## Experiment No: 1

# DISPLAY FUTURE LEAP YEAS FROM THE CURRENT YEAR TO FINAL YEAR

### AIM

To write a Python program to display future leap years from the current year to final year

### ALGORITHM

1. Read the year using input function
2. Convert this value into integers
3. Store this value in a variable.
4. Follow the steps to check whether the year is leap year or not
    a. If a year is evenly divisible by 4 means having no remainder then go to next step. If it is not divisible by 4. It is not a leap year.
    b. If a year is divisible by 4, but not by 100. For example: 2012, it is a leap year. If a year is divisible by both 4 and 100, go to next step.
5. If a year is divisible by 100, but not by 400. For example: 1900, then it is not a leap year. If a year is divisible by both, then it is a leap year.
6. Print the list of leap years till the final year entered by the user.

### PROGRAM

```
s=int(input("enter start year:"))
e=int(input("enter end year:"))
if(s<e):
    print("leap years are:",end="")
    for i in range(s,e):
        if i%4==0 and i%100!=0:
            print(i,end=" ")
else:
    print("invalid")
```

**OUTPUT**

```
C:\Users\Thejas\PycharmProjects\pythonProject2\S1MCA\Tejas\Scripts\python.exe
enter start year:2022
enter end year:2050
leap years are:2024 2028 2032 2036 2040 2044 2048
Process finished with exit code 0
```

**RESULT**

The program to display future leap yeas from the current year to final year has executed and verifiedsuccessfully.

## Experiment No: 2

# PROMPT THE USER FOR A LIST OF INTEGERS

### AIM

To write a Python program to prompt the user for a list of integers. For all values greater than 100, store ‘over’ instead.

### ALGORITHM

1. Read a list of numbers from user.
2. Convert this value into integers.
3. Store this value in a variable.
4. If the value is less than 100, then store the value
5. Else store over in the list

### PROGRAM

```
lit = []
n = int(input("Enter the limit of list :"))
for i in range(0, n):
    print("Enter the number ", i + 1, ":")
    num = int(input())
    if (num < 100):
        lit.append(num)
    else:
        lit.append("over")
print(lit)
```

**OUTPUT**

```
C:\Users\Thejas\PycharmProjects\pythonProject2\S1MCA\Tejas\Scripts\python.exe
Enter the limit of list :4
Enter the number  1 :
1
Enter the number  2 :
4
Enter the number  3 :
723
Enter the number  4 :
5
[1, 4, 'over', 5]

Process finished with exit code 0
```

**RESULT**

The program to prompt the user for a list of integers has executed and verified successfully

## Experiment No: 3

**AIM**

# AREA OF CIRCLE

To write a Python program to accept a radius from user and find the area of the circle by importing math package to assign pi value.

**ALGORITHM**

1. Import the math package
2. Read value of input from user
3. Convert the value into float
4. Find the area using the math module and radius
5. Print the area of circle

**PROGRAM**

```
import math
r=float(input(“Enter the radius of the circle:”))
area=math.pi*(r**2)
print (“Area of a circle is : %.2f " %area)
```

**OUTPUT**

```
C:\Users\Thejas\PycharmProjects\pythonProject2\S1MCA\Tejas\Scripts\python.exe
enter radius5
area of circle is 78.54

Process finished with exit code 0
```

**RESULT**

The program to find area of a circle has executed and verified successfully.

## Experiment No: 4

**AIM**

## BIGGEST OF THREE NUMBERS

To write a Python program to accept three numbers from user and find the biggest number.

**ALGORITHM**

1. Read three numbers from user.
2. Convert these values into float.
3. Store these value in a three variable num1, num2,num3.
4. If num1 is greater than num2 and num3 then print num1 is largest.
5. If num2 is greater than num1 and num3 then print num2 is largest.
6. Else print num3 is largest.

**PROGRAM**

```
num1 = float(input("Enter the first number: "))
num2 = float(input("Enter the second number: "))
num3 = float(input("Enter the third number: "))
if (num1 > num2) and (num1 > num3):
    largest = num1
elif (num2 > num1) and (num2 > num3):
    largest = num2
else:
    largest = num3
print("The largest number is",largest)
```

**OUTPUT**

```
C:\Users\Thejas\PycharmProjects\pythonProject2\S1MCA\Tejas\Scripts\python.exe
enter first number5
enter second number9
enter third number6
largest number 9

Process finished with exit code 0
```

**RESULT**

The program to find biggest among three give number has executed and verified successfully.

