**Tuple**

A **tuple** in Python is an **ordered, immutable** collection of items. Think of it as a **list that cannot be modified** after it is created. Tuples are useful when you want to store a collection of items that should not be changed, such as coordinates, fixed configurations, or constant values.

**What is a Tuple?**

* **Ordered**: The elements have a specific order and can be accessed by their index.
* **Immutable**: Once a tuple is created, you cannot modify, add, or remove elements.
* **Heterogeneous**: A tuple can contain elements of different data types.
* **Hashable**: Tuples can be used as dictionary keys if they contain only immutable elements.
* **Faster than lists**: Since tuples are immutable, they are optimized for performance.

**Tuple Methods**

| **Method** | **Description** | **Example** |
| --- | --- | --- |
| count(x) | Returns the number of occurrences of x | (1, 2, 3, 1).count(1) → 2 |
| index(x) | Returns the index of the first occurrence of x | (1, 2, 3).index(2) → 1 |

### ****When to Use Tuples****

* **Fixed Data**: Tuples are great for representing constant data that shouldn’t change.
  + Example: Coordinates (latitude, longitude)
* **Dictionary Keys**: Tuples can be used as keys in dictionaries because they are hashable.
* **Performance**: Tuples are faster than lists for iteration and are more memory-efficient.

**Tuple vs List: Key Differences**

| **Feature** | **Tuple** | **List** |
| --- | --- | --- |
| **Mutable** | No | Yes |
| **Syntax** | (1, 2, 3) | [1, 2, 3] |
| **Performance** | Faster | Slower |
| **Use Case** | Fixed data (coordinates) | Dynamic data (to-do list) |
| ****Summary****  * A **tuple** is an immutable, ordered collection of elements. * Tuples are used when you want to store data that should not be changed. * They are faster and more memory-efficient than lists. * Tuples can be nested, unpacked, and used as dictionary keys. |  |  |