**1. List**

A list is a mutable, ordered collection of elements. It allows duplicate values and supports indexing and slicing.

**Advantages:**

* Mutability: Lists can be modified after creation (e.g., add, remove, or update elements).
* Order: Maintains the order of elements.
* Diversity: Can store mixed data types (e.g., integers, strings, and objects).

**Disadvantages:**

* Performance: Slower than tuples for iteration or fixed-size collections since it’s mutable.
* Memory: Takes more memory due to additional features.

**2. Tuple**

A tuple is an immutable, ordered collection of elements. Like lists, it allows duplicate values and supports indexing.

**Advantages:**

* Immutability: Safer for fixed collections where elements should not be modified.
* Performance: Faster and consumes less memory compared to lists.
* Hashable: Can be used as keys in dictionaries or elements in sets if all its elements are hashable.

**Disadvantages:**

* Immutability: Cannot be changed after creation, which may require creating a new tuple for modifications.
* Limited Functionality: Does not have built-in methods for adding, removing, or modifying elements.

**3. Set**

A set is an unordered, mutable collection of unique elements. It does not allow duplicate values and does not support indexing or slicing.

**Advantages:**

* Unique Elements: Automatically removes duplicates.
* Efficiency: Fast membership testing (in operator) and mathematical operations (union, intersection, etc.).
* Flexibility: Mutable; elements can be added or removed.

**Disadvantages:**

* Unordered: No way to access elements by index or position.
* Mutable Only: Cannot contain mutable elements like lists or dictionaries.

**4. Dictionary**

A dictionary is an unordered collection of key-value pairs. Keys must be unique and immutable, while values can be of any type.

**Advantages:**

* Key-Value Pairing: Allows efficient data retrieval using keys.
* Mutability: Can add, remove, or update key-value pairs.
* Flexible Values: Values can be any data type, including lists and other dictionaries.

**Disadvantages:**

* Unordered (before Python 3.7): Keys and values had no defined order (now maintains insertion order).
* Memory: Takes more memory than lists due to hashing.
* Key Restrictions: Keys must be hashable and unique.