ', self.length, ' has the lesser area!')

elif other.area < self.area:

print("Rectangle with length =", other.length  
and width = ', other.width, ' has the lesser area!')

else:

print("They have equal area!")

def float(input('Enter length of 1st rectangle')):

wa float(input('Enter width of 1st rectangle:'))

$R_1 = \text{Rectangle}(l, w)$

$l = \text{float}(\text{input('Enter the length of 1st rectangle')})$

$w = \text{float}(\text{input('Enter width of 2nd rectangle')})$

$R_2 = \text{Rectangle}(l, w)$

$K_1 < R_2$

Result:

The program has been executed and  
the output was verified

## Output

Enter length of 1st Rectangle : 4

Enter width of 1st Rectangle : 6

Enter length of 2nd Rectangle : 3

Enter width of 2nd Rectangle : 6

Rectangle width length and width 6 has  
larger area

## Program No: 2

**Ques:** Create a class time with private attribute hours, minute and seconds, 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.__minute + other.__minute) / 60)
hh = (self.__hour + other.__hour) + second // 60
mm = (self.__minute + other.__minute) % 60
ss = (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)

T<sub>1</sub> = Time(12, 25, 45)

T<sub>2</sub> = Time(11, 45, 56)

T<sub>1</sub> + T<sub>2</sub>

Output

Time [hh:mm:ss]

## Program No.: 24

**Aim:** Create a class publisher (name). Define class book from publisher with attributes date and author. Define class python from Books with attributes price and no of pages. Write a program to print displaying information about a python book. Use base class constructor invocation and method overriding.

class publisher:

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

```
self.name = name
```

```
def show(self):
```

pass

class Book(publisher):

```
def __init__(self, title, author, price):
```

```
self.title = title
```

```
self.author = author
```

```
publisher.__init__(self, name)
```

```
def show(self):
```

pass

class Python(Book):

def \_\_init\_\_(self, q, no, title, author),  
name):

self. price = p

self. no. of pages = no

Book. \_\_init\_\_(self, title, author), name)

def show(self):

print('Book title:', self. title)

print('Author:', self. author)

print('Publisher:', self. price)

print('No. of pages', self. no. of pages)

P. = Python(999.99, 400, 'programming with  
Python', 'George Elliot', 'ABC Books')

P. show()

Result:

The programs has been executed  
and the output was verified.

Output

Book title : Rich Dad Poor Dad

Author : Charles Dickens

Publisher : CBD Books

Price : 1000

No. of pages : 500

## Program No: 30

Aim: Python program to read a file line by line and store it into a list.

```
def file_read(filename):
    with open(filename) as f:
```

```
        # content list is the list that
        contains the new lines
```

```
        c = f.readline()
```

```
        print(c)
```

```
        # print(len(c))
```

```
f = file_read("demo.txt")
```

Result:

The programs has been executed  
and the output was verified

Output

Q' A trailer is a vehicle designed for carrying bulk material, often on building sites. They are distinguished from dump trucks by configuration: a dump

## Program No: 31

Aim: Python program to copy odd lines  
of one file to other.

```
a = open ('demo.txt', 'r')
b = open ('x.txt', 'w')
c = a.readlines()
```

```
for i in range (0, len(c)):
    if (i % 2) == 0:
        b.write (c[i])
```

else:

```
pass
b.close
b = open ('x.txt', 'r')
d = b.read ()
print (d)
b.close ()
```

Result:

The program has been executed  
and output was verified.

## Output

They are distinguished from dump trucks by configuration: a dumper is usually an open top - should vehicle with the side tipping in front of driver.

Program 32:

Ans: Program to read each word from  
 a given CSV file and print a list  
 of strings.

```
import csv
with open('dep.csv', newline='') as
  csvfile:
    d = csv.reader(csvfile, delimiter=',',
                  quotechar='\'')
    for r in d:
      print(r[1].join(r)))
```

Result:

The program has been executed and  
 the output was verified.

