

SOLID

domingo, 23 de janeiro de 2022 19:07



SOLID é um acrônimo dos cinco primeiros princípios da programação orientada a objetos e design de código identificados por Robert C. Martin (ou Uncle Bob) por volta do ano 2000.

SRP - Single Responsibility Principle

Uma classe deve ter um, e apenas um, motivo para ser modificada.

OCP - Open Closed Principle

Entidades de Softwares (Classes Módulos, Funções, etc.) devem estar abertas para extensão, mas fechada para modificação.

LSP - Liskov Substitution Principle

Se $q(x)$ é uma propriedade demonstrável dos objetos x de tipo T . Então $q(y)$ deve ser verdadeiro para objetos y de tipo S onde S é um subtipo de T . Subclasses deve ser substituídas por suas Superclasses

"Se nada como um pato, voa como um pato, porém precisa de baterias, provavelmente você possui um problema de abstração"

ISP - Interface Segregation Principle

"Clientes não devem ser forçados a depender de métodos que não usam" (Clientes => Clientes da interface)

Muitas interface específicas são melhores que uma interface única

DIP - Dependency Inversion Principle

Módulos de alto nível não devem depender de módulos de baixo nível. Ambos devem depender de abstrações; Abstrações não devem depender de detalhes, Detalhes devem depender de abstrações.

Dependa de uma abstração e não de uma implementação