

## 1. Write a program in Python

- To accept 8 numbers, store them in a list NUMS
- To add adjacent pair of elements of NUMS and store it in another list MNUMS
- To copy the content of MNUMS on alternate places in the NUMS starting from second place of NUMS.
- To display content of NUMS as well as MNUMS

**Example:**

```
# -----#
# List-Program No      : L5-01
# Developed By         : Shesh Shiromani
# Date                 : 29th November 2024
# -----#
l = []
a = []
for i in range(8):
    x = int(input("Enter: "))
    l.append(x)

print(l)
for i in range(0, len(l)-1, 2):
    a.append(l[i] + l[i+1])
print(a)
```

2. Given a tuple **ALL** = (5, 8, 2, 'apple', 'banana', 'Grapes'), Write a Python code
- to print the second to fourth elements of the tuple **ALL**.
  - to print the content of **ALL** with its content reversed.

```
# -----#
# List-Program No      : L5-02
# Developed By         : Shesh Shiromani
# Date                 : 29th November 2024
# -----#

ALL = (5, 8, 2, 'apple', 'banana', 'Grapes')
a = ()
print(ALL[2:5])
for i in range(len(ALL)-1, -1, -1):
    a+=(ALL[i],)
print(a)
```

## 3. Consider the following tuples

**Main** = ('Roti', 'Sabji', 'Dal')

**Addon** = ('Papad', 'Salad')

Write a Python code

- to concatenate content of both the tuples and create another tuple **Meal**.
- to display the content of **Meal**.

```
# -----#
# List-Program No      : L5-03
# Developed By         : Shesh Shiromani
# Date                 : 29th November 2024
# -----#

Main = ('Roti', 'Sabji', 'Dal')
Addon = ('Papad', 'Salad')
```

```
Meal = Main + Addon
print(Meal)
```

4. Write a Python code to unpack the content of tuple ('Neeraj', 'A-8, ABC Nagar', 2500, 'Male', True) and store the results in variables Name, Address, Fee, Gender and Indian. Display the unpacked content from the variables.

```
# -----#
# List-Program No      : L5-04
# Developed By         : Shesh Shiromani
# Date                 : 29th November 2024
# -----#

T = ('Neeraj', 'A-8, ABC Nagar', 2500, 'Male', True)
Name = T[0]
Address = T[1]
Fee = T[2]
Gender = T[3]
Indian = T[4]

print(Name)
print(Address)
print(Fee)
print(Gender)
print(Indian)
```

5. Given the tuple **scores** = (1, 2, 3, 4, 2, 5, 2, 3, 4, 2, 1). Write a Python code to find the count of occurrences of each score from the tuple and display the result in the following format.  
**Score Frequency**

```
# -----#
# List-Program No      : L5-05
# Developed By         : Shesh Shiromani
# Date                 : 29th November 2024
# -----#

l = (1, 2, 3, 4, 2, 5, 2, 3, 4, 2, 1)
for i in range(min(l), max(l)+1):
    print(i, l.count(i))
```

6. Write a program in Python

- To accept item numbers of 5 items, store them in a list INO
- To accept item names of 5 items, store them in a list INAME
- To create a dictionary ITEM with Key-Value pair with Keys from INO and corresponding Values from INAME
- To display the content of INO and INAME
- To display the content of ITEM in ascending order of item numbers

```
# -----#
# List-Program No      : L5-06
# Developed By         : Shesh Shiromani
# Date                 : 29th November 2024
# -----#

INO = []
print("Enter item numbers of 5 items:")
```

```

for _ in range(5):
    INO.append(int(input()))
INAME = []
print("Enter item names of 5 items:")
for _ in range(5):
    INAME.append(input())
ITEM = {INO[i]: INAME[i] for i in range(5)}
print("INO:", INO)
print("INAME:", INAME)
print("ITEM ( by item numbers):")
for item_no in sorted(ITEM):
    print(item_no, ITEM[item_no])

```

7. **A. Assign the following contents in a tuple Names**

"JAYA", "AMAR", "PRIYA", "AKBAR", "RESHMA", "ANTHONY"

**B. Assign the following contents in a tuple Marks**

75, 56, 86, 92, 65, 86

- Create a dictionary **Results** with keys from tuple **Names** and corresponding values from tuple **Marks**.
- Display the content of **Results** with keys arranged alphabetically in ascending order.
- Display the content of values of **Results** arranged in ascending order.
- Create a new dictionary named **Toppers** to store only such items of dictionary **Results** where marks are more than 80.

```
T = ("JAYA", "AMAR", "PRIYA", "AKBAR", "RESHMA", "ANTHONY")
```

```
M = (75, 56, 86, 92, 65, 86)
```

```

Results = {}
for i in range(len(Names)):
    Results[Names[i]] = M[i]

sorted_results = dict(sorted(Results.items()))
for key, value in sorted_results.items():
    print(key, value)
print()

