

TASK AND SOLUTION (DAY-01)

01. Print the given strings as per stated format. Given strings:

"Data" "Science" "Program" "started"

"By" "IBM"

output - Data-Science-Program-started-By-IBM

In [1]:

```
print("Data", "Science", "Program", sep='-', end='-started-')
print("By", "IBM", sep='-')
```

Data-Science-Program-started-By-IBM

02. Write a program that will convert celsius value to fahrenheit.

In [2]:

```
celcius=float(input('Enter the temp in celcius:'))
faren=celcius*(9/5)+32
print(faren, 'F')
```

03. Take 2 numbers as input from the user. Write a program to swap the numbers without using any special python syntax.

In [3]:

```
a=int(input("Enter the first number:"))
b=int(input("Enter the second number:"))
temp=a
a=b
b=temp
print('value of a:',a)
print('value of b:',b)
```

Enter the first number:3
Enter the second number:5
value of a: 5
value of b: 3

04. Write a program to find the euclidean distance between two coordinates. Take both the coordinates from the user as input.

In [4]:

```
p1x=int(input('Enter x cood of 1st point:'))
p1y=int(input('Enter y cood of 1st point:'))
p2x=int(input('Enter x cood of 2nd point:'))
p2y=int(input('Enter y cood of 2nd point:'))
distance=((p2x-p1x)**2+(p2y-p1y)**2)**0.5
print(round(distance,2))
```

Enter x cood of 1st point:0
Enter y cood of 1st point:0
Enter x cood of 2nd point:2
Enter y cood of 2nd point:2

Enter y coord of 2nd point:2
2.83

05. Write a program to find the simple interest when the value of principle, rate of interest and time period is provided by the user.

In [5]:

```
p=int(input('Enter amount:'))
t=int(input('Enter time period:'))
r=float(input('Enter rate:'))
interest=(p*t*r)/100
print('the interest is', interest)
```

Enter amount:10000
Enter time period:2
Enter rate:5
the interest is 1000.0

06. Write a program that will tell the number of dogs and chicken are there when the user will provide the value of total heads and legs.

In [6]:

```
total_heads=int(input("Enter the number of heads:"))
total_legs=int(input("Enter the number of legs:"))
if total_legs<total_heads:
    print("invalid input!")
else:
    dogs=(total_legs-(2*total_heads))/2
    chicken=total_heads-dogs
    print("Number of dogs in farm are:"+str(dogs))
    print("Number of chicken farm are:"+str(dogs))
```

Enter the number of heads:4
Enter the number of legs:12
Number of dogs in farm are:2.0
Number of chicken farm are:2.0

07. Write a program to find the sum of squares of first n natural numbers where n will be provided by the user.

In [7]:

```
n=int(input("Enter the number:"))
i=1
sum=0
while(i<=n):
    sum=sum+(i*i)
    i=i+1
print("sum= ", sum)
#OR
n=int(input('Enter the number:'))
result=(n*(n+1)*(2*n+1))/6
print(result)
```

Enter the number:5
sum= 55
Enter the number:5
55.0

08. Given the first 2 terms of an Arithmetic Series. Find the Nth term of the series. Assume all inputs are provided by the user.

In [8]:

```
first_term=int(input('Enter 1st term:'))
```

```

first_term=int(input('Enter 1st term: '))
second_term=int(input('Enter 2nd term: '))
n=int(input('Enter the value of n'))
d=second_term-first_term
an=first_term+(n-1)*d
print(an)

```

```

Enter 1st term:3
Enter 2nd term:6
Enter the value of n5
15

```

09. Given 2 fractions, find the sum of those 2 fractions. Take the numerator and denominator values of the fractions from the user.

In [9]:

```

n1=int(input('Num1:'))
d1=int(input('Den1:'))
n2=int(input('Num2:'))
d2=int(input('Den2:'))
rn=n1*d2+n2*d1
rd=d1*d2
print('{} / {}'.format(rn, rd))

```

```

Num1:13
Den1:17
Num2:22
Den2:211
3117/3587

```

10. Given the height, width and breadth of a milk tank, you have to find out how many glasses of milk can be obtained? Assume all the inputs are provided by the user.

