



# *CSE- 322*

## *Software Engineering Lab*

Lab : 03 (part-2)

Testing using Selenium

**Fahad Ahmed**

Lecturer, Dept. of CSE

E-mail: [fahadahmed@uap-bd.edu](mailto:fahadahmed@uap-bd.edu)

# Selenium



Input Forms ▾ Date pickers ▾ Table ▾

Demo Home

Progress Bars ▾

Alerts & Modals ▾

List Box ▾

Others ▾

BOOTSTRAP: EASY TO USE  
START WRITING YOUR SELENIUM SCRIPTS  
WITH MOST POPULAR FRONTEND  
FRAMEWORK

## Menu List

### ▼ All Examples

- Input Forms
- Date pickers
- Table
- Progress Bars & Sliders
- Alerts & Modals
- List Box



WELCOME TO SELENIUM EASY DEMO

We have listed most of the components that are used by developers to build

web applications

<https://www.seleniumeasy.com/test/>

21-Feb-23

Fall\_22@FMD

## Two Input Fields

First Let us try with Two input fields and a Button

- Enter Value for a
- Enter Value for b
- Click on 'Get Total' button to display the sum of two numbers 'a and b'

Enter a

Enter b

Total a + b =

**Selenium** is a powerful tool for controlling web browsers through programs and performing browser automation.

It is functional for all browsers, works on all major OS and its scripts are written in various languages i.e Python, Java, C#, etc, we will be working with Python.

## We want to cover by using selenium

- Navigating web URL
- Click
- Type(Text)
- Read and Response
- Some basic functions ... ..

## Limitations of Selenium

**No support for desktop applications** – Selenium does not support testing for desktop applications.

**Open Source Forums** – Since Selenium is open source software, one has to rely on community forums to get your technical issues resolved.

**No support for REST and SOAP Platforms** – We can't perform automation tests on web services like SOAP or REST using Selenium.

**No Reporting capability** – Selenium does not have any inbuilt reporting capability, one has to rely on plug-ins like JUnit and TestNG for test reports.

**Image Testing** – It is not possible to perform testing on images. One needs to integrate Selenium with Sikuli for image testing.

# Check Selenium

```
C:\Users\Fahad Ahmed>pip list
```

Package	Version
---------	---------

-----

pip	20.2.3
-----	--------

setuptools	49.2.1
------------	--------

WARNING: You are using pip version 20.2.3; however, version 21.0 is available.

You should consider upgrading via the 'c:\users\fahad ahmed\appdata\local\programs\python\python38\python.exe -m pip install --upgrade pip' command.

```
C:\Users\Fahad Ahmed>
```

# Install Selenium

Use pip to install the selenium package. Python 3.6 has pip available in the standard library. Using pip, you can install selenium like this:

**pip install -U selenium**

```
C:\Users\Fahad Ahmed>pip install -U selenium
Collecting selenium
  Downloading selenium-3.141.0-py2.py3-none-any.whl (904 kB)
    |████████████████████████████████████████| 904 kB 80 kB/s
Collecting urllib3
  Downloading urllib3-1.26.3-py2.py3-none-any.whl (137 kB)
    |████████████████████████████████████████| 137 kB 142 kB/s
Installing collected packages: urllib3, selenium
Successfully installed selenium-3.141.0 urllib3-1.26.3
WARNING: You are using pip version 20.2.3; however, version 21.0 is available.
You should consider upgrading via the 'c:\users\fahad ahmed\appdata\local\programs\python\python38\python.exe
-m pip install --upgrade pip' command.
```

# Check Selenium

```
C:\Users\Fahad Ahmed>pip list
```

Package	Version
---------	---------

-----	-----
-------	-------

pip	20.2.3
-----	--------

selenium	3.141.0
----------	---------

setuptools	49.2.1
------------	--------

urllib3	1.26.3
---------	--------

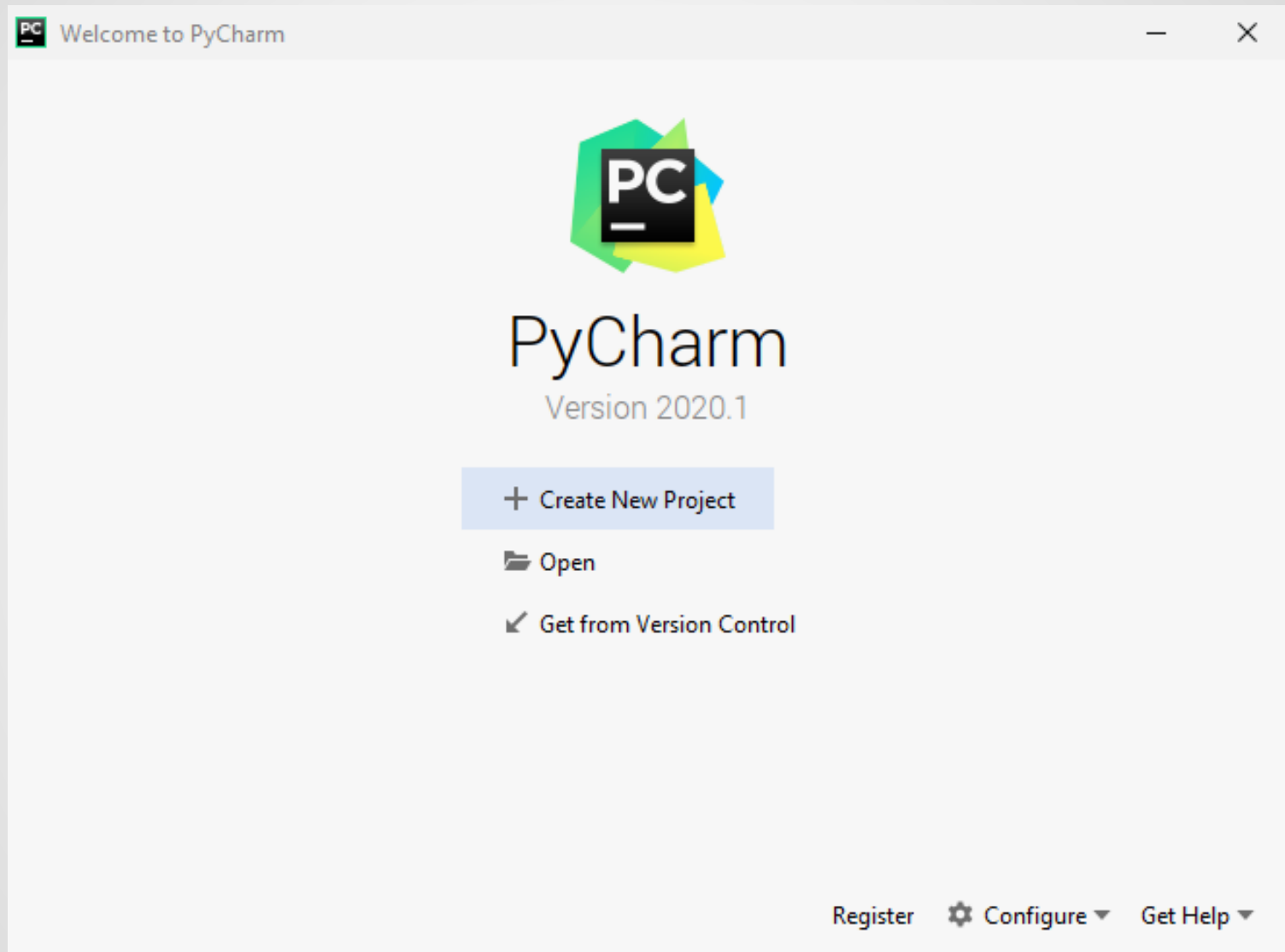
WARNING: You are using pip version 20.2.3;



