

In [ ]: Program 1  
Write a Python program to print "Hello Python".

```
In [1]: print("Hello Python")
```

Hello Python

In [ ]: Program 2  
Write a Python program to do arithmetical operations addition **and** division.

In [10]: *# Addition*  
`num1 = float(input("Enter the first number for addition: "))`  
`num2 = float(input("Enter the second number for addition: "))`  
`sum_result = num1 + num2`  
`print(f"sum: {num1} + {num2} = {sum_result}")`

sum: 7.0 + 8.0 = 15.0

In [4]: `print("enter two numerical values")`  
`x=input()`  
`y=input()`  
  
`a=int(x)`  
`b=int(y)`  
  
`c=a*b`  
  
`print("-"*50)`  
`print("product({},{}]={})".format(a,b,c))`

enter two numerical values

---

product(1,2)=2

In [ ]: Program 3  
Write a Python program to find the area of a triangle.

In [8]: `base=float(input("enter the length of the base of the triangle:"))`  
`height=float(input("enter the height of the triangle:"))`  
  
`Area_of_triangle=0.5*base*height`  
  
`print("the area of triangle is :",Area_of_triangle)`

the area of triangle is : 24.5

In [ ]: Program 4  
Write a Python program to swap two variables.

In [14]: `a=input("enter the first variable(a):")`  
`b=input("enter the second variable(b):")`  
  
`print(f"original values:a={a}, b={b}")`  
  
`x=a`  
`a=b`  
`b=x`

```
print(f"swap the variables: a={a}, b={b}")
```

original values:a=2, b=3  
swap the variables: a=3, b=2

In [ ]: Program 5

Write a Python program to generate a random number.

In [12]: `import random`

```
print(f"Random number: {random.randint(1, 100)}")
```

Random number: 43

In [ ]: Program 6

Write a Python program to convert kilometers to miles.

In [16]: `km=float(input("enter the distance in km:"))`

```
conversion_factor=0.621371
```

```
miles=km*conversion_factor
```

```
print(f"{km} km is equal to {miles} miles")
```

2.0 km is equal to 1.242742 miles

In [ ]: Program 7

Write a Python program to convert Celsius to Fahrenheit.

In [18]: `celsius = float(input("Enter temperature in Celsius: "))`

```
fahrenheit = (celsius * 9/5) + 32
```

```
print(f"{celsius} degrees Celsius is equal to {fahrenheit} degrees Fahrenheit.")
```

9.0 degrees Celsius is equal to 48.2 degrees Fahrenheit.

In [ ]: Program 8

Write a Python program to display calendar.

In [23]: `import calendar`

```
year = int(input("Enter year: "))
```

```
month = int(input("Enter month: "))
```

```
cal = calendar.month(year, month)
```

```
print(cal)
```

June 2002

Mo	Tu	We	Th	Fr	Sa	Su
----	----	----	----	----	----	----

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
----	----	----	----	----	----	----

In [ ]: Program 9

Write a Python program to solve quadratic equation.

The standard form of a quadratic equation is:

where

```
a, b and c are real numbers and

The solutions of this quadratic equation is given by:
ax + bx + c = 0
2

a ≠ 0

(-b ± (b - 4ac))/(2a)
```

```
In [25]: import math
a = float(input("Enter coefficient a: "))
b = float(input("Enter coefficient b: "))
c = float(input("Enter coefficient c: "))

discriminant = b**2 - 4*a*c

if discriminant > 0:

    root1 = (-b + math.sqrt(discriminant)) / (2*a)
    root2 = (-b - math.sqrt(discriminant)) / (2*a)
    print(f"Root 1: {root1}")
    print(f"Root 2: {root2}")
elif discriminant == 0:

    root = -b / (2*a)
    print(f"Root: {root}")
else:

    real_part = -b / (2*a)
    imaginary_part = math.sqrt(abs(discriminant)) / (2*a)
    print(f"Root 1: {real_part} + {imaginary_part}i")
    print(f"Root 2: {real_part} - {imaginary_part}i")
```

Root 1: -0.35714285714285715 + 0.8541614816500999i  
Root 2: -0.35714285714285715 - 0.8541614816500999i

```
In [ ]: Program 10
Write a Python program to swap two variables without temp variable.
```

```
In [26]: a = 5
b = 10

a, b = b, a

print("After swapping:")
print("a =", a)
print("b =", b)
```

