

Trabajo Práctico N°1

Programación 2

OBJETIVO GENERAL

Aplicar los conocimientos adquiridos sobre la instalación y configuración del entorno de desarrollo, manipulación de datos, operadores matemáticos y depuración de código en Java, mediante ejercicios prácticos introductorios.

MARCO TEÓRICO

Concepto	Aplicación en el proyecto
Instalación y entorno	Almacenan el conjunto de países
Variables y tipos de datos	Representan los datos de cada país (nombre, población, superficie, etc.)
Entrada y salida	Separan las operaciones: carga, búsqueda, estadísticas, ordenamientos
Operadores aritméticos	Aplican filtros y validaciones según criterios
Caracteres especiales	Permite ordenar países por población, nombre, superficie, etc.
Expresiones e instrucciones	Permiten obtener indicadores clave del dataset
Tipos de datos y conversiones	Lectura del dataset desde un archivo CSV
Debugging y errores comunes	Identificación y corrección de errores de compilación.
Pruebas de escritorio	Análisis paso a paso de ejecución de código.

CASO PRÁCTICO El trabajo consiste en resolver una serie de ejercicios introductorios en Java que permitan:

- Configurar correctamente el entorno de desarrollo (Java JDK y NetBeans).
- Crear programas básicos que imprimen mensajes en consola.
- Declarar variables de distintos tipos y manipular sus valores.
- Leer datos ingresados por el usuario usando Scanner.
- Realizar operaciones aritméticas básicas.
- Aplicar caracteres de escape para dar formato a la salida.
- Analizar diferencias entre expresiones e instrucciones.
- Detectar y corregir errores simples en el código.
- Comprender el comportamiento del lenguaje mediante pruebas de escritorio.

1. Verificar que tienes instalado Java JDK y NetBeans

a) Confirma que tienes Java JDK instalado ejecutando el siguiente comando en la terminal: `java -version`

b) Abre NetBeans, crea un nuevo proyecto y configura el modo oscuro.

c) Toma una captura de pantalla del entorno configurado y agrégala a tu entrega.

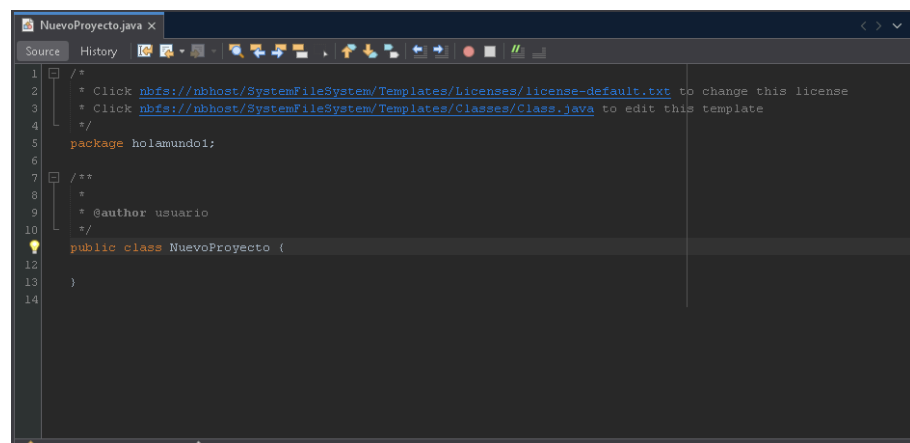
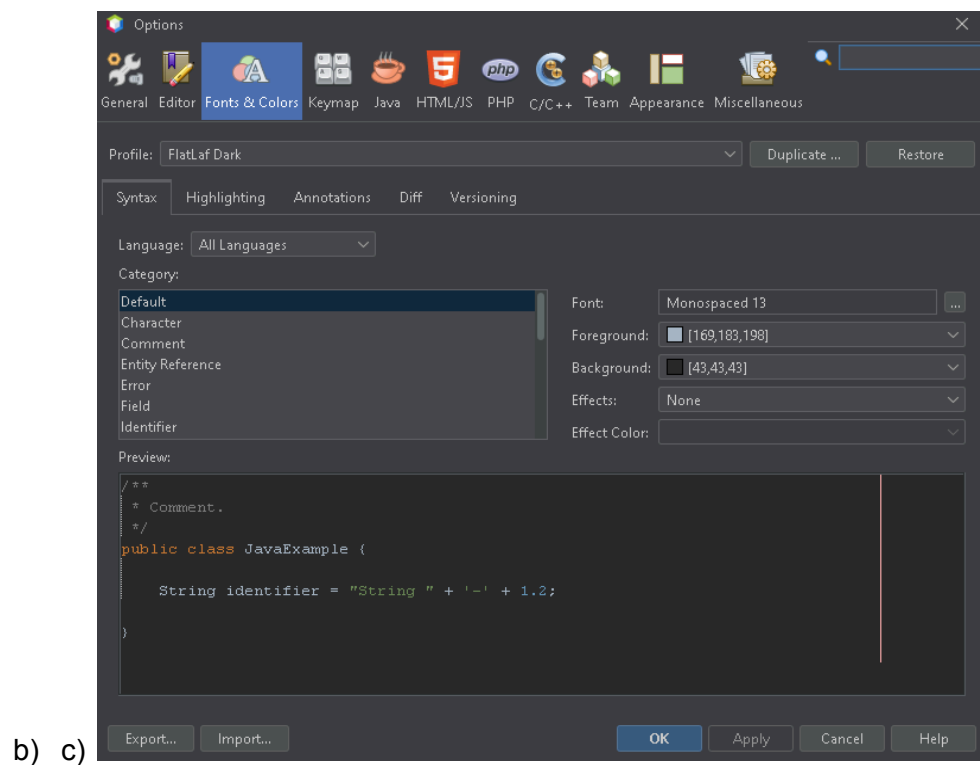
Respuesta:

