

**DISCIPLINA: PROJETO DE SISTEMAS APLICADO AS MELHORES PRÁTICAS EM QUALIDADE DE SOFTWARE E GOVERNANÇA DE TI**

**AULA:**

**18– TESTE DE SISTEMA COMPLETO APLICANDO BDD  
ESTUDOS DE CASO**

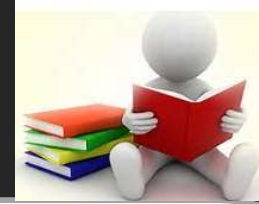
**PROFESSOR:**

**RENATO JARDIM PARDUCCI**

**PROFRENATO.PARDUCCI@FIAP.COM.BR**

**Solução das atividades de reforço da  
Aplicação do BDD**

## ESTUDO DE CASO SIMULADO



Tente imaginar que você tenha que criar a tela de login da FIAP.

Para criar a transação (front e backend, com o seu banco de dados), você quer primeiro, escrever seu teste de Sistema, o qual vai orientar a produção do software.

Monte os testes, segundo o TDD, aplicando a linguagem ubíqua, usando programação JAVA com CUCUMBER!

Nos testes, crie uma simulação onde o usuário digite um usuário e senha, se arrependa antes dar ENTER, depois digite (sobrescreva) o usuário e senha e tecle LOGIN, sendo que a combinação usuário/senha não existe.

***OBSERVAÇÃO: COMO A TELA DA FIAP JÁ EXISTE E VOCÊ VAI CRIAR OS TESTES APÓS A TELA PRONTA, FAÇA A INSPEÇÃO DA PÁGINA WEB PARA DESCOBRIR OS NOMES DOS CAMPOS A TESTAR!***