## Experiment No: 5

# FACTORIAL OF A NUMBER

### AIM

To write a Python program to accept a number from user and find the factorial of the number

### ALGORITHM

1. Read an input from user
2. Convert it into integer
3. Initialize a variable to one
4. If input is less than zero then print an error message
5. If input is equal to zero then display 1 as output
6. Else calculate factorial and print the result
7. Exit.

### PROGRAM

```
num = int(input("Enter the number to find factorial :"))
factorial = 1
if (num < 0):
        print("Sorry, 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 the give number ",num,"is ",factorial
```

**OUTPUT**

```
C:\Users\Thejas\PycharmProjects\pythonProject2\S1MCA\Tejas\Scripts\python.exe
enter the number to find factorial7
the factorial of the given number 7 is 5040

Process finished with exit code 0
```

**RESULT**

The program to find the factorial of a number has executed and verified successfully.

## Experiment No: 6

# GENERATE FIBONACCI SERIES OF N TERMS

**AIM**

To write a Python program to generate the Fibonacci series of n terms.

**ALGORITHM**

1. Input the 'n' value until which the Fibonacci series has to be generated
2. Initialize sum = 0, f1 = 0, f2 = 1 and count = 1
3. while (count <= n)
4. Print sum
5. Increment the count variable
6. Swap f1 and f2
7. while (count > n)
8. Else
9. Repeat from steps 4 to 7

**PROGRAM**

```
n = int(input("Enter the number of n terms : "))
f1 = 0
f2 = 1
sum = 0
count = 1
print("Fibonacci Series of n terms: ", end = " ")
while(count <= n):
    print(sum, end = " ")
    count += 1
    f1 = f2
    f2 = sum
    sum = f1 + f2
```

**OUTPUT**

```
C:\Users\Thejas\PycharmProjects\pythonProject2\S1MCA\Tejas\Scripts\python.exe
enter the number of terms15
fibanocci series of 15 terms 0 1 1 2 3 5 8 13 21 34 55 89 144 233 377
Process finished with exit code 0
```

**RESULT**

The program to generate Fibonacci series of n terms has executed and verified successfully.

## Experiment No: 7

# COUNT THE NUMBER OF VOWELS IN A STRING

### AIM

To write a Python program to accepted a string from user and count the number of vowels in the string.

### ALGORITHM

1. Read the string from user
2. Assign a variable count=0
3. Using a for loop traverse the string
4. if the character in contain ‘aeiouAEIOU’ the increment the count value by 1
5. print the count

### PROGRAM

```
str=raw_input(“Enter the string :”)
count=0
for i in str:
    if i in ‘aeiouAEIOU’:
        count +=1
print “The number of vowels in”, str, “is ” count
```

### OUTPUT

```
C:\Users\Thejas\PycharmProjects\pythonProject2\S1MCA\Tejas\Scripts\python.exe
enter stringPython Programming
number of vowels is 4

Process finished with exit code 0
```

### RESULT

The program to count the number of vowels in a given string has executed and verified successfully.

## Experiment No: 8

# TO PRINT THE PREFECT SQUARES

### AIM

To write a Python program to print the prefect squares upto the limit accepted from user

