

ANSWERS

PYTHON ASSIGNMENT 6 – Data Structure: List & Tuple

1) A python program to create a lists with different types of element.

Solution:

```
lists = [1, 'Sarika', 2, 'Pratiksha', 3, 'Sayali', 4, 'Chinmay']  
print(lists)
```

2). A python program to create a list using range function.

Solution:

```
print("Get numbers from Range 1 to 10")  
for i in range(1, 11):  
    print(i, end=', ')
```

3) A python program to access the list elements using loops.

Solution:

```
# Way 1:  
  
list1 = ['Rose', 'Lily', 'Lotus', 'Jasmine', 'Orchid', 'Daisy']  
  
# Using for loop  
for flowers in list1:  
    print(flowers, end=" - ")
```



```
# Way 2:

list2 = ['Rose', 'Lily', 'Lotus', 'Jasmine', 'Orchid', 'Daisy']

# getting length of list
length = len(list2)

# Iterating the index
# same as 'for i in range(len(list))'
for i in range(length):
    print(list2[i], end=" - ")

# Way 3:

list3 = ['Rose', 'Lily', 'Lotus', 'Jasmine', 'Orchid', 'Daisy']

# Getting length of list
length = len(list3)
i = 0

# Iterating using while loop
while i < length:
    print(list3[i], end=" - ")
    i += 1
```

4) A python program to display the elements of a list in reverse order.

Solution:

```
list1 = ['Rose', 'Lily', 'Lotus', 'Jasmine', 'Orchid', 'Daisy']
reverseList = list1[::-1]
print(reverseList)
```

5) A python program to understand list processing methods.

Solution:



```
list1 = [1, 2, 3, 4, 5]

# Adds List Element as value of List.
list1.append(6)
print(list1)

# Inserts an elements at specified position.
list1.insert(2, 'Gauri')
print(list1)

# Adds contents to List2 to the end of List1.
list2 = [7, 8, 9, 10]
list1.extend(list2)
print(list1)

# Calculates sum of all the elements of List.
List = [1, 2, 3, 4, 5]
print(sum(List))

# Calculates total occurrence of given element of List.
List = [1, 2, 3, 1, 2, 1, 2, 3, 2, 1]
print(List.count(1))

# Calculates total length of List
List = [1, 2, 3, 1, 2, 1, 2, 3, 2, 1]
print(len(List))

# Find index of first occurrence.
List = [1, 2, 3, 1, 2, 1, 2, 3, 2, 1]
print(List.index(2))
```

6) A python program to find the maximum and minimum elements in a list of elements.

Solution:

```
# Maximum:

List = [23, 44, 45, 3, 53, 105, 4, 25]
print("Maximum Element in List is:", max(List))
```



```
# Minimum:
List = [23, 44, 45, 3, 53, 105, 4, 25]
print("Minimum Element in List is:", min(List))
```

7) A python program to sort the list elements using bubble sort technique.

Solution:

```
# Way 1:

arr = [64, 34, 25, 12, 22, 11, 90]
n = len(arr)

# Traverse through all array elements
for i in range(n):
    # Last i elements are already in place
    for j in range(0, n - i - 1):
        # traverse the array from 0 to n-i-1
        # Swap if the element found is greater
        # than the next element
        if arr[j] > arr[j + 1]:
            arr[j], arr[j + 1] = arr[j + 1], arr[j]

# Driver code to test above
print("Sorted array is:")
for i in range(len(arr)):
    print("%d" % arr[i], end=" ")

# Way 2:

a = [16, 19, 11, 15, 10, 12, 14]
print("Original List:")
print(a)
# repeating loop len(a)(number of elements) number of times
for j in range(len(a)):
    # initially swapped is false
    swapped = False
    i = 0
    while i < len(a) - 1:
```



```
# comparing the adjacent elements
if a[i] > a[i + 1]:
    # swapping
    a[i], a[i + 1] = a[i + 1], a[i]
    # Changing the value of swapped

    swapped = True
    i = i + 1
# if swapped is false then the list is sorted
# we can stop the loop
if swapped == False:
    break

print("Sorted List:")
print(a)

# Way 3:

number = [16, 19, 11, 15, 10, 12, 14]
print("Original List:")
print(number)
n = len(number)
a = []
for i in number:
    a.append(i)

for i in range(n):
    for j in range(n - i - 1):
        if a[j] > a[j + 1]:
            temp = a[j]
            a[j] = a[j + 1]
            a[j + 1] = temp

print("The Sorted List in Ascending Order : ")
print(a)
```

