

Part 1: Depuració del Codi

- **Què fa el codi? (Explicar breument).** El código implementa un sistema bancario sencillo en java que permite gestionar una cuenta la clase Account representa una cuenta con atributos con el titular, el número de cuenta y el salario y proporciona métodos para ingresar y retirar dinero, controlado errores amb excepciones. La clase Main crea una cuenta con un saldo inicial de 2500, intenta retirar 2300 después ingresa 1695 y finalmente muestra el saldo actual si hay errores como insuficiente o importes negativos es gestionar con mensaje de error
- **Quins són els mètodes més importants i què fan?**

depositAmount(double amount)

- Permite ingresar dinero a la cuenta
- Si el valor es negativo lanza una excepción
- Si es valido se suma al saldo

withdrawAmount(double amount)

- Permite retirar dinero de la cuenta
- Si el monto es negativo lanza una excepción
- Si el saldo es insuficiente lanza una excepción
- Si es valido se descuenta del saldo

getBalance()

- Devuelve el saldo actual de la cuenta

- **Quin és el valor inicial del saldo (balance) abans de realitzar qualsevol operació?** El valor inicial del saldo antes de realizar cualquier acción es 2500

2. Posar punts de control (Breakpoints): Per depurar el codi, utilitza els punts de control (breakpoints). Això permet aturar l'execució del codi en determinats punts i examinar l'estat de les variables. Per afegir un punt de control, fes clic a la barra de l'esquerra de la línia on vols aturar el codi.

- **On has col·locat els punts de control (breakpoints) i per què?**

`myAccount = new Account("Flor Martinez",
"1000-1234-56-123456789", 2500);` → Para verificar que el objeto myAccount se crea correctamente con el saldo inicial

`myAccount.withdrawAmount(2300);` → Para ver si la funcion whitdrawAmount se llama correctamente y si el saldo se reduce y para comprobar que no se lanza una excepcion si el saldo es suficiente

`myAccount.depositAmount(1695);` → Para verificar que la funcion depositAmount agrega correctamente el valor al saldo

`System.out.println("El saldo actual es " +
myAccount.getBalance());` → Para confirmar que el saldo final refleja correctamente las operaciones previas

- Inclou una captura de pantalla de Eclipse amb els breakpoints activats abans de començar la depuració.

```
10 myAccount = new Account("Flor Martinez", "1000-1234-56-123456789", 2500);
11
12 try {
13     myAccount.withdrawAmount(2300);
14 } catch (Exception e) {
15
16 }
17
18 System.out.println("Ingres al compte");
19 myAccount.depositAmount(1695);
20 } catch (Exception e) {
21
22 }
23
24 System.out.println("El saldo actual es " + myAccount.getBalance());
25 }
26 }
27 }
```

3. Examina les variables i el flux d'execució:

- A mesura que el codi s'atura a cada punt de control, observa el valor de les variables name, account i balance. Inclou una captura de pantalles dels valors de les variables a mesura que avancen les operacions.

```
myAccount= Account (id=24)
> account= "1000-1234-56-123456789" (id=26)
> balance= 200.0
> name= "Flor Martinez" (id=33)

SimpleBankingSystem.Account@1d8d30f7
```

```
▼ ⓘ myAccount= Account (id=24)
  > • account= "1000-1234-56-123456789" (id=26)
  • balance= 1895.0
  > • name= "Flor Martinez" (id=33)

SimpleBankingSystem.Account@1d8d30f7
```

```
▼ ⓘ myAccount= Account (id=24)
  > • account= "1000-1234-56-123456789" (id=26)
  • balance= 1895.0
  > • name= "Flor Martinez" (id=33)

SimpleBankingSystem.Account@1d8d30f7
```

4. Explora les excepcions:

- Feu els canvis necessaris al Main per fer saltar les excepcions. Inclou la captura de pantalla d'un missatge d'error generat per una excepció i com es visualitza al terminal o a la consola de Eclipse.

```
try {
    System.out.println("Intentant retirar més diners dels disponibles...");
    myAccount.withdrawAmount(5000); // Provoca una excepció perquè el saldo és insuficient
} catch (Exception e) {
    System.err.println(e.getMessage());
    System.out.println("Error al retirar");
}

try {
    System.out.println("Intentant ingressar una quantitat negativa...");
    myAccount.depositAmount(-500); // Provoca una excepció
} catch (Exception e) {
    System.err.println(e.getMessage());
    System.out.println("Error en l'ingrés");
}

System.out.println("El saldo actual es " + myAccount.getBalance());
```

```
Intentant retirar més diners dels disponibles...
No hi ha suficient saldo
No es pot ingressar una quantitat negativa.
Error al retirar
Intentant ingressar una quantitat negativa...
Error en l'ingrés
El saldo actual es 2500.0
```