In [10]:

```

import math

h_t = float(input('Height:'))
b_t = float(input('Breadth:'))
l_t = float(input('Length'))

h_g = float(input('Height of glass'))
r_g = float(input('Radius of the glass'))

vol_tank = h_t*b_t*l_t
vol_glass = 3.14*r_g*r_g*h_g

print('no of glasses', math.floor(vol_tank/vol_glass))

```

```

Height:10
Breadth:19
Length10
Height of glass5
Radius of the glass2
no of glasses 30

```

In []:

```


```

TASK AND SOLUTION (DAY-02)

1. Write a program that will give you in hand monthly salary after deduction on CTC - HRA(10%), DA(5%), PF(3%) and taxes deduction as below: Salary(Lakhs) : Tax(%) Below 5 : 0% 5-10 : 10% 10-20 : 20% above 20 : 30%

In [1]:

```
ctc = int(input('Enter your anual CTC:'))
if ctc < 5000000:
    salary=ctc*.82
elif ctc<10000000:
    salary=ctc*.72
elif ctc<20000000:
    salary=ctc*.62
else:
    salary=ctc*.52
print("You in hand monthly salary will be-", round(salary/12,2))
```

```
Enter your anual CTC:1200000
You in hand monthly salary will be- 82000.0
```

2. write e a program that take a user input of three angles and will find out whether it can form a triangle or not.

In [2]:

```
first = int(input('Enter the 1st angle:'))
second = int(input('Enter the 2nd angle:'))
third = int(input('Enter the 3rd angle:'))
if (first+second+third) == 180 and first>0 and second>0 and third>0:
    print('forms a triangle')
else:
    print('Does not form a triangle')
```

```
Enter the 1st angle:30
Enter the 2nd angle:30
Enter the 3rd angle:90
Does not form a triangle
```

3. Write a program that will take user input of cost price and selling price and determines whether its a loss or a profit.

In [3]:

```
cost_price = int(input('Enter cost price:'))
selling_price = int(input('Enter selling price:'))
if cost_price < selling_price:
    print('Profit')
elif cost_price > selling_price:
    print('Loss')
else:
    print('No Loss No Gain')
```

```
Enter cost price:120
Enter selling price:210
Profit
```

4. Write a menu-driven program - 1.cm to ft 2.km to miles 3.USD to INR 4.exit Hint 1 cm = 0.032ft 1km = 0.62 1USD = 80 INR

In [4]:

```
menu = input("""
Hi select an option
1. cms to ft
2. km to miles
3. USD to INR
4. Exit
""")
if menu == '1':
    cm = float(input('Enter the cm value'))
    print('ft value is', 0.032*cm)
elif menu == '2':
    km = float(input('Enter the km value'))
    print('miles value is', km*0.62)
elif menu == '3':
    usd = float(input('Enter usd'))
    print('inr', usd*80)
else:
    exit()
```

```
Hi select an option
1. cms to ft
2. km to miles
3. USD to INR
4. Exit
1
Enter the cm value12
ft value is 0.384
```

5.Display Fibonacci series up to 10 terms.Note: The Fibonacci Sequence is a series of numbers.

The next number is found by adding up the two numbers before it. The first two numbers are 0 and 1. For example, 0, 1, 1, 2, 3, 5, 8, 13, 21. The next number in this series above is **13+21 = 34**

In [5]:

```
num1,num2 = 0,1
for i in range(10):
    print(num1)
    next = num1 + num2
    num1 = num2
    num2 = next
```

```
0
1
1
2
3
5
8
13
21
34
```

6.Find the factorial of a given number.Write a program to use the loop to find the factorial of a given number. The factorial (symbol: !) means to multiply all whole numbers from the chosen number down to 1.

For example: calculate the factorial of 5 $5! = 5 \times 4 \times 3 \times 2 \times 1 = 120$ o/p=120

In [6]:

```
num = int(input('Enter the number:'))
fact = 1
for i in range(1,num+1):
```

```
fact = fact*i
print(fact)
```

Enter the number:5
120

7.Reverse a given integer number. Example: Input: 76542 Output:24567

In [7]:

```
number = int(input('Enter the number:'))
rev = 0
while number>0:
    last = number%10
    rev = rev*10 + last
    number = number//10
print(rev)
```

Enter the number:123456
654321

8.Take a user input as integer N. Find out the sum from 1 to N. If any number if divisible by 5, then skip that number.And if the sum is greater than 300, don't need to calculate the sum further more. Print the final result. And don't use for loop to solve this problem.

