



Design Patterns

REVIEW



INTRODUÇÃO

O Design Pattern Factory Method atua permitindo adiar a instanciação de um objeto para as subclasses.

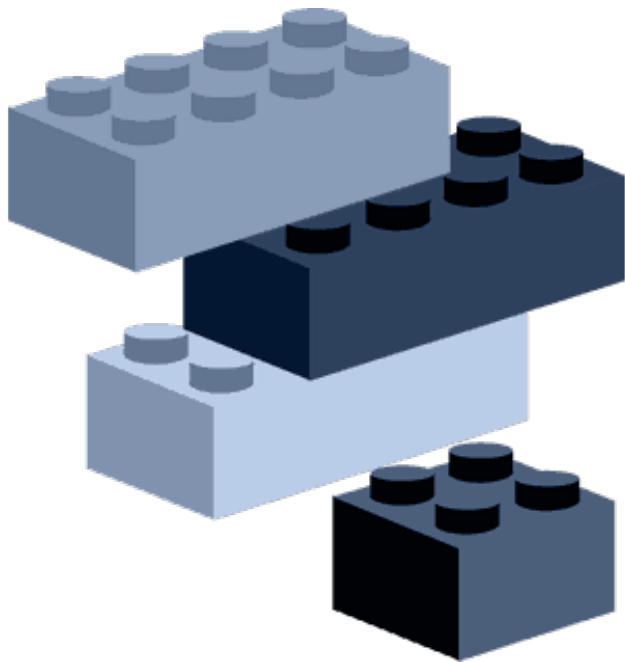
Em outras palavras, tem-se uma classe com um método que cria um objeto e força-se as subclasses a implementá-lo.



CARACTERÍSTICAS

- É um padrão de projeto de criação, ou seja, lida com a criação de objetos;
- Oculta a lógica de instanciação do código cliente, sendo responsável por instanciar as classes desejadas;
- É obtido através de herança, podendo ser criado ou sobrescrito por subclasses;
- Dá flexibilidade ao código cliente, permitindo a criação de novas factories sem alterar as já existentes, garantindo, assim, o Open/Closed Principle;
- Pode utilizar parâmetros para determinar o tipo dos objetos a serem criados ou parâmetros a serem enviados aos objetos.





ABSTRACT FACTORY

**APRESENTAÇÃO E PRÁTICA
DO PP**

INTRODUÇÃO

Este padrão permite a criação de famílias de objetos relacionados ou dependentes por meio de uma única interface e sem que a classe concreta seja especificada.

O objetivo é isolar a criação de objetos de seu uso e criar famílias de objetos relacionados sem ter que depender de suas classes concretas.



CARACTERÍSTICAS

- É um padrão de projeto de criação, ou seja, lida com a criação de objetos;
- É uma fábrica, assim como o Factory Method, e geralmente é composto por múltiplos Factory Methods;
- Visa agrupar famílias de produtos compatíveis, criando uma fábrica concreta por grupo de objetos compatíveis;
- Separa código que cria do que usa;
- Permite a fácil implementação de novas famílias de objetos;
- Toda a programação fica focada nas interfaces, ou classes abstratas.



QUANDO USAR?

Um sistema deve ser independente de como seus produtos são criados, compostos ou representados;

Um sistema deve ser configurado como um produto de uma família de múltiplos produtos;

