**Lists:**

* Use lists when you have a collection of elements that may need to be modified (added, removed, or updated) during the program's execution.
* Lists are ordered, mutable (modifiable), and allow duplicate elements.
* Lists are suitable for sequences of data where the order matters and you need to perform operations like appending, indexing, and slicing elements.

**Tuples:**

* Use tuples when you have a collection of elements that should remain unchanged (immutable) throughout the program.
* Tuples are ordered, immutable, and allow duplicate elements.
* Tuples are suitable when you want to use collections as keys in dictionaries (since they are immutable) or when you want to ensure data integrity by preventing modifications.

**Sets:**

* Use sets when you want to store a collection of unique elements with no specific order.
* Sets are unordered and do not allow duplicate elements. When you attempt to add a duplicate element, it will be ignored.
* Sets are suitable for tasks like finding unique elements in a collection, performing set operations like union, intersection, and difference, and removing duplicates from a list.

**Dictionaries:**

* Use dictionaries when you need to store data in a key-value pair format, where each key maps to a corresponding value.
* Dictionaries are unordered, mutable, and do not allow duplicate keys (keys must be unique).
* Dictionaries are ideal when you want to access elements by a unique identifier (key) and perform quick lookups.