|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Feature** | **ArrayList** | **LinkedList** | **HashSet** | **HashMap** |
| **Type** | List | List | Set | Map |
| **Stores** | Elements (ordered) | Elements (ordered) | Unique elements | Key–Value pairs |
| **Order** | Maintains insertion order | Maintains insertion order | No order guaranteed | No order of keys guaranteed |
| **Duplicates** | Allowed | Allowed | Not allowed | Keys: Not allowedValues: Allowed |
| **Nulls** | Many nulls allowed | Many nulls allowed | Only one null allowed | One null key + many null values allowed |
| **Access (get element)** | Fast (O(1)) | Slow (O(n)) | Not index-based | Fast by key (O(1)) |
| **Insert/Delete** | Slow in middle (shifting needed) | Fast in middle (O(1) if node known) | Fast (O(1) average) | Fast (O(1) average) |
| **Best Use Case** | When you need fast random access | When frequent insert/delete | When you need unique elements | When you need key → value mapping |

|  |  |  |
| --- | --- | --- |
| **Collection** | **Real-world Analogy** | **Why?** |
| **ArrayList** | 📚 **Books on a shelf** | You can quickly pick a book by its position (fast random access), but adding/removing books in the middle requires shifting others (slow). |
| **LinkedList** | ⛓️ **Train with connected coaches** | Each coach (node) is linked to the next and previous. Easy to attach or remove a coach, but to reach a coach in the middle, you must walk coach by coach (slow access). |
| **HashSet** | 🎟️ **Unique entry tickets in a box** | Each ticket is unique. No duplicates allowed. No guarantee of order when you pick tickets. |
| **HashMap** | 🗂️ **Dictionary (word → meaning)** | Each word (key) maps to one meaning (value). Easy to find meaning if you know the word (key). No guaranteed order of words. |

**Quick Memory Trick**:

* **ArrayList =** Best for reading/accessing data. 🡪**Shelf** (fast lookup by position).
* **LinkedList =** Best for adding/removing a lot. 🡪**Train** (easy add/remove coaches).
* **HashSet =** Best for unique items, no duplicates. 🡪**Ticket Box** (unique items, no order).
* **HashMap =** Best for key–value lookups. 🡪**Dictionary** (key → value).