- *Constructor Summary*

Constructors

Constructor and Description
ArrayList() Constructs an empty list with an initial capacity of ten.
ArrayList(Collection<? extends E> c) Constructs a list containing the elements of the specified collection, in the order they are returned by the
ArrayList(int initialCapacity) Constructs an empty list with the specified initial capacity.

- *Method Summary*

All Methods Instance MethodsConcrete Methods

Modifier and Type	Method and Description
boolean	add(E e) Appends the specified element to the end of this list.
void	add(int index, E element) Inserts the specified element at the specified position in this list.
boolean	addAll(Collection<? extends E> c) Appends all of the elements in the specified collection to the end of t returned by the specified collection's Iterator.
boolean	addAll(int index, Collection<? extends E> c) Inserts all of the elements in the specified collection into this list, sta
void	clear() Removes all of the elements from this list.
Object	clone() Returns a shallow copy of this ArrayList instance.

`boolean`	`contains(Object o)` Returns `true` if this list contains the specified element.
`void`	`ensureCapacity(int minCapacity)` Increases the capacity of this `ArrayList` instance, if necessary, to e number of elements specified by the minimum capacity argument.
`void`	`forEach(Consumer<? super E> action)` Performs the given action for each element of the `Iterable` until al action throws an exception.
`E`	`get(int index)` Returns the element at the specified position in this list.
`int`	`indexOf(Object o)` Returns the index of the first occurrence of the specified element in t the element.
`boolean`	`isEmpty()` Returns `true` if this list contains no elements.
`Iterator<E>`	`iterator()` Returns an iterator over the elements in this list in proper sequence.
`int`	`lastIndexOf(Object o)` Returns the index of the last occurrence of the specified element in th the element.
`ListIterator<E>`	`listIterator()` Returns a list iterator over the elements in this list (in proper sequen
`ListIterator<E>`	`listIterator(int index)` Returns a list iterator over the elements in this list (in proper sequen the list.
`E`	`remove(int index)` Removes the element at the specified position in this list.
`boolean`	`remove(Object o)` Removes the first occurrence of the specified element from this list, i
`boolean`	`removeAll(Collection<?> c)`

	Removes from this list all of its elements that are contained in the sp
boolean	**removeIf**(**Predicate**<? super **E**> filter) Removes all of the elements of this collection that satisfy the given pr
protected void	**removeRange**(int fromIndex, int toIndex) Removes from this list all of the elements whose index is between fr exclusive.
void	**replaceAll**(**UnaryOperator**<**E**> operator) Replaces each element of this list with the result of applying the oper
boolean	**retainAll**(**Collection**<?> c) Retains only the elements in this list that are contained in the specifi
E	**set**(int index, **E** element) Replaces the element at the specified position in this list with the spe
int	**size**() Returns the number of elements in this list.
void	**sort**(**Comparator**<? super **E**> c) Sorts this list according to the order induced by the specified **Compar**
Spliterator<**E**>	**spliterator**() Creates a ***late-binding*** and *fail-fast* **Spliterator** over the elemer
List<**E**>	**subList**(int fromIndex, int toIndex) Returns a view of the portion of this list between the specified fromI exclusive.
Object[]	**toArray**() Returns an array containing all of the elements in this list in proper s
<T> T[]	**toArray**(T[] a) Returns an array containing all of the elements in this list in proper s the runtime type of the returned array is that of the specified array.
void	**trimToSize**() Trims the capacity of this ArrayList instance to be the list's curren

- Methods inherited from class java.util.AbstractList

`equals, hashCode`

- Methods inherited from class java.util.AbstractCollection

`containsAll, toString`

- Methods inherited from class java.lang.Object

`finalize, getClass, notify, notifyAll, wait, wait, wait`

- Methods inherited from interface java.util.List

`containsAll, equals, hashCode`

- Methods inherited from interface java.util.Collection

`parallelStream, stream`