



Cloud, APIs and Alerts > Introduction Python

Tuples in Python

Introduction to Tuples

Tuples are quite similar to lists in Python except for a few dissimilarities. Tuples like lists can be used when you want to group data together based on similarities. However, unlike lists, tuples are immutable data types. What do I mean by Immutable? In programming languages, immutable means something that cannot be modified once you define it, whereas mutable means something that can be modified. Lists are mutable since we can add, update, and remove data from lists. Tuples, on the other hand, are immutable, which means you cannot add, update, or delete specific data in a tuple.

Below are a few examples of tuples:

```
countries = ('India', 'USA', 'Australia', 'South Africa', 'Germany')
flowers = ('rose', 'tulip', 'orchid', 'sunflower')
colors = 'green', 'red', 'blue', 'black', 'yellow'
exam_scores = (78, 90, 65, 83, 71)
```

As you can see, we have created 4 tuples. The tuples are created based on similarity such as the first one is a tuple of countries, the second one is types of flowers, the third one is colors, and the fourth one is marks obtained in an exam.

The syntax for tuples is quite similar to that of lists except that the types of brackets used are circular brackets. The syntax is that all the values should be separated by a comma. Also, the tuples should be surrounded by circular brackets. As you can see in the examples for the colors tuple, even if you create a sequence of data values separated by a comma and don't add circular brackets around them, it is considered as a tuple in Python. However, it is a good practice to add circular brackets so that the reader of the program is not confused with the data type and hence we shall use circular brackets in all our examples.

What are the limitations of tuples?

1. You can't add data to a tuple once you have created it.
2. You can't remove data from a tuple once you have created it.

Accessing data in a tuple



Indexes in tuples are same as the indexes we learnt in lists. Hence if we want to access the value of country at index 1 in the tuple then the syntax will be as shown below

```
countries = ('India', 'USA', 'Australia', 'South Africa', 'Germany')
print countries[1]
```

More built-in functions

Python provides the same built-in functions that we learnt for lists that we can use to do common tasks on tuples. Some of them are

1. `len(tuple_name)`
2. `max(tuple_name)`
3. `min(tuple_name)`

Let us use these above functions in a program to understand how they work

```
exam_scores = [78, 90, 65, 83, 71]

print("The total number of exam scores is ", len(exam_scores))
print("The lowest exam score is ", min(exam_scores))
print("The maximum exam score is ", max(exam_scores))
```

Write the above program and see what output you get. In this program, we have created a tuple of exam scores that contains the marks obtained in 5 subjects. The `len` function will print the total count of items present in the `exam_scores` tuple. The `min` function prints the item with the least value in the tuple. In this case it will print the marks of the subject with the lowest score i.e. 65. The `max` function will print the marks of the subject with the highest score i.e. 90.

How are tuples useful if they are immutable?

You will be wondering how would any data type be useful if you cannot modify it as required. Why do we need to use tuples if we can do the same things using lists. The below points describe some of the benefits of using tuples as they are immutable:

1. There are cases in programming, where the programmer does not want a data to be modified while the program is running, as this can cause the program to crash or result in incorrect output. In such cases, immutable data types such as Tuples are really useful. The tuples protect your data from getting modified.



2. If you are going to define the data only once in your program and only going to use a loop to access the data without modifying them then you should use a tuple instead of a list. As soon as you define the tuple, the processor knows your tuple data size and also knows that the data size will never change since it is immutable. However, in case of lists the data size will vary based on the data that you add or remove from the list. Hence the processor can do various memory optimizations in case of tuples as the data size is known and constant. This makes tuples faster than lists.