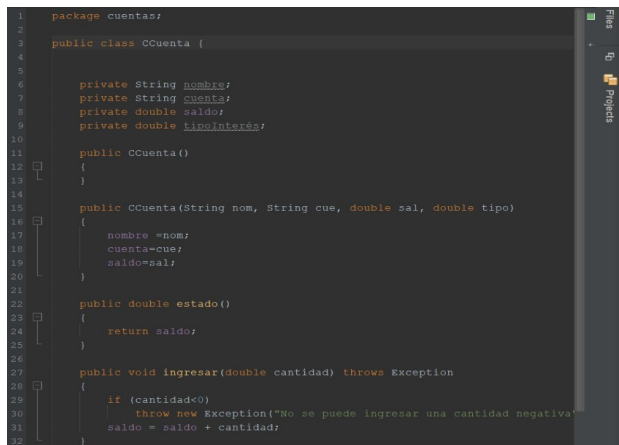
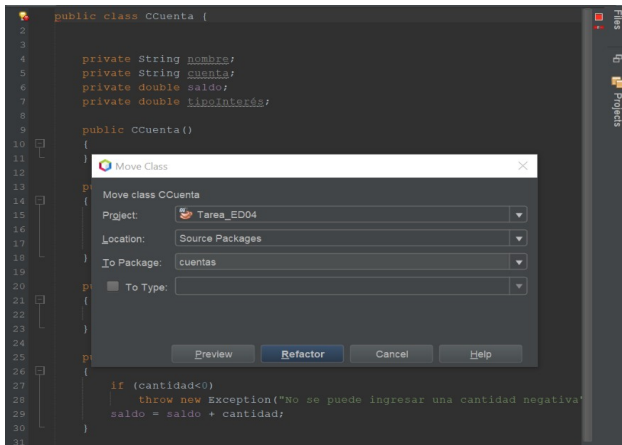


Tarea ED04.

Refactorización.

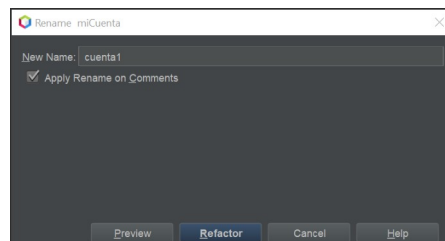
1. Las clases deberán formar parte del paquete cuentas.

Creamos un nuevo proyecto en NetBeans y un paquete *cuentas*, con refactor/move movemos los dos .java al paquete creado y pulsamos en *Refactor*. Podemos comprobar como en la parte superior de la clase se introduce en la primera linea la referencia al paquete al que pertenece.

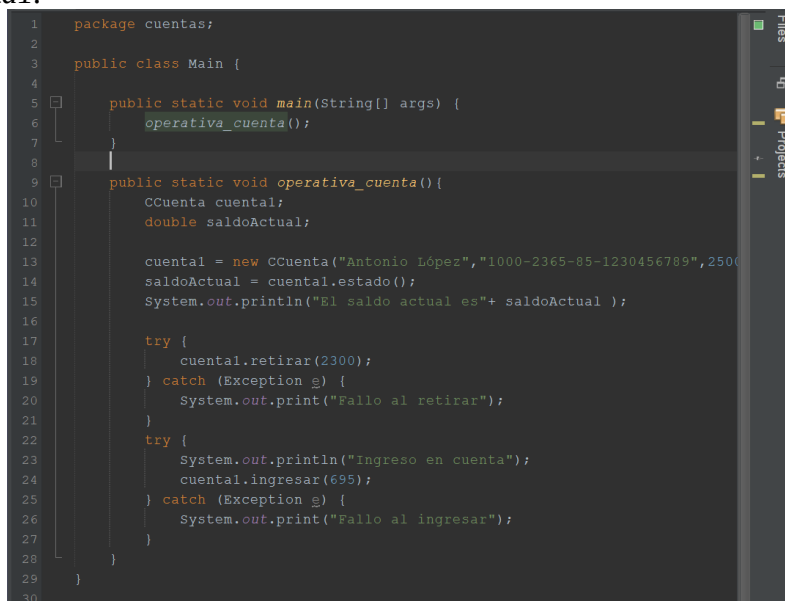


2. Cambiar el nombre de la variable “miCuenta” por “cuenta1”.

Hacemos click derecho sobre la variable que queremos refactorizar y le damos a refactor/rename.

