|  |  |
| --- | --- |
| Array | Vector |
| Uses array structure internally | Uses array structure internally |
| Increases size by 50% | Doubles the vector size |
| Non-Synchronized | Synchronized |
| Fast | Slower |
| Single Threaded | Multi-Threaded |

|  |  |
| --- | --- |
| Hash Set | Sorted Set |
| Data structure used is Hash Tables | Data structure used is Red-back tree (Balanced Binary trees) |
| O(1) | O(lgn) |
| if you don't need the elements to be sorted | If you need the elements to be sorted |

|  |  |
| --- | --- |
| Hash Set | Tree Set |
| Hash tables | Red-back structure |
| faster | slower |
| O(1) | O(lgn) |
| Allows null objects | Does not allows null objects because of comparison operation |
| No duplication because uses equals() for checking duplicate values | Duplication in set because of compare to function. |
| Does not maintain order | Maintains order |

|  |  |
| --- | --- |
| Array | List |
| fixed in size, or unlikely to grow much | Variable length data |
| Indexing | Iterated in it’s entirety |
| Preferably used to store Multi-Dimensional data | For memory issues |

|  |  |
| --- | --- |
| List | Set |
| Duplicates allowed | Duplicates not allowed |
| Ordered | Depends on implementation |
| Positional Access | No positional access |
| Ordered grouping of items | Un-ordered grouping of items without duplications |

|  |  |
| --- | --- |
| Navigable Set | Navigable Map |
| No Duplicates | No Duplicates |
| Don’t have key and value structure | Have key and value structure |
|  |  |