



# TESTE DE SOFTWARE 2

PEDRO LUCAS DALL'IGNA, HENRIQUE BONATTO, ESTEVÃO BONATTO

# Introdução

- JavaScript
- Jest

Nome

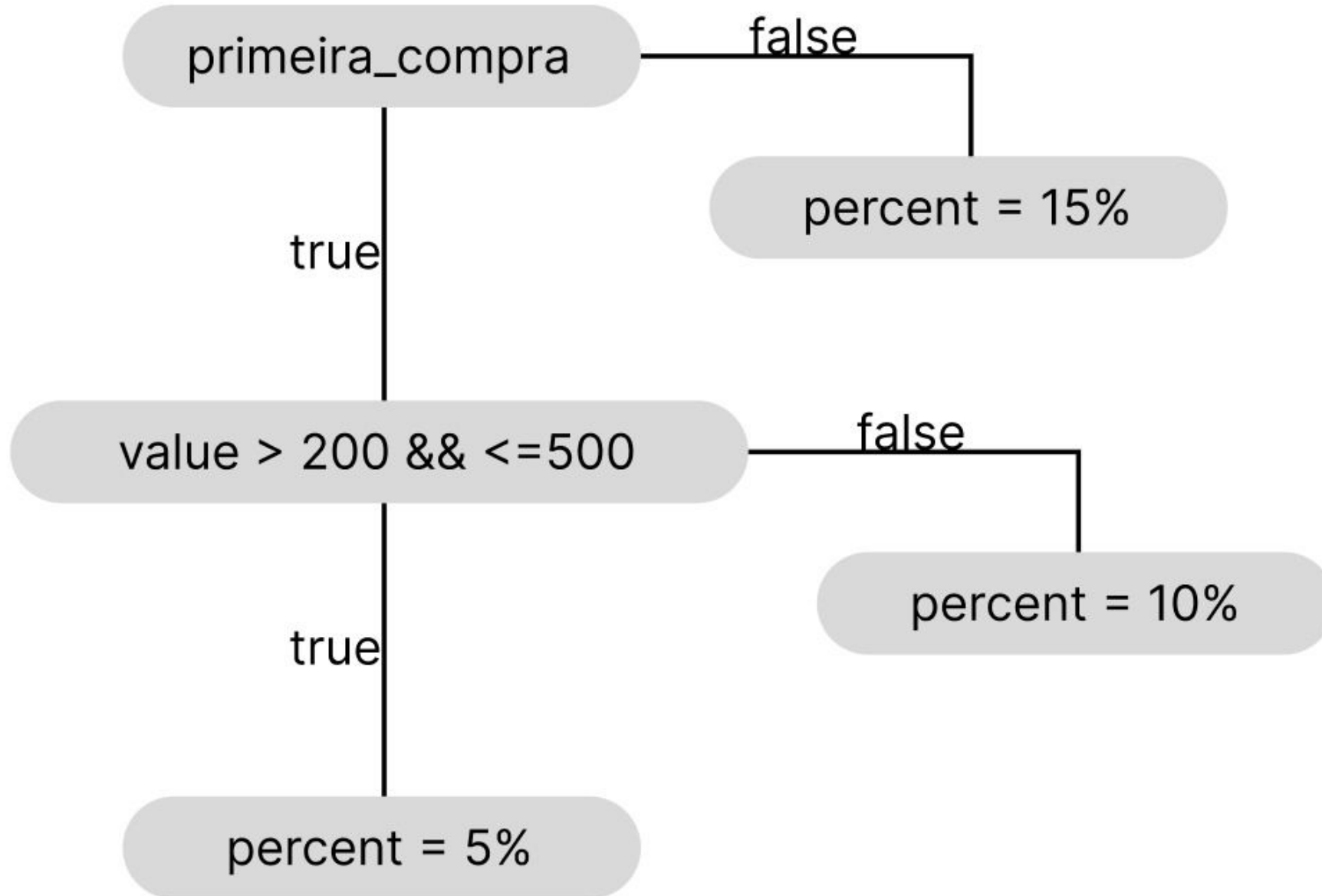
Vender

Valor

☒ Primeira compra

Confirmar

Lista de pedidos





```
src > utils > JS calcNewValue.js > ...  
1 function calcNewValue(value, percent) {  
2     const percent_value = percent / 100;  
3  
4     return value * (1 + percent_value);  
5 }  
6  
7 module.exports = { calcNewValue };  
8
```

```
src > test > JS calcNewValue.spec.js > ...  
1 const {calcNewValue} = require("../utils/calcNewValue.js");  
2  
3 describe("calcNewValue", () => {  
4     Run | Debug  
5     it("apply discount", () => {  
6         const newValue = calcNewValue(100, -15);  
7  
8         expect(newValue).toBe(85);  
9     })  
10  
11     Run | Debug  
12     it("apply increase", () => {  
13         const newValue = calcNewValue(100, 20);  
14  
15         expect(newValue).toBe(120);  
16     })  
17 })
```

