Detalles técnicos de la realización de la práctica

**Entorno de ejecución:**

**Para ejecutar el programa es requisito IMPRESCINDIBLE importar el proyecto en Eclipse, produciendo en caso contrario un NullPointer Exception. La clase principal a ejecutar es <<Interfaz>>.**

**Cuando se creen nuevas operaciones se recomienda editar el main dentro de la nueva clase para elegir los parámetros de ejecución, que por defecto serán <1>.**

# Entradas

El lenguaje utilizado para la realización de la práctica ha sido java. Para introducir las instrucciones al programa se ha utilizado el lenguaje natural. Las instrucciones que se introducen siguen una sintaxis concreta.

Las funciones predefinidas se detallan a continuación, aunque como ejemplo se añaden todas las funciones que se espera que el programa pueda crear.

**Suma** parametroA:int segundoParametro:int

**Negación** primerParametro:int

**Repeticion** primerParametro:int segundoParametro:Operaciones

Llamada de funciones: para reutilizar las funciones que ya se tienen, hay que llamarlas con los mismos nombres con las que se han creado y respetar su sintaxis.

Las funciones que devuelven algún valor actualizan el primer de sus parámetros, esto ayuda al generador de código en bucles.

Ej: Resta A B devolvera resultado en A.

Excepto en los bucles, las operaciones hacen un recorrido recursivo, de derecha a izquierda. Por eso se pueden pasar resultados de operaciones como parámetros de otras funciones:

Suma A **Negacion B** el resultado de la operación en negrita se utilizará como segundo argumento en la función Suma.

Los bucles suelen requerir que se indique la variable de la cual deseamos devolver su valor. Ademas a modo informativo se exportará por la terminal información adicional del resto de estado de las variables.

Los bucles se forman rodeando su ámbito entre las palabras clave ‘Mientras’ y ‘Fin’. Al comienzo del bucle se establecen las condiciones del while → X **condición** Y y le siguen las operaciones que se desean ejecutar en bucle.

Las condiciones soportadas son:

*mayorIgualQue menorIgualQue menorQue mayorQue igualQue*

Las variables que se deseen utilizar en el programa deben seleccionarse como si fuesen parámetros, no pueden crearse variables en tiempo de ejecución.

Es IMPORTANTÍSIMO que la sintaxis utilizada para generar las operaciones sea precisa, ya que en caso contrario el código se generará con errores o en el peor de los casos, no se generará. No se deben incluir acentos, caracteres o espacios en ningún lugar que sean ajenos a esa sintaxis.

Hemos propuesto una serie de sentencias que permiten crear las distintas operaciones requeridas para ser generadas.

Para realizar la resta la instrucción que hay que introducir es:

* Nueva función: Resta
* Parámetros: marcar A y B.
* Definición: Suma A Negacion B
* Variable a devolver: Dejar en blanco

Para realizar la multiplicación la instrucción que hay que introducir es:

* Nueva función: Multiplicacion
* Parámetros: marcar A y B.
* Definición: Repeticion A Suma B B
* Variable a devolver: Dejar en blanco

Para realizar la división la instrucción que hay que introducir es:

* Nueva función: Division
* Parámetros: marcar A, B, C.
* Definición: Mientras A mayorIgualQue B Resta A B Suma C 1 Fin

Mientras C mayorIgualQue B Resta C B Suma A 1 Fin

* Variable a devolver: C

Para realizar la potencia la instrucción que hay que introducir es:

* Nueva función: Potencia
* Parámetros: marcar A, B.
* Definición: Repeticion B Multiplicacion A A
* Variable a devolver: Dejar en blanco

Para realizar el factorial la instrucción que hay que introducir es:

* Nueva función: Factorial
* Parámetros: marcar A y B.
* Definición: Mientras B mayorQue 1 Resta B 1 Multiplicacion A B Fin
* Variable a devolver: Dejar en blanco

Para realizar el porcentaje la instrucción que hay que introducir es:

* Nueva función: Porcentaje
* Parámetros: marcar A, B.
* Definición: Dividir 0 B Multiplicacion A 100
* Variable a devolver: Dejar en blanco

Para realizar la media aritmética la instrucción que hay que introducir es:

* Nueva función: Media
* Parámetros: marcar A, B.
* Definición: Dividir 0 numParam Suma A Suma B Suma C D
* Variable a devolver: Dejar en blanco

Para realizar la raíz cuadrada la instrucción que hay que introducir es:

* Nueva función: Raiz
* Parámetros: marcar A, B, C.
* Definición: Mientras B menorIgualQue A Suma C 2 Resta B B Suma B C Multiplicacion B B Resta C 1 Fin.
* Variable a devolver: C.

