



Geek University

Evolua seu lado geek!

www.geekuniversity.com.br



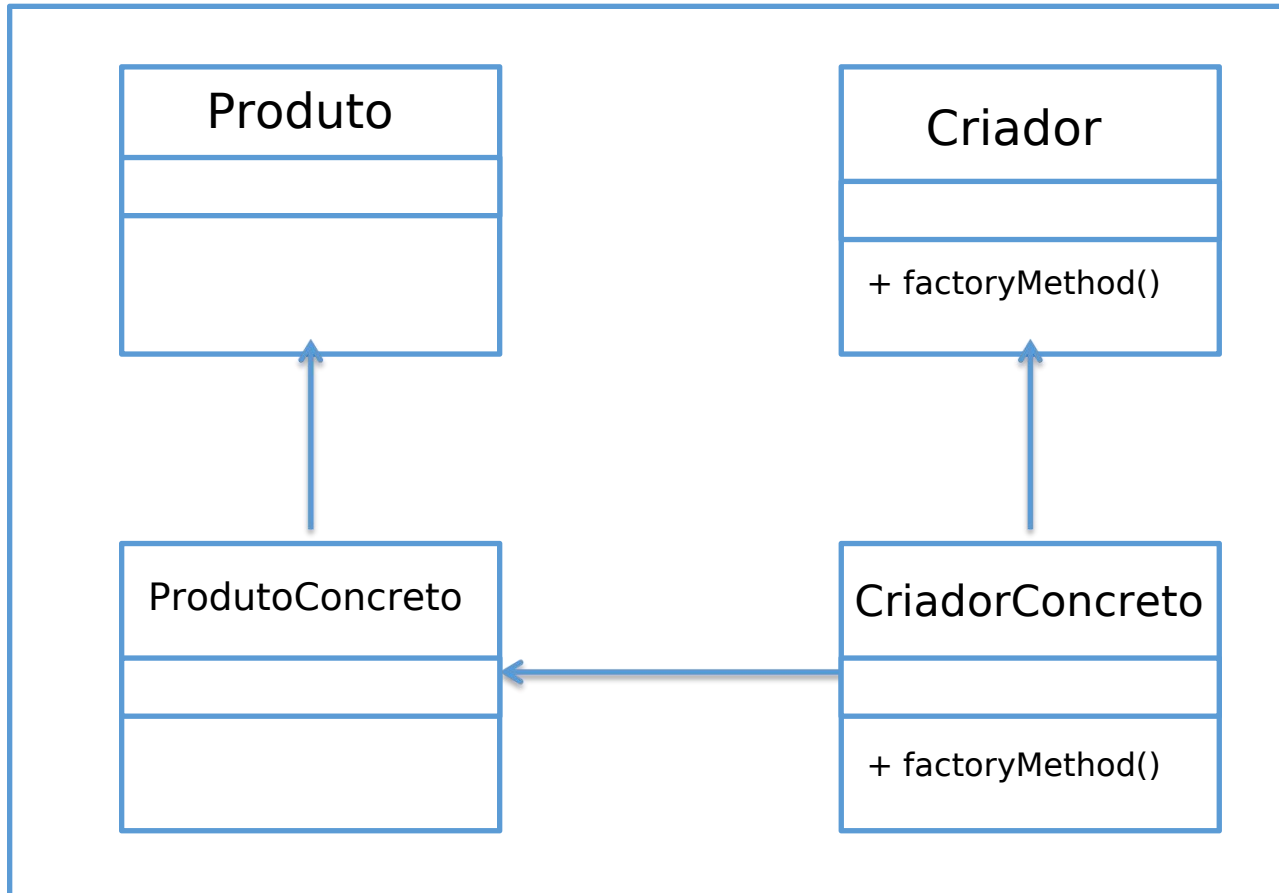
Padrão Factory Method





Padrão Factory Method

O diagrama abaixo nos ajuda a entender o funcionamento do Factory Method

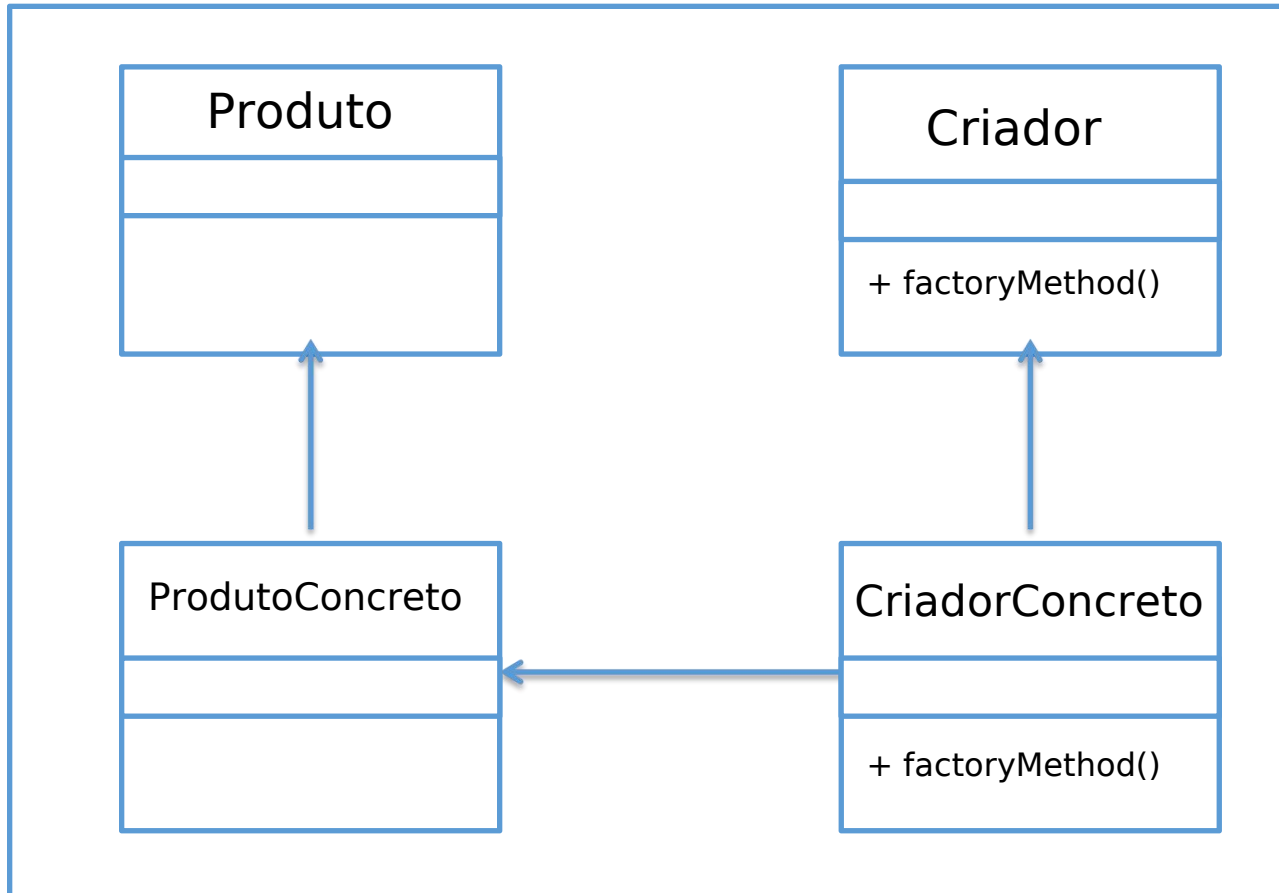


- Definimos uma interface para criar objetos, mas, em vez de a fábrica ser responsável pela criação dos objetos, a responsabilidade é passada para a subclasse, que decidirá a classe a ser instanciada.
- A criação do Factory Method não é feita por instanciação, mas sim por herança;
- O Factory Model deixa o design mais personalizável. Ele pode devolver a mesma instância ou subclasse no lugar de um objeto de determinado tipo, como ocorre no Simple Factory.



Padrão Factory Method

O diagrama abaixo nos ajuda a entender o funcionamento do Factory Method



No diagrama ao lado, temos uma classe abstrata Criador que contém `factoryMethod()`

O método `factoryMethod()` é responsável por criar objetos de determinado tipo.

A classe `CriadorConcreto` implementa o método `factoryMethod()` da classe abstrata `Criador`.