src > utils > JS checkOrder.js > ...

```
❶ 1  const { calcNewValue } = require("../calcNewValue.js");
2
3  function checkOrder(first, value) {
4
5      if (first) {
6          return calcNewValue(value, -15)
7      } else {
8          if (value > 200 && value <= 500) {
9              return calcNewValue(value, -5)
10             } else if (value > 500) {
11                 return calcNewValue(value, -10)
12             }
13         }
14
15         return value;
16     }
17
18     module.exports = {checkOrder}
```

```
src > test > JS checkOrder.spec.js > ...
1  const {checkOrder} = require('../utils/checkOrder.js')
2
Run | Debug
3  describe('checkOrder', () => {
4
Run | Debug
5    describe('first buy', () => {
6      Run | Debug
7      it('apply 15% of discount', () => {
8        const newValue = checkOrder(true, 100)
9
10       expect(newValue).toBe(85)
11     })
12
Run | Debug
13     it('apply 15% of discount greater than 200', () => {
14       const newValue = checkOrder(true, 250)
15
16       expect(newValue).toBe(212.5)
17     })
18
Run | Debug
19     it('apply 15% of discount buy greater than 500', () => {
20       const newValue = checkOrder(true, 1000)
21
22       expect(newValue).toBe(850)
23     })
24  })
```

```
Run | Debug
25  describe('not first buy', () => {
26    Run | Debug
27    it('apply none discount', () => {
28      const newValue = checkOrder(false, 100)
29
30      expect(newValue).toBe(100)
31    })
32
Run | Debug
33    it('apply 5% of discount greater than 200', () => {
34      const newValue = checkOrder(false, 250)
35
36      expect(newValue).toBe(237.5)
37    })
38
Run | Debug
39    it('apply 10% of discount buy greater than 500', () => {
40      const newValue = checkOrder(false, 1000)
41
42      expect(newValue).toBe(900)
43    })
44  })
```



src > utils > JS validate.js > ...

```
1  const validateValue = (value) => {
2    const regexp = /^[0-9]/g
3    const newValue = value.replace(regexp, '')
4    return newValue
5  }
6
7  const validateName = (name) => {
8    const regexp1 = /^[^a-zA-Z]/g
9    const regexp2 = /[\\W]/g
10   const newName = name.replace(regexp1, '').replace(regexp2, '')
11   return newName
12 }
13
14 module.exports = { validateValue, validateName }
15
16
17
18
19
```



src > JS index.js > ...

```
1  const { checkOrder } = require("./utils/checkOrder.js");
2  const { validateValue, validateName } = require("./utils/validate.js");
3
4  const pedido = document.querySelector(".pedidos");
5  const nome = document.querySelector("#name");
6  const valor = document.querySelector("#value");
7  const confirm = document.querySelector("#submit");
8
9  const check = document.querySelector("#first-buy");
10
11 valor.addEventListener('input', (e) => {
12   const value = e.target.value
13   e.target.value = validateValue(value)
14 })
15
16 nome.addEventListener('input', (e) => {
17   const value = e.target.value
18   e.target.value = validateName(value)
19 })
20
21 confirm.addEventListener("click", () => {
22   if(check.checked) {
23     createPedido(nome.value, checkOrder(true, valor.value), true);
24   } else {
25     createPedido(nome.value, checkOrder(false, valor.value), false);
26   }
27 })
28
29 function createPedido(nome, valor, first) {
30   if(first) {
31     pedido.innerHTML += `<p>${nome}: ${valor} reais (primeira compra)</p>`;
32   } else {
33     pedido.innerHTML += `<p>${nome}: ${valor} reais</p>`;
34   }
35 }
36 }
37
```

# Repositório GITHUB

<https://github.com/PedroLDalligna/software-quality-2>