DATE 10012023 DAY TUESDAY

from selenium import webdriver

from selenium.webdriver.chrome.service import Service

from selenium.webdriver.common.by import By

import time

service\_obj = Service("C:\\ChromeDriver\\chromedriver.exe")

driver = webdriver.Chrome(service=service\_obj)

# print(dir(webdriver))

#url lunch

driver.get('https://www.facebook.com/r.php?locale=en\_GB&display=page')

#get page title

print(driver.title)

# driver.quit() # all window close using quite

# only current single window

#locaters

#id,name,cssname,xpath,link\_text,class\_name,tag\_name,css selector

driver.find\_element(By.NAME,'firstname').send\_keys('Javed')

driver.find\_element(By.NAME,'lastname').send\_keys('Ali')

driver.find\_element(By.NAME,'reg\_email\_\_').send\_keys('k.csnayak108@gmail.com')

driver.find\_element(By.NAME,'reg\_email\_confirmation\_\_').send\_keys('k.csnayak108@gmail.com')

driver.find\_element(By.ID,'password\_step\_input').send\_keys('Gbest@108')

driver.find\_element(By.ID,'day').send\_keys(17)

driver.find\_element(By.ID,'month').send\_keys('Jan')

driver.find\_element(By.ID,'year').send\_keys(1997)

driver.find\_element(By.CLASS\_NAME,'\_8esa').click()

driver.find\_element(By.NAME,'websubmit').click()

time.sleep(100)

driver.close()

DATE 11012023 DAY WEDNESDAY

#ATM Oops

#I want a atm machine,where account balance is 15000,account holder name is mr.Ram babu,

# account pin is 8000, account bank name sbi,

# "welcome to our bank Mr.Ram babu"

# before using this you have to acess with your account pin.

# Q.if you press 0 then you are loging out

# Q.if you press 1 then it will show your account balance

# Q.if you press 2 then withdraw

# Q.if you press 3 add cash.

# Q.Credited 5000 show the account balance?

# Q.Debited 200 show the account balance?

DATE 16012023 DAY MONDAY

from selenium import webdriver

from selenium.webdriver.chrome.service import Service

from selenium.webdriver.common.by import By

import time

service\_obj = Service("C:\\ChromeDriver\\chromedriver.exe")

driver = webdriver.Chrome(service=service\_obj)

driver.get('https://rahulshettyacademy.com/angularpractice/')

driver.find\_element(By.NAME,'email').send\_keys('k.csnyak108@gmail.com')

driver.find\_element(By.ID,'exampleInputPassword1').send\_keys('12345667')

driver.find\_element(By.ID,'exampleCheck1').click()

#CSS\_SELECTOR(tagname[attribute='Value']) ,.classname,#id

driver.find\_element(By.CSS\_SELECTOR,"input[name='name']").send\_keys('Krishna')

# driver.find\_element(By.CSS\_SELECTOR,".form-control").send\_keys('Krishna')

# driver.find\_element(By.CSS\_SELECTOR,"#exampleInputPassword1").send\_keys(7867634)

#Xapth :- "//tagname[@attribute='Value']"

driver.find\_element(By.XPATH,"//input[@type='submit']").click()

message = driver.find\_element(By.CLASS\_NAME,'alert-success').text

# print(message)

assert 'Success!' in message #validation (assertion checking)

#Two Way Data Binding

driver.find\_element(By.XPATH,"(//input[@name='name'])[2]").send\_keys('Automation')

time.sleep(5)

driver.find\_element(By.XPATH,"(//input[@name='name'])[2]").clear()

time.sleep(10)

driver.close ()

DATE 19012023 DAY THURSDAY

Hi All,

I am stuck in traffic. I will try to connect 8:30pm.

If it's not possible then today have only practices session.

Link:- rahulshettyacademy.com/#/practice-project

Please check above link, and there is 3 automation link. Try to automate the webpage with the help of Locater.

Send me the python file in group. I will check and response back and will discuss same code tomorrow class and start Selenium Select () with Waits.