8) A python program to know how many times an element occurred in the list.

Solution:



```
# Way 1:

List = [1, 2, 3, 1, 2, 1, 2, 3, 2, 1]
print(List.count(1))

# Way 2:

lst = [8, 6, 8, 10, 8, 20, 10, 8, 8]
count = 0
x = 8
for ele in lst:
    if ele == x:
        count = count + 1
print('x has occurred {} times'.format(x))
```

9) A python program to find the common elements in two lists.

Solution:

```
#Way 1:

a = [2, 3, 4, 5]
b = [3, 5, 7, 9]

c = [value for value in a if value in b]
print(c)

#Way 2:

a = [2, 3, 4, 5]
b = [3, 5, 7, 9]
for value in a:
    if value in b:
        print(value)

#Way 3:
a = [1, 2, 3, 4, 5]

b = [5, 6, 7, 8, 9]
```



```
a_set = set(a)
b_set = set(b)
if (a_set & b_set):
    print(a_set & b_set)
else:
    print("No common elements")
```

10) A python program to create list with employee data and then retrieve a particular employee details.

Solution:

```
list1 = [
    {"name": "Chinmay", "age": 29, "empId": 89, "skills": "JavaScript"},
    {"name": "Sarika", "age": 22, "empId": 87, "skills": "Python"},
    {"name": "Sayali", "age": 26, "empId": 82, "skills": "Git"},
    {"name": "Tejaswi", "age": 25, "empId": 86, "skills": "Jquery"},
    {"name": "Amit", "age": 27, "empId": 83, "skills": "DataScience"}
]

name = input("Please enter the name you want to search...!!!!")

for listName in list1:
    if listName.get('name') == name:
        for key, value in listName.items():
            print(" {} and : {} ".format(key, value))
```

11) A python program to create nested list and display its elements.

Solution:

```
# Way 1:

list1 = [[0, 1, 2, 3],
         [1, 2, 3, 4],
         [2, 3, 4, 5],

         [3, 4, 5, 6],
         [7, 8, 9, 10]]
```



```
n = len(list1)
list2 = []

for i in range(n):
    # Append an empty sublist inside the list
    list2.append([])
    for j in range(n):

        list2[i].append(j)

print(list2)

# Way 2:

list1 = [[1, 2, 3], [4, 5], [6, 7, 8, 9]]

result = []

for sublist in list1:
    for val in sublist:
        result.append(val)

print(result)
```

12) A python program retrieves elements from a matrix and display them.

Solution:

```
Row = int(input("Enter the number of rows:"))
Column: int = int(input("Enter the number of columns:"))

matrix = []
print("Enter the Entries Row-wise:")

for i in range(Row):
    a = []
    for j in range(Column):

        a.append(int(input()))
    matrix.append(a)
```




```
# print(a) // matrix row

# For printing the matrix
for i in range(Row):
    for j in range(Column):
        print(matrix[i][j], end=" ")
    print()
```

13) A python program to add two matrices and display the sum matrix using lists.

