ACTIVITY – 2

FARAZ GHANI

SAP -ID-500109330

Create and access tuples.

- o Create a tuple of colors.

- o Access elements using indexing.

- o Try to modify an element in the tuple (to demonstrate immutability).

- o Find the number of occurrences of a specific element in the tuple.

```python
colors = ('red' , 'green' , 'blue' , 'yellow')
print(colors[0])  # Output: 'red'

# Accessing the third element
print(colors[2])  # Output: 'blue'

try:
    colors[1] = 'purple'  # Attempt to modify an element
except TypeError as e:
    print(e)  # Output: 'tuple' object does not support item assignment

count_blue = colors.count('blue')
print(count_blue)  # Output: 1

# Count the number of occurrences of 'red'
count_red = colors.count('red')
print(count_red)  # Output: 1
```

```
blue
'tuple' object does not support item assignment
1
1



...Program finished with exit code 0
Press ENTER to exit console.
```

Q2. Create and manipulate dictionaries.

- o Create a dictionary to store information about a person (name, age, city).

- o Access values using keys.

- o Add a new key-value pair to the dictionary.

- o Modify an existing value.

- o Check if a key exists in the dictionary.

- o Get a list of all keys and values.

```
#1
person = {
    'name': 'Alice',
    'age': 30,
    'city': 'New York'
}
#2
print(person['name'])
print(person['age'])

#3
person['occupation'] = 'Engineer'
print(person)

#4
person['city'] = 'San Francisco'
print(person)
#5
if 'age' in person:
    print("Key 'age' exists in the dictionary.")
else:
    print("Key 'age' does not exist in the dictionary.")

#6
keys = list(person.keys())
print(keys)

values = list(person.values())
```

```
Alice
30
{'name': 'Alice', 'age': 30, 'city': 'New York', 'occupation': 'Engineer'}
{'name': 'Alice', 'age': 30, 'city': 'San Francisco', 'occupation': 'Engineer'}
Key 'age' exists in the dictionary.
['name', 'age', 'city', 'occupation']
['Alice', 30, 'San Francisco', 'Engineer']
```

Demonstrate the difference between mutable and immutable data types.

- o   Create a list and a tuple.

- o   Try to modify an element in both the list and the tuple.

- o   Observe the results and explain the difference.

```python
my_list = [1, 2, 3, 4]

my_tuple = (1, 2, 3, 4)

my_list[2] = 99
print(my_list)


try:

    my_tuple[2] = 99
except TypeError as e:
    print(e)
```

```
[1, 2, 99, 4]
'tuple' object does not support item assignment
```

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

- o   Take a number as input from the user.

- o   Use a for loop to iterate from 1 to 10.

- o   Calculate the product of the input number and the current iteration.

- o   Print the multiplication table.

```python
number = int(input("Enter a number to print its multiplication table: "))

print(f"Multiplication table for {number}:")
for i in range(1, 11):

    product = number * i

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

```
Enter a number to print its multiplication table: 3
Multiplication table for 3:
3 x 1 = 3
3 x 2 = 6
3 x 3 = 9
3 x 4 = 12
3 x 5 = 15
3 x 6 = 18
3 x 7 = 21
3 x 8 = 24
3 x 9 = 27
3 x 10 = 30


...Program finished with exit code 0
Press ENTER to exit console.
```

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

```python
def factorial(n):

    result = 1


    for i in range(1, n + 1):
        result *= i

    return result


number = int(input("Enter a non-negative integer to find its factorial: "))


if number < 0:
    print("Factorial is not defined for negative numbers.")
else:

    fact = factorial(number)

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

```
Enter a non-negative integer to find its factorial: 7
The factorial of 7 is 5040.


...Program finished with exit code 0
Press ENTER to exit console.
```

Write a program to check if a given number is prime

```python
def is_prime(num):
    # Check if num is less than 2 (not a prime number)
    if num <= 1:
        return False

    # Check if num is 2 or 3 (both are prime numbers)
    if num in (2, 3):
        return True

    # Check if num is divisible by 2 or 3
    if num % 2 == 0 or num % 3 == 0:
        return False

    # Check for factors from 5 up to the square root of num
    i = 5
    while i * i <= num:
        if num % i == 0 or num % (i + 2) == 0:
            return False
        i += 6

    return True

# Take a number as input from the user
number = int(input("Enter a number to check if it is prime: "))

# Check if the number is prime
if is_prime(number):
    print(f"{number} is a prime number.")
```

```python
if is_prime(number):
    print(f"{number} is a prime number.")
else:
    print(f"{number} is not a prime number.")
```

```
Enter a number to check if it is prime: 3
3 is a prime number.


...Program finished with exit code 0
Press ENTER to exit console.
```

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

```python
total_sum = 0

for number in range(1, 101):

    if number % 2 == 0:

        total_sum += number

print(f"The sum of all even numbers between 1 and 100 is {total_sum}.")
```

```
The sum of all even numbers between 1 and 100 is 2550.


...Program finished with exit code 0
Press ENTER to exit console.
```

Implement a simple calculator using conditional statements and loops.

```python
def calculator():
    while True:
        # Display the menu
        print("\nSimple Calculator")
        print("1. Addition")
        print("2. Subtraction")
        print("3. Multiplication")
        print("4. Division")
        print("5. Exit")

        # Take the user's choice
        choice = input("Enter your choice (1/2/3/4/5): ")

        # Check if the user wants to exit
        if choice == '5':
            print("Exiting the calculator. Goodbye!")
            break

        # Ensure the choice is valid
        if choice in ('1', '2', '3', '4'):
            # Take input for the two numbers
            try:
                num1 = float(input("Enter the first number: "))
                num2 = float(input("Enter the second number: "))
            except ValueError:
                print("Invalid input. Please enter numerical values.")
                continue

            # Perform the chosen operation
            if choice == '1':
                result = num1 + num2
                print(f"The result of {num1} + {num2} is {result}.")
            elif choice == '2':
                result = num1 - num2
                print(f"The result of {num1} - {num2} is {result}.")
            elif choice == '3':
                result = num1 * num2
                print(f"The result of {num1} * {num2} is {result}.")
            elif choice == '4':
                if num2 != 0:
                    result = num1 / num2
                    print(f"The result of {num1} / {num2} is {result}.")
                else:
                    print("Error: Division by zero is not allowed.")
        else:
            print("Invalid choice. Please select a valid option.")

# Run the calculator
calculator()
```

```
Simple Calculator
1. Addition
2. Subtraction
3. Multiplication
4. Division
5. Exit
Enter your choice (1/2/3/4/5): 1
Enter the first number: 3
Enter the second number: 4
The result of 3.0 + 4.0 is 7.0.

Simple Calculator
1. Addition
2. Subtraction
3. Multiplication
4. Division
5. Exit
Enter your choice (1/2/3/4/5): 2
Enter the first number: 4
Enter the second number: 2
The result of 4.0 - 2.0 is 2.0.

Simple Calculator
1. Addition
2. Subtraction
3. Multiplication
4. Division
5. Exit
Enter your choice (1/2/3/4/5): 3
Enter the first number: 6
```

```
Enter the second number: 2
The result of 6.0 * 2.0 is 12.0.

Simple Calculator
1. Addition
2. Subtraction
3. Multiplication
4. Division
5. Exit
Enter your choice (1/2/3/4/5): 4
Enter the first number: 4
Enter the second number: 2
The result of 4.0 / 2.0 is 2.0.

Simple Calculator
1. Addition
2. Subtraction
3. Multiplication
4. Division
5. Exit
```