Static/Select Drop Down

#Select/Static Dropdown

dropdown = Select(driver.find\_element(By.ID,"exampleFormControlSelect1"))

dropdown.select\_by\_visible\_text("Male")

time.sleep(5)

dropdown.select\_by\_index(1)

DATE 23012023 DAY MONDAY

SCRIPT

Dynamic Checkbox and Radio Button

from selenium import webdriver

from selenium.webdriver.chrome.service import Service

from selenium.webdriver.common.by import By

from selenium.webdriver.support.select import Select

import time

service\_obj = Service("C:\\ChromeDriver\\chromedriver.exe")

driver = webdriver.Chrome(service=service\_obj)

driver.get('https://rahulshettyacademy.com/AutomationPractice/')

#Dynamic Checkbox

check\_box = driver.find\_elements(By.XPATH,"//input[@type='checkbox']")

print(len(check\_box))

for check in check\_box:

if check.get\_attribute("value") == "option2":

check.click()

assert check.is\_selected()

#Static Radion Button

radio\_button = driver.find\_elements(By.CSS\_SELECTOR,".radioButton")

radio\_button[2].click()

time.sleep(5)

driver.close()

Implicit Wait

An implicit wait tells WebDriver to poll the DOM for a certain amount of time when trying to find any element (or elements) not immediately available. The default setting is 0. Once set, the implicit wait is set for the life of the WebDriver

from selenium import webdriver

from selenium.webdriver.chrome.service import Service

from selenium.webdriver.common.by import By

from selenium.webdriver.support.select import Select

import time

service\_obj = Service("C:\\ChromeDriver\\chromedriver.exe")

driver = webdriver.Chrome(service=service\_obj)

driver.get('https://rahulshettyacademy.com/seleniumPractise/#/')

driver.implicitly\_wait(5)

driver.find\_element(By.CSS\_SELECTOR,".search-keyword").send\_keys("ber")

time.sleep(2)

result\_prodcuts = driver.find\_elements(By.XPATH,"//div[@class='products']/div")

count = len(result\_prodcuts)

assert count > 0

#selenium chaining

for result in result\_prodcuts:

result.find\_element(By.XPATH,"div/button").click()

driver.find\_element(By.CSS\_SELECTOR,"img[alt='Cart']").click()

driver.find\_element(By.XPATH,"//button[text()='PROCEED TO CHECKOUT']").click()

driver.find\_element(By.CSS\_SELECTOR,".promoCode").send\_keys('rahulshettyacademy')

driver.find\_element(By.CSS\_SELECTOR,".promoBtn").click()

msg = driver.find\_element(By.CLASS\_NAME,"promoInfo").text

print(msg)

time.sleep(10)

driver.close()

DATE 23012023 DAY MONDAY

Explicit Wait

An explicit wait is a code you define to wait for a certain condition to occur before proceeding further in the code. The extreme case of this is time.sleep(), which sets the condition to an exact time period to wait. There are some convenience methods provided that help you write code that will wait only as long as required. Explicit waits are achieved by using webdriverWait class in combination with expected\_conditions

from selenium import webdriver

from selenium.webdriver.chrome.service import Service

from selenium.webdriver.common.by import By

from selenium.webdriver.support import expected\_conditions

from selenium.webdriver.support.select import Select

import time

from selenium.webdriver.support.wait import WebDriverWait

service\_obj = Service("C:\\ChromeDriver\\chromedriver.exe")

driver = webdriver.Chrome(service=service\_obj)

driver.get('https://rahulshettyacademy.com/seleniumPractise/#/')

driver.implicitly\_wait(5)

driver.find\_element(By.CSS\_SELECTOR,".search-keyword").send\_keys("ber")

time.sleep(2)

result\_prodcuts = driver.find\_elements(By.XPATH,"//div[@class='products']/div")

count = len(result\_prodcuts)

assert count > 0

#Selenium Changing

for result in result\_prodcuts:

result.find\_element(By.XPATH,"div/button").click()

driver.find\_element(By.CSS\_SELECTOR,"img[alt='Cart']").click()

driver.find\_element(By.XPATH,"//button[text()='PROCEED TO CHECKOUT']").click()

driver.find\_element(By.CSS\_SELECTOR,".promoCode").send\_keys('rahulshettyacademy')

driver.find\_element(By.CSS\_SELECTOR,".promoBtn").click()

#Explict Wait

wait = WebDriverWait(driver,10)

wait.until(expected\_conditions.presence\_of\_element\_located(By.CSS\_SELECTOR,"promoInfo"))

msg = driver.find\_element(By.CLASS\_NAME,"promoInfo").text

print(msg)

