Selenium with Python: Detailed Cheat Sheet

1. WebDriver Methods

```
from selenium import webdriver
driver = webdriver.Chrome()
```

Method	Description	Example
get(url)	Opens a web page	driver.get('https://example.com')
close()	Closes current window	driver.close()
quit()	Closes all windows and ends session	driver.quit()
back()	Navigate back	driver.back()
forward()	Navigate forward	driver.forward()
refresh()	Reload current page	driver.refresh()
maximize_window()	Maximizes window	driver.maximize_window()
minimize_window()	Minimizes window	driver.minimize_window()
fullscreen_window()	Fullscreen window	driver.fullscreen_window()
current_url	Returns current URL	url = driver.current_url
title	Returns page title	t = driver.title
page_source	Returns HTML source	html = driver.page_source
get_window_size()	Returns dict with width and height	size = driver.get_window_size()
set_window_size(width, height)	Set window size	driver.set_window_size(1024,768)
get_window_position()	Returns window position	pos = driver.get_window_position()
set_window_position(x, y)	Set window position	driver.set_window_position(0,0)
switch_to.window(handle)	Switch to window	driver.switch_to.window(handle)
switch_to.frame(name_or_index)	Switch to iframe	driver.switch_to.frame('frame1')
switch_to.default_content()	Switch back to main page	driver.switch_to.default_content()

Method	Description	Example
switch_to.alert	Switch to alert	alert = driver.switch_to.alert
implicitly_wait(seconds)	Sets implicit wait	driver.implicitly_wait(10)
execute_script(script, *args)	Run JS	driver.execute_script('return document.title')
execute_async_script(script, *args)	Run async JS	driver.execute_async_script(...)
save_screenshot(filename)	Screenshot	driver.save_screenshot('screen.png')

2. WebElement Methods

```
element = driver.find_element(By.ID, 'username')
```

Method	Description	Example
click()	Click element	element.click()
submit()	Submit form	element.submit()
clear()	Clear input	element.clear()
send_keys(*value)	Type keys	element.send_keys('Hello')
text	Visible inner text	t = element.text
get_attribute(name)	Get attribute	val = element.get_attribute('id')
get_property(name)	Get property	val = element.get_property('checked')
is_displayed()	Is visible	element.is_displayed()
is_enabled()	Is enabled	element.is_enabled()
is_selected()	Is selected	element.is_selected()
tag_name	Get tag name	tag = element.tag_name
screenshot(filename)	Screenshot of element	element.screenshot('el.png')
rect	Location and size	element.rect
size	Element size	element.size
location	Element location	element.location

3. Finding Elements

Method	Description
find_element(by, value)	Returns first matching element
find_elements(by, value)	Returns list of matching elements

Locators using By: - ID, NAME, CLASS_NAME, TAG_NAME, LINK_TEXT, PARTIAL_LINK_TEXT, XPATH, CSS_SELECTOR

```
from selenium.webdriver.common.by import By
el = driver.find_element(By.ID, 'username')
```

---

4. ActionChains Methods

```
from selenium.webdriver import ActionChains
actions = ActionChains(driver)
```

Method	Description
click(on_element=None)	Click element or mouse
click_and_hold(on_element=None)	Click and hold mouse
context_click(on_element=None)	Right click
double_click(on_element=None)	Double click
drag_and_drop(source, target)	Drag source to target
drag_and_drop_by_offset(source, xoffset, yoffset)	Drag by offset
move_to_element(to_element)	Hover over element
move_by_offset(xoffset, yoffset)	Move mouse by offset
move_to_element_with_offset(to_element, x, y)	Move to point relative to element
release(on_element=None)	Release mouse button
send_keys(*keys)	Send keys to active element
send_keys_to_element(element, *keys)	Send keys to element
perform()	Execute actions

5. Waits

```
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC
WebDriverWait(driver, 10).until(EC.presence_of_element_located((By.ID,
'username')))
```

Common Expected Conditions: - presence_of_element_located - visibility_of_element_located - element_to_be_clickable - alert_is_present - title_contains - url_contains

6. Alert Methods

```
alert = driver.switch_to.alert
```

Method	Description
accept()	Click OK
dismiss()	Click Cancel
send_keys(text)	Enter text in prompt
text	Alert message

7. Select Methods

```
from selenium.webdriver.support.ui import Select
select = Select(driver.find_element(By.ID, 'dropdown'))
```

Method	Description
select_by_index(index)	Select option by index
select_by_value(value)	Select option by value
select_by_visible_text(text)	Select option by text
deselect_by_index(index)	Deselect by index
deselect_by_value(value)	Deselect by value

Method	Description
deselect_by_visible_text(text)	Deselect by text
deselect_all()	Deselect all
all_selected_options	List of selected options
options	List of all options
first_selected_option	First selected option

---

8. Keyboard Keys

```
from selenium.webdriver.common.keys import Keys
element.send_keys(Keys.ENTER)
```

Common Keys: ENTER, RETURN, TAB, ESCAPE, CONTROL, SHIFT, ARROW_DOWN, ARROW_UP, BACKSPACE

---

9. Miscellaneous

- driver.get_log('browser') → console logs
- driver.set_script_timeout(30) → JS timeout
- driver.set_page_load_timeout(30) → page load timeout
- driver.save_screenshot('screen.png') → Screenshot

This document can serve as a **detailed reference guide** for Selenium with Python, with examples included.