In [8]:

```
N = int(input('Enter the number:'))
sum = 0
i = 1
while i < N+1:
    if i % 5 == 0:
        i+=1
        continue

    sum += i

    if sum > 300:
        sum = sum - i
        break

    i+=1
print(sum)
```

Enter the number:50
276

9.Write a program that keeps on accepting a number from the user until the user enters Zero. Display the sum and average of all the numbers.

In [9]:

```
sum = 0
count = 0
while True:
    num = int(input('Enter number:'))
    if num == 0:
        break
    sum = sum + num
    count = count + 1
print('sum', sum)
print('avg', sum/count)
```

Enter number:1
Enter number:2
Enter number:3
Enter number:4
Enter number:5

```
Enter number:6
Enter number:7
Enter number:0
sum 28
avg 4.0
```

10. Write a program which will find all such numbers which are divisible by 7 but are not a multiple of 5, between 2000 and 3200 (both included). The numbers obtained should be printed in a comma-separated sequence on a single line.

In [10]:

```
L = []
for i in range(2000,3201):
    if i % 7 == 0 and i % 5 != 0:
        L.append(str(i))
print(", ".join(L))
```

```
2002,2009,2016,2023,2037,2044,2051,2058,2072,2079,2086,2093,2107,2114,2121,2128,2142,2149
,2156,2163,2177,2184,2191,2198,2212,2219,2226,2233,2247,2254,2261,2268,2282,2289,2296,230
3,2317,2324,2331,2338,2352,2359,2366,2373,2387,2394,2401,2408,2422,2429,2436,2443,2457,24
64,2471,2478,2492,2499,2506,2513,2527,2534,2541,2548,2562,2569,2576,2583,2597,2604,2611,2
618,2632,2639,2646,2653,2667,2674,2681,2688,2702,2709,2716,2723,2737,2744,2751,2758,2772,
2779,2786,2793,2807,2814,2821,2828,2842,2849,2856,2863,2877,2884,2891,2898,2912,2919,2926
,2933,2947,2954,2961,2968,2982,2989,2996,3003,3017,3024,3031,3038,3052,3059,3066,3073,308
7,3094,3101,3108,3122,3129,3136,3143,3157,3164,3171,3178,3192,3199
```

11: A robot moves in a plane starting from the original point (0,0). The robot can move toward UP, DOWN, LEFT and RIGHT with a given steps. The trace of robot movement is shown as the following: UP 5 DOWN 3 LEFT 3 RIGHT 2 !

In [11]:

```
pos = [0,0]
while True:
    s = input('Enter the robot path')
    if s == '!':
        break
    direction = s.split()[0]
    steps = int(s.split()[1])

    if direction == 'UP':
        pos[1] = pos[1] + steps
    elif direction == 'DOWN':
        pos[1] = pos[1] - steps
    elif direction == 'LEFT':
        pos[0] = pos[0] - steps
    elif direction == 'RIGHT':
        pos[0] = pos[0] + steps
    else:
        pass
print('new pos',pos)
print((pos[0]**2 + pos[1]**2)**0.5)
```

```
Enter the robot path!
new pos [0, 0]
0.0
```

12. Write a program to print whether a given number is a prime number or not.

In [12]:

```
num = int(input('Enter the num:'))

flag = True
for i in range(2,num):
    if num%i == 0:
```

```

        flag = False
        break

if flag == True:
    print('Prime')
else:
    print('Not Prime')

```

Enter the num:21
Not Prime

13:Print all the Armstrong numbers in a given range. Range will be provided by the user
Armstrong number is a number that is equal to the sum of cubes of its digits. For example
0, 1, 153, 370, 371 and 407 are the Armstrong numbers.