After swapping:  
a = 10  
b = 5

```
In [ ]: Program 11
Write a Python Program to Check if a Number is Positive, Negative
```

```
In [4]: num=float(input("enter a number:"))
if num>0:
    print("the num is positive:")
elif num == 0:
    print("zero")
```

```
else:
    print("the num is negative")
```

the num is positive:

In [ ]: Program 12  
Write a Python Program to Check if a Number is Odd or Even.

```
num=int(input("enter a number"))

if num%2 ==0:
    print("this is a even number")
else:
    print("this is odd")
```

this is a even number

In [ ]: Program 13  
Write a Python Program to Check Leap Year.

```
year=int(input("enter a year"))
if year%4==0 and year%100!=0 or year%400==0:
    print("leap year")
else:
    print("not leap year")
```

not leap year

In [ ]: Program 14  
Write a Python Program to Check Prime Number.

```
num = int(input("Enter a number: "))

flag = False

if num == 1:
    print(f"{num}, is not a prime number")
elif num > 1:
    for i in range(2, num):
        if (num % i) == 0:
            flag = True

if flag:
    print(f"{num}, is not a prime number")
else:
    print(f"{num}, is a prime number")
```

1, is not a prime number

1, is a prime number

In [ ]: Program 15  
Write a Python Program to Print all Prime Numbers in an Interval of 1-10.

```
lower = 1
upper = 10
print("Prime numbers between", lower, "and", upper, "are:")
for num in range(lower, upper + 1):
    if num > 1:
```

```

for i in range(2, num):
    if (num % i) == 0:
        break
else:
    print(num)

```

Prime numbers between 1 and 10 are:

2  
3  
5  
7

In [ ]: Program 16

Write a Python Program to Find the Factorial of a Number.

In [23]: num= int(input("enter a number"))

```

factorial=1
if num<0:
    print("Factirial does not exist for negative numbers")
elif num==0:
    print("Factorial of 0 is 1")
else:
    for i in range(1,num+1):
        factorial = factorial*i
    print(f'The factorial of {num} is {factorial}')

```

The factorial of 45 is 11962222086548019456196316149565771506438373376000000000000

In [ ]: Program 17

Write a Python Program to Display the multiplication Table.

In [27]: num = int(input("Display multiplication table of: "))

```

for i in range(1, 11):
    print(f'{num} X {i} = {num*i}')

```

18 X 1 = 18  
18 X 2 = 36  
18 X 3 = 54  
18 X 4 = 72  
18 X 5 = 90  
18 X 6 = 108  
18 X 7 = 126  
18 X 8 = 144  
18 X 9 = 162  
18 X 10 = 180

In [ ]: Program 18

Write a Python Program to Print the Fibonacci sequence.

In [29]: nterms = int(input("How many terms? "))

```

n1, n2 = 0, 1
count = 0

if nterms <= 0:
    print("Please enter a positive integer")

elif nterms == 1:
    print("Fibonacci sequence upto",nterms,:")
    print(n1)

```

```

else:
    print("Fibonacci sequence:")
    while count < nterms:
        print(n1)
        nth = n1 + n2
        n1 = n2
        n2 = nth
        count += 1

```

Fibonacci sequence:

```

0
1
1
2
3
5
8
13
21
34
55
89
144
233
377
610
987
1597
2584
4181
6765
10946
17711

```

In [ ]: Program 19  
Write a Python Program to Check Armstrong Number?

```

In [30]: num = int(input("Enter a number: "))

num_str = str(num)
num_digits = len(num_str)

sum_of_powers = 0
temp_num = num

while temp_num > 0:
    digit = temp_num % 10
    sum_of_powers += digit ** num_digits
    temp_num //= 10

if sum_of_powers == num:
    print(f"{num} is an Armstrong number.")
else:
    print(f"{num} is not an Armstrong number.")

```

1 is an Armstrong number.

