PATRONES DE DISEÑO

DEFINICION

Facilitan la creación de objetos de manera adecuada para la situación.

EJEMPLOS DE USO

Singleton: Recursos únicos como configuraciones o conexiones a bases de datos.

Abstract Factory: Sistemas que requieren la creación de productos relacionados, como interfaces gráficas.

Builder: Interfaces de usuario complejas que requieren un proceso de construcción personalizado.

Tipos

SINGLETON

Descripción: Asegura una única instancia de una clase

Uso: Proporciona un punto de acceso global.

FACTORY METHOD

Descripción: Define una interfaz para crear un objeto. Uso: Permite a las subclases decidir qué clase instanciar.

Abstract Factory

Descripción: Proporciona una interfaz para crear familias de objetos relacionados.

Uso: Crea productos sin especificar sus clases concretas.

Builder

Descripción: Separa la construcción de un objeto complejo de su representación. Uso: Permite crear diferentes representaciones de un

objeto. Prototype

Descripción: Permite crear nuevos objetos clonando una instancia existente.

Uso: Evita la creación de objetos a través de constructores

ODJETIVO

Controlar la creación de objetos y proporcionar flexibilidad

VENTAJAS

Flexibilidad: Facilitan cambios en la creación de objetos.

Reutilización: Promueven la reutilización de código y la modularidad.

Mantenimiento: Mejoran el mantenimiento y la escalabilidad del sistema.