In [13]:

```

start = int(input("Enter the start of the range: "))
end = int(input("Enter the end of the range: "))

for num in range(start, end+1):
    temp = num
    sum = 0
    order = len(str(num))
    while temp > 0:
        digit = temp % 10
        sum += digit ** order
        temp //= 10
    if num == sum:
        print(num)

```

Enter the start of the range: 1
Enter the end of the range: 100
1
2
3
4
5
6
7
8
9

14. Calculate the angle between the hour hand and minute hand.

Note: There can be two angles between hands; we need to print a minimum of two. Also, we need to print the floor of the final result angle. For example, if the final angle is 10.61, we need to print 10. Input: H = 9 , M = 0 Output: 90

In [14]:

```

h = int(input('Enter hours hand:'))
m = int(input('Enter minute hand:'))
# validate the input
if (h < 0 or m < 0 or h > 12 or m > 60):
    print('Wrong input')
# Idea is to minute angle - hour agnle from clockwise from 12 hour point
# 1 minute in minute angle manke 6 degree. (60 minute -> 360 degree)
m_angle = m*6
# every hour point yeilds to 30degree-- 12 hours-360degree plus if minute hand moves hour
hands move too
# Every minute after hour hand take 0.5 degree movement. clockwise
h_angle = h*30 + m*0.5
# Take abs difference b/w them
angle = abs(h_angle - m_angle)
if angle>180:
    print(360-angle)
else:

```



```
print(angle)
```

```
Enter hours hand:9  
Enter minute hand:0  
90.0
```

15. Given two rectangles, find if the given two rectangles overlap or not. A rectangle is denoted by providing the x and y coordinates of two points: the left top corner and the right bottom corner of the rectangle. Two rectangles sharing a side are considered overlapping. (L1 and R1 are the extreme points of the first rectangle and L2 and R2 are the extreme points of the second rectangle).

Note: It may be assumed that the rectangles are parallel to the coordinate axis.

In [15]:

```
print("Enter both Rectangel points coordinates <x y>")  
l1_x, l1_y = list(map(int, input('Enter L1 coordinate value (Two value space seperated): '  
' ).split()))  
r1_x, r1_y = list(map(int, input('Enter r1 coordinate value, (Two value space seperated): '  
) .split()))  
  
l2_x, l2_y = list(map(int, input('Enter L2 coordinate value, (Two value space seperated): '  
) .split()))  
r2_x, r2_y = list(map(int, input('Enter r2 coordinate value, (Two value space seperated): '  
) .split()))  
  
# if rectangle has area 0, no overlap  
if l1_x == r1_x or l1_y == r1_y or r2_x == l2_x or l2_y == r2_y:  
    print("Don't Overlap")  
    # If one rectangle is on left side of other  
elif l1_x > r2_x or l2_x > r1_x:  
    print("Don't Overlap")  
  
# If one rectangle is above other  
elif r1_y > l2_y or r2_y > l1_y:  
    print("Don't Overlap")  
else:  
    print("Overlap")
```

```
Enter both Rectangel points coordinates <x y>  
Enter L1 coordinate value (Two value space seperated):5 8  
Enter r1 coordinate value, (Two value space seperated):8 7  
Enter L2 coordinate value, (Two value space seperated):4 6  
Enter r2 coordinate value, (Two value space seperated):7 9  
Don't Overlap
```

In []:

01. Print the following pattern. Write a program to use for loop to print the following reverse number pattern.

5 4 3 2 1

4 3 2 1

3 2 1

2 1

1

In [1]:

```
rows = int(input('enter the rows:'))
for i in range(0, rows):
    for j in range(rows-i, 0, -1):
        print(j, end=' ')
    print()
```

enter the rows:2

2 1

1

02.Print the following pattern.

In [2]:

```
rows = int(input('enter the rows:'))
for i in range(1, rows+1):
    for j in range(0, i):
        print('*', end=' ')
    print()
for i in range(1, rows):
    for j in range(rows-i, 0, -1):
        print('*', end=' ')
    print()
```

enter the rows:9

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

03. Write a program to print the following pattern

*

*

In [3]:

```
rows = 6
for i in range(1, rows+1):
    print(' '*rows, end='')
    print('*'*i)
    rows = rows - 1
```

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

04. Write a program to print the following pattern 1 2 1 3 2 1 4 3 2 1 5 4 3 2 1

In [4]:

```
rows = 5
for i in range(1, rows+1):
    for j in range(i, 0, -1):
        print(j, end=' ')
    print()
```

```
1
2 1
3 2 1
4 3 2 1
5 4 3 2 1
```

05. Write a Python Program to Find the Sum of the Series till the nth term: $1 + x^2/2 + x^3/3 + \dots + x^n/n$ n will be provided by the user.