Output

"[11, 12, 13]", "[22, 29, 56]", "[99, 30, 90]",  
"[21, 40, 55]", "[81, 29, 55]", "[10, 10, 40]",  
"[41, 112, 13]", "[33, 29, 56]", "[33, 30, 30]",  
"[21, 40, 55]", "[81, 29, 55]", "[10, 10, 40]

## Program No: 33

Aim : Python program to accept a list of words and return the length of longest word.

def find(word):

wl = [ ]

for n in word

wl.append(len(n), n)

wl.sort()

result = wl[-1][0], wl[-1][1]

print ("longest word: " result[1])

print ("length of longest word: " result[0])

find ("hello", "morning", "hi")

Output

longest word: morning  
length of the longest word: 7

## Program No: 34

Aim: Python program to create a list of colors from comma separated color names enter by user. Display first and last color.

(down) -> (input ("Enter colors separated by commas:"))  
 . . . . . split (',')

print ('First color: ', color[0])

print ('Last color: ', color [len (color) - 1])

## Program No: 35

Aim: Python Program to print out all colors from color list not contained in color list 2.

(down) -> (input ("Enter colors separated by commas"))  
 . . . . . split (',')

color 2 = (input ("Enter colors separated by commas"))  
 . . . . . split (',')

print 'Colors in color list not contained in

Output

Enter colors separated by commas: red, black, yellow  
Red color: red.  
List color: yellow

Output

Enter colors separated by commas: red, yellow, brown  
Enter color separated by commas: black, white

What are colors left? not contained in either list  
(brown, 'red', 'yellow')

color list 2 one, find color difference (color)

Program No: 34

Aim: Python Program to create a package graphics with modules rectangle and circle and subpackage 3D graphics with modules cuboids and sphere. include methods to find area and perimeter of respective figures in each module. Write programs to find area and perimeter of figures by different importing statements.

Circle.py

def area(r):

print ('Area of circle with radius', 'r', 'is',  
 $\pi \cdot r^2$  % (3.14 \* r \* r), 'square')

def circumference(r):

print ('Circumference of a circle with radius',  
r, 'is', ' $2\pi r$ ' % (3.14 \* 2 \* r), 'units')

## cuboid.py

def area(l,b,h)

print('Total Surface area of cuboid with dimensions l, b, h is : ' "% .2f" % (2 \* (l \* b) + (b \* h) + (l \* h)), 'sq. units')

def perimeter(l,b,h)

print('Perimeter of cuboid with dimensions : l, b, h is : ' "% .2f" % (4 \* (l + b + h)), 'units')

## rectangle.py

def area(a,b)

print('Area of rectangle with sides, ' a ', and ' b ' is : ' "% .2f" % (a \* b), 'sq. units')

def perimeter(a,b):

print('Perimeter of rectangle with sides, ' a ' and ' b ' is : ' "% .2f" % (2 \* (a + b)), 'units')

## sphere.py

def area(r):

print('Area of sphere with radius, ' r ' is : ' "% 2f" % (4 \* 3.14 \* r \* r))

$\frac{1}{2} (4 \pi (3.14 \times r^2))$ , 1 sq. units)

def perimeter (r):

print ('perimeter of (largest circle of) sphere with radius, r is : ' + str(2 \* 3.14 \* r), 'units')

Print perimeter. By

tangential circle

from rectangle import \*

from graphics import graphics import cuboid, sphere

a = float (input ('Enter length of rectangle: '))

b = float (input ('Enter breadth of rectangle: '))

perimeter (a,b)

no float (input ('Enter the radius of the circle: '))

circle. circumference (r)

l = float (input ('Enter length of cuboid: '))

b = float (input ('Enter breadth of cuboid: '))

h = float (input ('Enter height of cuboid: '))

cuboid. perimeter (l, b, h)

no float (input ('Enter radius of the sphere: '))

sphere. perimeter (r)

Find area, by

import circle

from rectangle import \*

from Graphics\_3D import cube, sphere  
a = float(input("Enter the length of the rectangle:"))  
b = float(input("Enter the breadth of rectangle:"))  
area(a,b)

area = float(input("Enter the radius of circle:"))  
circle.area(area)

a = float(input("Enter the length of cuboid:"))

b = float(input("Enter the breadth of cuboid:"))

h = float(input("Enter the height of cuboid:"))

cuboid.area(a,b,h)

area = float(input("Enter the radius of sphere:"))  
sphere.area(area)

