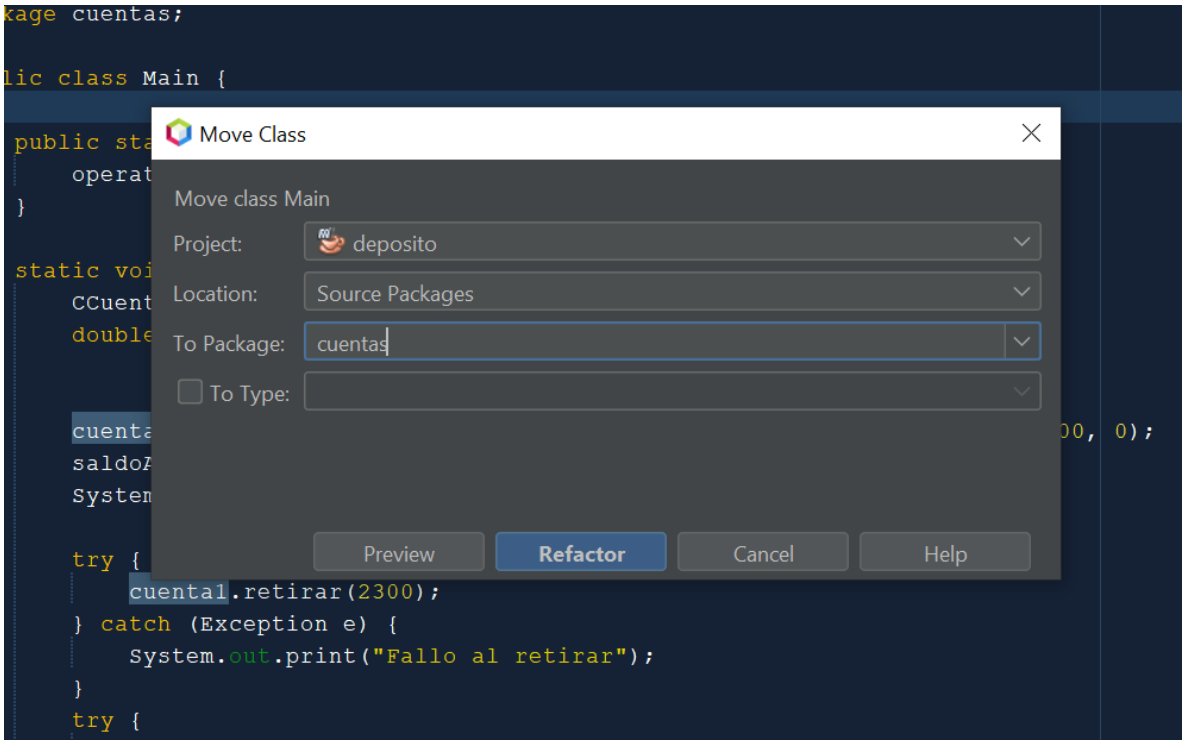
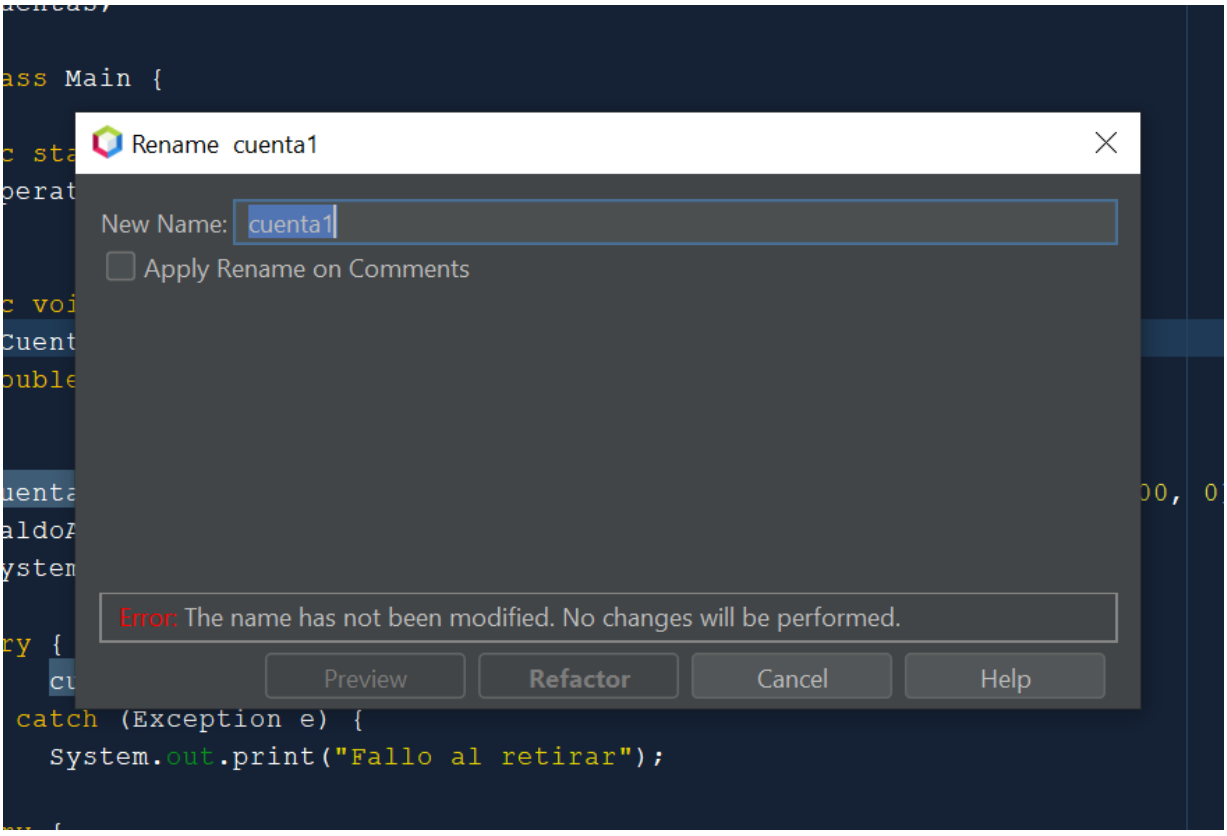
