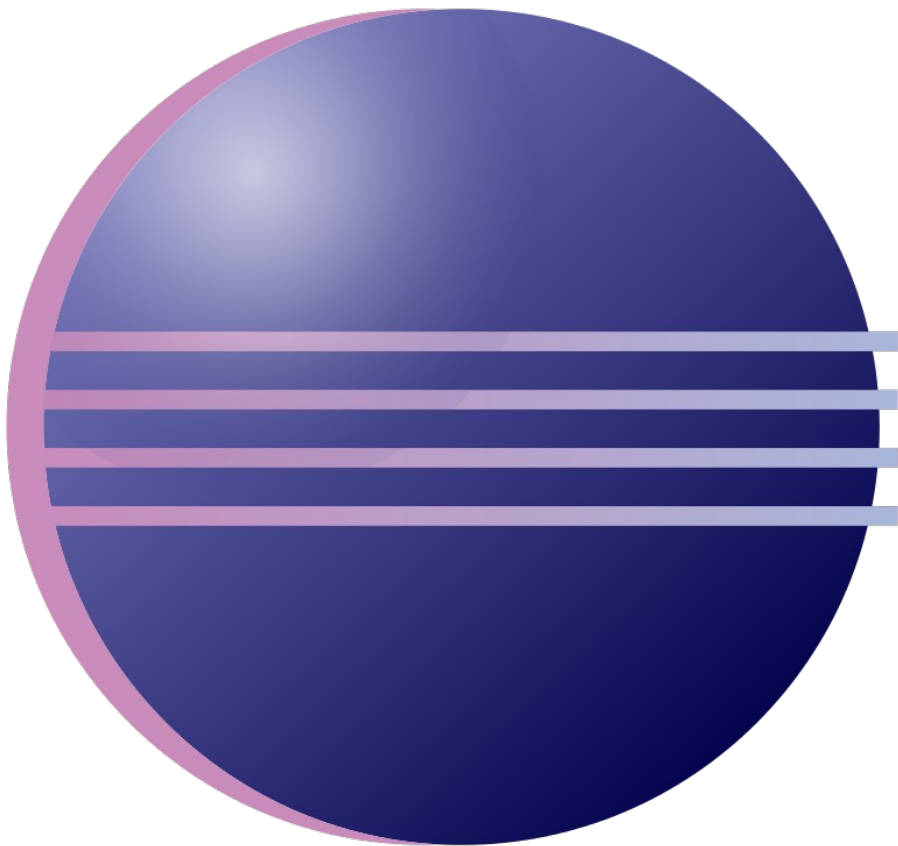


López Calderón Francisco Javier

REFACTORIZACIÓN PRÁCTICA

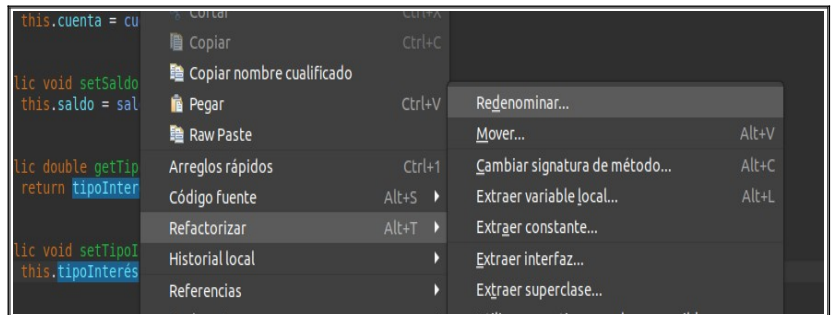


Índice

1) Uso de rename.....	3
2) Introducir constante.....	4
3) Sustituir bloques.....	4
4) Campos encapsulados.....	5
5) Mover clase de un paquete.....	5
6) Change Method signature.....	5
7) Extract Interface.....	6
8) Refactorización menu.....	7

1) Uso de rename

Hacemos clic derecho en la variable y procedemos cambiar el nombre

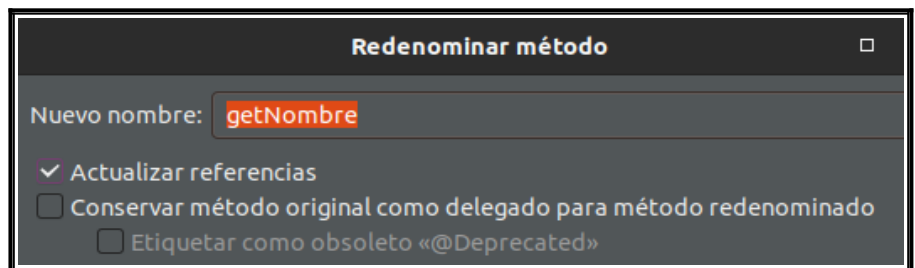


```
public double getTipoInterés() {
    return tipoInterés;
}

public void setTipoInterés(double tipoInterés) {
    this.tipoInterés = tipoInterés;
}
```

Resultado tras utilizar rename

Otro método



```
public String getCuenta() {
    return cuenta;
}
```

```
public double getTipoInterés() {
    return tipoInterés;
}
```

```
public void setTipoInterés(double tipoInterés) {
    this.tipoInterés = tipoInterés;
}
```

```
public double getSaldo() {  
    return saldo;  
}
```

```
public class CuentaCorriente {
```

Finalmente...