Output

Enter length of rectangle : 4

Enter breadth of rectangle : 3

Perimeter of a rectangle with side 4.0 and 3.0  
is 14.00 units

Enter the radius of circle : 2

Circumference of a circle with radius 2.0  
is 12.56 units

Enter the length of cuboid : 5

Enter the breadth of cuboid : 4

Enter the height of cuboid : 3

Perimeter of cuboid with dimension 5.0, 4.0,  
4.8.0 units

Enter the radius of sphere : 2

Perimeter of great circle of sphere with  
radius 2.0 is 12.56 units

Enter the length of rectangle : 2

Enter the breadth of rectangle : 3

Area of rectangle with side 2.0 and 3.0 is:  
6.00 sq units.

Enter the radius of circle : 4

Area of circle with radius 4.0 is 50.24  
sq units

Enter the length of cuboid : 4

Enter the breadth of cuboid : 7

Enter the height of cuboid : 2

Total surface area of cuboid with dimension  
4.0, 7.0 and 2.0 is 100.00 sq units

Enter the radius of sphere : 1

Area of sphere with radius 4.0 is 1256.44

## Program No. 35

**Ques:** Write a Python program to create a Bank Account with account number, name, type of account and balance. Write constructor and methods to deposit and withdraw an amount from bank.

class Bank Account:

```
def __init__(self, a, n, t, b):
    self. accno = a
    self. name = n
    self. type = t
    self. balance = b
```

def deposit(self, a):

    self. balance += a

print('Rs.', a, 'deposited', 'Current Balance is',

'Rs.', self. balance)

def withdraw(self, a):

    if self. balance >= a:

self.balance = 0

print ('Rs.', a, 'withdrawn, current balance  
is Rs.', self.balance)

else:

print ('Insufficient balance to make this  
transaction!')

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

n = input ("Enter name of account holder: ")

t = input ("Enter account type: ")

b = float(input ("Enter your balance: "))

ac = BankAccount(a, n, t, b)

ac.deposit (float(input ("Enter amount to  
deposit: ")))

ac.withdraw (float(input ("Enter amount to  
withdraw: ")))

Output

Banker Account Number: 34255740453

Banker Name of Account Holder: ATM

Banker account type: Savings

Banker current balance: 100000

Banker amount to deposit: 30,000

Rs 30000 deposit / current balance is 100000

Banker amount to withdraw: 5000

Rs 5000 withdrawal (current balance is

Rs 100000)

## Program No: 36

Aim: Python program to display future leap years from current year to final year entered by the user.

import datetime

a = datetime.datetime.now()

a = int(a.year)

b = int(input("Enter the final year:"))  
print("leap years: ")

for i in range(a, b+1):  
 if (i % 4 == 0):  
 print(i)

Output

Border final year : 2040  
leap years

2024

2028

2032

2036

2040

## Program No: 34

Aim: Python program to write a python dictionary to csv file. After writing the csv file read the csv file and display the contents.

```
import csv
```

```
file_name = ["best_book_id", "author", "original_title"]
book = [{"best_book_id": 1, "authors": "Suzanne Collins", "original_title": "The Hunger Games"}, {"best_book_id": 2, "authors": "J.K. Rowling, Mary Grand Pr\u00e9", "original_title": "Harry Potter and the Philosophers Stone"}, {"best_book_id": 3, "authors": "Stephenie Meyer", "original_title": "Twilight"}]
```

with open('d.csv', 'w') as file:  
writer = csv.DictWriter(file, fieldnames=

fieldnames)

writer.writeheader()

writer.writerow(book)

with open('d.csv', newline='') as file:

STORY  
BY SHRI

TRIVENI

PAGE NO.

DATE / /

d = CSV.reader(csvfile, delimiter=',')  
for row  
print (''.join(r))

Subject

1. Suzanne Collins, The Hunger Games
2. J.K. Rowling, Harry Potter and the Philosopher's Stone
3. Stephenie Meyer, Twilight