

Trabajo Práctico 2

Alumna: Diana Falla

Github : <https://github.com/Diana-Falla/Programacion2-TP2.git>

Desarrollo

1. Verificación de Año Bisiesto.

```
import java.util.Scanner;

public class TP2DianaFalla {

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.print("Ingrese un año: ");
        int año = scanner.nextInt();

        if ((año % 4 == 0 && año % 100 != 0) || (año % 400 == 0)) {
            System.out.println("El año " + año + " es bisiesto.");
        } else {
            System.out.println("El año " + año + " no es bisiesto.");
        }

        scanner.close();
    }
}
```

```
run:
Ingrese un año: 2024
El año 2024 es bisiesto.
BUILD SUCCESSFUL (total time: 4 seconds)
```

2.

```

import java.util.Scanner;

public class MayorDeTresNumeros {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        // Pedir los tres números al usuario
        System.out.print("Ingrese el primer número: ");
        int num1 = sc.nextInt();

        System.out.print("Ingrese el segundo número: ");
        int num2 = sc.nextInt();

        System.out.print("Ingrese el tercer número: ");
        int num3 = sc.nextInt();

        // Variable para almacenar el mayor
        int mayor;

        // Comparaciones
        if (num1 >= num2 && num1 >= num3) {
            mayor = num1;
        } else if (num2 >= num1 && num2 >= num3) {
            mayor = num2;
        } else {
            mayor = num3;
        }

        // Mostrar el resultado
        System.out.println("El mayor es: " + mayor);

        sc.close();
    }
}

```

```

run:
Ingrese el primer número: 8
Ingrese el segundo número: 12
Ingrese el tercer número: 5
El mayor es: 12
BUILD SUCCESSFUL (total time: 18 seconds)

```

3.

```
import java.util.Scanner;

public class ClasificacionEdad {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.print("Ingrese su edad: ");
        int edad = sc.nextInt();

        if (edad < 12) {
            System.out.println("Eres un Niño.");
        } else if (edad >= 12 && edad <= 17) {
            System.out.println("Eres un Adolescente.");
        } else if (edad >= 18 && edad <= 59) {
            System.out.println("Eres un Adulto.");
        } else { // edad >= 60
            System.out.println("Eres un Adulto mayor.");
        }

        sc.close();
    }
}
```

```
run:
Ingrese su edad: 25
Eres un Adulto.
BUILD SUCCESSFUL (total time: 9 seconds)
```

4.

```
import java.util.Scanner;
public class CalculadoraDescuento {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.print("Ingrese el precio del producto: ");
        double precio = sc.nextDouble();
        System.out.print("Ingrese la categoría del producto (A, B o C): ");
        char categoria = sc.next().toUpperCase().charAt(0);
        double descuento = 0;

        switch (categoria) {
            case 'A':
                descuento = 0.10;
                break;
            case 'B':
                descuento = 0.15;
                break;
            case 'C':
                descuento = 0.20;
                break;
            default:
                System.out.println("Categoría no válida. Use A, B o C.");
                sc.close();
                return;
        }

        double montoDescuento = precio * descuento;
        double precioFinal = precio - montoDescuento;

        System.out.println("Precio original: " + precio);
        System.out.println("Descuento aplicado: " + (int)(descuento * 100) + "%");
        System.out.println("Precio final: " + precioFinal);

        sc.close();
    }
}
```

```
run:
Ingrese el precio del producto: 1000
Ingrese la categoría del producto (A, B o C): B
Precio original: 1000.0
Descuento aplicado: 15%
Precio final: 850.0
BUILD SUCCESSFUL (total time: 18 seconds)
```

5. Suma de Números Pares (while).

```
import java.util.Scanner;

public class SumaNumerosPares {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        int numero;
        int suma = 0;

        System.out.print("Ingrese un número (0 para terminar): ");
        numero = sc.nextInt();

        while (numero != 0) {
            if (numero % 2 == 0) {
                suma += numero;
            }
            System.out.print("Ingrese un número (0 para terminar): ");
            numero = sc.nextInt();
        }

        System.out.println("La suma de los números pares es: " + suma);

        sc.close();
    }
}
```

```
run:
Ingrese un número (0 para terminar): 4
Ingrese un número (0 para terminar): 7
Ingrese un número (0 para terminar): 2
Ingrese un número (0 para terminar): 0
La suma de los números pares es: 6
BUILD SUCCESSFUL (total time: 20 seconds)
```