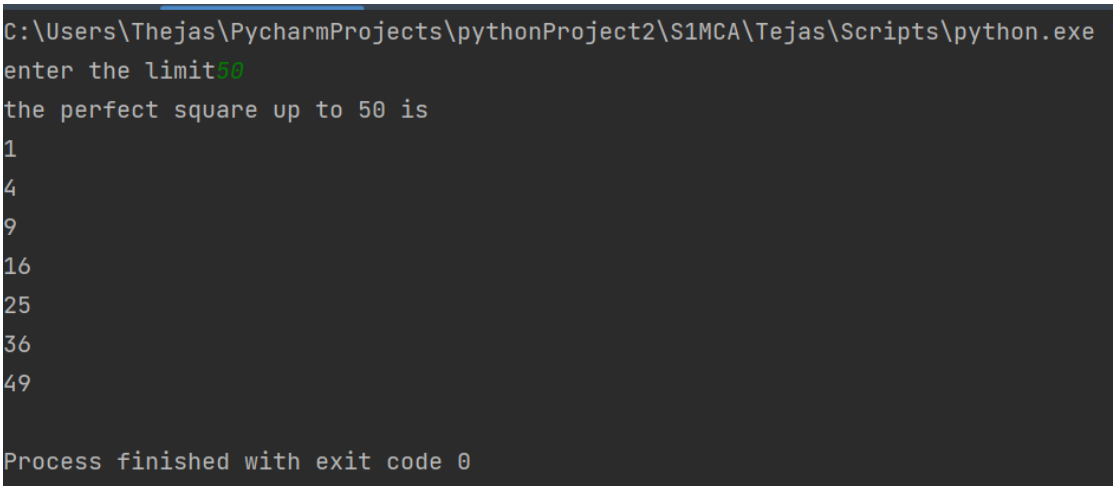
### ALGORITHM

1. Read the limit n
2. Assign B=1 and i=1
3. Use for loop to iterate I from B to n
4. Compute P= B*B
5. Increment B=B+1
6. check (P<N) if true go to step 7 otherwise step8
6. Print P
7. Increment the value of i by 1 and go to step 4 until (i< limit )
8. Exit

### PROGRAM

```
n = raw_input("Enter the limit :")
n=int(n)
B,i=1,1
print "The prefect squares upto", n, "is :"
for i in range (B,n):
        P=B*B
        B=B+1
        if (P<N):
                print P
```

## OUTPUT




## RESULT

The program to print prefect squares has executed and verified successfull

## Experiment No: 9

### GENERATE ALL FACTORS OF A NUMBER

**AIM**

To write a Python program to accept a number from user and generate all factors of that number.

**ALGORITHM**

1. Read an input from user and pass the value to the function
2. Use a for loop to iterate i from 1 to given number
3. check if number % i = = 0 if true print the i else increment the loop value by 1 until i<given number+1
4. Exit

**PROGRAM**

```
def print_factors(x):
        print("The factors of ",x,"are:")
        for i in range(1, x + 1):
                if (x % i == 0):
                        print(i)
num = raw_input(“Enter the number to find the factors :”)
print_factors(num)
```

**OUTPUT**

```
C:\Users\Thejas\PycharmProjects\pythonProject2\S1MCA\Tejas\Scripts\python.exe
enter the number to find factor20
the factors of 20 are
1
2
4
5
10
20

Process finished with exit code 0
```

**RESULT**

The program to find factorial of a given number has executed and verified successfully.

## Experiment No: 10

**AIM**

**INTEGER OPERATIONS**

Write a python program to accept an integer n and compute n+nn+nnn.

**ALGORITHM**

1. Read a values from user.
2. Convert it into integer
3. Convert to integer and store it in n1
4. Concatenate the string twice, convert into integer and store it in n2
5. Concatenate the string thrice, convert into integer and store it in n3
6. Add n1,n2 and n3 and print the result

**PROGRAM**

```
a = int(input("Input an integer : "))
n1 = int( "%s" % a )
n2 = int( "%s%s" % (a,a) )
n3 = int( "%s%s%s" % (a,a,a) )
print “n=”, n1, \n, “nn= ”, n2, \n,“ nnn= ”, n3
print “sum (n+nn+nnn)=”, (n1+n2+n3)
```

**OUTPUT**

```
C:\Users\Thejas\PycharmProjects\pythonProject2\S1MCA\Tejas\Scripts\python.exe
input an integer5
n= 5
 nn= 55
 nnn= 555
sum(n+nn+nnn)= 615

Process finished with exit code 0
```

**RESULT**

The program to perform integer operation has executed and verified successfully.

## Experiment No: 11

### LIST OPERATIONS

### AIM

Write a python program to Enter 2 lists of integers. Check

(a) Whether list are of samelength

(b) whether list sums to same value

(c) whether any value occur in both.

### ALGORITHM

1. Create two integer list and pass the values to each functions
2. Check whether two list contain common elements using for loop
3. Check whether the list sums to same value
4. Check whether list are of same length using sizeof operator.
5. Print the result

### PROGRAM

```
lst=[1,3,5,7,9,11,34]
lst1=[5,13,45,7,20,65,1]
s=int(0)
c=int(0)
for i in lst and lst1:
    if len(lst)==len(lst1):
        print("Lists are of same length")
        break
    else:
        print("Lists have different length")break
  for i in range(0,len(lst) and len(lst1)):
    s=s+lst[i]
    c=c+lst1[i]
  for i in range(0, len(lst) and len(lst1)):if
```

```
s==c:
    print("Sum of values are same")
    break
```

**OUTPUT**

```
C:\Users\Thejas\PycharmProjects\pythonProject2\S1MCA\Tejas\Scripts\python.exe
list are of same length
sum of values are diffrent
elements that matched are:
[1, 5, 7]

Process finished with exit code 0
```

**RESULT**

The program to perform list operation has executed and verified successfully.

