

# CALCULADORA SIMPLE

PÉREZ SELLERS, ALEJANDRO  
PROGRAMACIÓN 1º de daw

## 1º Explicación del código.

Se pide programar un calculadora simple, donde el usuario introduce el primer operando mediante teclado con un tipo de **variable double**, ante de eso nos da un mensaje de bienvenida, después nos muestra la cantidad de opciones que tiene nuestra calculadora y nos pide que elijamos una mediante teclado con un variable **tipo String**, como no me fio de que el usuaria introduzca la opción siempre en mayúscula he introducido un **Touppercase**, para que cambie la opción siempre a mayúscula, ya que mis valores en el switch están en ese formato.

```
package org.example;

import java.util.Scanner;

public class Calculadora_Simple {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        boolean repetir = false;
        do {
            try {
                double operando2 = 0;
                repetir = false;
                System.out.println("*** BIENVENIDO A LA CALCULADORA
RAPIDA ***");
                System.out.println("introduce operando: ");
                double operandol = sc.nextDouble();
                double numero = 0;
                System.out.println("-----");
                System.out.println("[+] -> sumar");
                System.out.println("[-] -> restar");
                System.out.println("[x] -> multiplicar");
                System.out.println("[/] -> dividir");
                System.out.println("[R] -> raiz cuadrada");
                System.out.println("-----");
                System.out.println("elige una opción");
                String opcion = sc.next();
                String opcion_mayusculas=opcion.toUpperCase();
```

Introduzco un condicional donde **if (!opcion.equals("R"))**, es decir sino seleccionamos la "R" nos pide el segundo operando, posteriormente nos pasa al bloque switch donde según la opción que hayamos elegido hará una operación u otra, aquí incluyo en la división un condicional **if** para controlar que no se pueda dividir entre 0 (**operando2 == 0**), también incluyo otro condicional **if** en la opción raíz cuadrada, donde **operando1 < 0** para evitar números negativos en raíz cuadrada. El programa está envuelto en el bucle **do-while** donde repite la pregunta en el caso de que introduzcas un valor inválido en el operando y en el segundo operando, estas preguntas las envuelve el **try** envuelve lo que puede fallar, como el tipo de entrada que le introduzcas, en caso de que introduzcas un valor inválido saltaría el **catch** con un mensaje y pasaría al **while** donde volvería a repetir el bloque **do**, esto se controla con booleanos, donde **repetir = false** si está bien el tipo de variable, si el tipo de variable es incorrecto cambia a **true**, salta el **catch** y el **while** controlado con **repetir = true** vuelve al bloque **do**. En el bloque switch si introduzco un valor que está fuera del **case** salta el **default** donde aparece un mensaje que muestra que el valor introducido es inválido.

```
if (!opcion.equals("R")) {
    System.out.println("introduce un segundo
operando");
    operando2 = sc.nextDouble();
}
switch (opcion_mayusculas) {
    case "+":
        double sumar = operando1 + operando2;
        System.out.println("el resultado de " +
operando1 + "+" + operando2 + "=" + sumar);
        break;
    case "-":
        double restar = operando1 - operando2;
        System.out.println("el resultado de " +
operando1 + "-" + operando2 + "=" + restar);
        break;
    case "x":
        double multiplicar = operando1 * operando2;
        System.out.println("el resultado de " +
operando1 + "x" + operando2 + "=" + multiplicar);
        break;
    case "/":
        if (operando2 == 0) {
            System.out.println("No se puede dividir un
numero entre cero");
        } else {
            double dividir = operando1 / operando2;
            System.out.println("el resultado de " +
operando1 + "/" + operando2 + "=" + dividir);
        }
        break;
    case "R":
        if (operando1 < 0) {
            System.out.println("no se puede realizar
la raíz cuadrada de un numero negativo");
        } else {
            double raiz = Math.sqrt(operando1);
            System.out.println("el resultado de la
raiz es: " + raiz);
        }
        break;
    default:
```

```

        System.out.println("la opcion no es
correcta");

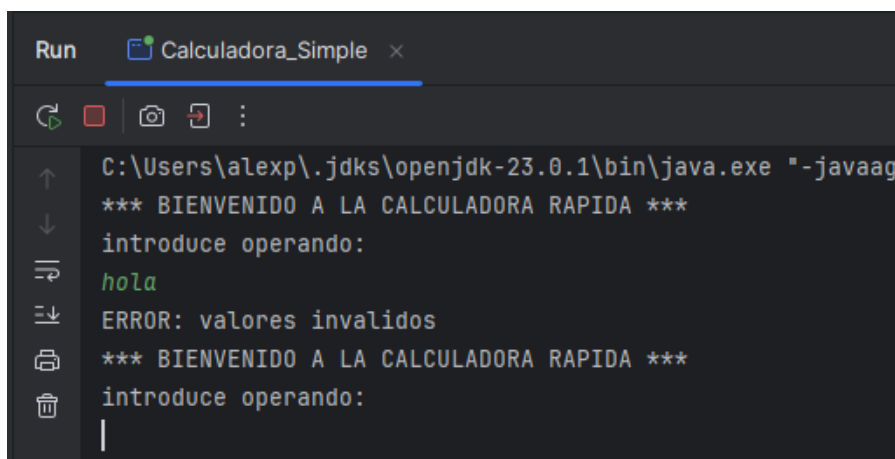
    }

    } catch (Exception err) {
        repetir = true;
        System.out.println("ERROR: valores invalidos");
    }
    sc.nextLine();
}while (repetir == true);
}
}