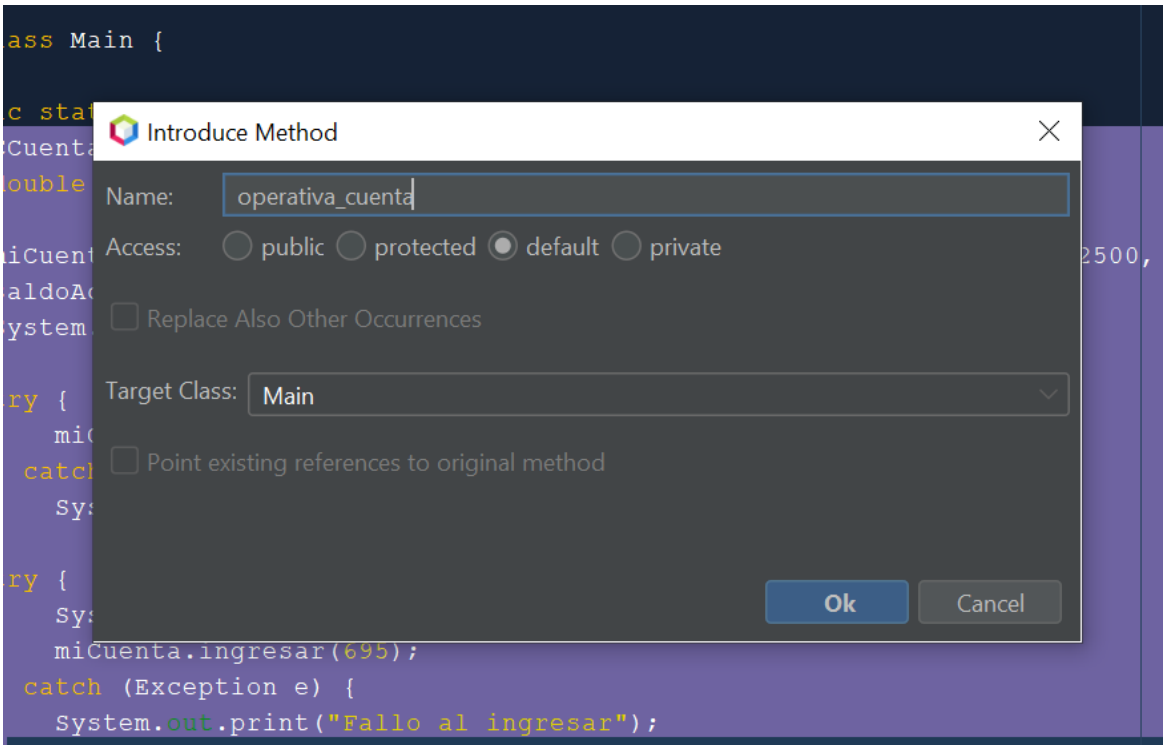
1


