

---

# **Introdução à Padrões de Projeto**

# O que é um padrão?

---

“Cada padrão descreve um problema que ocorre repetidas vezes em nosso ambiente, e então descreve o núcleo da sua solução para aquele problema, de tal maneira que seja possível usar essa solução milhões de vezes sem nunca fazê-la da mesma forma duas vezes”.

[Christopher Alexander sobre padrões em arquitetura de construções]

# O que é um padrão?

---

- São comuns em várias áreas
- Padrões de Projeto de Software
  - Inspirados em “**A Pattern Language**” de Christopher Alexander, sobre padrões de arquitetura de cidades, casas e prédios.
  - ***Design Patterns: Elements of Reusable Object-Oriented Software.***
    - Catálogo publicado em 1995.
    - Erich Gamma, John Vlissides, Ralph Jonhson e Richard Helm, conhecidos como “The Gang of Four” (GoF).
    - 23 padrões de projeto.

# O que é um padrão?

---

“Os padrões de projeto são descrições de objetos que se comunicam e classes que são customizadas para resolver um problema de projeto genérico em um contexto específico”.

[Gamma, Helm, Vlissides & Johnson, sobre padrões de projeto em software]

# Por que aprender padrões de projeto?

---

- Aprender com a experiência dos outros
  - Identificar problemas comuns de projeto de software e utilizar soluções testadas e bem documentadas.
  - Aprender os padrões ajudam um novato a agir mais como um experiente.
- Produzir bons projetos orientados a objetos
  - Os padrões de projeto normalmente utilizam boas práticas de OO.
  - Os padrões utilizam eficientemente polimorfismo, herança e composição.

# Por que aprender padrões de projeto?

---

- Vocabulário comum
  - Utilizar soluções que têm um nome facilita a comunicação.
  - Permitir que se fale em um nível mais alto de abstração.
- Ajuda na documentação
  - Utilizar soluções que têm um nome facilita a documentação.
  - Conhecer os padrões de projeto torna mais fácil a compreensão de sistemas existentes.
- Ajudam a converter um modelo de análise em um modelo de projeto

# Elementos essenciais de um padrão

---

- Nome
  - Procura descrever o problema, a solução e as conseqüências em uma ou duas palavras.
- Problema
  - Quando aplicar o padrão e em que condições.
- Solução
  - Descrição abstrata de um problema e como usar os elementos disponíveis (classes e objetos) para solucioná-lo.
- Conseqüências
  - Custos e benefícios de se aplicar o padrão.
  - Impacto na flexibilidade, reusabilidade e eficiência do sistema.

# Alguns livros sobre padrões de software

CATEGORIA DO PADRÃO	TÍTULO	AUTORES/ EDITORES
Análise OO	Analysis Patterns: Reusable Object Models	Martin Fowler
Projeto	Core J2EE™ Patterns: Best Practices and Design Strategies	Deepak Alur John Crupi Dan Malks
<b>Projeto</b>	<b>Design Patterns: Elements of Reusable Object-Oriented Software</b>	<b>Gamma et al.</b>
Projeto	AntiPatterns: Refactoring Software, Architectures, and Projects in Crisis	William J. Brown et al.
Projeto	Design Patterns Java™ Workbook	Steven John Metsker



# Padrões de Projeto da GoF

---

- Catálogo de 23 padrões
- Não apresenta nenhum padrão para um domínio de aplicação específico.

# Padrões de Projeto da GoF

---

1. Factory Method
2. Abstract Factory
3. Builder
4. Prototype
5. Singleton
6. Adapter
7. Bridge
8. Composite
9. Decorator
10. Facade
11. Flyweight
12. Proxy
13. Interpreter
14. Template Method
15. Chain of Responsibility
16. Command
17. Iterator
18. Mediator
19. Memento
20. Observer
21. State
22. Strategy
23. Visitor