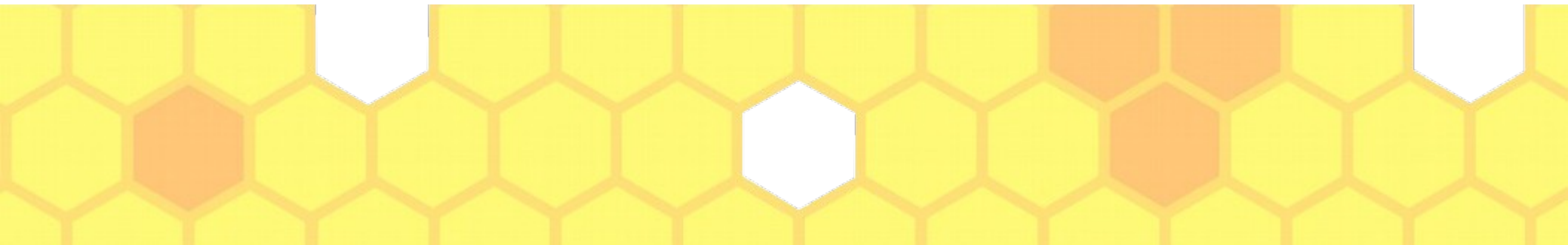




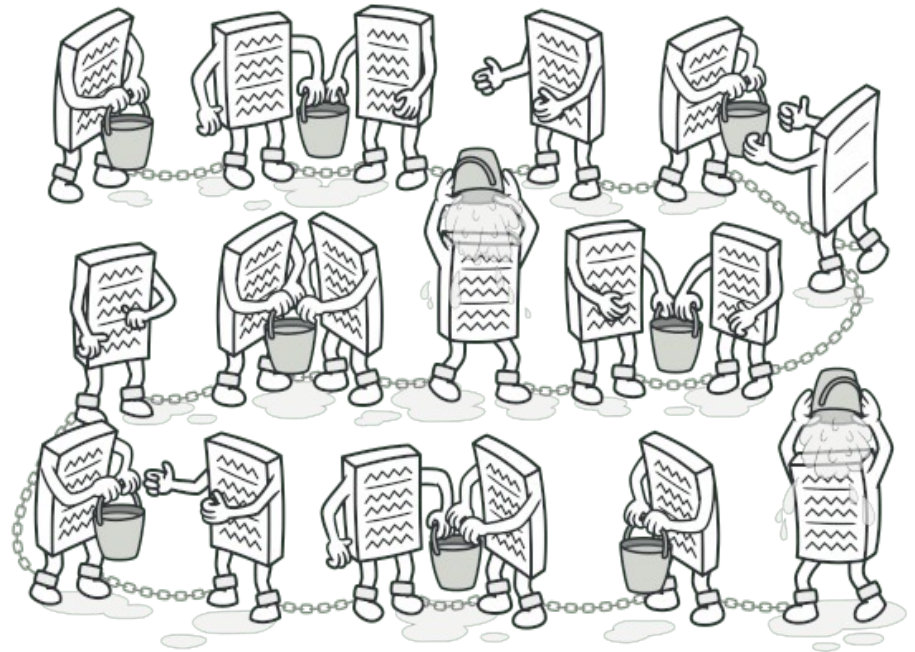
Chain Of Responsibility

- Categoria: Comportamental
Design Patterns



Propósito

- O Chain of Responsibility é um padrão de projeto comportamental que permite que você passe uma requisição por uma corrente de manipuladores. Ao receber uma requisição, cada manipulador decide se processa a requisição ou o passa adiante para o próximo objeto na corrente.