## Experiment No: 12

### STRING DATA TYPE OPERATION

**AIM**

Write a python program to get a string from an input string where all occurrences of first character replaced with '$', except first character.

**ALGORITHM**

1. Get a string value and pass the value to the function
2. Store the first character of string to a character variable
3. Replace the occurrence of first character in the string with '$'
4. Exit

**PROGRAM**

```
def change_char(str1):
        char = str1[0]
        str1 = str1.replace(char, '$')
        str1 = char + str1[1:]
        return str1
str= input("Enter the string:")
print "The original string :",str
print ("The replaced string :",change_char(str))
```

**OUTPUT**

```
C:\Users\Thejas\PycharmProjects\pythonProject2\S1MCA\Tejas\Scripts\python.exe
Enter the string:blub
the original string: blub
the replaced string: blu$

Process finished with exit code 0
```

**RESULT**

The program to get a string from an input string where all occurrences of first character replaced with '$', except first character has executed and verified successfully.

## Experiment No: 13

### SWAPING FIRST AND LAST CHARACTER IN A STRING

**AIM**

Write a python program to create a string from given string where first and last characters

**ALGORITHM**

1. Get a string value and pass the string to function
2. Apply slice operator ([] and[:]) where the index starting from 0 and ending at -1 and store value str2 and return the value
3. Exit

**PROGRAM**

```
def character_exchange(str1):
        str2=(str1[-1:] + str1[1: = -1] + str1[:1])
        return str2
str= input("Enter the string:")
print "The original string :",str
print ("String after swapping first and last character :",character_exchange(str))
```

**OUTPUT**

```
C:\Users\Thejas\PycharmProjects\pythonProject2\S1MCA\Tejas\Scripts\python.exe
enter the string:python
the original string: python
string after swapping first and last character: nythop

Process finished with exit code 0
```

**RESULT**

The program to create a string from given string where first and last characters exchanged hasexecuted and verified successfully

## Experiment No: 14

### TO PRINT FILENAME EXTENSION

### AIM

To write a Python program to print the filename extension.

### ALGORITHM

1. Read the filename from user
2. Split the filename on the occurrence of  dot
3. Store the extension in f_extns
4. Print the extension
5. Exit

### PROGRAM

```
filename = input("Input the Filename: ")
f_extns = filename.split(".")
print ("The extension of the file is : " + repr(f_extns[-1]))
```

### OUTPUT

```
C:\Users\Thejas\PycharmProjects\pythonProject2\S1MCA\Tejas\Scripts\python.exe
input the file name:PythonProgram.py
the extension of the file is:'py'

Process finished with exit code 0
```

### RESULT

The program to print the filename extension has executed and verified successfully.

## Experiment No: 15

### CHECK A STRING IS PALINDROME OR NOT

**AIM**

To write a Python program to read a string and check whether the string is palindrome or not.

**ALGORITHM**

1. Read the string from user
2. Declare a variable with the length of the string.
3. Declare a for loop, using half of the length of the string as a reference point.
4. Check if each letter is the same as its mirror equivalent — or, a character on the other side

**PROGRAM**

```
str = raw_input("Enter the string to check :")

i=0

s=0

k= len(str)

for j in range (k-1,-1,-1):


if (str[i]!= str[j]):
 s=1
 break
i=i+1
if(s==0):
        print " The given string", str, " is palindrome"

    else:

        print " The given string", str, " is not palindrome"
```

**OUTPUT**

```
C:\Users\Thejas\PycharmProjects\pythonProject2\S1MCA\Tejas\Scripts\python.exe
enter the string to check:CIVIC
the given string CIVIC is palindrome

Process finished with exit code 0
```

**RESULT**

The program has executed and verified successfully

**Experiment No: 16**

.

## PATTERN PRINTING

**AIM**

To write a Python program to construct the following pattern using nested for loop

```
#
# #
# # #
# # # #
# # # # #
# # # #
# # #
# #
#
```

