

add(): Este método se utiliza para agregar un elemento a una estructura de datos, como una lista o un conjunto.

remove(): El método **remove()** se utiliza para eliminar un elemento específico de una estructura de datos. Debes proporcionar el elemento que deseas eliminar como argumento.

get(): **get()** se utiliza para acceder a un elemento específico en una estructura de datos, como una lista o un mapa. Debes proporcionar un índice o clave como argumento y obtendrás el elemento correspondiente.

size(): Este método devuelve el número de elementos en una estructura de datos.

isEmpty(): **isEmpty()** se utiliza para verificar si una estructura de datos está vacía o no. Si la estructura de datos no contiene elementos, este método devuelve **true**, de lo contrario, devuelve **false**.

2. Investigue ¿Qué hace el método isEmpty de la clase ArrayList? y Escribe un programa que lo utilice.

```
import java.util.ArrayList;
```

```
public class Main {  
    public static void main(String[] args) {  
        ArrayList<String> miLista = new ArrayList<String>();  
  
        if (miLista.isEmpty()) {  
            System.out.println("La lista está vacía.");  
        } else {  
            System.out.println("La lista no está vacía.");  
        }  
  
        miLista.add("Ejemplo");  
  
        if (miLista.isEmpty()) {  
            System.out.println("La lista está vacía.");  
        } else {
```

```
        System.out.println("La lista no está vacía.");
```

```
    }  
}  
}
```

3. Dado un ArrayList de números enteros, encontrar e imprimir el número más grande.

```
import java.util.ArrayList;
```

```
public class Main {  
    public static void main(String[] args) {  
        ArrayList<Integer> numeros = new ArrayList<Integer>();  
  
        numeros.add(10);  
        numeros.add(5);  
        numeros.add(25);  
        numeros.add(2);  
        numeros.add(15);  
  
        int max = numeros.get(0);  
  
        for (int numero : numeros) {  
            if (numero > max) {  
                max = numero;  
            }  
        }  
  
        System.out.println("El número más grande en la lista es: " + max);  
    }  
}
```

4. Dado un ArrayList de números enteros, eliminar todos los números que sean pares.

```
import java.util.ArrayList;
```

```

public class Main {
    public static void main(String[] args) {
        ArrayList<Integer> numeros = new ArrayList<Integer>();

        numeros.add(10);
        numeros.add(5);
        numeros.add(25);
        numeros.add(2);
        numeros.add(15);

        for (int i = numeros.size() - 1; i >= 0; i--) {
            if (numeros.get(i) % 2 == 0) {
                numeros.remove(i);
            }
        }

        System.out.println("Lista después de eliminar los números pares: " + numeros);
    }
}

```

5. Dado un ArrayList de números, muestra el tercer elemento de la lista.

```

import java.util.ArrayList;

public class Main {
    public static void main(String[] args) {
        ArrayList<Integer> numeros = new ArrayList<Integer>();

        numeros.add(10);
        numeros.add(5);
        numeros.add(25);
        numeros.add(2);
    }
}

```

```
numeros.add(15);

if (numeros.size() > 2) {
    int tercerElemento = numeros.get(2);
    System.out.println("El tercer elemento de la lista es: " + tercerElemento);
} else {
    System.out.println("La lista no tiene suficientes elementos para mostrar el tercer elemento.");
}
}
```

6. Cambia el segundo nombre del ArrayList "nombres" por "Ana".

```
import java.util.ArrayList;
```

```
public class Main {
```

```
    public static void main(String[] args) {
        ArrayList<String> nombres = new ArrayList<String>();

        nombres.add("Juan");
        nombres.add("Pedro");
        nombres.add("Luis");
        nombres.add("María");
```

```
        if (nombres.size() > 1) {
```

```
            nombres.set(1, "Ana");
            System.out.println("El segundo nombre ha sido cambiado por 'Ana'.");
        } else {
```

```
            System.out.println("La lista no tiene suficientes nombres para realizar el cambio.");
        }
```

```
        System.out.println("Nombres después del cambio: " + nombres);
    }
}
```