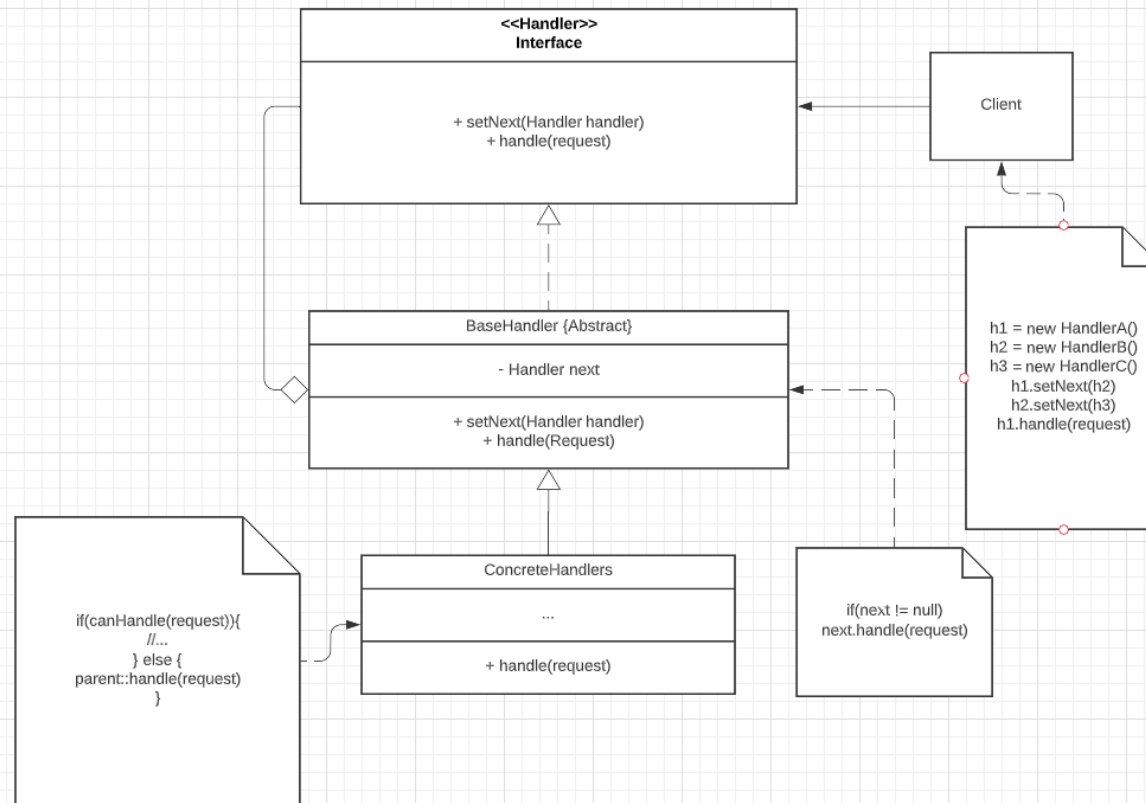
Sobre o Chain Of Responsibility

- É usado quando uma requisição precisa passar por uma sequência de operações até ser totalmente tratada.
- Separar a responsabilidade de quem envia e quem vai tratar a requisição.
- Permite que um objeto trate a requisição e chame o próximo objeto da cadeia.
- Permite que um objeto não trate a requisição e chame o próximo objeto da cadeia.
- Permite que um objeto trate a requisição e finalize a cadeia.
- Permite que um objeto não trate a requisição e finalize a cadeia.
- O cliente pode iniciar a requisição de qualquer objeto caso necessário.

Aplicabilidade

- Use o quando seu sistema precisa processar uma requisição em várias etapas diferentes e você não quer criar um ordem rígida para o processamento.
- Quando você quer aplicar o principio da responsabilidade única para tratamento de dados da requisição.
- Quando você quer permitir que os objetos responsáveis pelo tratamento de requisição possam variar em tempo de execução.

Estrutura



Prós e contras

- Você pode controlar a ordem de tratamento dos pedidos.
- Princípio de responsabilidade única. Você pode desacoplar classes que invocam operações de classes que realizam operações.
- Princípio aberto/fechado. Você pode introduzir novos objetos manipuladores na aplicação sem quebrar o código cliente existente.
- Alguns pedidos podem acabar sem tratamento.