

Ex.No: 1.a	Numeric and String literals
-------------------	------------------------------------

Aim:

To write a python script to illustrate numeric and string literals.

Algorithm:

1. Start the program
2. Get String Literals in the variables str1,str2 and str3 within single, double and triple quotes of strings
3. Get Numeric literals in the variables a=5 and b=10.5, calculate c=a+b and print c
4. Get complex literal in the variable d=10+5j
5. d.real and d.imag to get the real part and imaginary part of complex number

Program:

```
#string Literal
```

```
str1="This is first python Program"
```

```
str2='Numeric and String Literals'
```

```
str3=""" Addition of
```

```
Two Numbers"""
```

```
print(str1)
```

```
print(str2)
```

```
print(str3)
```

```
#Numeric Literal
```

```
a=5
```

```
b=10.5
```

```
c=a+b
```

```
print("Addition is :",c)
```

```
#Complex Literal  
d=10+5j  
print("Complex Literal is :",d)  
print("Real part is :",d.real)  
print("Imaginary Part is :",d.imag)
```

Output:

This is first python Program

Numeric and String Literals

Addition of

Two Numbers

Addition is : 15.5

Complex Literal is : (10+5j)

Real part is : 10.0

Imaginary Part is : 5.0

Ex.No: 1.b	Convert Kilometer to miles
-------------------	-----------------------------------

Aim:

To write a python Program to Convert Kilo meter to Miles

Algorithm:

1. Start the program
2. Get the floating value of kilometer in the variable km
3. Initialize factor=0.621371
4. Calculate mile=km*factor
5. Print the resultant value

Program:

```
km=float(input("Enter the Kilometer : "))  
factor=0.621371  
mile=km*factor  
print(f' {km} Kilometer is equal to {mile} Miles')
```

Output:

Enter the Kilometer : 2

2.0 Kilometer is equal to 1.242742 Miles

Ex.No: 2.a	Simple Calculator
-------------------	--------------------------

Aim:

To write a python Program to Make a Simple Calculator

Algorithm:

1. Start the program
2. Print the operations like 1 for addition, 2 for subtraction, 3 for multiplication and 4 for division.
3. Get the floating values of first and second number in the variables n1 and n2
4. Get the choice in the variable ch
5. Using multiple elif to perform addition, subtraction, multiplication and division
if ch==1:
 print(f"{n1} + {n2} = {n1+n2}")
elif ch==2:
 print(f"{n1} - {n2} = {n1-n2}")
elif ch==3:
 print(f"{n1} * {n2} = {n1*n2}")
elif ch==4:
 print(f"{n1} / {n2} = {n1/n2}")
else:
 print("\n Invalid Choice...")

Program:

```
#Simple Calculator

print("Select the Operation")

print("1.Addition")

print("2.Substraction")

print("3.Multiplication")

print("4.Division")

n1=float(input("Enter the First Number : "))

n2=float(input("Enter the Second Number : "))
```

```
ch=int(input("\n Enter Your Choice (1/2/3/4) : "))
if ch==1:
    print(f"{n1} + {n2} = {n1+n2}")
elif ch==2:
    print(f"{n1} - {n2} = {n1-n2}")
elif ch==3:
    print(f"{n1} * {n2} = {n1*n2}")
elif ch==4:
    print(f"{n1} / {n2} = {n1/n2}")
else:
    print("\n Invalid Choice...")
```

Output:

Select the Operation

1.Addition

2.Substraction

3.Multiplication

4.Division

Enter the First Number : 30

Enter the Second Number : 40

Enter Your Choice (1/2/3/4) : 3

30.0 * 40.0 = 1200.0

Ex.No: 2.b	Find the Performance of Employees
-------------------	--

Aim:

To write a python program that reads a rating from the user and indicates whether the performance was unacceptable, acceptable or meritorious.

Algorithm:

1. Start the program
2. Enter the float value of rating in the variable rate
3. Initialize amount_of_raise=2400
4. Calculate the performance based upon rate
if rate>=0.6:
 print("\n Meritorious Performance...")
 print("\n The Amount of Employees Raise is : ", rate*amount_of_raise)
elif rate==0.4:
 print("\n Acceptable Performance...")
 print("\n The Amount of Employees Raise is : ", rate*amount_of_raise)
elif rate==0.0:
 print("\n Unacceptable Performance...")
else:
 print("\n Invalid Rating...")

Program:

```
rate=float(input("\n Enter the Rating : "))
```

```
amount_of_raise=2400
```

```
if rate>=0.6:
```

```
    print("\n Meritorious Performance...")
```

```
    print("\n The Amount of Employees Raise is : ", rate*amount_of_raise)
```

```
elif rate==0.4:
```

```
    print("\n Acceptable Performance...")
```

```
    print("\n The Amount of Employees Raise is : ", rate*amount_of_raise)
```

```
elif rate==0.0:
```

```
    print("\n Unacceptable Performance...")
```

```
else:
```

```
    print("\n Invalid Rating...")
```

Output:

Enter the Rating : 0.6

Meritorious Performance...

The Amount of Emplpyees Raise is : 1440.0

Enter the Rating : 0.3

Invalid Rating...

Enter the Rating : 0.4

Accectable Performance...

