

# Documento de Pruebas para Java

Este documento está diseñado para proporcionar ejemplos básicos sobre cómo realizar pruebas en aplicación de consola en Java.

## Introducción a las pruebas en Java

Las pruebas en Java son esenciales para garantizar que el código funcione correctamente y evitar errores en producción. El testing ayuda a validar la lógica, detectar fallos y asegurar que las modificaciones futuras no rompan funcionalidades existentes. Ejemplo de prueba realizada en la raíz de un número negativo:

Prueba: raíz de un número negativo

Input: Radicando: -7, Índice: 4

Output: "No se puede obtener raíz de un número negativo"

Ingrese un radicando no negativo" (nuevo input permitido)

Input = 6, respuesta de la raíz: 1.5650845800732873

## Ejemplo de una clase Java sencilla

Veamos un ejemplo básico de una clase que queremos probar:

```
package app;

import java.util.Scanner;

import static java.lang.StrictMath.pow;

public class Operaciones {

    public double division(Numero a, Numero b) {
        if(b.getValor() == 0){
            System.out.println("No es posible dividir entre cero. Intentelo de nuevo");
            System.out.println("Ingrese un número diferente de cero");
            Scanner input = new Scanner(System.in);
            b.setValor(input.nextDouble());
        }
        return a.getValor() / b.getValor();
    }
}
```

```

public double radicacion(Numero a, Numero b) {
    Scanner input = new Scanner(System.in);
    if(a.getValor() == -1 && b.getValor() == 2){
        System.out.println("Numeros Complejos no estan incluidos.");
        System.out.println("No se puede usar Raiz Cuadrada de -1");
        System.out.println("Ingrese un radicando diferente de -1");
        a.setValor(input.nextDouble());
    }else if(a.getValor() < 0){
        System.out.println("No se puede obtener raiz de un numero
negativo");
        System.out.println("Ingrese un radicando no negativo");
        a.setValor(input.nextDouble());
    }
    return Math.pow( a.getValor(), 1.0/b.getValor() );
}
}

```

## Ejecutando las Pruebas

Las pruebas pueden ejecutarse desde un IDE como IntelliJ IDEA, Eclipse, o desde la línea de comandos si se usa Maven con el comando:

```
mvn test
```

Esto ejecutará todas las pruebas definidas en el proyecto y mostrará un resumen de resultados. En este caso solo se utilizara la consola de comandos y se registraran las respuestas en un archive .txt.

## Conclusión

Realizar pruebas en Java es fundamental para desarrollar software robusto y confiable. Empezar con pruebas simples y aplicar buenas prácticas permitirá mejorar la calidad del código y prevenir errores futuros.

Este documento brindó una introducción básica con ejemplos prácticos para comenzar a hacer pruebas en Java.