

Quizz 5

INFORMAÇÕES DOCENTE						
CURSO: ENGENHARIA DE SOFTWARE	DISCIPLINA: PROJETO DE SOFTWARE	TURNO	MANHÃ	TARDE	NOITE	PERÍODO/SALA: 4º
PROFESSOR (A): João Paulo Carneiro Aramuni			x		x	

Padrões GRASP – General Responsibility Assignment Software Patterns

Padrão GRASP Invenção pura (Pure Fabrication)

1) Qual das opções abaixo exemplifica melhor o uso do padrão GRASP Invenção Pura (Pure Fabrication)?

- A) Utilizar uma classe Controladora para delegar requisições entre a interface e o domínio.
- B) Criar uma classe utilitária para realizar operações de formatação de data e hora.
- C) Definir uma hierarquia de classes para representar diferentes tipos de contas bancárias.
- D) Delegar a responsabilidade de calcular impostos diretamente para a classe Produto.
- E) Criar interfaces para diferentes implementações de um mesmo comportamento.

2) O padrão Invenção Pura é especialmente útil quando:

- A) Desejamos reduzir a dependência entre diferentes camadas do sistema.
- B) Queremos encapsular a lógica de negócio em classes diretamente relacionadas ao domínio.
- C) É necessário promover a reutilização e reduzir a duplicação de código.
- D) Precisamos mapear diretamente os conceitos do mundo real para as classes do sistema.
- E) Queremos aplicar herança para reutilizar comportamento entre classes relacionadas.

3) Uma possível desvantagem do uso do padrão Invenção Pura é:

- A) Aumentar a complexidade do sistema devido à criação de classes adicionais.
- B) Dificultar a implementação de lógica de negócio nas classes de domínio.
- C) Complicar a definição de interfaces entre os diferentes módulos.
- D) Restringir a reutilização de classes em diferentes contextos.
- E) Reduzir a coesão das classes de domínio.

Padrão GRASP Polimorfismo (Polymorphism)

4) Como o padrão Polimorfismo ajuda na aplicação de princípios de design orientado a objetos?

- A) Ele reduz a quantidade de classes necessárias para implementar variações.
- B) Ele permite que múltiplas classes compartilhem o mesmo comportamento por composição.
- C) Ele possibilita que classes com comportamentos distintos sejam tratadas de maneira uniforme.
- D) Ele elimina a necessidade de herança em sistemas complexos.
- E) Ele promove o encapsulamento ao eliminar a necessidade de interfaces.

5) Um exemplo de aplicação prática do padrão Polimorfismo seria:

- A) Criar uma classe que implementa métodos estáticos para cálculos matemáticos.
- B) Definir uma interface e implementar várias classes que representam diferentes algoritmos.
- C) Criar um controlador único para gerenciar todas as requisições de um sistema.
- D) Utilizar bibliotecas externas para reduzir a necessidade de abstrações.
- E) Implementar uma classe base com comportamento genérico e especializá-la em subclasses.

6) Qual das opções abaixo ****não**** está diretamente relacionada ao uso do Polimorfismo?

- A) Substituir verificações condicionais por chamadas a métodos dinâmicos.
- B) Delegar responsabilidades a diferentes implementações de uma interface.
- C) Encapsular a lógica de decisão em uma hierarquia de classes.
- D) Adicionar métodos utilitários a uma classe concreta para reduzir o acoplamento.
- E) Garantir que o comportamento possa ser alterado dinamicamente em tempo de execução.

Padrão GRASP Indireção (Indirection)

7) O padrão GRASP Indireção (Indirection) é aplicado quando:

- A) Desejamos promover o baixo acoplamento entre módulos do sistema.
- B) Queremos encapsular a lógica de negócio diretamente no objeto de domínio.
- C) Precisamos evitar a utilização de herança em hierarquias complexas.
- D) Procuramos mapear conceitos do mundo real diretamente para o sistema.
- E) Buscamos melhorar a performance ao reduzir o número de chamadas de métodos.

- 8) Uma implementação correta do padrão Indireção seria:
- A) Criar uma classe que encapsula a comunicação com um sistema externo.
 - B) Utilizar métodos estáticos para compartilhar comportamento entre classes.
 - C) Reduzir o número de camadas na arquitetura para simplificar o design.
 - D) Estabelecer relações diretas entre todas as classes para melhorar o desempenho.
 - E) Delegar toda a lógica de interface ao modelo de domínio.

- 9) Qual das seguintes afirmações melhor descreve o objetivo do padrão Indireção?
- A) Facilitar a substituição de módulos sem impactar outros componentes do sistema.
 - B) Fornecer uma implementação genérica para variações de um comportamento.
 - C) Reduzir a complexidade estrutural através da simplificação de interfaces.
 - D) Promover o reuso direto de código sem criar abstrações adicionais.
 - E) Centralizar toda a lógica de negócio em um único módulo.