The Amount of Emplpyees Raise is : 960.0

Ex.No: 3.a

Using nested while loop to print 1-100

Aim:

To write a python program to displays the integer values 1–100 using nested while loop

Algorithm:

1. Start the program
2. Initialize loop1=10, loop2=10 and n=1
3. Using outer while loop to print rows and inner while loop to print columns of the values 1-100.

```
while loop1>0:
    while loop2>0:
        print(n,end=' ')
        n=n+1
        loop2=loop2-1
    print()
    loop2=10
    loop1=loop1-1
```

Program:

```
loop1=10
loop2=10
n=1
while loop1>0: #print row
    while loop2>0: #print column
        print(n,end=' ')
        n=n+1
        loop2=loop2-1
    print()
    loop2=10
```


loop1=loop1-1

Output:

1 2 3 4 5 6 7 8 9 10
11 12 13 14 15 16 17 18 19 20
21 22 23 24 25 26 27 28 29 30
31 32 33 34 35 36 37 38 39 40
41 42 43 44 45 46 47 48 49 50
51 52 53 54 55 56 57 58 59 60
61 62 63 64 65 66 67 68 69 70
71 72 73 74 75 76 77 78 79 80
81 82 83 84 85 86 87 88 89 90
91 92 93 94 95 96 97 98 99 100

Ex.No: 3.b	Using Single while loop to print 1-100
-------------------	---

Aim:

To write a python script to print 1-100 using single while loop.

Algorithm:

1. Start the program
2. Initialize i=1, col=10
3. Using while loop to execute until i<=100
4. Print 1-100 in 10 values per row and 10 values per column
if col>0:
 if i<10:
 print(i,end=' ')
 else:
 print(i,end=' ')
 i=i+1
 col=col-1
else:
 print()
 col=10

Program:

```
#Using Single while loop to print 1-100
```

```
i=1
```

```
col=10
```

```
while i<=100:
```

```
    if col>0:
```

```
        if i<10:
```

```
            print(i,end=' ' )
```

```
        else:
```

```
            print(i,end=' ' )
```

```
i=i+1  
col=col-1  
else:  
    print()  
    col=10
```

Output:

```
1 2 3 4 5 6 7 8 9 10  
11 12 13 14 15 16 17 18 19 20  
21 22 23 24 25 26 27 28 29 30  
31 32 33 34 35 36 37 38 39 40  
41 42 43 44 45 46 47 48 49 50  
51 52 53 54 55 56 57 58 59 60  
61 62 63 64 65 66 67 68 69 70  
71 72 73 74 75 76 77 78 79 80  
81 82 83 84 85 86 87 88 89 90  
91 92 93 94 95 96 97 98 99 100
```

Ex.No: 4

Generate spellings of the last four digits of a phone number

Aim:

To write a python script to generate all the possible spellings of the last four digits of any given phone number using Dictionaries.

Algorithm:

1. Start the program
2. Get Phone number in the format of XXX-XXX-XXXX in the variable phone_num
3. Use dictionary with keys are 0,1,2...9 and get the relevant values of the keys using:
translate={'0':('0'),'1':('1'),'2':('a','b','c'),'3':('d','e','f'),'4':('g','h','i'),\
 '5':('j','k','l'),'6':('m','n','o'),'7':('p','q','r','s'),'8':('t','u','v'),\
 '9':('w','x','y','z')}
4. Use if condition to check the length of the phone number is equal to 12, otherwise print the error message
5. Within if condition use nested for loop to print the relevant spellings of the last four digits.
for let1 in translate[phone_num[8]]:
 for let2 in translate[phone_num[9]]:
 for let3 in translate[phone_num[10]]:
 for let4 in translate[phone_num[11]]:
 print(phone_num[0:8]+let1+let2+let3+let4)

Program:

#To generate all the possible spellings of the last four digits of any given phone number – use Dictionaries

```
phone_num=input('Enter Phone number (XXX-XXX-XXXX): ')
```

```
translate={'0':('0'),'1':('1'),'2':('a','b','c'),'3':('d','e','f'),'4':('g','h','i'),  
          '5':('j','k','l'),'6':('m','n','o'),'7':('p','q','r','s'),'8':('t','u','v'), '9':('w','x','y','z')}
```

```
if len(phone_num)==12:
```

```
    for let1 in translate[phone_num[8]]:
```

```
        for let2 in translate[phone_num[9]]:
```

```
            for let3 in translate[phone_num[10]]:
```

```
        for let4 in translate[phone_num[11]]:
            print(phone_num[0:8]+let1+let2+let3+let4)
else:
    print('Enter valid phone number...')
```

Output:

Enter Phone number (XXX-XXX-XXXX): 945-564-078

Enter valid phone number...

Enter Phone number (XXX-XXX-XXXX): 934-045-0189

934-045-01tw

934-045-01tx

934-045-01ty

934-045-01tz

934-045-01uw

934-045-01ux

934-045-01uy

934-045-01uz

934-045-01vw

934-045-01vx

934-045-01vy

934-045-01vz

Ex.No: 5

Check Register Number Format

Aim:

To write a python script to check whether the register number format is correct or not using string methods.