```

## 2. Pruebas de ejecución del programa

### 2.1 Introduzco una palabra en operando.

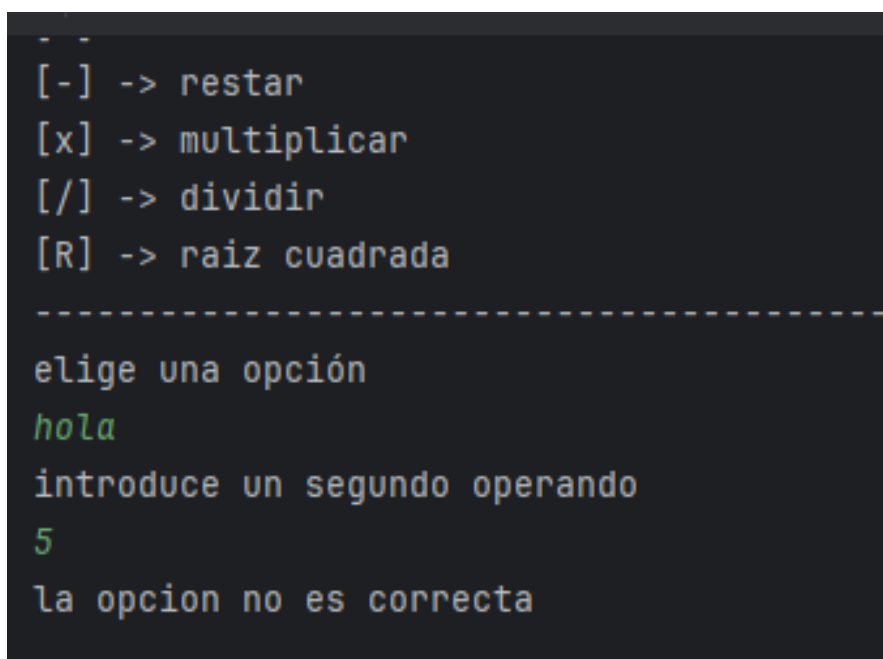


```

Run  Calculadora_Simple x
C:\Users\alexp\.jdk\openjdk-23.0.1\bin\java.exe "-javaag
*** BIENVENIDO A LA CALCULADORA RAPIDA ***
introduce operando:
hola
ERROR: valores invalidos
*** BIENVENIDO A LA CALCULADORA RAPIDA ***
introduce operando:
|

