TECNOLÓGICO NACIONAL DE MÉXICO

INSTITUTO TECNOLÓGICO DE TLAXIACO

**Arquitectura de computadoras**

**Presenta:**

Osorio Ramirez Marlene Maricela 22620269

**Producto**:

Conversor de Números

**Carrera**:

Ingeniería en Sistemas Computacionales

**Profesor**:

Osorio Salinas Edward

**Grupo**:

5BS

Tlaxiaco, Oax. Septiembre 2024.

***“Educación, Ciencia y Tecnología, Progreso día con día” ®***

**1. Introducción**  
  
La aplicación "Conversor de Números" desarrollada en Java tiene como objetivo convertir números entre diferentes bases numéricas: binario, decimal, octal y hexadecimal. La aplicación utiliza la consola para interactuar con el usuario, permitiendo la conversión de un número ingresado en una base específica a otro formato.  
  
**Objetivos del Proyecto**  
  
- Proveer una herramienta para convertir números entre bases numéricas comunes (binario, decimal, octal, hexadecimal).  
- Facilitar la comprensión de las bases numéricas y la conversión entre ellas.  
- Desarrollar habilidades de programación en Java, trabajando con entradas del usuario, manejo de excepciones y conversiones entre bases numéricas.  
  
Requisitos  
  
- Lenguaje de Programación: Java  
- Entorno de Desarrollo: NetBeans (o cualquier otro IDE compatible con Java)  
- Entrada: Número en formato de base específica (binario, octal, decimal, hexadecimal) y tipo de conversión deseada.  
- Salida: Número convertido al formato deseado.  
  
Descripción del Funcionamiento  
  
1. Entrada del Usuario:  
   - La aplicación solicita al usuario el tipo de número que se ingresará (binario, octal, decimal o hexadecimal).  
   - Luego solicita el número en el formato especificado.  
   - Finalmente, el usuario indica el formato al que desea convertir el número.  
  
2. Conversión:  
   - El programa convierte el número ingresado a su equivalente en decimal, utilizando el formato original.  
   - Luego, convierte el número decimal al formato deseado.  
  
3. Salida:  
   - Muestra el número convertido en la base especificada.  
  
5. Estructura del Código  
  
- Clase Principal: `NumberConverter`  
  - Método `main`: Controla la interacción con el usuario. Solicita entradas y muestra el resultado de la conversión.  
  - Método `convertNumber`: Realiza la conversión del número desde la base de origen al formato decimal y luego a la base de destino.  
  
java  
import [java.util.Scanner](http://java.util.Scanner);  
  
public class NumberConverter {  
  
    public static void main(String[] args) {  
        Scanner scanner = new Scanner([System.in](http://System.in));  
  
        [System.out.println](http://System.out.println)("Ingrese el tipo de número que desea convertir (decimal, octal, binario, hexadecimal):");  
        String inputType = [scanner.nextLine](http://scanner.nextLine)().toLowerCase();  
  
        [System.out.println](http://System.out.println)("Ingrese el número a convertir:");  
        String number = [scanner.nextLine](http://scanner.nextLine)();  
  
        [System.out.println](http://System.out.println)("Ingrese el tipo de conversión deseada (decimal, octal, binario, hexadecimal):");  
        String outputType = [scanner.nextLine](http://scanner.nextLine)().toLowerCase();  
  
        String result = convertNumber(number, inputType, outputType);  
        [System.out.println](http://System.out.println)("El resultado de la conversión es: " + result);  
    }  
  
    public static String convertNumber(String number, String inputType, String outputType) {  
        int decimalNumber;  
  
        // Convertir el número ingresado a decimal  
        switch (inputType) {  
            case "decimal":  
                decimalNumber = [Integer.parseInt](http://Integer.parseInt)(number);  
                break;  
            case "binary":  
                decimalNumber = [Integer.parseInt](http://Integer.parseInt)(number, 2);  
                break;  
            case "octal":  
                decimalNumber = [Integer.parseInt](http://Integer.parseInt)(number, 8);  
                break;  
            case "hexadecimal":  
                decimalNumber = [Integer.parseInt](http://Integer.parseInt)(number, 16);  
                break;  
            default:  
                throw new IllegalArgumentException("Tipo de número no reconocido.");  
        }  
  
        // Convertir el número decimal a la base deseada  
        switch (outputType) {  
            case "decimal":  
                return [Integer.toString](http://Integer.toString)(decimalNumber);  
            case "binary":  
                return [Integer.toBinaryString](http://Integer.toBinaryString)(decimalNumber);  
            case "octal":  
                return [Integer.toOctalString](http://Integer.toOctalString)(decimalNumber);  
            case "hexadecimal":  
                return [Integer.toHexString](http://Integer.toHexString)(decimalNumber).toUpperCase();  
            default:  
                throw new IllegalArgumentException("Tipo de conversión no reconocido.");  
        }  
    }  
}  
6. Pruebas  
  
- Prueba de Conversión Binario a Decimal: Entrada: `[1010](tel:1010)`, Salida Esperada: `10`  
- Prueba de Conversión Decimal a Hexadecimal: Entrada: `[255](tel:255)`, Salida Esperada: `FF`  
- Prueba de Conversión Octal a Binario: Entrada: `17`, Salida Esperada: `[1111](tel:1111)`  
- Prueba de Conversión Hexadecimal a Octal: Entrada: `1F`, Salida Esperada: `37`  
 7. Posibles Mejoras  
  
Interfaz Gráfica de Usuario (GUI): Implementar una GUI usando Swing o JavaFX para una experiencia de usuario más amigable.  
- \*\*Validación de Entrada\*\*: Agregar validaciones más robustas para manejar errores en la entrada del usuario.  
- \*\*Soporte para Números Negativos\*\*: Ampliar el código para manejar números negativos en las conversiones.

**8.CONCLUSIÓN**  
  
La aplicación desarrollada y nombrada "Conversor de Números" proporciona una solución simple y efectiva para convertir entre diferentes sistemas numéricos. Con una implementación básica en Java, permite al usuario realizar conversiones entre bases comunes y sirve como una buena práctica en programación y manejo de datos numéricos.