2) Introducir constante

Se asigna variable ZERO

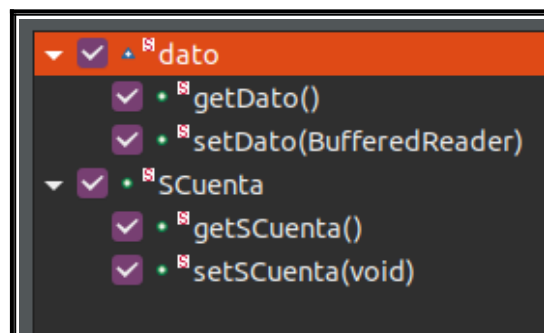
Nombre de constante:

Modificador de acceso: ☐ public ☐ protected ☐ package ☒ private

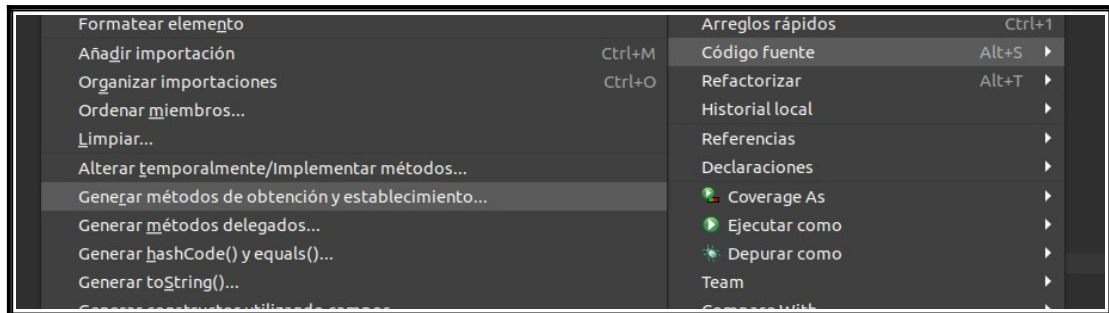
☒ Sustituir todas las apariciones de la expresión seleccionada por referencia

☐ Cualificar referencias a constantes con el nombre del tipo

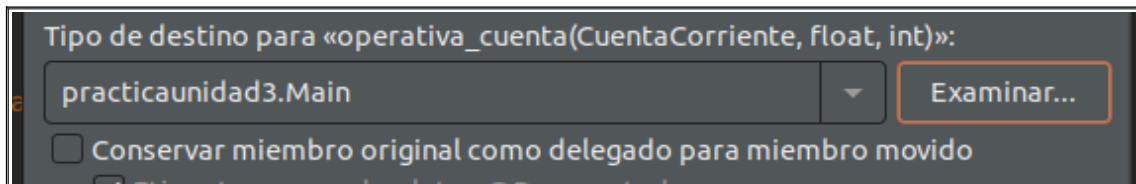
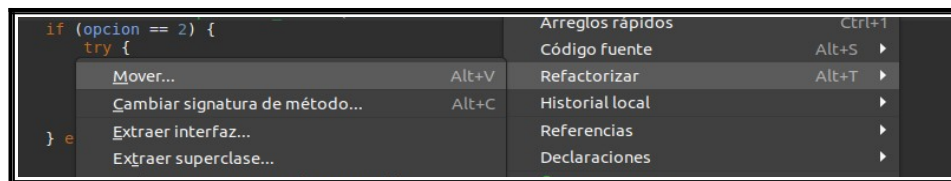
3) Sustituir bloques



