**Padrões de Projetos**

Os padrões de projeto, também conhecidos como design patterns, são soluções reutilizáveis para problemas comuns de desenvolvimento de software. Eles surgiram na década de 1990, quando um grupo de desenvolvedores identificou que muitos projetos de software apresentavam problemas semelhantes, que podiam ser resolvidos por soluções também semelhantes.

Os padrões de projeto podem ser divididos em três categorias:

1. Padrões de criação: são padrões que lidam com a criação de objetos, visando garantir a flexibilidade e reutilização do código. Alguns exemplos de padrões de criação são: Factory Method, Builder e Singleton.
2. Padrões de estrutura: são padrões que lidam com a organização de objetos em estruturas maiores, buscando simplificar a comunicação entre objetos e reduzir o acoplamento entre eles. Alguns exemplos de padrões de estrutura são: Adapter, Facade e Composite.
3. Padrões de comportamento: são padrões que lidam com a comunicação entre objetos, buscando definir o comportamento esperado em situações específicas. Alguns exemplos de padrões de comportamento são: Observer, Command e Strategy.

Ao utilizar padrões de projeto, é possível aumentar a qualidade do código, tornando-o mais legível, flexível e de fácil manutenção.

Caso você queira aprofundar os conhecimentos em padrões de projetos em Java, recomendamos os nossos seguintes cursos:

1. [Design Patterns em Java I: boas práticas de programação](https://cursos.alura.com.br/course/introducao-design-patterns-java)
2. [Design Patterns em Java II: avançando nas boas práticas de programação](https://cursos.alura.com.br/course/avancando-design-patterns-java)