Comprehensive Guide to Selenium WebDriver and SearchContext Methods

---

1. get()

- **Declared in:** WebDriver interface
- **Implemented by:** RemoteWebDriver class
- **Arguments:** Takes a URL as a String
- **Returns:** void
- **Purpose:** Launches a website

Example:

```
WebDriver driver = new ChromeDriver();
driver.get("https://www.example.com");
```

---

2. getTitle()

- **Declared in:** WebDriver interface
- **Implemented by:** RemoteWebDriver class
- **Arguments:** None
- **Returns:** Title of the page as String
- **Purpose:** Get the title of the current page

Example:

```
WebDriver driver = new ChromeDriver();
String title = driver.getTitle();
```

---

3. getPageSource()

- **Declared in:** WebDriver interface
- **Implemented by:** RemoteWebDriver class
- **Arguments:** None
- **Returns:** Source code of the page as String
- **Purpose:** Retrieve the HTML source of the current page

Example:

```
WebDriver driver = new ChromeDriver();
String pageSource = driver.getPageSource();
```

---

4. getCurrentUrl()

- **Declared in:** WebDriver interface
- **Implemented by:** RemoteWebDriver class
- **Arguments:** None
- **Returns:** URL of the current page as String
- **Purpose:** Get the current page URL

Example:

```
WebDriver driver = new ChromeDriver();
String currentUrl = driver.getCurrentUrl();
```

---

5. getWindowHandle()

- **Declared in:** WebDriver interface
- **Implemented by:** RemoteWebDriver class
- **Arguments:** None
- **Returns:** Handle value of the current browser window/tab (String)
- **Purpose:** Retrieve current window handle

Example:

```
WebDriver driver = new ChromeDriver();
String windowHandle = driver.getWindowHandle();
```

---

6. getWindowHandles()

- **Declared in:** WebDriver interface
- **Implemented by:** RemoteWebDriver class
- **Arguments:** None
- **Returns:** Set of all window handles (Set)
- **Purpose:** Retrieve handles of all open windows/tabs

Example:

```
WebDriver driver = new ChromeDriver();
Set<String> handles = driver.getWindowHandles();
```

---

7. close()

- **Declared in:** WebDriver interface
- **Implemented by:** RemoteWebDriver class
- **Arguments:** None
- **Returns:** void
- **Purpose:** Close the current browser window

Example:

```
WebDriver driver = new ChromeDriver();
driver.close();
```

---

8. quit()

- **Declared in:** WebDriver interface
- **Implemented by:** RemoteWebDriver class
- **Arguments:** None
- **Returns:** void
- **Purpose:** Closes all browser windows and terminates WebDriver session

Example:

```
WebDriver driver = new ChromeDriver();
driver.quit();
```

---

9. navigate()

- **Declared in:** WebDriver interface
- **Implemented by:** RemoteWebDriver class
- **Returns:** Navigation interface
- **Purpose:** Navigate through browser history

Example:

```
WebDriver driver = new ChromeDriver();
driver.navigate().to("https://www.google.com");
```

```
driver.navigate().back();
driver.navigate().forward();
driver.navigate().refresh();
```

---

10. switchTo()

- **Declared in:** WebDriver interface
- **Implemented by:** RemoteWebDriver class
- **Returns:** TargetLocator interface
- **Purpose:** Move focus to frames, alerts, or windows

---

i. switchTo().frame()

Used to switch focus to a specific frame.

Sub-Methods:

- `driver.switchTo().frame(String frameNameOrId);`
- `driver.switchTo().frame(int frameIndex);`
- `driver.switchTo().frame(WebElement frameElement);`
- `driver.switchTo().parentFrame();`
- `driver.switchTo().defaultContent();`

---

ii. switchTo().alert()

Handles browser alerts.

Alert Actions:

- `.accept()` – Click OK
- `.dismiss()` – Click Cancel
- `.getText()` – Get alert text
- `.sendKeys()` – Enter text into prompt alerts

Example:

```
Alert alert = driver.switchTo().alert();
System.out.println(alert.getText());
alert.accept();
```

alert.dismiss();
alert.sendKeys("Test Input");

---

iii. switchTo().window()

Switch between multiple tabs/windows.

Sub-Methods:

- `driver.switchTo().window(String windowHandle)`
- `driver.getWindowHandle()`
- `driver.getWindowHandles()`

Example Loop:

String mainWindow = driver.getWindowHandle();

for (String window : driver.getWindowHandles()) {
   driver.switchTo().window(window);
   if (!window.equals(mainWindow)) {
      driver.close();
   }
}

driver.switchTo().window(mainWindow);

---

iv. switchTo().activeElement()

Focuses on the currently active element.

Example:

WebElement activeElement = driver.switchTo().activeElement();
System.out.println(activeElement.getTagName());
activeElement.sendKeys("Testing active element");

---

v. driver.switchTo().newWindow(WindowType)

Purpose: Open a new tab or window.

Example:

```
driver.switchTo().newWindow(WindowType.TAB);
driver.switchTo().newWindow(WindowType.WINDOW);
```

11. manage()

- **Declared in:** WebDriver interface
- **Implemented by:** RemoteWebDriver class
- **Returns:** Options interface
- **Purpose:** Manage cookies, timeouts, browser window, logs

i. Cookies

Purpose: Handle browser cookies

Sub-Methods:

- `driver.manage().getCookies()`
- `driver.manage().getCookieNamed(String name)`
- `driver.manage().addCookie(Cookie cookie)`
- `driver.manage().deleteCookie(Cookie cookie)`
- `driver.manage().deleteCookieNamed(String name)`
- `driver.manage().deleteAllCookies()`

ii. Timeouts

Purpose: Define time limits for browser operations

Sub-Methods:

- `implicitlyWait(Duration duration)`
- `pageLoadTimeout(Duration duration)`
- `scriptTimeout(Duration duration)`

iii. Window Management

Used to manage browser state, size, and position.

Locating Elements with findElement()

i. By ID

Syntax:

driver.findElement(By.id("element_id"))

ii. By Name

Syntax:

driver.findElement(By.name("element_name"))

iii. By XPath

Syntax:

driver.findElement(By.xpath("xpath_expression"))

iv. By CSS Selector

Syntax:

driver.findElement(By.cssSelector("css_selector"))

v. By Class Name

Syntax:

driver.findElement(By.className("class_name"))