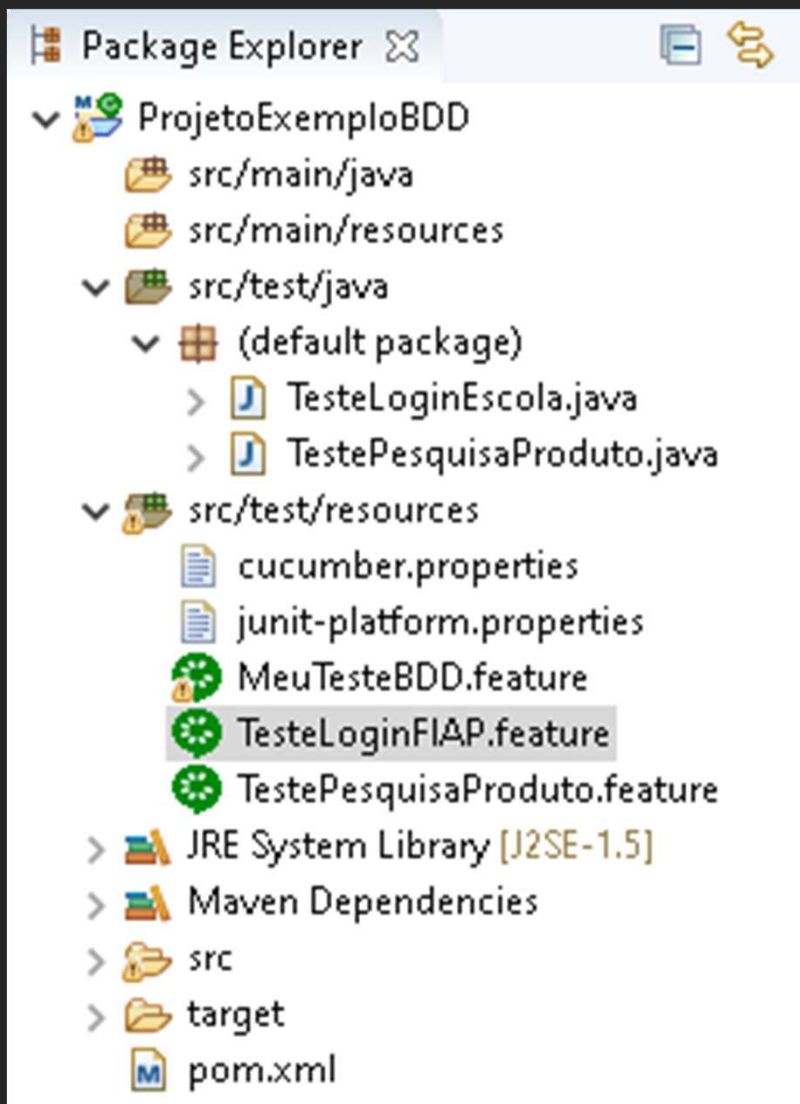
## SOLUÇÃO

- 1º) Acesse o portal do aluno da FIAP
- 2º) Inspecione a tela e descubra os nomes das variáveis ID e PWD do login (você pode usar o SELENIUM IDE ou o KATALON para gravar o uso da tela de login e descobrir os nomes das variáveis aplicadas na página WEB).
- 3º) Crie um arquivo .feature no src/test/resources do seu projeto MAVEN
- 4º) Construa o arquivo .feature com sentenças Before, Given, When, Then e o recurso de Examples para simular os 3 logins
- 5º) Gere o TestCase a partir da feature
- 6º) Edite o TestCase, colocando as variáveis e demais detalhes técnicos da tela acessada e botões acionados

*Resultado a seguir... (no exemplo, foi usado o recurso @datadriven e Examples do CUCUMBER).*

## SOLUÇÃO

Observe o TesteLoginFIAP.feature e a Classe TesteLoginEscola criada a partir da feature!



## SOLUÇÃO

Arquivo CUCUMBER da feature

```

TesteLoginFIAP.feature
1 Feature: TesteLogin
2   Quero validar o funcionamento da tela de login da instituicao de ensino
3
4   @datadriven
5 Scenario Outline: login sem sucesso
6   Given Acessei a pagina de login da instituicao de ensino
7   When Informar o meu ID com <idAluno>
8   And Informar a minha SENHA com <pwAluno>
9   And Acionar o botao de login
10  Then O login sera negado
11
12 Examples:
13   | idAluno | pwAluno |
14   | P2101  | banana |
15   | PF095  | laranja |
  
```

## SOLUÇÃO

Trecho da Classe de teste que contém o acesso ao portal da FIAP com a digitação da 1ª combinação de ID e Senha.

```
TesteLoginEscola.java
1 import java.util.concurrent.TimeUnit;
12
13 public class TesteLoginEscola {
14     WebDriver driver;
15     final String campoID = "//input[@id='usuario']";
16     final String campoPW = "//input[@id='senha']";
17     final String campoBotao = "//input[@id='a-login-btn']";
18
19     @Before ()
20     public void ajustar_driver() {
21         System.setProperty("webdriver.chrome.driver", "c:\\Users\\renat\\Documents\\Instaladores\\chromedriver.exe");
22         driver = new ChromeDriver();
23         driver.manage().window().maximize();
24         driver.manage().timeouts().implicitlyWait(5, TimeUnit.SECONDS);
25     }
26
27     @Given("Acessei a pagina de login da instituicao de ensino")
28     public void acessei_a_pagina_de_login_da_instituicao_de_ensino() {
29         driver.get("https://www2.fiap.com.br/");
30     }
31
32     // Primeira massa de teste
33     @When("Informar o meu ID com P2101")
34     public void informar_o_meu_id_com_p2101() {
35         driver.findElement(By.xpath(campoID)).sendKeys("P2101");
36     }
37
38     @When("Informar a minha SENHA com banana")
39     public void informar_a_minha_senha_com_banana() {
40         driver.findElement(By.xpath(campoPW)).sendKeys("banana");
41     }
}
```