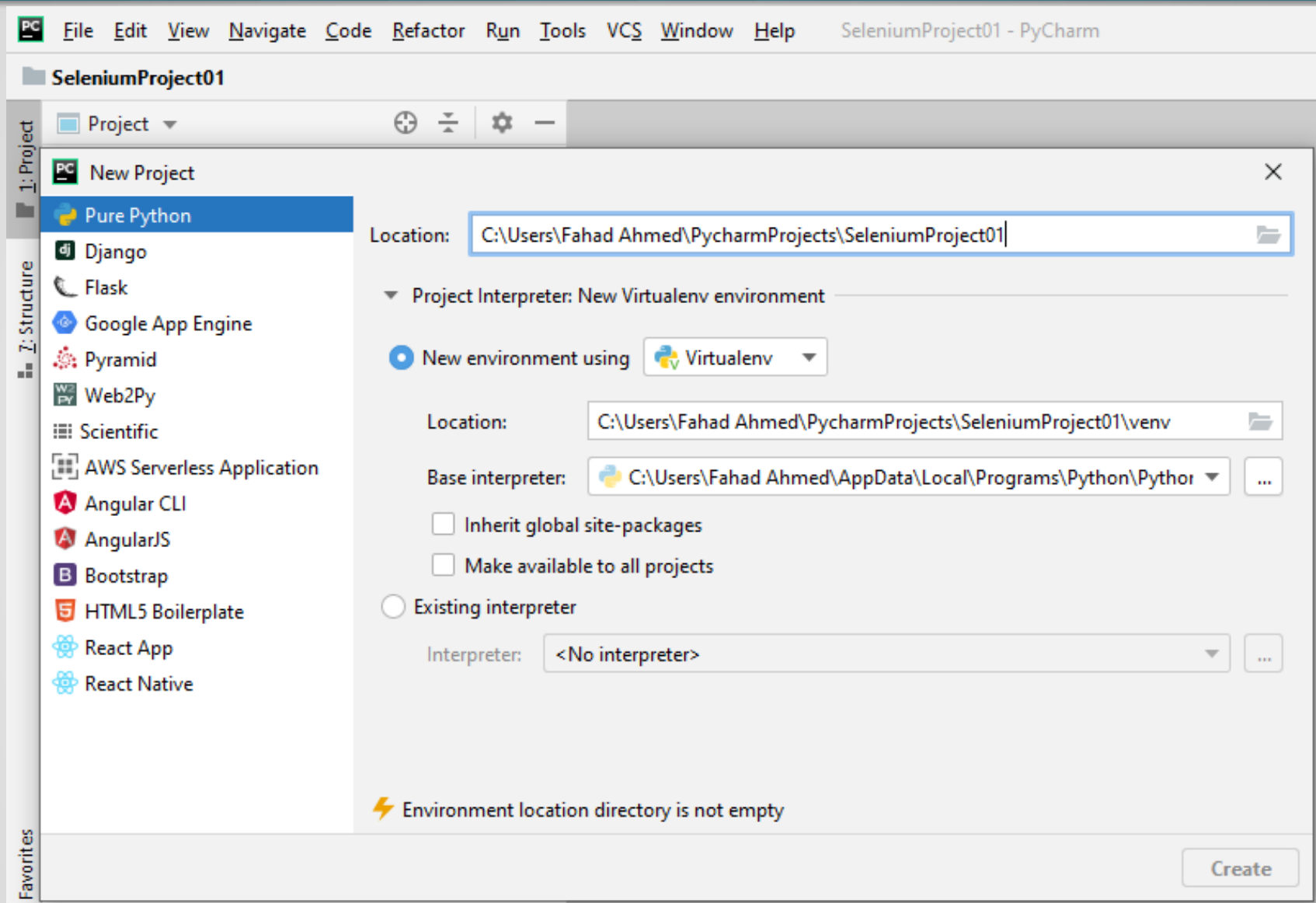
JetBrains PyCharm 2020.1 build 201.6668.115



