output

5

The factorial of 5 is : 120

Date

Expt. No. 25

Page No.

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)
```

JAI BHARAT

Teacher's Signature

output :

Enter a number : 5

Enter exponent : 3

125.0

Enter a number : 25

5

Date

Expt. No. 26

Page No.

Experiment - 26

objective : Demonstrate math Builtin function

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

Date

Expt. No. 27

Page No.

Experiment-27

objective : Demonstrate String Built in function

```
Text = 'Aryan is ASTRO'

print ("\n Converted String :")
print (Text.upper())

print ("\n Converted String :")
print (Text.lower())

print ("\n Converted String:")
print (Text.Title())

print ("\n Converted String:")
print (Text.capitalize())

print ("\n Converted String :")
print (Text.swapcase())

print ("\n original String")
print (Text)
```

output :

47.00 celcius is equivalent To : 116.60 Fahrenheit

Date

Expt. No. 28

Page No.

Experiment-28

Objective : Convert Temp to Fahrenheit

```
# Code
# Temp in celcius degree
Celcius = 47
# Converting Temp To Fahrenheit
# using formula
Fahrenheit = (Celsius * 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

Date:

Expt. No. 29

Page No.

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 words)
        thought and well explained solutions for
        selected questions")
print ("No of unique words in the file are:',
        Countunique words ('gfg.Txt'))
```

JAI BHARAT

Teacher's Signature

output :

5.0

5
5.3

Date

Expt. No. 30

Page No.

Experiment-30

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)
```

output :

8

Date

Expt. No. 31

Page No.

Experiment - 31

Objective : 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 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 Date

Page No.

Experiment - 34

Objective : Program to handle user defined expression.

Code

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

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

try:
    raise (MyError(3*2))

except MyError as error:
    print('A New Exception occurred:', error.
          value)
```

output:

This function is in parentclass

This function is in child class

Date

Expt. No. 35

Page No.

Experiment-35

objective: Demonstrate Singee and multiple Inheritance

```
# Singee 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)
```

JAI BHARAT

Teacher's Signature

output :

Father : RAM

Mother : SITA

Date

Expt. No.

Page No.

```
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()
```