```

### 2.2 Introduzco una palabra en opción.

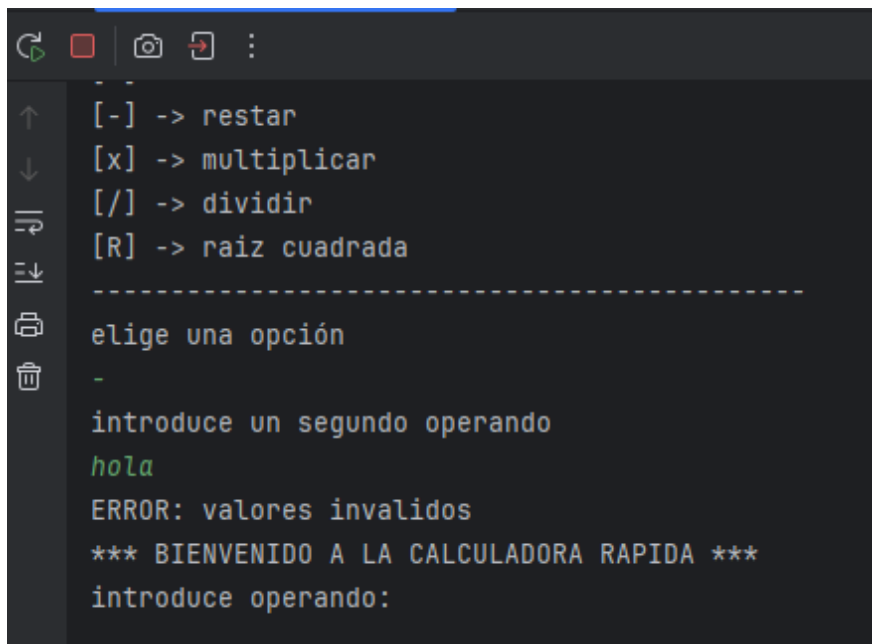


```

- -
[-] -> restar
[x] -> multiplicar
[/] -> dividir
[R] -> raiz cuadrada
-----
elige una opción
hola
introduce un segundo operando
5
la opcion no es correcta

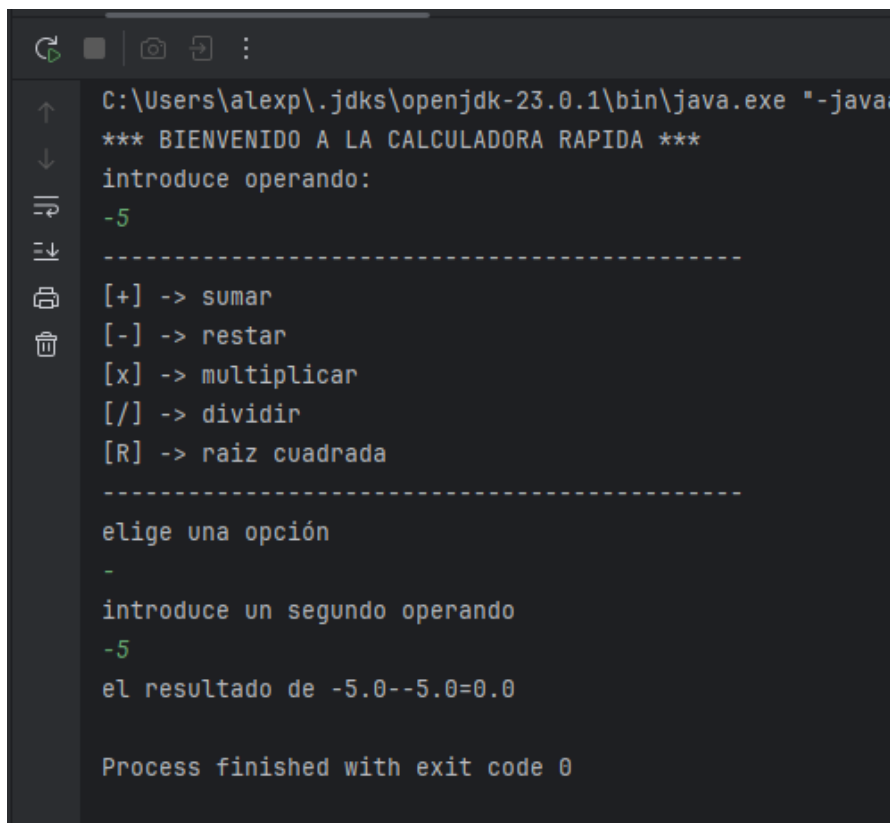
```

