1. Write a program to find the sum of all elements of a list.

In [1]:
```
print('Q1\nRollno: 21052410')
def find_sum_of_elements(my_list):
    total_sum = sum(my_list)
    return total_sum

sample_list = [1, 2, 3, 4, 5]
result = find_sum_of_elements(sample_list)

print(f"The sum of the elements in the list is: {result}")
```

```
Q1
Rollno: 21052410
The sum of the elements in the list is: 15
```

2. Write a program to find the maximum and minimum element of a list.

In [2]:
```
print('Q2\nRollno: 21052410')
def find_max_min_elements(my_list):
    max_element = max(my_list)
    min_element = min(my_list)
    return max_element, min_element

sample_list = [3, 1, 7, 4, 5, 9, 2]
max_result, min_result = find_max_min_elements(sample_list)

print(f"The maximum element in the list is: {max_result}")
print(f"The minimum element in the list is: {min_result}")
```

```
Q2
Rollno: 21052410
The maximum element in the list is: 9
The minimum element in the list is: 1
```

3. Write a program to find the position of maximum and minimum element's position in a list.

In [3]:
```
print('Q3\nRollno: 21052410')
def find_max_min_positions(my_list):
    max_element = max(my_list)
    min_element = min(my_list)

    max_position = my_list.index(max_element)
    min_position = my_list.index(min_element)

    return max_position, min_position

sample_list = [3, 1, 7, 4, 5, 9, 2]
max_position, min_position = find_max_min_positions(sample_list)

print(f"The position of the maximum element in the list is: {max_positi
print(f"The position of the minimum element in the list is: {min_positi
```

```
Q3
Rollno: 21052410
The position of the maximum element in the list is: 5
The position of the minimum element in the list is: 1
```

4. Write a program to print all the element in a list that are greater than a given value.

In [4]:
```
print('Q4\nRollno: 21052410')
def print_elements_greater_than_value(my_list, value):
    filtered_elements = [element for element in my_list if element > va
    return filtered_elements

sample_list = [3, 1, 7, 4, 5, 9, 2]
given_value = 5

result = print_elements_greater_than_value(sample_list, given_value)

print(f"Elements in the list greater than {given_value}: {result}")
```

```
Q4
Rollno: 21052410
Elements in the list greater than 5: [7, 9]
```

5. Write a program to print all the common elements of 2 list.

In [5]:
```
print('Q5\nRollno: 21052410')
def find_common_elements(list1, list2):
    common_elements = list(set(list1) & set(list2))
    return common_elements

list1 = [1, 2, 3, 4, 5]
list2 = [3, 4, 5, 6, 7]

common_elements_result = find_common_elements(list1, list2)

print(f"Common elements of the two lists: {common_elements_result}")
```

```
Q5
Rollno: 21052410
Common elements of the two lists: [3, 4, 5]
```

6. Write a program to reverse a list.

In [6]:
```
print('Q6\nRollno: 21052410')
def reverse_list(my_list):
    reversed_list = my_list[::-1]
    return reversed_list

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

reversed_result = reverse_list(sample_list)

print(f"Original List: {sample_list}")
print(f"Reversed List: {reversed_result}")
```

```
Q6
Rollno: 21052410
Original List: [1, 2, 3, 4, 5]
Reversed List: [5, 4, 3, 2, 1]
```

7. Write a program to find the second largest and second smallest value from a list.

In [7]:
```
print('Q7\nRollno: 21052410')
def find_second_largest_smallest(my_list):
    sorted_list = sorted(my_list)

    second_smallest = sorted_list[1]
    second_largest = sorted_list[-2]

    return second_largest, second_smallest

sample_list = [3, 1, 7, 4, 5, 9, 2]

second_largest, second_smallest = find_second_largest_smallest(sample_l

print(f"The second largest value in the list is: {second_largest}")
print(f"The second smallest value in the list is: {second_smallest}")
```

```
Q7
Rollno: 21052410
The second largest value in the list is: 7
The second smallest value in the list is: 2
```

8. Write a program to find the average of all elements in a list.

In [8]:
```
print('Q8\nRollno: 21052410')
def find_average(my_list):
    total_sum = sum(my_list)
    average = total_sum / len(my_list)

    return average

sample_list = [1, 2, 3, 4, 5]
average_result = find_average(sample_list)

print(f"The average of the elements in the list is: {average_result}")
```

```
Q8
Rollno: 21052410
The average of the elements in the list is: 3.0
```

