

WebDriver Methods For Selenium Using Python

Selenium Using Python Video Playlist

<https://www.youtube.com/playlist?list=PLfp-cJ6BH8u9sO59X4pnKwdbcMiAfOHXr>

In this session, let's look at webdriver methods. These methods are also known as browser methods because they are commands performed on the browser. Let's start by writing `driver = webdriver.Chrome(ChromeDriverManager().install())`. Do you see the red lines under `webdriver` and `ChromeDriverManager`?

```
driver = webdriver.Chrome(ChromeDriverManager().install())
```

In order, to continue, we must import. There are 2 ways to import. We can manually write the import statement or let the IDE import for us. In the last session, I manually wrote both import statements from selenium we import `webdriver`. Now, we see `webdriver` no longer has a red line. This time, I will let PyCharm import by hovering the red line for `ChromeDriverManager` then select the link to import. At this point, we have 2 import statements. Either way is okay.

```
from selenium import webdriver
from webdriver_manager.chrome import ChromeDriverManager

driver = webdriver.Chrome(ChromeDriverManager().install())
```

Now, we have access to Chrome. When I run, we see a blank screen on the Chrome browser. Notice, the window is restored down. That means the window is not maximized. Go back to the IDE then type `driver.window` and we see a lot of methods from `WebDriver`.

```
driver.wind
```

```
• window_handles  
• maximize_window(self)  
• set_window_size(self, width, height)  
• minimize_window(self)  
• current_window_handle  
• fullscreen_window(self)  
• get_window_rect(self)  
• get_window_position(self, windowHan
```

The 'm' in front of each name lets us know these are methods. We also see a 'p' in front of some names. The 'p' identifies a property decorator which defines 3 types of methods. A getter method, a setter method, and a deleter method. `window_handles` and `current_window_handle` are getter methods because they return a window handle. We can maximize the window, minimize the window, make the window a full screen. Also, we can set the window size by passing in width and height. Let's maximize the window and run again.

Now, the window is maximized. We can load a page by writing `driver.get()`. In this session, our application will be the [Internet](https://the-internet.herokuapp.com/). It has a lot of examples for automation. Copy the url. Pass the url into the `get()` method.

```
driver = webdriver.Chrome(ChromeDriverManager().install())  
driver.maximize_window()  
driver.get('https://the-internet.herokuapp.com/')
```

That's all we need to load a page. Type `driver.` to see the WebDriver methods. Let's get the name of the browser. How about the title of our application? `driver.title`. We see the url is the-internet.herokuapp.com but if you wanted your automation Test Script to return the url. Just write `driver.current_url`. Let's print the browser name, title, and `current_url`. Pass all 3 methods to the `print()` function. `print(driver.name)`, `print(driver.title)`, and `print(driver.current_url)`. Run and we see chrome, The Internet, and the url.

```
print(driver.name)
print(driver.title)
print(driver.current_url)
```

chrome

The Internet

<https://the-internet.herokuapp.com/>

Do you see how the url does not include www.? It's okay to not have www. But we must always include https://. Remove https:// and run then we do not see the page load but an `InvalidArgumentException` shows up. The same exception shows up if we add www. then run. A blank browser and `InvalidArgumentException`. Replace www. with https://.

An interview question may ask what's the difference between close and quit. Both are WebDriver methods and both close the browser. The difference is `close()` method only closes 1 window and that's the current window. `quit()` method closes all windows and quits the driver. Also, both methods return a different note in the console.

Let's say, we wrote our Test Script in a bad way by writing `driver.close` then `driver.delete_all_cookies` at the end. This is not a good way to write our Test Script because `driver.close` or `driver.quit` should be the last statement.

```
print(driver.name)
print(driver.title)
print(driver.current_url)
driver.close()
driver.delete_all_cookies()
```

Run and the console returns `InvalidSessionIdException`. The session is not valid because it was closed before trying to delete all cookies. Now, let's change close to quit then run.

```
print(driver.name)
print(driver.title)
print(driver.current_url)
driver.quit()
driver.delete_all_cookies()
```

The console shows our Test Script Failed to establish a new connection. I prefer to use driver.quit(). Remove driver.delete_all_cookies and run. The page loaded and closed without a problem. That's it for the WebDriver Methods.

Contact Info

- ✓ Website <https://www.rexjones2.com/>
- ✓ YouTube <https://www.youtube.com/c/RexJonesII/videos>
- ✓ Facebook <https://facebook.com/JonesRexII>
- ✓ Twitter <https://twitter.com/RexJonesII>
- ✓ GitHub <https://github.com/RexJonesII/Free-Videos>
- ✓ LinkedIn <https://www.linkedin.com/in/rexjones34/>