**UNIDADE 1**

**Questão 1: Qual a diferença entre Polimorfismo, Herança, Encapsulamento e Abstração?**

**Herança**

A herança é um mecanismo que permite criar uma classe baseada em uma classe existente. A nova classe herda os membros (métodos e propriedades) da classe existente e pode adicionar ou modificar seu comportamento.

**Encapsulamento**

O encapsulamento é o conceito de restringir o acesso direto aos detalhes internos de uma classe e fornecer uma interface para interagir com ela. Isso é alcançado usando modificadores de acesso (public, private, protected) e propriedades.

**Abstração**

A abstração é o conceito de simplificar complexidades, representando as características essenciais sem incluir os detalhes complicados. Em C#, a abstração pode ser alcançada usando classes abstratas e interfaces.

**Polimorfismo**

O polimorfismo permite que objetos de diferentes tipos sejam tratados de maneira uniforme. Existem dois tipos de polimorfismo: polimorfismo de tempo de compilação (sobrecarga de método) e polimorfismo de tempo de execução (sobreposição de método).

**Questão 2: O que é Associação, Agregação, Composição e Generalização?**

**Associação**

Associação é um relacionamento entre duas classes. Pode ser bidirecional ou unidirecional. Ela representa a conexão entre objetos de diferentes classes.

Exemplo: Uma classe Estudante pode estar associada a uma classe Curso. Pode ser uma associação simples, como um estudante que está matriculado em um curso.

**Agregação**

Agregação é um tipo específico de associação que representa uma relação "todo-parte", onde uma classe é composta por outras classes, mas as partes podem existir independentemente do todo.

Exemplo: Uma classe Time pode ter uma agregação com a classe Jogador. Os jogadores podem existir independentemente do time, e um time pode ter vários jogadores.

**Composição**

Composição é semelhante à agregação, mas as partes não podem existir independentemente do todo. Se o todo for destruído, as partes também serão. A composição é uma relação mais forte que a agregação.

Exemplo: Uma classe Carro pode ter uma composição com a classe Motor. Se o carro for destruído, o motor também será.

**Generalização**

Generalização é um relacionamento entre uma classe mais geral (superclasse) e uma classe mais específica (subclasse). A subclasse herda características da superclasse, permitindo a reutilização de código.

Exemplo: Uma classe Animal pode ser uma superclasse para as classes Mamífero, Réptil, etc. As subclasses herdam os atributos e métodos da superclasse, mas podem ter suas próprias características específicas.