

Factory Method

El Factory Method es un patrón de diseño creacional que provee una interfaz para la creación de objetos en una superclase, permitiendo que las subclasses decidan qué clase instanciar. Este patrón es especialmente útil cuando el tipo exacto de objetos a crear puede variar o necesita definirse en tiempo de ejecución. Esto permite una mayor flexibilidad y extensibilidad en la creación de objetos.

¿Qué es Factory Method?

El Factory Method es un patrón de diseño usado en el desarrollo de software. Proporciona una interfaz para crear objetos en una superclase, mientras permite que las subclasses especifiquen el tipo de objeto que será instanciado.

- Este patrón simplifica el proceso de creación de objetos al delegarlo a un método especializado, promoviendo una conexión más flexible entre el creador y los objetos que produce.
- Mejora la flexibilidad, extensibilidad y mantenibilidad del código, permitiendo que las subclasses implementen su propia versión del método de fábrica para crear tipos concretos de objetos.

¿Cuándo usar Factory Method?

Este patrón es especialmente útil cuando:

- La creación de objetos es compleja o puede variar bajo ciertas condiciones. Usar Factory Method permite mantener simple el código del cliente y promueve la reutilización.
- Se desea crear objetos a través de una interfaz o clase abstracta, ocultando los detalles de implementación concreta. Esto reduce las dependencias y facilita la modificación o expansión del sistema.
- Se anticipa la necesidad de múltiples versiones de un producto, o se espera introducir nuevos tipos en el futuro. El patrón permite manejar estas variaciones definiendo métodos de fábrica específicos.
- La lógica de configuración para crear un objeto puede encapsularse dentro del método de fábrica, permitiendo a los clientes personalizar el proceso de creación mediante parámetros u opciones.

Componentes del Factory Method

1. Creador:

Es una clase abstracta o una interfaz que declara el método de fábrica. Esta clase puede incluir lógica que utiliza el objeto creado, delegando la creación concreta a su subclase.

2. Creador Concreto:

Es una única subclase del Creador que implementa el método de fábrica, especificando qué tipo de objeto concreto se debe crear. A diferencia del patrón Abstract Factory, en Factory Method generalmente hay una sola subclase concreta que hereda del creador.

3. Producto:

Es una interfaz o clase abstracta que define el tipo de objeto que el método de fábrica debe crear. Proporciona una interfaz común para todos los productos que pueden ser generados.

4. Producto Concreto:

Es la implementación concreta del Producto. Es el tipo de objeto que retorna el método de fábrica definido en el Creador Concreto.