**Padrão de Projeto: Builder**

**O que é?**

O padrão de projeto Builder diz que, a partir de uma única classe, seja possível construir diferentes tipos de objetos com diferentes configurações.

**Como funciona?**

No padrão Builder, nós construímos a classe com métodos “builders” que “montam” determinadas partes do objeto. Esses métodos são independentes e podem ser chamados ou não quando tivermos uma nova instância dessa classe, dependendo da necessidade.

**Qual problema resolve?**

Digamos que eu tenha uma classe chamada “Casa” e essa classe tem diversos atributos como: número de portas, número de janelas, parede, telhado e piso. Uma casa básica tem, por exemplo: 2 janelas, 1 porta, 4 paredes, 1 telhado e 1 piso. Agora eu posso querer uma casa específica que possua cercado, quintal e outros atributos que o objeto base “Casa” não possui.

Uma solução para isso seria criar subclasses que herdam a classe “Casa” que possuem os métodos extras que a classe base não possui. E, nessas classes, criar construtores que chamam os atributos necessários de acordo com cada tipo de casa. Isso deixaria as chamadas das classes enormes.

Para solucionar este problema, com o padrão Builder, nós criamos uma classe “Casa” e, dentro dela, métodos independentes que constroem as partes da casa. Um método para construir o quintal, um método para construir o telhado, etc. E casa método só é chamado quando necessário. E, no fim, eu chamo o método que traz o objeto pronto.