**ALGORITHM**

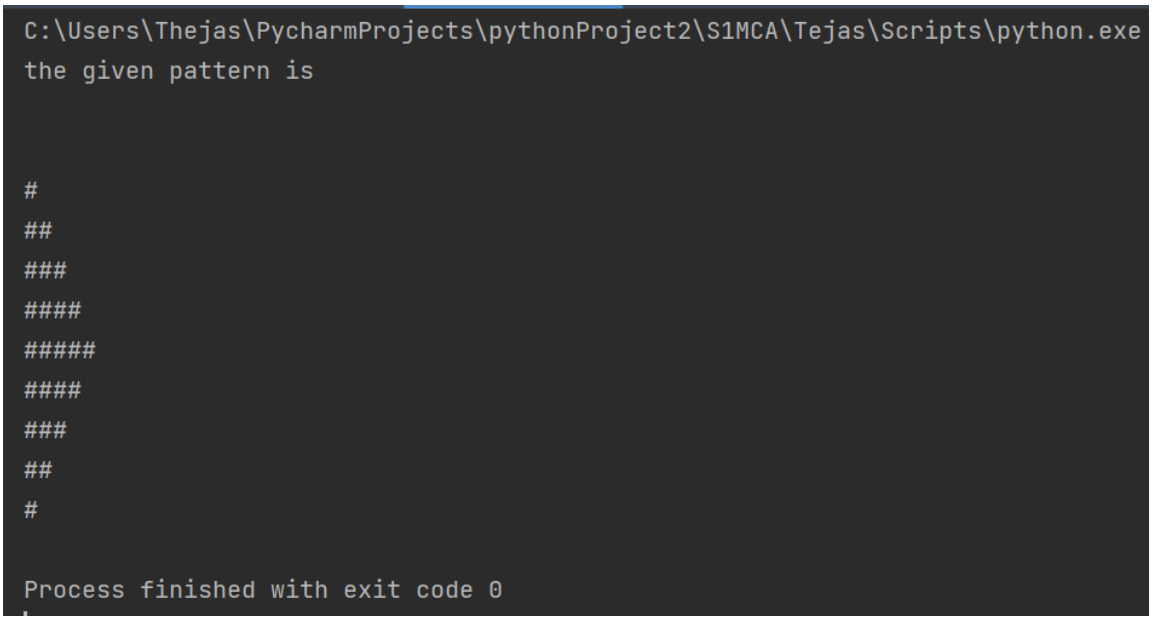
1. Initialize the value of N that decides the size of pattern.
2. Iterate the number of rows using outer for loop
3. Next, the inner loop to handle the number of columns. Inner loop iteration depends on the values of the outer loop.
4. Print the pattern using print() function.
5. Add a new line after each row, i.e. after each iteration of outer for loop so you can display the pattern appropriately.

**PROGRAM**

```
n= 5
print " The given pattern is", \n
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 " "
```

**OUTPUT**




**RESULT**

The program to print the given pattern has executed and verified successfully.

## Experiment No: 17

# PRINT A GIVEN PYRAMID OF NUMBERS

### AIM

Write a python program to print a given pyramid of numbers.

### ALGORITHM

1. Initialize the step size of n that decides the height of pyramid.
2. Iterate the number of rows using outer for loop where i starts from 1 and ends at i<n+1
3. Assign k=i
4. Next, the inner loop to handle the number of columns. Inner loop iteration depends on the values of the outer loop.
5. Print the value of k using print() function.
6. Add a new line after each row, i.e. after each iteration of outer for loop so you can display the pyramid.

### PROGRAM

```
n=int(input("Enter the step size:"))
print " The pyramid for the given size ",n, "is:" \n
for i in range(1,n+1):
    k=i
    for j in range (1,i+1):
        print k, end= " "
        k+=i
    print " "
```

**OUTPUT**

```
C:\Users\Thejas\PycharmProjects\pythonProject2\S1MCA\Tejas\Scripts\python.exe
enter the step size6
the pyramid for the given size  6 is:

1
2 4
3 6 9
4 8 12 16
5 10 15 20 25
6 12 18 24 30 36

Process finished with exit code 0
```

**RESULT**

The program to print the pyramid given size has executed and verified successfully.

## Experiment No:18

# CHARACTER SWAPPING

## AIM

Write a python program to create a single string separated with space from two strings by swapping the character at position1.

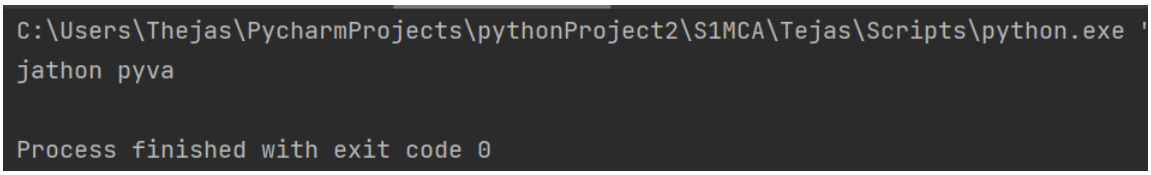
## ALGORITHM

1. Read two characters
2. Swap the first character of two string and store it in new variable.
3. Concatenate new string separated by space
4. Print the resultant string
5. Exit.