Para generar el código, una vez introducidos todos los datos, simplemente hay que hacer clic sobre el botón “Crear”.

En la lista de la izquierda del programa están las funciones que se pueden utilizar. Inicialmente únicamente están Suma, Negacion y Repeticion. Una vez que se genera una función, ésta también es incluida en dicha lista.

# Salidas

Una vez pulsado el botón crear, el programa genera el código fuente. Dicho código fuente incluye un método main desde el cual se podrían hacer las comprobaciones de las operaciones. Más adelante se explicará como realizar estas comprobaciones.

# Cómo se genera el código

En el momento en el que se hace clic sobre “Crear” el programa crea un nuevo documento con extensión java y con el nombre que se le haya asignado a la función. Este documento contiene el código fuente de la nueva función. Dentro de este código fuente se incluye un método main, que en primer lugar instancia un objeto sobre el que se realizará la operación, al que se le pueden pasar los argumentos necesarios para realizar operaciones, y posteriormente devolver por pantalla el resultado obtenido.

Las nuevas clases generadas heredan todas ellas de la clase operaciones que define su estructura interna.

El constructor de cada clase simplemente se encarga de inicializar los valores a las distintas variables indicadas por el usuario.

Además estas nuevas clases tienen el método operar sobrecargado (está instanciado dos veces).

Las nuevas funciones creadas han de tener acceso a las operaciones que necesite para su correcto funcionamiento, por ejemplo para la multiplicación son necesarias las clases suma y la repetición.

El codigo no se compila de forma automática, por lo que previamente a la ejecución, habría que compilar el código que sólo funcionará correctamente si se ha respetado la sintaxis del lenguaje indicadas previamente.

# Librerías externas utilizadas

Para realizar la práctica no ha sido necesaria la utilización de ninguna librería externa.

# Lenguajes y tecnologías utilizadas en cada paso

El lenguaje empleado para la realización de la práctica ha sido el lenguaje Java. Para diseñar la interfaz de usuario del programa se ha utilizado el plugin de Eclipse WindowBuilder que a su vez utiliza la biblioteca gráfica para Java, Swing.

# Flujo de datos

Los datos introducidos en los campos el en generador de código son analizados en el momento de generación (el usuario presiona el botón ‘Crear’) y la información creada se va almacenando en un buffer que se vierte sobre un nuevo fichero llamado como la operación.

Si se desea ejecutar operaciones creadas se deben indicar en el main, se esta forma JVM introduce los datos en stack de ejecución.

El valor devuelto por la operación será el resultado de tal.

