***ArrayList :--***

Java ArrayList is a **resizable** array in which you can store similar type of objects. We can add to or remove objects of ArrayList dynamically, because ArrayList is resizable.

## Create an ArrayList / Empty ArrayList in Java

To create an Empty ArrayList in Java, you can use new keyword and ArrayList constructor with no arguments passed to it.

Syntax to create an empty ArrayList.

|  |
| --- |
| ArrayList<DataType> myList = new ArrayList<DateType>(); |

### Example

|  |
| --- |
| import java.util.ArrayList;   public class ArrayListExample {      public static void main(String[] args) {  **ArrayList<String> names = new ArrayList<String>();**  **ArrayList<Integer> Age = new ArrayList<Integer>();**      }  } |
| Java ArrayList.add() To add an element to an ArrayList in Java, we can use add() method of ArrayList class. add() method has two variations based on the number of arguments.  **------ ArrayList.add(element) and**  **------ ArrayList.add(index, element**) |

* Declaration Time –

ArrayList<String>arrayList = new ArrayList<String>Arrays.asList("a","b));

**------ ArrayList.add(element)**

ArrayList.add() appends the specified element to the end of this ArrayList.

Syntax –

**ArrayList.add(E element)**

### Example 1 – add(element)

In this example, we will take an ArrayList of Strings initialized with four elements. We then use add(element) method to append the String "k" at the end of this ArrayList.

**Java Program**

|  |
| --- |
| public class Example {      public static void main(String[] args) {          ArrayList<String> arrayList = new ArrayList<String>());          String element = "k";          arrayList.add(element);  arrayList.add("a");  arrayList.add("b");          System.out.println("ArrayList : "+arrayList);      }  } |

**Output**

|  |
| --- |
| ArrayList : [k, a, b]  **------ ArrayList.add(index, element**)  Syntax –  **ArrayList.add(index, element)** Example 1 – add(index, element) arrayList.add("a");  arrayList.add(0, "b");  arrayList.add(1, "c");          System.out.println("ArrayList : "+arrayList);  **Output**    ArrayList : [b, c, a] |
|  |

**# Get the size of an ArrayList in Java : list.size()**

The size of an ArrayList can be obtained by using **the java.util.ArrayList.size(**) method as it returns the number of elements in the ArrayList i.e. the size.

import java.util.ArrayList;

import java.util.List;

public class Demo {

public static void main(String[] args) {

List aList = new ArrayList();

aList.add("Apple");

aList.add("Mango");

aList.add("Guava");

aList.add("Orange");

aList.add("Peach");

System.out.println("The size of the ArrayList is: " + **aList.size()**);

}

}

## Output

The size of the ArrayList is: 5

## get(int index)

ArrayList.get() returns the element at the specified position in this ArrayList.

### Syntax

The syntax of get() method with index as argument is

|  |
| --- |
| ArrayList.get(int index) |

### Example 1 – get(index)

In this example, we will define an ArrayList of Strings and initialize it with some elements in it. We will use ArrayList.get() method to get the elements at index 0, 1, 2 and 3 and print them to console.

**Java Program**

|  |
| --- |
| import java.util.\*;   public class Example {      public static void main(String[] args) {          ArrayList<String> arrayList = new ArrayList<String>();          arrayList.add("a");          arrayList.add("b");          arrayList.add("c");          arrayList.add("d");          arrayList.add("e");            System.out.println(arrayList.get(0));          System.out.println(arrayList.get(1));          System.out.println(arrayList.get(2));          System.out.println(arrayList.get(3));      }  } |

**Output**

|  |
| --- |
| a  b  c  d |

## Java ArrayList.set() – Examples

ArrayList.set() replaces the element at the specified position in this ArrayList with the specified element.

### Syntax

The syntax of set() method with index and element as arguments is

|  |
| --- |
| ArrayList.set(int index, E element) |

### Example 1 – set(index, element)

In this example, we will initialize an ArrayList with four strings "a", "b", "c" and "d". We will replace the element at index 2 with the element "m". Since, the element at index 2 is "c", set() method sets the element at index 2 with element "m" and returns the element "c".

**Java Program**

|  |
| --- |
| public class Example {      public static void main(String[] args) {          ArrayList<String> arrayList = new ArrayList<String>();          arrayList.add("a");          arrayList.add("b");          arrayList.add("c");          arrayList.add("d");          System.out.println("Original ArrayList    : " + arrayList);            int index = 2;          String element = "m";          String result = arrayList.set(index, element);          System.out.println("Return value from set() : " + result);          System.out.println("ArrayList after set() : " + arrayList);      }  } |

**Output**

|  |
| --- |
| Original ArrayList    : [a, b, c, d]  Return value from set() : c  ArrayList after set() : [a, b, m, d] |