## PROGRAM

```
def swap(a,b):
  new_a=b[:2]+a[2:]
  new_b=a[:2]+b[2:]
  return new_a+' '+new_b
print(swap('python','java'))
```

## OUTPUT




## RESULT

The program to create a single string separated with space from two strings by swappingthe character at position 1.has executed and verified successfully

## Experiment No: 19

### GCD OF 2 NUMBERS

AIM

Write a Python program to find gcd of 2 numbers.

**ALGORITHM**

1. Import the math package
2. Read two values from user.
3. Convert it into integer using int function.
4. Calculate and print gcd of input using gcd function.

**PROGRAM**

```
import math
num1= int(input("Enter first number:"))
num2=int(input("Enter second number:"))
print("The gcd of",num1, " and ",num2," is : ")
print(math.gcd(60, 48))
```

**OUTPUT**

```
C:\Users\Thejas\PycharmProjects\pythonProject2\S1MCA\Tejas\Scripts\python.exe
enter 1st number:60
enter 2nd number:48
GCD: 12

Process finished with exit code 0
```

**RESULT**

The program to calculate the gcd of 2 numbers has executed and verified successfully.

## Experiment No: 20

### WORKING WITH GRAPHIC PACKAGES

**AIM**

Write a python program to find area and perimeter of figures by different importing statements.

**ALGORITHHM**

1. Create a package called graphics.
2. Include modules rectangle and circle
3. Include methods to find area and perimeter of rectangle and circle.
4. Create a sub package called three graphics.
5. Include modules cuboid and sphere.
6. Include methods to find area and perimeter of rectangle and circle.
7. Import statements to print the area and perimeter of figures.
8. Exit.

**PROGRAM :**

circle.py

```
print("****CIRCLE****")
import math
def circle_perimeter(radius):
    return math.pi*radius*2
def circle_area(radius):
   return math.pi*radius*radius
def main():
  radius=float(input("Enter value for radius of circle :"))
  area=circle_area(radius)
  perimeter=circle_perimeter(radius)
 area=round(area,4)
  perimeter=round(perimeter,4)
  print("The Area of circle is :",area)
```

```
   print("The perimeter of cicle is :",perimeter)main()
```

**rectanle.py**

```
print("****RECTANGLE****")
def rect_perimeter(l,w):
  return 2*(l+w)
def rect_area(l,w):
  return l*w
def main():
  l=int(input("Enter the length of rectangle:"))
  w=int (input("Enter the width of rectangle:"))
  area=rect_area(l,w)
  perimeter=rect_perimeter(l,w)
  area=round(area,4)
  perimeter=round(perimeter,4)
  print("Area of rectangle :",area)
  print("perimeter of rectangle :",perimeter)
main()
```

**cuboid.py**

```
print("****CUBOID****")
def cuboid_SA(length,width,height):
  return 2*(length*width+height+width*height)
def cuboid_volume(length,width,height):
  return length*width*height
def main()
  length=float(input("please enter the length of a cuboid:"))
  width=float(input("please enter the width of a cuboid:"))
  height=float(input("please enter the height of a cuboid:"))
  area=cuboid_SA(length,width,height)
  perimeter=cuboid_volume(lenth,width,height)
  area= round(area,4)
  perimeter=round(perimeter,4)
  print("The Area of cuboid is :",area)
```

```
  print("The perimeter of cuboid is :",perimeter)
main()
```

**sphere.py**

```
print("****SPHERE****")
import math
def sphere_area(radius):
  return 4*math.pi*radius**2
def sphere_vol(radius):
  return(4/3)*math.pi*radius**3
def main():
  radius=float(input("enter radius of sphere:"))
  area=sphere_area(radius)
  volume=sphere_vol(radius)
  area=round(area,4)
  volume=round(volume,4)
  print("The area of sphere is:",area)
  print("The perimeter of sphere is:",volume)
main()
```

**main.py**

```
import graphics.cicle
import graphics.rectangle
from graphics.threedgra.cuboid import*
from graphics.threedgra.sphere import*
```

**OUTPUT**

```
C:\Users\Thejas\PycharmProjects\pythonProject2\S1MCA\Tejas\Scripts\python.exe
****CIRCLE****
enter the radius of the circle2
the area of the circle is 12.5664
the perimeter of the circle is 12.5664
****RECTANGLE****
enter length of the rectangle:3
enter the width of the rectangle:2
area of rectangle is: 6
perimeter of rectangle is 10
****CUBOID****
enter the length2
enter the width3
enter the height4
the area of the cuboid is 44.0
the volume of the cuboid is 24.0
****SPHERE****
enter the radius of the sphere2
area of the sphere is : 50.2655
volume of the sphere is: 33.5103

Process finished with exit code 0
```

**RESULT**

The python program to find area and perimeter of figures by different importing statements.