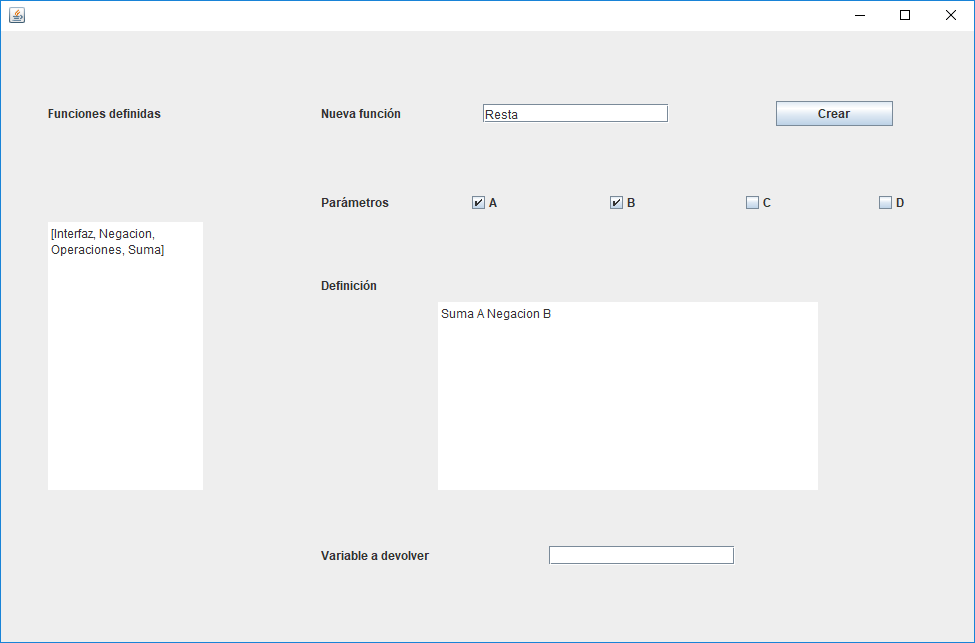
# IDE utilizado para el desarrollo

El IDE utilizado para el desarrollo de la práctica ha sido Eclipse.

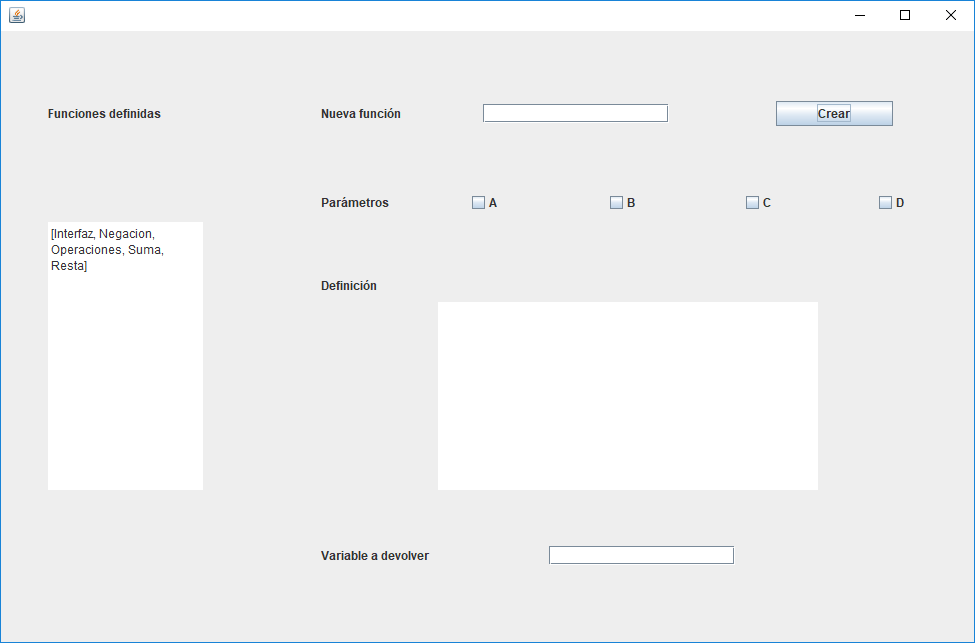
Ejemplo práctico con capturas de pantalla

En primer lugar habría que ejecutar el programa, para ello se importa el proyecto en Eclipse y se ejecuta la clase interfaz, que es la clase principal. Una vez dentro habría que generar la operación. Para explicar como hacerlo se va a utilizar la generación de la operación resta.

