Collections

* List - Insertion order preserved, Duplicates allowed

1. ArrayList - Non Synchronized
2. LinkedList - Non Synchronized
3. Vector - Synchronized

* Set – Duplicates not allowed

1. HashSet – Insertion order not preserved
2. LinkedHashSet – Insertion order preserved
3. SortedSet - Interface

TreeSet – Stores with a particular sorting order

* Map

1. HashMap – Insertion order of keys is not preserved
2. LinkedHashMap – Insertion order of keys is preserved
3. TreeMap – Sorting is possible

Constructors

* ArrayList – InitialCapacity=10

1. new ArrayList()
2. new ArrayList(Collection c)
3. new ArrayList(int initialCapacity)

* LinkedList –

1. new LinkedList()
2. new LinkedList(Collection c)

* Vector –

1. new Vector()
2. new Vector(Collection c)
3. new Vector(int initialCapacity)
4. new Vector(int initialCapacity, int capacityIncrement)

* HashSet – initialCapacity=16, loadFactor=0.75

1. new HashSet()
2. New HashSet(collection c)
3. New HashSet(int initialCapacity)
4. New HashSet(int initialCapacity, float loadFactor)

* LinkedHashSet –

1. New LinkedHashSet()
2. New LinkedHashSet(collection c)
3. New LinkedHashSet(int initialCapacity)
4. New LinkedHashSet(int initialCapacity, float loadFactor)

* TreeSet –

1. New TreeSet()
2. New TreeSet(Collection c)
3. New TreeSet(Comparator comparator)

* HashMap – InitialCapacity=16, LoadFactor=0.75

1. New HashMap()
2. New HashMap(int initialCapacity)
3. New HashMap(int initialCapacity, float loadFactor)

* LinkedHashMap – InitialCapacity=16, LoadFactor=0.75

1. New LinkedHashMap()
2. New LinkedHashMap(int initialCapacity)
3. New LinkedHashMap(int initialCapacity, float loadFactor)

* TreeMap –

1. New TreeMap()
2. New TreeMap(Map m)
3. New TreeMap(Comparator comparator)