Inbuilt Methods

In this lesson, the mostly used inbuilt methods of ArrayList class are explained.

We'll cover the following

- Adding an element
- Getting or Setting an element
- Size of an ArrayList
- Some more inbuilt methods

ArrayLists have plenty of inbuilt methods that can be accessed by the programmers to store and manage data. We will frequently come across method overloading in these inbuilt methods below.

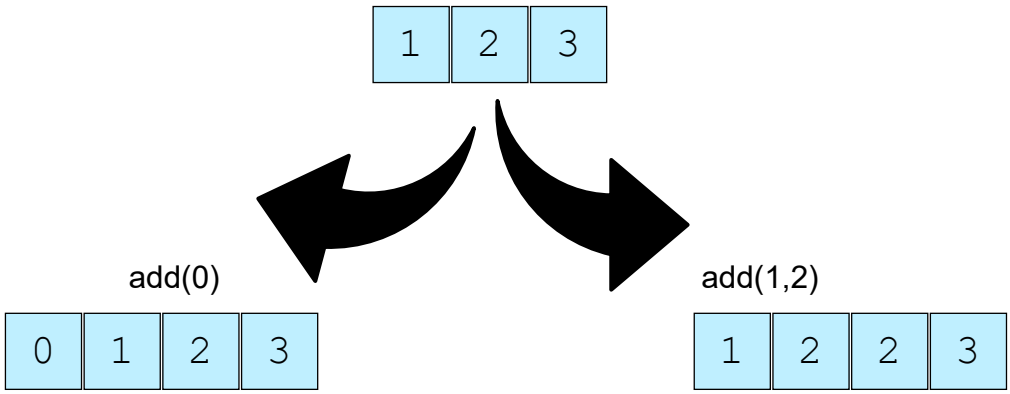
Adding an element

Addition of the elements to ArrayLists can be achieved using `add()` method:

Method Name	Description	Return Type
`add(` object `)`	The item is added to the end of the current ArrayList	`boolean`
`add(` *index*, object `)`	Adds a single object of the respective type at the specified *index* to the ArrayList	`void`

> **Note:** `add(` *index*, object `)` method pushes the element on the current index to

Note: `add( index, object )` method pushes the element on the current index to the next index.

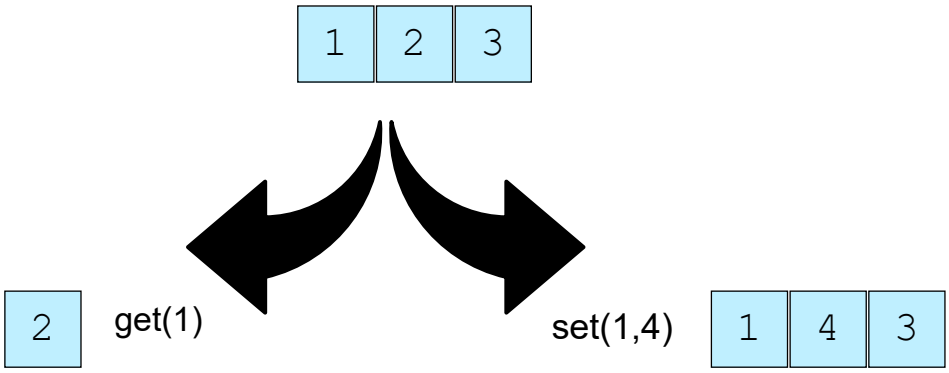


Adding an element

Getting or Setting an element

Access to the stored elements in an ArrayList can be achieved using `get()` method. We can also *set* the element at a specific position using the method `set()`.

Method Name	Description	Return Type
`get(` *index* `)`	Gets the object stored at the specified *index* in the ArrayList	`Object`
`set(` *index*, object `)`	Replaces the object at the specified *index* in the ArrayList with the *object* passed as parameter	`Object`



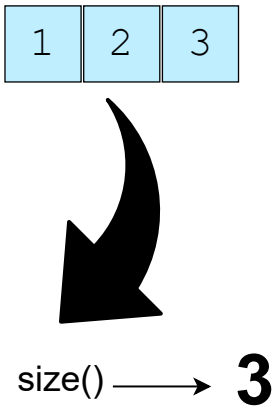
Getting or Setting an element

Size of an ArrayList

We can get the number of elements stored in an ArrayList using the `size()` method.

Method Name	Description	Return Type
`size()`	Returns the No. of elements stored in an ArrayList	`int`

Let's use the above methods in a coding example to strengthen our understanding.