**Experiment No: 21**

## COMPARE TWO RECTANGLE OBJECTS BY THEIR AREA

### AIM

Create Rectangle class with attributes length and breadth and methods to find area and perimeter. Compare two rectangle objects by their area.

### ALGORITHM

1. Create a class rectangle with attributes length and breadth to find area and perimeter of a rectangle.
2. Create three function inside class Rectangle.
    i. def__int__() for inputing length & breadth.
    ii. defarea() to find area of the rectangle.
    iii. def perimeter() to find perimeter of a rectangle
3. Display the result of area of both rectangles and find it whether it is matching or noy.
4. Display the result.

### PROGRAM

```
class rect:
  def init (self,l,b): self.length=l
       self.width=b
  def findArea(self): b=(self.length * self.width)
    return b
  def findPerimeter(self):
    c=2*(self.length+self.width) return c
a1=rect(10,3) a2=rect(1,3)
print(a1.findArea())
print(a1.findPerimeter())
 if a1.findArea()>a2.findArea():print("a1 is greater")
else:
  print("a2 is greater")
```

**OUTPUT**

```
C:\Users\Thejas\PycharmProjects\pythonProject2\S1MCA\Tejas\Scripts\python.exe
30
26
a1 is greater

Process finished with exit code 0
```

**RESULT**

The program to find area of rectangle and perimeter of rectangle and compare the area of rectangle is executed and verified successful.

## Experiment No: 22

### BANK ACCOUNT TRANSACTION USING CLASS

**AIM**

To create a bank account account with members account number, name, type of account and balance. Write constructor and methods to deposit at the bank and withdraw an amount from the bank.

**ALGORITHM**

1. Create a class Bank_Account
2. Define a function _ _init _ _( )
3. Define a function deposit( ) for deposit functions
4. Define a function withdraw ( ) for withdrawing money

**5**.Define a function display ( ) to display the balance amount after transaction

**PROGRAM**

```
Class bank_account:
   def getData (self,name,accno,acctype,balance):
      self.name=name
      self.accno=accno
      self.acctype=acctype
      self.balance=balance
   def displayCustomer (self):
      print("customer Name:", self.name)
      print("Account Number:",self.accno)
      print("Account Type:", self.acctype)
      print("Balance amount :", self.balance)
   def deposit(self,amount):
      self.balance=self.balance+amount
   def withdrawal(self,amount):
      if self.balance-amount<0:
         print("insufficent funds")
      else:
         self.balance=self.balance-amount


print("Hello Welcome to BANKING SYSTEM")
ch=0
obj=bank_account()
```

```
        while ch!=5:
            print("Select your option")
            print("1.New customer")
            print("2.Deposit")
            print("3.Withdrawal")
            print("4.Display")
            print("5.Exit")
            ch=int(input("Enter your choice:"))
            if ch==1:
                obj=bank_account()
                n=input("Enter name:")
                no=int(input("Enter account number:"))
                t=input("Enter Account Type(SB/C):")
                b=int(input("Enter the Amount:"))
                obj.getData(n,no,t,b)
            if ch==2:
                b=int(input("Enter the amount to be deposited :"))
                obj.deposit(b)
            if ch==3:
                b=int(input("Enter the amount to be withdrawns:"))
                obj.withdrawal(b)
            if ch==4:
                obj.displayCustomer()
else:print("program terminated !!!")
```

## OUTPUT

```
C:\Users\Thejas\PycharmProjects\pythonProject2\S1MCA\Tejas\Scripts\python.exe
Hello Welcome To Banking System
select your option
1,New customer
2,Deposit
3,withdraw
4,Display
5,Exit
enter your choice:1
enter nameThejas
enter account number:20010200102
enter account type(SB/C):SB
enter balance500000
program terminated!!!
select your option
1,New customer
2,Deposit
3,withdraw
4,Display
5,Exit
enter your choice:3
enter amount to withdraw:10000
program terminated!!!
```

```
program terminated!!!
select your option
1,New customer
2,Deposit
3,withdraw
4,Display
5,Exit
enter your choice:4
customer name: Thejas
account number: 20010200102
account type: SB
balance: 490000
select your option
1,New customer
2,Deposit
3,withdraw
4,Display
5,Exit
enter your choice:
```

## RESULT

The program to create bank account and display the transaction has executed and verifiedsuccessfully

## Experiment No: 23

### DISPLAY INFORMATION ABOUT A PYTHON BOOK

**AIM**

To create a class publisher (name).derive class book from publisher with attribute title and author.Derive class python from book with attribute price and no: of pages. Display information about a python book .