a)

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Versión 10.0.18363.657]
(c) 2019 Microsoft Corporation. Todos los derechos reservados.

C:\Windows\system32>java -version
java version "24.0.2" 2025-07-15
Java(TM) SE Runtime Environment (build 24.0.2+12-54)
Java HotSpot(TM) 64-Bit Server VM (build 24.0.2+12-54, mixed mode, sharing)

C:\Windows\system32>
```



2. Escribir y ejecutar un programa básico en Java.

a) Creá una clase llamada HolaMundo.

- b) Escribe un programa que imprima el mensaje: ¡Hola, Java!
c) Ejecuta el programa en NetBeans y adjunta una captura del resultado en la consola.

Respuestas:

```

    */
    public class Holamundo1 {

        /**
         * @param args the command line arguments
         */
        public static void main(String[] args) {
            System.out.println("Hola Java");
        }

    }
}


```

Run - Holamundo1 (run) x

Deleting: D:\Programacion 2\Netbeans aplicaciones\deps-jar:
Updating property file: D:\Programacion 2\Netbeans compile:
run:
Hola Java
BUILD SUCCESSFUL (total time: 0 seconds)

a, b y c)

3. Crea un programa que declare las siguientes variables con valores asignados:
- a) String nombre
 - b) int edad
 - c) double altura
 - d) boolean estudiante Imprime los valores en pantalla usando System.out.println().

Respuestas:

```
15  */
16  public static void main(String[] args) {
17      // TODO code application logic here
18
19      String nombre = "Milagros Airalde";
20      int edad = 26;
21      double altura = 1.73;
22      boolean estudiante = true;
23
24      System.out.println("Mi nombre es: " + nombre);
25      System.out.println("Mi edad es: " + edad);
26      System.out.println("Mi altura es: " + altura);
27      System.out.println("Estudiante: " + estudiante);
28  }
29
30  }
31
```

practicando5.Practicando5 > main >

Output - trabajopractico (run) x

run:
Mi nombre es: Milagros Airalde
Mi edad es: 26
Mi altura es: 1.73
Estudiante: true
BUILD SUCCESSFUL (total time: 0 seconds)

4. Escribe un programa que solicite al usuario ingresar su nombre y edad, y luego los muestre en pantalla. Usa Scanner para capturar los datos.

Respuestas:

```
13  /**
14      * @param args the command line arguments
15      */
16  public static void main(String[] args) {
17      // TODO code application logic here
18      Scanner scanner = new Scanner (System.in);
19
20      System.out.println("Ingrese su nombre: ");
21      String nombre = scanner.nextLine();
22
23      System.out.println("Ingresa tu edad: ");
24      int edad = scanner.nextInt();
25
26      System.out.println("Nombre: " + nombre);
27      System.out.println("Edad: " + edad);
28
29      scanner.close();
30
31
32
33
34
```

ejercicio4tp.Ejercicio4tp > main > scanner >

Output - ejercicio4tp (run) x