7. Elimina el primer nombre del ArrayList "nombres".

```
import java.util.ArrayList;
```

```
public class Main {
    public static void main(String[] args) {
        ArrayList<String> nombres = new ArrayList<String>();
```

```
        nombres.add("Juan");
        nombres.add("Pedro");
        nombres.add("Luis");
        nombres.add("María");
```

```
        if (!nombres.isEmpty()) {
            nombres.remove(0);
            System.out.println("El primer nombre ha sido eliminado.");
        } else {
            System.out.println("La lista está vacía, no se puede eliminar el primer nombre.");
        }
```

```
        System.out.println("Nombres después de la eliminación: " + nombres);
    }
}
```

8. Escribe un programa que verifique si un ArrayList está vacío o no.

```
import java.util.ArrayList;
```

```
public class Main {  
    public static void main(String[] args) {  
        ArrayList<String> miLista = new ArrayList<String>();  
  
        miLista.add("Elemento 1");  
        miLista.add("Elemento 2");  
  
        if (miLista.isEmpty()) {  
            System.out.println("La lista está vacía.");  
        } else {  
            System.out.println("La lista no está vacía. Contiene " + miLista.size() + " elementos.");  
        }  
    }  
}
```

9. Verifica si el nombre "Carlos" está presente en el ArrayList "nombres".

```
import java.util.ArrayList;  
  
public class Main {  
    public static void main(String[] args) {  
        ArrayList<String> nombres = new ArrayList<String>();  
  
        nombres.add("Juan");  
        nombres.add("Pedro");  
        nombres.add("Luis");  
        nombres.add("María");  
    }  
}
```

```
String nombreBuscado = "Carlos";
```

```
boolean encontrado = false;
```

```

for (String nombre : nombres) {
    if (nombre.equals(nombreBuscado)) {
        encontrado = true;
        break;
    }
}

if (encontrado) {
    System.out.println(nombreBuscado + " está presente en la lista de nombres.");
} else {
    System.out.println(nombreBuscado + " no está presente en la lista de nombres.");
}
}
}

```

10. Escribe un programa que recorra un ArrayList de números y muestre cada número.

```

import java.util.ArrayList;

public class Main {
    public static void main(String[] args) {
        ArrayList<Integer> numeros = new ArrayList<Integer>();

        numeros.add(10);
        numeros.add(5);
        numeros.add(25);
        numeros.add(2);
        numeros.add(15);

        int i = 1;
        for (int numero : numeros) {

```

```
        System.out.println("Número " + i + ": " + numero);
        i++;
    }
}

}
```

11. Duplica todos los elementos de un ArrayList de números enteros.

```
import java.util.ArrayList;
```

```
public class Main {
    public static void main(String[] args) {
        ArrayList<Integer> numeros = new ArrayList<Integer>();
        // Agrega números a la lista
        numeros.add(10);
        numeros.add(5);
        numeros.add(25);
        numeros.add(2);
        numeros.add(15);
```

```
        ArrayList<Integer> numerosDuplicados = new ArrayList<Integer>();
        for (int numero : numeros) {
            numerosDuplicados.add(numero * 2);
        }
```

```
        System.out.println("Lista original: " + numeros);
        System.out.println("Lista con elementos duplicados: " + numerosDuplicados);
    }
}
```

12. Escribe un programa que elimine todos los elementos de un ArrayList.

```
import java.util.ArrayList;
```

```

public class Main {

    public static void main(String[] args) {
        ArrayList<Integer> numeros = new ArrayList<Integer>();

        numeros.add(10);
        numeros.add(5);
        numeros.add(25);
        numeros.add(2);
        numeros.add(15);

        numeros = new ArrayList<Integer>();

        if (numeros.isEmpty()) {
            System.out.println("Todos los elementos han sido eliminados. La lista está vacía.");
        } else {
            System.out.println("La lista no está vacía. Contiene elementos.");
        }
    }
}

```

13. Añade el nombre "Pedro" en la segunda posición del ArrayList "nombres".

```

import java.util.ArrayList;

public class Main {

    public static void main(String[] args) {
        ArrayList<String> nombres = new ArrayList<String>();

        nombres.add("Juan");
        nombres.add("Luis");