## SOLUÇÃO

Trecho da Classe de teste que contém a correção do ID e Senha com a tentativa de LOGIN

```

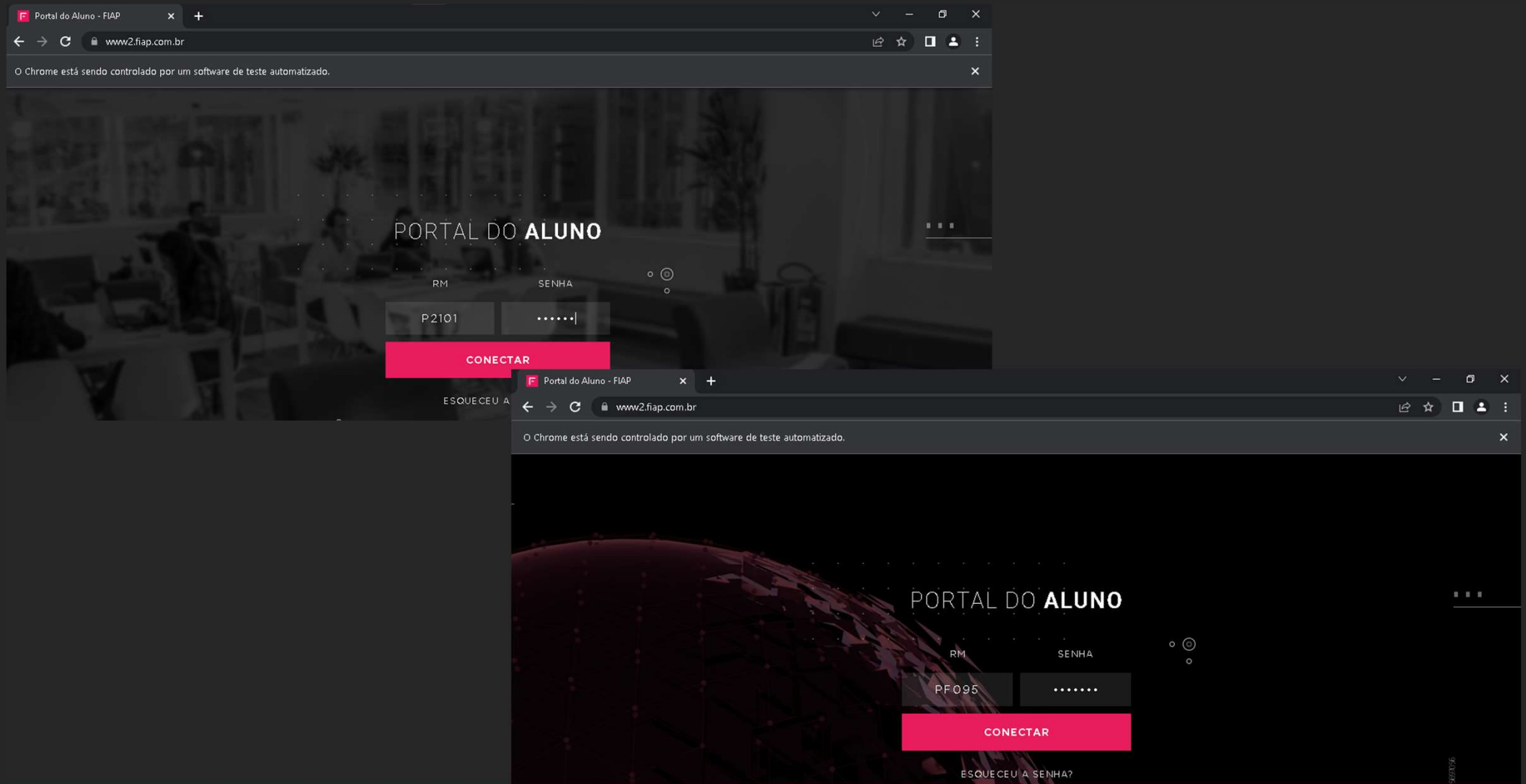
42
43
44 // Segunda massa de teste, corrigindo os dados de login
45 @When("Informar o meu ID com PF095")
46 public void informar_o_meu_id_com_pf095() {
47     driver.findElement(By.xpath(campoID)).sendKeys("PF095");
48 }
49
50 @When("Informar a minha SENHA com laranja")
51 public void informar_a_minha_senha_com_laranja() {
52     driver.findElement(By.xpath(campoPW)).sendKeys("laranja");
53 }
54
55 @When("Acionar o botao de login")
56 public void acionar_o_botao_de_login() {
57     driver.findElement(By.xpath(campoBotao)).sendKeys(Keys.ENTER);
58 }
59
60
61 @Then("O login sera negado")
62 public void o_login_sera_negado() {
63     // Pagina principal nao abre;
64 }
65 }

```



## SOLUÇÃO

Ao executar os testes, a janela de LOGIN da FIAP se abre duas vezes, mostrando as duas combinações de digitação de usuário e senha, nenhuma realizando o login com sucesso.