In [ ]: Program 20  
Write a Python Program to Find Armstrong Number in an Interval.

```
In [33]: lower = int(input("Enter the lower limit of the interval: "))
upper = int(input("Enter the upper limit of the interval: "))

for num in range(lower, upper + 1):
    order = len(str(num))
    temp_num = num
    sum = 0
    while temp_num > 0:
        digit = temp_num % 10
        sum += digit ** order
        temp_num //= 10

    if num == sum:
        print(num)
```

153  
370  
371  
407

In [ ]: Program 21  
Write a Python Program to Find the Sum of Natural Numbers.

```
In [3]: limit=int(input("enter the limit:"))

sum=0

for i in range(1,limit+1):
    sum +=i

print("the Sum of Natural Numbers up to", limit, "is:",sum)
```

the Sum of Natural Numbers up to 10 is: 55

In [ ]: Program 22  
Write a Python Program to Find LCM.  
*#Least Common Multiple (LCM):*

```
In [5]: def compute_lcm(x,y):
    if x>y:
        greter=x
    else:
        greter=y

    while(True):
        if((greter % x == 0) and (greter % y == 0)):
            lcm=greter
            break
        greter += 1
    return lcm

num1=int(input("enter a number:"))
num2=int(input("enter a number:"))

print("the l.c.m is:", compute_lcm(num1,num2))
```

the l.c.m is: 216

In [ ]: Program 23  
Write a Python Program to Find HCF.

```
#Highest Common Factor(HCF):
```

```
In [20]: def compute_hcf(x,y):
    if x>y:
        smaller=y
    else:
        smaller=x
    for i in range(1,smaller+1):
        if((x % i == 0) and (y % i == 0)):
            hcf=i
    return hcf

num1=int(input("enter a number:"))
num2=int(input("enter a number:"))

print("The H.C.F is:",compute_hcf(num1,num2))
```

The H.C.F is: 6

```
In [ ]: Program 24
Write a Python Program to Convert Decimal to Binary, Octal and Hexadecimal.
```

```
In [23]: dct_num=int(input("enter the decimal number"))
```

```
print(" the decimal value of ",dct_num,"is:")
print("the conversion of decimal to binary is",bin(dct_num))
print("the conversion of decimal to octal is",oct(dct_num))
print("the conversion of decimal to hexadecimal is",hex(dct_num))
```

the decimal value of 54 is:  
the conversion of decimal to binary is 0b110110  
the conversion of decimal to octal is 0o66  
the conversion of decimal to hexadecimal is 0x36

```
In [ ]: Program 25
Write a Python Program To Find ASCII value of a character.
```

```
In [27]: char = str(input("Enter the character: "))
print("The ASCII value of '" + char + "' is", ord(char))
```

The ASCII value of 'p' is 112

```
In [28]: char = input("Enter a character: ")

ascii_value = ord(char)

print(f"The ASCII value of '{char}' is {ascii_value}")
```

The ASCII value of 'p' is 112

```
In [ ]: Program 26
Write a Python Program to Make a Simple Calculator with 4 basic mathematical operations.
```

```
In [29]: # Simple Calculator in Python
```

```
def add(x, y):
    return x + y
```

```

def subtract(x, y):
    return x - y

def multiply(x, y):
    return x * y

def divide(x, y):
    if y == 0:
        return "Error! Division by zero is not allowed."
    return x / y

print("Select operation:")
print("1. Add")
print("2. Subtract")
print("3. Multiply")
print("4. Divide")

choice = input("Enter choice (1/2/3/4): ")

num1 = float(input("Enter first number: "))
num2 = float(input("Enter second number: "))

if choice == '1':
    print(f"The result is: {add(num1, num2)}")
elif choice == '2':
    print(f"The result is: {subtract(num1, num2)}")
elif choice == '3':
    print(f"The result is: {multiply(num1, num2)}")
elif choice == '4':
    print(f"The result is: {divide(num1, num2)}")
else:
    print("Invalid input! Please select 1/2/3/4.")

```

Select operation:  
 1. Add  
 2. Subtract  
 3. Multiply  
 4. Divide  
 The result is: 7.666666666666667

In [ ]: Program 27  
 Write a Python Program to Display Fibonacci Sequence Using Recursion.

