

Programming and Problem Solving through Python Language O Level / A Level

Chapter - 5: Sequence Data Types

Python Collections (Arrays)

There are four collection data types in the Python programming language:

- **List** is a collection which is ordered and changeable. Allows duplicate members.
- **Tuple** is a collection which is ordered and unchangeable. Allows duplicate members.
- **Set** is a collection which is unordered and unindexed. No duplicate members.
- **Dictionary** is a collection which is unordered, changeable and indexed. No duplicate members.

Tuple

- A tuple is a collection which is ordered and unchangeable (immutable).
- The tuple is a datatype available in Python which can be written as a comma-separated values (items). Optionally, you can put these comma-separated values between parentheses also.
- Items in a tuple need not be of the same type.

Creating Tuple

```
tup1 = ( 'physics', 'chemistry', 1997, 2000)
```

```
tup2 = (1, 2, 3, 4, 5 )
```

```
tup3 = "a", "b", "c", "d"
```

To create a tuple with only one item, you have to add a comma after the item, otherwise Python will not recognize it as a tuple.

```
thistuple = "apple", "mango"
```

```
print(type(thistuple))
```

```
thistuple = ()
```

```
print(type(thistuple))
```

a tuple with one item

```
thistuple = ("apple",)
```

```
print(type(thistuple))
```

#NOT a tuple

```
thistuple = ("apple")
```

```
print(type(thistuple))
```

Output

```
<class 'tuple'>
```

```
<class 'tuple'>
```

```
<class 'tuple'>
```

```
<class 'str'>
```

Access Items

To access values in **tuple**, use the square brackets for slicing along with the index or indices to obtain value available at that index.

```
tup1 = ['physics', 'chemistry', 1997, 2000]
tup2 = [1, 2, 3, 4, 5, 6, 7 ]
print ("tup1[0]: ", tup1[0])
print ("tup2[1:5]: ", tup2[1:5])
print ("tup1[3]: ", tup1[-1])
```

Output–

```
tup1[0]: physics
tup2[1:5]: [2, 3, 4, 5]
tup1[3]: 2000
```

Negative Indexing

Negative indexing means beginning from the end, -1 refers to the last item, -2 refers to the second last item etc. **tup1[-1]**

Range of Indexes (Slicing)

- You can specify a range of indexes by specifying where to start and where to end the range.
- **Tup1[2:5]** - The search will start at index 2 (included) and end at index 5 (not included).
- Remember that the first item has index 0.
- **Tup1[:5]** - By leaving out the start value, the range will start at the first item:

Range of Negative Indexes

Specify negative indexes if you want to start the search from the end of the list. **Tup1[-4:-1]**

Updating Tuple

Tuples are immutable which means we cannot update or change the values.

```
tup1 = (12, 34.56)
tup2 = ('abc', 'xyz')

# Following action is not valid for tuples
# tup1[0] = 100

# So let's create a new tuple as follows
tup3 = tup1 + tup2
print (tup3)
```

Output

```
(12, 34.56, 'abc', 'xyz')
```

Adding new item in Tuple

```
t = (10, 20, 30, 40)
t = t + (60, )      # this will do modification of t.
print (t)
```

Output

(10, 20, 30, 40, 60)

Method to update the values in Tuple

You can convert the tuple into a list, change the list, and convert the list back into a tuple.

```
x = ("apple", "banana", "cherry")
y = list(x)
y[1] = "kiwi"
x = tuple(y)
print(x)
```

Output

("apple", "kiwi", "cherry")

Loop Through a Tuple

You can loop through the Tuple items by using a `for` loop:

```
list = ( "apple", "banana", "cherry" )
for x in list:
    print(x)
```

Check if Item Exists

To determine if a specified item is present in a tuple use the `"in"` keyword:

```
tuple = ( "apple", "banana", "cherry" )
if "apple" in tuple:
    print("Yes, 'apple' is in the fruits tuple")
```

Tuple Length

To determine how many items a tuple has, use the `len()` function. e.g. `print(len(tuple))`

Removing Item from the Tuple

Tuples are **unchangeable**, so we cannot remove items from it, but we can delete the tuple completely

- The `del` keyword removes the tuple completely.
`del tuple`

Basic Tuple Operations

Python Expression	Results	Description
len((1, 2, 3))	3	Length
(1, 2, 3) + (4, 5, 6)	(1, 2, 3, 4, 5, 6)	Concatenation
('Hi!') * 4	['Hi!', 'Hi!', 'Hi!', 'Hi!']	Repetition
3 in (1, 2, 3)	True	Membership
for x in (1,2,3) : print (x,end = ' ')	1 2 3	Iteration

Built-in Function

Sr.No.	Function & Description
1	len(tuple) : Gives the total length of the tuple.
2	max(list) : Returns item from the tuple with max value.
3	min(list) : Returns item from the tuple with min value.
4	tuple(seq): Converts a list into tuple.
5	cmp(tuple1, tuple2) : Compares elements of both tuples.

Tuple Built-in Methods

Sr.No.	Methods & Description
1	tuple.count(obj): Returns count of how many times obj occurs in list
2	tuple.index(obj) : Returns the lowest index in tuple that obj appears

Example

Write a program to input 'n' numbers and store it in tuple.

```
t=tuple()
n=input("Enter any number")

print " enter all numbers one after other"
for i in range(n):
    a=input("enter number")
    t=t+(a,)

print "output is"
for i in range(n):
    print t[i]
```

Assignment

1. Define Tuple
2. What is the output of the following code:
a) `print type ((1,2))` b) `a= (1, 2, 3, None, ( ), [ ])`
3. Write the output from the following code:

```
A=(2,4,6,8,10)
L=len(L)
S=0
for I in range(1,L,2):
S+=A[I]
print "Sum=",S
```
4. Write a program to input 'n' numbers and store it in a tuple and find maximum & minimum values in the tuple.
5. Find the output from the following code:

```
t=tuple()
t = t +(PYTHON,)
print t
print len(t)
t1=(10,20,30)
print len(t1)
```
6. Write a program to input two set values and store it in tuples and also do the comparison.
7. Write a program to input 'n' employees' salary and find minimum & maximum salary among 'n' employees.
8. Write a program to input 'n' customers' name and store it in tuple and display all customers' names on the output screen.
9. Write a program to input 'n' numbers and separate the tuple in the following manner.

Example

```
T=(10,20,30,40,50,60)
T1 =(10,30,50)
T2=(20,40,60)
```

10. Find the errors from the following code:

```
t=tuple {}
n=input(Total number of values in tuple)
for i in range(n)
a=input("enter elements")
t=t+(a)
print "maximum value=",max(t)
print "minimum value=",min(t)
```