**Trabajo practico nro 9**

|  |  |  |
| --- | --- | --- |
|  | **Asignatura: LABORATORIO 1** | |
|  | |
| **Cursado:** | **Horas** **semanales**: |
| IGNACIO MOLINA | **Horas semestrales:** |
| **Carrera**: *Tecnicatura Universitaria en Programación* | **Nivel (Año):** |
| **Ciclo Lectivo: 2023** |

* **Tema:**

**Colecciones ArrayList**

* **Enunciados:**

**Empezaremos por unos ejercicios básicos de programas Java con estructura secuencial, es decir, en estos programas no hay instrucciones condicionales ni repetitivas. En la mayoría de ellos las operaciones a realizar son: lectura de datos por teclado, realizar alguna operación con esos datos y mostrar resultados por pantalla.**

**Recomendaciones: Leer la teoría, investigar, consultar si existen dudas y realizar el trabajo en clases. La entrega de este trabajo practico es individual.**

* Crea un ArrayList de números enteros y agrega 5 números a la lista.

import java.util.ArrayList;

public class Main {

public static void main(String[] args) {

// Crear un ArrayList de números enteros

ArrayList<Integer> numeros = new ArrayList<>();

// Agregar 5 números a la lista

numeros.add(10);

numeros.add(20);

numeros.add(30);

numeros.add(40);

numeros.add(50);

// Mostrar la lista

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

}

}

* Crea un ArrayList de cadenas y agrega 3 nombres a la lista.

import java.util.ArrayList;

public class Main {

public static void main(String[] args) {

// Crear un ArrayList de cadenas

ArrayList<String> nombres = new ArrayList<>();

// Agregar 3 nombres a la lista

nombres.add("Ana");

nombres.add("Juan");

nombres.add("María");

// Mostrar la lista

System.out.println("ArrayList de nombres: " + nombres);

}

}

* Imprime todos los elementos de un ArrayList de enteros.

import java.util.ArrayList;

public class ImprimirArrayList {

public static void main(String[] args) {

ArrayList<Integer> numeros = new ArrayList<>();

numeros.add(1);

numeros.add(2);

numeros.add(3);

numeros.add(4);

numeros.add(5);

// Imprimir todos los elementos

System.out.println("Elementos del ArrayList de enteros:");

for (Integer numero : numeros) {

System.out.println(numero);

}

}

}

* Encuentra la longitud (número de elementos) de un ArrayList de caracteres.

import java.util.ArrayList;

public class LongitudArrayList {

public static void main(String[] args) {

ArrayList<Character> caracteres = new ArrayList<>();

caracteres.add('a');

caracteres.add('b');

caracteres.add('c');

// Encontrar la longitud

int longitud = caracteres.size();

System.out.println("Longitud del ArrayList de caracteres: " + longitud);

}

}

* Elimina un elemento específico de un ArrayList de cadenas.

import java.util.ArrayList;

public class EliminarElementoArrayList {

public static void main(String[] args) {

ArrayList<String> nombres = new ArrayList<>();

nombres.add("Juan");

nombres.add("Ana");

nombres.add("María");

// Eliminar un elemento específico

String nombreEliminar = "Ana";

nombres.remove(nombreEliminar);

// Mostrar la lista después de la eliminación

System.out.println("ArrayList de nombres después de eliminar '" + nombreEliminar + "': " + nombres);

}

}

* Comprueba si un ArrayList de números enteros está vacío.

import java.util.ArrayList;

public class ComprobarVacioArrayList {

public static void main(String[] args) {

ArrayList<Integer> numeros = new ArrayList<>();

// Comprobar si está vacío

boolean estaVacio = numeros.isEmpty();

System.out.println("El ArrayList de enteros está vacío: " + estaVacio);

}

}

* Encuentra el elemento más grande en un ArrayList de números.

import java.util.ArrayList;

import java.util.Collections;

public class ElementoMasGrandeArrayList {

public static void main(String[] args) {

ArrayList<Integer> numeros = new ArrayList<>();

numeros.add(10);

numeros.add(5);

numeros.add(20);

numeros.add(15);

// Encontrar el elemento más grande

Integer maximo = Collections.max(numeros);

System.out.println("El elemento más grande en el ArrayList de enteros: " + maximo);

}

}

* Copia todos los elementos de un ArrayList en otro ArrayList nuevo.

import java.util.ArrayList;

public class CopiarArrayList {

public static void main(String[] args) {

ArrayList<Integer> numerosOriginal = new ArrayList<>();

numerosOriginal.add(1);

numerosOriginal.add(2);

numerosOriginal.add(3);

// Copiar elementos a un nuevo ArrayList

ArrayList<Integer> numerosCopia = new ArrayList<>(numerosOriginal);

// Mostrar el ArrayList original y la copia

System.out.println("ArrayList original: " + numerosOriginal);

System.out.println("ArrayList copia: " + numerosCopia);

}

}

* Invierte un ArrayList de enteros (el primer elemento se convierte en el último y viceversa).