```

```
nombres.add("María");

nombres.add(1, "Pedro");

System.out.println("Nombres después de agregar 'Pedro' en la segunda posición: " + nombres);

}

}
```

14. Convierte un ArrayList de cadenas en un array de cadenas.

```
import java.util.ArrayList;
```

```
import java.util.Arrays;
```

```
public class Main {
```

```
    public static void main(String[] args) {
```

```
        ArrayList<String> listaDeCadenas = new ArrayList<String>();
```

```
        listaDeCadenas.add("Manzana");
```

```
        listaDeCadenas.add("Banana");
```

```
        listaDeCadenas.add("Cereza");
```

```
        String[] arrayDeCadenas = listaDeCadenas.toArray(new String[0]);
```

```
        System.out.println("Array de cadenas: " + Arrays.toString(arrayDeCadenas));
```

```
    }
```

```
}
```

15. Dado un ArrayList de nombres, ordénalos alfabéticamente.

```
import java.util.ArrayList;
```

```

public class Main {
    public static void main(String[] args) {
        ArrayList<String> nombres = new ArrayList<String>();
        nombres.add("Juan");
        nombres.add("Pedro");
        nombres.add("Luis");
        nombres.add("María");

        // Ordena los nombres alfabéticamente
        nombres.sort(null);

        System.out.println("Nombres ordenados alfabéticamente: " + nombres);
    }
}

```

16. Muestra el primer elemento de un ArrayList de cadenas.

```

import java.util.ArrayList;

public class Main {
    public static void main(String[] args) {
        ArrayList<String> cadenas = new ArrayList<String>();

        cadenas.add("Manzana");
        cadenas.add("Banana");
        cadenas.add("Cereza");

        if (!cadenas.isEmpty()) {
            String primerElemento = cadenas.get(0);
            System.out.println("Primer elemento de la lista: " + primerElemento);
        }
    }
}

```

```
    } else {
        System.out.println("La lista está vacía.");
    }
}
```

17. Muestra el último elemento de un ArrayList de cadenas.

```
import java.util.ArrayList;
```

```
public class Main {
    public static void main(String[] args) {
        ArrayList<String> cadenas = new ArrayList<String>();

        cadenas.add("Manzana");
        cadenas.add("Banana");
        cadenas.add("Cereza");

        if (!cadenas.isEmpty()) {
            String ultimoElemento = cadenas.get(cadenas.size() - 1);
            System.out.println("Último elemento de la lista: " + ultimoElemento);
        } else {
            System.out.println("La lista está vacía.");
        }
    }
}
```

18. Crea un ArrayList de números con los valores 5, 10 y 15.

```
import java.util.ArrayList;
```

```
public class Main {
    public static void main(String[] args) {
```

```
ArrayList<Integer> numeros = new ArrayList<Integer>();
numeros.add(5);
numeros.add(10);
numeros.add(15);

System.out.println("ArrayList de números: " + numeros);
}
```

19. Cambia todos los elementos de un ArrayList de cadenas por el valor "Hola".

```
import java.util.ArrayList;

public class Main {
    public static void main(String[] args) {
        ArrayList<String> cadenas = new ArrayList<String>();

        cadenas.add("Giannis");
        cadenas.add("#Lamelo");
        cadenas.add("Ja");

        for (int i = 0; i < cadenas.size(); i++) {
            cadenas.set(i, "Hola");
        }
    }
}
```

System.out.println("ArrayList de cadenas después del cambio: " + cadenas);
}

20. Crea un ArrayList que contenga tres ArrayLists vacíos.

```
import java.util.ArrayList;
```

```
public class Main {  
    public static void main(String[] args) {  
  
        ArrayList<ArrayList<Integer>> arrayListDeArrayLists = new ArrayList<ArrayList<Integer>>();  
  
        arrayListDeArrayLists.add(new ArrayList<Integer>());  
        arrayListDeArrayLists.add(new ArrayList<Integer>());  
        arrayListDeArrayLists.add(new ArrayList<Integer>());  
  
        System.out.println("ArrayList de ArrayLists: " + arrayListDeArrayLists);  
    }  
}
```