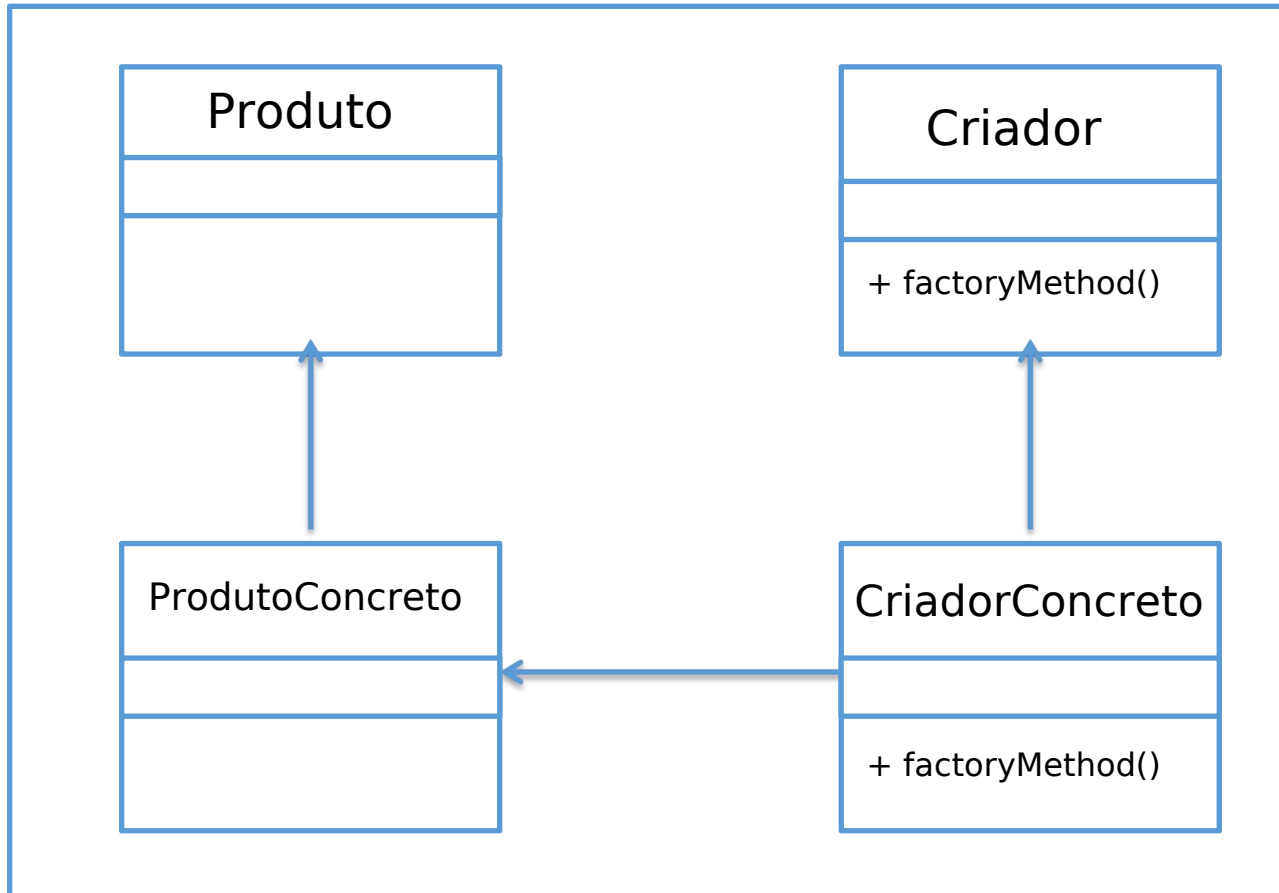
`CriadorConcreto` cria o `ProdutoConcreto` e garante que o objeto criado implemente a classe `Produto` e forneça uma implementação para todos os seus métodos.

Em resumo, o método `factoryMethod()` da classe `Criador/CriadorConcreto` decide qual subclasse de `Produto` deve ser criada. Desta forma o padrão `Factory Method` define uma interface para criar um objeto, mas adia a decisão sobre qual classe deve ser instanciada, passando-a para as suas subclasses.



Padrão Factory Method

O diagrama abaixo nos ajuda a entender o funcionamento do Factory Method



No diagrama ao lado, temos uma classe abstrata Criador que contém `factoryMethod()`

O método `factoryMethod()` é responsável por criar objetos de determinado tipo.

A classe `CriadorConcreto` implementa o método `factoryMethod()` da classe abstrata `Criador`.

`CriadorConcreto` cria o `ProdutoConcreto` e garante que o objeto criado implemente a classe `Produto` e forneça uma implementação para todos os seus métodos.

Em resumo, o método `factoryMethod()` da classe `Criador/CriadorConcreto` decide qual subclasse de `Produto` deve ser criada. Desta forma o padrão `Factory Method` define uma interface para criar um objeto, mas adia a decisão sobre qual classe deve ser instanciada, passando-a para as suas subclasses.

Na próxima aula iremos criar uma implementação deste padrão...



Geek University

Evolua seu lado geek!

www.geekuniversity.com.br