

# Capture screenshot using Python Selenium WebDriver

## Table of content

- [Screenshot in selenium python](#)
- [Take element Screenshot](#)

## Screenshot in selenium python

Test cases may fail while executing the test suite in selenium automation. When manual tester faces any discrepancy in the verification, i.e. when there is a difference in actual and expected values, manual testers tend to take a screenshot as proof for failure.

**Failures may occur because of below reasons:**

- Actual and Expected values are not matching
- When there is no element
- When the page takes more time to load
- When an unexpected alert comes in to focus
- When there are Assertion issues

We can take a screenshot of a webpage in selenium python using the **save\_screenshot()** function; **save\_screenshot()** will take the complete page as a screenshot.

```
from selenium import webdriver
driver = webdriver.Firefox(executable_path='[Browser Driver Path]');
driver.get('https://www.google.co.in');
driver.save_screenshot("screenshot.png");
```

## Take element Screenshot

Selenium Python does not provide any methods to take a screenshot of an element, so we will be using the PIL library to crop the element out of the webpage.

```
from selenium import webdriver
from PIL import Image
driver = webdriver.Firefox(executable_path='[Browser Driver Path]');
driver.get('https://www.google.co.in');
element = driver.find_element_by_name("q");
location = element.location;
size = element.size;
driver.save_screenshot("pageImage.png");
x = location['x'];
y = location['y'];
width = location['x']+size['width'];
height = location['y']+size['height'];
im = Image.open('pageImage.png')
im = im.crop((int(x), int(y), int(width), int(height)))
im.save('element_image.png')
```

-----END-----

Muntasir Abdullah Mizan  
muntasir.abdullah01@gmail.com