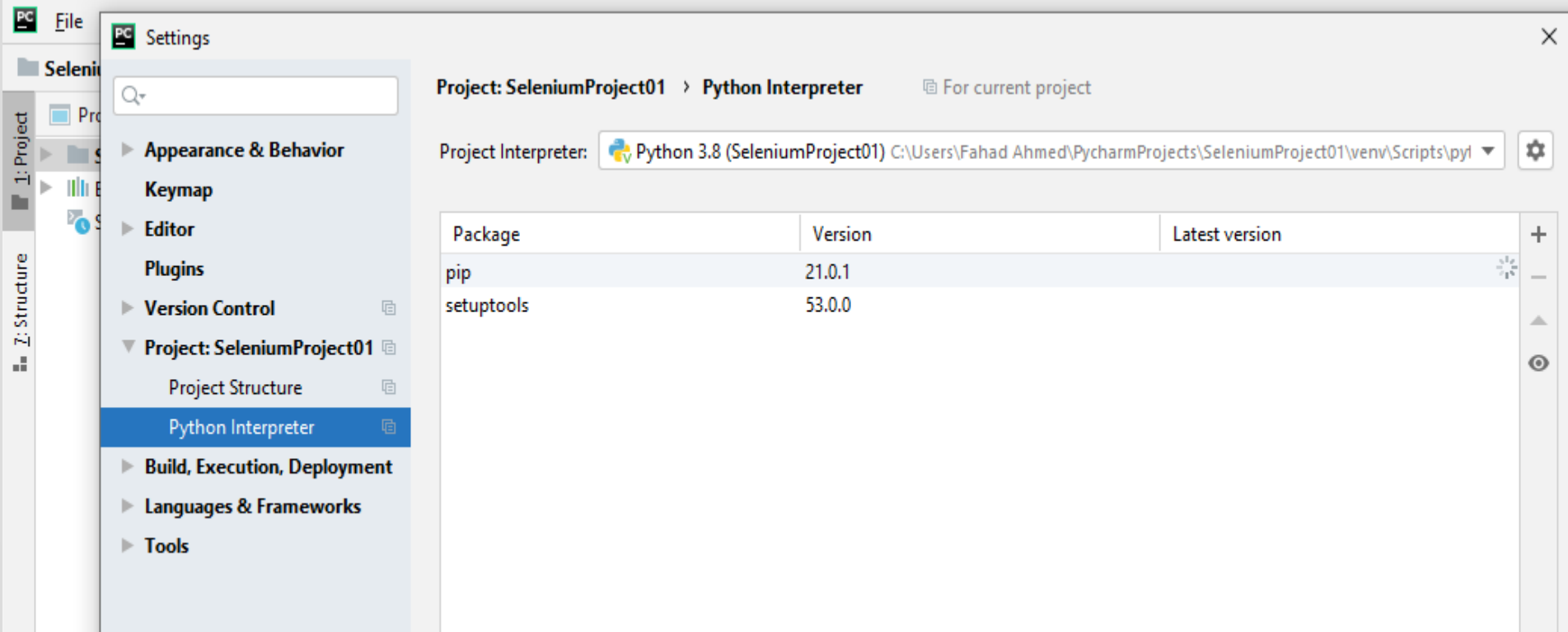
# PyCharm: Create new project



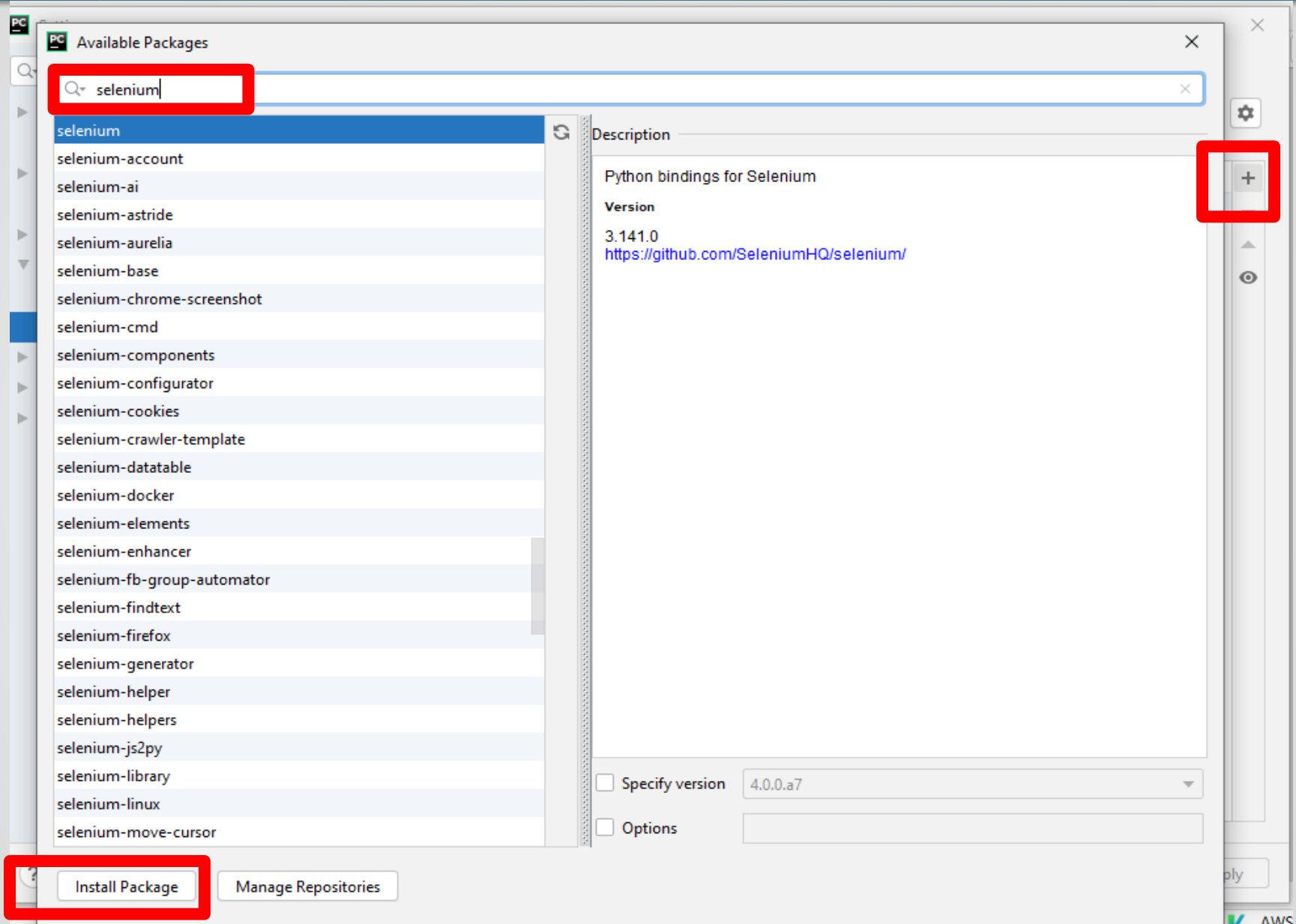
# PyCharm: Create new project



# PyCharm: Selenium configuration in your project



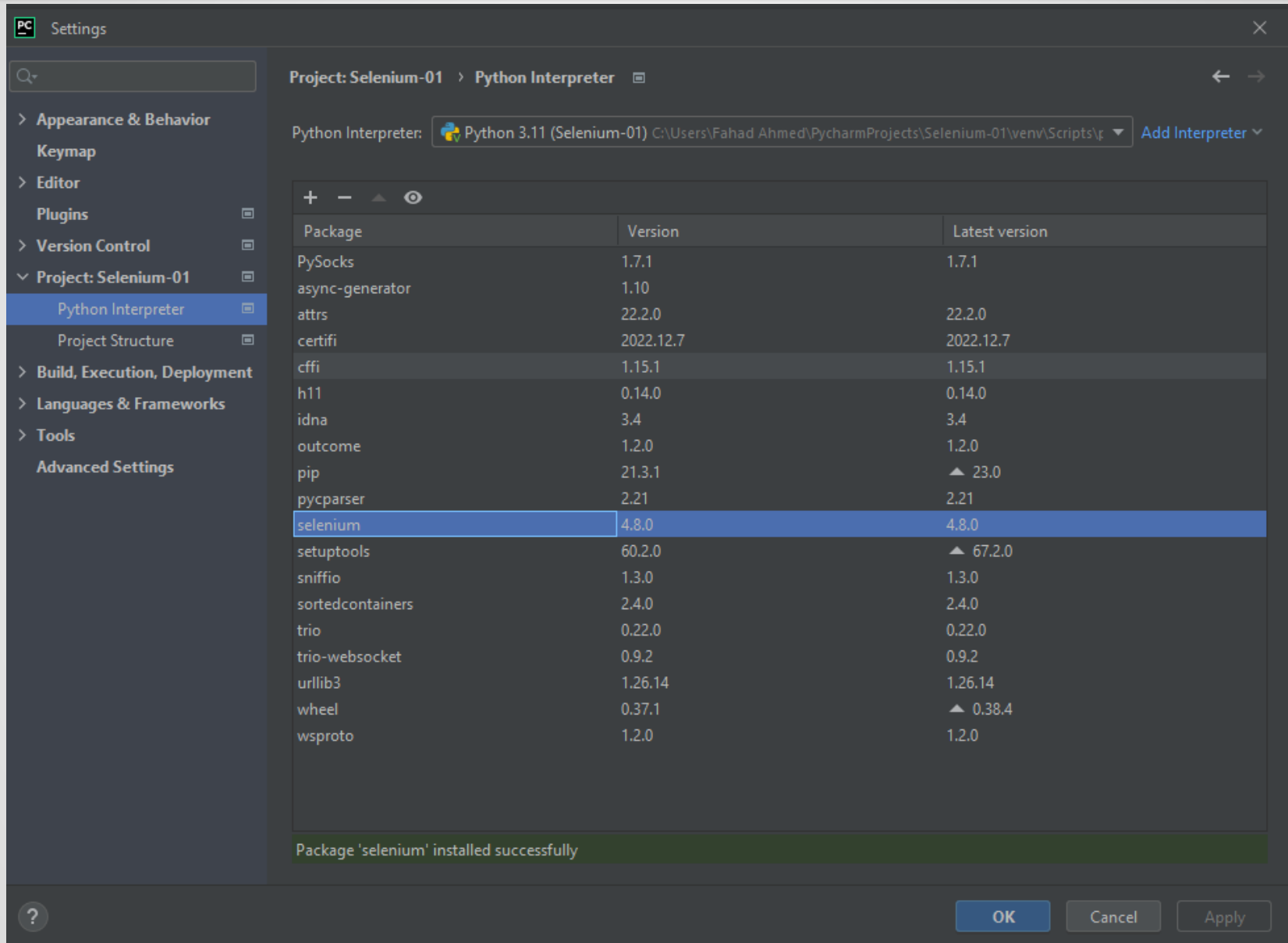
# PyCharm: Selenium configuration in your project



# PyCharm: Selenium configuration in your project

Package	Version	Latest version	+
pip	21.0.1	21.0.1	-
selenium	3.141.0	3.141.0	▲
setuptools	53.0.0	53.0.0	ⓘ
urllib3	1.26.3	1.26.3	

