

# Actividad A3.3: Depurando código Java en IntelliJ IDEA

## Introducción

El objetivo de esta práctica es aprender a utilizar las herramientas de depuración en IntelliJ IDEA para resolver una serie de errores en un programa Java. Para ello, se proporcionará un código con errores lógicos, de concurrencia y de ejecución. Además, deberás usar logs y breakpoints para monitorizar el flujo del programa, entender el origen de los errores y corregirlos.

## Contexto del Programa

El código simula una aplicación de ventas que realiza operaciones como agregar productos a un carrito, calcular el precio total con descuento, y gestionar inventario. Sin embargo, hay varios errores en el código que impiden que funcione correctamente. Los alumnos deberán encontrar y corregir estos errores usando herramientas de depuración.

## Objetivos de la Práctica

1. Configurar breakpoints y logs en IntelliJ IDEA para rastrear el flujo de ejecución.
2. Utilizar el modo de depuración para inspeccionar variables y valores de retorno de métodos.
3. Identificar y resolver errores lógicos, de concurrencia y de ejecución en el código.
4. Modificar valores en tiempo real para probar hipótesis y validar soluciones.
5. Guardar los logs en un archivo para documentar el proceso de depuración.
6. Utilizar la consola de depuración para analizar la pila de llamadas en IntelliJ IDEA.

## Instrucciones

1. **Importa el Proyecto en IntelliJ IDEA:**
  - Descarga y abre el proyecto que contiene el código en IntelliJ IDEA. Para ello, haz un fork del repositorio de este [link](#) en tu equipo, luego clónalo y abre el proyecto en el IDE.
2. **Estudia el Código Inicial:**
  - Revisa los métodos en el proyecto para familiarizarte con su funcionamiento. A continuación, algunos de los métodos incluyen comentarios que indican su propósito.
3. **Configura los Logs y Breakpoints:**

- Añade breakpoints en métodos claves que gestionan, el carrito, los impuestos y los descuentos.
  - Configura logs para registrar valores importantes, como el precio del carrito, el cálculo del impuesto asociado y el descuento aplicado.
4. **Inicia la Depuración y Detecta los Errores:**
    - Usa el modo de depuración para observar los valores y flujos de ejecución. Corrige los errores conforme los encuentres.
    - Algunos errores requieren que detengas el programa en ciertos puntos para ver los valores en ese instante; otros, simplemente requieren logs para observar su comportamiento.
  5. **Investiga la posibilidad de guardar los mensajes de log en un archivo de texto de forma automática.**
  6. **Una vez realices todos los cambios, versiónalos en Git utilizando las facilidades del IDE.**

## Entrega

Debes entregar un informe en formato pdf indicando:

- Capturas de pantalla con las operaciones realizadas.
- Los fallos localizados, indicando cómo los has localizado y cómo los has corregido.
- Las pruebas que has realizado.
- Enlace a tu repositorio de github con los cambios realizados.
- Cualquier otra aclaración que consideres relevante.

Debes entregar el informe con el nombre

**Apellidos\_nombre\_ed03\_3.pdf**

La actividad será calificada como **apto** o **no apto**.