DATE 24012023 DAY TUESDAY

\*Functional Automation with Selenium\*

\*Array Validation and Sum Validation\*

from selenium import webdriver

from selenium.webdriver.chrome.service import Service

from selenium.webdriver.common.by import By

from selenium.webdriver.support import expected\_conditions

from selenium.webdriver.support.select import Select

import time

from selenium.webdriver.support.wait import WebDriverWait

service\_obj = Service("C:\\ChromeDriver\\chromedriver.exe")

driver = webdriver.Chrome(service=service\_obj)

driver.get('https://rahulshettyacademy.com/seleniumPractise/#/')

driver.implicitly\_wait(5)

driver.find\_element(By.CSS\_SELECTOR,".search-keyword").send\_keys("ber")

time.sleep(2)

\*#Assertion check for product\*

excepted\_prodcut = ['Cucumber - 1 Kg','Raspberry - 1/4 Kg','Strawberry - 1/4 Kg']

final\_prodcut = []

result\_prodcuts = driver.find\_elements(By.XPATH,"//div[@class='products']/div")

for product in result\_prodcuts:

final\_prodcut.append(product.find\_element(By.XPATH,'h4').text)

assert excepted\_prodcut == final\_prodcut

count = len(result\_prodcuts)

assert count > 0

\*#selenium chaining\*

for result in result\_prodcuts:

result.find\_element(By.XPATH,"div/button").click()

driver.find\_element(By.CSS\_SELECTOR,"img[alt='Cart']").click()

driver.find\_element(By.XPATH,"//button[text()='PROCEED TO CHECKOUT']").click()

driver.find\_element(By.CSS\_SELECTOR,".promoCode").send\_keys('rahulshettyacademy')

driver.find\_element(By.CSS\_SELECTOR,".promoBtn").click()

\*#Sum Validation\*

product\_prices= driver.find\_elements(By.CSS\_SELECTOR,"tr td:nth-child(5) p")

# print(len(product\_prices))

sum = 0

for price in product\_prices:

sum = sum+int(price.text)

total\_amount = driver.find\_element(By.CSS\_SELECTOR,".totAmt").text

assert sum == int(total\_amount)

wait = WebDriverWait(driver,10)

wait.until(expected\_conditions.presence\_of\_element\_located((By.CSS\_SELECTOR,".promoInfo")))

msg = driver.find\_element(By.CLASS\_NAME,"promoInfo").text

print(msg)

driver.close()

DATE 30012023 DAY MONDAY

Selenium Exception Handeling

try:

user\_name = driver.find\_element(By.CSS\_SELECTOR,"input[name='na']")#Name

user\_name.send\_keys('Krishna')

except NoSuchElementException:

print(traceback.format\_exc())

# driver.find\_element(By.CSS\_SELECTOR,"input[name='nam']").send\_keys('Krishna')

driver.find\_element(By.NAME,"email").send\_keys("krishn108@gmail.com")#email

driver.find\_element(By.ID,"exampleInputPassword1").send\_keys("1234567")#password

driver.find\_element(By.ID,"exampleCheck1").click()#check box

#Xpath and CSS\_SELECTOR

#//tagname[@attribute='value']

#CSS:- tagname[attribute='value'] #id, .classname

driver.find\_element(By.XPATH,"//input[@type='submit']").click()

message = driver.find\_element(By.CLASS\_NAME,"alert-success").text # sucess message

print(message)

assert 'Success' in message

=========================

[9:22 PM, 1/30/2023] KRISHNA SIR: Selenium Exception

An exception is an unexpected incident that is encountered during the program execution. It halts the normal execution flow of the program, can cause system crash or stop all the other processes.

Selenium exception classes are a part of selenium.common.execeptions. Let us discuss some of the common Selenium exceptions.

ElementClickInterceptedException

This exception is encountered if a click action could not be performed as the web element we are currently dealing with is hidden.

ElementNotInteractableException

This exception is encountered if we want to perform an action on the web element is unavailable in DOM.

ElementNotSelectableException

This exception is encountered with a web element having <select> tag. This exception comes if we try to select an option which is unavailable or not selectable.

ElementNotVisibleException

This exception is encountered if we want to perform an action on a web element which is invisible on the web page but present in DOM.

ErrorInResponseException

This exception is encountered if there exists an error on the server side.

ImeActivationFailedException

This exception is encountered if IME engine activation failed.