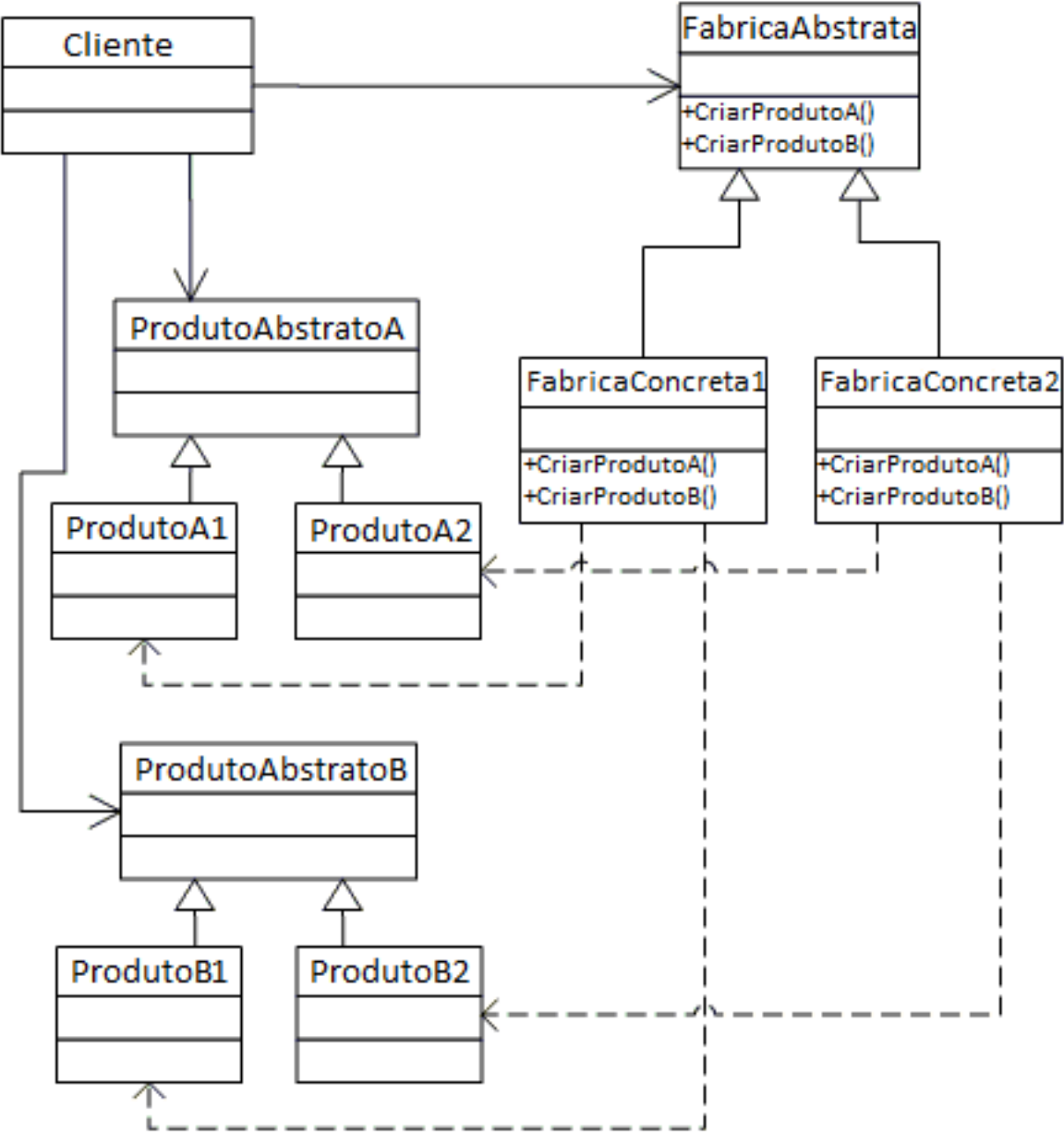
Uma família de objetos-produto for projetada para ser usada em conjunto e você necessita garantir esta restrição;

Você quer fornecer uma biblioteca de classes de produtos e quer revelar somente suas interfaces, não suas implementações.



ESTRUTURA





ESTRUTURA

ProdutoAbstrato

Interface ou classe abstrata que declara os componentes do produto.

ProdutoA, B...

Estende ou implementa ProdutoAbstrato.

FábricaAbstrata

Classe abstrata ou concreta que declara os métodos de criação de objetos do tipo ProdutoAbstrato.



ESTRUTURA

FábricaConcreta

Estende ou implementa a FábricaAbstrata.

Vejamos o primeiro exemplo:



CONSEQUÊNCIAS

Positivas:

- Produtos sempre serão compatíveis entre si;
- Aplicação clara do Open/Closed Principle;
- Aplicação do Single Responsibility Principle.



CONSEQUÊNCIAS

Negativas:

🚫 Muitas classes e maior complexidade de código.





Obrigado!

Professor Gustavo Dias
luizdias@univas.edu.br