



# S.O.L.I.D.

**Os 5 princípios da programação  
orientada a objetos**

# Introdução

Os princípios SOLID são um conjunto de diretrizes para a criação de software de forma mais organizada, compreensível e fácil de manter. Neste e-book, vamos explorar cada um desses princípios de maneira simples, acompanhados de exemplos práticos para facilitar o entendimento.

01

# **Princípio da Responsabilidade Única (SRP)**

# Princípio da Responsabilidade Única (SRP)

O que é?

O Princípio da Responsabilidade Única (Single Responsibility Principle) diz que uma classe deve ter apenas uma razão para mudar, ou seja, deve ter uma única responsabilidade ou objetivo.

Exemplo prático:

Imagine que temos uma classe que lida com pedidos de uma loja online (python):

```
class Pedido:
    def calcular_total(self):
        # Lógica para calcular o total do pedido
        pass

    def salvar_no_banco(self):
        # Lógica para salvar o pedido no banco de dados
        pass

    def enviar_email_confirmacao(self):
        # Lógica para enviar um email de confirmação
        pass
```

# Princípio da Responsabilidade Única (SRP)

Essa classe tem múltiplas responsabilidades. Vamos separá-las:

```
class Pedido:
    def calcular_total(self):
        # Lógica para calcular o total do pedido
        pass

class PedidoRepository:
    def salvar_no_banco(self, pedido):
        # Lógica para salvar o pedido no banco de dados
        pass

class PedidoEmailService:
    def enviar_email_confirmacao(self, pedido):
        # Lógica para enviar um email de confirmação
        pass
```

02

# **Princípio Aberto/Fechado (OCP)**

03

# **Princípio da Substituição de Liskov (LSP)**

# 04

## **Princípio da Segregação de Interfaces (ISP)**



05

# **Princípio da Inversão de Dependência (DIP)**