

output

5

The factorial of 5 is : 120

Experiment-25

Objective :- Find factorial of a number using recursion

Code

```
def factorial(n):
```

```
    if n < 0:
```

~~raise ValueError("Factorial is not defined for negative numbers")~~

```
    elif n == 0:
```

```
        return 1
```

```
    else:
```

```
        return n * factorial(n-1)
```

Get Input from user

```
number = int(input("Enter a non-negative number :"))
```

Calculate & print the factorial

Try :

```
result = factorial(number)
print(f"The Factorial of {number} is : {result}")
```

Except ValueError as e:
print(e)

output :

Enter a number : 5

Enter Exponent : 3

125.0

Enter a number : 25
5

Experiment - 26

objective : Demonstrate math Built-in functions

Code

```
import math
a = int(input("Enter a number :"))
e = int(input("Enter exponent :"))
```

```
result = math.pow(a,e)
print(result)
```

Code

```
import math
a = int(input("Enter a number :"))
result = math.sqrt(a)
print(result)
```

Output :

Converted String :

ARYAN IS ASTRO

Converted String :

aryan is astro

Converted String :

Aryan Is Astro

Converted String:

Aryan Is Astro

Converted String:

ARYAN IS astro

Expt. No. 27

Experiment-27

Objective : Demonstrate String Built-in function

Text = 'Aryan is ASTRO'

print ("In Converted String :")
print (Text.upper())

print ("In Converted String :")
print (Text.lower())

print ("In Converted String :")
print (Text.title())

print ("In Converted String :")
print (Text.capitalize())

print ("In Converted String :")
print (Text.swapcase())

print ("In original string")
print (Text)

output :

47.00 celcius is equivalent to : 116.60 fahrenheit

Date _____

Page No. _____

Expt. No. 28

Experiment-28

objective : Convert Temp to fahrenheit

Code

Temp in celcius degree

$$\text{celcius} = 47$$

Converting Temp To fahrenheit

using formula

$$\text{fahrenheit} = (\text{celcius} \times 1.8) + 32$$

Printing the Result

print ("%.2f celcius is equivalent to
: %.2f fahrenheit" % (celcius, fahrenheit))

output

No of unique words in the file are : 18

Experiment - 29

objective : Input a Text file & print all unique words in the file.

```

def countuniquewords (filename):
    file = open (filename, 'r')
    read_file = file.read ().lower ()
    words_in_file = read_file .split ()
    count_map = {}
    for i in words_in_file :
        if i in count_map :
            count_map [i] += 1
        else :
            count_map [i] = 1
    count = 0
    for i in count_map :
        if count_map [i] == 1 :
            count += 1
    file.close ()
    return count
with open ("gfg.txt", "w") as file :
    file.write ("Vallabh was created with a goal in mind To provide well written codes with thought and well explained solutions for selected questions")
print ("No of unique words in the file are : ", countuniquewords ('gfg.txt'))

```

output :

5.0
5.3

Expt. No. 30

Date _____

Page No. _____

Experiment-30 : ~~100000~~

objective : Convert Integer into floating value

a - int = 3

b - int = 2

c - float - sum = float (a - int + b - int)
print (c - float - sum)

OR

c - int - sum = int (a - float + b - float)
print (c - int - sum)

c - float - sum = a - float + b - float
print (c - float - sum)

Expt. No. 31output:

8.

Experiment - 31objective: Class To Implement flow (x,n)# Python Program for the above approach
def power (x,n):

pow=1

for i in range (n):
 pow = pow * x

return pow

driver code
if __name__ == '__main__':

x=2

n=3

function call

print (power (x,n))

output :

apple

Banana
cherry

b
a
n
a
n
a

apple
banana

Expt. No. 32

Date _____

Page No. _____

Experiment - 32

objective : Program using for loop

Code 1

```
fruits = ["apple", "banana", "cherry"]
for x in fruits:
    print(x)
```

loop through a string

```
for x in "banana":
```

print(x)

Code 3

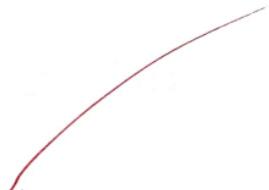
```
fruits = ["apple", "banana", "cherry"]
for x in fruits:
    print(x)
```

```
if x == "banana":
    break
```

output :

1
2
3
4
5
6

1
2
3



Expt. No. 33

Date _____
Page No. _____

Experiment - 33

objective : Program using while loop

Print i as long as i is less than 6 :

```
i = 1
while i < 6:
    print(i)
```

i + 1

Exit loop when i is 3 :

i = 1

while i < 6:

print(i)

if i == 3:

break

i + 1

output :

A New Exception occurred : 6

Expt. No. 34

Page No. _____

Experiment - 34

objective : Program to handle user defined expression .

code

```
class My_Error(Exception):  
    def __init__(self, value):  
        self.value = value
```

```
    def __str__(self):  
        return repr(self.value)
```

```
try:  
    raise (My_Error(3*2))
```

```
except My_Error as error:  
    print ('A New exception occurred:', error.  
          value)
```

output :

This function is in parent class

This function is in child class



Date _____
Expt. No. 35 _____

Page No. _____

Experiment - 35

Objective : Demonstrate Simple and multiple Inheritance

Single Inheritance

class Parent :

def fun1(self):

print ("This function is in Parent class.")

~~class Child (Parent):~~

def fun2(self):

print ("This function is in child class.")

object = child()

object = fun1()

object = fun2()

Multiple Inheritance

class Mother :

mothername = " "

def mother (self):

print (self.mothername)

class Father :

fathername = " "

def father (self):

print (self.fathername)

Output:

Father : RAM

Mother : SITA

class Son (mother, father):

def parents (self):

print ("father : ", self.Fathername)

print ("mother : ", self.Mothername)

S1 = Son()

S1.Fathername = "RAM"

S1.Mothername = "SITA"

S1.Parent()

✓ VOLY