**Name:- Gayatri Gorakh Jadhav**

**Roll No:- CS024**

**Date:-\_\_\_\_\_\_\_\_\_\_\_\_**

**Practical No:1**

# ----------------------------------------------------------------

**AIM:- Write a python program to display all types of pyramids of stars**

**---------------------------------------------------------------------------------------------------------------**

**CODE:-**

def print\_right\_angle\_triangle(n):

print("Right-Angle Triangle:")

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

print('\*' \* i)

print()

def print\_isosceles\_triangle(n):

print("Isosceles Triangle:")

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

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

print()

def print\_inverted\_triangle(n):

print("Inverted Triangle:")

for i in range(n, 0, -1):

print('\*' \* i)

print()

def print\_full\_pyramid(n):

print("Full Pyramid:")

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

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

print()

def print\_diamond(n):

print("Diamond Shape:")

# Upper part

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

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

# Lower part

for i in range(n - 1, 0, -1):

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

print()

# Set the height of the pyramids

n = 5

print\_right\_angle\_triangle(n)

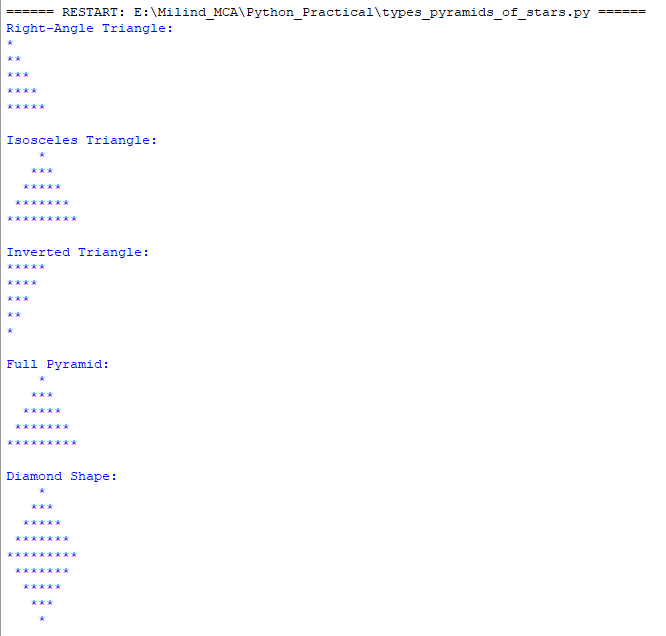
print\_isosceles\_triangle(n)

print\_inverted\_triangle(n)

print\_full\_pyramid(n)

print\_diamond(n)

**Output:-**



**Name:- Gayatri Gorakh Jadhav**

**Roll No:- CS024**

**Date:-\_\_\_\_\_\_\_\_\_\_\_\_**

**Practical No:2**

# ----------------------------------------------------------------

**AIM:- Write a program to display multiplication table of all numbers from 1 to 10.**

**----------------------------------------------------------------------------------------------------------------**

**CODE:-**

# Loop through numbers 1 to 10 for the multiplication table

for i in range(1, 11):

# Print a header for the current multiplication table

print("\n\nMULTIPLICATION TABLE FOR %d\n" % (i))

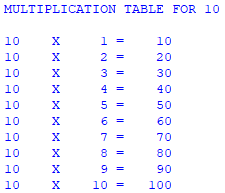
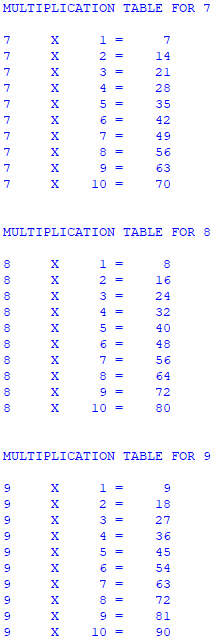
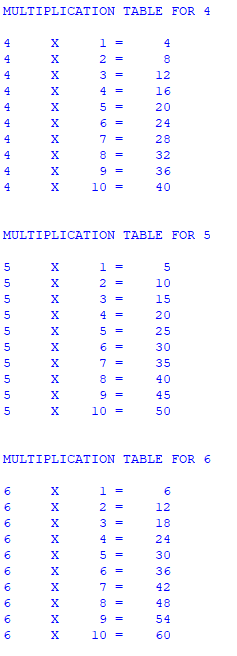
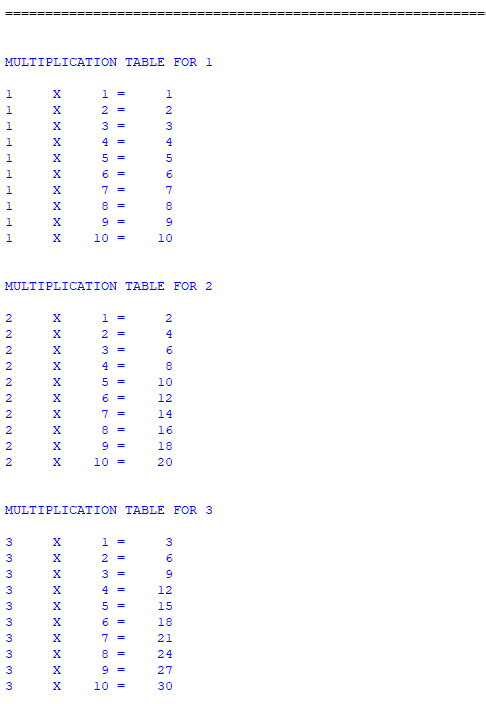
# Loop through numbers 1 to 10 for the multipliers

