

ÍNDICE

PRÁCTICAS DE PRUEBAS: CLASE CCUENTA 2

ENUNCIADO DEL PROBLEMA: 2

Prácticas de Pruebas: clase CCuenta

Enunciado del Problema:

En esta tarea se considera una clase Java **CCuenta** que dispone de los métodos **main**, **ingresar** y **retirar**. Se proporciona el código de los métodos **main** e **ingresar** que deberás tener en cuenta para resolver la tarea. El código de la clase se adjunta también en esta tarea.

Deberás dar respuesta a los siguientes apartados:

- Realiza un análisis de caja blanca completo del método **ingresar**.
- Realiza un análisis de caja negra, incluyendo valores límite y conjetura de errores del método **retirar**. Debes considerar que este método recibe como parámetro la cantidad a retirar, que no podrá ser menor a 0. Además, en ningún caso esta cantidad podrá ser mayor al saldo actual. Al tratarse de pruebas funcionales no es necesario conocer los detalles del código.
- Crea la clase **CCuentaTest** del tipo Caso de prueba **JUnit** en Eclipse que nos permita pasar las pruebas unitarias de caja blanca del método **ingresar**. Los casos de prueba ya los habrás obtenido en el primer apartado del ejercicio.
- Genera los siguientes puntos de ruptura para validar el comportamiento del método **ingresar** en modo depuración.
 - Punto de parada sin condición al crear el objeto **miCuenta** en la función **main**. Línea 3 del código del método **main** que se presenta en la siguiente página de este libro.
 - Punto de parada en la instrucción **return** del método **ingresar** sólo si la cantidad a ingresar es menor de 0. Línea 20 del código del método **ingresar** que se presenta más adelante.
 - Punto de parada en la instrucción donde se actualiza el saldo, sólo deberá parar la tercera vez que sea actualizado. Línea 16 del código del método **ingresar** que se presenta más adelante.

Indicaciones de entrega.

Una vez realizada la tarea elaborarás un único documento donde figuren las respuestas correspondientes. Dentro del documento irán los nombres de los alumnos que han realizado el ejercicio. El fichero se nombrará con el nombre de uno de ellos (**apellido1_apellido2_nombre.pdf**) y se subirá a la plataforma en PDF.

Método main

1	<code>public static void main(String[] args) {</code>
2	<code> // Depuracion. Se detiene siempre</code>
3	<code> CCuenta miCuenta = new CCuenta();</code>
4	<code> System.out.println("Saldo Inicial: " + miCuenta.dSaldo + " euros");</code>
5	<code> // Depuracion. Provoca parada por ingreso con cantidad menor de 0</code>
6	<code> miCuenta.ingresar(-100);</code>
7	<code> System.out.println("Saldo Inicial: " + miCuenta.dSaldo + " euros");</code>
8	<code> miCuenta.ingresar(100);</code>

9	System.out.println("Saldo tras ingreso: " + miCuenta.dSaldo + " euros");
10	miCuenta.ingresar(200);
11	System.out.println("Saldo tras ingreso: " + miCuenta.dSaldo + " euros");
12	// Depuracion. Provoca parada con codicion de tercer ingreso
13	miCuenta.ingresar(300);
14	System.out.println("Saldo tras ingreso: " + miCuenta.dSaldo + " euros");
15	miCuenta.retirar(50);
16	System.out.println("Saldo tras retirada: " + miCuenta.dSaldo + " euros");
17	}

Método ingresar

1	public int ingresar(double cantidad)
2	int iCodErr;
3	if (cantidad < 0)
4	{
5	System.out.println("No se puede ingresar una cantidad negativa");
6	iCodErr = 1;
7	}
8	else if (cantidad == -3)
9	{
10	System.out.println("Error detectable en pruebas de caja blanca");
11	iCodErr = 2;
12	}
13	else
14	{
15	// Depuración. Punto de parada. Solo en el 3 ingreso
16	dSaldo = dSaldo + cantidad;
17	iCodErr = 0;
18	}
19	// Depuración punto de parada cuando la cantidad es menor de 0
20	return iCodErr;
21	}