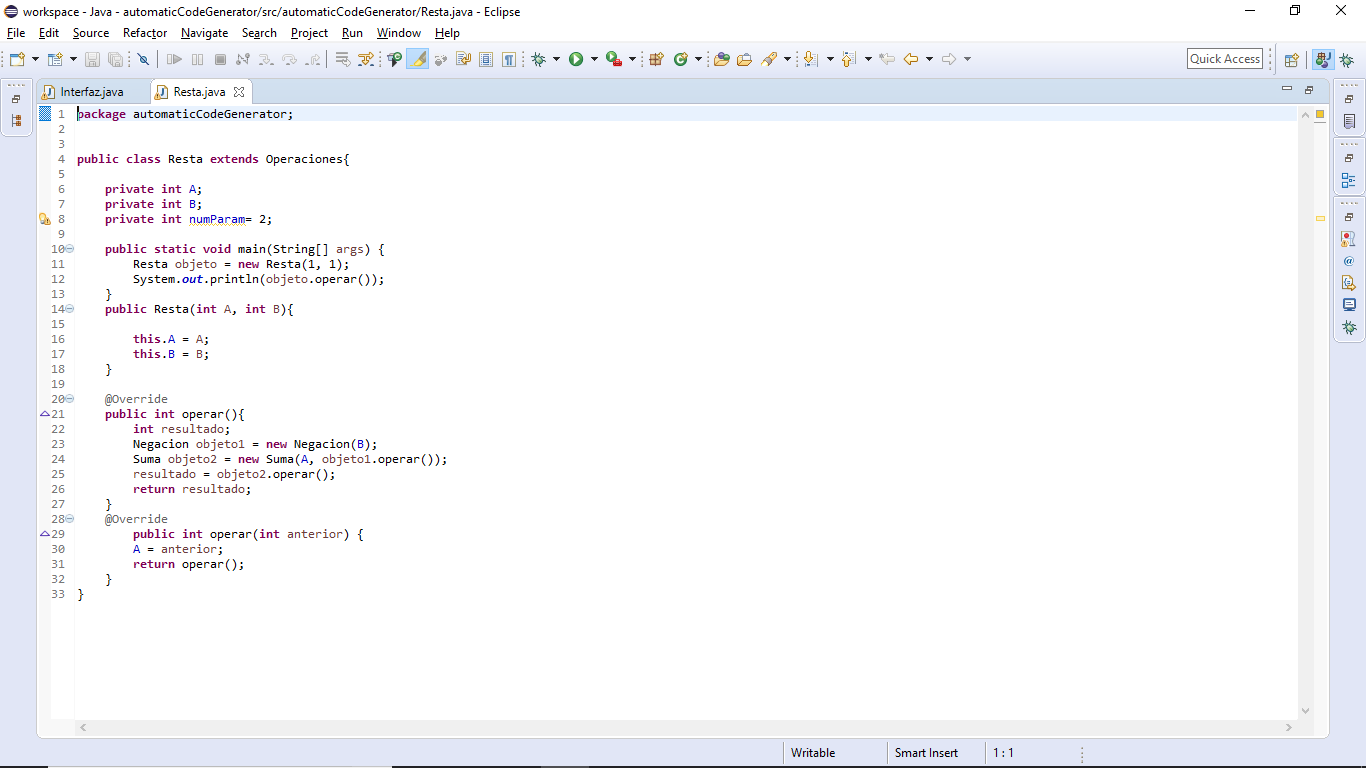
El primer paso sería introducir todos los datos necesarios como se explicó en el apartado entradas.