In [35]: def recur\_fibo(n):
 if n <= 1:
 return n
 else:
 return(recur\_fibo(n-1) + recur\_fibo(n-2))
nterms = int(input("Enter the number of terms (greater than 0): "))
# check if the number of terms is valid
if nterms <= 0:
 print("Please enter a positive integer")
else:
 print("Fibonacci sequence:")
for i in range(nterms):
 print(recur\_fibo(i))

Fibonacci sequence:

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

In [ ]: Program 28

Write a Python Program to Find Factorial of Number Using Recursion.

In [38]: `def recur_factorial(n):`

```
    if n == 1:
        return n
    else:
        return n*recur_factorial(n-1)
num = int(input("Enter the number: "))
# check if the number is negative
if num < 0:
    print("Sorry, factorial does not exist for negative numbers")
elif num == 0:
    print("The factorial of 0 is 1")
else:
    print("The factorial of", num, "is", recur_factorial(num))
```

The factorial of 7 is 5040

In [ ]: Program 29

Write a Python Program to calculate your Body Mass Index.

In [39]: `def bodymassindex(height, weight):`

```
    return round((weight / height**2),2)

h = float(input("Enter your height in meters: "))
w = float(input("Enter your weight in kg: "))

print("Welcome to the BMI calculator.")
bmi = bodymassindex(h, w)
print("Your BMI is: ", bmi)

if bmi <= 18.5:
    print("You are underweight.")
elif 18.5 < bmi <= 24.9:
    print("Your weight is normal.")
elif 25 < bmi <= 29.29:
    print("You are overweight.")
else:
    print("You are obese.")
```

Welcome to the BMI calculator.

Your BMI is: 0.1

You are underweight.

In [ ]: Program 30

Write a Python Program to calculate the natural logarithm of any number.

In [ ]: `import math`

```
num = float(input("Enter a number: "))
```

```

if num <= 0:
    print("Please enter a positive number.")
else:
    # Calculate the natural logarithm (base e) of the number
    result = math.log(num)
    print(f"The natural logarithm of {num} is: {result}")

```

In [ ]: Program 31  
Write a Python Program **for** cube sum of first n natural numbers?

```

In [15]: def cube_sum_of_natural_numbers(n):
    if n<=0:
        return 0
    else:
        total=sum([i**3 for i in range(1,n+1)])
        return total
n=int(input("enter the value of n:"))
if n<=0:
    print("please enter a positive integer")
else:
    result=cube_sum_of_natural_numbers(n)
    print(f"the cube sum of the first{n} natural number is:{result}")

```

the cube sum of the first7 natural number is:784

In [ ]: Program 32  
Write a Python Program to find sum of array.

```

In [17]: arr=[1,2,3]
ans=sum(arr)
print('sum of array is ',ans)

```

sum of array is 6

```

In [20]: def sum_of_array(arr):
    return sum(arr)

arr=[1,2,3]
print("array:",arr)

result=sum_of_array(arr)
print("sum of array:",result)

```

array: [1, 2, 3]  
sum of array: 6

In [ ]: Program 33  
Write a Python Program to find largest element **in** an array.

```

In [28]: def find_largest_element(arr):
    largest=arr[0]
    for num in arr:
        if num > largest:
            largest = num

```

```

    return largest

arr=[1,2,3]
print("array:",arr)
print("the largest element in array : ",find_largest_element(arr))

```

```

array: [1, 2, 3]
the largest element in array : 3

```

In [ ]: Program 34  
Write a Python Program **for** array rotation.

```

In [1]: # Function to rotate an array to the right by d positions
def rotate_array(arr, d):
    n = len(arr)
    d = d % n # In case d > n
    return arr[-d:] + arr[:-d]

# Example usage
arr = [1, 2, 3, 4, 5, 6, 7]
d = 3 # Number of rotations
rotated_arr = rotate_array(arr, d)

print("Original array:", arr)
print(f"Array after {d} right rotations:", rotated_arr)

```

```

Original array: [1, 2, 3, 4, 5, 6, 7]
Array after 3 right rotations: [5, 6, 7, 1, 2, 3, 4]

```

In [ ]: Program 35  
Write a Python Program to Split the array **and** add the first part to the end?

```

In [2]: def split_and_rotate(arr, split_index):
    return arr[split_index:] + arr[:split_index]

arr = [1, 2, 3, 4, 5, 6, 7]
split_index = 3
rotated_arr = split_and_rotate(arr, split_index)

print("Original array:", arr)
print(f"Array after splitting at index {split_index} and rotating:", rotated_arr)

