Questões relacionadas a C#

1. Orientação a Objetos:

• Explique o conceito de herança múltipla e como C# aborda esse cenário.

Em herança múltipla você pode herdar de classes abstratas ou não e de N’s interfaces, aplicando suas bases, sobrecargas e implementações obrigatórias. Classes abstratas ou não em primeiro e depois as N’s interfaces.

• Explique o polimorfismo em C# e forneça um exemplo prático de como ele

pode ser implementado.

O conceito de polimorfismo define que uma classe que herda de uma classe pai, pode ser outro objeto e implementar em um método virtual tendo outro tipo de retorno não que o do seu pai;

Exemplo:

public class animal {

public virtual void Raca (){

Console.WriteLine(“animal”);

}  
  
}

public class cachorro : animal {

public override void Raca (){

Console.WriteLine(“pastor alemao”);

}

}

public class gato : animal {

public override void Raca (){

Console.WriteLine(“gatinho”);

}  
}

2. SOLID:

• Descreva o princípio da Responsabilidade Única (SRP) e como ele se aplica em

um contexto de desenvolvimento C#.

Uma classe, método ou função deve ter somente uma única responsabilidade.   
Ou seja, uma classe deve ter somente uma razão para ser alterada e um método/função deve executar somente uma ação; CQRS, por exemplo;

• Como o princípio da inversão de dependência (DIP) pode ser aplicado em um

projeto C# e como isso beneficia a manutenção do código?

Devemos depender somente de abstrações (interfaces) ao invés das implementações concretas (classes) a fim de ter um menor acoplamento entre as diferentes camadas da aplicação. Diminuindo o alto acoplamento, facilita a manutenção em alterar contextos das classes e diminuindo o risco de impacto em outras regiões da aplicação.

3. Entity Framework (EF):

• Como o Entity Framework gerencia o mapeamento de objetos para o banco de

dados e vice-versa?

Há a criação do modelo conceitual utilizando o contexto quando o nome da entitidade ou model é a mesma da classe e enviado para o contexto; Há o modelo lógico em que é realizado o mapeamento do tipo repositório definindo os nomes das colunas no mapeamento;

• Como otimizar consultas no Entity Framework para garantir um desempenho

eficiente em grandes conjuntos de dados? Evitar retornar todos os objetos de banco de dados em único modelo de entidade; Desativar o controle de tracking; Utilizar views para reduzir o tempo de respostas; Evitar retornar campos não obrigatórios; Utilizar coleções apropriadas para retornar, como: Ienumerable, Iqueryable, Ilist; Utilizar consultas compiladas sempre que necessário; Retornar sempre o número requerido na paginação; Evitar utilizar contains;

4. WebSockets:

• Explique o papel dos WebSockets em uma aplicação C# e como eles se

comparam às solicitações HTTP tradicionais.

Em um protocolo websocket você estabelece um (handshake) abrindo uma conexão entre os cliente e servidor, mantendo aberto para o tráfego de dados e somente após isso a conexão é fechada;

• Quais são as principais considerações de segurança ao implementar uma

comunicação baseada em WebSockets em uma aplicação C#?

Tendo estabelecido um handshake entre o cliente e servidor, você pode restringir a aplicação que pode se conectar ao websocket e que fará a troca de frames entre as aplicações;

5. Arquitetura:

• Descreva a diferença entre arquitetura monolítica e arquitetura de

microsserviços. Qual seria sua escolha ao projetar uma aplicação C#?

A arquitetura monolítica se utiliza de um mecanismo vertical, tendo todos os módulos e domínios na mesma aplicação. A arquitetura de micro serviços, utiliza se de pequenas aplicações que se responsabilizam somente pelo seu contexto e responsabilidade, realizando integração seja por rest ou eventos entre as aplicações; Não há a melhor resposta, depende da proposta e do problema no qual quer se resolver;

• Como você escolheria entre a arquitetura de microsserviços e a arquitetura

monolítica ao projetar uma aplicação C# que precisa ser altamente escalável?

Caso a aplicação possa ter seus dados sub divididos entre domínios e contextos, integrando seus dados de forma rastreavel, manteria por meio de microserviços para aplicações que precisam de alto processamento de informações e escalabilidade, aquela que pode ser utilizada para transitar estas informações com os micro serviços utilizando somente de uma integração com os microserviços pode vir a ser monolítica;