# Resposta da Questão #1

É um padrão de design de software amplamente utilizado para separar as responsabilidades em aplicações, especialmente em desenvolvimento web. Ele divide a aplicação em três componentes principais: O **Model** (Modelo) - A camada que lida com as regras de negócios e a manipulação dos dados, A **View** (Visão) - A camada de apresentação da aplicação, responsável por exibir os dados ao usuário e o **Controller** - Atua como intermediário entre o Model e a View.

# Resposta da Questão #2

## Conceito:

**Singleton Pattern** é um padrão de design de software que garante que uma classe tenha apenas uma instância em todo o ciclo de vida da aplicação e fornece um ponto global de acesso a essa instância.

## Objetivo:

1. Instância única: A classe só pode ter uma única instância durante a execução do programa. Isso é útil quando se deseja limitar o número de objetos de uma classe;
2. Acesso controlado: O padrão oferece um método público que permite acessar a instância única da classe; e
3. Construtor privado: Para garantir que nenhuma outra classe ou código crie novas instâncias da classe, o construtor da classe é marcado como privado. Isso impede a criação direta de objetos fora da classe Singleton.

## Cenário de Uso:

Servidores de Aplicação: Controla a configuração global para garantir o mesmo setup para todas as sessões e/ou usuários. O controle centralizado em uma única instância, fica mais fácil gerenciar recursos globais e acesso controlado, pois o objeto é acessado de um único ponto, é mais simples monitorar e alterar seu comportamento.