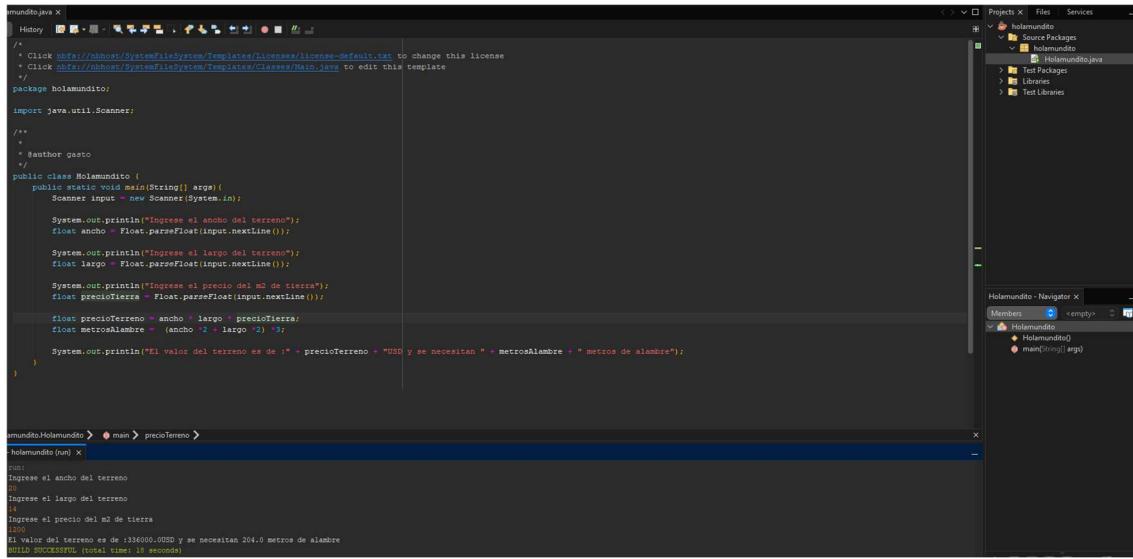


1. Verificar que tienes instalado Java JDK y NetBeans

- Confirma que tienes Java JDK instalado ejecutando el siguiente comando en la terminal: `java -version`
- Abre NetBeans, crea un nuevo proyecto y configura el modo oscuro.
- Toma una captura de pantalla del entorno configurado y agrégala a tu entrega.



```
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Main.java to edit this template
 */
package holamundo;

import java.util.Scanner;

/**
 * Author: gasto
 */
public class Holamundo {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);

        System.out.println("Ingrese el ancho del terreno");
        float ancho = Float.parseFloat(input.nextLine());

        System.out.println("Ingrese el largo del terreno");
        float largo = Float.parseFloat(input.nextLine());

        System.out.println("Ingrese el precio del m\u00b2 de tierra");
        float precioTierra = Float.parseFloat(input.nextLine());

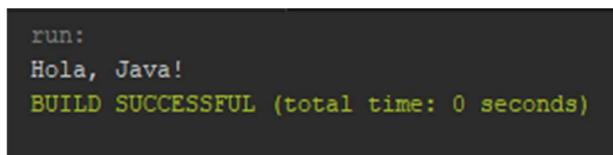
        float precioTerreno = ancho * largo * precioTierra;
        float metrosAlambre = (ancho + largo) * 2;

        System.out.println("El valor del terreno es de :" + precioTerreno + "USD y se necesitan " + metrosAlambre + " metros de alambre");
    }
}
```

```
amundito.Holamundo > 0 main > precioTerreno >
holamundo (run) X
run:
Ingrese el ancho del terreno
Ingrese el largo del terreno
10
Ingrese el precio del m\u00b2 de tierra
100000
El valor del terreno es de :1000000.0USD y se necesitan 204.0 metros de alambre
BUILD SUCCESSFUL (total time: 18 seconds)
```

