

Orientação a Objeto - Composição de Objetos

Objetivos da seção

- Aprender as vantagens da composição de objetos sobre a herança

Composição e Herança

- Composição e herança são dois mecanismos para reutilizar funcionalidade
- Alguns anos atrás (e na cabeça de alguns programadores ainda!), a herança era considerada a ferramenta básica de extensão e reutilização de funcionalidade
- A composição estende uma classe pela delegação de trabalho para outro objeto
- A herança estende atributos e métodos de uma classe
- Hoje, considera-se que a composição é muito superior à herança na maioria dos casos
 - A herança deve ser utilizada em alguns contextos

Composição versus Herança: Um Exemplo

- O exemplo de refatoramento visto na seção anterior já inclui um excelente exemplo do uso da composição em vez da herança
- As grandes vantagens da composição são:
 - A herança gera um acoplamento muito forte. A composição pode reduzir acoplamento, principalmente se interfaces forem usadas
 - A composição pode ser mudada em tempo de execução
- A grande vantagem da herança é que ela gera código mais simples de ler e entender

- A composição usa muitos objetos pequenos e só sabemos quem vai ser composto com quem em tempo de execução
- Isso é mais difícil de entender

oo-10 [programa anterior](#) [próxima](#)