



**INSTITUTO FEDERAL**

Mato Grosso do Sul

## Matcher (comparadores) – parte 1

- Realiza a comparação com o operador '===', ou seja, valor e tipo são comparados
- Deve ser utilizado para comparar valores de forma mais efetiva pelo fato de também verificar o tipo do objeto

```
describe('Suíte de testes do Matcher toBe', function(){
  var valorBooleano = true;
  var valorBooleanoCopia = valorBooleano;
  var valorBooleanoTexto = 'true';
  var obj1 = {valor: valorBooleano};
  var obj2 = {valor: valorBooleano};

  it('deve validar o uso do matcher toBe', function(){
    expect(valorBooleano).toBe(true);
    expect(valorBooleanoCopia).toBe(valorBooleano);
    expect(valorBooleano).not.toBe(valorBooleanoTexto);
    expect(valorBooleanoTexto).toBe('true');
    expect(obj1).not.toBe(obj2);
  });
});
```

## toEqual

- Realiza a comparação de dois elementos de modo muito similar ao toBe
- a diferença é que não compara o tipo do objeto somente o seu valor
- é recomendado seu uso para comparação de valores literais

```
describe('Suíte de testes do comparador doEqual', function(){
  var valorBooleano = true;
  var valorBooleano2 = true;
  var valorBooleanoCopia = valorBooleano;
  var valorBooleanoTexto = 'true';
  var obj1 = {valor: valorBooleano};
  var obj2 = {valor: valorBooleano};

  it('deve validar o uso do matcher toEqual', function(){
    expect(valorBooleano).toEqual(true);
    expect(valorBooleano).toEqual(valorBooleano2);
    expect(valorBooleanoCopia).toEqual(valorBooleano);
    expect(valorBooleano).not.toEqual(valorBooleanoTexto);
    expect(valorBooleanoTexto).toEqual('true');
    expect(obj1).toEqual(obj2);
  });
});
```

- Realiza a comparação utilizando expressões regulares
- Caso seja passada uma string como parâmetro, o comparador tentará encontrar o texto passado dentro do valor a ser comparado
- Expressões regulares em JavaScript sempre começam e terminam com barra '/'

```
describe('Suíte de testes do comparador toMatch', function(){  
  var textoComparar = 'Testes unitários com Jasmine';  
  it('deve validar o uso do matcher toMatch', function(){  
    expect(textoComparar).toMatch(/Jasmine/);  
    expect(textoComparar).toMatch("Jasmine");  
    expect(textoComparar).toMatch(/jasmine/i);  
    expect(textoComparar).not.toMatch(/Javascript/);  
    expect('12345-123').toMatch(/^\\d{5}-\\d{3}$/);  
    expect('10/03/2023').toMatch(/^\\d{2}\\\\/\\\\d{2}\\\\/\\\\d{4}$/);  
  });  
});
```

## toContain

- Realiza a busca por determinado item em um array
- Também pode ser usado para buscar uma substring dentro de uma string
- Não suporta busca por expressões regulares





```
describe('Suíte de testes do toContain',function(){  
  var nomes = ['João', 'Maria', 'Jonas'];  
  var nomesTexto = 'João Maria Jonas';  
  
  it('Deve validar o uso do matcher toContain', function(){  
    expect(nomes).toContain('João');  
    expect(nomesTexto).toContain('Maria');  
    expect(nomesTexto).toContain('Jon');  
    expect(nomes).not.toContain('Ciclano');  
    expect(nomes).not.toContain('maria');  
  });  
});
```

## toBeLessThan

- Compara se um valor numérico é menor do que outro valor
- Realiza a conversão para valor numérico antes da comparação, podendo o valor ser passado em formato texto
- Prefira usar 'toBeGreaterThan' ao invés de 'not.toBeLessThan', para deixar o código de mais fácil compreensão
- Para valores iguais utilize o 'toEqual'

```
describe('Suíte de testes do toBeLessThan', function() {  
  const PI = 3.1415;  
  var numero = 2;  
  
  it('deve validar o uso do matcher toBeLessThan', function() {  
    expect(numero).toBeLessThan(PI);  
    expect("1.2").toBeLessThan(PI);  
    expect(5).not.toBeLessThan(PI);  
    expect(PI).not.toBeLessThan(PI);  
  });  
});
```

## toBeGreaterThan

- Compara se um valor numérico é maior do que outro valor
- Realiza a conversão para o valor numérico antes da comparação, podendo o valor ser passado em formato texto
- Prefira usar 'toBeGreaterThan' ao invés de 'not.toBeLessThan' para deixar o código de mais fácil compreensão
- Para valores iguais utiliza o 'toEqual'

```
describe('Suíte de testes do toBeGreaterThan', function(){  
  const PI = 3.1415;  
  var numero = 4;  
  
  it('deve validar o uso do matcher toBeGreaterThan', function(){  
    expect(numero).toBeGreaterThan(PI);  
    expect('3.2').toBeGreaterThan(PI);  
    expect(2).not.toBeGreaterThan(PI);  
    expect(PI).not.toBeGreaterThan(PI);  
  });  
});
```