```

```

Original array: [1, 2, 3, 4, 5, 6, 7]
Array after splitting at index 3 and rotating: [4, 5, 6, 7, 1, 2, 3]

```

```

In [3]: # Original array
arr = [1, 2, 3, 4, 5, 6, 7]

# Number of elements to move from start to end
n = 3

# Split and add first part to the end
arr = arr[n:] + arr[:n]

print(arr)

```

```
[4, 5, 6, 7, 1, 2, 3]
```

In [ ]: Program 36  
Write a Python Program to check if given array is Monotonic.

```
In [5]: def is_monotonic(arr):
    return all(arr[i] <= arr[i+1] for i in range(len(arr)-1)) or \
           all(arr[i] >= arr[i+1] for i in range(len(arr)-1))

arr1 = [1, 2, 2, 3, 4]
arr2 = [5, 4, 4, 2, 1]
arr3 = [1, 3, 2, 4]

print("arr1 is monotonic:", is_monotonic(arr1))
print("arr2 is monotonic:", is_monotonic(arr2))
print("arr3 is monotonic:", is_monotonic(arr3))
```

arr1 is monotonic: True  
arr2 is monotonic: True  
arr3 is monotonic: False

In [ ]: Program 37  
Write a Python Program to Add Two Matrices.

```
In [6]: A = [
    [1, 2, 3],
    [4, 5, 6],
    [7, 8, 9]
]

B = [
    [9, 8, 7],
    [6, 5, 4],
    [3, 2, 1]
]

result = [
    [0, 0, 0],
    [0, 0, 0],
    [0, 0, 0]
]

for i in range(len(A)):
    for j in range(len(A[0])):
        result[i][j] = A[i][j] + B[i][j]

# Print result
print("Sum of matrices:")
for row in result:
    print(row)
```

Sum of matrices:  
[10, 10, 10]  
[10, 10, 10]  
[10, 10, 10]

In [7]: #Program 38  
#Write a Python Program to Multiply Two Matrices.  
A = [
 [1, 2, 3],
 [4, 5, 6]

```
[1]

B = [
    [7, 8],
    [9, 10],
    [11, 12]
]

result = [[0 for _ in range(len(B[0]))] for _ in range(len(A))]

for i in range(len(A)):
    for j in range(len(B[0])):
        for k in range(len(B)):
            result[i][j] += A[i][k] * B[k][j]

print("Product of matrices:")
for row in result:
    print(row)
```

Product of matrices:

```
[58, 64]
[139, 154]
```

In [ ]: Program 39  
Write a Python Program to Transpose a Matrix.

```
In [8]: matrix = [
    [1, 2, 3],
    [4, 5, 6],
    [7, 8, 9]
]
transpose = [[0 for _ in range(len(matrix))] for _ in range(len(matrix[0]))]

for i in range(len(matrix)):
    for j in range(len(matrix[0])):
        transpose[j][i] = matrix[i][j]

print("Original Matrix:")
for row in matrix:
    print(row)

print("\nTranspose of Matrix:")
for row in transpose:
    print(row)
```

Original Matrix:

```
[1, 2, 3]
[4, 5, 6]
[7, 8, 9]
```

Transpose of Matrix:

```
[1, 4, 7]
[2, 5, 8]
[3, 6, 9]
```

In [ ]: Program 40  
Write a Python Program to Sort Words in Alphabetic Order.

In [9]: sentence = input("Enter a sentence: ")

```
words = sentence.split()

words.sort()

print("Words in alphabetical order:")
print(words)
```

Words in alphabetical order:  
['sdasf']

In [ ]: Program 41  
Write a Python Program to Remove Punctuation From a String.

```
punctuations = '''!()-[]{};:'"\,;<>./?@#$%^&*_~'''

my_str = input("Enter a string: ")

no_punct = ""
for char in my_str:
    if char not in punctuations:
        no_punct = no_punct + char

print(no_punct)
```

RTDFWWYGBFUFGNCB BNMGGMxghm

In [ ]: Program 43  
Write a Python program to check if the given number is a Disarium Number.

```
def is_disarium(num):

    num_str = str(num)
    length = len(num_str)

    total = sum(int(digit) ** (i + 1) for i, digit in enumerate(num_str))

    return total == num

number = int(input("Enter a number: "))

if is_disarium(number):
    print(f"{number} is a Disarium Number.")
else:
    print(f"{number} is not a Disarium Number.)
```

2123 is not a Disarium Number.