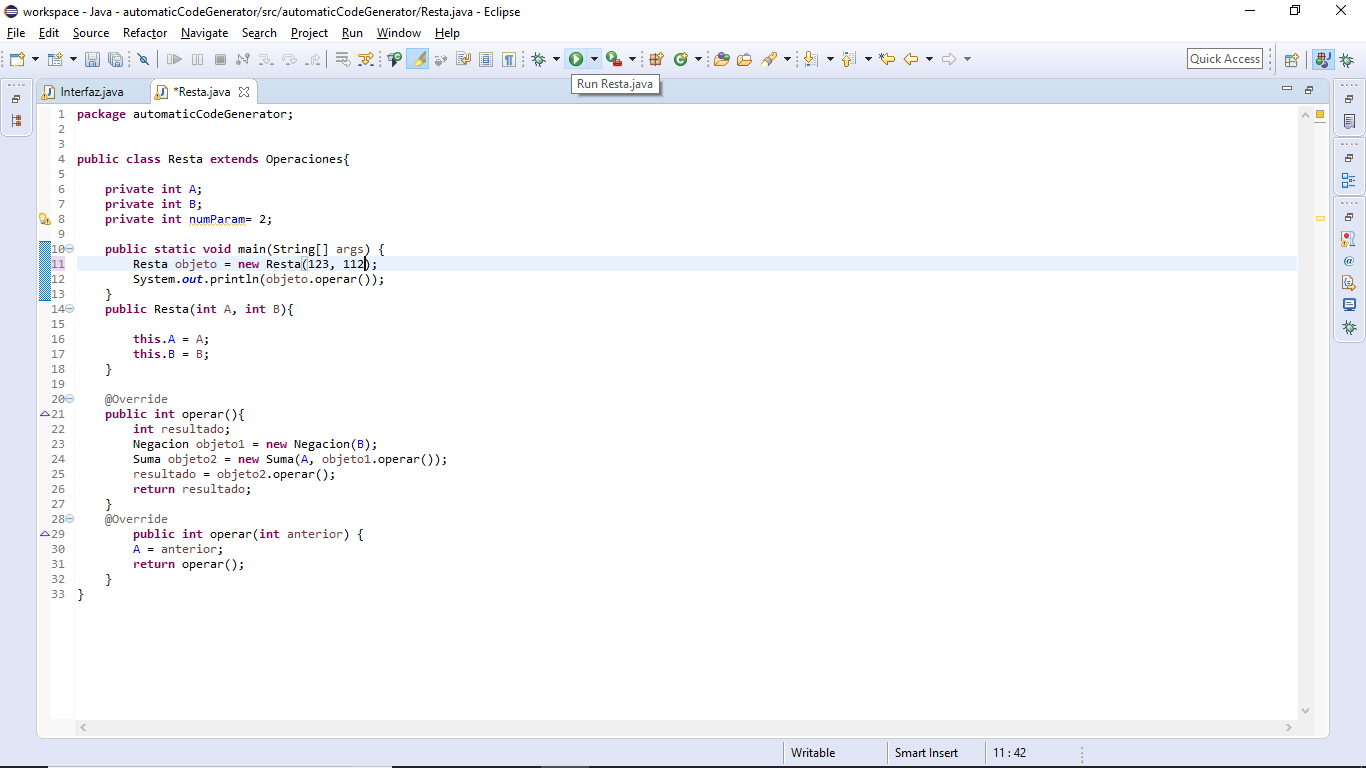
Una vez introducidos los datos se pulsa sobre el botón crear, obteniendo como resultado el código fuente de la nueva operación. Además se incluye en la lista la nueva función disponible.



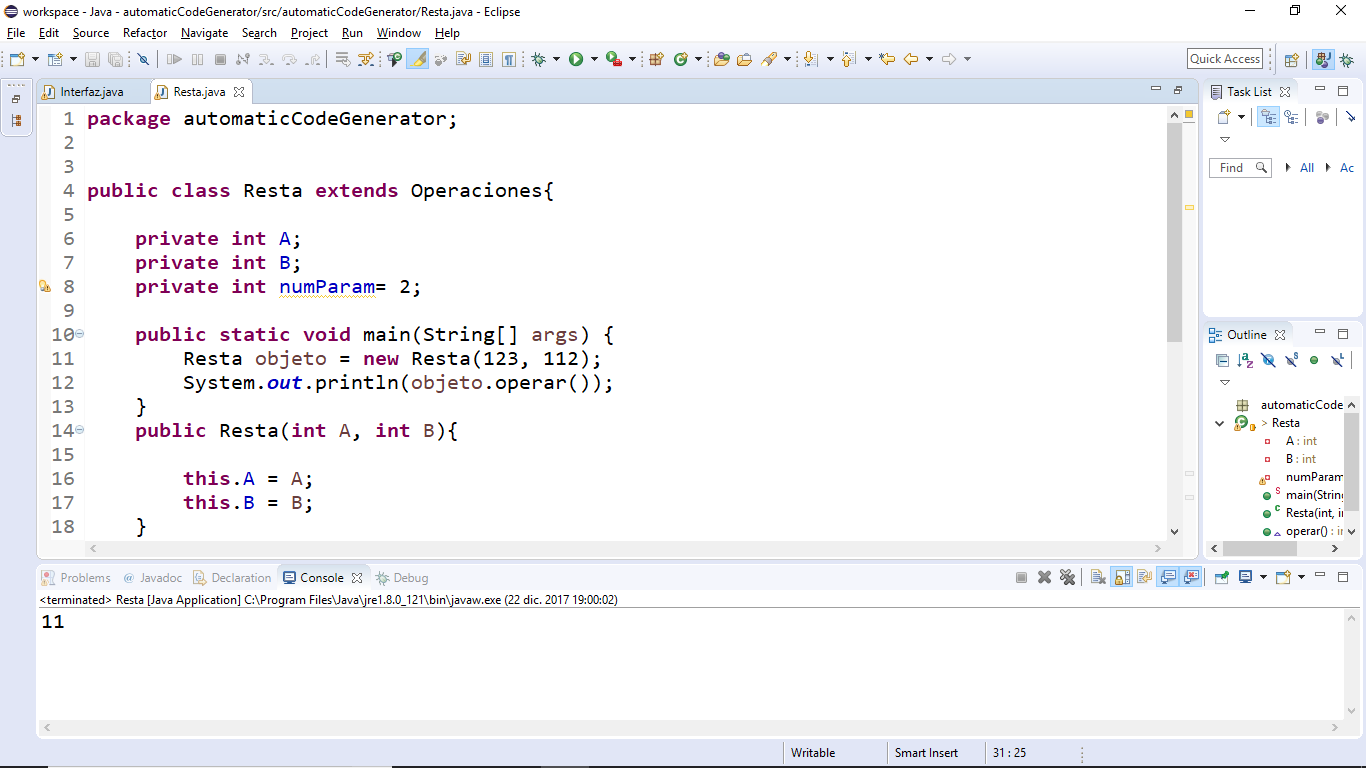
El código generado por el programa sería el siguiente. Para hacer restas habría que cambiar los valores del nuevo objeto de tipo Resta. En la imagen estos valores son 1 y 1.



