CLEAN CODE

O USO DE BOAS PRÁTICAS DE CÓDIGO LIMPO COM NODEJS

Oficina de NodeJS com Typescript que apresentará a importância do Código Limpo no desenvolvimento de software e utilizará de boas práticas para construir uma simples aplicação back-end de criação de usuários.

https://balta.io/blog/clean-code#o-que-%C3%A9-o-clean-code

O QUE É?

Clean Code é uma filosofia de desenvolvimento cuja o principal objetivo é aplicar técnicas simples que visam facilitar a escrita e leitura de um código, tornando-o de fácil compreensão e revelando a sua real intenção.

https://medium.com/desenvolvendo-com-paixao/1-clean-code-o-que-%C3%A9-porque-usar-1e4f9f4454c6

POR QUE USAR?

01. Legibilidade
02. Manutenibilidade
03. Escalabilidade
04. Colaboração
05. Eficiência
06. Depuração
07. Reutilização

https://www.linkedin.com/pulse/benefits-clean-code-your-application-development-daniel-donald

ROBERT C. MARTIN

APRESENTAÇÃO

Robert Cecil Martin, mais conhecido como "Uncle Bob," é um renomado autor, consultor e palestrante na área de desenvolvimento de software. Ele é amplamente reconhecido por seu papel na promoção das melhores práticas de engenharia de software e é um dos principais defensores da metodologia ágil e do desenvolvimento orientado a objetos.

Martin é autor de vários livros influentes sobre programação e design de software, incluindo "Clean Code: A Handbook of Agile Software Craftsmanship." Ele é um dos fundadores do Manifesto Ágil e tem desempenhado um papel fundamental na disseminação dos princípios ágeis na indústria de desenvolvimento de software.

Além disso, Robert C. Martin é co-autor do Princípio SOLID, um conjunto de diretrizes de design de software que promovem a criação de código limpo e de alta qualidade. Ele é um proponente entusiástico da importância da manutenção e melhoria contínuas do código-fonte para criar sistemas robustos e escaláveis.

Não basta que o código funcione.

https://www.goodreads.com/author/quotes/45372.Robert_C_Martin

PRINCÍPIO KISS

Keep It Simple, Stupid é um princípio de design que afirma que projetos e/ou sistemas devem ser tão simples quanto possível. Sempre que possível, a complexidade deve ser evitada, pois a simplicidade garante maiores níveis de aceitação e interação do usuário. O *KISS* é usado em uma variedade de disciplinas, como design de interface, design de produto e desenvolvimento de software.

https://uxdesign.blog.br/a-origem-do-keep-it-simple-stupid-kiss-b24085dc1327

PRINCÍPIO DRY

Don't Repeat YourSelf é um princípio de desenvolvimento de software que visa reduzir a repetição de informações de todos os tipos, especialmente no código-fonte. Isso é alcançado pela divisão do código em partes menores, chamadas funções ou métodos, que podem ser reutilizadas em vez de duplicadas.

https://www.linkedin.com/pulse/o-princ%C3%ADpio-dry-dont-repeat-yourself-em-de-software-de-oliveira/?originalSubdomain=pt

REGRAS DE CÓDIGO

Existem algumas regras que o Clean Code dita, importantes para manter o propósito da filosofia, são regras que se seguidas, deixam o código mais limpo e entendível.

MANTENHA CONSTÂNCIA

ABUSE DE VARIÁVEIS DESCRITIVAS

```
1   const total = 0
2
3   const valorTotalDoCarrinho = 0
```

FOQUE EM DISTINÇÕES SIGNIFICANTES

```
1  const cron = '*/5 * * * *'
2
3  const temporizador = '*/5 * * * *'
4
5  const aCadaCincoSegundos = '*/5 * * * *'
```

TODOS OS NOMES PRECISAM SER BUSCÁVEIS

NÃO USE PREFIXOS

NÃO USE CARACTERES ESPECIAIS

```
1   const tipoCartão = 'CRÉDITO'
2
3   const função = 'PROGRAMADOR'
```

APENAS UM OBJETIVO

```
async cadastrarUsuario() {
  // Valida CPF
  // Cadastra no banco
}

async validaCPF() { }
async persiste() { }
```

PRÁTICA