**Trabajo práctico nro. 11 – Parte 1**

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

* **Tema:**

**Arraylist y Clases en Java**

* **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.**

* Diseñar un programa que lea y guarde marcas de Autos en un ArrayList de tipo

String. El programa pedirá una marca de Auto en un bucle, el mismo se guardará

en la lista y después se le preguntará al usuario si quiere guardar otro Auto o si

quiere salir. Si decide salir, se mostrará todos los autos guardados en el ArrayList.

-°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°-

import java.util.ArrayList;

import java.util.Scanner;

public class ProgramaMarcasAutos {

public static void main(String[] args) {

// Crear un ArrayList para almacenar las marcas de autos

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

// Crear un objeto Scanner para leer la entrada del usuario

Scanner scanner = new Scanner(System.in);

// Bucle para pedir marcas de autos al usuario

while (true) {

System.out.print("Ingrese la marca de un auto (o escriba 'salir' para terminar): ");

String marca = scanner.nextLine();

// Verificar si el usuario quiere salir

if (marca.equalsIgnoreCase("salir")) {

break; // Salir del bucle si el usuario decide salir

}

// Agregar la marca de auto al ArrayList

marcasAutos.add(marca);

}

// Mostrar todas las marcas de autos almacenadas en el ArrayList

System.out.println("Marcas de autos ingresadas:");

for (String marca : marcasAutos) {

System.out.println(marca);

}

// Cerrar el objeto Scanner

scanner.close();

}

}

-°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°-

* Define una clase EquipoDeFútbol que utiliza un ArrayList para mantener una lista de jugadores. La clase debe permitir agregar nuevos jugadores, eliminar jugadores y listar la plantilla.

-°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°-

import java.util.ArrayList;

public class EquipoDeFútbol {

// ArrayList para almacenar la lista de jugadores

private ArrayList<String> jugadores;

// Constructor para inicializar la lista de jugadores

public EquipoDeFútbol() {

jugadores = new ArrayList<>();

}

// Método para agregar un nuevo jugador a la lista

public void agregarJugador(String nombreJugador) {

jugadores.add(nombreJugador);

System.out.println("Jugador " + nombreJugador + " agregado al equipo.");

}

// Método para eliminar un jugador de la lista

public void eliminarJugador(String nombreJugador) {

if (jugadores.remove(nombreJugador)) {

System.out.println("Jugador " + nombreJugador + " eliminado del equipo.");

} else {

System.out.println("El jugador " + nombreJugador + " no está en el equipo.");

}

}

// Método para listar la plantilla de jugadores

public void listarPlantilla() {

if (jugadores.isEmpty()) {

System.out.println("La plantilla está vacía.");

} else {

System.out.println("Plantilla del equipo:");

for (String jugador : jugadores) {

System.out.println("- " + jugador);

}

}

}

public static void main(String[] args) {

// Ejemplo de uso de la clase EquipoDeFútbol

EquipoDeFútbol equipo = new EquipoDeFútbol();

equipo.agregarJugador("Messi");

equipo.agregarJugador("Ronaldo");

equipo.agregarJugador("Neymar");

equipo.listarPlantilla();

equipo.eliminarJugador("Ronaldo");

equipo.listarPlantilla();

}

}

-°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°-

* Crear una clase llamada Factura que represente una factura comercial. La clase debe tener los siguientes atributos:
* Número de factura: Un número único para identificar la factura.
* Fecha de emisión: La fecha en la que se emitió la factura.
* Cliente: El cliente al que se le emite la factura.
* Artículos: Una lista de artículos vendidos en la factura.
* Precio unitario: El precio unitario de cada artículo.
* Cantidad: La cantidad de cada artículo vendido.
* Subtotal: El subtotal de la factura sin impuestos.
* IVA: El impuesto sobre el valor añadido (IVA) de la factura.
* Total: El total de la factura, incluyendo el IVA.

La clase debe tener los siguientes métodos:

* Constructor: Un constructor que inicializa los atributos de la factura.
* Agregar artículo: Un método que agrega un artículo a la factura.
* Calcular subtotal: Un método que calcula el subtotal de la factura.
* Calcular IVA: Un método que calcula el IVA de la factura.
* Calcular total: Un método que calcula el total de la factura

-°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°-

import java.util.ArrayList;

import java.util.Date;

import java.util.List;

public class Factura {

private int numeroFactura;

private Date fechaEmision;

private String cliente;

private List<Articulo> articulos;

private double precioUnitario;

private int cantidad;

private double subtotal;

private double iva;

private double total;

// Constructor

public Factura(int numeroFactura, Date fechaEmision, String cliente) {

this.numeroFactura = numeroFactura;

this.fechaEmision = fechaEmision;

this.cliente = cliente;

this.articulos = new ArrayList<>();

this.precioUnitario = 0.0;

this.cantidad = 0;

this.subtotal = 0.0;

this.iva = 0.0;

this.total = 0.0;