Algorithm:

1. Start the program
2. Get the string value of Register number in the variable regno
3. Check length of the register number is equal to 10 then check each character of the register number is digit or not using:

```
if (regno[0].isdigit()==True and regno[1].isdigit()==True
    and regno[2].isdigit()==True and regno[3].isdigit()==True
    and regno[4].isdigit()==True and regno[5].isalpha()==True
    and regno[5].isupper()==True or regno[5].islower()==True
    and regno[6].isdigit()==True and regno[7].isdigit()==True
    and regno[8].isdigit()==True and regno[9].isdigit()==True):
    print(" Register number is Correct...")
    print(" Entered Register Number is : ",regno)
```

else:

```
print(" Register Number is incorrect...")
```

4. If the condition failed then print the error message.

Program:

#if the register number format is correct or not using a Python string methods . (Hint: sample register number format- 20751A0500).

```
regno=input(' Enter the Register Number (Sample 20751A0500) : ')
```

```
if len(regno)==10:
```

```
if (regno[0].isdigit()==True and regno[1].isdigit()==True \
    and regno[2].isdigit()==True and regno[3].isdigit()==True \
    and regno[4].isdigit()==True and regno[5].isalpha()==True \
```

```
and regno[5].isupper()==True or regno[5].islower()==True and regno[6].isdigit()==True \
and regno[7].isdigit()==True and regno[8].isdigit()==True \
and regno[9].isdigit()==True):
print(" Register number is Correct...")
print(" Entered Register Number is : ",regno)
else:
print(" Register Number is incorrect...")
else:
print(" Check the Length of the Register number...(Total length of Register number is 10)")
```

Output:

Enter the Register Number (Sample 20751A0500) : 20751A0103

Register number is Correct...

Entered Register Number is : 20751A0103

Enter the Register Number (Sample 20751A0500) : 20751a0

Check the Length of the Register number...(Total length of Register number is 10)

Enter the Register Number (Sample 20751A0500) : 2075104104

Register Number is incorrect...

Ex.No: 6	Generate a Random password
-----------------	-----------------------------------

Aim:

To write a python script to generates a random password using function.

Algorithm:

1. Start the program
2. import random module
3. Define the function: passwordgen()
4. Within the function, password have a random length of between 7 and 10 characters using `length = random.randint(7,10)`
5. Initialize password=""
6. Each character should be randomly selected from positions 33 to 126 in the ASCII table within for loop.
for i in range(length):
 randomchar = chr(random.randint(33,126))
 password=password+randomchar
7. Return the resultant value of password.
8. In the main function, call the user defined function passwordgen()
9. Call the main() function

Program:

```
import random
```

```
def passwordgen():
```

```
    length = random.randint(7,10)
```

```
    password=""
```

```
    for i in range(length):
```

```
        randomchar = chr(random.randint(33,126))
```

```
        password=password+randomchar
```

```
    return password
```



```
def main():  
    print("Your Random password is : ", passwordgen())  
main()
```

Output:

Your Random password is : IWk__w

Ex.No: 7.a

Print half pyramid pattern

Aim:

To write a python program to print half pyramid pattern with star (asterisk)

Algorithm:

1. Start the program
2. Using outer for loop to generate 5 rows and inner for loop to print * using:
for i in range(0, 5):
 for j in range(0, i + 1):
 print("*", end="")
 print()

Program:

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

Output:

```
*  
  
**  
  
***  
  
****  
  
*****
```

Ex.No: 7.b	Print right-angled triangle pattern
-------------------	--

Aim:

To write a python program to print the characters/alphabets in right-angled triangle pattern.

Algorithm:

1. Start the program
2. Initialize asciivalue = 65
3. Using outer for loop to generate 7 rows and inner for loop to print the characters based on asciivalue.

```
for i in range(0, 7):  
    for j in range(0, i + 1):  
        alpha = chr(asciivalue)  
        print(alpha, end=' ')  
        asciivalue = asciivalue+1  
    print()
```

Program:

```
asciivalue = 65
```

```
for i in range(0, 7):
```

```
    for j in range(0, i + 1):
```

```
        alpha = chr(asciivalue)
```

```
        print(alpha, end=' ')
```

```
        asciivalue = asciivalue+1
```

```
    print()
```

Output:

A

BC

DEF

GHIJ

KLMNO

PQRSTU

VWXYZ[\

Ex.No: 8

Draw basic Shapes using Turtle

Aim:

To write a python script to draw the basic shapes using turtle (Square, circle, triangle).

Algorithm:

1. Start the program
2. Import turtle module
3. Create turtle using `t = turtle.Turtle()`
4. Canvas area is created
`win=turtle.Screen()`
`win.setup(500,400)`
5. Set turtle color, shape and size
`t.pencolor('blue')`
`t.pensize(4)`
`t.shape('turtle')`
6. Draw circle with radius 60 using `t.circle(60)`
7. Draw a square :
`for i in range(4):`
`t.left(90)`
`t.forward(100)`
`t.clear()`
8. Draw a triangle :
`for i in range(3):`
`t.left(120)`
`t.forward(100)`

Program:

```
import turtle

t = turtle.Turtle()

win=turtle.Screen()

win.setup(500,400)
```

```
t.pencolor('blue')
```

```
t.pensize(4)
```

```
t.shape('square')
```

```
#draw circle with radius 60 pixels
```

```
t.circle(60)
```

```
t.clear()
```

```
#draw square
```

```
for i in range(4):
```

```
    t.left(90)
```

```
    t.forward(100)
```

```
t.clear()
```