for j in range(1, 11):

# Print the multiplication expression and result with formatted output

print("%-5d X %5d = %5d" % (i, j, i \* j))

**Output:-**



**Name:- Gayatri Gorakh Jadhav**

**Roll No:- CS024**

**Date:-\_\_\_\_\_\_\_\_\_\_\_\_**

**Practical No: 3**

# ----------------------------------------------------------------

**AIM:- Write a program to calculate simple interest except amount, duration and rate of interest from user.**

**----------------------------------------------------------------------------------------------------------------**

**CODE:-**

# Function to calculate simple interest

def calculate\_simple\_interest(principal, rate, duration):

# Calculate simple interest using the formula: (P \* R \* T) / 100

return (principal \* rate \* duration) / 100

# Prompt the user to enter the principal amount and convert it to a float

principal = float(input("Enter the principal amount: "))

# Prompt the user to enter the duration in years and convert it to a float

duration = float(input("Enter the duration (in years): "))

# Prompt the user to enter the rate of interest and convert it to a float

rate = float(input("Enter the rate of interest (in %): "))

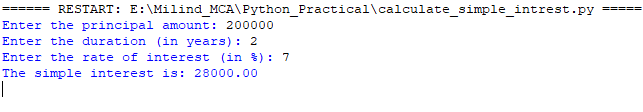
# Calculate simple interest using the user inputs

simple\_interest = calculate\_simple\_interest(principal, rate, duration)

# Print the calculated simple interest, formatted to two decimal places

print(f"The simple interest is: {simple\_interest:.2f}")

**Output:-**



**Name:- Gayatri Gorakh Jadhav**

**Roll No:- CS024**

**Date:-\_\_\_\_\_\_\_\_\_\_\_\_**

**Practical No: 4**

# ----------------------------------------------------------------

**AIM:- Write a program to count even and odd number in the list.**

**----------------------------------------------------------------------------------------------------------------**

**CODE:-**

# Function to count even and odd numbers in a list

def count\_even\_odd(numbers):

# Initialize counters for even and odd numbers

even\_count = 0

odd\_count = 0

# Iterate through each number in the provided list

for number in numbers:

# Check if the number is even

if number % 2 == 0:

even\_count += 1 # Increment even count

else:

odd\_count += 1 # Increment odd count

# Return the counts of even and odd numbers

return even\_count, odd\_count

# List of numbers from 1 to 20

numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20]

# Call the function and store the results in even\_count and odd\_count

even\_count, odd\_count = count\_even\_odd(numbers)

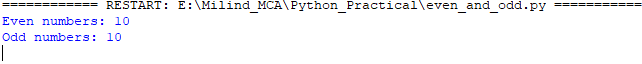
# Print the count of even numbers

print(f"Even numbers: {even\_count}")

# Print the count of odd numbers

print(f"Odd numbers: {odd\_count}")

**Output:-**



**Name:- Gayatri Gorakh Jadhav**

**Roll No:- CS024**

**Date:-\_\_\_\_\_\_\_\_\_\_\_\_**

**Practical No: 5**

# ----------------------------------------------------------------

**AIM:- Write a program to find sum of all numbers, mean, max, average of numbers in a list.**

**----------------------------------------------------------------------------------------------------------------**

**CODE:-**

from collections import Counter

def calculate\_statistics(numbers):

if not numbers:

return None, None, None, None, None

# Calculate sum

total\_sum = sum(numbers)

# Calculate minimum

minimum = min(numbers)

# Calculate maximum

maximum = max(numbers)

# Calculate mean

mean = total\_sum / len(numbers)

# Calculate mode

frequency = Counter(numbers)

mode\_data = frequency.most\_common()

mode = [num for num, freq in mode\_data if freq == mode\_data[0][1]]

return total\_sum, minimum, maximum, mean, mode

# Example usage

numbers = [1, 2, 2, 3, 4, 4, 4, 5]

total\_sum, minimum, maximum, mean, mode = calculate\_statistics(numbers)

print(f"Sum: {total\_sum}")

print(f"Min: {minimum}")

print(f"Max: {maximum}")

print(f"Mean: {mean}")

print(f"Mode: {mode}")

**Output:-**

