



Geek University

Evolua seu lado geek!

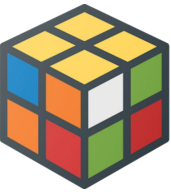
www.geekuniversity.com.br

Introdução ao padrão de projeto Template Method



pythonTM

Design Patterns



Introdução ao padrão de projeto Template Method

Aprendemos em aulas anteriores que os padrões de projeto comportamentais têm como foco as responsabilidades de um objeto, pois eles lidam com a interação entre objetos para alcançar funcionalidades mais complexas.

O padrão **Template Method** é um padrão de projeto comportamental que define o esqueleto do programa ou um algoritmo em um método chamado *Método Template*.



Introdução ao padrão de projeto Template Method

Você pode, por exemplo, definir os passos para preparar uma bebida como um algoritmo em um Método Template.

O padrão Template Method também ajuda a redefinir ou personalizar os passos do algoritmo adiando a implementação de alguns desses passos para as subclasses. Ou seja, as subclasses podem redefinir o seu próprio comportamento.

Desta forma uma subclasse poderia fazer uso do Método Template para preparar uma bebida e preparar chá enquanto outra subclasse poderia usar o mesmo Método Template para preparar café já que ambas são bebidas.



Introdução ao padrão de projeto Template Method

Vale destacar que a alteração dos passos nas subclasses não exerce impacto na estrutura do algoritmo original.

Desta forma, o recurso das subclasses de poder sobrescrever no padrão Template Method permite a criação de diferentes comportamentos ou algoritmos.

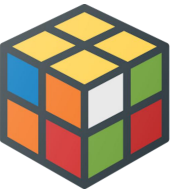


Introdução ao padrão de projeto Template Method

No contexto de desenvolvimento de software, uma classe abstrata é usada para definir os passos do algoritmo. Esses passos são conhecidos como operações primitivas no contexto do padrão Template Method.

Estas operações são definidas como métodos abstratos e o Método Template define o algoritmo.

Uma classe concreta que for subclasse desta classe abstrada implementa os passos do algoritmo específicos para a subclasse.



Introdução ao padrão de projeto Template Method

O padrão Template Method é usado nos seguintes casos:

- Quando vários algoritmos ou classes implementam uma lógica semelhante ou idêntica;
- Quando a implementação dos algoritmos em subclasses ajuda a reduzir a duplicação de código;
- Quando vários algoritmos podem ser definidos ao deixar que as subclasses implementem o comportamento usando o recurso de sobrescrita.



Introdução ao padrão de projeto Template Method

Voltando ao nosso exemplo da bebida

Pense no que geralmente fazemos quando preparamos chá ou café:



Introdução ao padrão de projeto Template Method

Voltando ao nosso exemplo da bebida

Pense no que geralmente fazemos quando preparamos chá ou café:

Preparando Café	Preparando Chá
1. Ferva a água	1. Ferva a água
2. Passe a água fervente pelo pó de café	2. Coloque o saquido de chá
3. Coloque o café em uma xícara	3. Coloque o chá em uma xícara
4. Adicione açúcar ou leite na xícara	4. Adicione limão ao chá
5. Misture, e o café estará pronto	5. Misture, e o chá estará pronto



Introdução ao padrão de projeto Template Method

Voltando ao nosso exemplo da bebida

Pense no que geralmente fazemos quando preparamos chá ou café:

Preparando Café	Preparando Chá
1. Ferva a água	1. Ferva a água
2. Passe a água fervente pelo pó de café	2. Coloque o saquido de chá
3. Coloque o café em uma xícara	3. Coloque o chá em uma xícara
4. Adicione açúcar ou leite na xícara	4. Adicione limão ao chá
5. Misture, e o café estará pronto	5. Misture, e o chá estará pronto

Se analisarmos as preparações acima, iremos perceber que os dois procedimentos são mais ou menos iguais.

Neste caso, podemos usar o padrão Template Method de modo eficiente.



Introdução ao padrão de projeto Template Method

Voltando ao nosso exemplo da bebida

Pense no que geralmente fazemos quando preparamos chá ou café:

Preparando Café	Preparando Chá
1. Ferva a água	1. Ferva a água
2. Passe a água fervente pelo pó de café	2. Coloque o saquido de chá
3. Coloque o café em uma xícara	3. Coloque o chá em uma xícara
4. Adicione açúcar ou leite na xícara	4. Adicione limão ao chá
5. Misture, e o café estará pronto	5. Misture, e o chá estará pronto

Para isso podemos definir uma classe **Bebida** que tem métodos abstratos comuns para preparação de chá e café, por exemplo, *ferver_agua()*.

Também definimos o Método Template *preparar()* que acionará a sequência de passos na preparação da bebida, ou seja, o algoritmo.

Após isso criamos as classes concretas **PreparaCafe** e **PreparaCha** para definirem os passos personalizados para atingir os objetivos que são preparar café ou chá.



Introdução ao padrão de projeto Template Method

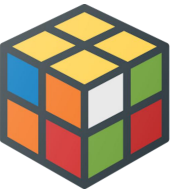
Outro exemplo simples é o compilador usado pelas linguagens de programação.

Um compilador executa essencialmente duas tarefas:

- Reúne o código fonte;
- Compila gerando um objeto-alvo;

Desta forma se precisarmos definir um compilador para dispositivos iOS ou Android, podemos implementar isso com a ajuda do padrão Template Method.

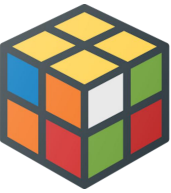
É assim que o padrão Template Method evita duplicação de código.



Introdução ao padrão de projeto Template Method

Os principais objetivos do padrão Template Method são:

- Definir o esqueleto de um algoritmo com operações primitivas;
- Redefinir determinadas operações na subclasse sem alterar a estrutura do algoritmo;
- Reutilizar o código e evitar esforços duplicados;
- Tirar proveito de interfaces ou implementações comuns;



Introdução ao padrão de projeto Template Method

Os principais objetivos do padrão Template Method são:

- Definir o esqueleto de um algoritmo com operações primitivas;
- Redefinir determinadas operações na subclasse sem alterar a estrutura do algoritmo;
- Reutilizar o código e evitar esforços duplicados;
- Tirar proveito de interfaces ou implementações comuns;

Vamos ao código fazer uma implementação simples deste padrão...



Geek University

Evolua seu lado geek!

www.geekuniversity.com.br