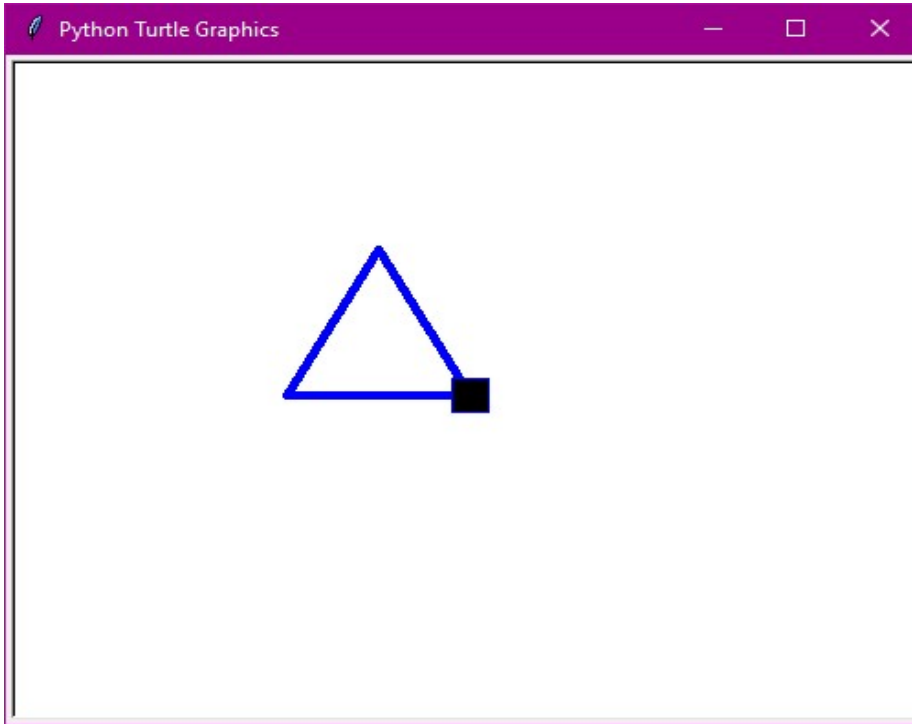
```
#draw triangle
```

```
for i in range(3):
```

```
    t.left(120)
```

```
    t.forward(100)
```

Output:



Ex.No: 9.a	Handling Text File
-------------------	---------------------------

Aim:

To write a python script to create, read, write and display the content of a text file.

Algorithm:

1. Start the program
2. Create ruffl.txt file in the write mode using
file=open("ruffl.txt","w")
3. Write the content into a file using file.write()
4. Open ruffl.txt file in the read mode
5. Print the content of the file using file.read()
6. Finally close the file.

Program:

```
file=open("ruffl.txt","w")  
file.write("DELHI\nMUMBAI\nCHENNAI\nHYDERABAD\nBANGALORE")  
file=open("ruffl.txt","r")  
print(file.read())  
file.close()
```

Output:

DELHI

MUMBAI

CHENNAI

HYDERABAD

BANGALORE

Ex.No: 9.b	Handling Excel File
-------------------	----------------------------

Aim:

To write a python script to create and write on excel file.

Algorithm:

1. Start the program
2. import xlswriter module
3. Create workbook using
book=xlswriter.Workbook('ruff.xlsx')
4. Create worksheet using
sheet=book.add_worksheet()
5. Write the content into a excel file using
sheet.write('name of the shell','data')

6. Finally close the workbook.

Program:

```
import xlswriter

book=xlswriter.Workbook('ruff.xlsx')

sheet=book.add_worksheet()

sheet.write('A1','Science')

sheet.write('B1',90)

sheet.write('A2', 'Mathematics')

sheet.write('B2', 85)

sheet.write('A3', 'Social Science')

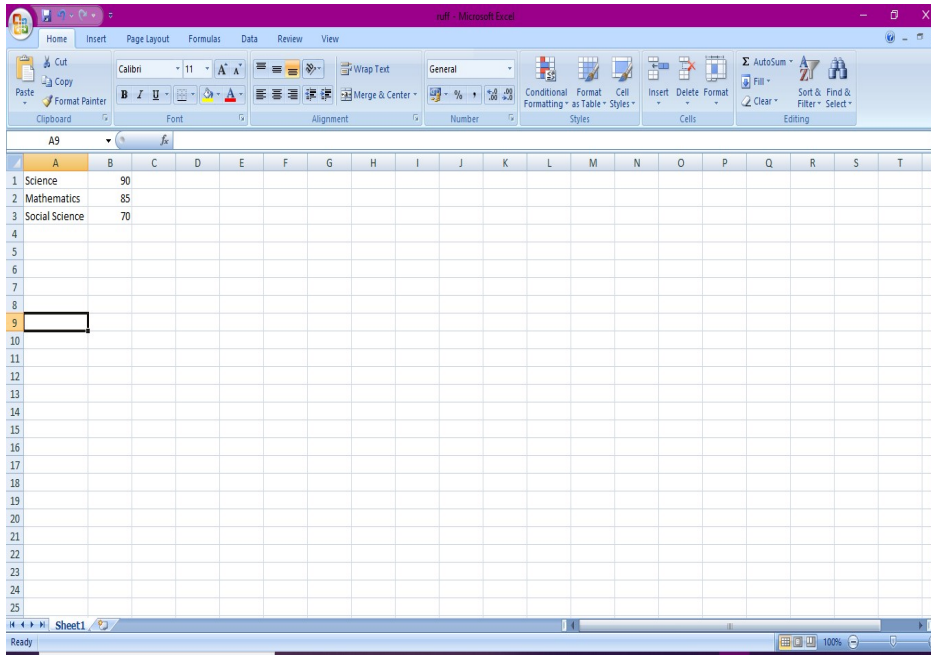
sheet.write('B3', 70)

print("Excel file is created successfully...")

book.close()
```

Output:

Excel file is created successfully...



Ex.No: 9.c

Handling csv File

Aim:

To write a python script to write the contents into a csv file.

