

Questões teóricas do slide 45 do conjunto "Padrões de Projeto - c"

Diferença entre Adapter e Proxy

- Enquanto o adapter adapta uma classe a uma nova interface, geralmente através de uma herança múltipla empregada, o proxy meramente fornece a mesma interface, só que com algumas tarefas de limpeza, deixando o objeto mais leve.

Diferença entre Observer e Mediator

- Enquanto o Observer oferece uma dependência 1:n e acopla fortemente todos os dependentes, o Mediator encapsula todos os objetos/classes e age como intermediário entre as interações deles.

Diferença entre Flyweight e Composite

- O flyweight tem a finalidade de suportar múltiplos objetos e classes através de compartilhamento entre classes semelhantes otimizando o uso de memória, enquanto o composite, enquanto Composite trata objetos individuais ou composições do objeto como se fossem a mesma coisa, em um formato de árvore.

Diferença entre Singleton e Façade

- O Façade se estrutura mais como uma interface simplificada para que outros objetos e classes tirem proveito, facilitando a sequência de chamadas de métodos. O Singleton, por sua vez, é utilizado para que sua instância de um objeto seja exclusivamente única, em toda a aplicação.

Diferença entre Mediator e Proxy

- Enquanto o proxy fornece cópias de uma interface mais leve com tarefas de limpeza, o Mediator encapsula os objetos diretamente.

Diferença entre Chain of Responsibility e Adapter

- O padrão adapter tem o intuito de adaptar uma classe a uma nova interface com herança múltipla empregada, enquanto o Chain of Responsibility trata o fluxo de comportamento dos objetos.