

Complete Selenium Python Guide: WebDriver, WebElement & Tools

1. WebDriver - Common Methods

`driver.get(url)` - Navigate to URL

`driver.current_url` - Get current URL

`driver.title` - Get page title

`driver.page_source` - Get page HTML source

`driver.close()` - Close current window

`driver.quit()` - Quit the entire browser session

`driver.back()` - Go back in browser history

`driver.forward()` - Go forward in browser history

`driver.refresh()` - Refresh page

`driver.maximize_window()` - Maximize browser window

`driver.minimize_window()` - Minimize browser window

`driver.set_window_size(x, y)` - Set window dimensions

`driver.get_window_size()` - Get window dimensions

`driver.get_window_position()` - Get window position

`driver.set_window_position(x, y)` - Set window position

`driver.save_screenshot(path)` - Take screenshot

`driver.execute_script(js)` - Execute JavaScript

2. WebElement - Attributes and Methods

`element.text` - Get visible text

`element.tag_name` - Get HTML tag name

`element.size` - Get element dimensions

`element.location` - Get element coordinates

`element.rect` - Get element rect (size + location)

`element.is_displayed()` - Check if visible

`element.is_enabled()` - Check if enabled

`element.is_selected()` - Check if selected (checkbox/radio)

`element.get_attribute(attr)` - Get HTML attribute value

`element.get_property(prop)` - Get DOM property value

`element.send_keys(text)` - Type text

`element.clear()` - Clear input field

`element.click()` - Click element

`element.submit()` - Submit form

`element.value_of_css_property(prop)` - Get CSS property value

`element.screenshot(path)` - Take element screenshot

3. Locator Strategies (By)

`By.ID`

`By.NAME`

`By.XPATH`

`By.LINK_TEXT`

`By.PARTIAL_LINK_TEXT`

`By.TAG_NAME`

`By.CLASS_NAME`

`By.CSS_SELECTOR`

4. WebDriverWait and Expected Conditions

`wait = WebDriverWait(driver, 10)`

`element = wait.until(EC.presence_of_element_located((By.ID, 'id')))`

`EC.title_is(title)`

`EC.title_contains(text)`

EC.presence_of_element_located(locator)

EC.visibility_of_element_located(locator)

EC.invisibility_of_element_located(locator)

EC.element_to_be_clickable(locator)

EC.text_to_be_present_in_element(locator, text)

5. ActionChains (Mouse and Keyboard Actions)

actions = ActionChains(driver)

actions.click(element)

actions.double_click(element)

actions.context_click(element) # right-click

actions.drag_and_drop(source, target)

actions.move_to_element(element)

actions.send_keys(keys)

actions.perform() # Execute the chain

6. Handling Dropdowns with Select

select = Select(driver.find_element(By.TAG_NAME, 'select'))

select.select_by_index(index)

select.select_by_value(value)

select.select_by_visible_text(text)

select.deselect_all() # For multi-select dropdowns only