ImeNotAvailableException

This exception is encountered if IME support is unavailable from all the IME related method invocation.

InsecureCertificateException

This exception is encountered when a navigation by a user agent comes across a certificate warning. The warning from a certificate is generated by an improper TLS certificate.

InvalidArgumentException

This exception is encountered when improper arguments are passed to a command.

InvalidCookieDomainException

This exception is encountered in addition to a cookie from a different URL to the present URL.

InvalidCoordinatesException

This exception is encountered if we try to do mouse movement to an invalid coordinate.

InvalidElementStateException

This exception is encountered when an action is being performed on a web element which is in disabled state.

InvalidSelectorException

This exception is encountered if a selector fails to identify a web element on the page. This is common with customized xpath expressions having incorrect syntax or formation.

InvalidSessionIdException

This exception is encountered if the provided session id is not among the list of sessions which are active. This means the session id is inactive.

InvalidSwitchToTargetException

This exception is encountered if the user tries to switch to a nonexistent window or frame.

MoveTargetOutOfBoundsException

This exception is encountered when a user makes an attempt to drag or move a web element or cursor beyond the visible portion of the screen.

NoAlertPresentException

This exception is encountered if the user makes an attempt to switch to a nonexistent alert pop up.

NoSuchAttributeException

This exception is encountered if an attribute of a web element could not be traced.

NoSuchCookieException

This exception is encountered if there is no identical cookie in the list of available cookies in the active browser.

NoSuchElementException

This exception is encountered when there is no matching web element on the page.

NoSuchFrameException

This exception is encountered if the user makes an attempt to switch to a nonexistent frame.

NoSuchWindowException

This exception is encountered if the user makes an attempt to switch to a nonexistent browser window.

ScreenshotException

This exception is encountered when there is a failure to capture a screenshot of a page.

SessionNotCreatedException

This exception is encountered when there is a failure to build a new session.

StaleElementReferenceException

This exception is encountered if the web element has been deleted or detached from the DOM.

TimeoutException

This exception is encountered if the web element cannot be identified within the specified wait time provided by the user.

UnableToSetCookieException

This exception is encountered if the driver object could not position a cookie.

UnexpectedAlertPresentException

This exception is encountered due to the appearance of an unanticipated alert pop up.

UnexpectedTagNameException

This exception is encountered when a user attempts to access a web element with a class which does not work with a particular tag name. For example, using the Select class with an input tag.

WebDriverException

This exception extends the Runtime Exception class. WebDriverException class is the super class of all exceptions in Selenium.

[9:22 PM, 1/30/2023] KRISHNA SIR: TimeoutException

This exception is encountered if the web element cannot be identified within the specified wait time provided by the user.

UnableToSetCookieException

This exception is encountered if the driver object could not position a cookie.

UnexpectedAlertPresentException

This exception is encountered due to the appearance of an unanticipated alert pop up.

UnexpectedTagNameException

This exception is encountered when a user attempts to access a web element with a class which does not work with a particular tag name. For example, using the Select class with an input tag.

WebDriverException

This exception extends the Runtime Exception class. WebDriverException class is the super class of all exceptions in Selenium.

DATE 10022023 DAY FRIDAY

from selenium import webdriver

from selenium.webdriver.chrome.service import Service

from selenium.webdriver.common.by import By

import time

service\_obj = Service("C:\\ChromeDriver\\chromedriver.exe")

driver = webdriver.Chrome(service=service\_obj)

driver.get('https://www.selenium.dev/selenium/docs/api/java/index.html?overview-summary.html')

#link\_text

# driver.find\_element(By.LINK\_TEXT,"org.openqa.selenium.devtools.idealized.log").click() -- wrong approch

#Switch into frame

#switch\_to.frame(name)

#switch\_to.frame(id)

driver.switch\_to.frame("packageListFrame") # First Frame

driver.find\_element(By.LINK\_TEXT,"org.openqa.selenium.devtools").click()

time.sleep(3)

driver.switch\_to.default\_content()

driver.switch\_to.frame("packageFrame") #Frame-2

driver.find\_element(By.LINK\_TEXT,"Connection").click()

time.sleep(3)

driver.switch\_to.default\_content()

driver.switch\_to.frame("classFrame")#Frame-3

driver.find\_element(By.XPATH,"/html/body/header/nav/div[1]/div[1]/ul/li[6]").click()

time.sleep(3)

driver.switch\_to.default\_content()

time.sleep(5)

driver.close()

