

3. Como você se atualiza tecnicamente?

Utilizo o site de cursos online Udemey.

4.

4.1 Qual a solução é mais performática, iterativa ou recursiva? Por quê?

No caso do código acima é mais performática a solução iterativa, já que a solução recursiva é ineficiente, pois recalcula várias vezes a solução para valores intermediários.

4.2 Opcional: Qual é o 5287º elemento da sequência?

O elemento é 1852806896.

5. O que significa SOLID?

Solid é um acrônimo que contém os cinco princípios de orientação a objetos e design de código. Seu nome é composto pelos cinco princípios:

S = Single Responsibility Principle

O = Open-Closed Principle

L = Liskov Substitution Principle

I = Interface Segregation Principle

D = Dependency Inversion Principle

6. O que são design patterns?

1. Quais são os tipos de design patterns?

Comportamental (Behavioral), Estrutural (Structural) e Criacional (Criação).

2. Com quais você está familiarizado? Qual é a função deles?

Criação, utilizando o Singleton. Singleton especifica que apenas uma instância da classe pode existir, e esta será utilizada por toda a aplicação. Dessa forma temos apenas um ponto de acesso central a esta instância da classe, temos mais controle sobre o acesso às propriedades e métodos de uma classe, e reduzimos o consumo de memória desnecessário por utilizar várias instâncias desnecessárias de uma classe.

3. Opcional: Qual é sua opinião quanto ao uso de design patterns?

De extrema importância no desenvolvimento de qualquer software. A utilização desses padrões nos ajuda a desenvolver de forma mais rápida frente a desafios semelhantes, fornece uma linguagem comum durante a documentação e discussões técnicas além de organizar o código fonte do software que estamos desenvolvendo.

7. Qual foi o último livro técnico que você leu? Quando foi isso?

Tenho preferência em estudar cursos, conforme a resposta inicial destas questões. Atualmente estou fazendo curso de ReactJS.

8. Cite 3 maneiras diferentes de implementar Dependency Inversion.

9. O que são ORMs?

É uma técnica que aproxima o desenvolvimento de aplicações orientadas a objetos ao banco de dados relacional.

9.1. Quais você conhece bem?

Entity Framework.

9.2 Opcional: Cite pelo menos 2 vantagens e 2 desvantagens de seu uso.

Vantagens:

- Obriga-nos a escrever o código MVC, que, no final, torna seu código um pouco mais limpo;
- Ele permite que você use OOP como herança;

Desvantagens:

- Suas bibliotecas não são leves;
- Necessidade de configuração;

10. O que são microserviços?

Microserviços são uma abordagem de arquitetura para a criação de aplicações. O que diferencia a arquitetura de microserviços das abordagens monolíticas tradicionais é como ela decompõe a aplicação por funções básicas. Cada função é denominada um serviço e pode ser criada e implantada de maneira independente. Isso significa que cada serviço individual pode funcionar ou falhar sem comprometer os demais.

10.1 Quais são suas vantagens e desvantagens?

Vantagens:

- Altamente escalável;
- Fácil de implementar;

Desvantagens:

- Alto custo financeiro;
- Altamente escalável;