Python Tuples

What is Tuple?

A Tuple is **a collection of Python objects separated by commas**. In some ways, a tuple is similar to a list in terms of indexing, nested objects and repetition but a tuple is immutable (unchangeable) unlike lists which are mutable (changeable).

How to create a Tuple?

my_tuple = (“Alice”, 23, True)

print(my_tuple)

Output : >> (‘Alice’, 23, True)

Tuple Items

Tuple items are ordered, unchangeable, and allow duplicate values.

Tuple items are indexed like lists and strings.

- When we say that tuples are ordered, it means that the items have a defined order, and that order will not change.
- Tuples are unchangeable, meaning that we cannot change, add or remove items after the tuple has been created.
- Since tuples are indexed, they can have items with the same value.
- To determine how many items a tuple has, we can use len() function.

Tuple Items - Data Types

- Tuple items can be any of data types : Strings, int, float, Boolean

tuple_1 = (1, 2, 3, 4, 5)

tuple_ 2 = (True, False, False)

tuple_3 = (“Hello”, 23, False)

The tuple() Constructor

- It is also possible to use tuple() to create a tuple.

```
my_tuple = tuple(("Alice", 23, True))
print(my_tuple)
```

Note : We must use parenthesis inside the tuple() method as it takes only one argument.

Python - Access Tuple Items

We can access a tuple in following ways :

- With index number
- With negative indexing
- With slicing

We can check if an item exists in the tuple with [in, not in] keywords.

Python - Update Tuples

- Once a tuple is created, you cannot change its values. Tuples are **unchangeable**, or **immutable** as it also is called.
- Tuples are unchangeable, meaning that you cannot change, add, or remove items once the tuple is created. But there are some workarounds : You can convert the tuple into a list, change the list, and convert the list back into a tuple.
- You are allowed to add tuples to tuples, so if you want to add one item, (or many), create a new tuple with the item(s), and add it to the existing tuple:

```
my_tuple = (“Bob”, 25, False)

my_tuple_2 = (“Alice”)

my_tuple = my_tuple + my_tuple_2

print(my_tuple) # Output : >>(“Bob”, 25, False, “Alice”)
```

Python - update Tuples

- **Note:** You cannot remove items in a tuple.
- Tuples are **unchangeable**, so you cannot remove items from it, but you can use the same workaround as we used for changing and adding tuple items:
  - Convert the tuple into a list, remove "apple", and convert it back into a tuple.
  - Or Just use del keyword to delete the tuple completely.

```
tp = (“hello”, 1, 2, 3)
del tp
print(tp) # This will raise an error as the tuple is already deleted.
```

Python - Unpack Tuples

- Unpacking a Tuple

  When we create a tuple, we normally assign values to it. This is called "packing" a tuple:

  my_tuple = (1, 2, 3)

  But, in Python, we are also allowed to extract the values back into variables. This is called "unpacking":

  my_tuple = (“apple”, “mango”, “banana”)

  (red, green, blue) = my_tuple

  print(f’1 = {red}, 2 = {green}, 3 = {blue}’)

  Output : >> 1 = apple, 2 = mango, 3 = banana

Joining a tuple

- Joining two tuples:
- tp1 = (1, 2, 3, 4, 5)
- tp2 = (6, 7, 8)
- tp3 = tp1 + tp2
- print(tp3)

Multiplying a tuple

- If you want to multiply the content of a tuple a given number of times, you can use the * operator

- Try it yourself and observe the output :)

Looping through a tuple

- We can loop through the tuple items with loops like
- lists.
- With for loop
- With while loop

Tuple methods

Method	Description
count()	Returns the number of times a specified value occurs in a tuple
index()	Searches the tuple for a specified value and returns the position of where it was found

References

- https://www.w3schools.com/python/python_tuples_methods.asp
- https://www.tutorialspoint.com/python/python_tuples.htm

Thank you!