2. Escribir y ejecutar un programa b\u00e1sico en Java.

- Cre\u00e1 una clase llamada HolaMundo.
- Escribe un programa que imprima el mensaje: ¡Hola, Java!
- Ejecuta el programa en NetBeans y adjunta una captura del resultado en la consola.



```
run:
Hola, Java!
BUILD SUCCESSFUL (total time: 0 seconds)
```

3. Crea un programa que declare las siguientes variables con valores asignados:

- String nombre
- int edad
- double altura
- boolean estudiante Imprime los valores en pantalla usando `System.out.println()`.

```
public class Holamundito {
    public static void main(String[] args){
        String nombre = "Gaston";
        int edad = 24;
        double altura = 1.84;
        boolean estudiante = true;

        System.out.println("Mi nombre es " + nombre + ", tengo " + edad + " años, mido " + altura + "mts. y estudiante = " + estudiante);
    }
}

Holamundito > main >
- holamundito (run) ×
run:
Mi nombre es Gaston, tengo 24 años, mido 1.84mts. y 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.

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

        System.out.println("Ingresa tu nombre:");
        String nombre = input.nextLine();
        System.out.println("Ingresa tu edad:");
        int edad = input.nextInt();

        System.out.println("Nombre: " + nombre + ".\nEdad: " + edad + ".");
    }
}
```

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.

```
System.out.println("Ingresa un numero:");
int n1 = input.nextInt();
System.out.println("Ingresa otro numero:");
int n2 = input.nextInt();

int suma = n1+n2;
int resta = n1-n2;
int prod = n1*n2;
double div = 1;
//No hay validacion para ingreso de 0 en la variable n2... seria cuestion de hacer while y listo.
if (n2 != 0){
    div = 1.0*n1/n2;
}
else{
    System.out.println("No se puede dividir por 0");
    return;
}

System.out.println("Suma: "+suma+"\nResta: "+resta+"\nMulti: "+prod+"\nDivision: "+div);
}
```

\amundito.Holamundito > main > if (n2 != 0) >

- holamundito (run) ×

run:

Ingresa un numero:
58
Ingresa otro numero:
11
Suma: 69
Resta: 47
Multi: 638
Division: 5.2727272727272725
BUILD SUCCESSFUL (total time: 2 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().

```
System.out.println("Nombre: Juan Perez\nEdad: 30 anios\nDireccion: \"Callecita llll\"");
```

Nombre: Juan Perez
Edad: 30 anios
Direccion: "Callecita llll"

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 Instrucción.

x = x + 5; // Línea 2 Instrucción.

System.out.println(x); // Línea 3 Instrucción.

Las expresiones están contenidas en las instrucciones, la diferencia es que una instrucción lleva a cabo una acción o proceso en el programa mientras que las expresiones definen la lógica de los pequeños engranajes de la instrucción, que pueden ser una o muchas dependiendo de la complejidad de la línea o bloque de código.

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.

R. Ya hice esto en un punto anterior 😊

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.nextLine();

        System.out.println("Hola, " + nombre);

    }

}
```

R. El objeto “scanner” esta usando el método nextInt para un String, por eso no compila. Entonces usamos nextLine(); que corresponde con el tipo de dato String.

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);  
    }  
}
```

R. La variable “resultado”, dado los tipos de datos usados en la función main, es 4. Esto sucede porque al usar int's en una división, Java devuelve valores enteros, perdiendo el resto por el camino. De hecho si usamos % como operador, obtendremos el 1 perdido que + 4 da los 5 del dividendo en cuestión. Hay varias soluciones posibles, de las cuales mi favorita hasta el momento es multiplicar cualquiera de los dos números enteros por 1.0 para que se transforme en double, ya que al, cualquiera de los valores, ser double, el resultado de la operación también lo será. Dejando el 1.0 siempre a la izquierda para no afectar a la legibilidad de la operación, por muy sencilla que sea.

Es muy interesante como Java interpreta todo esto y como funciona el Casting explícito y automático en el lenguaje.