#The sum of Two Numbers

Objective: Write a program to calculate the sum of two numbers.

Input: Two integers.

Program :

```
n1 = int(input("Enter first Number  : "))
n2 = int(input("Enter second Number : "))
Sum_of_numbers = n1+n2
print("The sum of",n1,"and",n2,"is",Sum_of_numbers)
```

output :

```
Enter first Number  : 50
Enter second Number : 30
The sum of 50 and 30 is 80
```

#Odd or Even

Objective: Determine whether a number is odd or even.

Input: A single integer.

Program :

```
num = int(input("Enter the number : "))
if num % 2 == 0:
   print(num,"is even")
else:
   print(num,"is odd")
```

output :

```
Enter the number : 9
9 is odd
```

#Factorial Calculation

Objective: Compute the factorial of a given number nnn (i.e., n!=n×(n−1)×⋯×1n! = n \times (n-1) \times \dots \times 1n!=n×(n−1)×⋯×1).

Input: A single integer nnn.

Program :

```
import math

num = int(input("Enter a non-negative integer: "))
if num < 0:
   print("Factorial is not defined for negative numbers")
else:
   fact = math.factorial(num)
   print(fact)
```

output :

Enter a non-negative integer: 6

720

#Fibonacci Sequence

Objective: Generate the first nnn numbers in the Fibonacci sequence (e.g., 0, 1, 1, 2, 3, 5, ...).

Input: Integer nnn.

Program :

```
def fibonacci_sequence(n):
 if n <= 0:
   return []
 elif n == 1:
   return [0]
 else:
   list = [0, 1]
   while len(list) < n:
     next = list[-1]+list[-2]
     list.append(next)
   return list
num = int(input("Enter the number of fibonacci numbers : "))
fib = fibonacci_sequence(num)
print(fib)
```

output :

Enter the number of fibonacci numbers : 7

[0, 1, 1, 2, 3, 5, 8]

#Reverse a String

Objective: Reverse the characters in a string.

Input: A string

Program :

```
def reverse_string(string):
  return string[::-1]
my_string = str(input("Enter the String : "))
reversed_string = reverse_string(my_string)
```

print(reversed_string)

output :

Enter the String : Anusha

ahsunA

#Palindrome Check

Objective: Check if a string reads the same backward as forward.

Input: A string.

Program :

```
def is_palindrome(string):
  return string == string[::-1]
string1 = str(input("Enter the String1 : "))
string2 = str(input("ENter the string2 : "))

print(string1, "is palindrome:", is_palindrome(string1))
print(string2, "is palindrome:", is_palindrome(string2))
```

output :

Enter the String1 : Fruit

ENter the string2 : sweets

Fruit is palindrome: False

sweets is palindrome: False

#Leap Year Check

Objective: Determine whether a year is a leap year.

Input: An integer year (e.g., 2024).

Program :

```
def is_leap_year(year):
  if (year % 4 == 0 and year % 100 != 0) or year % 400 == 0:
    return True
  else:
    return False
year1 = int(input("Enter the number : "))
year2 = int(input("Enter the number : "))
year3 = int(input("Enter the number : "))
year4 = int(input("Enter the number : "))

print(year1, "is leap year:", is_leap_year(year1))
print(year2, "is leap year:", is_leap_year(year2))
print(year3, "is leap year:", is_leap_year(year3))
print(year4, "is leap year:", is_leap_year(year4))
```

output :

```
Enter the number : 1995
Enter the number : 2000
Enter the number : 1998
Enter the number : 2005
1995 is leap year: False
2000 is leap year: True
1998 is leap year: False
```

2005 is leap year: False

#Armstrong Number

Objective: Check if a number equals the sum of its digits raised to the power of the number of digits.

Program:

```
def armstrong_number(n):
  digit = str(n)
  return n == sum(int(d) ** len(digit) for d in digit)
num = int(input("Enter a number: "))
print(f"Is {num} an Armstrong number? {armstrong_number(num)}.")
```

output :

Enter a number: 153

Is 153 an Armstrong number? True.

#Custom Encryption-Decryption System

Program:

```
def encrypt(text, shift):
    encrypted_text = ""
    for char in text:
        if char.isalpha():
            shift_base = 65 if char.isupper() else 97
            encrypted_text += chr((ord(char) - shift_base + shift) % 26 + shift_base)
        else:
            encrypted_text += char
    return encrypted_text
```

```
def decrypt(encrypted_text, shift):
    return encrypt(encrypted_text, -shift)
message = "Hello World!"
shift_key = 3
encrypted_message = encrypt(message, shift_key)
decrypted_message = decrypt(encrypted_message, shift_key)
print(f"Original: {message}")
print(f"Encrypted: {encrypted_message}")
print(f"Decrypted: {decrypted_message}")
```

output :

Enter the string : Anusha Chinthala

Original: Anusha Chinthala

Encrypted: Dqxvkd Fklqwkdod

Decrypted: Anusha Chinthala