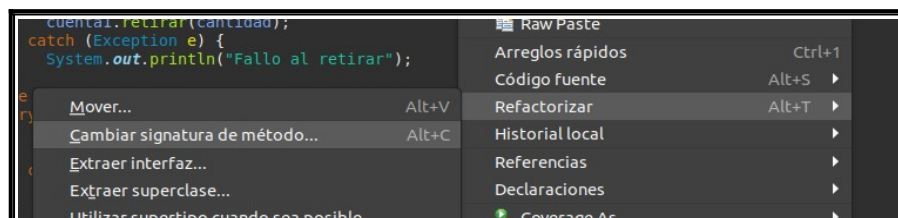
4) Campos encapsulados



5) Mover clase de un paquete

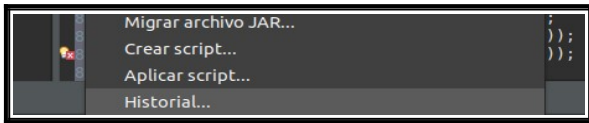


6) Change Method signature

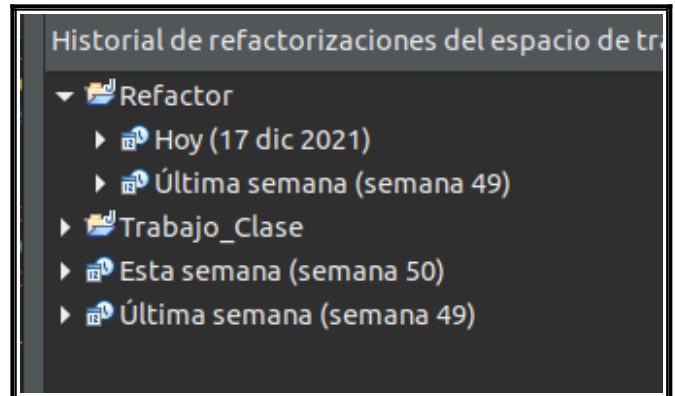


```
public static void VisualizarDatosCuenta(CuentaCorriente cuental, float cantidad, int opcion) {
    VisualizarDatosCuenta(cuental, cantidad, opcion, 4);
    System.out.println("Buenos días");
}
```

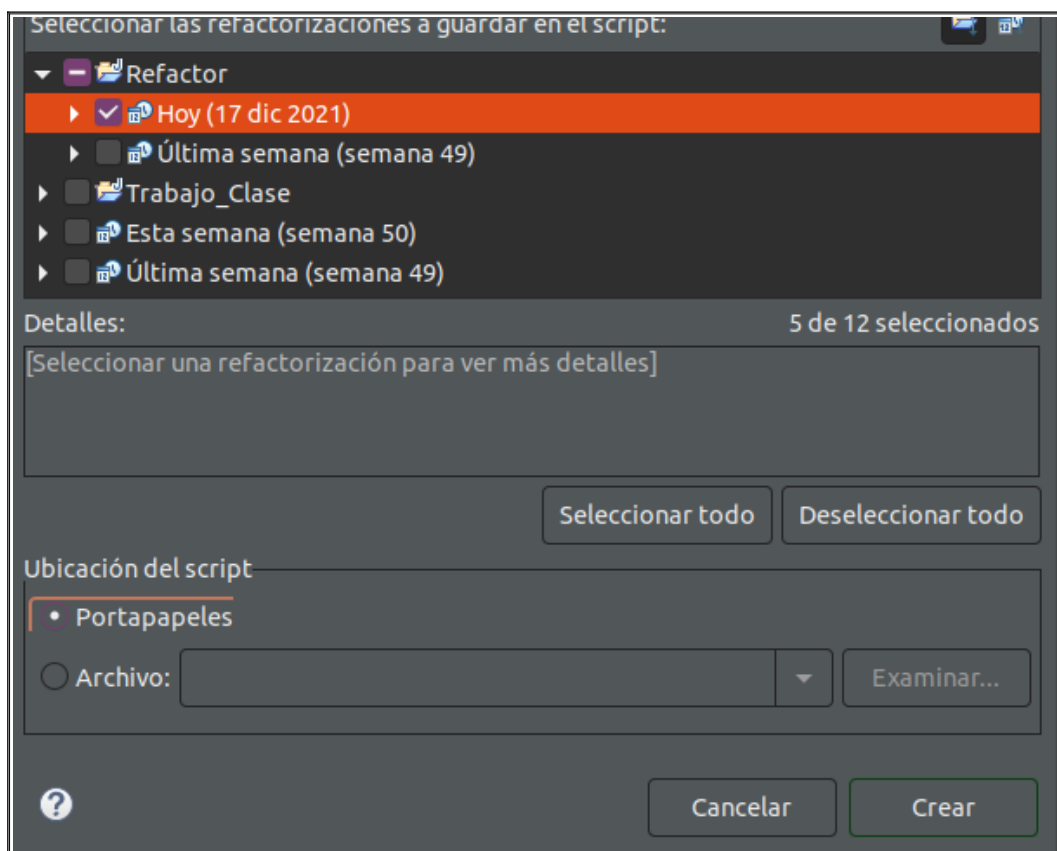
7) Extract Interface



Se presenta el historial de refactorizaciones



Creación del script de refactorización



8) Refactorización menu