In [5]:

```
x = 10
n = 5
sum = 1
s = ''
```

```
print('1 + ',end='')
for i in range(2,n+1):
    sum = sum + x**i/i
    s = s + 'x^{}/{} +'.format(i,i)
print(s[:-1])
print(sum)
```

1 + x^2/2 +x^3/3 +x^4/4 +x^5/5
22884.33333333332

06.The natural logarithm can be approximated by the following series

$(1/1)((x-1)/x)^1 + (1/2)((x-1)/x)^2 + (1/3)((x-1)/x)^3 + (1/4)((x-1)/x)^4 + (1/5)((x-1)/x)^5 + \dots$
 $(1/n)((x-1)/x)^n$

If x is input through the keyboard, write a program to calculate the sum of the first seven terms of this series.

In [6]:

```
x = 10
n = 5
sum = 0
s = ''
for i in range(1,n+1):
    sum = sum + (1/i)*((x-1)/x)**i
    s = s + '(1/{})*((x-1)/x)^{} +'.format(i,i)
print(s[:-1])
print(sum)
```

$(1/1)*((x-1)/x)^1 + (1/2)*((x-1)/x)^2 + (1/3)*((x-1)/x)^3 + (1/4)*((x-1)/x)^4 + (1/5)*((x-1)/x)^5$
1.8301230000000004

07.Write a program to calculate the sum of series up to n term. Take the user input and then calculate. And the output style should match which is given in the example. Input:5 Output:2+22+222+2222+22222 Sum of above series is: 24690

In [7]:

```
n = int(input('enter the number of terms'))
start = 2
sum = 0
for i in range(0,n):
    if i < n-1:
        print(start,end='+')
    else:
        print(start)
    sum = sum + start
    start = start*10 + 2
print(sum)
```

enter the number of terms8
2+22+222+2222+22222+222222+2222222+22222222+222222222
24691356

08.Write a program to print all the unique combinations of 1,2,3 and 4.

In [8]:

```
for i in range(1,5):
    for j in range(1,5):
        for k in range(1,5):
            for m in range(1,5):
                print(i,j,k,m)
```

1 1 1 1
1 1 1 2
1 1 1 3
1 1 1 4
1 1 2 1
1 1 2 2
1 1 2 3
1 1 2 4
1 1 3 1
1 1 3 2
1 1 3 3
1 1 3 4
1 1 4 1
1 1 4 2
1 1 4 3
1 1 4 4
1 2 1 1
1 2 1 2
1 2 1 3
1 2 1 4
1 2 2 1
1 2 2 2
1 2 2 3
1 2 2 4
1 2 3 1
1 2 3 2
1 2 3 3
1 2 3 4
1 2 4 1
1 2 4 2
1 2 4 3
1 2 4 4
1 3 1 1
1 3 1 2
1 3 1 3
1 3 1 4
1 3 2 1
1 3 2 2
1 3 2 3
1 3 2 4
1 3 3 1
1 3 3 2
1 3 3 3
1 3 3 4
1 3 4 1
1 3 4 2
1 3 4 3
1 3 4 4
1 4 1 1
1 4 1 2
1 4 1 3
1 4 1 4
1 4 2 1
1 4 2 2
1 4 2 3
1 4 2 4
1 4 3 1
1 4 3 2
1 4 3 3
1 4 3 4
1 4 4 1
1 4 4 2
1 4 4 3
1 4 4 4
2 1 1 1
2 1 1 2
2 1 1 3
2 1 1 4
2 1 2 1
2 1 2 2
2 1 2 3
2 1 2 4
2 1 2 4
2 1 2 1