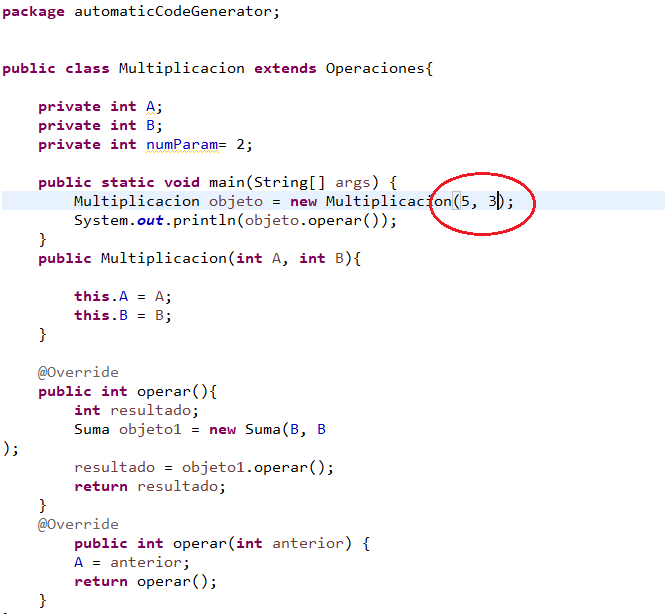
En este ejemplo se restarán los valores 123 y 112. Para realizar la operación se ejecuta la clase.



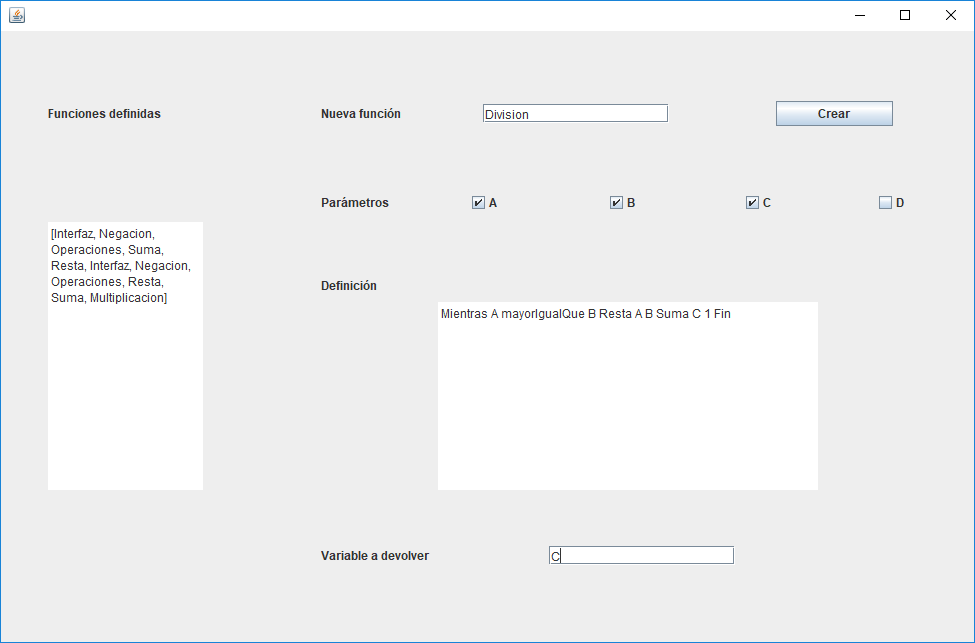
Al ejecutar la clase se muestra por pantalla el resultado de la operación.