In [ ]: Program 44  
Write a Python program to print all disarium numbers between 1 to 100.

```
def is_disarium(num):

    digits = str(num)
    length = len(digits)

    total = sum(int(digits[i]) ** (i+1) for i in range(length))

    return total == num
```

```

print("Disarium numbers between 1 to 100 are:")
for i in range(1, 101):
    if is_disarium(i):
        print(i, end=" ")

```

In [ ]: Program 45

Write a Python program to check if the given number is Happy Number.

In [7]: `def is_happy(num):`

```

seen = set()

while num != 1 and num not in seen:
    seen.add(num)
    num = sum(int(digit) ** 2 for digit in str(num))

return num == 1

```

```
n = int(input("Enter a number: "))
```

```

if is_happy(n):
    print(n, "is a Happy Number :)")
else:
    print(n, "is NOT a Happy Number :( ")

```

```
32 is a Happy Number :)
```

In [ ]: Program 46

Write a Python program to print all happy numbers between 1 and 100.

In [3]: `def is_happy_number(num):`

```

seen = set()

while num != 1 and num not in seen:
    seen.add(num)
    num = sum(int(i) ** 2 for i in str(num))

return num == 1

```

```
happy_numbers = []
```

```

for num in range(1, 101):
    if is_happy_number(num):
        happy_numbers.append(num)

```

```

print("happy numbers between 1 and 100:")
print(happy_numbers)

```

```
happy numbers between 1 and 100:
```

```
[1, 7, 10, 13, 19, 23, 28, 31, 32, 44, 49, 68, 70, 79, 82, 86, 91, 94, 97, 100]
```

In [ ]: Program 47

Write a Python program to determine whether the given number is a Harshad Number.

In [7]: `def is_harshad_numbers(num):`

```

digit_sum = sum(int(i) for i in str(num))

return num % digit_sum == 0

```

```

num = int(input("enter a number:"))

if is_harshad_numbers(num):
    print(f"{num} is a Harshad Number.")
else:
    print(f"{num} is not Harshad Number.")

```

34 is not Harshad Number.

In [ ]: Program 48

Write a Python program to print all pronic numbers between 1 and 100.

In [9]:

```

def is_pronic_number(num):
    for n in range(1,int(num**0.5)+1):
        if n * (n+1) == num:
            return True
    return False

print("pronic numbers between 1 and 100 are:")
for i in range(1,100):
    if is_pronic_number(i):
        print(i,end=" / ")

```

pronic numbers between 1 and 100 are:

2 / 6 / 12 / 20 / 30 / 42 / 56 / 72 / 90 /

In [ ]: Program 49

In [11]: numbers=[10,20,30,40,50]

```

sum_of_numbers = 0

for i in numbers:
    sum_of_numbers+=i

print("sum of element in the list:",sum_of_numbers)

```

sum of element in the list: 150

In [ ]: Program 50

Write a Python program to Multiply all numbers in the list

In [13]: numbers = [10,20,30,40,50]

```

product_of_numbers = 1

for i in numbers:
    product_of_numbers *=i

print("product of element in the list:",product_of_numbers)

```

product of element in the list: 12000000

In [ ]: Program 51

Write a Python program to find smallest number in a list.

In [1]: numbers = [30,10,-45,5,20]

```
minimum = numbers[0]

for i in numbers:
    if i < minimum:
        minimum = i

print("The smallest number in the list is:", minimum)
```

The smallest number in the list is: -45

In [ ]: Program 52

Write a Python program to find largest number **in** a list.

In [2]: numbers = [30,10,-45,5,20]

```
minimum = numbers[0]

for i in numbers:
    if i > minimum:
        minimum = i

print("The largest number in the list is:", minimum)
```

The largest number in the list is: 30

In [ ]: Program 53

Write a Python program to find second largest number **in** a list.

