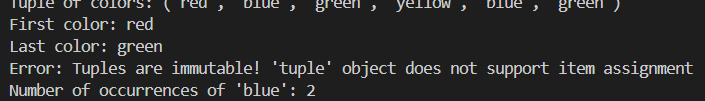
**Name:- Harsh Bhartia**

**SAP ID:- 500106274 B-4(AIML-NH)**

**Q1): Create and access tuples.**

* + Create a tuple of colors.
  + Access elements using indexing.
  + Try to modify an element in the tuple (to demonstrate immutability).
  + Find the number of occurrences of a specific element in the tuple.
* # Create a tuple of colors
* colors = ("red", "blue", "green", "yellow", "blue", "green")
* print("Tuple of colors:", colors)
* # Access elements using indexing
* print("First color:", colors[0])   # Output: red
* print("Last color:", colors[-1])  # Output: green
* try:
* colors[0] = "purple"  # Attempt to modify
* except TypeError as e:
* print("Error: Tuples are immutable!", e)
* # Count the occurrences of "blue"
* count\_blue = colors.count("blue")
* print("Number of occurrences of 'blue':", count\_blue)

OUTPUT:-

**Create and manipulate dictionaries.**

* + Create a dictionary to store information about a person (name, age, city).
  + Access values using keys.
  + Add a new key-value pair to the dictionary.
  + Modify an existing value.Check if a key exists in the dictionary.

Get a list of all keys and values.

# Create a dictionary

person = {"name": "Alice", "age": 25, "city": "New York"}

print("Initial dictionary:", person)

# Access values using keys

print("Name:", person["name"])  # Output: Alice

print("City:", person["city"])  # Output: New York

# Add a new key-value pair

person["profession"] = "Engineer"

print("After adding a new key-value pair:", person)

# Modify an existing value

person["age"] = 26

print("After modifying age:", person)

# Check if a key exists

key\_to\_check = "name"

if key\_to\_check in person:

    print(f"Key '{key\_to\_check}' exists in the dictionary.")

else:

    print(f"Key '{key\_to\_check}' does not exist.")

    # Get all keys and values

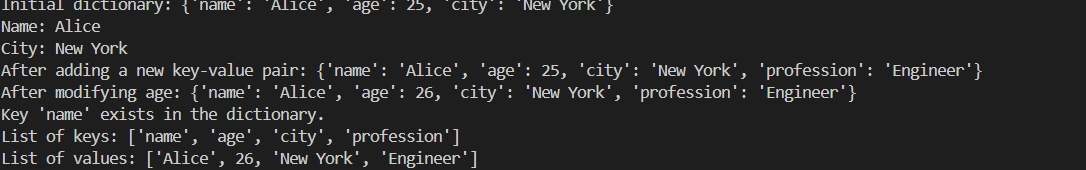
keys = list(person.keys())

values = list(person.values())

print("List of keys:", keys)

print("List of values:", values)

OUTPUT:-



**Demonstrate the difference between mutable and immutable data types.**

* + Create a list and a tuple.
  + Try to modify an element in both the list and the tuple.
  + Observe the results and explain the difference.
* # Create a list and a tuple
* my\_list = [1, 2, 3]
* my\_tuple = (1, 2, 4)
* print("Original list:", my\_list)
* print("Original tuple:", my\_tuple)
* # Modify an element in the list
* my\_list[0] = 10
* print("Modified list:", my\_list)  # Successfully modified
* try:
* my\_tuple[0] = 10  # Attempt to modify
* except TypeError as e:
* print("Error:", e)  # Tuples are immutable

**Create a program to print the multiplication table of a number.**

* + Take a number as input from the user.
  + Use a for loop to iterate from 1 to 10.
  + Calculate the product of the input number and the current iteration.
  + Print the multiplication table.

# Take a number as input from the user

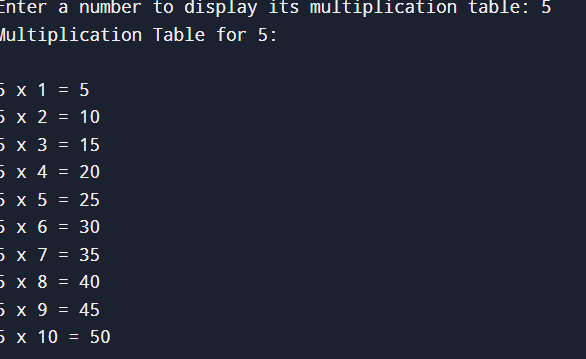
number = int(input("Enter a number to display its multiplication table: "))

# Print the multiplication table using a for loop

print(f"Multiplication Table for {number}:\n")

for i in range(1, 11): # Iterate from 1 to 10

product = number \* i

 print(f"{number} x {i} = {product}")

**Create a program to find the factorial of a number using a loop and conditional statements.**

# Take input from the user

number = int(input("Enter a number to find its factorial: "))

# Initialize factorial result

factorial = 1

# Check if the number is negative, zero, or positive

if number < 0:

print("Factorial is not defined for negative numbers.")

elif number == 0:

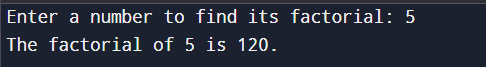
print("The factorial of 0 is 1.")

else:

for i in range(1, number + 1): # Loop from 1 to the number

factorial \*= i # Multiply each number with factorial

print(f"The factorial of {number} is {factorial}.")



**Write a program to check if a given number is prime.**

# Take input from the user

number = int(input("Enter a number to check if it is prime: "))

# Prime check logic

if number <= 1:

print(f"{number} is not a prime number.")

else:

is\_prime = True

for i in range(2, int(number\*\*0.5) + 1): # Loop from 2 to the square root of the number

if number % i == 0:

is\_prime = False

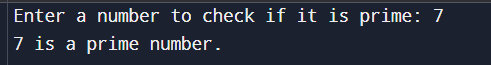
break

if is\_prime:

print(f"{number} is a prime number.")

else:

print(f"{number} is not a prime number.")



**Create a program to find the sum of all even numbers between 1 and 100.**

# Initialize a variable to store the sum

sum\_of\_evens = 0

# Loop through numbers from 1 to 100

for number in range(1, 101): # Range is inclusive of 1 and exclusive of 101

if number % 2 == 0: # Check if the number is even

sum\_of\_evens += number # Add the even number to the sum

# Print the result

print(f"The sum of all even numbers between 1 and 100 is: {sum\_of\_evens}")



**Implement a simple calculator using conditional statements and loops.**

# Simple calculator program

while True:

print("\nSimple Calculator")

print("Select an operation:")

print("1. Addition (+)")

print("2. Subtraction (-)")

print("3. Multiplication (\*)")

print("4. Division (/)")

print("5. Exit")

# Take user input for operation

choice = input("Enter the number of the operation (1/2/3/4/5): ")

# Exit the loop if the user selects option 5

if choice == '5':

print("Exiting the calculator. Goodbye!")

break

# Check if the choice is valid

if choice in ['1', '2', '3', '4']:

# Take two numbers as input

try:

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

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

# Perform the selected operation

if choice == '1':

print(f"Result: {num1} + {num2} = {num1 + num2}")

elif choice == '2':

print(f"Result: {num1} - {num2} = {num1 - num2}")

elif choice == '3':

print(f"Result: {num1} \* {num2} = {num1 \* num2}")

elif choice == '4':

if num2 != 0:

print(f"Result: {num1} / {num2} = {num1 / num2}")

else:

print("Error: Division by zero is not allowed.")

except ValueError:

print("Error: Invalid input. Please enter numeric values.")

else:

print("Invalid choice. Please select a valid operation.")

