O Pesquise sobre padrões de projeto e escolha um para apresentar e descrever o seu funcionamento. Além disso, explique quais as vantagens e desvantagens comparados a outros e mostre suas referências.

Realize essa atividade no WORD ou no Bloco de Notas, suba esse arquivo para algum repositório e compartilhe o link no campo ao lado para que outros desenvolvedores possam analisá-lo.

**Padrão Criacional – Singleton vs Factory**

O **padrão Singleton** permite criar objetos únicos para os quais há apenas uma instância. Poe ser instanciada e usada apenas quando necessário, diferentemente se criássemos uma variável global em que o objeto é sempre criado quando o aplicativo é inicializado e poderá estar usando recursos que não são necessários neste momento.

É utilizado em diversos aplicativos e projetos de software como em Drivers que precisam de um ponto de acesso único e global para gerenciar diversos recursos.

O melhor exemplo para o uso do padrão singleton é na conexão com o banco de dados. Ao invés de vários pontos no nosso projeto, otimizando com o banco de dados podemos ter apenas uma classe singleton que realiza esta conexão, fazendo uma performance da aplicação.

É um padrão mais simples de se aprender em relação aos demais

Ao utilizar **Singleton** temos mais controle sobre o acesso às propriedades e métodos de uma classe, e reduzimos o consumo de memória desnecessário por utilizar várias instancias desnecessárias de uma classe.

Um padrão **Singleton** garante que você sempre receba de volta a mesma instância de qualquer tipo que esteja recuperando, enquanto o padrão de **Factory** geralmente fornece uma instância diferente de cada tipo. Sua principal vantagem em relação ao padrão **Factory**.