

Ejercicio 1

La reutilización de código tiene varias ventajas que la hacen atractiva en el desarrollo de software. Primero, permite ahorrar tiempo y esfuerzo, ya que al utilizar código que ya ha sido escrito y probado, se evita la necesidad de crear nuevas soluciones desde cero. Esto no sólo acelera el proceso de desarrollo, sino que también disminuye la posibilidad de cometer errores, ya que el código reutilizado ya ha sido depurado. Además, la reutilización fomenta la consistencia en el software, ya que el mismo código o funciones pueden ser utilizadas en distintas partes de la aplicación, lo que facilita su mantenimiento y mejora la calidad general del producto.

Por otro lado, la reutilización de código también tiene algunas desventajas. Puede generar problemas si el código reutilizado depende de componentes o bibliotecas externas, que podrían volverse obsoletos o incompatibles con nuevas versiones de la aplicación. Además, integrar fragmentos de código reutilizado puede hacer que el software se vuelva más complejo y difícil de entender, especialmente si el código no fue diseñado originalmente para el contexto en el que se está utilizando. También existe el riesgo de que el código reutilizado no sea el más eficiente para la situación particular, lo que podría afectar el rendimiento del sistema.

Ejercicio 2

Los depuradores son herramientas utilizadas por los desarrolladores para identificar y corregir errores en el software. Sus principales características son las siguientes:

- Permiten pausar la ejecución del programa en puntos específicos para analizar su estado (puntos de interrupción).
- Ofrecen la posibilidad de inspeccionar valores de variables en tiempo de ejecución.
- Ayudan a seguir el flujo de ejecución del código línea por línea.
- Facilitan la identificación de excepciones y errores lógicos.

Ejercicio 3

Apartado A: El operador en `if (numero % 2 = 0)` debe ser `==` para comparar, no `=` que es de asignación. Además, en el bloque `else`, el método `System.print` es incorrecto, debe ser `System.out.println`.

Apartado B: Falta un punto y coma al final de la línea `String mensaje = "Hola mundo!"`. Además, el método `System.out.println` está incompleto y carece de un argumento para imprimir.

Apartado C: El operador `==` en `int edad == 18;` es incorrecto, ya que el operador correcto para asignar un valor es `=`. También falta un punto y coma después de los bloques de impresión (`System.out.println`).

Apartado D: Falta un punto `int contador = 0`. Además `contador--` hará que `contador++` para `System.out.println(contador)`.

Apartado E: el resultado = `sumar(5)`, ya que le falta el `cesumar()`, debería `sumar(5, 3)`. En el `sumar()`, el operador en `int suma = num1 - num2` debe+Para realizar la-¿Quién eres tú?`int suma = num1 - num2`.

Apartado F: En la línea `num2 == 5`, `mi==es inco=`para un `num2`. `AnuncioString sum = num1 + num2`, la variable `sum` debería ser `int` en lugar de ``SString`, ya qu+)
`esSystem.out.println("La suma es: " suma)`, debe`System.out.println("La suma es: " + suma);`.

Apartado G: En la línea `int suma = num1 - num2`,+Pensilvania-¿Qué es para la
`System.out.println("La multiplicación es: " + multi)`, la variedad `multiNo` esta multiplicación
para que `System.out.println("La división es: " + división)`.