Padrão GRASP Variações Protegidas (Protected Variations)

- 10) O padrão Variações Protegidas (Protected Variations) é mais eficaz quando:
- A) Há uma necessidade de garantir a estabilidade de uma interface em um ambiente em constante mudança.
 - B) É importante minimizar a quantidade de classes em um sistema orientado a objetos.
 - C) Desejamos encapsular comportamento em classes concretas.
 - D) Precisamos reduzir o número de dependências entre diferentes camadas do sistema.
 - E) Procuramos criar estruturas estáticas que não precisam de extensibilidade.

- 11) Qual estratégia reflete melhor a aplicação do padrão Variações Protegidas?
- A) Criar uma classe base abstrata e subclasses concretas para diferentes variações.
 - B) Delegar a lógica de variação para um método estático de uma classe utilitária.
 - C) Garantir que a lógica de variação esteja presente apenas na interface do usuário.
 - D) Utilizar uma classe com responsabilidades múltiplas para gerenciar todas as variações.
 - E) Definir constantes no código para substituir valores variáveis.

- 12) Qual é o impacto direto de usar o padrão Variações Protegidas em um sistema?
- A) Aumentar o acoplamento para garantir maior flexibilidade.
 - B) Promover a extensibilidade sem impactar os módulos existentes.
 - C) Simplificar a lógica interna ao remover abstrações desnecessárias.
 - D) Centralizar a tomada de decisão em classes de controle específicas.
 - E) Eliminar a necessidade de abstrações complexas no design.

Gabarito:

- 1) B - Criar uma classe utilitária para realizar operações de formatação de data e hora.
O padrão GRASP Invenção Pura é utilizado quando criamos classes que não representam diretamente conceitos do domínio, mas que existem para fornecer funcionalidade, como utilitários para tarefas comuns (neste caso, formatação de data e hora).
- 2) C - É necessário promover a reutilização e reduzir a duplicação de código.
A Invenção Pura é útil para encapsular funcionalidades comuns e reutilizáveis que não pertencem diretamente ao domínio, como bibliotecas utilitárias, ajudando a reduzir a duplicação de código.
- 3) A - Aumentar a complexidade do sistema devido à criação de classes adicionais.
Uma desvantagem da Invenção Pura é que ela pode adicionar complexidade ao sistema, pois introduz classes que não têm uma representação direta no domínio do problema.
- 4) C - Ele possibilita que classes com comportamentos distintos sejam tratadas de maneira uniforme.
O Polimorfismo permite que diferentes classes implementem comportamentos distintos, mas sejam tratadas da mesma forma, simplificando o código ao permitir que o comportamento de diferentes classes seja invocado de maneira uniforme.
- 5) B - Definir uma interface e implementar várias classes que representam diferentes algoritmos.
Um exemplo clássico de Polimorfismo é a definição de uma interface comum e a implementação de diferentes classes com comportamentos variados, como diferentes algoritmos.
- 6) D - Adicionar métodos utilitários a uma classe concreta para reduzir o acoplamento.
Polimorfismo foca na substituição de condicionais por chamadas dinâmicas a métodos, e adicionar métodos utilitários a uma classe concreta não está diretamente relacionado ao conceito de Polimorfismo, que foca na variação de comportamentos.
- 7) A - Desejamos promover o baixo acoplamento entre módulos do sistema.
O padrão Indireção busca reduzir o acoplamento entre módulos, permitindo que um módulo interaja com outro através de um intermediário, em vez de uma dependência direta.
- 8) A - Criar uma classe que encapsula a comunicação com um sistema externo.

Um exemplo de Indireção é criar uma classe que serve como intermediária para lidar com a comunicação com sistemas externos, como uma API, promovendo o baixo acoplamento e facilitando a manutenção.

9) A - Facilitar a substituição de módulos sem impactar outros componentes do sistema. O objetivo principal do padrão Indireção é promover a flexibilidade no sistema, permitindo que módulos possam ser trocados ou modificados sem afetar diretamente outros módulos, o que melhora a manutenção e a extensibilidade do sistema.

10) A - Há uma necessidade de garantir a estabilidade de uma interface em um ambiente em constante mudança.

O padrão Variações Protegidas é mais eficaz quando se busca proteger um sistema de variações constantes, garantindo que a interface do sistema permaneça estável, mesmo quando a implementação interna muda.

11) A - Criar uma classe base abstrata e subclasses concretas para diferentes variações. Uma forma de aplicar o padrão Variações Protegidas é criar uma classe base que define o comportamento comum, com subclasses específicas para diferentes variações, permitindo flexibilidade sem afetar os módulos existentes.

12) B - Promover a extensibilidade sem impactar os módulos existentes.

O uso do padrão Variações Protegidas permite que novas variações sejam introduzidas no sistema sem afetar as funcionalidades existentes, promovendo extensibilidade e flexibilidade sem quebrar a compatibilidade.