Solution:

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

Y = [[10, 11, 12],
      [13, 14, 15],
      [16, 17, 18]]

Result = [[0, 0, 0],
          [0, 0, 0],
          [0, 0, 0]]

# iterate through rows
for i in range(len(X)):
    # iterate through columns
    for j in range(len(X[0])):
        Result[i][j] = X[i][j] + Y[i][j]
for r in Result:
    print(r)
```

14) A python program to accept elements in the form of a tuple and display their sum and average.

Solution:

```
# Way 1:
```



```
tupleA = [(6, 4, 5), (7, 8, 5, 9), (2, 4, 10)]

print("The original list is : " + str(tupleA))

sum = 0
for sub in tupleA:
    for i in sub:
        sum = sum + i

print(sum)
res = sum / len(tupleA)

print("The Average of tuple list is : " + str(res))

# Way 2:

total = 0
avg = 0
tuple1 = (11, 5, 17, 18, 23)
n = len(tuple1)
# print(n)
for ele in range(0, len(tuple1)):
    total = total + tuple1[ele]
    avg = total/n
print("Sum of all elements in given list: ", total)
print("Average of all elements in given list: ", avg)

# Way 3
total = 0
ele = 0
avg = 0

tuple1 = [11, 5, 17, 18, 23]
n = len(tuple1)
while ele < len(tuple1):
    total = total + tuple1[ele]
    avg = total / n

    ele += 1

print("Sum of all elements in given list: ", total)
```



```
print("Average of all elements in given list: ", avg)
```

15) A python program to find the first occurrence of an element in a tuple.

Solution:

```
tuple1 = (1, 2, 3, 1, 2, 1, 2, 3, 2, 1)
print(tuple1.index(2))
```

16) A python program to sort a tuple with nested tuples.

Solution:

```
# Way 1:

tup1 = (("Arman", 28), ("Tanya", 30), ("Abhishek", 29),
        ("Navya", 21), ("B", "C"))

tup = list(tup1)
n = len(tup)

for i in range(n):
    for j in range(n - i - 1):
        if tup[j][0] > tup[j + 1][0]:
            tup[j], tup[j + 1] = tup[j + 1], tup[j]
print(tup)

# Way 2:

Input1 = ((4, 5), (2, 3), (6, 7), (2, 8))
Input = list(Input1)

print("The original list of tuple is ")
print(Input1)

lst = len(Input)

for i in range(lst):
```



```
for j in range(lst - i - 1):
    if (Input[j][0] + Input[j][1]) > (Input[j + 1][0] + Input[j + 1][1]):
        Input[j], Input[j + 1] = Input[j + 1], Input[j]

Input1 = tuple(Input)
print("\nThe answer is")
print(Input1)
```

17) A python program to insert a new element into a tuple of elements at a specified position.

Solution:

```
tuplex = (4, 6, 2, 8, 3, 1)
print(tuplex)

# tuples are immutable, so you cannot add new elements
# using merge of tuples with the + operator you can add an
# element and it will create a new tuple

tuplex = tuplex + (9,)
print(tuplex)

# adding items in a specific index
tuplex = tuplex[:5] + (15, 20, 25) + tuplex[:5]
print(tuplex)

# converting the tuple to list
listx = list(tuplex)

# use different ways to add items in list
listx.insert(3, 30)

tuplex = tuple(listx)
print(tuplex)

# Way 2:

a = ('Rose', 'Lily', 'Lotus', 'Jasmine', 'Daisy')
print(a)
b = list(a)
```



```
b.insert(3, 'Orchid')  
a = tuple(b)  
print(a)
```

18) A python program to modify or replace an existing element of a tuple with a new element.

Solution:

```
a = ('Rose', 'Lily', 'Lotus', 'Orchid', 'Jasmine')  
  
print(a)  
b = list(a)  
b[2] = 'Daisy'  
a = tuple(b)  
print(a)
```

19) A python program to delete an element from a particular position in a tuple.

Solution:

```
a = ('Rose', 'Lily', 'Lotus', 'Orchid', 'Jasmine', 'Daisy')  
  
print(a)  
b = list(a)  
b.remove('Lily')  
a = tuple(b)  
print(a)
```