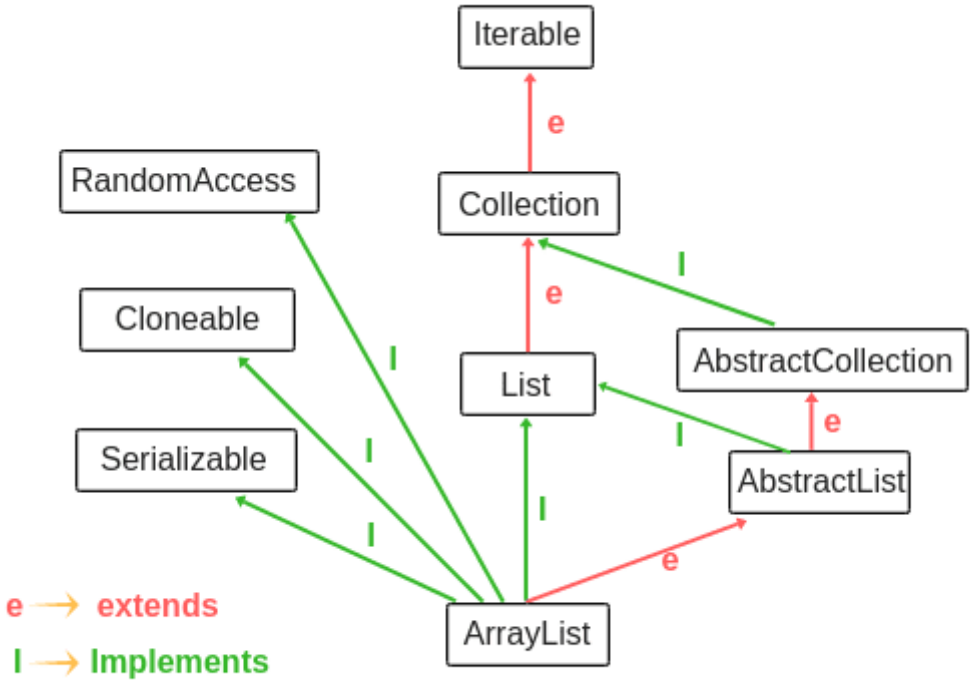
ArrayList

ArrayList in Java is a dynamic array that allows us to store multiple objects of any class or data type. It is similar to an array, but there is no fixed size limit.




```
package ArrayListTest;
import java.util.ArrayList;
public class AddExample {
public static void main(String[] args)
{
// Create an object of the non-generic ArrayList.
   ArrayList al = new ArrayList(); // list 1 with default capacity 10.
    al.add("A");
    al.add("B");
    al.add(20);
    al.add("A");
    al.add(null);
   System.out.println(al);

// Create an object of another non-generic ArrayList.
   ArrayList al1 = new ArrayList(); // list 2.
    al1.add("a");
    al1.add("b");
    al1.add("c");
```

```
// Call addAll(Collection c) method using reference variable al to add all elements at
the end of the list1.
    al.addAll(al1);
    System.out.println(al);

// Call addAll(int index, Collection c) method using reference al1 to add all elements
at specified position 2.
    al1.addAll(2, al);
   System.out.println(al1);
  }
}
```

```
package ArrayListTest;
import java.util.ArrayList;
public class RemoveEx
{
public static void main(String[] args)
{
// Create a generic Arraylist object of String type.
// This means the compiler will show an error if we try to put any other element than
 String.
   ArrayList<String> al = new ArrayList<String>(); // Default capacity is 10.

// Adding elements of String type.
   al.add("A");
   al.add("B");
   al.add("C");
   al.add("D");
   al.add(null);
   al.add("D");
   System.out.println(al);

// Call remove() method to remove element D.
// This statement removes the first occurrence of the specified element D at position
 3, not from the position 5.
   al.remove("D");
   System.out.println(al);

   al.remove(3);
   System.out.println(al);

// Call set() method to replace an element D with a null element at position 3.
    al.set(3, null);
    System.out.println(al);
  }
}
```

```
package arrayListPrograms;
import java.util.ArrayList;
public class ArrayListTest {
public static void main(String[] args)
{
```

```
 ArrayList al = new ArrayList();
  al.add("Apple");
  al.add("Orange");
  al.add("Banana");
  al.add("Gauva");
 System.out.println(al);

// Call get() method using object reference variable 'al' to get the specified elemen
t.
// Since return type of get() method is String, we will store it by using a fruitsName
variable with data type String.
   String fruitsName = al.get(2);
   System.out.println(fruitsName);

// Call size() method to get the number of elements present in the list.
// Since return type of size method is an integer, we will store it by using variable
 numberOfElements with data type integer.
   int numberOfElements = al.size();
   System.out.println(numberOfElements);

// Check apple element is present or not.
   boolean check = al.contains("Apple");
   System.out.println(check);
 }
}
```

```
package ArrayListTest;
import java.util.ArrayList;
public class Test {
public static void main(String[] args)
{
 ArrayList<Integer> list = new ArrayList<Integer>();
  list.add(10);
  list.add(20);
  list.add(30);
  list.add(40);

  System.out.println(list);
  int pos = list.indexOf(30);
  System.out.println(pos);

  int lastPos = list.lastIndexOf(40);
  System.out.println(lastPos);
 }
}
```

```
Output:
```