Tuples in Python

Introduction

- A **tuple** is an **ordered**, **immutable** collection of elements.
- Tuples are defined using **parentheses ()**, but even without them, comma-separated values are considered a tuple.
- Tuples can contain elements of **different data types**.

```
t1 = (1, 2, 3)
t2 = 'a', 'b', 'c'  # also a tuple
empty = ()          # empty tuple
single = (5,)       # single element tuple must end with a comma
```

Indexing

- Tuples are **indexed** like lists.
- Index starts from 0 (positive indexing) or from -1 (negative indexing).

```
t = (10, 20, 30, 40)
print(t[0])   # 10
print(t[-1])  # 40
```

Tuple Operations

1. Concatenation

```
t1 = (1, 2)
t2 = (3, 4)
t3 = t1 + t2
print(t3)  # (1, 2, 3, 4)
```

2. Repetition

```
t = (1, 2)
print(t * 2)  # (1, 2, 1, 2)
```

3. Membership

```
t = (10, 20, 30)
print(20 in t)     # True
print(50 not in t) # True
```

4. Slicing

```
t = (10, 20, 30, 40, 50)
print(t[1:4])  # (20, 30, 40)
```

Built-in Functions / Methods

Function / Method	Description	Example
len()	Returns number of elements	len((1, 2, 3)) → 3
tuple()	Converts other sequence to tuple	tuple([1, 2]) → (1, 2)
count(x)	Counts number of times x appears	(1, 2, 2).count(2) → 2
index(x)	Returns index of first occurrence of x	(1, 2, 3).index(2) → 1
sorted()	Returns sorted list from tuple	sorted((3, 1, 2)) → [1, 2, 3]
min()	Smallest value in tuple	min((3, 1, 2)) → 1
max()	Largest value in tuple	max((3, 1, 2)) → 3
sum()	Sum of numeric values	sum((1, 2, 3)) → 6

Tuple Assignment

You can unpack a tuple directly into variables:

```
t = (10, 20, 30)
a, b, c = t
print(a, b, c)  # 10 20 30
```

Nested Tuple

A tuple inside another tuple.

```
nested = ((1, 2), (3, 4))
print(nested[0][1])  # 2
```

Finding Minimum, Maximum, and Mean

```
data = (15, 25, 35, 45)
print("Min:", min(data))
print("Max:", max(data))
print("Mean:", sum(data) / len(data))
```

Linear Search in Tuple

```
nums = (5, 10, 15, 20)
key = 15
found = False

for n in nums:
    if n == key:
        found = True
        break

if found:
    print("Element found")
else:
    print("Not found")
```

Counting Frequency of an Element

```
t = (1, 2, 3, 2, 4, 2)
x = 2
print("Frequency of", x, ":", t.count(x))  # 3
```