**S12.s1 - Clases genéricas**

**CONOCIMIENTOS PREVIOS   
  
¿Qué es un tipo genérico en Java y cuál es su propósito en la programación?**Un tipo genérico permite trabajar con tipos de datos específicos definidos por el usuario, lo que hace que el código sea más flexible y reutilizable sin perder seguridad de tipo.

**¿Cómo se declara una clase genérica en Java y cuáles son algunos ejemplos de situaciones en las que podrías utilizarlas?**  
Se declara utilizando el operador <>, donde el tipo genérico se coloca entre los corchetes. Ejemplo:

**public class Caja<T> {**

**private T contenido;**

**public T obtenerContenido() { return contenido; }**

**}**

**¿Qué ventajas ofrece el uso de clases genéricas en términos de reutilización de código y seguridad de tipos en comparación con enfoques no genéricos?**

* Reutilización de código: Escribes una clase o método una vez y puedes usarla con cualquier tipo de datos.
* Seguridad de tipos: El compilador verifica los tipos en tiempo de compilación, evitando errores en tiempo de ejecución.
* Mantenimiento simplificado: Al reducir el código repetido, es más fácil de mantener y modificar.
* Mayor flexibilidad: Permite trabajar con diferentes tipos de datos de forma segura y eficiente, sin duplicar código. **Utilidad**

**¿Cómo la programación en Java permite la creación de código que puede trabajar con diferentes tipos de datos de manera más flexible y dinámica?**Java permite crear código flexible y dinámico mediante genéricos y polimorfismo, lo que facilita trabajar con diferentes tipos de datos sin perder la seguridad de tipos.

**¿Cuáles son las ventajas de escribir código que sea más reutilizable y adaptable a diferentes situaciones de programación?**Reutilización de código mejora el mantenimiento, reduce errores, aumenta la eficiencia, y reduce costos de desarrollo y mantenimiento.

**¿En qué medida la programación en Java puede contribuir a reducir la redundancia y mejorar la seguridad de los programas al permitir un manejo más eficiente de los tipos de datos?**  
Los genéricos reducen la redundancia en el código, mejoran la seguridad de tipos y evitan errores de conversión de tipos, aumentando la fiabilidad y mantenibilidad del programa.

La utilización de conceptos como las clases genéricas en Java permite a los programadores escribir código que puede trabajar con una variedad de tipos de datos, lo que promueve la reutilización de código y mejora la seguridad de tipo en sus aplicaciones. **Introducción**

Las clases genéricas en Java son un mecanismo que permite crear clases, interfaces y métodos que operan sobre tipos de datos de manera parametrizada. Esto significa que puedes definir una clase o método que puede trabajar con diferentes tipos de datos sin tener que escribir una implementación específica para cada tipo. Esto mejora la reutilización de código y la seguridad de tipo.  
Aquí, T es un tipo de dato genérico que se puede utilizar en la clase. Puedes utilizar T en lugar de tipos de datos específicos, y Java sustituirá T con el tipo de dato real en tiempo de compilación.  
Por ejemplo, puedes crear una instancia de MiClaseGenerica con un tipo específico, como Integer: Las clases genéricas son útiles para escribir código que es más genérico y flexible. Puedes crear clases genéricas para contenedores de datos, algoritmos y estructuras de datos, entre otros.  
**Cierre**

**¿Cómo pueden las clases genéricas en Java simplificar la escritura de código que funciona con diferentes tipos de datos?**Simplificación de código: Las clases genéricas permiten escribir un solo bloque de código que trabaja con diferentes tipos de datos, eliminando la necesidad de duplicar código para cada tipo específico, lo que hace el código más flexible y fácil de mantener.

**¿Qué beneficios ofrece el uso de clases genéricas en términos de reutilización de código?**Reutilización de código: Usar clases genéricas reduce la duplicación y facilita la adaptabilidad del código para diferentes tipos de datos, lo que mejora la eficiencia y simplifica el mantenimiento**.**

**¿Por qué es importante considerar la seguridad de tipos al diseñar programas en Java y cómo las clases genéricas contribuyen a esta seguridad?**Seguridad de tipos: Las clases genéricas garantizan que los tipos sean correctos en tiempo de compilación, eliminando la necesidad de hacer conversiones de tipo manuales y evitando errores en tiempo de ejecución, lo que mejora la seguridad del programa.