

Template Method

Análise e Desenvolvimento de Sistemas - 6º Período

Template Method

- Existem alguns cenários onde temos várias subclasses, cada uma com sua responsabilidade. Mas por serem relativamente parecidas, podem conter código repetido
- Define um template de um algoritmo na superclasse e deixa as subclasses sobrescreverem comportamentos (métodos) específicos sem modificar sua estrutura

Template Method

- O Template suporta a criação de uma super classe onde os comportamentos dos objetos serão divididos em métodos diferentes, ou seja, será executado em etapas.
- Essas etapas serão colocadas dentro de um **único método**, chamado de método padrão. Essas etapas podem ser métodos abstratos para que suas classes filhas façam suas implementações ou implementações já pré-definidas.

Template Method

- Isso nos permite ter execuções personalizadas desse métodos padrão, já que cada subclasse irá implementar as etapas de sua maneira
- Com isso o código repetido será removido das subclasse
- E por fim, temos um "template" em formato de classe para seguir
- Observação: O Template Method é um padrão baseado apenas em herança

Template Method

- Portanto temos dois elementos importantes:
Definição dos métodos abstratos e o método padrão
- Ambos são obrigatórios para a construção do template
- No entanto, podemos definir métodos abstratos onde as subclasses podem ou não implementar. E ainda colocar esses métodos em qualquer etapa de execução dentro do método padrão. Esse método opcional é chamado de **hook (gancho)**

Template Method - Quando Usar

- Usamos quando queremos que o client forneça suas implementações de partes do algoritmo
- Também usamos quando temos classes diferentes com implementações parecidas. Com o template removemos essa duplicidade

