Alumna: Macedo Huaman Vanessa Mayra

Laboratorio 5: Pruebas Funcionales y xUnit

Junit

1. Implementar Casos de prueba para la funcionalidad "Percentage Calculator"

Diseñar casos de prueba: Entradas, proceso, Resultado Esperado

% Valor	of Valor	RESULTADO	RESULTADO ESPERADO	
10	100	10	10	☑
10	-100	-10	-10	☑

2. Implementar los scripts de prueba (e.g. JAVA - @Test): 1 método x TEST_CASE

```
@Test
public void testGoogleTitle() {
    // Use assert to verify Title
    Boolean verifyTitle = driver.getTitle().equalsIgnoreCase("Calculator.net: Free Online Calculators -
    assertNotNull(verifyTitle);
    assertTrue(verifyTitle);
}
```

```
@Test
public void testGoogleOperations() {
    driver.manage().timeouts().implicitlyWait(20, TimeUnit.SECONDS);
```

Script 1

```
// Click on Math Calculators
driver.findElement(By.xpath("//*[@id=\"contentout\"]/table/tbody/tr/td[3]/div[2]/a")).click();

// Click on Percent Calculators
driver.findElement(By.xpath("//*[@id=\"content\"]/table[2]/tbody/tr/td/div[3]/a")).click();

// Enter value 10 in the first number of the percent Calculator
driver.findElement(By.id("cpar1")).sendKeys("10");

// Enter value 100 in the second number of the percent Calculator
driver.findElement(By.id("cpar2")).sendKeys("100");

// Click Calculate Button
driver.findElement(By.xpath(".//*[@id = 'content']/table/tbody/tr[2]/td/input[2]")).click();

// Get the Result Text based on its xpath
String result =
   driver.findElement(By.xpath(".//*[@id = 'content']/p[2]/font/b")).getText();

// Verify the result
String ExpectedResult = "10";
Assert.assertEquals(ExpectedResult, result);
```

Script 2

```
// Click on Percent Calculators
driver.findElement(By.xpath("//*[@id=\"breadcrumbs\"]/span[3]/a")).click();

// Enter value 30 in the first number of the percent Calculator
driver.findElement(By.id("cpar1")).sendKeys("10");

// Enter value 100 in the second number of the percent Calculator
driver.findElement(By.id("cpar2")).sendKeys("-100");

// Click Calculate Button
driver.findElement(By.xpath(".//*[@id = 'content']/table/tbody/tr[2]/td/input[2]")).click();

// Get the Result Text based on its xpath
String result_ =
   driver.findElement(By.xpath(".//*[@id = 'content']/p[2]/font/b")).getText();

// Verify the result
String ExpectedResult_ = "-10";
Assert.assertEquals(ExpectedResult_, result_);
```

3. Implementacion de pre-condidiones o post-condiciones: setUp(), tearDown()

setUp()

```
@Before
public void setUp(){
    System.setProperty("webdriver.chrome.driver", "src/test/resources/drivers/chromedriver/chromedriver'
    driver = new ChromeDriver();
    driver.get("https://www.calculator.net");
    driver.manage().window().maximize();
}
```

tearDown()

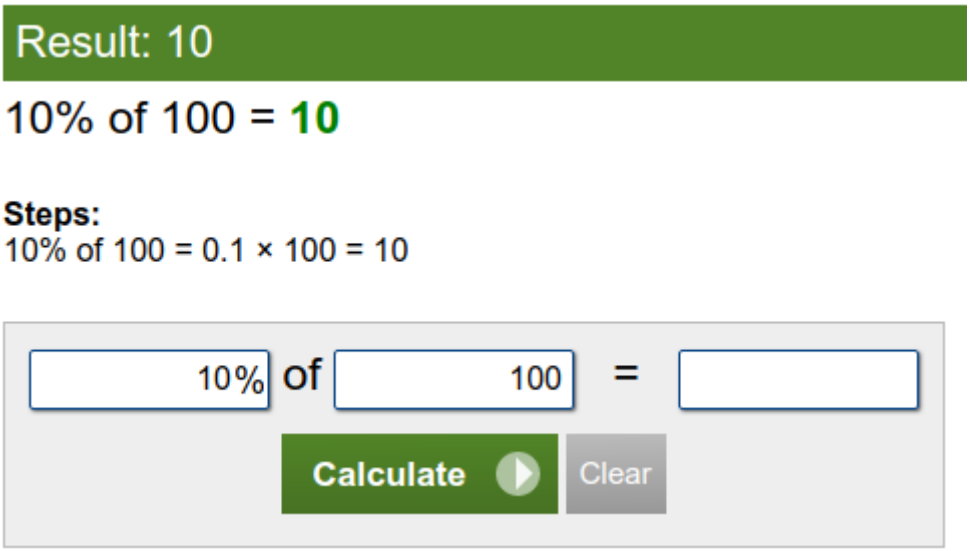
```
@After
public void tearDown() {
    driver.quit();
}
```

4. Uso de ASSERTS o EXPECTS para verificar el resultado esperado

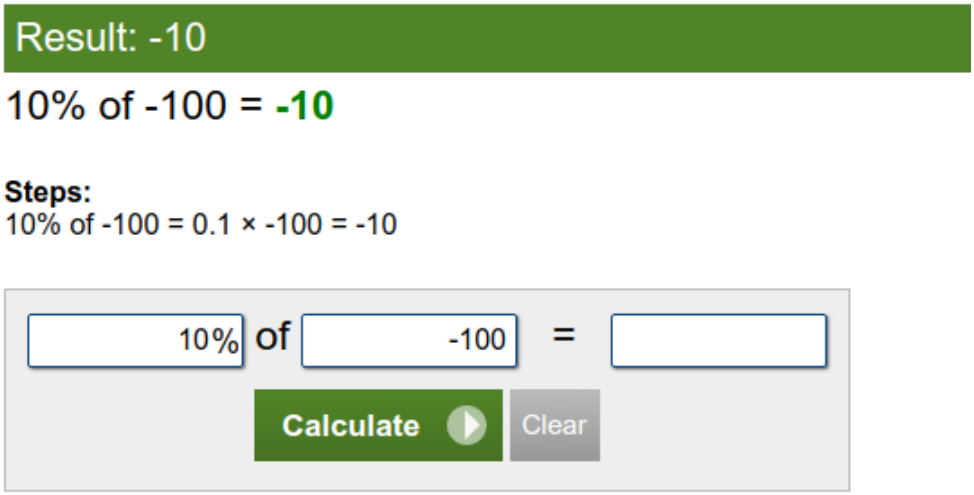
```
// Verify the result
String ExpectedResult_ = "-10";
Assert.assertEquals(ExpectedResult_, result_);
```

5. Ejecutando los scripts de prueba

Resultados de ejecución 1



Resultados de ejecución 2



6. Resultados del test