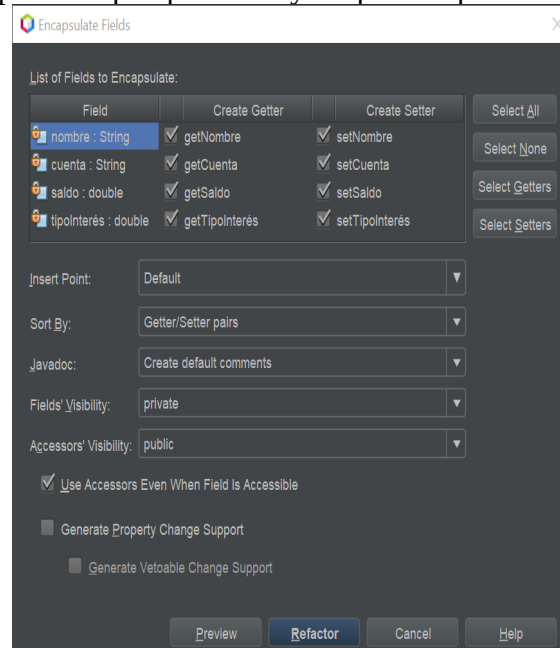


3. Introducir el método operativa_cuenta, que englobe las sentencias de la clase Main que operan con el objeto cuenta1.



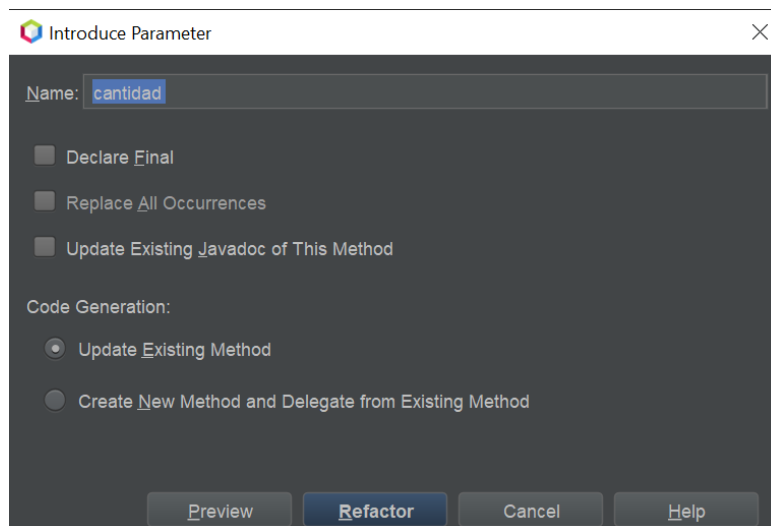
4. Encapsular los atributos de la clase Ccuenta.

De nuevo hacemos click derecho en la clase que queremos refactorizar, Refactor/Encapsulate Fields, seleccionamos las opciones que queramos y aceptamos pulsando en Refactor.



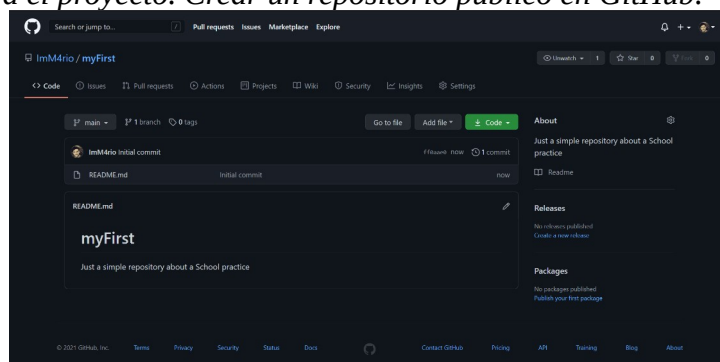
5. Añadir un nuevo parámetro al método operativa_cuenta, de nombre cantidad y de tipo float.

Creamos la nueva variable, de nuevo hacemos click derecho sobre ella, Refactor/Introduce/Parameter luego tendremos que pasar algún valor de tipo float donde se llama al método desde main

