

Comandos de básicos

En este documento encontraras algunos comandos de Selenium WebDriver los cuales permiten realizar los scripts de las pruebas, en este documento se abordaran los comandos **Get**, **Navigate** y **WebElements**.

Comandos de navegador

Estos comandos ayudan a buscar información importante sobre la página o elementos.

Algunos de los comandos importantes que usted debe tener presentes son los siguientes:

Comando	Función	Ejemplo
<i>get("String")</i>	Recibe como parámetro la cadena de la URL. Permite cargar una página web en la ventana del navegador. Este comando usa la operación get del protocolo HTTP.	driver.get("https://www.google.com.co")
<i>getTitle()</i>	Captura el título de la página web actual.	System.out.println(driver.getTitle());
<i>getCurrentUrl()</i>	Captura una cadena que representa la URL que se encuentra abierta por el navegador.	System.out.println(driver.getCurrentUrl());
<i>getPageSource()</i>	Captura el código fuente de la página web actualmente cargada en el navegador.	System.out.println(driver.getPageSource());
<i>close()</i>	Cierra la ventana actual, si hay varias ventanas, cerrará la ventana actual que está	driver.close();

	activa y cierra el navegador si es la última ventana actualmente abierta.	
<i>quit()</i>	Finaliza la instancia del driver, cerrando todas las ventanas asociadas a éste que estén abiertas.	driver.quit();
<i>manage()</i>	<p>Habilita el acceso una serie de métodos de una interface que permiten hacer lo que se podría hacer en el menú de un navegador. Algunas de las opciones son:</p> <ul style="list-style-type: none"> • .fullscreen() • .getPosition() • .getSize() • .maximize() • .setPosition() • .setSize() • .implicitlyWait() • .pageLoadTimeout() • .setScriptTimeout() 	driver.manage().window().getSize() o driver.manage().timeouts().implicitlyWait()

Comandos de navegación

Comando	Función	Ejemplo
<i>navigate().to()</i>	Este método carga una nueva página web en el navegador actual. Hace lo	driver.navigate().to("https://www.google.com");

	mismo que driver.get(url)	
<i>navigate().forward()</i>	Este método navega a una página adelante en el historial de navegación.	driver.navigate().forward();
<i>navigate().back()</i>	Este método navega a una página atrás en el historial de navegación.	driver.navigate().back();
<i>navigate().refresh()</i>	Este método recarga la página actual.	driver.navigate().refresh();

Comandos WebElement

Comando	Función	Ejemplo
<i>element.clear()</i>	Si el elemento es un textBox , este comando limpiará el valor contenido en él. Solo funciona para objetos de tipo input y textarea .	driver.findElement(By.id("U serName")).clear();
<i>element.sendKeys("text")</i>	Este comando simula la entrada de texto, la cual puede ser su valor. Trabaja bien con elementos de tipo input y textarea .	driver.findElement(By.id("U serName")).sendKeys("Sem illeroQVision");
<i>element.click()</i>	Este comando simula el envío de un clic sobre un elemento. Hay algunas condiciones para poder dar clic sobre un elemento: debe ser visible y debe tener dimensiones	driver.findElement(By.linkT ext("Semillero QVision")).click();

	mayores a 0, es decir, Height > 0 y Width > 0.	
<i>element.isDisplayed()</i>	Este comando determina si un elemento es actualmente mostrado o no. Devuelve true o false .	boolean status = driver.findElement(By.id("U serName")).isDisplayed();
<i>element.isEnabled()</i>	Este comando determina si un elemento es actualmente activo o no. Devuelve true o false .	boolean status = element.isEnabled();
<i>element.isSelected()</i>	Este comando determina si un elemento está seleccionado o no. Devuelve true o false . Aplica para elementos de tipo input como: <ul style="list-style-type: none"> • Checkboxes • Drop Down List • Radio Buttons. 	boolean status = element.isSelected();
<i>element.submit()</i>	Este comando funciona mejor que el clic si el elemento es un formulario o un elemento dentro de un formulario.	element.submit();
<i>element.getText()</i>	Este comando recuperará el inner text visible de un elemento. Retorna el inner text del elemento, incluyendo sub-elementos sin espacios en blanco al principio o al final.	element.getText();