

Boas Práticas de Desenvolvimento

Dicas e estratégias para manter o código limpo, legível e sustentável.

1. ****Escreva código limpo e legível**** — Use nomes claros e descritivos para variáveis, funções e classes. Evite abreviações confusas e siga um padrão de nomenclatura consistente (ex: camelCase, PascalCase).
2. ****Comente com propósito**** — Comentários devem explicar o 'porquê' e não o 'como'. Um código bem escrito deve ser autoexplicativo, e os comentários servem para esclarecer decisões importantes.
3. ****Mantenha funções curtas e objetivas**** — Cada função deve fazer apenas uma coisa e fazê-la bem. Isso facilita a leitura, manutenção e testes automatizados.
4. ****Aplique o princípio DRY (Don't Repeat Yourself)**** — Evite duplicar código. Centralize lógicas repetidas em funções reutilizáveis para reduzir erros e retrabalho.
5. ****Organize o projeto**** — Estruture pastas e arquivos de forma lógica. Separe camadas de responsabilidade (ex: modelo, visualização e controle) e mantenha o padrão ao longo do projeto.
6. ****Use controle de versão**** — Ferramentas como Git permitem acompanhar mudanças, colaborar em equipe e reverter alterações com segurança.
7. ****Crie e siga convenções de código**** — Definir um guia de estilo (como PEP8 no Python ou o padrão do .NET) mantém o código consistente, mesmo entre diferentes desenvolvedores.
8. ****Priorize testes**** — Automatize testes unitários e de integração. Isso garante que novas funcionalidades não quebrem partes existentes do sistema.
9. ****Documente o projeto**** — Inclua README, instruções de instalação, dependências e exemplos de uso. Uma boa documentação é essencial para continuidade do desenvolvimento.
10. ****Revise e refatore com frequência**** — Código bom não é o que nunca muda, mas o que melhora com o tempo. Faça revisões regulares e simplifique trechos complexos.

Essas práticas tornam o desenvolvimento mais ágil, colaborativo e sustentável — garantindo qualidade e longevidade aos projetos.