

QUESTÃO 1

- PRINCÍPIO STRATEGY: Dar prioridade à composição.

Criar sistemas usando composição dá a você muito mais flexibilidade. Além de permitir encapsular uma família de algoritmos em seu próprio conjunto de classes, permite alterar o comportamento no tempo de execução.

- PRINCÍPIO OBSERVER: Busque designs levemente ligados entre objetos que interagem.

Projetos levemente ligados permitem construir sistemas OO flexíveis que podem lidar com mudanças porque minimizam a interdependência entre objetos.

-PRINCÍPIO DECORATOR: As classes devem estar abertas para extensão, mas fechadas para modificação.

É importante pois compondo objetos de formas dinâmicas é possível adicionar novas funcionalidades através da criação de um código novo, ao invés de alterar o já existente.

- PRINCÍPIO FAÇADE: Princípio do mínimo Conhecimento – só fale com seus amigos mais próximos.

É importante porque impede de criar projetos com um grande número de classes interconectadas, fazendo com que qualquer alteração numa parte do sistema exerça um efeito em cascata sobre outras partes.