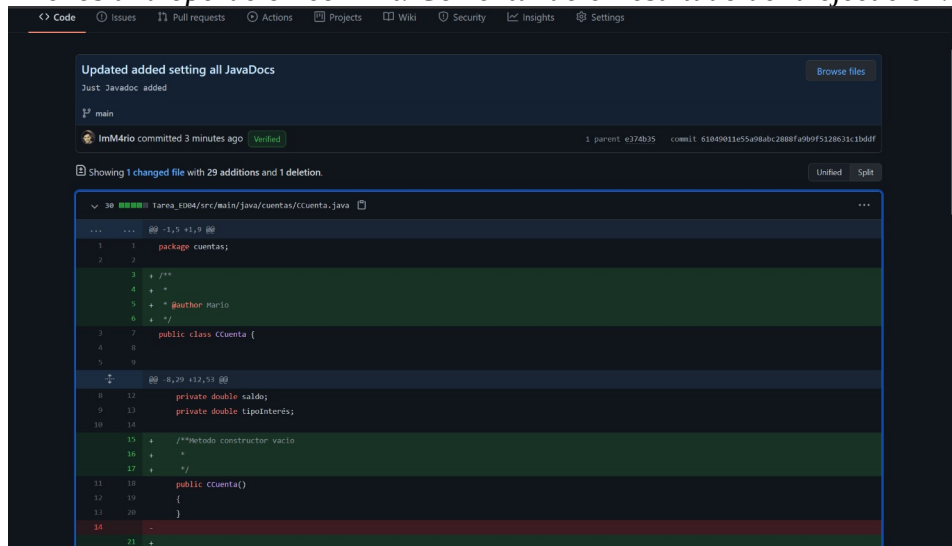


Git.

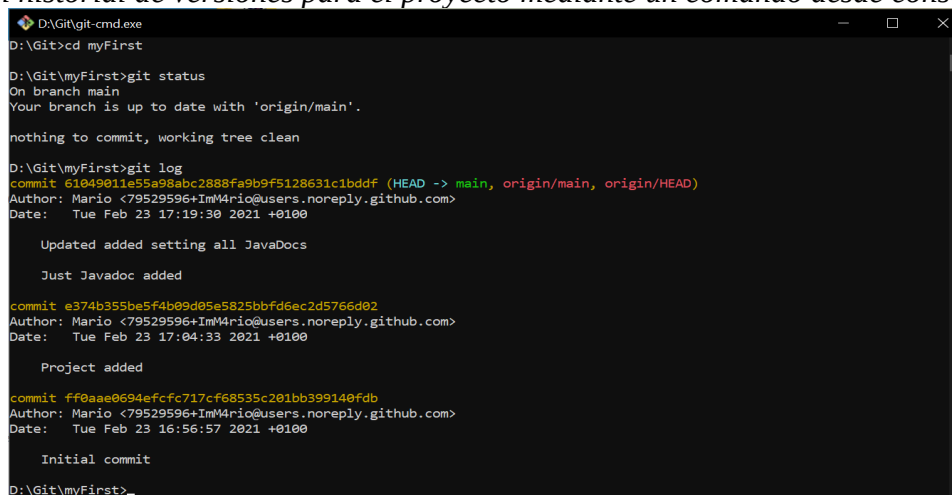
1. Configura GIT para el proyecto. Crear un repositorio público en GitHub.



2. Realizar al menos una operación commit. Comentando el resultado de la ejecución.

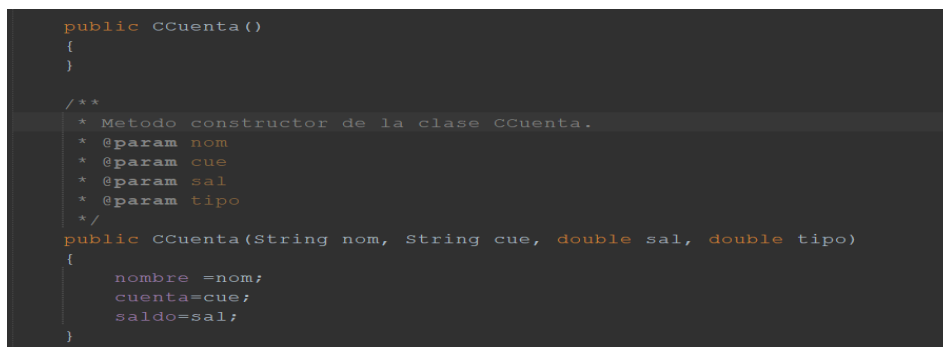


3. Mostrar el historial de versiones para el proyecto mediante un comando desde consola.



Javadoc.

1. Insertar comentarios Javadoc en la clase Ccuenta.



2. Generar documentación Javadoc para todo el proyecto y comprueba que abarca todos los métodos y atributos de la clase Ccuenta.

Si queremos añadir el javadoc automáticamente tenemos que ir a la pestaña de *Projects* y hacer click derecho sobre la clase que se quiere generar el código.

Tools/Analyze Javadoc → seleccionamos los Javadoc