Algorithm:

1. Start the program
2. import csv module
3. Add column heading using
 `head=['REGNO','NAME','BRANCH','YEAR']`
4. Add data using
 `data=[['111','AAA','MECH','I'],['222','BBB','CIVIL','I']]`
5. Create file name i.e stu.csv
6. writing data into csv file using
 with `open(filename,'w')` as file:
 `csvfile=csv.writer(file)`
 `csvfile.writerow(head)`
 `csvfile.writerows(data)`
7. reading data from csv file using
 with `open(filename,'r')` as file:
 `csvfile=csv.reader(file)`
 for display in csvfile:
 `print(display)`

Program:

```
import csv

head=['REGNO','NAME','BRANCH','YEAR']

data=[['111','AAA','MECH','I'],['222','BBB','CIVIL','I'],\
      ['333','CCC','MECH','II'],['444','DDD','CIVIL','II']]

filename="stu.csv"

#writing data into csv file

with open(filename,'w') as file:
```

```
csvfile=csv.writer(file)
```

```
csvfile.writerow(head)
```

```
csvfile.writerows(data)
```

```
#reading data from csv file
```

```
with open(filename,'r') as file:
```

```
    csvfile=csv.reader(file)
```

```
    for display in csvfile:
```

```
        print(display)
```

Output:

```
['REGNO', 'NAME', 'BRANCH', 'YEAR']
```

```
['111', 'AAA', 'MECH', 'I']
```

```
['222', 'BBB', 'CIVIL', 'I']
```

```
['333', 'CCC', 'MECH', 'II']
```

```
['444', 'DDD', 'CIVIL', 'II']
```

Ex.No: 10.a	Copy the contents from one file into another file
--------------------	--

Aim:

To write a python script to copy the contents of one file into another file.

Algorithm:

1. Start the program
2. Open first.txt file in the write mode
3. Use write() to write the content into a text file of first.txt
4. Open first.txt file in the read mode and second.txt file in the write mode
5. Using for loop to copy the content from first.txt file into second.txt file.
with open("first.txt","r") as f1:
 with open("second.txt","w") as f2:
 for a in f1:
 f2.write(a)
6. Open second.txt file in the read mode and print read() method to read the content.
file=open("second.txt","r")
print(file.read())

Program:

```
# Copy the contents of one file into another file.
```

```
f=open("first.txt","w")
```

```
f.write("RED \n")
```

```
f.write("BLUE \n")
```

```
f.write("GREEN \n")
```

```
f.close()
```

```
with open("first.txt","r") as f1:
```

```
    with open("second.txt","w") as f2:
```

```
        for a in f1:
```

```
            f2.write(a)
```

```
file=open("second.txt","r")
```

```
print(file.read())
```

Output:

RED

BLUE

GREEN

Ex.No: 10.b	Count the Number of Lines in a File
--------------------	--

Aim:

To write a python script to Count the number of lines in a file.

Algorithm:

1. Start the program
2. Initialize numline=0
3. Open first.txt file in read mode and count the number of lines in a file.
with open("first.txt","r") as file:
 for line in file:
 numline=numline+1
print(f" Number of Lines in a file is : {numline}")

Program:

```
# Count number of lines in a file  
  
numline=0  
  
with open("first.txt","r") as file:  
  
    for line in file:  
  
        numline=numline+1  
  
print(f" Number of Lines in a file is : {numline}")
```

Output:

Number of Lines in a file is : 3

Ex.No: 10.c	Count the number of characters in a file
--------------------	---

Aim:

To write a python script to count the number of characters in a file.

Algorithm:

1. Start the program
2. Initialize numchar=0
3. Open first.txt file in the read mode and count the number of characters in a file using:
4. with open("first.txt","r") as file:
 for line in file:
 numchar=numchar+len(line)
 print(f' Number of Characters in a file is : {numchar}')

Program:

```
# Count number of characters in a file  
  
numchar=0  
  
with open("first.txt","r") as file:  
  
    for line in file:  
  
        numchar=numchar+len(line)  
  
print(f' Number of Characters in a file is : {numchar}')
```

Output:

Number of Characters in a file is : 18

Ex.No: 10.d	Count the number of words in a file
--------------------	--

Aim:

To write a python script to count the number of words in a file.

Algorithm:

1. Start the program
2. Initialize numword=0
3. Open first.txt file in the read mode and count the number of words in a file.
with open("first.txt","r") as file:
 for line in file:
 word=line.split()
 numword=numword+len(word)
print(f" Number of Words in a file is : {numword}")

Program:

```
# Count number of words in a file  
numword=0  
with open("first.txt","r") as file:  
    for line in file:  
        word=line.split()  
        numword=numword+len(word)  
print(f" Number of Words in a file is : {numword}")
```

Output:

Number of Words in a file is : 3

Ex.No: 11

Area of the Rectangle using single inheritance

Aim:

To write a python script to calculate the area of the rectangle using single inheritance.

Algorithm:

1. Start the program
2. Create a parent class i.e first
3. In the class first, get the values of length and breadth in the variable l and b.
4. Create a child class name as second, it inherited parent class using
class second(first):
5. In the child class, create a function i.e fl()
6. In the function fl() in child class inherit the l and b variable of the parent class using
super()
7. Calculate area of the rectangle using $area=super().l*super().b$
8. Print the result
9. Create an object of the class second and call the function fl()

Program:

```
#area of the rectangle using single inheritance
```

```
class first:
```

```
    l=float(input("Enter the Length : "))
```

```
    b=float(input("Enter the Breadth : "))
```

```
class second(first):
```

```
    def fl(self):
```

```
        area=super().l*super().b
```

```
        print(f'Area of the Rectangle is : {area}')
```

```
ob=second()
```

```
ob.fl()
```

Output:

Enter the Length : 10

Enter the Breadth : 20

Area of the Rectangle is : 200.0

Ex.No: 12.a	Area of a Circle using math Module
--------------------	---

Aim:

To write a python script to calculate area of a circle using math module.