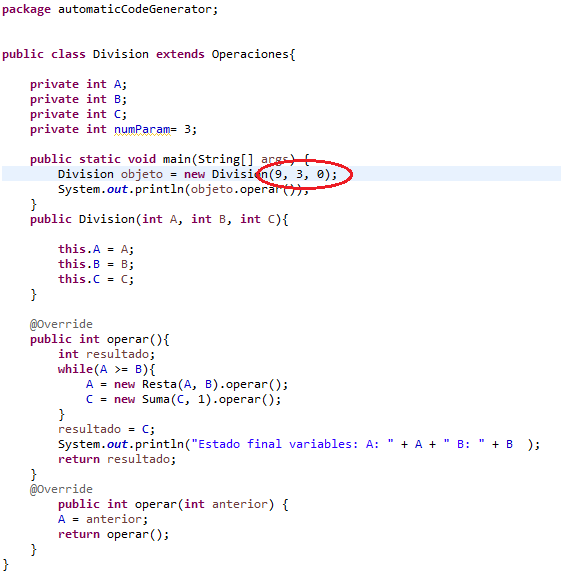
Para la multiplicación las variables que habría que cambiar son:



Para generar la división, que se trata de una de las operaciones más complejas, los datos a introducir serían:

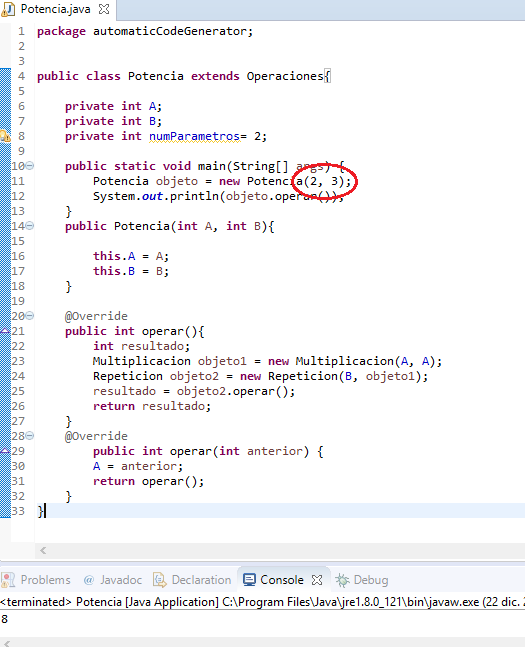


Y para cambiar los valores a dividir:



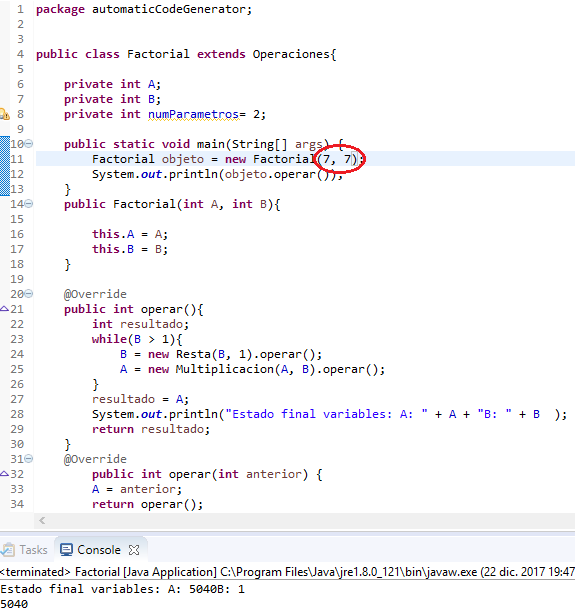
El primer valor sería el dividendo, el segundo el divisor, y el tercer valor siempre hay que ponerlo a 0, ya que se trata de un contador.

