Need of Collections:

An Array is an indexed collection of fixed number of homogenous data elements

Advantage: can represents multiple values in single variable

Limitations:

1. Fixed in Size
2. Only Homogenous elements
   1. Can resolve this by Object Arrays
3. There is no underlying Data structure

Overcome of above limitations go for Collections:

1. Grow able in nature
2. Both homogenous and Heterogeneous
3. Is implemented based on some slandered

|  |  |  |
| --- | --- | --- |
| **S.No** | **Arrays** | **Collections** |
| 1 | Fixed in Size | Grow able in nature |
| 2 | Memory-Not Recommended | Recommended |
| 3 | Performance-Recommended | Not Recommended |
| 4 | Hold Homogeneous elements | Hold Heterogeneous |
| 5 | Readymade method not available | ready method is available |
| 6 | hold Primitives and objects | hold objects |

Collection:

If we want to represent a group of individual objects as single entity then we should go for Collection.

Collection Framework:

Defines several classes/interfaces which can be used to represent a single entity

1. **Collection :( Interface) 1.2**

* If we want to represent a group of individual objects as single entity then we should go for Collection
* Defines most common methods
* In general it is the root interface for all collection framework

Collections: (class)

* Utility class which defined several utility methods to collection

1. **List:(1.2)**
   1. Duplicates are allowed
   2. Insertion order is preserved

List is below types:

1. Array list 1.2
2. Linked list 1.2
3. Vector 1.0
   * Stack 1.0
4. **Set: 1.2**

* Duplicates are not allowed
* Insertion order is not preserved

1. Hash Set 1.2
2. Linked hash set 1.4

|  |  |
| --- | --- |
| **List** | **Set** |
| Duplicates allowed | not allowed |
| insertion order is preserved | not preserved |

1. **Sorted Set (child of set):1.2**

* Duplicates are not allowed
* Some sorting order

1. **Navigable set(Child of Set)1.6**

* Tree set 1.2

1. **Queue(Child of Collection):**

**Key Value pairs**

1. **Map: 1.2**
   1. Key-Value
   2. Duplicates key not allowed
   3. Values duplicates allowed

Types:

1. Hash Map 1.2
   1. Linked Hash Map 1.4
2. Weak Hash Map 1.2
3. Identity Hash Map 1.4
4. Dictionary 1.0
5. Hash Table 1.0
   1. Properties 1.0
6. **Sorted Map(Child of Map):**
   1. Some sorting order of key
7. **Navigable Map(child of Sorted map):**
8. Tree Map

1. Array List:

* Resizable array
* Duplicates are allowed
* Insertion order
* Heterogeneous objects are allowed\*
  + Tree Set
  + Tree Map
* Null insertion is possible

New Capacity= (Current Capacity\*3/2)+1

Linked List:

1. Double liked list
2. Insertion order
3. Duplicates are allowed
4. Heterogeneous are allowed
5. Null can be added
6. Liked list is serializable, Clonable and