Playwright - Java Cheat Sheet - Browser Auto...

Author: Naveen Khunteta

Linkedin: https://www.linkedin.com/in/naveenkhunteta/

Playwright - Java Youtube Series: https://t.ly/hZ9Rh

Playwright is an open-source Node.js library for automating web browsers such as Chromium, Firefox, and WebKit through a single API.

It allows developers to write browser automation scripts in a simple and intuitive way using the programming language of their choice, including Java, JavaScript, Python, and others.

Playwright provides powerful features such as cross-browser testing, headless mode, request interception, and more, making it a popular tool for testing and scraping web applications.

Playwright is maintained by Microsoft and is designed to be fast, reliable, and easy to use.

Method/Property	Description
Playwright.create()	Creates a new instance of Playwright.
BrowserType.launch()	Launches a new browser instance.
Browser.newContext()	Creates a new browser context.
Context.newPage()	Creates a new page in the current context.
Page.navigate()	Navigates the current page to a new URL.
Page.querySelector()	Finds the first element that matches the specified select or.

Page.querySelectorAll()	Finds all elements that match the specified selector.
Page.waitForSelector()	Waits for the specified selector to appear on the page.
Page.waitForLoadState()	Waits for the page to reach the specified load state.
Page.title()	Gets the title of the page.
Page.url()	Gets the URL of the page.
Page.goBack()	Navigates the page back in history.
Page.goForward()	Navigates the page forward in history.
Page.reload()	Reloads the page.
Page.close()	Closes the page.
Page.setViewportSize()	Sets the size of the viewport.
Page.setExtraHTTPHeaders()	Sets extra HTTP headers to be sent with every request.
Page.setRequestInterception()	Enables or disables request interception.
Page.setOfflineMode()	Enables or disables offline mode.
Page.route()	Intercepts network requests and invokes the route handler.
Page.setUserAgent()	Sets the user agent for the page.
Page.waitForNavigation()	Waits for the page to finish navigating to a new URL.
Page.waitForPopup()	Waits for a popup window to appear.
Page.waitForDownload()	Waits for a download to finish.
Page.evaluate()	Executes a JavaScript function in the page context.
Page.evaluateHandle()	Evaluates a JavaScript expression in the page context and returns a handle to the result.
Page.addInitScript()	Injects a JavaScript file or script into the page context.

Page.exposeBinding()	Exposes a Java object as a JavaScript function in the page context.
Page.click()	Simulates a click event on an element.
Page.fill()	Sets the value of an input element.
Page.selectOption()	Selects an option from a dropdown menu.
Page.hover()	Simulates a hover event on an element.
Page.focus()	Sets focus on an element.
Page.press()	Sends a keyboard event to an element.
Page.type()	Types the specified text into an element.
Page.keyboard()	Provides methods for simulating keyboard input.
Page.mouse()	Provides methods for simulating mouse input.
Page.touchscreen()	Provides methods for simulating touchscreen input.
ElementHandle.innerText()	Gets the inner text content of an element.
ElementHandle.getAttribute()	Gets the value of the specified attribute for an element.

Method/Property	Description
Browser.newContext(options: Browser ContextOptions?)	Creates a new browser context.
Browser.close()	Closes the browser.
Browser.newPage(): Page	Creates a new page in the current context.
BrowserContext.pages(): List<Page>	Gets a list of all pages in the current context.
BrowserContext.clearCookies()	Clears all cookies in the current context.

BrowserContext.addCookies(cookies: List<Cookie>)	Adds cookies to the current context.
BrowserContext.clearPermissions()	Clears all permissions in the current context.
BrowserContext.setGeolocation(geolocation: Geolocation)	Sets the geolocation for the current context.
BrowserContext.exposeBinding(name: String, callback: BindingCallback)	Exposes a Java object as a JavaScript function in the current context.
BrowserType.launch(options: BrowserType.LaunchOptions?)	Launches a new browser instance.
BrowserType.connectOverCDP(endpointURL: String)	Connects to an existing browser instance using the Chrome DevTools Protocol.

Method/Property	Description
Playwright.chromium()	Returns a **BrowserType** instance for Chromium.
Playwright.firefox()	Returns a **BrowserType** instance for Firefox.
Playwright.webkit()	Returns a **BrowserType** instance for WebKit.
Playwright.selectors().register(name: String, script: String)	Registers a new selector engine with the specified name and JavaScript script.
Playwright.selectors().register(name: String, scriptPath: Path)	Registers a new selector engine with the specified name and JavaScript script file.
Playwright.selectors().register(name: String, script: URL)	Registers a new selector engine with the specified name and JavaScript script URL.

Method	Description
Browser.newContext()	Creates a new browser context.
Context.newPage()	Creates a new page in the current context.

Page.navigate()	Navigates the current page to a new URL.
Page.querySelector()	Finds the first element that matches the specified select or.
Page.querySelectorAll()	Finds all elements that match the specified selector.
ElementHandle.innerText()	Gets the inner text content of an element.
ElementHandle.getAttribute()	Gets the value of the specified attribute for an element.
ElementHandle.click()	Simulates a click event on the element.
ElementHandle.fill()	Sets the value of an input element.
ElementHandle.selectOption()	Selects an option from a dropdown menu.
ElementHandle.hover()	Simulates a hover event on the element.
ElementHandle.focus()	Sets focus on the element.
ElementHandle.press()	Sends a keyboard event to the element.
ElementHandle.type()	Types the specified text into the element.
ElementHandle.isVisible()	Returns true if the element is visible on the page.
ElementHandle.isDisabled()	Returns true if the element is disabled.
ElementHandle.isEditable()	Returns true if the element is editable.
ElementHandle.boundingBox()	Returns the bounding box of the element.
Page.waitForSelector()	Waits for the specified selector to appear on the page.
Page.waitForLoadState()	Waits for the page to finish loading.
Page.screenshot()	Takes a screenshot of the current page.

THANKS FOR
WATCHING

NAVEEN AUTOMATION LABS