

### What are Java Collections?

Collections represent a single **unit of objects** or **groups**. They are used to **store, retrieve, sort, and manipulate data** within your application. Collections are saved in **volatile memory** which means data does not persist after the program stopped running.

### Collections vs. Arrays

We use collections instead of arrays because collections are dynamic. An array size cannot be changed after it has been created while collections **can expand or reduce its size** based on what you add or delete.

### Common Implementation of Classes

Type	Random-Access to Elements	Duplicate Elements	Key-Value	Ordering /Sorting	Description
ArrayList	Yes	Yes	No	Yes- insertion order, can be sorted	Like a regular array but with dynamic length; use this if you need a dynamic collection with fast, random access to elements
LinkedList	No	Yes	No	Yes- insertion order, unsorted	Objects can be added multiple times; use this if you need fast adding and removing of items
TreeSet	No	No	No	Yes- ascending order according to comparator, pre-sorted	Object can only be added once; Slower, items must not change state when used.
HashMap	Yes	No	Yes	No- undefined order, unsorted	Set of key/value pairs; use key to get value from map; Keys of a map must not change state
TreeMap	Yes	No	Yes	Yes- ascending order of keys according to comparator	Sorted tree of values; Slower, Keys of a map must not change state