kage cuentas;
lic class Main {
public sta
operat
}
static voi
CCuent
double
cuenta
saldoA
Systen
try {
cuenta1.retirar(2300);
} catch (Exception e) {
System.out.print("Fallo al retirar");
}
try {
Move Class
Move class Main
Project:
deposito
Location:
Source Packages
To Package:
cuentas
To Type:
Preview
Refactor
Cancel
Help

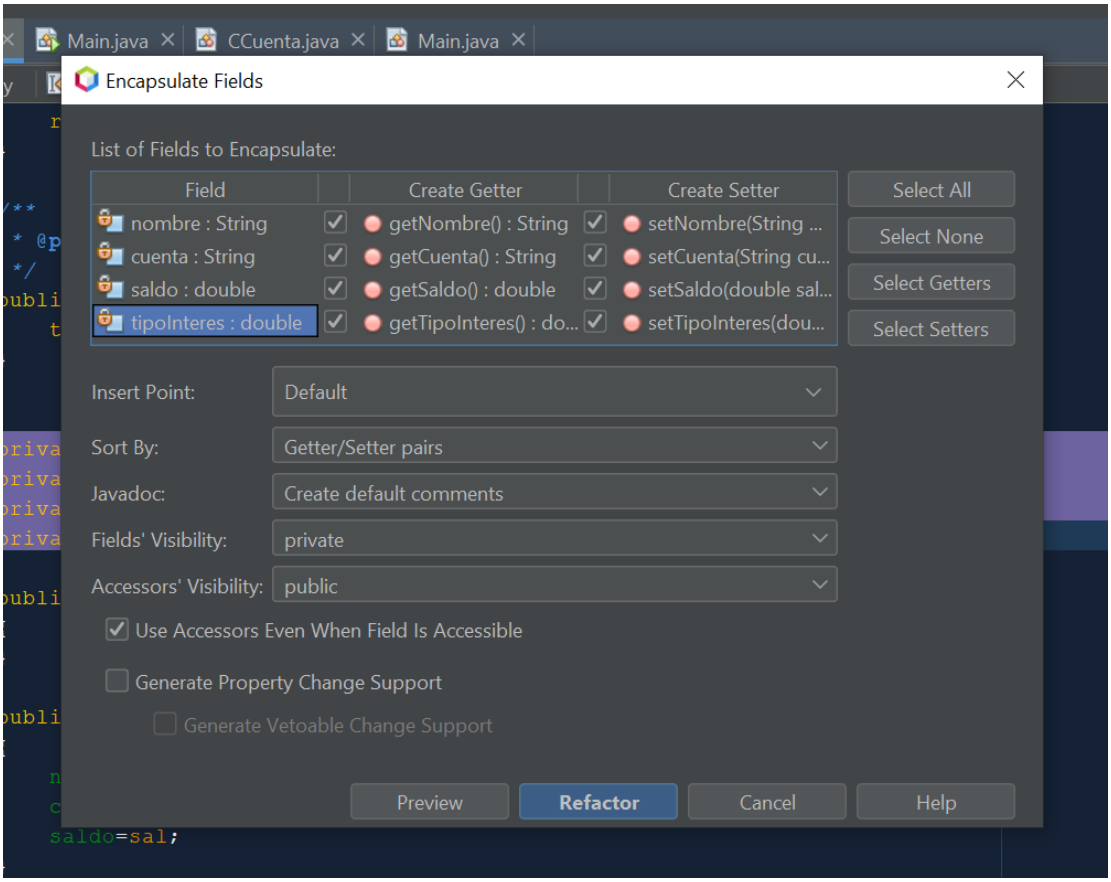
2


ass Main {
c sta
perat
c voi
Cuent
ouble
uenta
aldoA
ystem
ry {
cu
catch (Exception e) {
System.out.print("Fallo al retirar");
Rename cuenta1
New Name:
cuenta1
Apply Rename on Comments
Error: The name has not been modified. No changes will be performed.
Preview
Refactor
Cancel
Help