import java.util.ArrayList;

import java.util.Collections;

public class InvertirArrayList {

public static void main(String[] args) {

ArrayList<Integer> numeros = new ArrayList<>();

numeros.add(1);

numeros.add(2);

numeros.add(3);

numeros.add(4);

// Invertir el ArrayList

Collections.reverse(numeros);

// Mostrar el ArrayList invertido

System.out.println("ArrayList de enteros invertido: " + numeros);

}

}

* Combina dos ArrayLists de enteros en uno solo.

import java.util.ArrayList;

import java.util.Collections;

public class CombinarArrayLists {

public static void main(String[] args) {

// ArrayLists originales

ArrayList<Integer> numeros1 = new ArrayList<>();

numeros1.add(1);

numeros1.add(2);

numeros1.add(3);

ArrayList<Integer> numeros2 = new ArrayList<>();

numeros2.add(4);

numeros2.add(5);

// Combinar los ArrayLists

ArrayList<Integer> numerosCombinados = new ArrayList<>(numeros1);

numerosCombinados.addAll(numeros2);

// Mostrar el ArrayList combinado

System.out.println("ArrayList combinado: " + numerosCombinados);

}

}

* Crea un ArrayList de números enteros y elimina todos los elementos pares.

import java.util.ArrayList;

import java.util.Iterator;

public class EliminarElementosPares {

public static void main(String[] args) {

ArrayList<Integer> numeros = new ArrayList<>();

numeros.add(1);

numeros.add(2);

numeros.add(3);

numeros.add(4);

numeros.add(5);

// Eliminar elementos pares

Iterator<Integer> iterador = numeros.iterator();

while (iterador.hasNext()) {

if (iterador.next() % 2 == 0) {

iterador.remove();

}

}

// Mostrar el ArrayList después de eliminar elementos pares

System.out.println("ArrayList después de eliminar elementos pares: " + numeros);

}

}

* Busca un elemento específico en un ArrayList de cadenas y devuelve su índice.

import java.util.ArrayList;

public class BuscarIndiceElemento {

public static void main(String[] args) {

ArrayList<String> cadenas = new ArrayList<>();

cadenas.add("Manzana");

cadenas.add("Banana");

cadenas.add("Cereza");

// Elemento a buscar

String elementoBuscado = "Banana";

// Buscar el índice del elemento

int indice = cadenas.indexOf(elementoBuscado);

// Mostrar el resultado

if (indice != -1) {

System.out.println("El elemento '" + elementoBuscado + "' se encuentra en el índice: " + indice);

} else {

System.out.println("El elemento '" + elementoBuscado + "' no se encuentra en el ArrayList.");

}

}

}

* Comprueba si dos ArrayLists son iguales (tienen los mismos elementos en el mismo orden).

import java.util.ArrayList;

public class ComprobarIgualdadArrayLists {

public static void main(String[] args) {

// ArrayLists originales

ArrayList<Integer> numeros1 = new ArrayList<>();

numeros1.add(1);

numeros1.add(2);

numeros1.add(3);

ArrayList<Integer> numeros2 = new ArrayList<>();

numeros2.add(1);

numeros2.add(2);

numeros2.add(3);

// Comprobar igualdad

boolean sonIguales = numeros1.equals(numeros2);

// Mostrar el resultado

if (sonIguales) {

System.out.println("Los ArrayLists son iguales.");

} else {

System.out.println("Los ArrayLists no son iguales.");

}

}

}

* Encuentra el elemento más pequeño en un ArrayList de números.

import java.util.ArrayList;

import java.util.Collections;

public class ElementoMasPequenoArrayList {

public static void main(String[] args) {

ArrayList<Integer> numeros = new ArrayList<>();

numeros.add(10);

numeros.add(5);

numeros.add(20);

numeros.add(15);

// Encontrar el elemento más pequeño

Integer minimo = Collections.min(numeros);

// Mostrar el resultado

System.out.println("El elemento más pequeño en el ArrayList de enteros: " + minimo);

}

}

* Crea un ArrayList de números enteros y suma todos los elementos.

import java.util.ArrayList;

public class SumaElementosArrayList {

public static void main(String[] args) {

ArrayList<Integer> numeros = new ArrayList<>();

numeros.add(1);

numeros.add(2);

numeros.add(3);

numeros.add(4);

numeros.add(5);

// Sumar todos los elementos

int suma = 0;

for (int numero : numeros) {

suma += numero;

}

// Mostrar el resultado

System.out.println("La suma de los elementos en el ArrayList de enteros: " + suma);

}

}

* Crea un ArrayList de cadenas y concatena todas las cadenas en una sola.

import java.util.ArrayList;

public class ConcatenarCadenasArrayList {

public static void main(String[] args) {

ArrayList<String> cadenas = new ArrayList<>();

cadenas.add("Hola");

cadenas.add(" ");

cadenas.add("Mundo");

// Concatenar todas las cadenas

StringBuilder resultado = new StringBuilder();

for (String cadena : cadenas) {

resultado.append(cadena);