Algorithm:

1. Start the program
2. Import math module
3. Get the float value of radius in the variable r
4. Call pi attribute and pow() function from math module to calculate area of a circle.
area=math.pi*math.pow(r,2)
5. Print the resultant value

Program:

```
#Area of the Circle  
  
import math  
  
r=float(input("Enter the Radius : "))  
  
area=math.pi*math.pow(r,2)  
  
print(f'Area of the Circle is : {area}')
```

Output:

Enter the Radius : 5

Area of the Circle is : 78.53981633974483

Ex.No: 12.b	Volume of a Sphere using math Module
--------------------	---

Aim:

To write a python program to calculate volume of a sphere using math module.

Algorithm:

1. Start the program
2. Import math module
3. Get the float value of radius in the variable r
4. Call pi attribute and pow() function from math module to calculate volume of a sphere.
 $vol=(4/3)*math.pi*math.pow(r,3)$
5. Print the resultant value

Program:

```
#Volume of a Sphere
import math
r=float(input("Enter the Radius : "))
vol=(4/3)*math.pi*math.pow(r,3)
print(f"Volume of a Sphere is : {vol}")
```

Output:

Enter the Radius : 5

Volume of a Sphere is : 523.5987755982989

Ex.No: 13

Draw Line plot using Pandas

Aim:

To write a python script to create a line plot of the historical stock prices of a company between two specific dates using Pandas library.

Algorithm:

1. Start the program
2. Import pandas and matplotlib.pyplot modules
3. Read the content of stock.csv file
`df = pd.read_csv("stock.csv")`
4. to_datetime() method helps to convert string Date time into Python Date time object.
`start = pd.to_datetime('2020-4-1')`
`end = pd.to_datetime('2020-7-31')`
5. Get the Date column using list from stock.csv file and convert into Python Date time object
`df['Date']=pd.to_datetime(df['Date'])`
6. Get the in between dates of '2020-4-1' and '2020-07-31'
`new_df=(df['Date']>= start) & (df['Date']<= end)`
7. DataFrame.loc[] method that takes only index labels and returns row or dataframed
`f1=df.loc[new_df]`
8. Draw Pandas set_index() method is used to set the List, Series, or DataFrame as an index of a Data Frame.
`df2 = df1.set_index('Date')`
9. figure() function in pyplot module of matplotlib library is used to create a new figure with figsize=(6,6) using plt.figure(figsize=(6,6))
10. subtitle() method of matplotlib library is used to Add a centered title to the figure.
`plt.subtitle('Stock prices of a Company (01-04-2020 to 31-07-2020)', fontsize=18, color='blue')`
11. xlabel() and ylabel() function of matplotlib library is used to set the label for the x-axis and yaxis.
`plt.xlabel("Date",fontsize=16, color='red')`

```
plt.ylabel("$ Price", fontsize=16, color='red')
```

12. Draw a line plot of the historical stock prices of a company using
`df2['Close'].plot(color='green')`

13. `Show()` is used to display the figures using `plt.show()`

Program:

```
import pandas as pd
```

```
import matplotlib.pyplot as plt
```

```
df = pd.read_csv("stock.csv")
```

```
start = pd.to_datetime('2020-4-1')
```

```
end = pd.to_datetime('2020-7-31')
```

```
df['Date']=pd.to_datetime(df['Date'])
```

```
new_df=(df['Date']>= start) & (df['Date']<= end)
```

```
df1=df.loc[new_df]
```

```
df2 = df1.set_index('Date')
```

```
plt.figure(figsize=(6,6))
```

```
plt.suptitle('Stock prices of a Company (01-04-2020 to 31-07-2020)',fontsize=18, color='blue')
```

```
plt.xlabel("Date",fontsize=16, color='red')
```

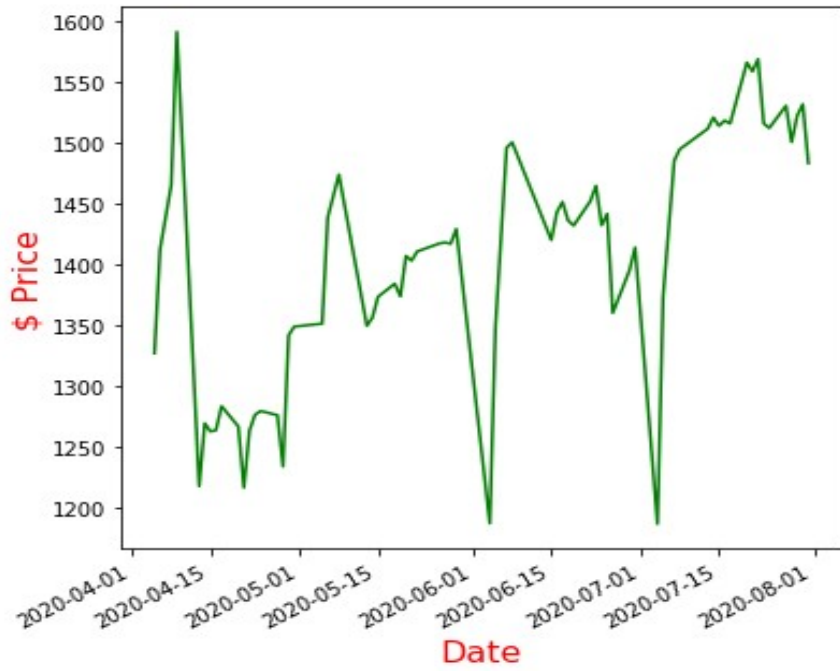
```
plt.ylabel("$ Price", fontsize=16, color='red')
```

```
df2['Close'].plot(color='green')
```

```
plt.show()
```

