# Tuple: It is a collection of different data elements separated by comma enclosed

within the round brackets. It is immutable.

mutable-- immutable

list ==[]--- tuple ()

mutable--- immutable

-------------------------------------------

tuple2 = (1,2,33,4,5)

tuple3 =('one','two','three')

tuple4 = (1,2,3,4,5,'one','two','three')

print(tuple2)

print(type(tuple1))

print(tuple3)

print(tuple4)

-------------------------------------

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

print(tup)

tup1 = ("One","Two","Three",'Four')

print(tup1)

#Accessing elements from tuple

print(tup1[0])

print(tup1[2])

tup2 = (23,44,66,1,3,'One','Two','Python',44,1,2,34,65) #mixed items

print(tup2)

tup3=() #empty tuple

print(tup3)

print(type(tup3))

--------------------------------------------

# Creating a tuple without parentheses.

tuple1 = 2,3,4,5

print(tuple1) # packing

a,b,c,d=tuple1 # Unpacking

print(a,b,c,d)

#creating a list without parentheses.

list1 = [1,2,3,4]

print(type(list1))

a,b,c,d = list1

print(a,b,c,d)

-------------------------------------------------

# Tuple with a single element

tuple1 = ('Python')

print(tuple1)

print(type(tuple1))

tuple1 = (123)

print(tuple1)

print(type(tuple1))

-----------------------------------------

# Iterate a tuple using For loop

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

print(tup)

tup1 = ("One","Two","Three",'Four')

print(tup1)

for element in tup1:

print(element)

# Check if item exists in tuple

if "One" in tup1:

print("Present")

else:

print("Not")

-------------------------------------------

# del element using pop,remove

#since the tuple immutable the delete operations are not available

# del tup1 -- available (completely removing the object)

=============================================

#slice a tup from 3rd index to last index

#slice a tup from second last index to first

#slice a tup in reverse order

#slice a tup printing even elements

#slice a tup -print only last element

#slice a tup -print only 2nd element

#slice a tup print only second last element== tup[-2]

-------------------------------------------

#Using the sorted() function

tup4 = (29,32,43,22,1,12,30)

print("The tuple is:",tup4)

a = sorted(tup4) # returns a list of sorted elements

print("sorted tuple:\n",a)

print(type(a))

sort1 = tuple(sorted(tup4))

print("sorted tuple:\n",sort1)

print(type(sort1))

=========================================

# concatenating the tuples

tup = (200,300,400,500,700)

print("Tuple 1: ",tup)

tup1 = ("Nashik","Pune","Mumbai","Khalapur","Goa","Pune")

print("Tuple 2: ",tup1)

print("Concatenation of tuples is:")

tup2 = tup+tup1

print(tup2)

# Getting the index of element using index()

tup1 = ("Nashik","Pune","Mumbai","Khalapur","Goa","Pune")

print("Tuple 2: ",tup1)

idx = tup1.index("Pune")

print(idx)

# Counting the occurances of elements in tuple

n = tup1.count("Pune")

print("Elements occurs %d times."%n)

# Checking the length of tuple

tup1 = ("Nashik","Pune","Mumbai","Khalapur","Goa","Pune")

print("Tuple 2: ",tup1)

length = len(tup1)

print("The length of tuple is: ",length)

========================================================

1.wap to create a nested tuple

2.provided a tuple with nested tuple, try to access the

elements from the inner tuple

3.given a tuple with nested list with 5 elements.

try to access the elements of list and modify elements of

list. add elements to the list and delete 1 element

from the inner list.

4. Given a tuple with nested tuples like -((1,2),(3,4),(5,6)).

iterate through the tuple to print 2nd element

from every nested tuple

5.given a tuple, return the position of element 'x'.

6.given a tuple, check how many times 'abc' occurs.