}

// Mostrar el resultado

System.out.println("Cadenas concatenadas: " + resultado.toString());

}

}

* Elimina todos los elementos duplicados de un ArrayList de cadenas.

import java.util.ArrayList;

import java.util.HashSet;

public class EliminarDuplicadosArrayList {

public static void main(String[] args) {

ArrayList<String> cadenas = new ArrayList<>();

cadenas.add("A");

cadenas.add("B");

cadenas.add("A");

cadenas.add("C");

cadenas.add("B");

// Eliminar duplicados utilizando un HashSet

HashSet<String> conjunto = new HashSet<>(cadenas);

cadenas.clear();

cadenas.addAll(conjunto);

// Mostrar el ArrayList después de eliminar duplicados

System.out.println("ArrayList después de eliminar duplicados: " + cadenas);

}

}

* Crea un ArrayList de enteros y encuentra la suma de los elementos en índices pares.

import java.util.ArrayList;

public class SumaElementosIndicesPares {

public static void main(String[] args) {

ArrayList<Integer> numeros = new ArrayList<>();

numeros.add(1);

numeros.add(2);

numeros.add(3);

numeros.add(4);

numeros.add(5);

// Calcular la suma de los elementos en índices pares

int suma = 0;

for (int i = 0; i < numeros.size(); i += 2) {

suma += numeros.get(i);

}

// Mostrar el resultado

System.out.println("La suma de los elementos en índices pares: " + suma);

}

}

* Comprueba si un ArrayList de enteros contiene un número específico.

import java.util.ArrayList;

public class Main {

public static void main(String[] args) {

ArrayList<Integer> numeros = new ArrayList<>();

numeros.add(1);

numeros.add(2);

numeros.add(3);

int numeroBuscado = 2;

if (numeros.contains(numeroBuscado)) {

System.out.println("El ArrayList contiene el número " + numeroBuscado);

} else {

System.out.println("El ArrayList no contiene el número " + numeroBuscado);

}

}

}

* Encuentra el elemento en un ArrayList de cadenas que tiene la longitud más larga.

import java.util.ArrayList;

public class Main {

public static void main(String[] args) {

ArrayList<String> cadenas = new ArrayList<>();

cadenas.add("Hola");

cadenas.add("Hola, mundo");

cadenas.add("¡Hola a todos!");

String masLarga = cadenas.stream()

.max((s1, s2) -> Integer.compare(s1.length(), s2.length()))

.orElse(null);

System.out.println("La cadena más larga es: " + masLarga);

}

}

* Encuentra el promedio de todos los elementos en un ArrayList de números enteros.

import java.util.ArrayList;

public class Main {

public static void main(String[] args) {

ArrayList<Integer> numeros = new ArrayList<>();

numeros.add(5);

numeros.add(10);

numeros.add(15);

double promedio = numeros.stream().mapToDouble(Integer::doubleValue).average().orElse(0);

System.out.println("El promedio de los números es: " + promedio);

}

}

* Crea un ArrayList de enteros y ordénalo de forma ascendente.

import java.util.ArrayList;

import java.util.Collections;

public class Main {

public static void main(String[] args) {

ArrayList<Integer> numeros = new ArrayList<>();

numeros.add(3);

numeros.add(1);

numeros.add(5);

Collections.sort(numeros);

System.out.println("ArrayList ordenado: " + numeros);

}

}

* Crea un ArrayList de números enteros y elimina todos los elementos mayores que un valor específico.

import java.util.ArrayList;

import java.util.Iterator;

public class Main {

public static void main(String[] args) {

ArrayList<Integer> numeros = new ArrayList<>();

numeros.add(3);

numeros.add(8);

numeros.add(5);

int valorLimite = 6;

Iterator<Integer> iterador = numeros.iterator();

while (iterador.hasNext()) {

if (iterador.next() > valorLimite) {

iterador.remove();

}

}

System.out.println("ArrayList después de eliminar elementos mayores que " + valorLimite + ": " + numeros);

}

}

* Encuentra la cantidad de veces que un elemento específico aparece en un ArrayList de cadenas.

import java.util.ArrayList;

import java.util.Collections;

public class Main {

public static void main(String[] args) {

ArrayList<String> cadenas = new ArrayList<>();

cadenas.add("hola");

cadenas.add("mundo");

cadenas.add("hola");

String elementoBuscado = "hola";

int frecuencia = Collections.frequency(cadenas, elementoBuscado);

System.out.println("La cadena '" + elementoBuscado + "' aparece " + frecuencia + " veces en el ArrayList.");

}

}

* Crea un ArrayList de cadenas y ordena las cadenas alfabéticamente.

import java.util.ArrayList;

import java.util.Collections;

public class Main {

public static void main(String[] args) {

ArrayList<String> cadenas = new ArrayList<>();

cadenas.add("zorro");

cadenas.add("perro");

cadenas.add("gato");

Collections.sort(cadenas);

System.out.println("ArrayList ordenado alfabéticamente: " + cadenas);

}

}