2 1 3 1
2 1 3 2
2 1 3 3
2 1 3 4
2 1 4 1
2 1 4 2
2 1 4 3
2 1 4 4
2 2 1 1
2 2 1 2
2 2 1 3
2 2 1 4
2 2 2 1
2 2 2 2
2 2 2 3
2 2 2 4
2 2 3 1
2 2 3 2
2 2 3 3
2 2 3 4
2 2 4 1
2 2 4 2
2 2 4 3
2 2 4 4
2 3 1 1
2 3 1 2
2 3 1 3
2 3 1 4
2 3 2 1
2 3 2 2
2 3 2 3
2 3 2 4
2 3 3 1
2 3 3 2
2 3 3 3
2 3 3 4
2 3 4 1
2 3 4 2
2 3 4 3
2 3 4 4
2 4 1 1
2 4 1 2
2 4 1 3
2 4 1 4
2 4 2 1
2 4 2 2
2 4 2 3
2 4 2 4
2 4 3 1
2 4 3 2
2 4 3 3
2 4 3 4
2 4 4 1
2 4 4 2
2 4 4 3
2 4 4 4
3 1 1 1
3 1 1 2
3 1 1 3
3 1 1 4
3 1 2 1
3 1 2 2
3 1 2 3
3 1 2 4
3 1 3 1
3 1 3 2
3 1 3 3
3 1 3 4
3 1 4 1
3 1 4 2
3 1 4 3
3 1 4 4
2 2 1 1

3 2 1 1
3 2 1 2
3 2 1 3
3 2 1 4
3 2 2 1
3 2 2 2
3 2 2 3
3 2 2 4
3 2 3 1
3 2 3 2
3 2 3 3
3 2 3 4
3 2 4 1
3 2 4 2
3 2 4 3
3 2 4 4
3 3 1 1
3 3 1 2
3 3 1 3
3 3 1 4
3 3 2 1
3 3 2 2
3 3 2 3
3 3 2 4
3 3 3 1
3 3 3 2
3 3 3 3
3 3 3 4
3 3 4 1
3 3 4 2
3 3 4 3
3 3 4 4
3 4 1 1
3 4 1 2
3 4 1 3
3 4 1 4
3 4 2 1
3 4 2 2
3 4 2 3
3 4 2 4
3 4 3 1
3 4 3 2
3 4 3 3
3 4 3 4
3 4 4 1
3 4 4 2
3 4 4 3
3 4 4 4
4 1 1 1
4 1 1 2
4 1 1 3
4 1 1 4
4 1 2 1
4 1 2 2
4 1 2 3
4 1 2 4
4 1 3 1
4 1 3 2
4 1 3 3
4 1 3 4
4 1 4 1
4 1 4 2
4 1 4 3
4 1 4 4
4 2 1 1
4 2 1 2
4 2 1 3
4 2 1 4
4 2 2 1
4 2 2 2
4 2 2 3
4 2 2 4
4 2 2 4
4 2 2 1

4 2 3 1
4 2 3 2
4 2 3 3
4 2 3 4
4 2 4 1
4 2 4 2
4 2 4 3
4 2 4 4
4 3 1 1
4 3 1 2
4 3 1 3
4 3 1 4
4 3 2 1
4 3 2 2
4 3 2 3
4 3 2 4
4 3 3 1
4 3 3 2
4 3 3 3
4 3 3 4
4 3 4 1
4 3 4 2
4 3 4 3
4 3 4 4
4 4 1 1
4 4 1 2
4 4 1 3
4 4 1 4
4 4 2 1
4 4 2 2
4 4 2 3
4 4 2 4
4 4 3 1
4 4 3 2
4 4 3 3
4 4 3 4
4 4 4 1
4 4 4 2
4 4 4 3
4 4 4 4

09. Write a program that will take a decimal number as input and prints out the binary equivalent of the number.

In [9]:

```
n = int(input('enter the number'))
binary = []
while n > 0:
    binary.append(n%2)
    n = n//2
for i in binary[::-1]:
    print(i,end='')
```

enter the number8
1000

10. Write a program that will take 2 numbers as input and prints the LCM and HCF of those 2 numbers.

In [11]:

```
x = int(input('enter 1st number'))
y = int(input('enter 2nd number'))
if x>y:
    greater = x
else:
    greater = y
while True:
```



```

    if (greater % x == 0) and (greater % y == 0):
        lcm = greater
        break

```

```

    greater = greater + 1
print(lcm)

```

```

enter 1st number9
enter 2nd number8
72

```

In [12]:

```

x = int(input('enter 1st number:'))
y = int(input('enter 2nd number:'))
if x<y:
    smaller = x
else:
    smaller = y
for i in range(1,smaller+1):
    if (x % i == 0) and (y % i == 0):
        hcf = i
print(hcf)

```

```

enter 1st number:7
enter 2nd number:8
1

```

11.Create Short Form from initial character.Given a string create short form ofthe string from Initial character.

