**package** etsjavadocs;

/\*\*

\*Esta clase emula una calculadora, pudiendo

\*realizar sumas, restas, multiplicaciones y

\*divisiones entre dos operandos.

\* **@author** Daniel Castro Cruz

\* **@version** 1.0

\*/

**public** **class** CalculadoraJavadocs {

**private** **int** operando1;

**private** **int** operando2;

**private** **int** resultado;

/\*\*

\* Metodo constructor parametrizado

\* **@param** operando1 Valor del operando1

\* **@param** operando2 Valor del operando2

\* **@param** resultado Resultado de la operacion

\*/

**public** CalculadoraJavadocs(**int** operando1, **int** operando2, **int** resultado) {

**this**.operando1 = operando1;

**this**.operando2 = operando2;

**this**.resultado = resultado;

}

/\*\*

\* Metodo para realizar la suma

\* entre los dos operandos

\* **@param** operando1 Int que se le asignara al operando1

\* **@param** operando2 Int que se le asignara al operando2

\* **@return** Regresa el resultado de la suma entre

\* operando1 y operando2

\*/

**public** **int** suma(**int** operando1, **int** operando2) {

resultado = operando1 + operando2;

**return** resultado;

}

/\*\*

\* Metodo para realizar la resta

\* entre los dos operandos

\* **@param** operando1 Int que se le asignara al operando1

\* **@param** operando2 Int que se le asignara al operando2

\* **@return** Regresa el resultado de la resta entre

\* operando1 y operando2

\*/

**public** **int** resta(**int** operando1, **int** operando2) {

resultado = operando1 - operando2;

**return** resultado;

}

/\*\*

\* Metodo para realizar la multiplicacion

\* entre los dos operandos

\* **@param** operando1 Int que se le asignara al operando1

\* **@param** operando2 Int que se le asignara al operando2

\* **@return** Regresa el resultado de la multiplicacion entre

\* operando1 y operando2

\*/

**public** **int** multiplicacion(**int** operando1, **int** operando2) {

resultado = operando1 \* operando2;

**return** resultado;

}

/\*\*

\* Metodo para realizar la division

\* entre los dos operandos

\* **@param** operando1 Int que se le asignara al operando1

\* **@param** operando2 Int que se le asignara al operando2

\* **@return** Regresa el resultado de la division entre

\* operando1 y operando2

\*/

**public** **int** division(**int** operando1, **int** operando2) {

resultado = operando1 / operando2;

**return** resultado;

}

/\*\*

\* Metodo para establecer el valor del operando1

\* **@param** operando1 Int que se le asignará al operando1

\*/

**public** **void** setOperando1(**int** operando1) {

**this**.operando1 = operando1;

}

/\*\*

\* Metodo para retornar el valor del operando 1

\* **@return** retorna el valor del operando 1

\*/

**public** **int** getOperando1() {

**return** operando1;

}

/\*\*

\* Metodo para establecer el valor del operando2

\* **@param** operando2 Int que se le asignara al operando2

\*/

**public** **void** setOperando2(**int** operando2) {

**this**.operando2 = operando2;

}

/\*\*

\* Metodo para retornar el valor del operando 2

\* **@return** retorna el valor del operando 2

\*/

**public** **int** getOperando2() {

**return** operando2;

}

/\*\*

\* Metodo para establecer el valor del resultado

\* **@param** resultado Int que se le asignara al resultado

\*/

**public** **void** setResultado(**int** resultado) {

**this**.resultado = resultado;

}

/\*\*

\* Metodo para retornar el valor del resultado

\* **@return** retorna el valor del resultado

\*/

**public** **int** getResultado() {

**return** resultado;

}

}