

03 - Apartamento

O código apresentado possui um problema de design que viola o Princípio Aberto-Fechado (OCP) dos princípios SOLID. A classe **UpgradeApartamento** está aberta para modificações, pois se novos tipos de apartamentos forem adicionados, será necessário modificar o código para lidar com esses novos tipos.

Para corrigir esse problema, podemos utilizar o padrão Strategy e criar uma interface que define o comportamento de atualização de apartamento e, em seguida, implementar essa interface para cada tipo de apartamento que precisamos atualizar. Dessa forma, o código pode lidar com novos tipos de apartamentos sem modificação e seguindo o OCP.

No código, temos a interface **ApartamentoUpgrader**, que define o método upgrade para atualizar um apartamento. Em seguida, implementamos essa interface para cada comportamento que precisamos atualizar. Nesse caso, temos **ApartamentoAreaUpgrader** para atualizar a área do apartamento e **ApartamentoQuartosUpgrader** para atualizar o número de quartos.

Na classe **UpgradeApartamento**, agora temos uma lista de **ApartamentoUpgraders**, que são adicionados ao construtor da classe. O método upgrade agora itera sobre essa lista e chama o método upgrade de cada atualizador para atualizar o apartamento.