

DOCUMENTACION BIBLIOTECA OPERACIONESMATH

OperacionesMath es una biblioteca Java desarrollada para realizar operaciones matemáticas básicas y avanzadas. Es reutilizable y diseñada para ser integrada fácilmente en otros proyectos

Miguel Antonio
Amaya
Hernández

Introducción

La biblioteca OperacionesMath fue desarrollada por Miguel Antonio Amaya Hernández como parte de la Prueba Técnica para Desarrollador Java Web, la cual tuvo como objetivo evaluar las habilidades técnicas y competencias en el desarrollo de aplicaciones web y herramientas complementarias con Java. Esta prueba técnica fue realizada bajo las siguientes condiciones:

Propósito de la Biblioteca OperacionesMath

La biblioteca OperacionesMath es una solución modular y altamente reutilizable que permite realizar operaciones matemáticas básicas (suma, resta, multiplicación y división) y avanzadas (factorial, raíz cuadrada, potenciación y logaritmos). Está diseñada para:

- **Fácil integración** en cualquier proyecto basado en Java.
- **Compatibilidad con Maven** para facilitar la gestión de dependencias.
- Ser **portable** gracias al empaquetado en un archivo .jar, permitiendo su distribución en diferentes proyectos.

Además de su diseño funcional, la biblioteca refleja un enfoque en la calidad del código y modularidad, facilitando futuras extensiones y el mantenimiento a largo plazo. Esta herramienta es ideal para desarrolladores que necesiten cálculos matemáticos en sus proyectos sin tener que escribir las funciones desde cero.

Características

- **Operaciones Básicas:**
 - ✓ Suma, resta, multiplicación y división.
- **Operaciones Avanzadas:**
 - ✓ Raíz cuadrada.
 - ✓ Potenciación.
 - ✓ Factorial.
 - ✓ Logaritmos (base 10 y natural).
- **Diseño Modular:**
 - ✓ Clases separadas para operaciones básicas y avanzadas.
- **Extensibilidad:**
 - ✓ Puede ampliarse para incluir funciones matemáticas adicionales.

Estructura del Proyecto

Paquetes y Clases

1. Paquete operaciones:

- ✓ **MathOperaciones:** Contiene métodos para operaciones básicas.
- ✓ **AdvancedMathOperations:** Contiene métodos para operaciones avanzadas.
- ✓ **Main:** Clase principal de prueba.

2. Contenido del archivo JAR:

operaciones/

└─ MathOperaciones.class

└─ AdvancedMathOperations.class

└─ Main.class

Configuración del Proyecto

1. Uso del archivo JAR

1. Descarga el archivo operaciones-math-1.0-SNAPSHOT.jar.
2. Colócalo en una carpeta llamada libs dentro de tu proyecto.
3. Configura tu IDE:
 - **Eclipse:** Haz clic derecho en tu proyecto > Build Path > Add External Archives y selecciona el archivo .jar.
 - **IntelliJ IDEA:** Ve a File > Project Structure > Libraries > Add y selecciona el archivo .jar.

2. Integración con Maven

Si deseas integrarlo en un proyecto Maven, sigue estos pasos:

1. Instala el .jar en tu repositorio local de Maven:

```
mvn install:install-file -Dfile=OperacionesMath-1.0-SNAPSHOT.jar -  
DgroupId=com.operaciones -DartifactId=operaciones-math -  
Dversion=1.0-SNAPSHOT -Dpackaging=jar
```

2. Añade la siguiente dependencia en tu archivo pom.xml:

```
<dependency>
  <groupId>com.operaciones</groupId>
  <artifactId>operaciones-math</artifactId>
  <version>1.0-SNAPSHOT</version>
</dependency>
```

3. Verifica que Maven reconozca la biblioteca con:

```
mvn clean install
```

Guía de Uso

1. Importar las Clases

Asegúrate de importar las clases en tu proyecto:

```
import operaciones.MathOperaciones;
```

```
import operaciones.AdvancedMathOperations;
```

2. Uso de los Métodos

Una vez importadas, puedes usar los métodos de las clases:

- **Operaciones Básicas:**
 - `int suma = MathOperaciones.sumar(10, 5);`
 - `int resta = MathOperaciones.restar(10, 5);`
 - `double division = MathOperaciones.dividir(20, 4);`
- **Operaciones Avanzadas:**
 - `double raiz = AdvancedMathOperations.raizCuadrada(16);`
 - `long factorial = AdvancedMathOperations.factorial(5);`
 - `double log10 = AdvancedMathOperations.logaritmoBase10(100);`

API de la Biblioteca

1. Métodos Básicos (MathOperaciones)

- **sumar(int a, int b)**
 - **Descripción:** Retorna la suma de dos números enteros.
 - **Uso:**
 - `int resultado = MathOperaciones.sumar(10, 5);`
- **restar(int a, int b)**
 - **Descripción:** Retorna la resta de dos números enteros.
- **multiplicar(int a, int b)**
 - **Descripción:** Retorna la multiplicación de dos números enteros.
- **dividir(double a, double b)**
 - **Descripción:** Retorna la división de dos números. Maneja divisiones por cero.

2. Métodos Avanzados (AdvancedMathOperations)

- **raizCuadrada(double numero)**
 - ✓ **Descripción:** Calcula la raíz cuadrada de un número.
 - ✓ **Uso:**
 - ✓ `double resultado = AdvancedMathOperations.raizCuadrada(25);`
- **potencia(double base, double exponente)**
 - ✓ **Descripción:** Calcula la potencia de un número.
- **factorial(int numero)**
 - ✓ **Descripción:** Calcula el factorial de un número.
- **logaritmoBase10(double numero)**
 - ✓ **Descripción:** Calcula el logaritmo base 10.
- **logaritmoNatural(double numero)**
 - ✓ **Descripción:** Calcula el logaritmo natural (base e).

Ejemplo de Implementación

Código de Ejemplo:

```
import operaciones.MathOperaciones;

import operaciones.AdvancedMathOperations;


public class Main {
    public static void main(String[] args) {
        // Operaciones Básicas

        System.out.println("Suma: " + MathOperaciones.sumar(5, 3));
        System.out.println("Resta: " + MathOperaciones.restar(10, 4));
        System.out.println("Multiplicación: " +
MathOperaciones.multiplicar(2, 3));
        System.out.println("División: " + MathOperaciones.dividir(10,
2));


        // Operaciones Avanzadas

        System.out.println("Raíz cuadrada: " +
AdvancedMathOperations.raizCuadrada(16));

        System.out.println("Potencia: " +
AdvancedMathOperations.potencia(2, 5));

        System.out.println("Factorial: " +
AdvancedMathOperations.factorial(5));

        System.out.println("Logaritmo base 10: " +
AdvancedMathOperations.logaritmoBase10(100));
    }
}
```

Salida Esperada:

Suma: 8

Resta: 6

Multipliación: 6

División: 5.0

Raíz cuadrada: 4.0

Potencia: 32.0

Factorial: 120

Logaritmo base 10: 2.0

Futuras Extensiones

- Añadir más operaciones matemáticas avanzadas (trigonometría, integrales).
- Optimizar la biblioteca para cálculos de alto rendimiento.
- Documentación en más idiomas y soporte multilingüe.