Output:

Stock prices of a Company (01-04-2020 to 31-07-2020)



Ex.No: 14

Draw Bar Plot using Pandas

Aim:

To write a python script to create a bar plot of the trading volume of a company stock between two specific dates.

Algorithm:

1. Start the program
2. Import pandas and matplotlib.pyplot modules
3. Read the content of stock.csv file
`df = pd.read_csv("stock.csv")`
4. to_datetime() method helps to convert string Date time into Python Date time object.
`start = pd.to_datetime('2020-4-1')`
`end = pd.to_datetime('2020-7-31')`
5. Get the Date column using list from stock.csv file and convert into Python Date time object
`df['Date']=pd.to_datetime(df['Date'])`
6. Get the in between dates of '2020-5-1' and '2020-5-31'
`new_df=(df['Date']>= start) & (df['Date']<= end)`
7. DataFrame.loc[] method that takes only index labels and returns row or dataframe
`f1=df.loc[new_df]`
8. Draw Pandas set_index() method is used to set the List, Series, or DataFrame as an index of a Data Frame.
`df2 = df1.set_index('Date')`
9. figure() function in pyplot module of matplotlib library is used to create a new figure with figsize=(6,6) using plt.figure(figsize=(6,6))
10. subtitle() method of matplotlib library is used to Add a centered title to the figure.
`plt.subtitle("Trading Volume of a Company (01-05-2020 to 31-05-2020)",
fontsize=15, color='green')`
11. xlabel() and ylabel() function of matplotlib library is used to set the label for the x-axis and yaxis.
`plt.xlabel("Date",fontsize=15, color='red')`

```
plt.ylabel("Trading Volume", fontsize=15, color='red')
```

12. Draw a bar plot of the trading volume of a company using

```
df2['Volume'].plot(kind='bar',color='orange')
```

13. Show() is used to display the figures using plt.show()

Program:

```
import pandas as pd
```

```
import matplotlib.pyplot as plt
```

```
df = pd.read_csv("stock.csv")
```

```
start = pd.to_datetime('2020-5-1')
```

```
end = pd.to_datetime('2020-5-31')
```

```
df['Date'] = pd.to_datetime(df['Date'])
```

```
new_df = (df['Date']>= start) & (df['Date']<= end)
```

```
df1 = df.loc[new_df]
```

```
df2 = df1.set_index('Date')
```

```
plt.figure(figsize=(6,6))
```

```
plt.suptitle("Trading Volume of a Company (01-05-2020 to 31-05-2020)", fontsize=15,  
color='green')
```

```
plt.xlabel("Date",fontsize=15, color='red')
```

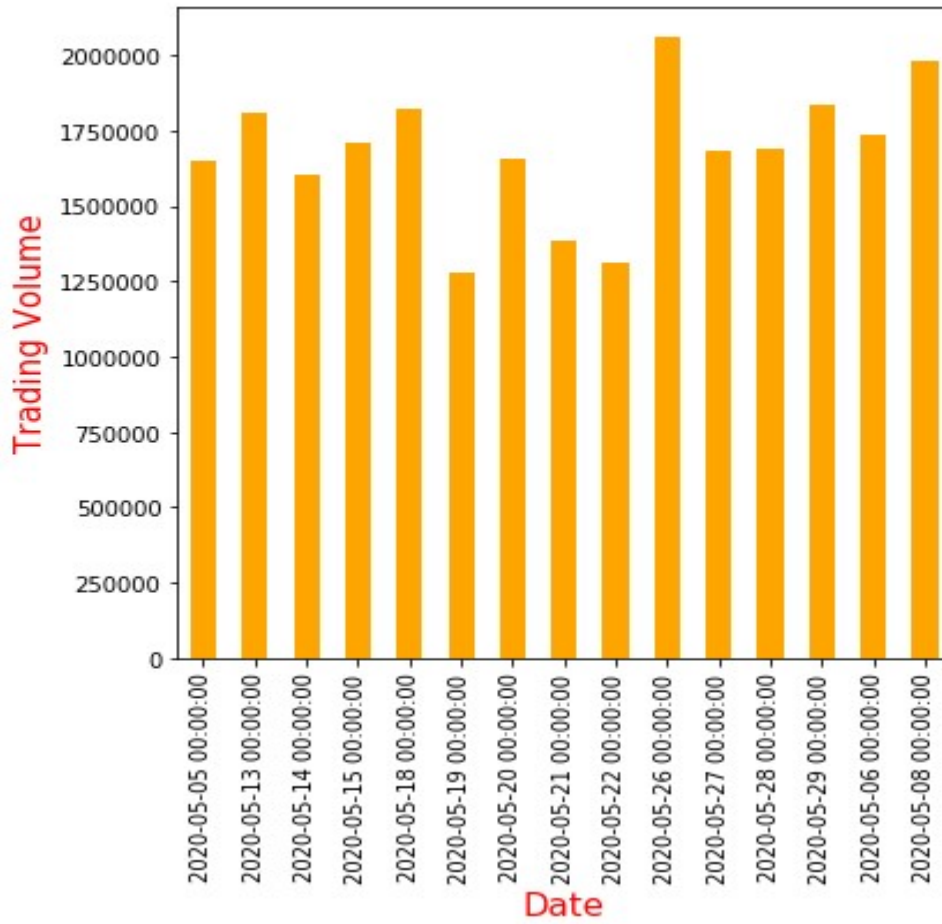
```
plt.ylabel("Trading Volume", fontsize=15, color='red')
```

```
df2['Volume'].plot(kind='bar',color='orange');
```

```
plt.show()
```