DATE 16022023 DAY THURSDAY

Selenium Interview Questions:-

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

1. Write Selenium Xpath

> Absolute and Relative

https://www.decathlon.in/search?query=tshirt - green check box (without use indexing) l1/l2

2.How many locaters in selenium, explain about it. l1/l2

3.Explain About selenium wait. l1/l2

4. How to do static dropdown? l1 :-

ans:-

#Select/Static Dropdown

dropdown = Select(driver.find\_element(By.ID,"exampleFormControlSelect1"))

dropdown.select\_by\_visible\_text("Male")

time.sleep(5)

dropdown.select\_by\_index(1)

5. How you identify auto suggestive drop down. l2

ans:-

driver.find\_element(By.ID,"autosuggest").send\_keys('ind')

time.sleep(2)

countries = driver.find\_elements(By.CSS\_SELECTOR,"li[class='ui-menu-item'] a")

for country in countries:

if country.text == "India":

country.click()

break

time.sleep(10)

6. How to click snap shot in selenium. l1

ans:- driver.get\_screenshot\_as\_file("screen\_shot\_1.png")

7. are you handel any alret,pop up or captcha? l2

>> Simple Alert , Prompt Alert ,Confirmation Alert

In Selenium 3x we can use switchto() for handeling alert

1.void dismiss :- driver.switchTo().alert().dismiss()

2.void accept :- driver.switchTo().alert().accept()

3.string getText:- driver.switchTo().alert().get().Text()

pop-up

------

driver.getWindowHandels()

8. Selenium Component? l2

1.selenium GRID

2.selenium IDE

3.selenium RC

4.selenium webdriver

9. What is selenium webdriver, expalin about yout tought. l1/l2

10. How can we get text of a web element? l1

11.what are the differenet way of selecting the values in dropdown? l1

13. deiver.quiet vs driver.close() l1

14.how to do handel multuple window? l1/l2

>> get.windowhandel()

>>set = get.windowhandels()

set.switchTo.window(id/name)

15. findelement vs findelements? l2

16. What Action class in selenium? l1

> Intreaction with API

> Keyboard and Mouse Event(imp)

>clickAndHold() :- current mouse location

contextClick():- Right Click

doubleClick():-

dragAndDrop

17. what are switch class in selenium? l1

18. A---> B----> C ----> A iframe l2/l1

19.driver.refersh() vs driver.back() vs driver.forward() l1

20. What is selenium navigation/navigate , explain about it. l2

21. css\_seletor nth -child? l1

22. link text vs partial text l1

Selenium notes python files:-

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

<https://github.com/ChandraSekharNayak-AT-Py/SeleniumAutomationPythonFiles.git>

Date 17022023 Day Friday

Selenium Window Handel

from selenium import webdriver

from selenium.webdriver.common.keys import Keys

# Replace the URL, username, and password with your own values

url = "https://example.com"

username = "yourusername"

password = "yourpassword"

# Create a new instance of the Firefox browser

driver = webdriver.Firefox()

# Navigate to the URL

driver.get(url)

# Find the username and password textboxes and enter the values

username\_box = driver.find\_element\_by\_name("username")

username\_box.send\_keys(username)

password\_box = driver.find\_element\_by\_name("password")

password\_box.send\_keys(password)

# Submit the form by pressing Enter

password\_box.send\_keys(Keys.RETURN)

# Switch to the new window that is opened after submitting the form

new\_window = driver.window\_handles[1]

driver.switch\_to.window(new\_window)

# Find the textbox and read the text

text\_box = driver.find\_element\_by\_name("text\_box\_name")

text = text\_box.get\_attribute("value")

print(text)

# Close the browser

driver.quit()

PYTEST:

Pytest

pip install:-

-----------

curl https://bootstrap.pypa.io/get-pip.py -o get-pip.py

python get-pip.py

Upgarde pip:-

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

python -m pip install --upgrade pip

Install:-

-------

pip install pytest

or

pip install pytest==7.1.2

Run The Test:-

-----------

All Test Case:- py.test

Versious Test :- py.test -v

ALL :- py.test -v -s

GIT HUB :-PYTEST NOTES

<https://github.com/ChandraSekharNayak-AT-Py/PyTestCode>

RESULT:

