1. Write a Python Program to Check Prime Number.

Prime Numbers:
A prime number is a whole number that cannot be evenly divided by any other number except for 1 and itself. For example, 2, 3, 5, 7, 11, and 13 are prime numbers because they cannot be divided by any other positive integer except for 1 and their own value.

```
num = int(input("Enter a number: "))
# define a flag variable
flag = False
if num == 1:
    print(f"{num}, is not a prime number")
elif num > 1:
    # check for factors
    for i in range(2, num):
        if (num % i) == 0:
            flag = True
            break
# if factor is found, set flag to True
# break out of loop
# check if flag is True
if flag:
    print(f"{num}, is not a prime number")
else:
    print(f"{num}, is a prime number")
```

OUTPUT: Enter a number: 37
37, is a prime number

2. Write a python program to display all prime numbers with in an interval.

```
# Python program to display all the prime numbers within an interval
start = int(input("Enter the starting number:"))
end = int(input("Enter the ending number:"))
print("Prime numbers between", start, "and",end, "are:")
for num in range(start, end + 1):
    # all prime numbers are greater than 1
    if num > 1:
        for i in range(2, num):
            if (num % i) == 0:
                break
        else:
            print(num,end=",")
```

OUTPUT: Enter the starting number:1
Enter the ending number:50
Prime numbers between 1 and 50 are:
2,3,5,7,11,13,17,19,23,29,31,37,41,43,47.

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

```
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}')
```

OUTPUT: Enter a number: 5
The factorial of 5 is 120.

4.Write a Python Program to Display the multiplication Table.

```
num = int(input("Display multiplication table of: "))
for i in range(1, 11):
    print(f"{num} X {i} = {num*i}")
```

OUTPUT: Display multiplication table of: 5

5 X 1 = 5
5 X 2 = 10
5 X 3 = 15
5 X 4 = 20
5 X 5 = 25
5 X 6 = 30
5 X 7 = 35
5 X 8 = 40
5 X 9 = 45
5 X 10 = 50

5. Write a Python Program to Print the Fibonacci sequence.
Fibonacci sequence:
The Fibonacci sequence is a series of numbers where each number is the sum of the two preceding ones, typically starting with 0 and 1. So, the sequence begins with 0 and 1, andthe next number is obtained by adding the previous two numbers. This pattern continuesindefinitely, generating a sequence that looks like this: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, and so on.
Mathematically, the Fibonacci sequence can be defined using the following recurrence relation:
$F(0) = 0$ $F(1) = 1$ $F(n) = F(n-1)+F(n-2)$ for $n > 1$

OUTPUT: How many terms? 10
Fibonacci sequence:0 1 1 2 3 5 8 13 21 34

```
nterms = int(input("How many terms? "))
# first two terms
n1, n2 = 0, 1
count = 0
# check if the number of terms is valid
if nterms <= 0:
    print("Please enter a positive integer")
# if there is only one term, return n1
elif nterms == 1:
    print("Fibonacci sequence upto",nterms,":")
    print(n1)
# generate fibonacci sequence
else:
    print("Fibonacci sequence:")
    while count < nterms:
        print(n1,end=" ")
        nth = n1 + n2
        # update values
        n1 = n2
        n2 = nth
        count += 1
```

6. Write a Python Program to Check given number is Armstrong Number or not.

```
num = int(input("Enter a number: "))
# Calculate the number of digits in num
num_str = str(num)
num_digits = len(num_str)
# Initialize variables
sum_of_powers = 0
temp_num = num
# Calculate the sum of digits raised to the power of num_digits
while temp_num > 0:
    digit = temp_num % 10
    sum_of_powers += digit ** num_digits
    temp_num //= 10
# Check if it's an Armstrong number
if sum_of_powers == num:
    print(f"{num} is an Armstrong number.")
else:
    print(f"{num} is not an Armstrong number.")
```

OUTPUT: Enter a number: 153
153 is an Armstrong number.

7. Write a Python Program to Find Armstrong Number in an Interval.

```
# Input the interval from the user
lower = int(input("Enter the lower limit of the interval: "))
upper = int(input("Enter the upper limit of the interval: "))
print("Prime numbers between", lower, "and", upper, "are:")
for num in range(lower, upper + 1):
    # Iterate through the numbers i
    order = len(str(num))  # Find the number of digits in 'num'
    temp_num = num
    sum = 0
    while temp_num > 0:
        digit = temp_num % 10
        sum += digit ** order
        temp_num //= 10
    # Check if 'num' is an Armstrong number
    if num == sum:
        print(num,end=" ")
```

OUTPUT: Enter the lower limit of the interval: 100
Enter the upper limit of the interval: 500
Prime numbers between 100 and 500 are:
153 370 371 407

8.Write a Python Program to Find LCM.

```
num1 = int(input('Enter the number: '))
num2 = int(input('Enter the number: '))
if num1 > num2:
    # choose the greater number
    greater = num1
else:
    greater = num2
while(True):
    if((greater % num1 == 0) and (greater % num2 == 0)):
        lcm = greater
        break
    greater += 1

print("The L.C.M. is", lcm)
```

OUTPUT: Enter the number: 24

Enter the number: 54
The L.C.M. is 216

09. Write a Python Program to Find HCF.

```
num1 = int(input('Enter the number: '))
num2 = int(input('Enter the number: '))
if num1 > num2:
    # choose the greater number
    smaller = num2
else:
    smaller = num1
for i in range(1,smaller+1):
    if((num1 % i == 0) and (num2 % i == 0)):
        hcf =i

print("The H.C.F. is", hcf)
```

OUTPUT:
Enter the number: 12
Enter the number: 18
The H.C.F. is 6

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

A Disarium number is a number that is equal to the sum of its digits each raised to thepower of its respective position.

```
num = int(input("Enter a number: "))
temp = num
digits = list(map(int, str(num)))
print(len(digits))

sum_power = 0
for i in range(len(digits)):
    sum_power += digits[i] ** (i + 1)

if sum_power == num:
    print(num, "is a Disarium Number")
else:
    print(num, "is not a Disarium Number")
```

OUTPUT: Enter a number: 89
89 is a Disarium Number

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

Happy Number: A Happy Number is a positive integer that, when you repeatedly replacethe number by the sum of the squares of its digits and continue the process, eventuallyreaches 1. If the process never reaches 1 but instead loops endlessly in a cycle, the numberis not a Happy Number.

```
start = int(input("Enter starting number: "))
end = int(input("Enter ending number: "))

print("Happy Numbers in the given range:")
for i in range(start, end + 1):
    seen = set()
    num=i
    while i != 1 and i not in seen:
        seen.add(i)
        total = 0
        while i > 0:
            digit = i % 10
            total += digit ** 2
            i //= 10
        i = total
        #print(i)
    if i == 1:
        print(num, end=" ")
```

OUTPUT:

Enter starting number: 19
Enter ending number: 50
Happy Numbers in the given range:
19 23 28 31 32 44 49