```
Ingrese su nombre:
Mili
Ingresa tu edad:
26
Nombre: Mili
Edad: 26
BUILD SUCCESSFUL (total time: 12 seconds)
```

5. Escribe un programa que solicite dos números enteros y realice las siguientes operaciones:

- a) Suma
- b) Resta
- c) Multiplicación
- d) División Muestra los resultados en la consola.

Respuesta:

```

public class JavaApplication14 {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here
        Scanner scanner = new Scanner(System.in);

        System.out.println("Ingrese el primer numero entero positivo: ");
        int num1 = scanner.nextInt();

        System.out.println("Ingrese el segundo numero positivo: ");
        int num2 = scanner.nextInt();

        System.out.println("suma: " + (num1+num2));
        System.out.println("resta: " + (num1-num2));
        System.out.println("multiplicacion: " + (num1*num2));
        System.out.println("divison: " + (double) (num1/num2));
    }
}

```

```

run:
Ingrese el primer numero entero positivo:
20
Ingrese el segundo numero positivo:
10
suma: 30
resta: 10
multiplicacion: 200
divison: 2.0
BUILD SUCCESSFUL (total time: 6 seconds)

```

6. Escribe un programa que muestre el siguiente mensaje en consola:

Nombre: Juan Pérez

Edad: 30 años

Dirección: "Calle Falsa 123"

Usa caracteres de escape (\n, \") en System.out.println().

Respuesta:

```

public class JavaApplication17 {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here
        System.out.println("Nombre: Milagros Airalde \nEdad: 26\nDireccion: \"Calle falsa 123\"");
    }
}

```

javaapplication17JavaApplication17 > main >

run:

Nombre: Milagros Airalde

Edad: 26

Direccion: "Calle falsa 123"

BUILD SUCCESSFUL (total time: 0 seconds)

7. Analiza el siguiente código y responde:

¿Cuáles son expresiones y cuáles son instrucciones? Explica la diferencia en un breve párrafo.

```
int x = 10; // Línea 1
x = x + 5; // Línea 2
System.out.println(x); // Línea 3
```

Respuesta:

Expresiones: Son fragmentos de código que producen un valor. En este caso:

- 10 (Línea 1): Valor constante.
- $x + 5$ (Línea 2): Operación que produce un nuevo valor.

Instrucciones: Son líneas completas que realizan una acción. En este caso:

- `int x = 10;` (Línea 1): Declara y asigna un valor a x.
- `x = x + 5;` (Línea 2): Asigna un nuevo valor a x.
- `System.out.println(x);` (Línea 3): Imprime el valor de x.

Diferencia: Una **expresión** es una combinación de variables, operadores y valores que evalúa a un solo valor (por ejemplo, $x + 5$). Una **instrucción** es una unidad completa de ejecución que realiza una acción (como asignar un valor o imprimir). Las expresiones suelen formar parte de instrucciones.

8. Manejar conversiones de tipo y división en Java.

- a. Escribe un programa que divida dos números enteros ingresados por el usuario.
- b. Modifica el código para usar *double* en lugar de *int* y compara los resultados.

Respuestas:

```

public class JavaApplication18 {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here

        Scanner scanner = new Scanner(System.in);

        System.out.println("Ingresa el primer numero entero: ");
        int num1 = scanner.nextInt();

        System.out.println("Ingresa el segundo numero entero: ");
        int num2 = scanner.nextInt();

        if (num2 != 0) {
            System.out.println("Resultado (entero): " + (num1 / num2));
        } else {
            System.out.println("Error: No se puede dividir por cero");
        }

        scanner.close();
    }
}

```

a)

```

public class JavaApplication19 {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here
        Scanner scanner = new Scanner(System.in);

        System.out.println("Ingresa el primer numero: ");
        double num1 = scanner.nextDouble();

        System.out.println("Ingresa el segundo numero: ");
        double num2 = scanner.nextDouble();

        if (num2 != 0) {
            System.out.println("Resultado (double): " + (num1 / num2));
        } else {
            System.out.println("Error: No se puede dividir por cero");
        }

        scanner.close();
    }
}

```

b)

Output - JavaApplication19 (run) ×

```

run:
Ingresa el primer numero:
50
Ingresa el segundo numero:
20
Resultado (double): 2.5
BUILD SUCCESSFUL (total time: 5 seconds)