9. Write a program to to generate random numbers within a given range and store it in a list.

In [9]:
```
print('Q9\nRollno: 21052410')
import random

def generate_random_numbers(start, end, count):
    random_numbers = [random.randint(start, end) for _ in range(count)]
    return random_numbers

start_range = 1
end_range = 100
number_of_random_numbers = 5

random_numbers_list = generate_random_numbers(start_range, end_range, r

print(f"Generated random numbers: {random_numbers_list}")
```

```
Q9
Rollno: 21052410
Generated random numbers: [30, 31, 32, 14, 61]
```

10. Write a program to split the even and odd elements into 2 different lists.

In [10]:
```
print('Q10\nRollno: 21052410')
def split_even_odd_elements(my_list):
    even_elements = [element for element in my_list if element % 2 == 0
    odd_elements = [element for element in my_list if element % 2 != 0]
    return even_elements, odd_elements

sample_list = [1, 2, 3, 4, 5, 6, 7, 8, 9]

even_elements_list, odd_elements_list = split_even_odd_elements(sample_

print(f"Original List: {sample_list}")
print(f"Even Elements: {even_elements_list}")
print(f"Odd Elements: {odd_elements_list}")
```

```
Q10
Rollno: 21052410
Original List: [1, 2, 3, 4, 5, 6, 7, 8, 9]
Even Elements: [2, 4, 6, 8]
Odd Elements: [1, 3, 5, 7, 9]
```

11. Write a Python program to create a tuple

In [11]:
```
print('Q11\nRollno: 21052410')
my_tuple = (1, 2, 'hello', 3.14, True)

print("Tuple:", my_tuple)
```

```
Q11
Rollno: 21052410
Tuple: (1, 2, 'hello', 3.14, True)
```

12. Write a Python program to create a tuple with different data types.

In [12]:
```
print('Q12\nRollno: 21052410')
mixed_tuple = (1, 'hello', 3.14, True, [5, 6, 7])

print("Mixed Tuple:", mixed_tuple)
```

```
Q12
Rollno: 21052410
Mixed Tuple: (1, 'hello', 3.14, True, [5, 6, 7])
```

13. Write a Python program to unpack a tuple in several variables.

In [13]:
```
print('Q13\nRollno: 21052410')
my_tuple = (1, 'hello', 3.14)

a, b, c = my_tuple

print("Variable a:", a)
print("Variable b:", b)
print("Variable c:", c)
```

```
Q13
Rollno: 21052410
Variable a: 1
Variable b: hello
Variable c: 3.14
```

14. Write a Python program to add an item in a tuple.

In [14]:
```
print('Q14\nRollno: 21052410')
original_tuple = (1, 2, 3, 4, 5)
new_item = 6

updated_tuple = original_tuple + (new_item,)

print("Original Tuple:", original_tuple)
print("Updated Tuple:", updated_tuple)
```

```
Q14
Rollno: 21052410
Original Tuple: (1, 2, 3, 4, 5)
Updated Tuple: (1, 2, 3, 4, 5, 6)
```

15. Write a Python program to convert a tuple to a string.

In [15]:
```
print('Q15\nRollno: 21052410')
my_tuple = ('Hello', ' ', 'World', '!')

result_string = ''.join(map(str, my_tuple))

print("Original Tuple:", my_tuple)
print("Converted String:", result_string)
```

```
Q15
Rollno: 21052410
Original Tuple: ('Hello', ' ', 'World', '!')
Converted String: Hello World!
```

16. Write a Python program to get the 4th element and 4th element from last of a tuple.

In [16]:
```
print('Q16\nRollno: 21052410')
my_tuple = (1, 2, 3, 4, 5, 6, 7, 8)

fourth_element = my_tuple[3]

fourth_from_last = my_tuple[-4]

print("Original Tuple:", my_tuple)
print("4th Element:", fourth_element)
print("4th Element from Last:", fourth_from_last)
```

```
Q16
Rollno: 21052410
Original Tuple: (1, 2, 3, 4, 5, 6, 7, 8)
4th Element: 4
4th Element from Last: 5
```

17. Write a Python program to find the repeated items of a tuple.

In [17]:
```
print('Q17\nRollno: 21052410')
def find_repeated_items(my_tuple):
    repeated_items = set()
    unique_items = set()

    for item in my_tuple:
        if item in unique_items:
            repeated_items.add(item)
        else:
            unique_items.add(item)

    return list(repeated_items)

sample_tuple = (1, 2, 3, 2, 4, 5, 6, 6, 7)

repeated_items_list = find_repeated_items(sample_tuple)

print("Original Tuple:", sample_tuple)
print("Repeated Items:", repeated_items_list)
```

```
Q17
Rollno: 21052410
Original Tuple: (1, 2, 3, 2, 4, 5, 6, 6, 7)
Repeated Items: [2, 6]
```

18. Write a Python program to check whether an element exists within a tuple.

In [18]:
```
print('Q18\nRollno: 21052410')
def check_element_existence(my_tuple, element):
    return element in my_tuple

sample_tuple = (1, 2, 3, 4, 5)

element_to_check = 3

element_exists = check_element_existence(sample_tuple, element_to_check

print("Original Tuple:", sample_tuple)
print(f"Does {element_to_check} exist in the tuple? {element_exists}")
```

```
Q18
Rollno: 21052410
Original Tuple: (1, 2, 3, 4, 5)
Does 3 exist in the tuple? True
```

19. Write a Python program to slice a tuple.

In [19]:
```
print('Q19\nRollno: 21052410')
my_tuple = (1, 2, 3, 4, 5, 6, 7, 8, 9, 10)

sliced_tuple = my_tuple[2:7]

print("Original Tuple:", my_tuple)
print("Sliced Tuple:", sliced_tuple)
```

```
Q19
Rollno: 21052410
Original Tuple: (1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
Sliced Tuple: (3, 4, 5, 6, 7)
```

20. Write a Python program to find the index of an item of a tuple.

In [20]:
```
print('Q20\nRollno: 21052410')
def find_item_index(my_tuple, item):
    try:
        index = my_tuple.index(item)
        return index
    except ValueError:
        return f"{item} not found in the tuple"

sample_tuple = (10, 20, 30, 40, 50)

item_to_find = 30

item_index = find_item_index(sample_tuple, item_to_find)

print("Original Tuple:", sample_tuple)
print(f"Index of {item_to_find} in the tuple: {item_index}")
```

```
Q20
Rollno: 21052410
Original Tuple: (10, 20, 30, 40, 50)
Index of 30 in the tuple: 2
```

21. Write a Python program to find the length of a tuple.

In [21]:
```
print('Q21\nRollno: 21052410')
my_tuple = (1, 2, 3, 4, 5)

tuple_length = len(my_tuple)

print("Original Tuple:", my_tuple)
print("Length of the Tuple:", tuple_length)
```

```
Q21
Rollno: 21052410
Original Tuple: (1, 2, 3, 4, 5)
Length of the Tuple: 5
```

22. Write a Python program to convert a tuple to a dictionary.

In [22]:
```
print('Q22\nRollno: 21052410')
def tuple_to_dict(my_tuple):
    my_dict = dict(my_tuple)
    return my_dict

sample_tuple = (("a", 1), ("b", 2), ("c", 3))

converted_dict = tuple_to_dict(sample_tuple)

print("Original Tuple:", sample_tuple)
print("Converted Dictionary:", converted_dict)
```

```
Q22
Rollno: 21052410
Original Tuple: (('a', 1), ('b', 2), ('c', 3))
Converted Dictionary: {'a': 1, 'b': 2, 'c': 3}
```

23. Write a Python program to unzip a list of tuples into individual lists.

In [23]:
```
print('Q23\nRollno: 21052410')
list_of_tuples = [(1, 'one'), (2, 'two'), (3, 'three')]

numbers, words = zip(*list_of_tuples)

print("List of Tuples:", list_of_tuples)
print("Numbers List:", list(numbers))
print("Words List:", list(words))
```

```
Q23
Rollno: 21052410
List of Tuples: [(1, 'one'), (2, 'two'), (3, 'three')]
Numbers List: [1, 2, 3]
Words List: ['one', 'two', 'three']
```

24. Write a Python program to reverse a tuple.

In [24]:
```
print('Q24\nRollno: 21052410')
my_tuple = (1, 2, 3, 4, 5)

reversed_tuple = my_tuple[::-1]

print("Original Tuple:", my_tuple)
print("Reversed Tuple:", reversed_tuple)
```

```
Q24
Rollno: 21052410
Original Tuple: (1, 2, 3, 4, 5)
Reversed Tuple: (5, 4, 3, 2, 1)
```

25. Write a Python program to convert a list of tuples into a dictionary.

In [26]:
```
print('Q25\nRollno: 21052410')
def list_of_tuples_to_dict(list_of_tuples):
    my_dict = dict(list_of_tuples)
    return my_dict
sample_list_of_tuples = [("a", 1), ("b", 2), ("c", 3)]

converted_dict = list_of_tuples_to_dict(sample_list_of_tuples)

print("List of Tuples:", sample_list_of_tuples)
print("Converted Dictionary:", converted_dict)
```

```
Q25
Rollno: 21052410
List of Tuples: [('a', 1), ('b', 2), ('c', 3)]
Converted Dictionary: {'a': 1, 'b': 2, 'c': 3}
```

26. Write a Python program to print a tuple with string formatting.
Sample tuple : (100, 200, 300)
Output : This is a tuple (100, 200, 300)

In [32]:
```
print('Q26\nRollno: 21052410')
my_tuple = (100, 200, 300)

output_string = f"This is a tuple {my_tuple}"

print(output_string)
```

```
Q26
Rollno: 21052410
This is a tuple (100, 200, 300)
```

27. Write a Python program to replace last value of tuples in a list.
Sample list: [(10, 20, 40), (40, 50, 60), (70, 80, 90)]
Expected Output: [(10, 20, 100), (40, 50, 100), (70, 80, 100)]

In [28]:
```
print('Q27\nRollo: 21052410')
list_of_tuples = [(10, 20, 40), (40, 50, 60), (70, 80, 90)]

new_value = 100

updated_list_of_tuples = [(t[0], t[1], new_value) for t in list_of_tupl
print("Original List of Tuples:", list_of_tuples)
print("Updated List of Tuples:", updated_list_of_tuples)
```

```
Q27
Rollo: 21052410
Original List of Tuples: [(10, 20, 40), (40, 50, 60), (70, 80, 90)]
Updated List of Tuples: [(10, 20, 100), (40, 50, 100), (70, 80, 100)]
```

28. Write a Python program to replace last value of tuples in a list.
    Sample data: [(), (), ('',), ('a', 'b'), ('a', 'b', 'c'), ('d')]
    Expected output: [('',), ('a', 'b'), ('a', 'b', 'c'), 'd']

In [34]:
```
print('Q28\nRollno: 21052410')
list_of_tuples = [(), (), ('',), ('a', 'b'), ('a', 'b', 'c'), ('d')]
updated_list_of_tuples = [t for t in list_of_tuples if len(t) > 0 and t

print("Original List of Tuples:", list_of_tuples)
print("Updated List of Tuples:", updated_list_of_tuples)
```

```
Q28
Rollno: 21052410
Original List of Tuples: [(), (), ('',), ('a', 'b'), ('a', 'b', 'c'),
'd']
Updated List of Tuples: [('a', 'b'), ('a', 'b', 'c'), 'd']
```

29. Write a Python program to sort a tuple by its float element.
    Sample data: [('item1', '12.20'), ('item2', '15.10'), ('item3', '24.5')]
    Expected Output: [('item3', '24.5'), ('item2', '15.10'), ('item1', '12.20')]

In [29]:
```
print('Q29\nRollno: 21052410')
data = [('item1', '12.20'), ('item2', '15.10'), ('item3', '24.5')]

sorted_data = sorted(data, key=lambda x: float(x[1]), reverse=True)

print("Original Data:", data)
print("Sorted Data:", sorted_data)
```

```
Q29
Rollno: 21052410
Original Data: [('item1', '12.20'), ('item2', '15.10'), ('item3', '24.
5')]
Sorted Data: [('item3', '24.5'), ('item2', '15.10'), ('item1', '12.2
0')]
```

30. Write a Python program to count the elements in a list until an element is a tuple.

In [30]:

```
print('Q30\nRollno: 21052410')
my_list = [1, 2, 'a', (3, 4), 'b', 'c', (5, 6, 7)]

count = 0
for element in my_list:
    if isinstance(element, tuple):
        break
    count += 1

print("Original List:", my_list)
print(f"Number of elements until a tuple is encountered: {count}")
```

```
Q30
Rollno: 21052410
Original List: [1, 2, 'a', (3, 4), 'b', 'c', (5, 6, 7)]
Number of elements until a tuple is encountered: 3
```