

ALL SELENIUM

Taking Screenshot using Python Selenium WebDriver

The Blog To Learn Selenium and Test Automation

```

self.options.set_preference("browser.download.folderList", 2);
self.options.set_preference("browser.download.manager.showWhenStarting", False);
self.options.set_preference("browser.download.defaultContentHandler", "application/octet-stream,application/vnd.ms-excel");
self.driver = webdriver.Firefox(firefox_options=self.options);
driver(self);

```

Taking Screenshot using Python Selenium WebDriver

[admin](#) [Python Selenium Test Automation](#) [January 27, 2018](#) | [3](#)

Page Contents

1. save_screenshot()
2. get_screenshot_as_file()
3. get_screenshot_as_png()

Hello, welcome to Python Selenium tutorials and in this post let's see how to capture Screenshot in Python Selenium Webdriver

- Ability to take screenshots is most important and desirable feature for bug analysis.
- Sometimes, apart from bug analysis, one may want to capture screenshots to see the flow of test steps.
- Screenshots help automation testers a lot when test cases fails, one can identify what went wrong in test script or application.

Selenium can take screenshots during execution and save it in a file. We need to type cast WebDriver instance to TakesScreenshot in case of Java/Selenium. Where as Python/Selenium directly calls methods to take screenshot and these methods support screenshot file as '.png'.

WebDriver offers total **three APIs** to take screenshot of a web page.

1. save_screenshot('filename')
2. get_screenshot_as_file('filename')
3. get_screenshot_as_png()

First two APIs are used to take and store screenshots as '.png' files.

Third API, get_screenshot_as_png(), returns a binary data. This binary data will create an image in memory and can be useful if we want to manipulate before saving it.



An important note to store screenshots is that save_screenshot('filename') and get_screenshot_as_file('filename') will work only when extension of file is '.png'. Otherwise content of the screenshot can't be viewed and Python throws a warning message.

Python



```

1 /usr/local/lib/python2.7/dist-packages/selenium/webdriver/remote/webdriver.py:907: UserWarning: name used for saved screenshot does
not match file type. It should end with a '.png' extension type. It should end with a '.png' extension, UserWarning

```

save_screenshot()

Python



```

1 from selenium import webdriver
2
3 driver = webdriver.Firefox(executable_path='[Browser-Driver-Path]')

```

ALL SELENIUM

```
5
6 driver.find_element_by_id('username').send_keys('admin')
7 driver.find_element_by_id('password').send_keys('admin')
8 driver.find_element_by_id('login').click()
9
10 driver.save_screenshot('sample_screenshot_1.png')
```

get_screenshot_as_file()

```
Python
1 from selenium import webdriver
2
3 driver = webdriver.Firefox(executable_path=[Browser Driver Path])
4 driver.get(['URL to Open'])
5
6 driver.find_element_by_id('username').send_keys('admin')
7 driver.find_element_by_id('password').send_keys('admin')
8 driver.find_element_by_id('login').click()
9
10 driver.get_screenshot_as_file('sample_screenshot_2.png')
```

get_screenshot_as_png()

```
Python
1 from selenium import webdriver
2 import StringIO
3 from PIL import Image
4
5 driver = webdriver.Firefox(executable_path=[Browser Driver Path])
6 driver.get(['URL to Open'])
7
8 driver.find_element_by_id('username').send_keys('admin')
9 driver.find_element_by_id('password').send_keys('admin')
10 driver.find_element_by_id('login').click()
11
12 screenshot = driver.get_screenshot_as_png()
13
14 size = (0, 0, 680, 400)
15 image = Image.open(StringIO.StringIO(screenshot))
16 region = image.crop(size)
17 region.save('sample_screenshot_3.jpg', 'JPEG', optimize=True, quality=95)
```

Here we are taking screenshot, cropping it to a particular size, and storing it a '.jpg' file named sample_screenshot_3.jpg.



PYTHON SELENIUM

SCREENSHOT

[« Previous: Test Automation: Reduce time to analyze and report failures](#)[Next: Capture screenshot of an Element using Python Selenium WebDriver »](#)

3 Responses

ALL SELENIUM



March 26, 2019 at 9:29 pm

Hi,

hm, your post is interesting but what I am missing is, how to make a screenshot from the whole entire page, as you can do it with Phantom JS. Unfortunately, Phantom JS is depreciated, so whats the alternative?

Another thing... where do you get the library "StringIO"? It seems, that it is not included in the regular anaconda data base or at PyPI, which is the reason I am not able to reproduce your examples. This makes your post interesting but somehow weak on the other side.

Greetings.

[Reply](#)

admin

March 26, 2019 at 10:48 pm

Hi, Thank you for the comment. I will check these two items and get back.

[Reply](#)

Kevin B

July 6, 2019 at 12:04 am

Also interested in this response.

[Reply](#)

Leave a Reply

Your email address will not be published. Required fields are marked *

Comment

Name *

Email *

☐ Save my name, email, and website in this browser for the next time I comment.[Post Comment](#)

[Python Selenium all mouse actions using ActionChains](#)

[TestNG Annotations 101](#)

[How to debug tests running on Docker containers](#)

ARCHIVES

[January 2019](#)

[December 2018](#)

[November 2018](#)

[October 2018](#)

[September 2018](#)

[August 2018](#)

[June 2018](#)

[May 2018](#)

[April 2018](#)

[March 2018](#)

[February 2018](#)

[January 2018](#)

[December 2017](#)

[November 2017](#)

CATEGORIES

[General](#)

[Java Selenium](#)

[Python](#)

[Python Selenium](#)

[Test Automation](#)

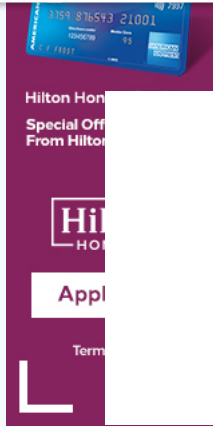
[Testing](#)

[TestNG](#)

[Tools](#)



ALL SELENIUM



© 2020 All Selenium. Built using WordPress and the Mesmerize Theme