Finding the length/size of the array list

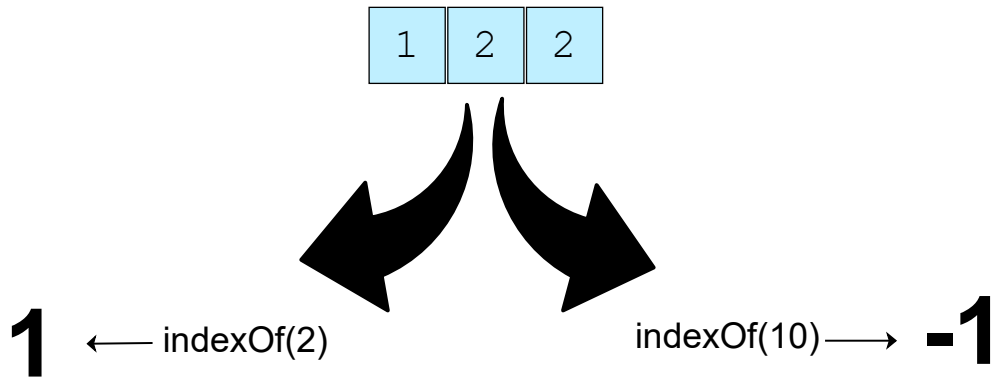
```
class ArrList {
    public static void main(String args[]) {
        ArrayList < String > emails = new ArrayList < String > (); //Instantiation
        emails.add("user1@abc.com"); //adds at index 0
        emails.add("user3@abc.com"); //adds at index 1

        System.out.println("The two added elements are:");
        System.out.println("1. " + emails.get(0));
        System.out.println("2. " + emails.get(1));
        System.out.println("The current size of the ArrayList is: " + emails.size() + "\n");

        emails.add(1, "user2@abc.com"); //adds at index 1 pushes the index 1
        //element to index 2
        System.out.println("After adding an element to index 1:");
        System.out.println("1. " + emails.get(0));
        System.out.println("2. " + emails.get(1));
        System.out.println("3. " + emails.get(2));
        System.out.println("The current size of the ArrayList is: " + emails.size());
    }
}
```

Inbuilt ArrayList Methods

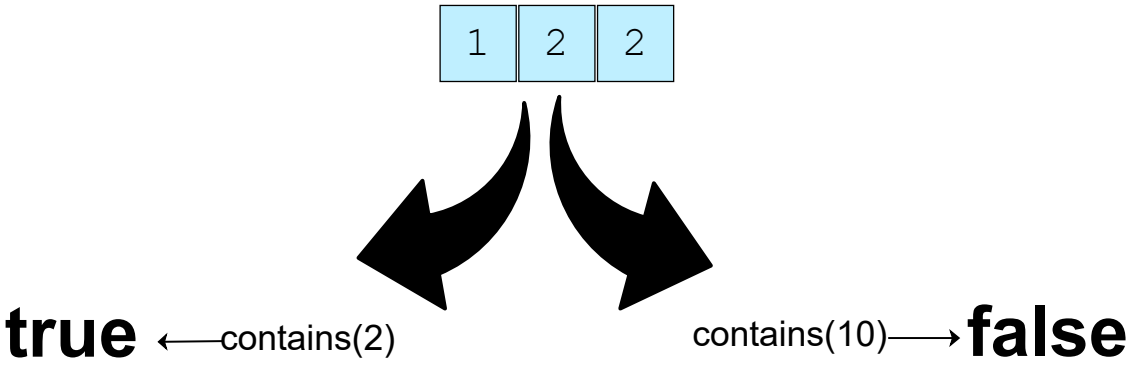
Some more inbuilt methods

Method Name	Description	Return Type
`indexOf(` object `)`	Returns the index of the first occurrence of the specified element, or -1 if the List does not contain this element	`int`
`contains(` object `)`	Returns true if this list contains the specified element	`boolean`
`remove(` object `)`	Returns true after removing the specified element	`boolean`

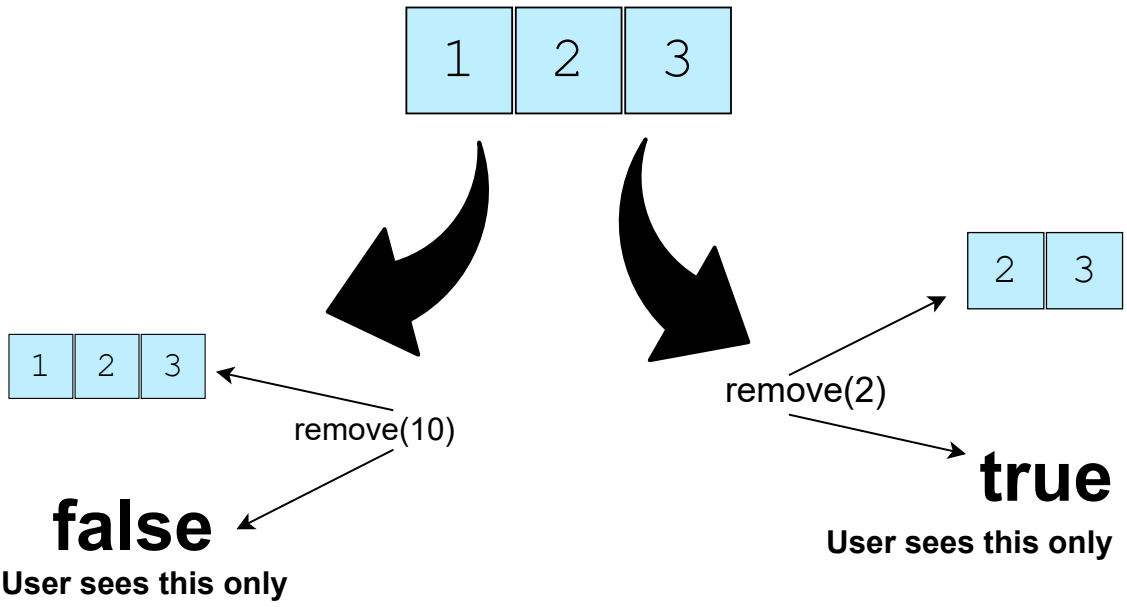
> *Duplicate* elements are **allowed** in the ArrayLists.



Finding the index of an element



Does the array list contain this value



Removing an element from the arraylist

A list of all the inbuilt methods can be found here.

Let’s put our understanding of ArrayLists into practice by taking up some coding challenges.