6. Contador de Positivos, Negativos y Ceros (for).

```
import java.util.Scanner;

public class ContadorNumeros {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        int positivos = 0;
        int negativos = 0;
        int ceros = 0;

        for (int i = 1; i <= 10; i++) {
            System.out.print("Ingrese el número " + i + ": ");
            int num = sc.nextInt();

            if (num > 0) {
                positivos++;
            } else if (num < 0) {
                negativos++;
            } else {
                ceros++;
            }
        }

        System.out.println("\nResultados:");
        System.out.println("Positivos: " + positivos);
        System.out.println("Negativos: " + negativos);
        System.out.println("Ceros: " + ceros);

        sc.close();
    }
}
```

```
run:
Ingrese el número 1: -5
Ingrese el número 2: 3
Ingrese el número 3: 0
Ingrese el número 4: -1
Ingrese el número 5: 6
Ingrese el número 6: 0
Ingrese el número 7: 9
Ingrese el número 8: -3
Ingrese el número 9: 4
Ingrese el número 10: -8

Resultados:
Positivos: 4
Negativos: 4
Ceros: 2
BUILD SUCCESSFUL (total time: 45 seconds)
```

7. Validación de Nota entre 0 y 10 (do-while).

```
import java.util.Scanner;

public class ValidarNota {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        int nota;

        while (true) {
            System.out.print("Ingrese una nota (0-10): ");
            nota = sc.nextInt();

            if (nota >= 0 && nota <= 10) {
                System.out.println("Nota guardada correctamente.");
                break;
            } else {
                System.out.println("Error: Nota inválida. Ingrese una nota entre 0 y 10.");
            }
        }

        sc.close();
    }
}
```

```

run:
Ingrese una nota (0-10): 15
Error: Nota inválida. Ingrese una nota entre 0 y 10.
Ingrese una nota (0-10): -2
Error: Nota inválida. Ingrese una nota entre 0 y 10.
Ingrese una nota (0-10): 8
Nota guardada correctamente.
BUILD SUCCESSFUL (total time: 15 seconds)

```

8. Cálculo del Precio Final con impuesto y descuento.

```

import java.util.Scanner;

public class PrecioFinalProducto {

    public static double calcularPrecioFinal(double precioBase, double impuesto, double descuento) {
        double impDecimal = impuesto / 100;
        double descDecimal = descuento / 100;
        double precioFinal = precioBase + (precioBase * impDecimal) - (precioBase * descDecimal);
        return precioFinal;
    }

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.print("Ingrese el precio base del producto: ");
        double precioBase = sc.nextDouble();

        System.out.print("Ingrese el impuesto en porcentaje (Ejemplo: 10 para 10%): ");
        double impuesto = sc.nextDouble();

        System.out.print("Ingrese el descuento en porcentaje (Ejemplo: 5 para 5%): ");
        double descuento = sc.nextDouble();

        double precioFinal = calcularPrecioFinal(precioBase, impuesto, descuento);

        System.out.println("El precio final del producto es: " + precioFinal);
        sc.close();
    }
}

```

```

run:
Ingrese el precio base del producto: 100
Ingrese el impuesto en porcentaje (Ejemplo: 10 para 10%): 10
Ingrese el descuento en porcentaje (Ejemplo: 5 para 5%): 5
El precio final del producto es: 105.0
BUILD SUCCESSFUL (total time: 7 seconds)

```


9. Composición de funciones para calcular costo de envío y total de compra.

```
import java.util.Scanner;

public class CalculoCompraEnvio {

    public static double calcularCostoEnvio(double peso, String zona) {
        double costoEnvio = 0;

        if (zona.equalsIgnoreCase("Nacional")) {
            costoEnvio = peso * 5;
        } else if (zona.equalsIgnoreCase("Internacional")) {
            costoEnvio = peso * 10;
        } else {
            System.out.println("Zona inválida. Debe ser 'Nacional' o 'Internacional'.");
        }

        return costoEnvio;
    }

    public static double calcularTotalCompra(double precioProducto, double costoEnvio) {
        return precioProducto + costoEnvio;
    }
}
```

```
public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);

    System.out.print("Ingrese el precio del producto: ");
    double precioProducto = sc.nextDouble();

    System.out.print("Ingrese el peso del paquete en kg: ");
    double peso = sc.nextDouble();
    sc.nextLine();

    System.out.print("Ingrese la zona de envío (Nacional/Internacional): ");
    String zona = sc.nextLine();

    double costoEnvio = calcularCostoEnvio(peso, zona);

    double total = calcularTotalCompra(precioProducto, costoEnvio);

    System.out.println("El costo de envío es: " + costoEnvio);
    System.out.println("El total a pagar es: " + total);

    sc.close();
}
```

```

run:
Ingrese el precio del producto: 50
Ingrese el peso del paquete en kg: 2
Ingrese la zona de envío (Nacional/Internacional): Nacional
El costo de envío es: 10.0
El total a pagar es: 60.0
BUILD SUCCESSFUL (total time: 19 seconds)