## ESTUDO DE CASO SIMULADO



Agora, considere que você tenha que criar uma calculadora digital que funciona na WEB.

Assim como no caso anterior, para criar a transação, você quer primeiro escrever seu teste de Sistema, o qual vai orientar a produção do software.

Monte os testes, segundo o TDD, aplicando a linguagem ubíqua, usando programação JAVA com CUCUMBER!

*OBSERVAÇÃO: USE UM SITE DE CALCULADORA NA WEB QUE JÁ EXISTA PARA FAZER O EXERCÍCIO, DADO QUE O NOSSO OBJETIVO NÃO É A PROGRAMAÇÃO – ESSE SITE SERVIRÁ COMO SE FOSSE O “PROTÓTIPO” DA APLICAÇÃO. FAÇA A INSPEÇÃO DA PÁGINA WEB PARA DESCOBRIR OS NOMES DOS CAMPOS A TESTAR!*

## SOLUÇÃO

1º) Acesse o site de calculadora do seu protótipo (no caso, o exemplo utiliza o site <https://www.calculator.net/>)

2º) Inspecione a tela e descubra os nomes das variáveis ID e PWD do login (você pode usar o SELENIUM IDE ou o KATALON para gravar o uso da tela de login e descobrir os nomes das variáveis aplicadas na página WEB).

3º) Crie um arquivo .feature no src/test/resources do seu projeto MAVEN

4º) Construa o arquivo .feature com sentenças Before, Given, When, Then e o recurso de Examples para simular uma soma

