**1-Write a Python program to determine the largest among three numbers entered by the user.**

**Ans:-**

# Get three numbers from the user

num1 = float(input("Enter the first number: "))

num2 = float(input("Enter the second number: "))

num3 = float(input("Enter the third number: "))

# Determine the largest number

if (num1 >= num2) and (num1 >= num3):

largest = num1

elif (num2 >= num1) and (num2 >= num3):

largest = num2

else:

largest = num3

# Print the result

print("The largest number is", largest)

**2-Write a Python program to check if a given year is a leap year.**

**Hints-**

**1. Check if the year is divisible by 4**

**2. Check if the year is divisible by 100, but not divisible by 400**

**3. If the year satisfies either of the above conditions, it's a leap year**

**Ans:-**

# Function to check if a year is a leap year

def is\_leap\_year(year):

# Check if the year is divisible by 4

if year % 4 == 0:

# Check if the year is divisible by 100

if year % 100 == 0:

# If the year is divisible by 400, it's a leap year

if year % 400 == 0:

return True

else:

return False

else:

return True

else:

return False

# Get a year from the user

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

# Check if the entered year is a leap year

if is\_leap\_year(year):

print(f"{year} is a leap year.")

else:

print(f"{year} is not a leap year.")

**3-Write a Python program to print the following pattern:**

**Ans:-**

# Number of levels in the pattern

levels = 5

# Generate the pattern

for i in range(1, levels + 1):

print(str(i) \* i)

**4-Write a Python program to print the following pattern:**

**Ans:-**

# Number of levels in the pattern

levels = 5

# Generate the pattern

for i in range(1, levels + 1):

print('\*' \* i)

**5-Write a Python program to print the following pattern:**

**Ans:-**

# Number of levels in the pattern

levels = 5

# Generate the pattern

for i in range(1, levels + 1):

# Print spaces ' ' for the indentation and '\*' for the pattern

print(' ' \* (levels - i) + '\*' \* i)

**6-Write a program that prints the numbers 1 to 100. However, for multiples of 3, print "Fizz" instead of the number. For multiples of 5, print "Buzz". For numbers that are multiples of both 3 and 5, print "FizzBuzz".**

for num in range(1, 101):

if num % 3 == 0 and num % 5 == 0:

print("FizzBuzz")

elif num % 3 == 0:

print("Fizz")

elif num % 5 == 0:

print("Buzz")

else: print(num)

**7Write a Python program to find the sum of all prime numbers up to n.**

**Ans:**

def is\_prime(num):

if num <= 1:

return False

for i in range(2, int(num\*\*0.5) + 1):

if num % i == 0:

return False

return True

def sum\_of\_primes(n):

return sum([i for i in range(2, n+1) if is\_prime(i)])

# Example usage:

n = int(input("Enter the value of n: "))

print(f"The sum of all prime numbers up to {n} is {sum\_of\_primes(n)}")

**8-Write a Python program to print the Fibonacci sequence up to n terms. The Fibonacci series is a sequend numbers where each number is the sum of the two preceding ones, usually starting with 0 and 1. i.e**

**Unset 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, …**

**Ans:-**

def fibonacci\_recursive(n):

if n <= 1:

return n

else:

return fibonacci\_recursive(n - 1) + fibonacci\_recursive(n - 2)

n = int(input("Enter the number of terms: "))

print("Fibonacci sequence up to", n, "terms:")

for i in range(n):

print(fibonacci\_recursive(i), end=" ")

**9-Write a Python program to print the following pattern:**

**Unset**

**\***

**\*\*\***

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

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

**Ans:-**

def print\_pattern(n):

for i in range(n):

print(' ' \* (n - i - 1) + '\*' \* (2 \* i + 1))

# Call the function with the number of lines for the pattern

print\_pattern(4)

**10-Write a Python program to print the following pattern:**

**Unset**

**A**

**BBB**

**CCCCC**

**DDDDDDD**

**EEEEEEEEEE**

**ANS:-**

def print\_pattern(n):

for i in range(n):

print(' ' \* (n - i - 1) + chr(65 + i) \* (2 \* i + 1))

# Call the function with the number of lines for the pattern

print\_pattern(5)