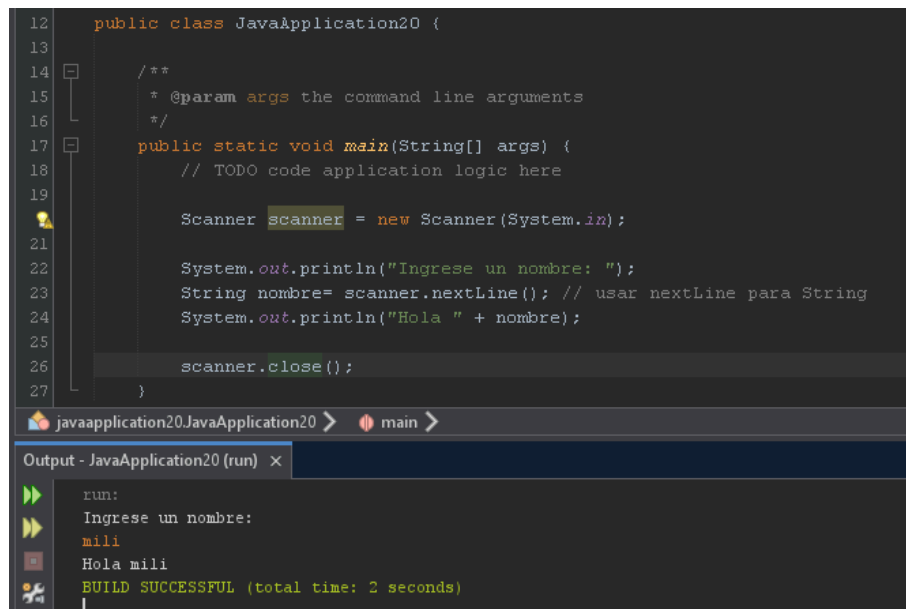
```


9. Corrige el siguiente código para que funcione correctamente.
Explica qué error tenía y cómo lo solucionaste.

```
import java.util.Scanner;
public class ErrorEjemplo {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Ingresa tu nombre: ");
        String nombre = scanner.nextInt(); // ERROR
        System.out.println("Hola, " + nombre);
    }
}
```

Respuestas:

```
import java.util.Scanner;
public class ErrorEjemplo {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Ingresa tu nombre: ");
        String nombre = scanner.nextLine(); // usar nextLine para String
        System.out.println("Hola, " + nombre);
        scanner.close()
    }
}
```



The screenshot shows an IDE window with a Java file named `JavaApplication20.java`. The code is as follows:

```
12 public class JavaApplication20 {
13
14     /**
15      * @param args the command line arguments
16      */
17     public static void main(String[] args) {
18         // TODO code application logic here
19
20         Scanner scanner = new Scanner(System.in);
21
22         System.out.println("Ingrese un nombre: ");
23         String nombre= scanner.nextLine(); // usar nextLine para String
24         System.out.println("Hola " + nombre);
25
26         scanner.close();
27     }
28 }
```

Below the code editor, the 'Output' window shows the execution results:

```
run:
Ingrese un nombre:
mili
Hola mili
BUILD SUCCESSFUL (total time: 2 seconds)
```

El método `nextInt()` es para leer enteros, no cadenas. Usar `nextLine()` es apropiado para capturar texto como un nombre.

El error de tipo (asignar un `int` a un `String`) es un error de compilación, y el uso incorrecto de `nextInt()` para un nombre también podría causar una excepción en tiempo de ejecución si el usuario ingresa texto.

10. Completa la tabla de prueba de escritorio para el siguiente código.
¿Cuál es el valor de resultado y por qué?

```
public class PruebaEscritorio {
    public static void main(String[] args) {
        int a = 5;
        int b = 2;
        int resultado = a / b;
        System.out.println("Resultado: " + resultado); }
    }
```

Respuesta:

Tabla de prueba.

Paso	línea de código	a	b	resultado	salida
1	int a = 5;	5	-	-	-
2	int b = 2;	5	2	-	-
3	int resultado = a / b;	5	2	2	-
4	System.out.println("Resultado: " + resultado);	5	2	2	2

El valor de resultado es 2.

Por que la división a / b (es decir, $5 / 2$) se realiza entre dos enteros (int). En Java, la división de enteros trunca la parte decimal, por lo que $5 / 2 = 2$ (el cociente entero, ignorando el residuo 1).