Output:

Trading Volume of a Company (01-05-2020 to 31-05-2020)



Ex.No: 15	Various List operations
------------------	--------------------------------

Aim:

To write a python script to perform various list operations.

Algorithm:

1. Start the program
2. Create list in the variable li,
li=['apple','orange','mango','banana']
3. print(type(li)) – print the class list
4. print the first value in the list, print(li[0])
5. print(li[1:]) – print the second value onwards from the list
6. print(li[0]) and print(li[1:]) – print the first value and print the second value from the list
7. append() method used to add an item in end of the list
8. remove() method used to delete an item from the list
9. Insert an item based on index using:
li.insert(2,"apple")
10. creat mixed list, mi=['hai',2,5.999]
11. Concatenation of two list.
print(li+mi)
12. Clear the list
print(mi.clear())

Program:

```
li=['apple','orange','mango','banana']  
  
print(type(li))  
  
print(li[0])  
  
print(li[1:])  
  
#append() add an item in end of the list  
  
li.append("berry")  
  
print(li)  
  
#remove() delete an item
```

```
li.remove("apple")
print(li)
#insert an item based on index
li.insert(2,"apple")
print(li)
#creat mixed list
mi=['hai',2,5.999]
print(mi)
#Concatenation of two list
print(li+mi)
#clear the list
print(mi.clear())
```

Output:

```
<class 'list'>
apple
['orange', 'mango', 'banana']
['apple', 'orange', 'mango', 'banana', 'berry']
['orange', 'mango', 'banana', 'berry']
['orange', 'mango', 'apple', 'banana', 'berry']
['hai', 2, 5.999]
['orange', 'mango', 'apple', 'banana', 'berry', 'hai', 2, 5.999]
None
```

Ex.No: 16	Method Overriding
------------------	--------------------------

Aim:

To write a python script to calculate area of the circle, square and rectangle using method overriding.

Algorithm:

1. Start the program
2. Create base class as cir
3. Define the function name is area(), to get the radius in the variable r and calculate area of the circle and print the resultant value.
4. Create a child class and the name is sqr which is inherit parent class cir
class sqr(cir):
5. Define the function name is area() in the class sqr, to get the side in the variable a and calculate area of the square and print the resultant value.
6. Create a child class and the name is rect which is inherit parent class cir
class rect(cir):
7. Define the function name is area() in the class rect, to get the length and breadth in the variables l and b and calculate area of the rectangle and print the resultant value.
8. Create objects of the classes cir,sqr and rect
9. Using for loop to call the function area() of three objects of classes

Program:

```
# Method overriding
```

```
class cir:
```

```
    def area(self):
```

```
        r=float(input("Enter the radius : "))
```

```
        res=3.14*r*r
```

```
        print(f'Area of the Circle is : {res}')
```

```
class sqr(cir):
```

```
    def area(self):
```

```
        a=float(input("Enter the side : "))
```

```
    res=a*a

    print(f'Area of the Square is : {res}')

class rect(cir):

    def area(self):

        l=float(input("Enter the length : "))

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

        res=l*b

        print(f'Area of the rectangle is : {res}')

ob1=cir()

ob2=sqr()

ob3=rect()

for a in (ob1,ob2,ob3):

    a.area()
```

Output:

Enter the radius : 5

Area of the Circle is : 78.5

Enter the side : 10

Area of the Square is : 100.0

Enter the length : 6

Enter the breadth : 7

Area of the rectangle is : 42.0

Ex.No: 17

Handling Excel file using Pandas

Aim:

To write a python script to read and write excel file using pandas library.

Algorithm:

1. Start the program
2. Import pandas library alias name is pd
3. DataFrame is like a table with rows and columns
`df = pd.DataFrame({'States':['TN', 'AP', 'UP', 'MP'],
 'Capitals':['CHENNAI', 'AMARAVATHI', 'LUCKNOW', 'BHOPAL']})`
4. `to_excel()` method is used to export the DataFrame to the excel file.
`df.to_excel('data1.xlsx')`
5. `read_excel()` method to read the excel file data into a DataFrame object.
`data = pd.read_excel('data1.xlsx')`
6. print the content of Excel file

Program:

```
import pandas as pd
```

```
df = pd.DataFrame({'States':['TN', 'AP', 'UP', 'MP'],  
                  'Capitals':['CHENNAI', 'AMARAVATHI', 'LUCKNOW', 'BHOPAL']})
```

```
df.to_excel('data1.xlsx')
```

```
data = pd.read_excel('data1.xlsx')
```

```
print(data)
```

Output:

Unnamed: States Capitals

0 TN CHENNAI

1 AP AMARAVATHI

2 UP LUCKNOW

3 MP BHOPAL