

10 - Classes

2025-10-03 17:13 pm

- Por convenção classes devem começar com uma lista de variáveis
 - Nesta ordem:
 - Constantes públicas
 - Constantes privadas
 - Variável de instância privada.
 - Seguido de funções públicas
 - Funções privadas ficam listadas conforme são usadas
- Expor variáveis para testes é uma má ideia
 - Devem ser testados, mas por meio de encapsulação.
- Classes devem ser pequenas!
 - Medido em "responsabilidades"
 - Nome da classe deve descrever responsabilidades
 - Deve ser conciso
 - "Processor", "Manager" ou "Super" são maus sinais
 - Descrição da classe deve ser breve e sem "e", "se", "mas" ou "ou"
 - **Single Responsibility Principle (SRP)**
 - Uma classe ou um módulo deve ter uma e apenas uma razão para mudar.
 - Uma única responsabilidade (e.g. não juntar "gerir versão" com "mostrar dashboard")
 - É melhor ter muitas classes pequenas que poucas classes imensas
 - **Coesão**
 - Classe deve ter poucas variáveis
 - Quanto mais variáveis um método manipula, maior a coesão do método à classe
 - Coesão máxima: todas as variáveis são usadas em todas os métodos
 - Queremos coesão alta
 - Significa que fazem parte da mesma lógica
 - Se tem algum conjunto de variáveis que só é usado por alguns métodos, costuma significar que aquilo podia se tornar outra classe separada
 - Quando classes têm baixa coesão, convém dividi-las em classes menores.
 - Organizar para mudanças
 - Mudança é contínua
 - Escrever código que requeira poucas alterações e tenha poucos pontos de falha
 - **Open-Closed principle (OCP)**
 - Classes devem ser tais que permitem extensão mas não modificação.
 - Novos features não devem requerer mudar nada, só adicionar.
 - **Dependency Inversion Principle (DIP)**
 - Classes devem depender de abstrações, não de detalhes concretos.