#calculator pgm

def add(num1,num2):*#calculator program* return num1+num2  
def sub(num1,num2):  
 return num1-num2  
def mul(num1,num2):  
 return num1\*num2  
def div(num1,num2):  
 return num1/num2  
print(**"select your operation :**\n**"**,**"1.add**\n**"**,**"2.sub**\n**"**,**"3.mul**\n**"**,**"4.div**\n**"**)  
select=int(input(**"select your options from 1,2,3,4:"**))  
num\_1=int(input(**"enter your 1st num:"**))  
num\_2=int(input(**"enter your 2nd num:"**))  
if select==1:  
 print(num\_1,**"+"**,num\_2,**"="**,add(num\_1,num\_2))  
elif select==2:  
 print(num\_1,**"-"**,num\_2,**"="**,sub(num\_1,num\_2))  
elif select==3:  
 print(num\_1,**"\*"**,num\_2,**"="**,mul(num\_1,num\_2))  
elif select==4:  
 print(num\_1,**"/"**,num\_2,**"="**,div(num\_1,num\_2))  
else:  
 print(**"invalid"**)

output:

select your operation :

1.add

2.sub

3.mul

4.div

select your options from 1,2,3,4:2

enter your 1st num:2

enter your 2nd num:3

2 - 3 = -1

#next pgm…….arguments

def calculatesum(\*arguments):*#'\*'variable no.of func* sum=0  
 for num in arguments:  
 sum += num  
 print(sum)  
calculatesum(5,5,5,5)  
def calc(\*\*arguments):*#'\*\*'keywords of arguments* for arg in arguments.items():  
 print(arg)  
calc(arguments1=**"div"**,arguments2=**"uv"**)

output

20

('arguments1', 'div')

('arguments2', 'uv')  
  
#next pgm……..pattern pgm

lastNumber = 6*#number pattern*for row in range(1, lastNumber):  
 for column in range(1, row + 1):  
 print(column, end=**' '**)  
 print(**""**)

output:

1

1 2

1 2 3

1 2 3 4

1 2 3 4 5

*#pyramid star pattern*num=int(input(**"enter the number of rows:"**))  
for i in range(0,num):  
 for j in range(0,num-1):*#loop for space* print(end=**""**)  
 for j in range(0,2\*i+1):*#loop for print star* print(**"\*"**,end=**""**)  
 print()

Output:

enter the number of rows:5

\*

\*\*\*

\*\*\*\*\*

\*\*\*\*\*\*\*

#next pgm……data types

task=(1,2,3,4)*#list*print(list(task))  
set1=[**"divya"**,**"yuvi"**,**"rema"**]*#set*print(**"reama in set1"**)  
tuple=(**"divya"**,**"yuvi"**,**"rema"**) *#tuple*print(tuple[1])  
result=[]*#dictionary*print(dict(result))  
w=lambda a:a\*10*#lambda*print(w(2))  
tassk=(1,2,3,5)*#range*print(list(range(3)))

output:

[1, 2, 3, 4]

reama in set1

yuvi

{}

20

[0, 1, 2]

#next pgm….basic pgm

def fact(n):#...........factorial pgm  
 return 1 if (n == 1 or n == 0) else n \* fact(n - 1);  
num = 5;  
print(**"Factorial of"**, num, **"is"**,fact(num))

output:

Factorial of 5 is 120

def isPrime(n):#.........prime number  
 if n <= 1:  
 return False  
 for i in range(2, n):  
 if n % i == 0:  
 return False;  
  
 return True  
print(**"true"**) if isPrime(11) else print(**"false"**)  
print(**"true"**) if isPrime(14) else print(**"false"**)

output:

true

false

import math#.........fibonacci pgm  
def isPerfectSquare(x):  
 s = int(math.sqrt(x))  
 return s \* s == x  
def isFibonacci(n):  
 return isPerfectSquare(5 \* n \* n + 4) or isPerfectSquare(5 \* n \* n - 4)  
for i in range(1, 11):  
 if (isFibonacci(i) == True):  
 print(i, **"is a Fibonacci Number"**)  
 else:  
 print(i, **"is a not Fibonacci Number "**)

output:

1 is a Fibonacci Number

2 is a Fibonacci Number

3 is a Fibonacci Number

4 is a not Fibonacci Number

5 is a Fibonacci Number

6 is a not Fibonacci Number

7 is a not Fibonacci Number

8 is a Fibonacci Number

9 is a not Fibonacci Number

10 is a not Fibonacci Number

year = int(input(**"Enter Year: "**))#..........leapyear pgm  
if year % 4 == 0 and year % 100 != 0:  
 print(year, **"is a Leap Year"**)  
elif year % 100 == 0:  
 print(year, **"is not a Leap Year"**)  
elif year % 400 ==0:  
 print(year, **"is a Leap Year"**)  
else:  
 print(year, **"is not a Leap Year"**)

output:

Enter Year: 2019

2019 is not a Leap Year

def convert24(str1):#.............timeconventor pgm  
 if str1[-2:] == **"AM"** and str1[:2] == **"12"**:  
 return **"00"** + str1[2:-2]   
 elif str1[-2:] == **"AM"**:  
 return str1[:-2]  
 elif str1[-2:] == **"PM"** and str1[:2] == **"12"**:  
 return str1[:-2]  
 else:  
 return str(int(str1[:2]) + 12) + str1[2:8]  
print(convert24(**"08:05:45 PM"**))

output:

20:05:45

c = **'y'#............ascii character**print(**"The ASCII value of '"** + c + **"' is"**, ord(c))

output:

The ASCII value of 'y' is 121

def squaresum(n):*#sum of square* sm = 0  
 for i in range(1, n + 1):  
 sm = sm + (i \* i)  
 return sm  
n = 4  
print(squaresum(n))

output:

30

def simple\_interest(p, t, r):*#simple interset* print(**'The principal is'**, p)  
 print(**'The time period is'**, t)  
 print(**'The rate of interest is'**, r)  
  
 si = (p \* t \* r) / 100  
  
 print(**'The Simple Interest is'**, si)  
 return si  
simple\_interest(2, 3, 2)

output:

The principal is 2

The time period is 3

The rate of interest is 2

The Simple Interest is 0.12