

A Arte da Simplicidade: Escrevendo Código Limpo e Eficiente

Porque diminuir
as linhas de
codigo?



Um código simples e eficiente é como uma obra de arte: bonito, funcional e fácil de entender.

Benefícios da Redução de Linhas de Código

- Manutenibilidade
- Legibilidade
- Desempenho

Simplicidade e eficiência do código



Zen do Python:

- Bonito é melhor que feio.
- Explícito é melhor que implícito.
- Simples é melhor que complexo.
- Complexo é melhor que complicado...

CONCEITOS FUNDAMENTAIS

DRY (Don't repeat yourself)
(Não se repita)

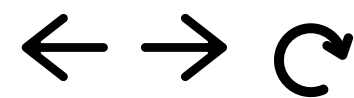
KISS (Keep it simple, stupid)
(Mantenha isso simples, idiota)

YAGNI (you aren't gonna need it)
(Você não vai precisar disso)

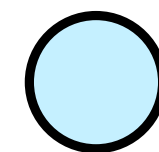
O que é
refatoração e
por que é
importante?

TECNICAS DE REFATORAÇÃO

- Remover código duplicado
- Tornar o código mais direto
- Dividir o código em partes menores e mais gerenciáveis



www.exemplo_de_refatoração_em_Python.com.br



```
1  # Antes:
2
3  def par_impar2(num):
4      if num % 2 == 0:
5          print("Par")
6      else:
7          print("Ímpar")
8
9  par_impar2(6)
10
11 # Depois:
12
13 def par_impar(num):
14     print("Par!" if num % 2 == 0 else "Ímpar!")
15
16 par_impar(3)
```

ALGORITIMOS OTIMIZADOS



Abstração e
encapsulamento

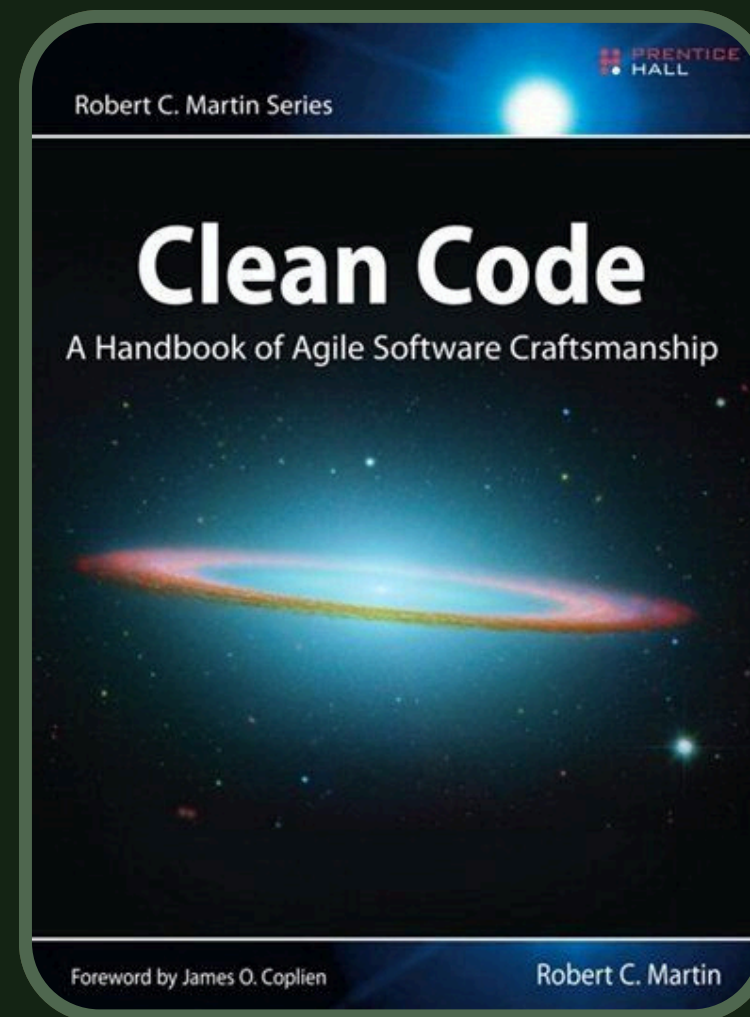
Uso de
bibliotecas de
frameworks

Avaliação de
quando e como
usar dependências
externas

“um dos meus dias
mais produtivos foi
quando eu joguei fora
1000 linhas de código”

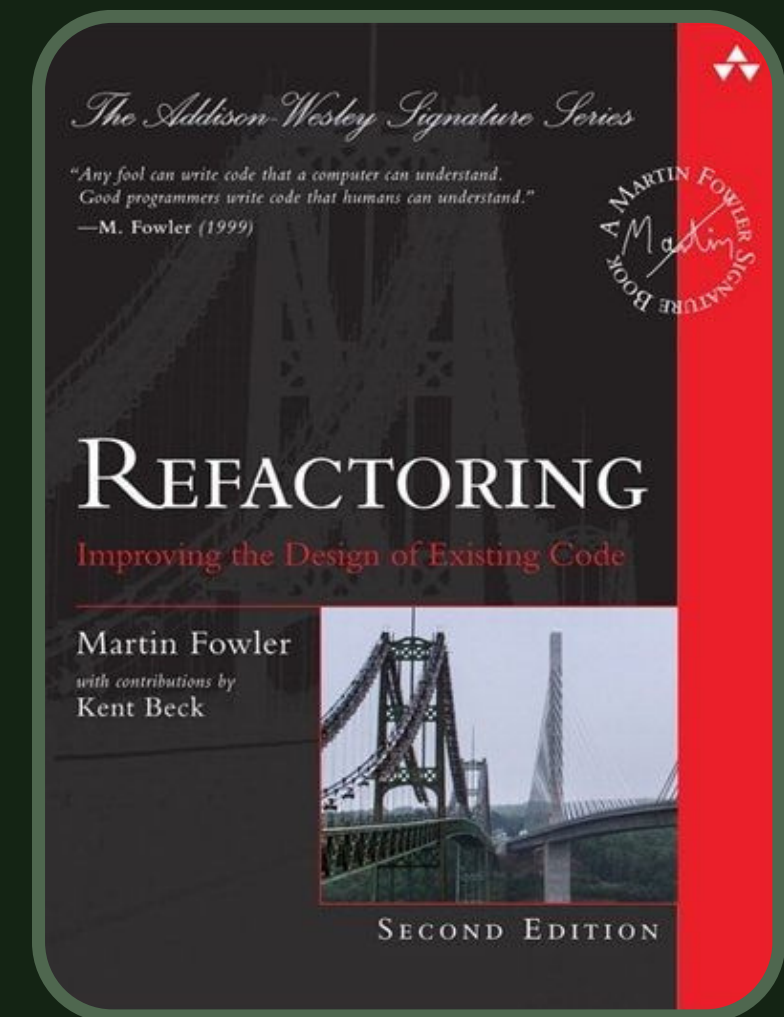
- Frase dita por – Ken Thompson

Artigos e Livros



Clean Code

Robert C. Martin



Refactoring

Martin Fowler



CONCLUSÃO

Reduzir linhas de código traz muitos benefícios, como melhor manutenibilidade, legibilidade e desempenho.



Agradeço!