

Padrão de Notação

Padrão de Notação de Código

Adotar o padrão de notação CamelCase para nomes de variáveis e funções.

Exemplo: minhaVariavel, calcularValorTotal.

Utilizar PascalCase para nomes de classes e interfaces. Exemplo: MinhaClasse, InterfaceDeExemplo.

Documentação e Comentários de Código:

Todo código deve ser comentado para explicar a lógica complexa ou o motivo por trás de decisões específicas.

Utilizar comentários de cabeçalho em cada função/método para descrever seu propósito, parâmetros, e valor de retorno.

Evitar comentários redundantes ou óbvios que apenas repetem o que o código já expressa claramente.

Aplicação dos Princípios SOLID:

Seguir os princípios SOLID para garantir que o código seja modular, fácil de entender e de manter.

Exemplo: Seguir o princípio de responsabilidade única, assegurando que cada classe ou função tenha uma única responsabilidade bem definida.

Aplicação de Clean Code:

Escrever código limpo e legível, focando em nomes descritivos para variáveis, funções e classes.

Evitar código duplicado e manter funções/métodos curtos, preferencialmente com uma única responsabilidade.

Manter a Consistência do Código:

Seguir o mesmo padrão de formatação em todo o código, como indentação, espaçamento, e uso de chaves.

Definir e seguir um conjunto de convenções de nomenclatura, estrutura de diretórios e arquivos.

Controle de Versionamento:

Utilizar uma ferramenta de controle de versão (Git) para gerenciar e registrar todas as alterações no código.

Seguir uma estratégia de branch definida, como main para versões estáveis, develop para integração de novas funcionalidades e branches específicas para cada feature ou correção de bug.

Realizar commits frequentes e bem descritos, facilitando o rastreamento de mudanças e colaborando com a equipe.