# PyCharm: Selenium configuration in your project



## Agenda:

### Run testing on Browser (Like Chrome, Firefox)

#### Pre-Requisite :

1. Download drivers
2. Some packages to import
  - Web-driver
  - Common-keys



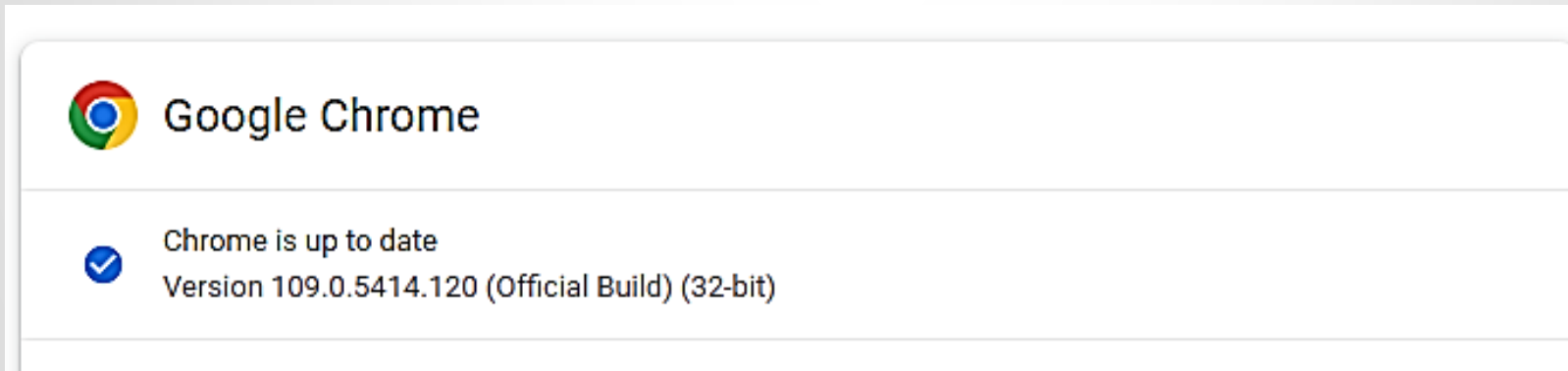
# Selenium: Drivers download

Drivers:

<https://www.selenium.dev/downloads/>

Selenium requires a driver to interface with the chosen browser. Firefox, for example, requires **geckodriver**, which needs to be installed before the below examples can be run.

Check version:      `chrome://settings/help`



# Selenium: Drivers download

Drivers:

<https://www.selenium.dev/downloads/>

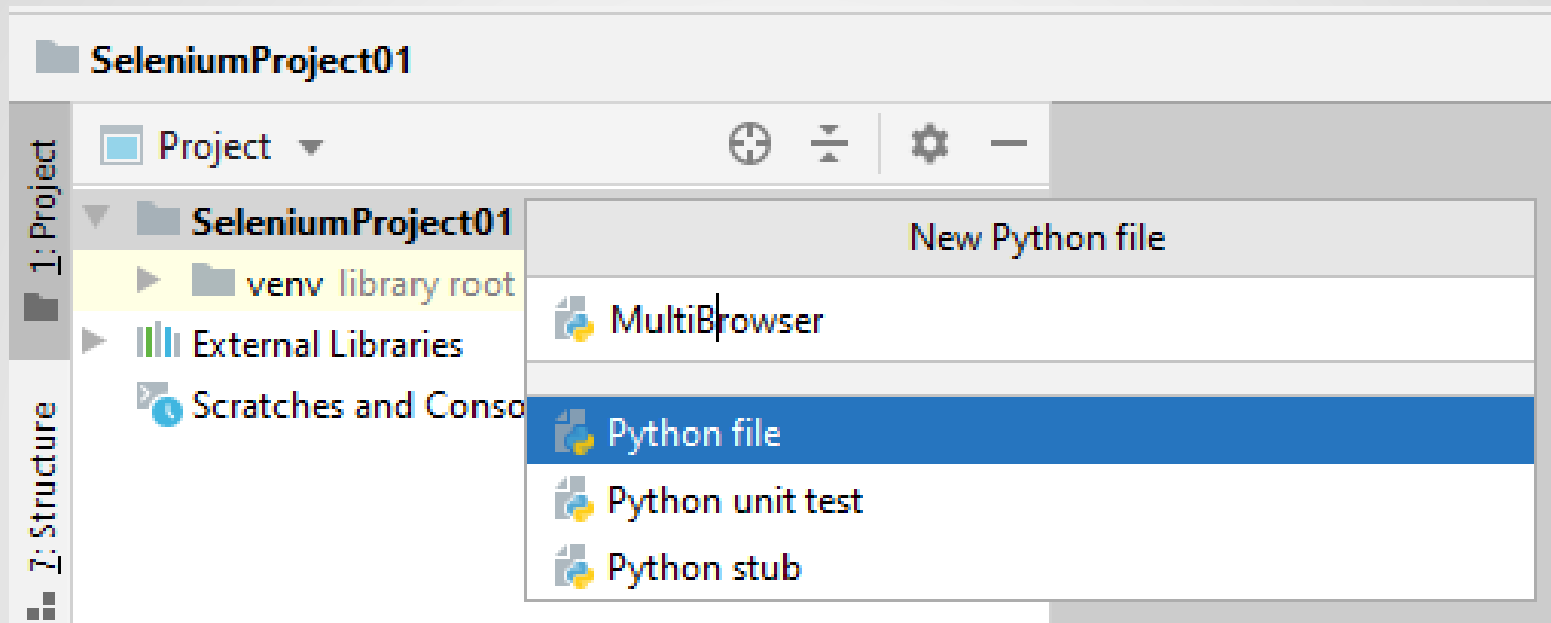
Selenium requires a driver to interface with the chosen browser. Firefox, for example, requires **geckodriver**, which needs to be installed before the below examples can be run.

<b>Chrome:</b>	<a href="https://sites.google.com/a/chromium.org/chromedriver/downloads">https://sites.google.com/a/chromium.org/chromedriver/downloads</a>
<b>Firefox:</b>	<a href="https://github.com/mozilla/geckodriver/releases/tag/v0.32.1">https://github.com/mozilla/geckodriver/releases/tag/v0.32.1</a>

<https://chromedriver.chromium.org/>

<https://github.com/mozilla/geckodriver/releases>

# Create new python file under project




**Selenium WebDriver** refers to both the language bindings and the implementations of the individual browser controlling code. This is commonly referred to as just WebDriver. Selenium WebDriver is a W3C Recommendation. WebDriver is designed as a simple and more concise programming interface.

# Selenium: web driver

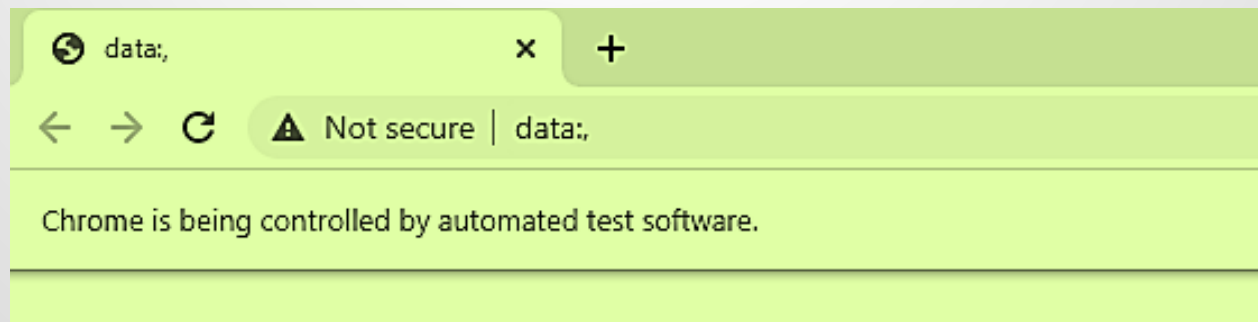
```
from selenium import webdriver
```

```
driver = webdriver.Chrome(  
    executable_path="C:\\Users\\Fahad  
Ahmed\\PycharmProjects\\SeleniumProject01\\drivers\\chromedriver.exe")
```

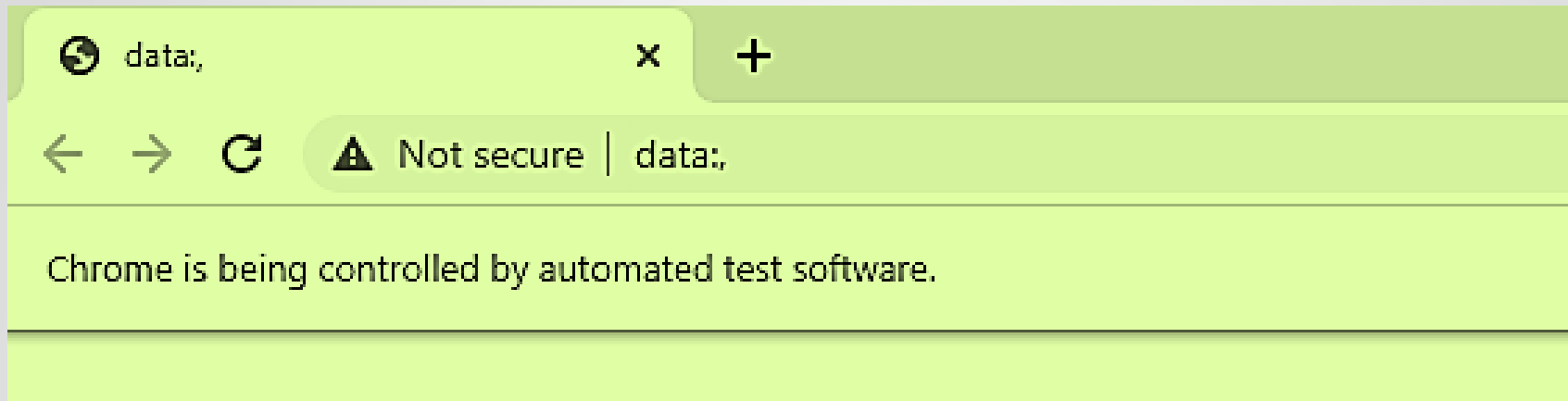


A screenshot of a code editor window titled 'MultiBrowser.py'. The code is as follows:

```
1 from selenium import webdriver  
2  
3 #the instance of browser WebDriver is created  
4 #chrome  
5 driver = webdriver.Chrome(  
6     executable_path="C:\\Users\\Fahad Ahmed\\PycharmProjects\\SeleniumProject01\\drivers\\chromedriver.exe")  
7
```



# Selenium: web driver



## Run Window:

"C:\Users\Fahad Ahmed\PycharmProjects\SeleniumProject01\venv\Scripts\python.exe"

"C:/Users/Fahad Ahmed/PycharmProjects/SeleniumProject01/MultiBrowser.py"

Process finished with exit code 0

# Selenium: web driver

```
from selenium import webdriver
```

```
#firefox
```

```
driver = webdriver.Firefox(  
    executable_path="C:\\Users\\Fahad  
Ahmed\\PycharmProjects\\SeleniumProject01\\drivers\\geckodriver.exe")
```

```
from selenium import webdriver
```

```
#the instance of browser WebDriver is created
```

```
#chrome
```

```
#driver = webdriver.Chrome(executable_path="C:\\Users\\Fahad Ahmed\\PycharmProjects\\SeleniumProject01\\drivers\\chromedriver.exe")
```

```
#firefox
```

```
driver = webdriver.Firefox(  
    executable_path="C:\\Users\\Fahad Ahmed\\PycharmProjects\\SeleniumProject01\\drivers\\geckodriver.exe")
```

# WebDriver Commands

```
from selenium import webdriver
#the instance of browser WebDriver is created
#chrome
driver = webdriver.Chrome(
    executable_path="C:\\Users\\Fahad
Ahmed\\PycharmProjects\\SeleniumProject01\\drivers\\chr
omedriver.exe")
driver.close() # if we want to auto close browser after run
```

Report with example based on own project:  
driver.close() vs driver.quit()

# WebDriver Commands

```
from selenium import webdriver
import time
#the instance of browser WebDriver is created
#chrome
driver = webdriver.Chrome(
    executable_path="C:\\Users\\Fahad
Ahmed\\PycharmProjects\\SeleniumProject01\\drivers\\chromedriver.exe")
time.sleep(5) # if we want to wait some time then close the browser
driver.close()# if we want to auto close browser after run
```

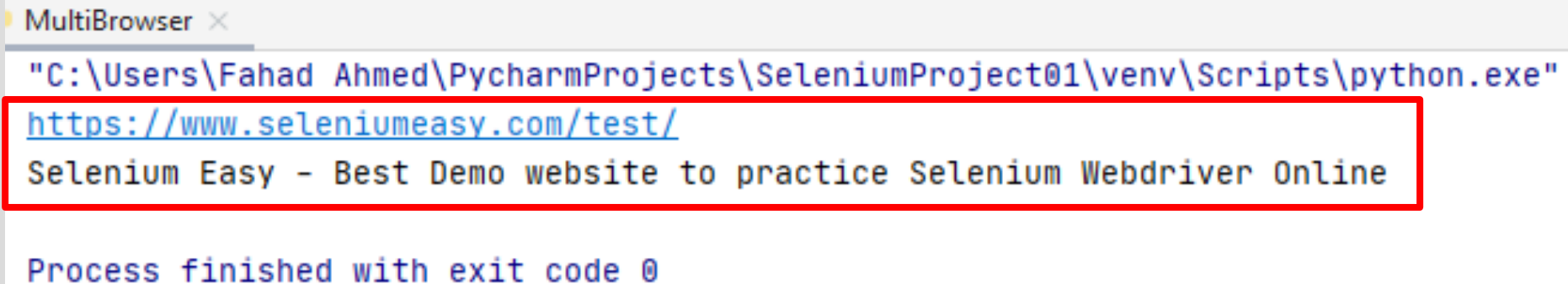


# Navigating web URL

```
from selenium import webdriver
import time
#the instance of browser WebDriver is created
#chrome
driver = webdriver.Chrome(
    executable_path="C:\\Users\\Fahad
Ahmed\\PycharmProjects\\SeleniumProject01\\drivers\\chromedriver.exe")
driver.get("https://www.seleniumeasy.com/test/")
time.sleep(5)
driver.close()# if we want to auto close browser after run
```

# Retrieve data from web URL

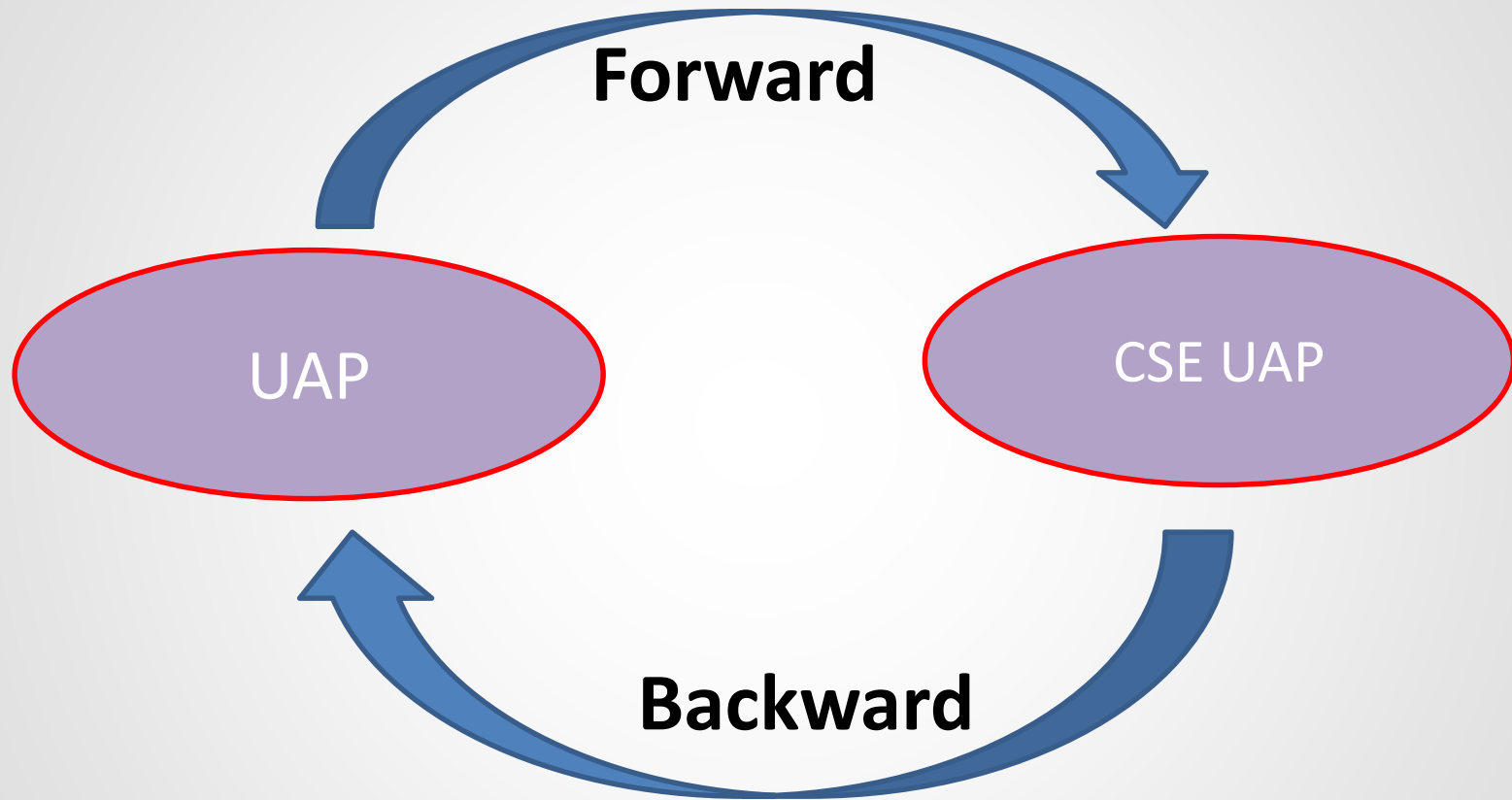
```
from selenium import webdriver
import time
#the instance of browser WebDriver is created
#chrome
driver = webdriver.Chrome(
    executable_path="C:\\Users\\Fahad
Ahmed\\PycharmProjects\\SeleniumProject01\\drivers\\chromedriver.exe")
driver.get("https://www.seleniumeasy.com/test/")
print(driver.current_url)# print url
print(driver.title)# print title
time.sleep(5)
driver.close()# if we want to auto close browser after run
```



```
MultiBrowser x
"C:\Users\Fahad Ahmed\PycharmProjects\SeleniumProject01\venv\Scripts\python.exe"
https://www.seleniumeasy.com/test/
Selenium Easy - Best Demo website to practice Selenium Webdriver Online

Process finished with exit code 0
```

# WebDriver Navigational Commands



# WebDriver Navigational Commands

```
from selenium import webdriver
import time

driver = webdriver.Chrome(
    executable_path="C:\\Users\\Fahad
Ahmed\\PycharmProjects\\SeleniumProject01\\drivers\\chromedriver.exe")

driver.get("https://www.google.com/")
print(driver.current_url)
time.sleep(2)

driver.get("https://www.youtube.com/")
print(driver.current_url)
time.sleep(2)

driver.back() # back to google
print("\nBack to : ")
print(driver.title)
time.sleep(2)

driver.forward() # forward to UAP
print("\nforward to :")
print(driver.title)
time.sleep(2)

driver.close()
```

# WebDriver Navigational Commands

```
from selenium import webdriver
import time
driver = webdriver.Chrome(executable_path="C:\\Users\\Fahad
Ahmed\\PycharmProjects\\SeleniumProject01\\drivers\\chromedriver.exe")

driver.get("https://www.python.org/")
print(driver.current_url)
time.sleep(2)

driver.get("https://www.uap-bd.edu/")
print(driver.current_url)
time.sleep(2)
driver.back() # back to google
print("\nBack to : ")
print(driver.title)
time.sleep(2)
driver.forward() # forward to youtube
print("\nforward to :")
print(driver.title)
time.sleep(2)
driver.close()
```

# WebDriver Navigational Commands

```
"C:\Users\Fahad Ahmed\PycharmProjects\SeleniumProject01\venv\Scripts\python.exe"
```

```
"C:/Users/Fahad Ahmed/PycharmProjects/SeleniumProject01/MultiBrowser.py"
```

```
https://www.google.com/
```

```
Google
```

```
https://www.youtube.com/
```

```
YouTube
```

```
Back to :
```

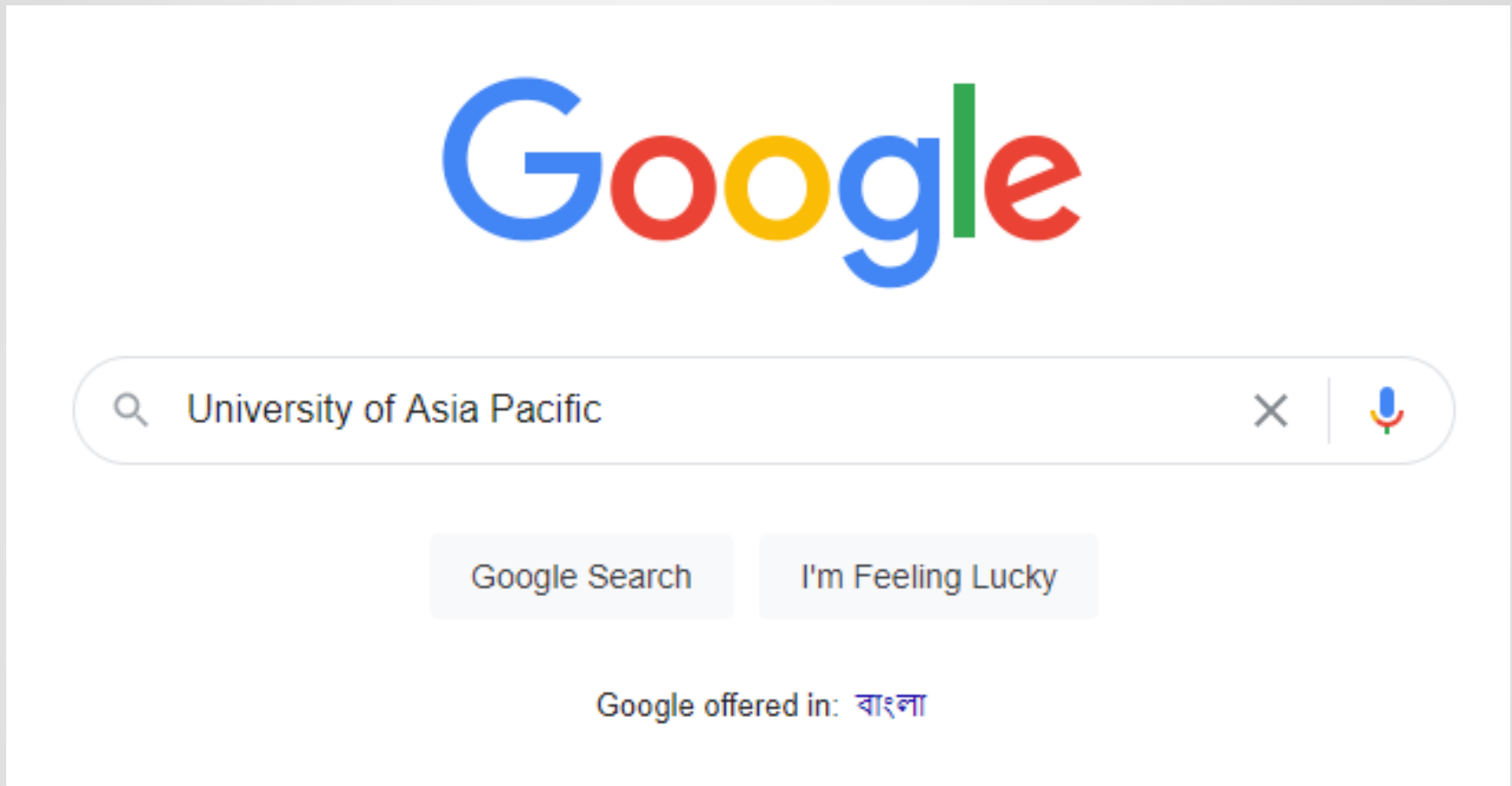
```
Google
```

```
forward to :
```

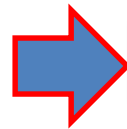
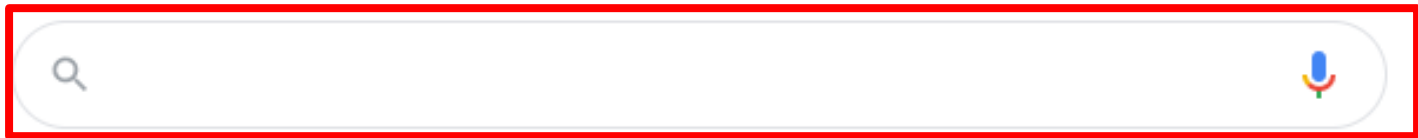
```
YouTube
```

```
Process finished with exit code 0
```

# WebDriver Navigational Commands



# WebDriver Navigational Commands



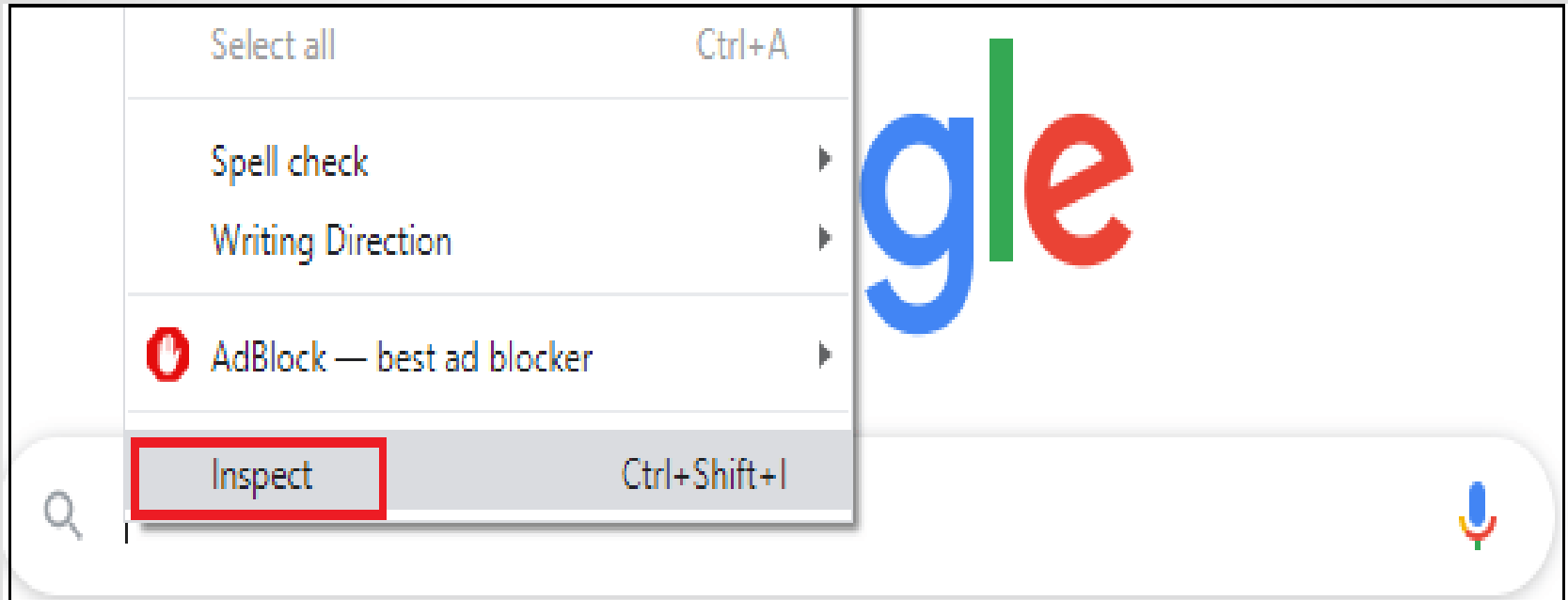
Google Search

I'm Feeling Lucky

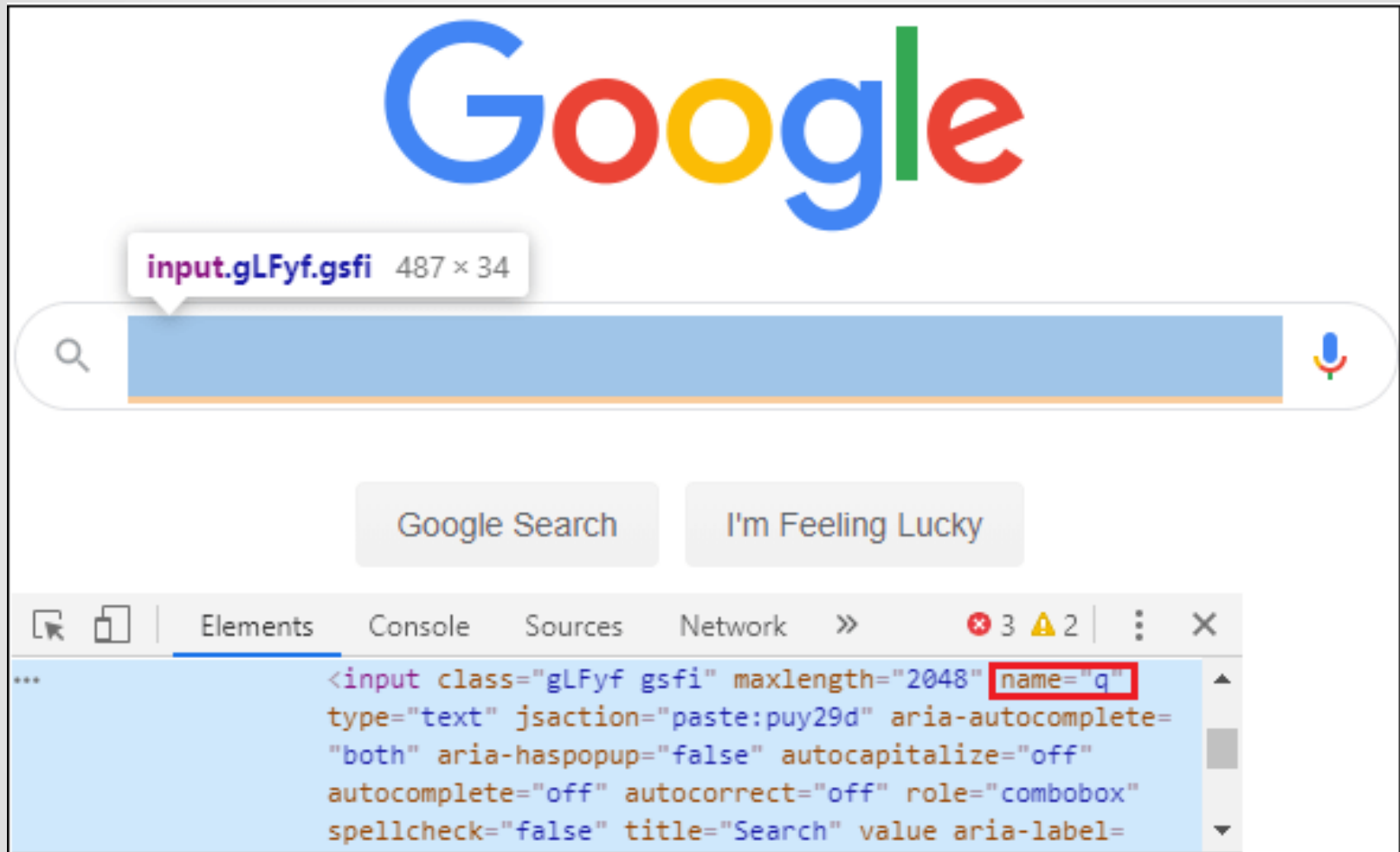
Google offered in: বাংলা



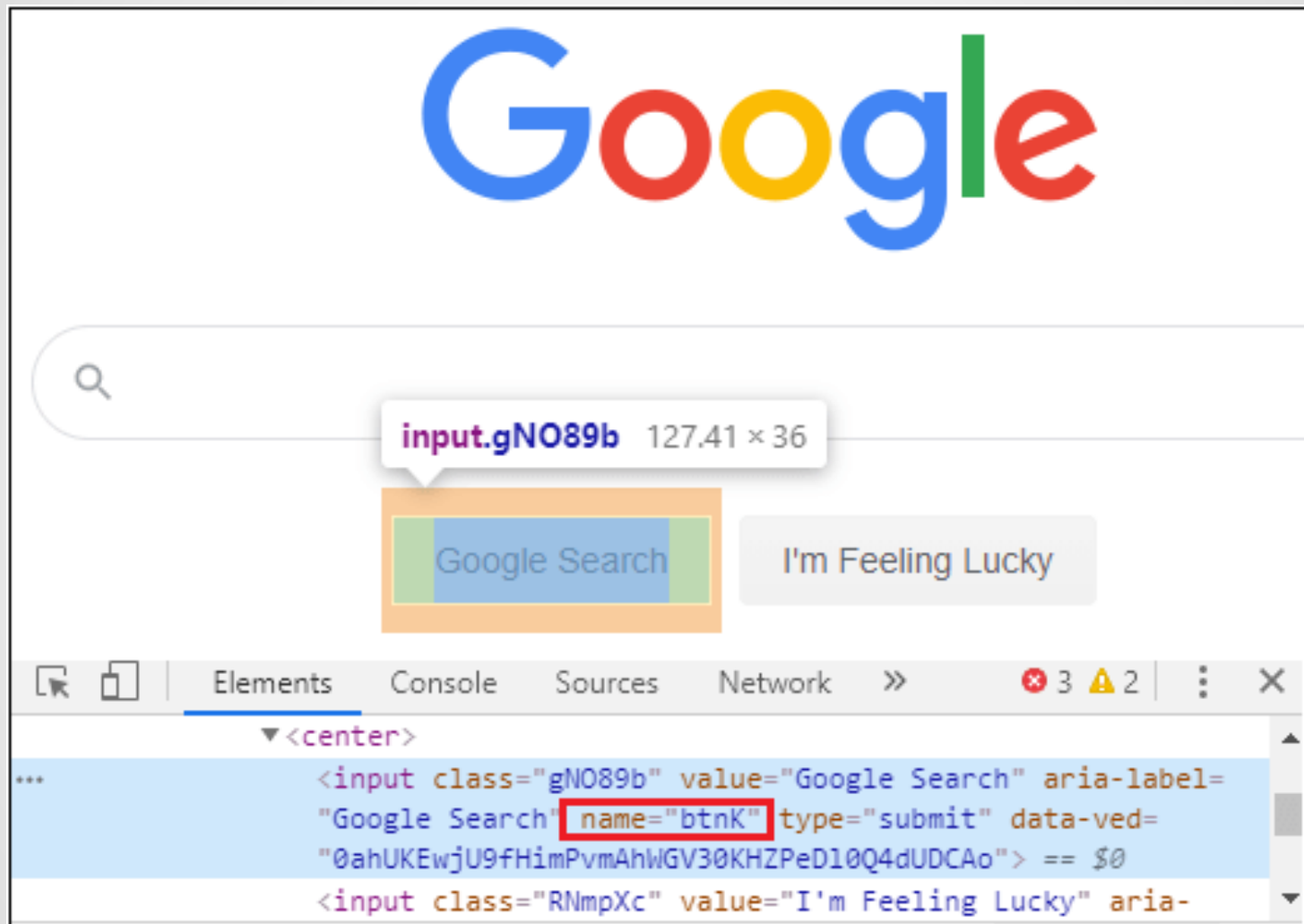
# WebDriver Navigational Commands



# WebDriver Navigational Commands



# WebDriver Navigational Commands



# WebDriver Navigational Commands

Some times may occur problem

```
from selenium import webdriver
from selenium.webdriver.common.keys import Keys
import time

driver = webdriver.Chrome(
    executable_path="C:\\Users\\Fahad
Ahmed\\PycharmProjects\\SeleniumProject01\\drivers\\chromedriver.exe")

driver.maximize_window()
driver.get("https://www.google.com/")
print(driver.current_url)

driver.find_element_by_name("q").send_keys("University of Asia Pacific")
time.sleep(3)
#click on the Google search button
driver.find_element_by_name("btnK").send_keys(Keys.ENTER)
time.sleep(3)
#close the browser
driver.close()
print("sample test case successfully completed")
```

# WebDriver Navigational Commands

Python

PSF

Docs

PyPI

Jobs

Community



Donate



Search

GO

Socialize

About

Downloads

Documentation

Community

Success Stories

News

Events

# Python 3: Fibonacci series up to n

```
>>> def fib(n):  
>>>     a, b = 0, 1  
>>>     while a < n:  
>>>         print(a, end=' ')  
>>>         a, b = b, a+b  
>>>     print()  
>>> fib(1000)  
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377 610 987
```

>\_

## Functions Defined

The core of extensible programming is defining functions. Python allows mandatory and optional arguments, keyword arguments, and even arbitrary argument lists. [More about defining functions in Python 3](#)

1

2

3

4

5

Python is a programming language that lets you work quickly  
and integrate systems more effectively. >>> [Learn More](#)

# WebDriver Navigational Commands

```
from selenium import webdriver
from selenium.webdriver.common.keys import Keys
import time
##driver = webdriver.Firefox(executable_path="C:\\Users\\Fahad
Ahmed\\PycharmProjects\\SeleniumProject01\\drivers\\geckodriver.exe")
driver = webdriver.Chrome(executable_path="C:\\Users\\Fahad
Ahmed\\PycharmProjects\\SeleniumProject01\\drivers\\chromedriver.exe")

driver.get("http://www.python.org")
assert "Python" in driver.title
elem = driver.find_element_by_name("q")
elem.clear()

elem.send_keys("pycon")
elem.send_keys(Keys.RETURN)
time.sleep(5)
assert "No results found." not in driver.page_source
driver.close()
print("sample test case successfully completed")
```

# WebDriver Navigational Commands

```
from selenium import webdriver
from selenium.webdriver.common.keys import Keys
import time
driver=webdriver.Chrome(executable_path="C:\\Users\\Fahad Ahmed\\PycharmProjects\\web-drivers\\chromedriver.exe")

print("Enter to google\n")
driver.get("https://www.youtube.com/")
print(driver.title)
time.sleep(2)

driver.find_element_by_name("search_query").send_keys("university of asia pacific notable alumni")
time.sleep(1)
driver.find_element_by_id("search-icon-legacy").send_keys(Keys.ENTER)
time.sleep(4)
driver.close()
```



Thanks to All