**ALGORITHM**

1. Create a class publisher
2. Define a function _ _init _ _( )
3. Create a derived class book
4. Create a derived class python

**5**.Define a function display ( ) to display the details of python book

## PROGRAM

```
class publisher:
  def__init__(self,n):
    self.name=n
class book(publisher):
  def__init__(self,n,t,a):
    super(). init (n)
    self.title=t
    self.author=a
class python(book):
  def__init__(self,n,t,a,p,pgs):
    super(). init (n,t,a)
    self.price=p
    self.pages=pgs
  def display(self):
    print(p1.name)
     print(p1.title)
    print(p1.author)
    print(p1.price)
    print(p1.pages)
p1 = python("python", "introduction to python", "jeeva jose", 450, 300)
p1.display()
```

## OUTPUT

```
C:\Users\Thejas\PycharmProjects\pythonProject2\S1MCA\Tejas\Scripts\python.exe
python
introduction to python
jeeva jose
450
300

Process finished with exit code 0
```

## RESULT

The program to class publisher and display information about a python book has executed and verifiedsuccessfully

## Experiment No: 24

### READ FILE LINE BY LINE AND STORE IN LIST

**AIM**

To write a Python program to read a file line by line and store it in a list

**ALGORITHM**

1. Create a function named “read_file(file_name)”
2. Create a text file named file1.txt for writing the content
3. Read the file file1.txt and store it in the list and display the output.

**PROGRAM**

```
fn=open("newfile1.txt",'r')
s=fn.readline()
l=s.split()
print(l)
```

**INPUT TEXT FILE;newfile1.txt**

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance.

## OUTPUT

```
C:\Users\Thejas\PycharmProjects\pythonProject2\S1MCA\Tejas\Scripts\python.exe C:\Users\Thejas\PycharmProjects\pythonProject2\S1MCA\Tejas\PYTHON-
['Python', 'is', 'an', 'interpreted,', 'object-oriented,', 'high-level', 'programming', 'language', 'with', 'dynamic', 'semantics.', "Python's",

Process finished with exit code 0
```

```
1MCA\Tejas\PYTHON-LAB-main\python\pg25.py
ics.', "Python's", 'simple,', 'easy', 'to', 'learn', 'syntax', 'emphasizes', 'readability', 'and', 'therefore', 'reduces', 'the', 'cost', 'of',
```

```
ython\pg25.py
 , 'easy', 'to', 'learn', 'syntax', 'emphasizes', 'readability', 'and', 'therefore', 'reduces', 'the', 'cost', 'of', 'program', 'maintenance.']
```

## RESULT

The program to print the filename extension has executed and verified successfully

**Experiment No:25**

## COPY ODD LINES OF ONE FILE TO ANOTHER

### AIM

To write a python program to copy odd lines of one text file to another text file

### ALGORITHM

1) Create a file named file1.txt and file2.txt
2) Write some data line by line in file1.txt file
3) file2.txt file is used to store odd lines from file1.txt file
4) After copying the odd lines to file2.txt file display the content of file2.txt file

### PROGRAM

```
file1 = open('file1.txt', 'r')
file2 = open('file2.txt', 'w')
lines = file1.readlines() type(lines)
for i in range(0, len(lines)):
if(i % 2 == 0):
file2.write(lines[i]) file1.close()
file2.close()
file1 = open('file1.txt', 'r')
file2 = open('file2.txt', 'r')
str1 = file1.read()
str2 = file2.read()
 print("file1 content...")
print(str1)
print()
print("file2 content...")
print(str2)
file1.close()
 file2.close()
```

**OUTPUT:**

```
C:\Users\Thejas\PycharmProjects\pythonProject2\S1MCA\Tejas\Scripts\python.exe
file1 content...
aa
bb
cc
dd

file2 content...

aa
cc



Process finished with exit code 0
```

**RESULT**

The python program to copy odd lines of one text file to another text file has executed and verified successfully.