Usama Arif Rollno 14

Task 1:Make a mini-calculator using functions. All the functions should be accessed from module.Your program should ask inputs and option from user. ¶

In [7]:

```python
def add(a, b):
    return a + b

def subtract(a, b):
    return a - b

def multiply(a, b):
    return a * b

def divide(a, b):
    if b != 0:
        return a / b
    else:
        print("Error: Cannot divide by zero!")

while True:
    print("Select operation:")
    print("1. Add")

    print("2. Subtract")
    print("3. Multiply")
    print("4. Divide")
    print("5. Exit")

    choice = int(input("Enter your choice (1-5): "))

    if choice == 5:
        break

    num1 = float(input("Enter the first number: "))
    num2 = float(input("Enter the second number: "))

    if choice == 1:
        result = add(num1, num2)
    elif choice == 2:
        result = subtract(num1, num2)
    elif choice == 3:
        result = multiply(num1, num2)
    elif choice == 4:
        result = divide(num1, num2)
    else:
        print("Invalid choice!")

    print("Result:", result)

```

```
Select operation:
1. Add
2. Subtract
3. Multiply
4. Divide
5. Exit
Enter your choice (1-5): 1
Enter the first number: 2
Enter the second number: 3
Result: 5.0
Select operation:
1. Add
2. Subtract
3. Multiply
4. Divide
5. Exit
Enter your choice (1-5): 4
Enter the first number: 4
Enter the second number: 3
Result: 1.3333333333333333
Select operation:
1. Add
2. Subtract
3. Multiply
4. Divide
5. Exit
Enter your choice (1-5): 2
Enter the first number: 1
Enter the second number: 2
Result: -1.0
Select operation:
1. Add
2. Subtract
3. Multiply
4. Divide
5. Exit
Enter your choice (1-5): 3
Enter the first number: 2
Enter the second number: 1
Result: 2.0
Select operation:
1. Add
2. Subtract
3. Multiply
4. Divide
5. Exit
Enter your choice (1-5): 2
Enter the first number: 3
Enter the second number: 2
Result: 1.0
Select operation:
1. Add
2. Subtract
3. Multiply
4. Divide
5. Exit
Enter your choice (1-5): 1
```

```
Enter the first number: 2
Enter the second number: 5
Result: 7.0
Select operation:
1. Add
2. Subtract
3. Multiply
4. Divide
5. Exit
Enter your choice (1-5): 3
Enter the first number: 4
Enter the second number: 5
Result: 20.0
Select operation:
1. Add
2. Subtract
3. Multiply
4. Divide
5. Exit
Enter your choice (1-5): 5
```

Task 2: Write a function which can apply this function in all elements of list. Y=x $\sqrt{}$ using list comprehension.

In [ ]:

```python
import math

def square_root(x):
    return math.sqrt(x)

my_list = [1, 4, 9, 16, 25]

result = [square_root(x) for x in my_list]

print(result)

```

Task 3: Apply all arithmetic functions on Numpy array and discus output.

In [8]:
```
import numpy as np

my_array = np.array([1, 2, 3, 4, 5])

# Example arithmetic functions
result_sum = np.sum(my_array)          # Sum of all elements
result_mean = np.mean(my_array)        # Mean of all elements
result_max = np.max(my_array)          # Maximum value
result_min = np.min(my_array)          # Minimum value
result_std = np.std(my_array)          # Standard deviation

print("Sum:", result_sum)
print("Mean:", result_mean)
print("Max:", result_max)
print("Min:", result_min)
print("Standard Deviation:", result_std)

```

```
Sum: 15
Mean: 3.0
Max: 5
Min: 1
Standard Deviation: 1.4142135623730951
```

Task 4:Make a Numpy array by using all possible ways.

In [14]:

```
import numpy as np

# Create an array from a list
array1 = np.array([1, 2, 3, 4, 5])

# Create an array of zeros
array2 = np.zeros((3, 3))

# Create an array of ones
array3 = np.ones((2, 2))

# Create a range of values
array4 = np.arange(0, 10, 2)

# Create a random array
array5 = np.random.rand(3, 3)
#random array using seed() and randint
array6=np.random.randint(10,size=(3,3,3))

print("Array 1:", array1)
print("\n")
print("Array 2:", array2)
print("\n")
print("Array 3:", array3)
print("\n")
print("Array 4:", array4)
print("\n")
print("Array 5:", array5)
print("\n")
print("array6: ",array6)

```

```
Array 1: [1 2 3 4 5]


Array 2: [[0. 0. 0.]
 [0. 0. 0.]
 [0. 0. 0.]]


Array 3: [[1. 1.]
 [1. 1.]]


Array 4: [0 2 4 6 8]


Array 5: [[0.24048389 0.94567847 0.04863794]
 [0.70354035 0.39061623 0.36544407]
 [0.07845401 0.96785182 0.39634651]]


array6:  [[[7 4 8]
  [4 2 9]
  [4 6 2]]

 [[9 2 9]
  [7 1 4]
  [6 4 1]]

 [[4 4 5]
  [1 7 5]
  [0 0 6]]]
```

Task 5: Perform basic indexing and slicing on multi-dimension array.

In [15]:
```python
import numpy as np

# Create a 2D array
my_array = np.array([[1, 2, 3],
                     [4, 5, 6],
                     [7, 8, 9]])

# Basic indexing
print("Element at (0, 0):", my_array[0, 0])
print("Element at (1, 2):", my_array[1, 2])

# Slicing
print("Row 0:", my_array[0, :])
print("Column 1:", my_array[:, 1])
print("Subarray (2x2):")
print(my_array[:2, :2])

```

```
Element at (0, 0): 1
Element at (1, 2): 6
Row 0: [1 2 3]
Column 1: [2 5 8]
Subarray (2x2):
[[1 2]
 [4 5]]
```

In [ ]:
```python

```

In [ ]:
```python

```

In [ ]:
```python

```