2.3 introduzco una palabra en operando.



```
[-] -> restar
[x] -> multiplicar
[/] -> dividir
[R] -> raiz cuadrada
-----
elige una opción
-
introduce un segundo operando
hola
ERROR: valores invalidos
*** BIENVENIDO A LA CALCULADORA RAPIDA ***
introduce operando:
```

2.4 Introduzco el primer operando y segundo operando en negativo.



```
C:\Users\alexp\.jdk\openjdk-23.0.1\bin\java.exe "-javaa
*** BIENVENIDO A LA CALCULADORA RAPIDA ***
introduce operando:
-5
-----
[+] -> sumar
[-] -> restar
[x] -> multiplicar
[/] -> dividir
[R] -> raiz cuadrada
-----
elige una opción
-
introduce un segundo operando
-5
el resultado de -5.0--5.0=0.0

Process finished with exit code 0
```

2.5 Introduzco uno de los operandos en negativo.

```
C:\Users\alexp\.jdk\openjdk-23.0.1\bin\java.exe "
*** BIENVENIDO A LA CALCULADORA RAPIDA ***
introduce operando:
-5
-----
[+] -> sumar
[-] -> restar
[x] -> multiplicar
[/] -> dividir
[R] -> raiz cuadrada
-----
elige una opción
x
introduce un segundo operando
5
el resultado de -5.0x5.0=-25.0

Process finished with exit code 0
```

2.6 Introduzco en el divisor un cero

```
C:\Users\alexp\.jdk\openjdk-23.0.1\bin\java.exe "
*** BIENVENIDO A LA CALCULADORA RAPIDA ***
introduce operando:
5
-----
[+] -> sumar
[-] -> restar
[x] -> multiplicar
[/] -> dividir
[R] -> raiz cuadrada
-----
elige una opción
/
introduce un segundo operando
0
No se puede dividir un numero entre cero

Process finished with exit code 0
```

## 2.7 Introduzco número negativo en raíz cuadrada.

```
C:\Users\alexp\.jdk\openjdk-23.0.1\bin\java.exe "-javaagent:C:\Users\alexp\.jdk\openjdk-23.0.1\bin\javaagent.jar"
*** BIENVENIDO A LA CALCULADORA RAPIDA ***
introduce operando:
-5
-----
[+] -> sumar
[-] -> restar
[x] -> multiplicar
[/] -> dividir
[R] -> raiz cuadrada
-----
elige una opción
r
no se puede realizar la raiz cuadrada de un numero negativo

Process finished with exit code 0
|
```

## 2.8 Introduzco cambio de signo en el formato de entrada

```
C:\Users\alexp\.jdk\openjdk-23.0.1\bin\java.exe "-javaagent:C:\Users\alexp\.jdk\openjdk-23.0.1\bin\javaagent.jar"
*** BIENVENIDO A LA CALCULADORA RAPIDA ***
introduce operando:
10
-----
[+] -> sumar
[-] -> restar
[x] -> multiplicar
[/] -> dividir
[R] -> raiz cuadrada
-----
elige una opción
y
introduce un segundo operando
5
la opcion no es correcta

Process finished with exit code 0
|
```