

## JUnit

---

# Introdução

- Framework de testes unitários para **JavaScript**.
- Criado pela equipe do jQuery, hoje é um projeto independente.
- É possível de ser executado tanto em navegadores quanto em ambiente Node.js.

## Vantagens

- **Sintaxe clara**: é um grande facilitador na escrita dos testes
- **Relatórios visuais** detalhados sobre sucessos e falhas.
- Totalmente **independente**, não exige outras bibliotecas.
- Flexível para testes de **unidade** e de **regressão**.

## Categorização

- **Nível de teste:** **Teste de unidade**, foco em validar funções isoladas.
- **Teste de Caixa-Preta:** Validação das entradas e saídas sem acessar a lógica interna.
- **Tipo de teste:** **Funcional** (garante que a função faz o que promete) e de **Regressão** (garante que o código novo não quebrou o antigo).

# Instalação e configuração

- Ambiente [Node.js](#) com o gerenciador de pacotes NPM.
- Comando de instalação: `npm install --save-dev qunit`
- Configuração do script npm test no arquivo `package.json`.

# Estratégia de Testes

- Foi usado o **particionamento de equivalência** para definir os testes.
- As entradas foram divididas em grupos: números positivos, negativos e zero.
- Foi aplicada a análise de valor limite para testar o caso de borda mais crítico: a **divisão por zero**.

# O CÓDIGO

```
JS calculadora.js > ...
You, 1 hour ago | 1 author (You)
1  function somar(a, b) {
2    |   return a + b;
3  }
4
5  function subtrair(a, b) {
6    |   return a - b;
7  }
8
9  function multiplicar(a, b) {
10   |   return a * b;
11 }
12
13 function dividir(a, b) {
14   |   if (b === 0) {
15     |     throw new Error("nao eh possivel dividir por zero.");
16   }
17   |   return a / b;
18 }
19 module.exports = { somar, subtrair, multiplicar, dividir };
20
```

# Implementação dos Testes

- JUnit.module para organizar os testes por função.
- JUnit.test para definir cada caso de teste individual.
- Usamos `assert.equal` para validar resultados e `assert.thrwos` para validar o erro.



# Demonstração

- Vamos para a execução?

Obrigado!

---