Short form should be capitalised. Example: Input:Data science mentorship program
Output:DSMP

In [13]:

```

inp = 'Data science Mentorship Program'
res = ''
for i in inp.split():
    res = res + i[0].upper()
print(res)

```

```

DSMP

```

12.Append second string in the middle of first string

Input:campusx

data

Output:camdatapusx

In [14]:

```

s1 = input('enter the 1st string')
s2 = input('enter the 2nd string')
print(s1[0:int(len(s1)/2)] + s2 + s1[int(len(s1)/2):])

```

```

enter the 1st stringcampusx
enter the 2nd stringdata
camdatapusx

```

13.Given string contains a combination of the lower and upper case letters. Write a program to arrange the characters of a string so that all lowercase letters should come first.

Given: str1 = PyNaTive Expected Output: yaivePNT

In [15]:

```
s = 'PyNaTive'
upper = ''
lower = ''
for i in s:
    if i.islower():
        lower = lower + i
    else:
        upper = upper + i
print(lower + upper)
```

yaivePNT

14. Take a alphanumeric string input and print the sum and average of the digits that appear in the string, ignoring all other characters.

Input: hel123O4every093 Output: Sum: 22 Avg: 2.75 count:7

In [16]:

```
s = 'hel123O4every093'
sum = 0
count = 0
for i in s:
    if i.isdigit():
        sum = sum + int(i)
        count += 1
print(sum)
print(sum/count)
print(count)
```

22
3.142857142857143
7

15. Removal of all characters from a string except integers Given: str1 = 'I am 25 years and 10 months old' Expected Output: 2510

In [17]:

```
s = 'I am 25 years and 10 months old'
res = ''
for i in s:
    if i.isdigit():
        res = res + i
print(res)
```

2510

16. Check whether the string is Symmetrical.

Statement: Given a string. the task is to check if the string is symmetrical or not. A string is said to be symmetrical if both the halves of the string are the same. Example 1:

Input: khokho Output: The entered string is symmetrical

In [18]:

```
s = input('enter the string')
if len(s)%2 == 0:
    s1 = s[0:len(s)//2]
    s2 = s[len(s)//2:]
```

```

else:
    s1 = s[0:len(s)//2]
    s2 = s[len(s)//2 + 1:]
if s1 == s2:
    print('symmetrical')
else:
    print('not symmetrical')

```

enter the stringkokokololki
not symmetrical

17.Reverse words in a given String.Statement: We are given a string and we need to reverse words of a given string.

Example:Input: geeks quiz practice code Output: code practice quiz geeks

In [19]:

```

# Code here
s = 'code practice quiz geeks'
L = []
for i in s.split():
    L.append(i)
L = L[::-1]
print(" ".join(L))

```

geeks quiz practice code

18.Find uncommon words from two Strings.

Statement: Given two sentences as strings A and B. The task is to return a list of all uncommon words. A word is uncommon if it appears exactly once in any one of the sentences, and does not appear in the other sentence. Note: A sentence is a string of space-separated words. Each word consists only of lowercase letters.

In [18]:

```

A = "apple banana mango"
B = "banana fruits mango"
L = []
for i in A.split():
    if i not in B and i not in L:
        L.append(i)
for j in B.split():
    if j not in A and j not in L:
        L.append(j)
print(L)

```

['apple', 'fruits']

19.Word location in String.

Statement: Find a location of a word in a given sentence.Example 1: Input:Sentence: We can learn data science through campusx mentorship program. word: campusx Output:Location of the word is 7. Note- Don't use index/find functions

In [17]:

```

s = 'We can learn data science through campusx mentorship program.'
word = 'campusx'
pos = 0
for i in s.split():
    pos += 1
    if i == word:
        break

```

```
print(pos)
```

7

20. Write a program that can remove all the duplicate characters from a string. User will provide the input.

In [16]:

```
s = 'aaaabbbbccccdddeeeffff'
res = ''
for i in s:
    if i not in res:
        res = res + i
print(res)
```

abcdef

In []: