# TestWorkspace

For my test project to write the automation test case for creating an account in the test.crowdstreet.com website, I am using Selenium WebDriver’s Python bindings with Google Chrome and ChromeDriver.

I am using Selenium which is an open-source, web-based automation tool. I chose Python because it’s a very easy language to use and Python APIs make it easy to connect with a browser through Selenium Web driver. Selenium can send the standard Python commands to different browsers such as Chrome, Firefox, IE on different operating systems to perform different tasks on the browser.

This test workspace contains files to demonstrate step-by-step process to write the automation test the python PyCharm IDE.

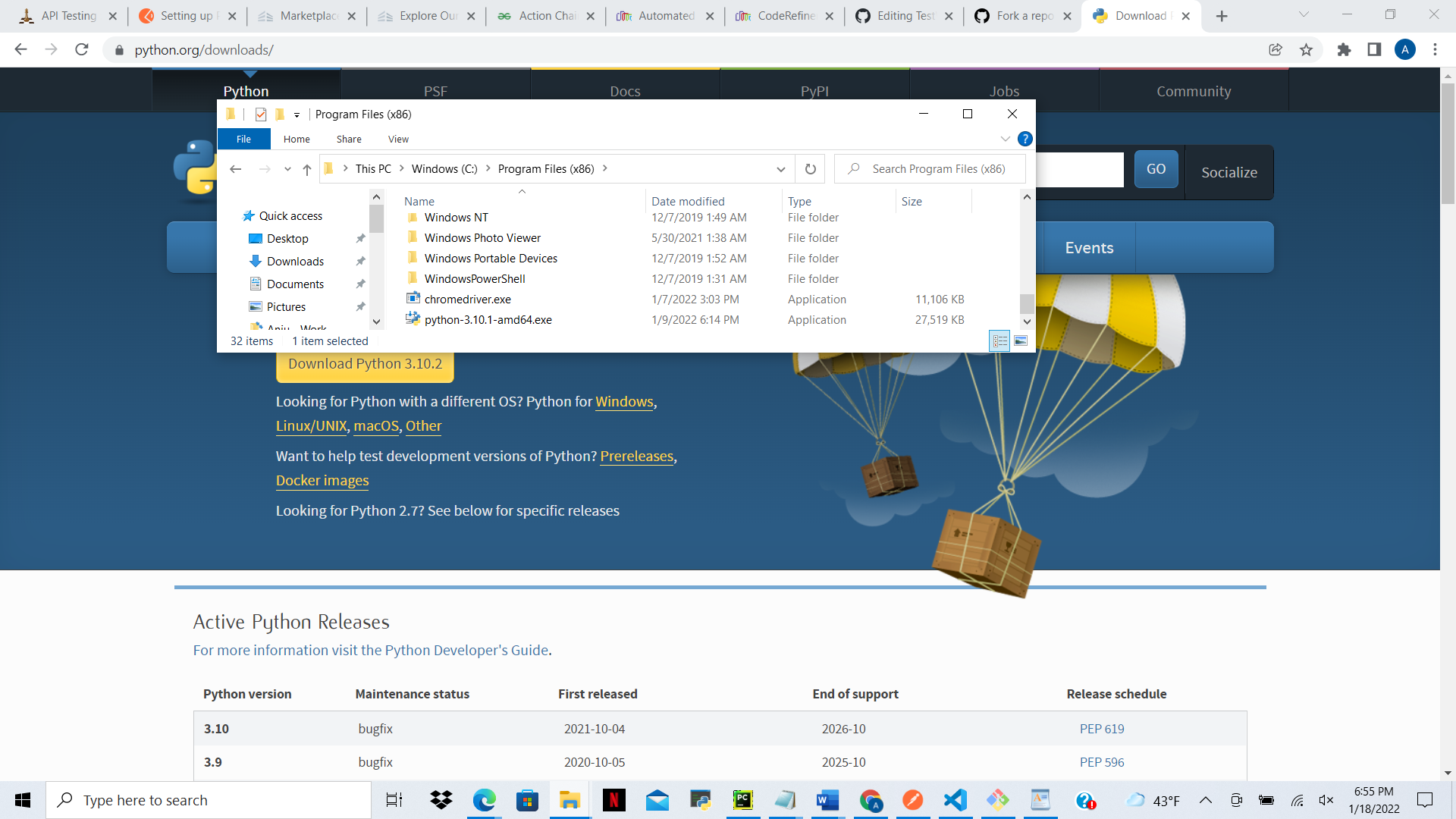
Selenium project using PyCharm IDE:

Pre-Requisites:

* Python on Windows.
* PyCharm IDE.
* Selenium Webdriver Packages
* Chrome Driver

Python Installation:

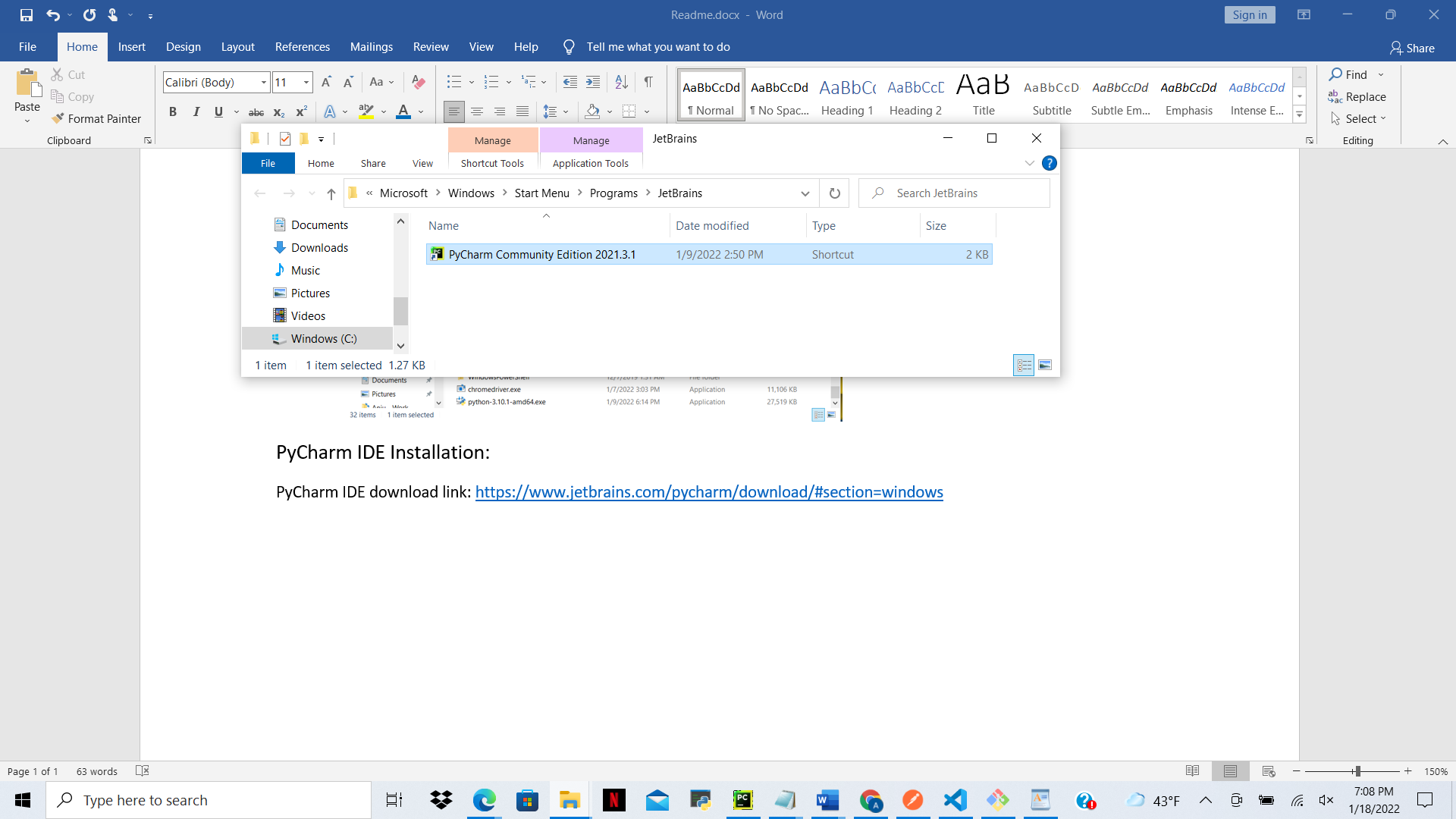
Python download link - <https://www.python.org/downloads/>



PyCharm IDE Installation:

Install the PyCharm community edition.

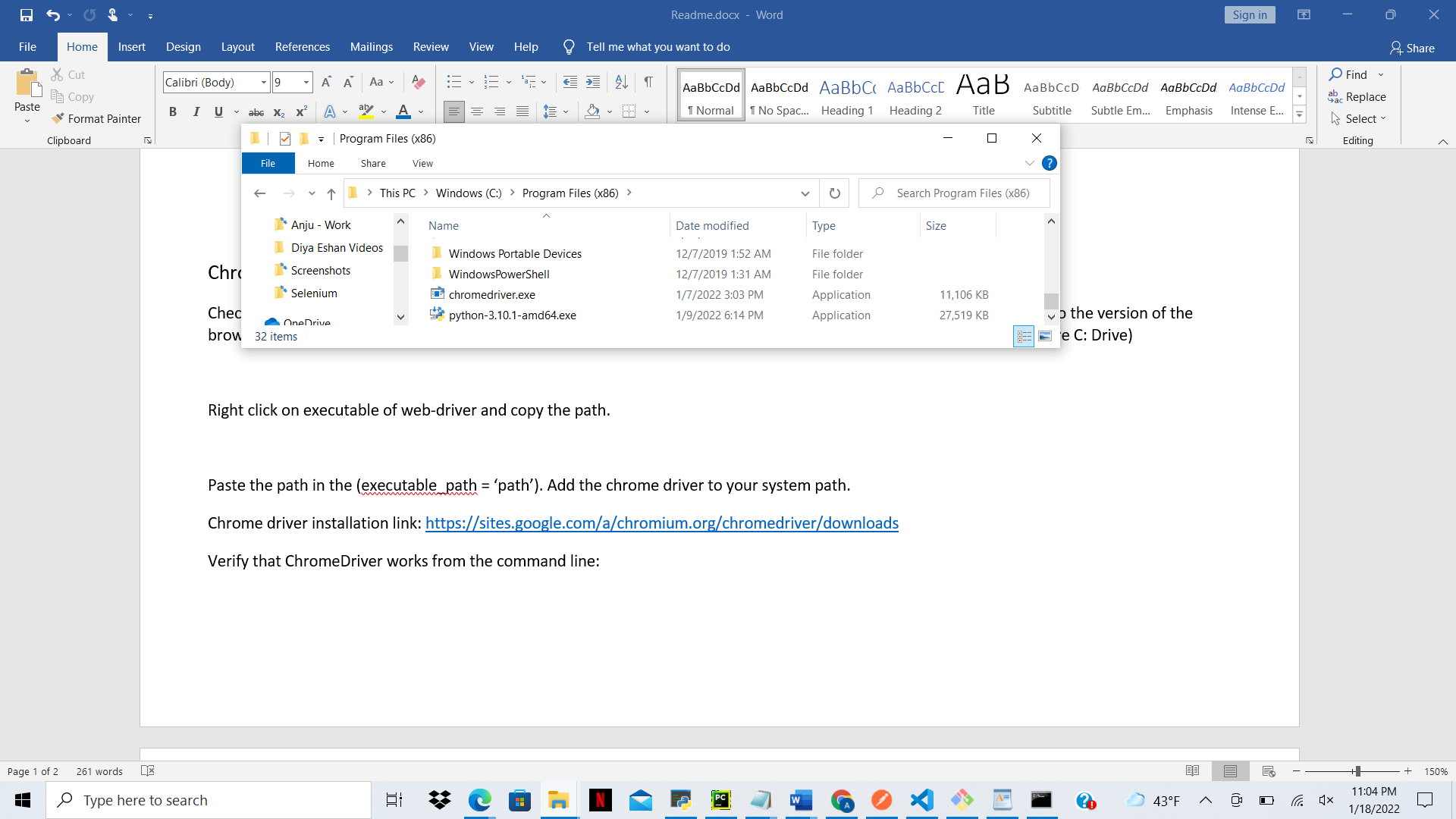
PyCharm IDE download link: <https://www.jetbrains.com/pycharm/download/#section=windows>



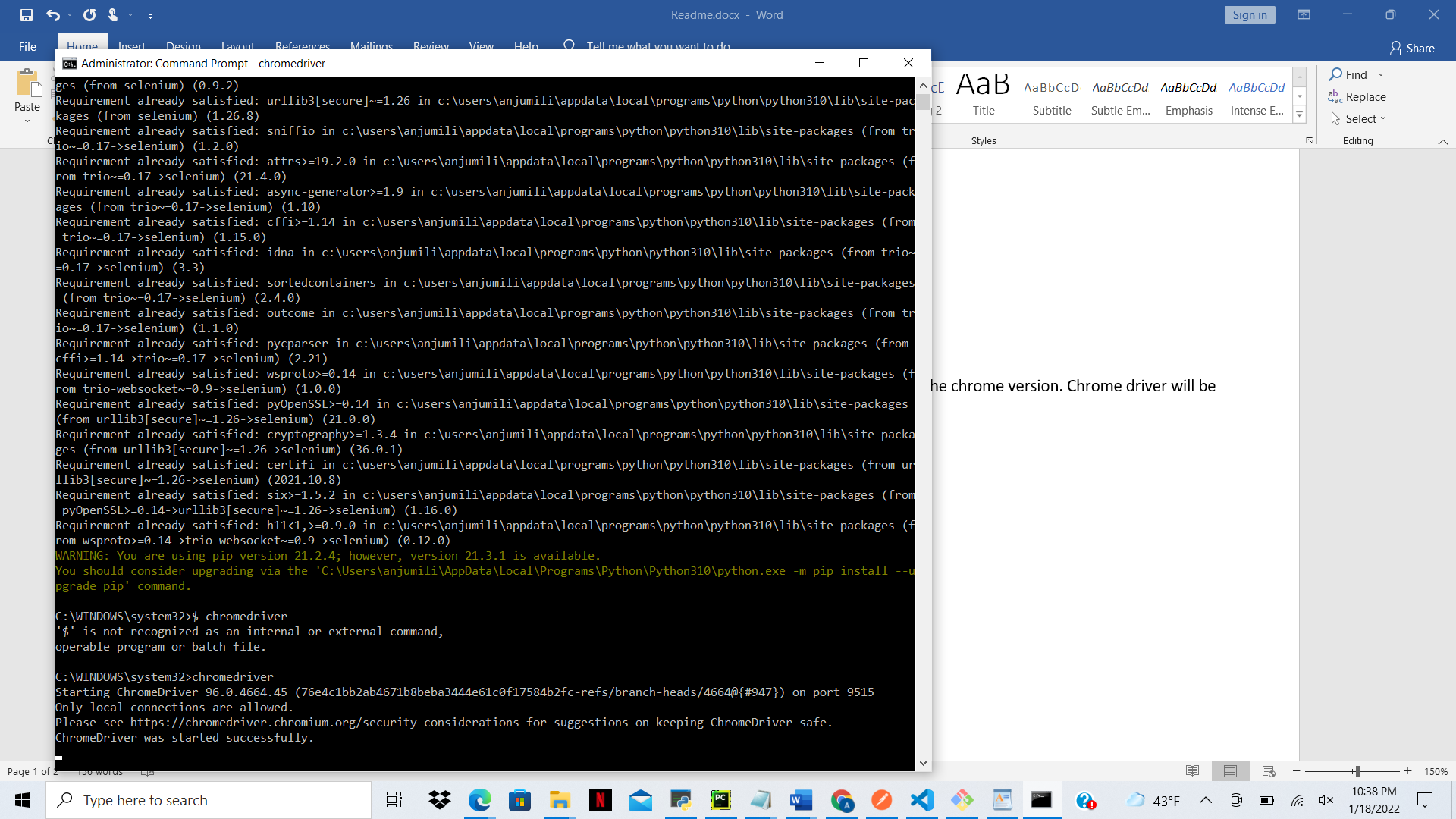
Chrome Driver Installation:

Chrome driver installation link: <https://sites.google.com/a/chromium.org/chromedriver/downloads>

Check the version of Chrome installed on your machine and download the web-driver zip file from the below link suitable to the version of the browser. Chrome driver will be installed in the form an executable for windows. Extract the zip file to desired location. Right click on executable of web-driver and copy the path. Add the chrome driver to your system path.



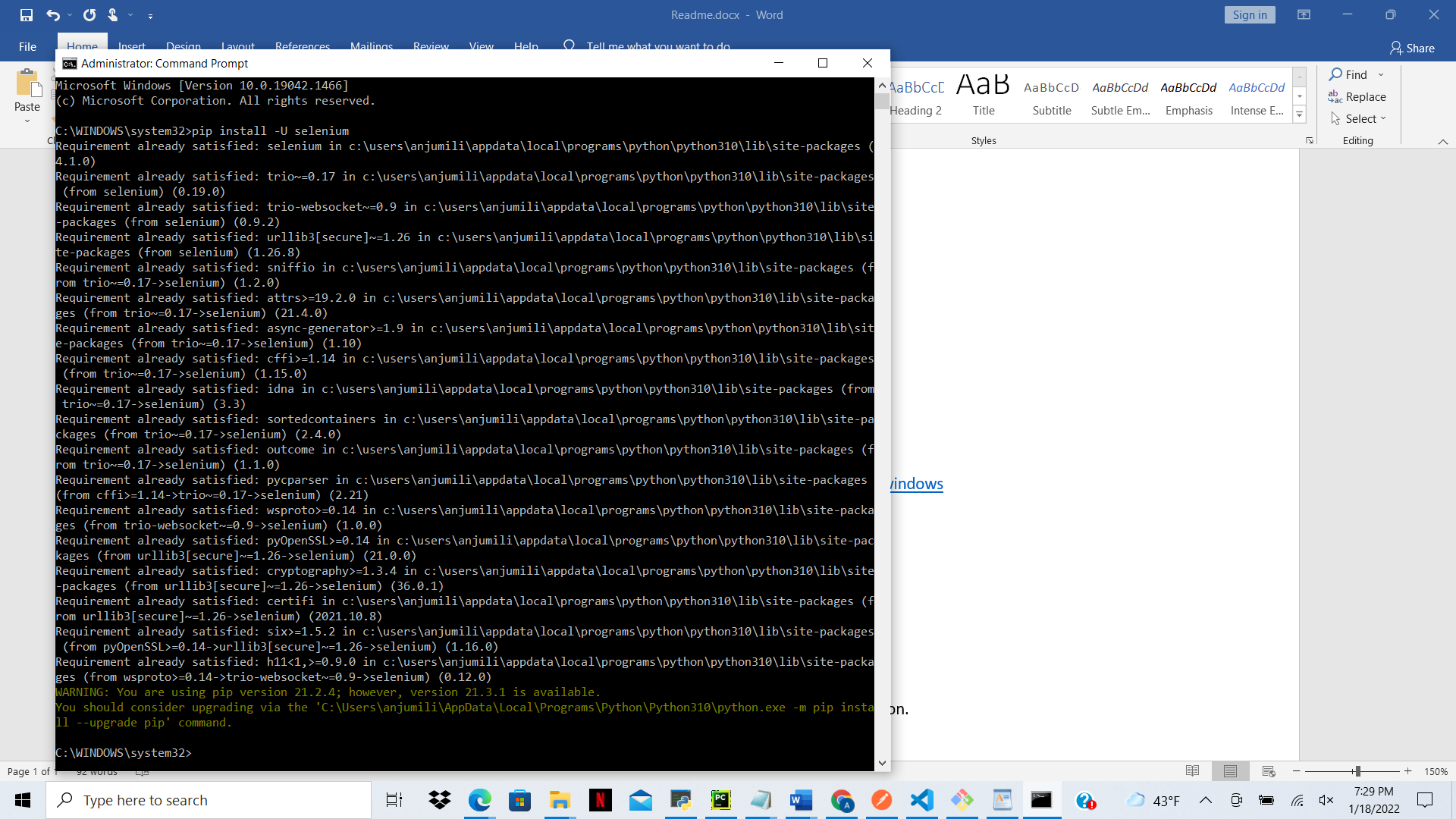
Verify that ChromeDriver works from the command line:



Selenium Installation in Python:

Open a Terminal/cmd and write the below command to install selenium libraries in python.

*pip install -U selenium*



**Test case to CREATE AN ACCOUNT:**

1. From Selenium Import webdriver and other libraries.
2. Initialize the chrome driver.
3. Maximize the window using maximize\_window() function.
4. Use get() function and pass URL as the parameter.
5. Use find\_element\_by\_id(), find\_element\_by\_xpath(), find\_element\_by\_css\_selectors() to find elements on the webpage.
6. Use ActionChains to hover over elements, click() function to click on buttons, send\_keys() function to enter text in text boxes and quit() function to close the browser.
7. Use switch\_to.frame(iFrame) and driver.switch\_to.default\_content() when working with the reCAPTCHA frame.