

23. Commands in Selenium Webdriver

Commands or **Methods** used in Selenium Webdriver are

1. Application commands
2. Conditional commands
3. Browser commands
4. Navigation commands
5. Wait commands

Application and Conditional commands

Application commands
<code>get()</code> - to open the application URL
<code>title</code> - to capture the title of current webpage
<code>current_url</code> - to capture current url of web page
<code>page_source</code> - to capture source code of the page

- **current_window_handle**: Retrieves the unique identifier for the current browser window.
- **window_handles**: Retrieves the unique identifiers for all open browser windows.

Conditional commands
<code>is_displayed()</code> - used for any elements
<code>is_enabled()</code> - used for any elements
<code>is_selected()</code> - especially radio buttons, check boxes

[WebdriverCommands.py](#)

```
import time
```

```
from selenium import webdriver
```

```
from selenium.webdriver.common.by import By
```

```
options = webdriver.ChromeOptions()
```

```
options.add_experimental_option("detach", True)
```

```
driver = webdriver.Chrome(options=options)
```

Get the URL

```
driver.get("https://demo.nopcommerce.com/register")
```

```
driver.maximize_window()
```

```
driver.implicitly_wait(3)
```

Get the title of the current page

```
print("Title:", driver.title)
```

Get the current URL

```
print("Current URL:", driver.current_url)
```

Get page source length

```
print("Length of the page source:", len(driver.page_source))
```

Find elements and check their display and enabled status

```
search_box = driver.find_element(By.XPATH, "//*[@id='small-searchterms']")
```

```
print("Search box display status:", search_box.is_displayed())
```

```
print("Search box enabled status:", search_box.is_enabled())
```

Check radio button selections

```
male_rd = driver.find_element(By.XPATH, "///input[@id='gender-male']")
female_rd = driver.find_element(By.XPATH, "///input[@id='gender-female']")
print("Before selection.....")
print("Male radio selected:", male_rd.is_selected()) # false
print("Female radio selected:", female_rd.is_selected()) # false
male_rd.click() # Select male radio button
print("After selecting male radio button.....")
print("Male radio selected:", male_rd.is_selected()) # true
print("Female radio selected:", female_rd.is_selected()) # false
female_rd.click() # Select female radio button
print("After selecting female radio button.....")
print("Male radio selected:", male_rd.is_selected()) # false
print("Female radio selected:", female_rd.is_selected()) # true
```

Check the status of the newsletter checkbox

```
newsletter_status = driver.find_element(By.XPATH, "///input[@id='Newsletter']").is_selected()
print("Status of Newsletter checkbox:", newsletter_status) # true
```

current window handle and window handles

```
print("Current Window ID:", driver.current_window_handle)
driver.find_element(By.XPATH, "///a[text()='Facebook']").click() # Opens new browser window
time.sleep(5) # Wait for new window to open
window_ids = driver.window_handles
print("Window IDs:", window_ids)
driver.quit()
```

Browser commands

- **close()** - close single browser window (where driver focused) i.e, **parent window**
- **quit()** - close multiple browser windows (this will kill the process).

[BrowserCommands.py](#)

```
from selenium import webdriver
options = webdriver.ChromeOptions()
options.add_experimental_option("detach", True)
driver = webdriver.Chrome(options=options)
driver.get("https://www.facebook.com")
driver.switch_to.new_window("window")
driver.get("https://www.amazon.com/")
driver.close()
driver.quit()
```

Navigational commands - used to control the browser's navigation.

- **back()** - go to previous page in the browser history. Same as "Back" button in the browser.
- **forward()** - go to next page in the browser history. Same as "Forward" button in the browser.
- **refresh()** - reloads the current page. Same as the "Refresh" button in the browser.

```
import time
from selenium import webdriver
options = webdriver.ChromeOptions()
options.add_experimental_option("detach", True)
driver = webdriver.Chrome(options=options)
driver.get("https://demowebshop.tricentis.com/")
driver.maximize_window()
driver.get("https://demo.nopcommerce.com/register")
```

Navigation commands

```
driver.back()
print(driver.title)
time.sleep(3)
driver.forward()
print(driver.title)
time.sleep(3)
driver.refresh()
print(driver.title)
driver.quit()
```

Note

- Conditional commands always returns a **boolean value** (True / False).
- To close a **specific browser window among multiple windows**, we use **browser-specific** commands.