Padrões de projeto

Padrões de projeto são soluções documentadas e testadas para problemas comuns de implementação de código de programação. Essas soluções são aprovadas por grande parte dos que as conhecem e são aplicadas nos mais variados projetos e sistemas, por isso são padronizadas. Apesar de existir tantos padrões, isso vai além de ser algo tão especifico pois a ideia é pensar em uma solução para um determinado problema, mas já temos algo com eficácia comprovada diante de um empecilho então não é necessário ficar inventando rodas para solucionar isso.  
  
Dentro do desenvolvimento de software existem 3 tipos de padrões de projeto e dentro dele há vários padrões estabelecidos. São eles:

De comportamento:

Strategy – Permite a operação de formulas distintas para um objetivo de forma independente mantendo a coesão e organização do código.

Observer – Trabalha com uma forma de comunicação entre dependências que necessitam ser executadas após uma alteração na classe informante. Ajuda a eliminar grande acoplamento.

De estrutura:

Decorator – É uma forma de compor (ou decorar como o nome diz) um objeto com outro objeto que pode compor com outro objeto e assim por diante para que haja possibilidade de atribuir responsabilidades extras no sistema. Isso evita a criação de varias classes que teoricamente é um conjunto de outras.

Adapter – Uma classe adaptadora fica responsável pela integração de dois modelos que não se comunicam isoladamente. Isso melhora a utilização de código e possibilita a interação de N objetos incompetiveis.

De criação:

Factory Method – Modo de encapsular a criação de novos objetos. Esses novos objetos chamados de produtos são determinados por subclasses da creator. Isso remove grande acoplamento entre classes.

Builder – Tem o propósito de isolar a construção de um objeto muito difícil/chato de ser criado. Essa “criação chata” geralmente acontece quando o construtor tem vários parâmetros ou propriedades do tipo de outros objetos. Isso facilita a leitura e mantem limpeza do código.

Singleton – A ideia aqui é ter uma única instancia de uma classe especifica que possa ser usada em todo escopo como uma classe global e uniforme. A ideia de ter uma variável global não é muito aceito porém isso é usado em muitos exemplos que retornam uma instancia de conexão de banco de dados para ser usado.