|  |  |  |
| --- | --- | --- |
| **Collections** | | |
| List | Map | Set |
| ArrayList | HashMap | HashSet |
| LinkedList | TreeMap | TreeSet |
| Vector |  | LinkedHashSet |
| Stack |  |  |
| Maintains insertion order  Allows duplicate elements  Provides indexed access (elements can be accessed by position) | Key-value pairs (each key maps to a single value)  Keys must be unique, but values can be duplicated  Does not maintain insertion order (except LinkedHashMap) | No duplicate elements allowed  Does not maintain insertion order (except LinkedHashSet)  Fast lookups using HashSet |

**Key Differences Summary Table**

| **Feature** | **List** | **Set** | **Map** |
| --- | --- | --- | --- |
| **Duplicates Allowed?** | ✅ Yes | ❌ No | ❌ No (Keys), ✅ Yes (Values) |
| **Maintains Order?** | ✅ Yes | ❌ No (HashSet) / ✅ Yes (LinkedHashSet) | ❌ No (HashMap) / ✅ Yes (LinkedHashMap) |
| **Indexed Access?** | ✅ Yes | ❌ No | ❌ No |
| **Key-Value Structure?** | ❌ No | ❌ No | ✅ Yes |

**When to Use?**

* Use **List** when **ordering & duplicates** are needed (e.g., storing ordered items).
* Use **Set** when **uniqueness** is required (e.g., storing unique values).
* Use **Map** when **key-value pairs** are needed (e.g., storing user IDs with names).