```

10. Actualización de stock a partir de venta y recepción de productos.

```

import java.util.Scanner;

public class GestionStock {

    public static int actualizarStock(int stockActual, int cantidadVendida, int cantidadRecibida) {
        int nuevoStock = stockActual - cantidadVendida + cantidadRecibida;
        return nuevoStock;
    }

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.print("Ingrese el stock actual del producto: ");
        int stockActual = sc.nextInt();

        System.out.print("Ingrese la cantidad vendida: ");
        int cantidadVendida = sc.nextInt();

        System.out.print("Ingrese la cantidad recibida: ");
        int cantidadRecibida = sc.nextInt();

        int nuevoStock = actualizarStock(stockActual, cantidadVendida, cantidadRecibida);

        System.out.println("El nuevo stock del producto es: " + nuevoStock);

        sc.close();
    }
}

```

```

run:
Ingrese el stock actual del producto: 50
Ingrese la cantidad vendida: 20
Ingrese la cantidad recibida: 30
El nuevo stock del producto es: 60
BUILD SUCCESSFUL (total time: 10 seconds)
|

```

11. Cálculo de descuento especial usando variable global.

```
import java.util.Scanner;

public class DescuentoEspecial {

    static final double DESCUENTO_ESPECIAL = 0.10; // 10%

    public static void calcularDescuentoEspecial(double precio) {

        double descuentoAplicado = precio * DESCUENTO_ESPECIAL;
        double precioFinal = precio - descuentoAplicado;

        System.out.println("El descuento especial aplicado es: " + descuentoAplicado);
        System.out.println("El precio final con descuento es: " + precioFinal);
    }

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.print("Ingrese el precio del producto: ");
        double precio = sc.nextDouble();

        calcularDescuentoEspecial(precio);

        sc.close();
    }
}
```

```
run:
Ingrese el precio del producto: 200
El descuento especial aplicado es: 20.0
El precio final con descuento es: 180.0
BUILD SUCCESSFUL (total time: 6 seconds)
|
```

Arrays y Recursividad:

12. Modificación de un array de precios y visualización de resultados.

```
public class ModificarPrecios {  
    public static void main(String[] args) {  
  
        double[] precios = {199.99, 299.5, 149.75, 399.0, 89.99};  
  
        System.out.println("Precios originales:");  
        for (double precio : precios) {  
            System.out.println("Precio: $" + precio);  
        }  
  
        precios[2] = 129.99;  
  
        System.out.println("Precios modificados:");  
        for (double precio : precios) {  
            System.out.println("Precio: $" + precio);  
        }  
    }  
}
```

```
run:  
Precios originales:  
Precio: $199.99  
Precio: $299.5  
Precio: $149.75  
Precio: $399.0  
Precio: $89.99  
Precios modificados:  
Precio: $199.99  
Precio: $299.5  
Precio: $129.99  
Precio: $399.0  
Precio: $89.99  
BUILD SUCCESSFUL (total time: 0 seconds)
```

13. Impresión recursiva de arrays antes y después de modificar un elemento.

```
public class PreciosRecursivos {  
  
    public static void main(String[] args) {  
  
        double[] precios = {199.99, 299.5, 149.75, 399.0, 89.99};  
  
        System.out.println("Precios originales:");  
        mostrarPreciosRecursivo(precios, 0);  
  
        precios[2] = 129.99;  
  
        System.out.println("Precios modificados:");  
        mostrarPreciosRecursivo(precios, 0);  
    }  
  
    public static void mostrarPreciosRecursivo(double[] array, int indice) {  
        if (indice < array.length) {  
            System.out.println("Precio: $" + array[indice]);  
            mostrarPreciosRecursivo(array, indice + 1); // llamada recursiva al siguiente índice  
        }  
    }  
}
```

```
run:  
Precios originales:  
Precio: $199.99  
Precio: $299.5  
Precio: $149.75  
Precio: $399.0  
Precio: $89.99  
Precios modificados:  
Precio: $199.99  
Precio: $299.5  
Precio: $129.99  
Precio: $399.0  
Precio: $89.99  
BUILD SUCCESSFUL (total time: 0 seconds)
```