



## O que aprendemos?

- Que é mais interessante e mais seguro para o nosso código depender de interfaces (classes abstratas, assinaturas de métodos e interfaces em si) do que das implementações de uma classe;
- Que as interfaces são menos propensas a sofrer mudanças enquanto implementações podem mudar a qualquer momento;
- Que o Princípio de Inversão de Dependência (DIP) diz que implementações devem depender de abstrações e abstrações não devem depender de implementações;
- Que as interfaces devem definir apenas os métodos que fazem sentido para seu contexto;
- Que o Princípio de Segregação de Interfaces (ISP) diz que uma classe não deve ser obrigada a implementar um método que ela não precisa;
- Os conceitos aprendidos neste treinamento formam o acrônimo SOLID
  - **S**ingle Responsibility Principle
  - **O**pen Closed Principle
  - **L**iskov Substitution Principle
  - **I**nterface Segregation Principle
  - **D**ependency Inversion Principle