



Padrão Factory



**Certified
Developer**
The Ultimate Tech Degree

DigitalHouse >
Coding School



Temas

1

Definição

2

UML

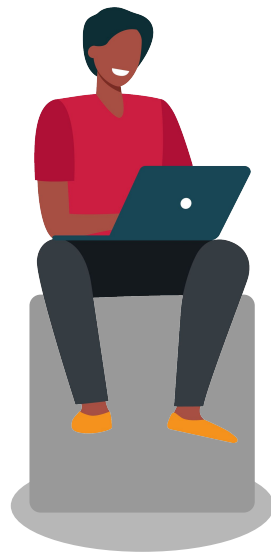


1 | Definição



Padrão Factory

- O padrão de projeto **Factory** é um dos principais padrões de projeto e mais utilizados nas linguagens de programação mais atuais.
- O padrão **Factory** tem duas variações:
 - **Factory Method**
 - **Abstract Factory**

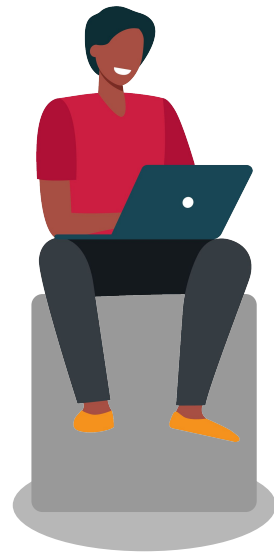




Factory Method

Definição

O padrão **Factory Method** define uma interface para criar um objeto, mas permite que as subclasses decidam qual classe instanciar.

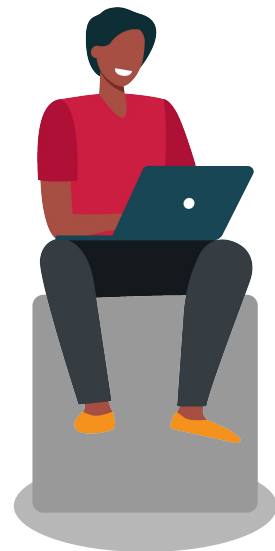




Abstract Factory

Definição

O **Abstract Factory** é um padrão que fornece uma interface para criar famílias de objetos dependentes ou relacionados sem especificar suas classes concretas.