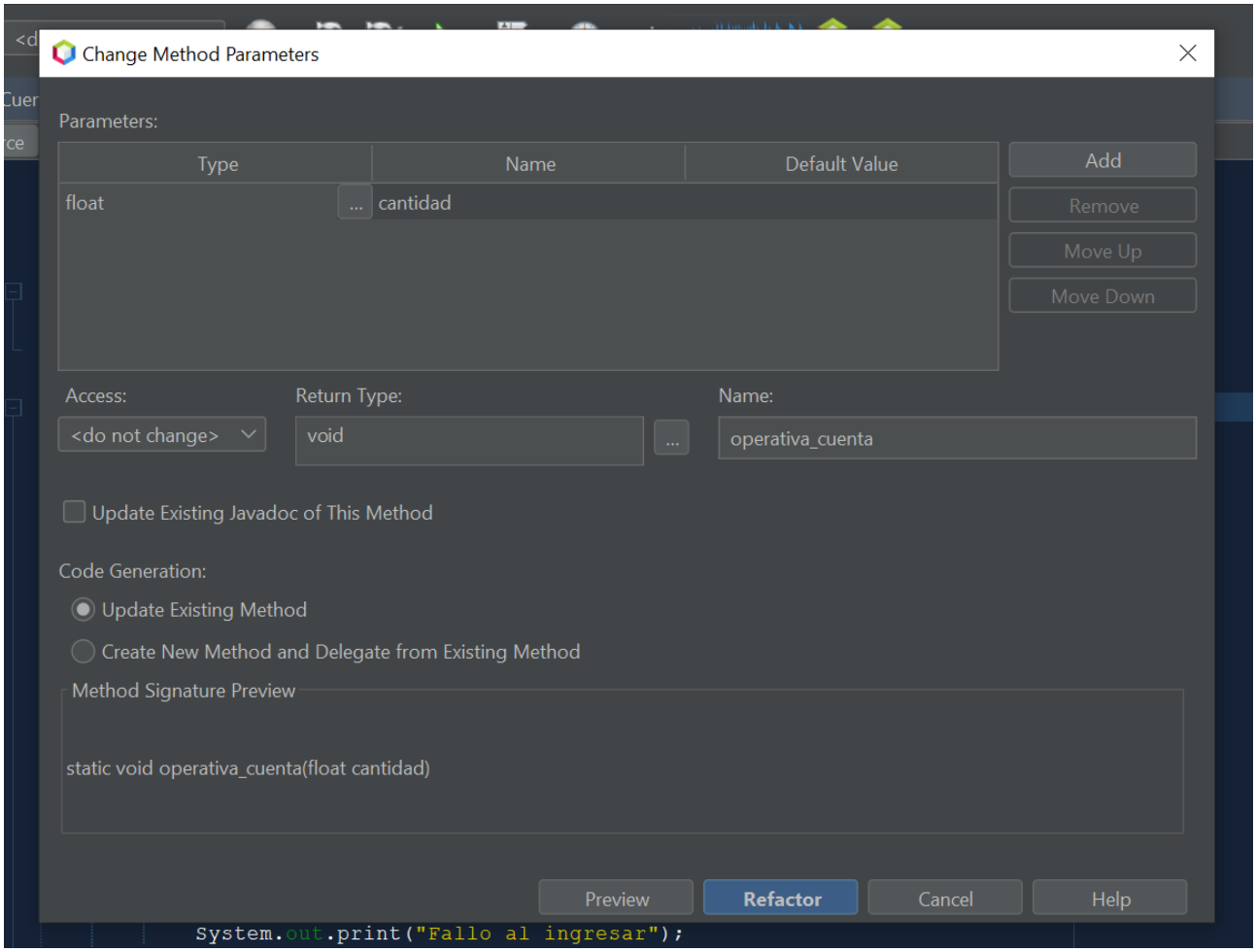
3


Introduce Method
Name: operativa_cuenta
Access: public protected default private
Replace Also Other Occurrences
Target Class: Main
Point existing references to original method
Ok
Cancel
miCuenta.ingresar(695);
catch (Exception e) {
System.out.print("Fallo al ingresar");

4


Main.java
CCuenta.java
Main.java
Encapsulate Fields
List of Fields to Encapsulate:
Field
Create Getter
Create Setter
nombre : String
getNombre() : String
setNombre(String ...
cuenta : String
getCuenta() : String
setCuenta(String cu...
saldo : double
getSaldo() : double
setSaldo(double sal...
tipoInteres : double
getTipoInteres() : do...
setTipoInteres(dou...
Select All
Select None
Select Getters
Select Setters
Insert Point: Default
Sort By: Getter/Setter pairs
Javadoc: Create default comments
Fields' Visibility: private
Accessors' Visibility: public
Use Accessors Even When Field Is Accessible
Generate Property Change Support
Generate Vetoable Change Support
Preview
Refactor
Cancel
Help
saldo=sal;

5


Change Method Parameters
Parameters:
Type
Name
Default Value
float
cantidad
Add
Remove
Move Up
Move Down
Access:
<do not change>
Return Type:
void
Name:
operativa_cuenta
Update Existing Javadoc of This Method
Code Generation:
Update Existing Method
Create New Method and Delegate from Existing Method
Method Signature Preview
static void operativa_cuenta(float cantidad)
Preview
Refactor
Cancel
Help
System.out.print("Fallo al ingresar");