```

```

sorted_values = sorted(Results.values())
for value in sorted_values:
    print(value)
print()

```

```

Toppers = {}
for key, value in Results.items():
    if value > 80:
        Toppers[key] = value

for key, value in Toppers.items():
    print(key, value)
print()

```

8. **Write a Python code to perform the following:**

- To accept 8 numbers in a loop, store them in a Tuple T (with the help of re-assignment method)
- To display the content of T in reverse order
- To add and display the sum of values stored in T
- To find and display minimum and maximum values present in T
- To display sum of each adjacent pair of values
- To find those pairs (any pair 1st-2nd, 1st-5th, 3rd-6th,...) of values from the content of T, whose sum is the same as one of the values in the tuple T.

```
# -----#
# List-Program No          : L5-08
# Developed By             : Shesh Shiromani
# Date                     : 29th November 2024
# -----#

T = tuple(int(input("Enter number {}: ".format(i + 1))) for i in range(8))

print("Content of T in reverse order:")
print(T[::-1])

print("Sum of values in T:")
print(sum(T))

print("Minimum value in T:")
print(min(T))

print("Maximum value in T:")
print(max(T))
```

9. Write a Python code to perform the following:

- To initialize a tuple WD containing (1,2,3,4,5,6,7)
- To initialize a list WDN containing ['SUN', 'MON', 'TUE', 'WED', 'THU', 'FRI', 'SAT']
- To create a dictionary W with key-value pairs with corresponding values from WD and WDN
- To display content of W
- To re-arrange the content of dictionary in such a way that it becomes as follows:  
{1: 'MON', 2: 'TUE', 3: 'WED', 4: 'THU', 5: 'FRI', 6: 'SAT', 7: 'SUN'}
- To display the content of W
- To copy the partial content of W in dictionaries **MyDays** and **OfficeDays**, **MyDays** should have content from keys 2,4 and 7 and rest from W to become the content of **OfficeDays**.
- To display the contents of **MyDays** and **OfficeDays**

```
# -----#
# List-Program No          : L5-09
# Developed By             : Shesh Shiromani
# Date                     : 29th November 2024
# -----#

WD = (1, 2, 3, 4, 5, 6, 7)
WDN = ['SUN', 'MON', 'TUE', 'WED', 'THU', 'FRI', 'SAT']
W = {}

for i in range(len(WD)):
    W[WD[i]] = WDN[i]

print("Content of W:")
print(W)
W = {1: 'MON', 2: 'TUE', 3: 'WED', 4: 'THU', 5: 'FRI', 6: 'SAT', 7: 'SUN'}
print("\nRe-arranged content of W:")
print(W)
my_days_indices = [1, 3, 6] # Indices of 2, 4, 7 in original list (1-based)
MyDays = {}
OfficeDays = {}
index = 0
for day in W:
    if index in my_days_indices:
        MyDays[WD[index]] = W[day]
    else:
        OfficeDays[WD[index]] = W[day]
```

```

        index += 1

    print("\nMyDays:")
    print(MyDays)
    print("\nOfficeDays:")
    print(OfficeDays)

```

10. Write a Python code to perform the following operations:

- To initialize a tuple **TL**=('RED', 'YELLOW', 'GREEN')
- To accepts names of 10 colors from user and store them in a list **CL**
- To display the color names from **CL** along with corresponding message "TRAFFIC LIGHT" and "NOT TRAFFIC LIGHT" after checking from the content of **TL**
- To initialize another tuple **TM**=('STOP', 'BE READY TO START/STOP', 'GO')
- To create a dictionary **TLM** by combining corresponding key-value pairs from **TL** and **TM**.
- To display the content of **TLM**

```

# -----#
# List-Program No       : L5-10
# Developed By          : Shesh Shiromani
# Date                  : 29th November 2024
# -----#

TL = ('RED', 'YELLOW', 'GREEN')
CL = []
for i in range(10):
    color = input("Enter color name " + str(i + 1) + ": ")
    CL.append(color.upper())

print("Color Names and Traffic Light Status:")
for color in CL:
    if color == 'RED':
        print(color + ": TRAFFIC LIGHT (STOP)")
    elif color == 'YELLOW':
        print(color + ": TRAFFIC LIGHT (BE READY TO START/STOP)")
    elif color == 'GREEN':
        print(color + ": TRAFFIC LIGHT (GO)")
    else:
        print(color + ": NOT TRAFFIC LIGHT")

TM = ('STOP', 'BE READY TO START/STOP', 'GO')
TLM = {}
TLM[TL[0]] = TM[0]
TLM[TL[1]] = TM[1]
TLM[TL[2]] = TM[2]
print("\nContent of TLM:")
print(TLM)

```