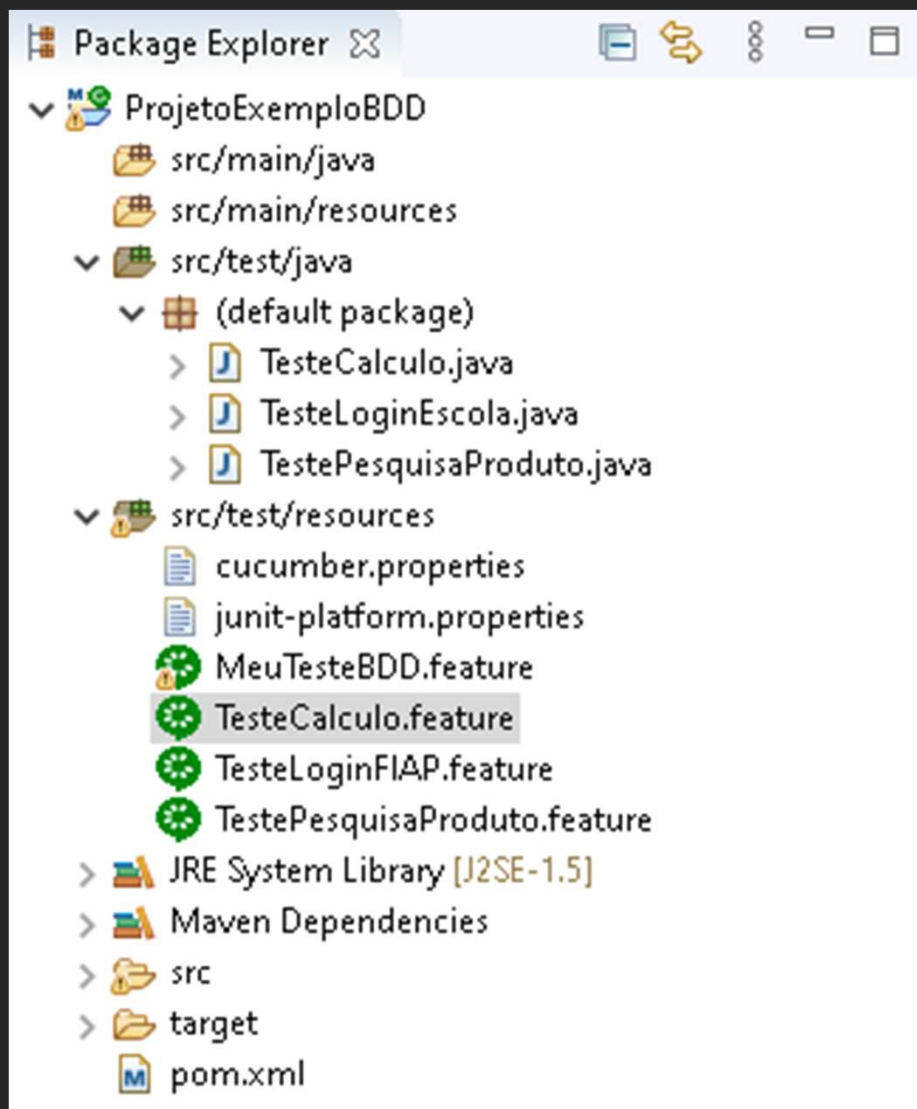
5º) Gere o TestCase a partir da feature

6º) Edite o TestCase, colocando as variáveis e demais detalhes técnicos da tela acessada e botões acionados

*Resultado a seguir... (no exemplo, foi usado o recurso @datadriven e Examples do CUCUMBER).*