In [4]: numbers = [30,10,45,5,20]

```
numbers.sort(reverse=True)

if len(numbers) >= 2:
    second_largest = numbers[1]
    print("The largest number in the list is:", second_largest)
else:
    print("The list does not contain a second largest number.")
```

The largest number in the list is: 30

In [ ]: Program 54

Write a Python program to find N largest elements **from** a list.

In [9]: **def** find\_n\_largest\_elements(lst,n):

```
    sorted_lst=sorted(lst,reverse=True)
```

```
    largest_elements = sorted_lst[:n]
    return largest_elements
```

```
numbers = [30,10,45,20,50,3,333,44,34567,243,24]
```

```
N = int(input("N = " ))
```

```
result = find_n_largest_elements(numbers,N)
print(f"The {N} largest elements in the list are:",result)
```

The 3 largest elements in the list are: [34567, 333, 243]

In [ ]: Program 55

Write a Python program to print even numbers **in** a list.

```
In [8]: numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

even_numbers = [num for num in numbers if num % 2 == 0]

print("Even numbers in the list:", even_numbers)
```

Even numbers in the list: [2, 4, 6, 8, 10]

```
In [10]: numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
y=[]
for i in numbers:
    if i%2==0:
        y.append(i)
print(y)
```

[2, 4, 6, 8, 10]

```
In [11]: numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
y={}
for i in numbers:
    y[i]=i**3
print(y)
```

{1: 1, 2: 8, 3: 27, 4: 64, 5: 125, 6: 216, 7: 343, 8: 512, 9: 729, 10: 1000}

```
In [18]: numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 7, 53]
y={i:i**3 for j in numbers}
print(y)
```

{10: 1000}

```
In [ ]: Program 56
Write a Python program to print odd numbers in a List.
```

```
In [1]: numbers = [1,2,3,4,5,6,7,8,910]
even_numbers = [num for num in numbers if num % 2 == 0]

print("even numbers in the list:", even_numbers)
```

even numbers in the list: [2, 4, 6, 8, 910]

```
In [ ]: 57
write a python program to Remove empty List from List
```

```
In [2]: list_of_lists = [[1,2,3],[],[4,5],[], [6,7,8],[]]

filtered_list = [i for i in list_of_lists if i]

print("list after removing empty lists:", filtered_list)
```

list after removing empty lists: [[1, 2, 3], [4, 5], [6, 7, 8]]

```
In [ ]: Program 56
Write a Python program to print odd numbers in a List.
```

```
In [3]: numbers = [1,2,3,4,5,6,7,8,9,10]
even_numbers = [num for num in numbers if num % 2 != 0]

print("odd numbers in the list:", even_numbers)
```

odd numbers in the list: [1, 3, 5, 7, 9]

In [ ]: 58  
Write a py program to Cloning or Copying a list

In [4]: # 1. Using the slice Operator

```
original_list = [1,2,3,4,5]
cloned_list = original_list[:]
print(cloned_list)
```

[1, 2, 3, 4, 5]

In [5]: # 2. Using the List() constructor

```
original_list = [1,2,3,4,5]
cloned_list = original_list[:]
print(cloned_list)
```

[1, 2, 3, 4, 5]

In [6]: # 3. Using List Comprehension

```
original_list = [1,2,3,4,5]
cloned_list = list(original_list)
print(cloned_list)
```

[1, 2, 3, 4, 5]

In [ ]: 59

write a python program to Count occurrences of an element in list.

In [ ]: def count\_occurrences(l,element):  
 count = l.count(element)

```
my_list = [1,2,3,4,5,2,3,4,6,5]
element_to_count = 2
```

```
occurrences = count_occurrences(my_list,element_to_count)
print(f"The element{element_to_count} appears {occurrences} times in
```

In [ ]: Program 60

Write a Python program to find words which are greater than given length k.

In [3]: def count\_occurrences(l,element):  
 count = l.count(element)  
 return count

```
my_list = [1,2,3,4,5,]
element_to_count = 2
```

```
occurrences = count_occurrences(my_list,element_to_count)
print(f"The element{element_to_count} appears {occurrences} time in the list")
```

The element2 appears 1 time in the list

In [ ]: