Daily Assignment-2

1) Find the size of the tuple.

CODE:

```
print("2203031240711")
my_tuple = (1, "hello", 3.14,"hi")
size = len(my_tuple)
print(f"The size of the tuple is: {size}")
```

```
220303124071l
The size of the tuple is: 4

=== Code Execution Successful ===
```

2) Find the sum of the tuple

CODE:

```
Print("2203031240711")
tuple=(11,23,34,65,78)
sum=sum(tuple)
print("the sum of number in the given tuple : ",{sum})
```

```
2203031240711
the sum of number in the given tuple :  {211}

=== Code Execution Successful ===
```

3) Create a list of tuples from given list consisting number and its square in each tuple.

CODE:

```
def list_of_tuples(numbers):
   return [(number, number ** 2) for number in numbers]
tuple1 = [1, 2, 3, 4, 5]
result = list_of_tuples(tuple1)
print("tuple1= ",result)
tuple2=[6,7,8,9,0]
```

```
res=list_of_tuples(tuple2)
print("tuple2= ",res)
```

```
2203031240711
tuple1=   [(1, 1), (2, 4), (3, 9), (4, 16), (5, 25)]
tuple2=   [(6, 36), (7, 49), (8, 64), (9, 81), (0, 0)]

=== Code Execution Successful ===
```

4) Write a program to find whether the given number is even or odd.

CODE:

```
Print("2203031240711")
tuple=[1,2,3,4,5,6,7,8,9]
for i in tuple:
   if i%2==0:
      print("even",i)
   else:
      print("odd",i)
```

```
2203031240711
odd 1
even 2
odd 3
even 4
odd 5
even 6
odd 7
even 8
odd 9

=== Code Execution Successful ===
```

5)Write a program for an iterative factorial of a number.

CODE:

```
def factorial(num):
   if num<0:
      print("enter a valid number")
   elif num==1:
      print("factorial of 1 = 1")
   else:
```

```
        fact=i*fact
    return fact
a,b=5,7
res=factorial(a)
res2=factorial(b)
print("the factorial of ",a,"is",res)
print("the factorial of ",b," is ",res2)
```

```
2203031240711
the factorial of  5 is 120
the factorial of  7  is  5040

=== Code Execution Successful ===
```

---

5) write a python program to sort the dictionary by key or a value.

CODE:

```
import operator
print("2203031240711")
def sort_key(input_dict):
   return dict(sorted(input_dict.items()))
def sort_value(input_dict):
   return dict(sorted(input_dict.items(), key=operator.itemgetter(1)))
sample_dict = {'apple': 3, 'banana': 1, 'cherry': 2}
sorted_by_key = sort_key(sample_dict)
print("Sorted by key:", sorted_by_key)
sorted_by_value = sort_value(sample_dict)
print("Sorted by value:", sorted_by_value)
```

```
2203031240711
Sorted by key: {'apple': 3, 'banana': 1, 'cherry': 2}
Sorted by value: {'banana': 1, 'cherry': 2, 'apple': 3}

=== Code Execution Successful ===
```

---

6) Write the python program to find sum of all the items in the dictionary.

CODE:

```
dict_1= {'apple': 5, 'banana': 9, 'cherry': 7}
result=sum(dict_1.values())
print(f"The sum of all values in the dictionary-1 is {result}")
```

```
dict_2={'hi':1,'hello':4,'namaste':8}
res=sum(dict_2.values())
print(f"The sum of all values in the dictionary-2 is {res}")
```

```
2203031240711
Sorted by key: {'apple': 3, 'banana': 1, 'cherry': 2}
Sorted by value: {'banana': 1, 'cherry': 2, 'apple': 3}

=== Code Execution Successful ===
```

---

7) write a python program to find the size of the dictionary.

CODE:

```
8) print("2203031240711")
dict_1 = {'apple': 3, 'banana': 1, 'cherry': 2,'grape':4,'avacado':5}
result = len(dict_1)
print(f"The size of the dictionary-1 is {result}")
dict_2={'hi':1,'hello':4,'namaste':8}
res=len(dict_2)
print(f"The size of the dictionary-2 is {res}")
```

```
2203031240711
The size of the dictionary-1 is 5
The size of the dictionary-2 is 3

=== Code Execution Successful ===
```

9)write a python program to find the size of a set.

CODE:

```
print("2203031240711")
set_1 = {1, 2, 3, 4, 5}
result = len(set_1)
print(f"The size of set-1 is {result}")
set_2={12, 13, 14, 45, 34, 56}
res=len(set_2)
print(f"the size of set-2 is {res}")
```

```
2203031240711
The size of set-1 is 5
the size of set-2 is 6

=== Code Execution Successful ===
```

Daily Assignment -3

1) Write a python program for the addition and subtraction of the matrices.

CODE:

```
import numpy

A = numpy.array([[1, 2], [3, 4]])
B = numpy.array([[4, 5], [6, 7]])


print("Printing elements of first matrix")
print(A)
print("Printing elements of second matrix")
print(B)


print("Addition of two matrix")
print(numpy.add(A, B))


print("Subtraction of two matrix")
print(numpy.subtract(A, B))
```

```
2203031240711
Printing elements of first matrix
[[1 2]
 [3 4]]
Printing elements of second matrix
[[4 5]
 [6 7]]
Addition of two matrix
[[ 5  7]
 [ 9 11]]
Subtraction of two matrix
[[-3 -3]
 [-3 -3]]

=== Code Execution Successful ===
```

2) Write a python program for row-wise addition in tuple matrix.

CODE:

```
def row_wise_addition(matrix):
    num_rows = len(matrix)
    num_cols = len(matrix[0])
    result = []
    for i in range(num_rows):
        row_sum = sum(matrix[i][j] for j in range(num_cols))
        result.append(row_sum)
    return result
tuple_matrix = [
    (1, 2, 3),
    (4, 5, 6),
    (7, 8, 9)
]
result = row_wise_addition(tuple_matrix)
print("Row-wise addition result:")
print(result)
```

```
2203031240711
Row-wise addition result:
[6, 15, 24]

=== Code Execution Successful ===
```

---

3) Write a python program to print multiple arguments.

CODE:

```
def print_arguments(*args):
    for arg in args:
        print(arg)
print_arguments("Hello", "world", 123, [1, 2, 3])
```

```
2203031240711
Hello
world
123
[1, 2, 3]

=== Code Execution Successful ===
```

---

4) Write a python program to print the power of a number using recursion.

CODE:

```
def power(N, P):
    if P == 0:
        return 1
    return (N * power(N, P - 1))
N,P=5,2
print(f"{N} power {P} is ",power(N,P))
a,b=12,3
print(f"{a} power {b} is ",power(a,b))
```

```
2203031240711
5 power 2 is  25
12 power 3 is  1728

=== Code Execution Successful ===
```