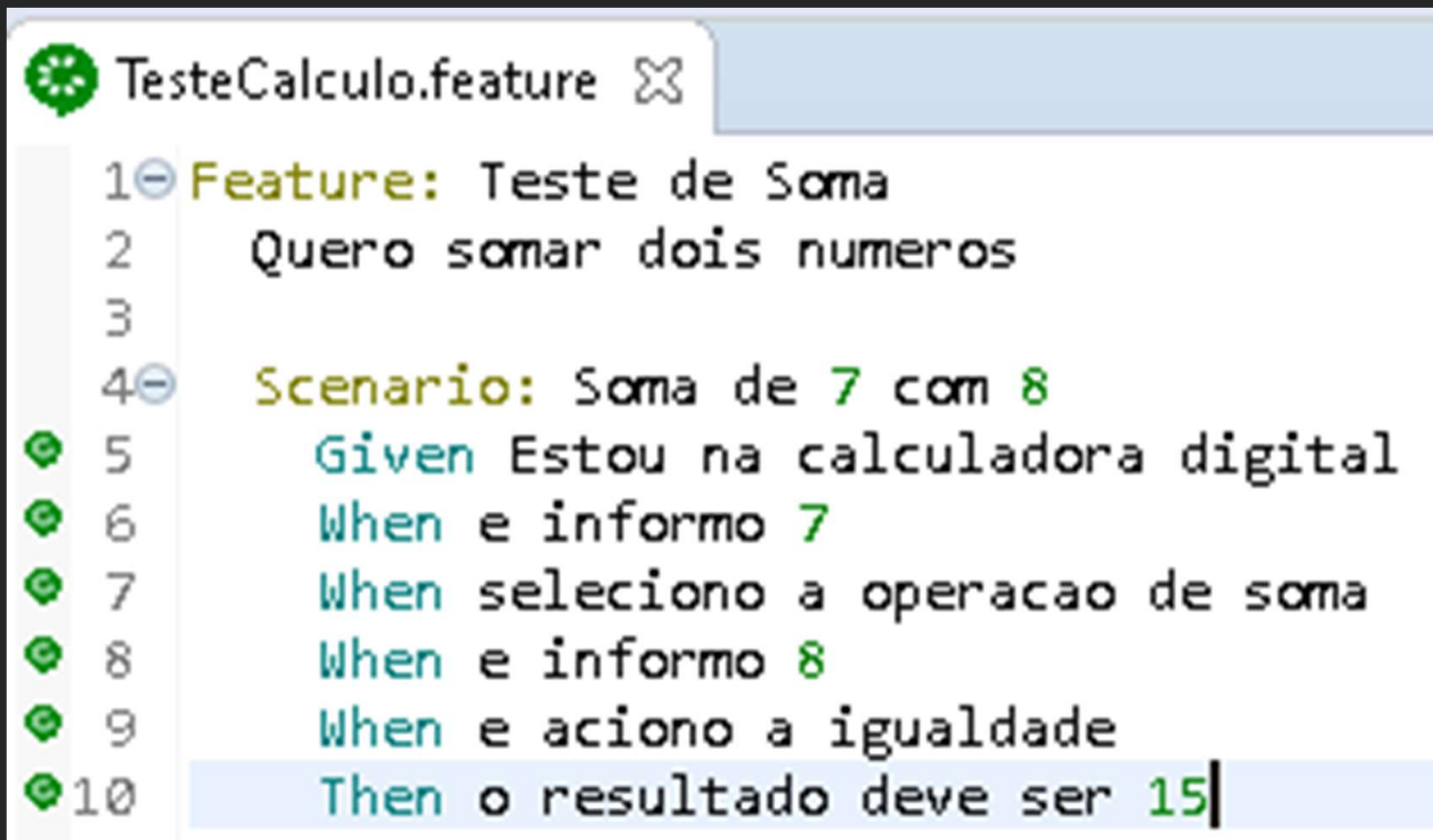
## SOLUÇÃO

Observe o TesteCalculo.feature e a Classe TesteCalculo criada a partir da feature!



## SOLUÇÃO

Arquivo .feature:



```

TesteCalculo.feature ✕
1 Feature: Teste de Soma
2   Quero somar dois numeros
3
4 Scenario: Soma de 7 com 8
5   Given Estou na calculadora digital
6   When e informo 7
7   When seleciono a operacao de soma
8   When e informo 8
9   When e aciono a igualdade
10  Then o resultado deve ser 15
    
```

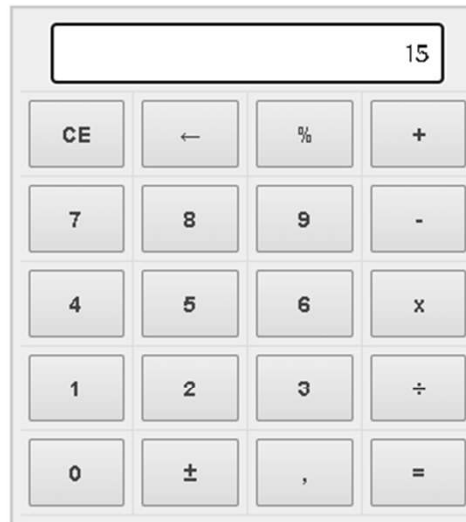
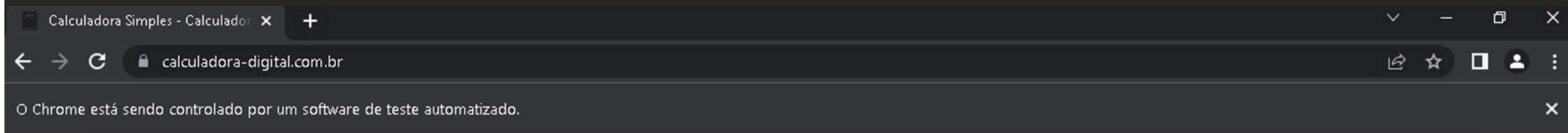
## SOLUÇÃO