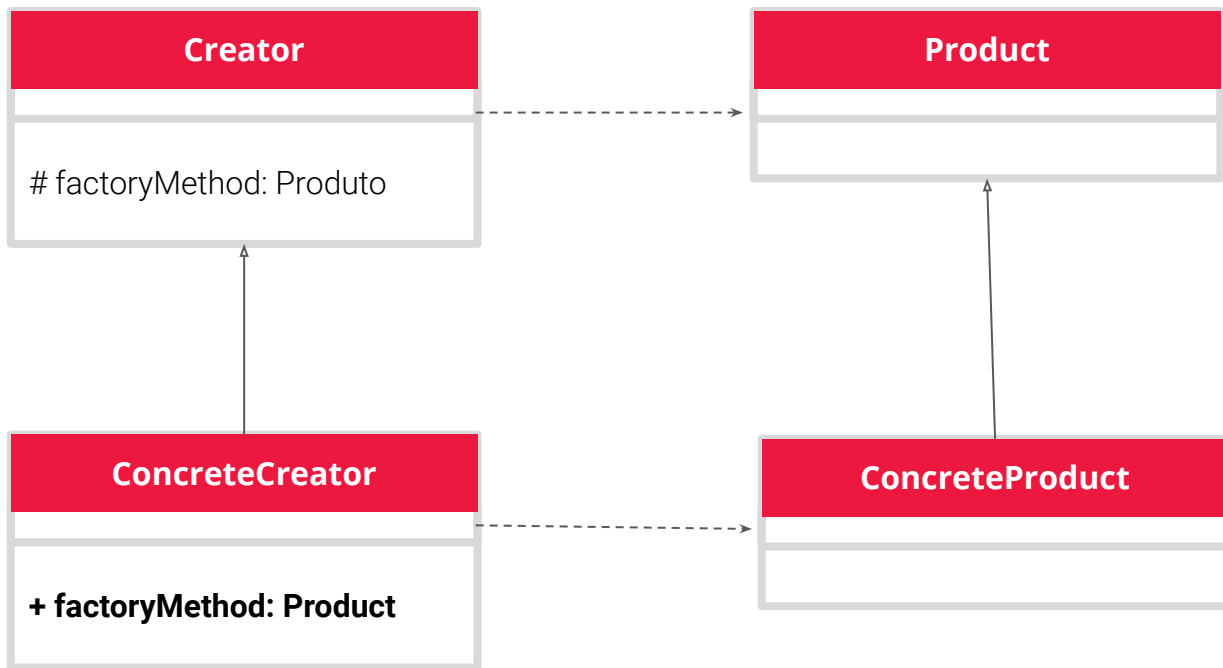


2

UML



Factory Method





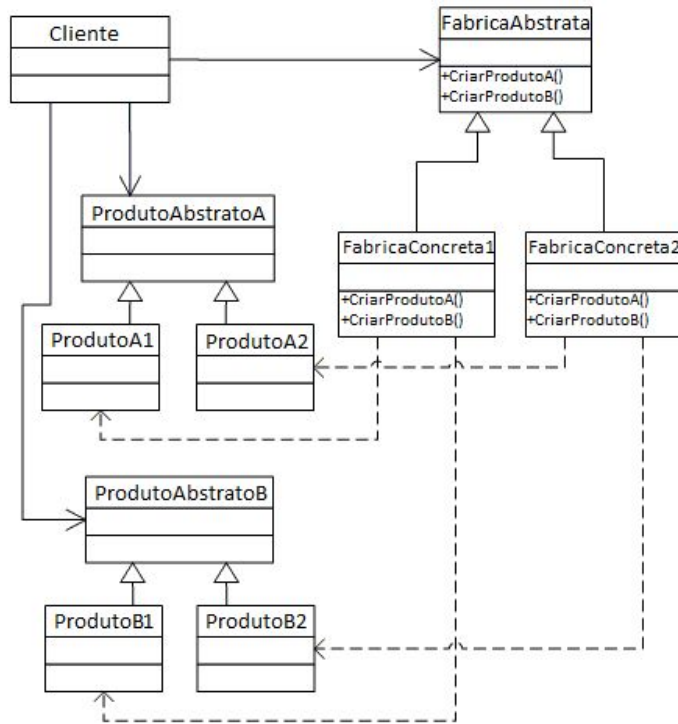
Factory Method

- **Creator(Criador abstrato)** – declara o *factory method* (método de fabricação) que retorna o objeto da classe Product (produto). Este elemento também pode definir uma implementação básica que retorna um objeto de uma classe ConcreteProduct (produto concreto) básica;
- **ConcreteCreator(Criador concreto)** – sobrescreve o *factory method* e retorna um objeto da classe ConcreteProduct;
- **Product(Produto abstrato)** – define uma interface para os objectos criados pelo *factory method*;
- **ConcreteProduct(Produto concreto)** – uma implementação para a interface Product.





Abstract Factory





Abstract Factory

- **Fábrica Abstrata** – Seu objetivo é declarar métodos de criação de objetos do tipo *ProdutoAbstrato*, que são implementados por uma classe do tipo *FabricaConcreta*, que estende ou implementa a *FabricaAbstrata*.
- **Produto Abstrato** – declara os métodos que são implementados por classes do tipo *ProdutoConcreto*. *FabricaConcreta* cria internamente um objeto do tipo *ProdutoConcreto*, mas esse objeto é retornado como um *ProdutoAbstrato*.
- **Fábrica Concreta** – implementa os métodos declarados em *FabricaAbstrata*, criando um objeto do tipo *ProdutoConcreto* e retornando-o como um *ProdutoAbstrato*.
- **Produto concreto** – implementada os métodos declarados em *ProdutoAbstrato*. Essa é a classe que faz uma instância concreta ser criada.





Conclusão

A proposta do padrão **Factory** é de criar objetos, por isso este é considerado um padrão *criacional*. Basicamente, a lógica criacional é encapsulada dentro do *factory* e, ou é fornecido um método que retorna um novo objeto criado (Padrão **Factory Method**) ou delega-se a criação do objeto para uma subclasse (Padrão **Abstract Factory**).

DigitalHouse>
Coding School