Para la potencia, los valores que habría que cambiar son:

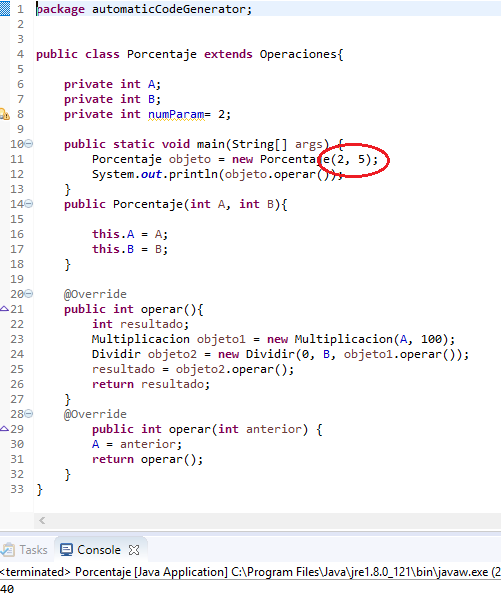


Siendo 2 la base y 3 el exponente.

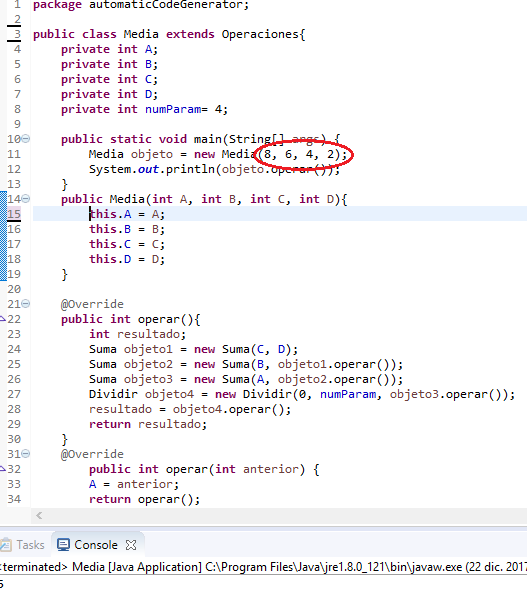
Para el factorial habría que cambiar los dos valores marcados con el valor del número del que se quiera obtener el factorial.

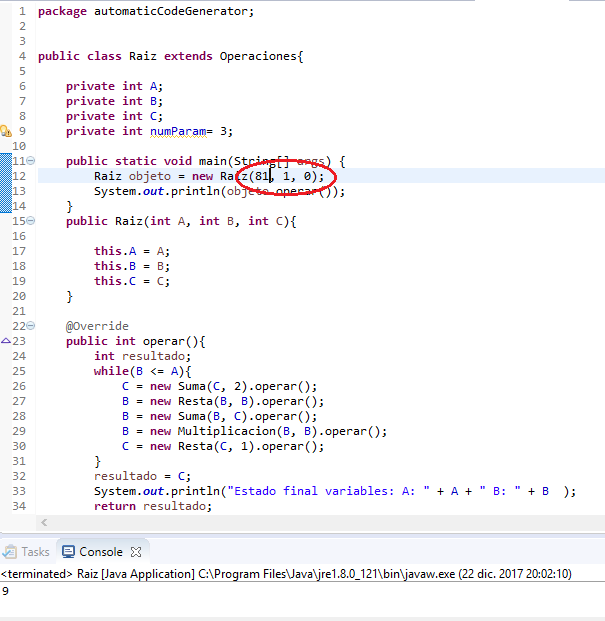


Para el porcentaje, el primer valor sería el dividendo y el segundo el divisor:



Para la media, habría que cambiar los cuatro valores por los que se deseen:



Para la raíz, se cambiaría el primer valor por el deseado, y el segundo y tercer parámetros se dejan como aparece en la imagen:

# Integrantes del grupo

Adolfo Jiménez Prieto 71745763T

Victor de Vierna Valcarce 71457636V

Adrián Arias Viñuela 71466129T