## Classe de teste:

```
TesteCalculo.java
1 import java.util.concurrent.TimeUnit;
12
13 public class TesteCalculo {
14
15     WebDriver driver;
16     final String primeiroValor = "//input[@value='7']";
17     final String segundoValor = "//input[@value='8']";
18     final String campoBotaoSoma = "//input[@value='+']";
19     final String campoBotaoIgual = "//input[@value='=']";
20     final String campoResultado = "Tela deve exibir 15";
21
22     @Before ()
23     public void ajustar_driver() {
24         System.setProperty("webdriver.chrome.driver", "c:\\Users\\renat\\Documents\\Instaladores\\chromedriver.exe");
25         driver = new ChromeDriver();
26         driver.manage().window().maximize();
27         driver.manage().timeouts().implicitlyWait(5, TimeUnit.SECONDS);
28     }
29
30     @Given("Estou na calculadora digital")
31     public void estou_na_calculadora_digital() {
32         driver.get("https://calculadora-digital.com.br/");
33     }
34
35     @When("e informo 7")
36     public void e_informo_7() {
37         driver.findElement(By.xpath(primeiroValor)).sendKeys(Keys.ENTER);
38     }
39
40     @When("seleciono a operacao de soma")
41     public void seleciono_a_operacao_de_soma() {
42         driver.findElement(By.xpath(campoBotaoSoma)).sendKeys(Keys.ENTER);
43     }
44
45     @When("e informo 8")
46     public void e_informo_8() {
47         driver.findElement(By.xpath(segundoValor)).sendKeys(Keys.ENTER);
48     }
49
50     @When("e aciono a igualdade")
51     public void e_aciono_a_igualdade() {
52         driver.findElement(By.xpath(campoBotaoIgual)).sendKeys(Keys.ENTER);
53     }
54
55     @Then("o resultado deve ser 15")
56     public void o_resultado_deve_ser_15() {
57         System.out.println(campoResultado);
58     }
59 }
```

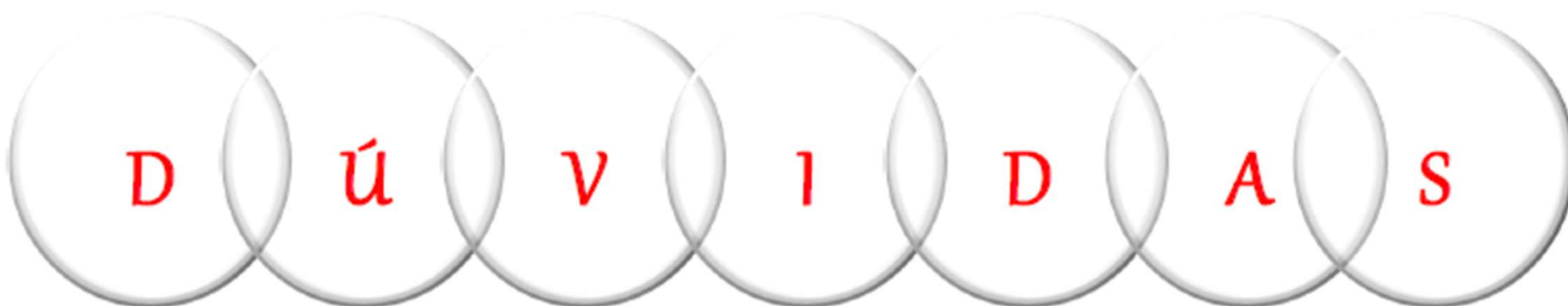
## SOLUÇÃO

Resultado da execução deve ser conferido visualmente pelo testador:



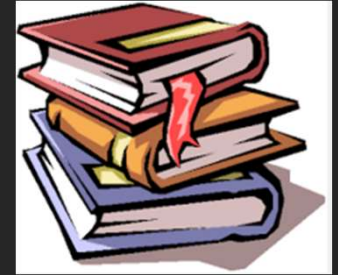
Calculadora Online, ideal para iPad, iPhone ou Desktop.







## Referência bibliográficas



### BIBLIOGRAFIA :

- MOLINARI, Leonardo. Testes de Software – Produzindo Sistemas Melhores e Mais Confiáveis, 4a. Edição. Editora Erica, 2013.
- MOLINARI, Leonardo. Inovação e Automação de Testes de Software, 1ª edição. Érica, 2010.
- CMMi V3. SEI - Software Engineering Institute., USA, 2007. Disponpivel na biblioteca online da Carnegie Melon University.
- Reis, Luís Filipe Souza. ISO 9000/Auditorias de sistemas da qualidade.Editora: Érica, 1995.

## TESTE DE SOFTWARE

**Continua na próxima aula...**

PROFESSOR:  
**RENATO JARDIM PARDUCCI**

PROFRENATO.PARDUCCI@FIAP.COM.BR