}

// Método para agregar un artículo a la factura

public void agregarArticulo(Articulo articulo) {

articulos.add(articulo);

}

// Método para calcular el subtotal de la factura

public void calcularSubtotal() {

subtotal = 0.0;

for (Articulo articulo : articulos) {

subtotal += articulo.getPrecioUnitario() \* articulo.getCantidad();

}

}

// Método para calcular el IVA de la factura

public void calcularIVA(double porcentajeIVA) {

iva = subtotal \* (porcentajeIVA / 100);

}

// Método para calcular el total de la factura

public void calcularTotal() {

total = subtotal + iva;

}

// Otros métodos, getters y setters según sea necesario

public static void main(String[] args) {

// Ejemplo de uso de la clase Factura

Factura factura = new Factura(1, new Date(), "Cliente1");

// Agregar artículos

Articulo articulo1 = new Articulo("Producto1", 10.0, 2);

Articulo articulo2 = new Articulo("Producto2", 15.0, 3);

factura.agregarArticulo(articulo1);

factura.agregarArticulo(articulo2);

// Calcular subtotal, IVA y total

factura.calcularSubtotal();

factura.calcularIVA(18.0); // Supongamos un IVA del 18%

factura.calcularTotal();

// Mostrar resultados

System.out.println("Subtotal: " + factura.getSubtotal());

System.out.println("IVA: " + factura.getIVA());

System.out.println("Total: " + factura.getTotal());

}

}

class Articulo {

private String nombre;

private double precioUnitario;

private int cantidad;

public Articulo(String nombre, double precioUnitario, int cantidad) {

this.nombre = nombre;

this.precioUnitario = precioUnitario;

this.cantidad = cantidad;

}

public double getPrecioUnitario() {

return precioUnitario;

}

public int getCantidad() {

return cantidad;

}

}

-°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°-

* Escriba una aplicación que:

a) lea 20 números decimales ingresados por teclado por el usuario y los coloque en un ArrayList.

b) determine y muestre el mayor de los números en el arreglo

c) determine y muestre el menor de los números del arreglo calcule y muestre el rango (diferencia entre el mayor y el menor) de los elementos en el arreglo

-°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°-

import java.util.ArrayList;

import java.util.Scanner;

public class ProcesarNumeros {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

// Paso (a): Leer 20 números decimales e insertarlos en un ArrayList

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

for (int i = 0; i < 20; i++) {

System.out.print("Ingrese el número #" + (i + 1) + ": ");

double numero = scanner.nextDouble();

numeros.add(numero);

}

// Paso (b): Determinar y mostrar el mayor de los números

double maximo = numeros.get(0);

for (double numero : numeros) {

if (numero > maximo) {

maximo = numero;

}

}

System.out.println("El mayor número es: " + maximo);

// Paso (c): Determinar y mostrar el menor de los números

double minimo = numeros.get(0);

for (double numero : numeros) {

if (numero < minimo) {

minimo = numero;

}

}

System.out.println("El menor número es: " + minimo);

// Calcular y mostrar el rango

double rango = maximo - minimo;

System.out.println("El rango es: " + rango);

// Cerrar el objeto Scanner

scanner.close();

}

}

-°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°-

* Escriba una aplicación para almacenar en un ArrayList<Integer>

aleatoriamente 20 números enteros positivos pares del 1 al 100, de los

cuales se desea saber:

a) su promedio aritmético

b) cuántos de los números son iguales al promedio aritmético

c) cuántos de los números son mayores que el promedio aritmético

d) cuántos de los números son menores que el promedio aritmético

-°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°--°-°-

import java.util.ArrayList;

import java.util.Random;

public class ProcesarNumeros {

public static void main(String[] args) {

// Paso (a): Almacenar 20 números enteros positivos pares en un ArrayList

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

Random random = new Random();

for (int i = 0; i < 20; i++) {

// Generar números pares aleatorios del 1 al 100

int numero = random.nextInt(50) \* 2 + 1;

numeros.add(numero);

}

// Paso (b): Calcular el promedio aritmético

int suma = 0;

for (int numero : numeros) {

suma += numero;

}

double promedio = (double) suma / numeros.size();

// Paso (c): Contar cuántos números son iguales, mayores y menores que el promedio aritmético

int igualesAlPromedio = 0;

int mayoresAlPromedio = 0;

int menoresAlPromedio = 0;

for (int numero : numeros) {

if (numero == promedio) {

igualesAlPromedio++;

} else if (numero > promedio) {

mayoresAlPromedio++;

} else {

menoresAlPromedio++;

}

}

// Mostrar resultados

System.out.println("Números generados: " + numeros);

System.out.println("Promedio aritmético: " + promedio);

System.out.println("Cantidad de números iguales al promedio: " + igualesAlPromedio);

System.out.println("Cantidad de números mayores al promedio: " + mayoresAlPromedio);

System.out.println("Cantidad de números menores al promedio: " + menoresAlPromedio);

}

}