Vishal Salvi - 2019230069

Shivam Pawar - 2019230068

Batch C

EXPERIMENT NO. 8

Aim: To Perform automating testing of web application using selenium automated testing tool

Theory:

What Is Selenium?

Selenium is an open-source tool that is used for automating the tests carried out on web browsers (Web applications are tested using any web browser).

Wait, before you get carried away, let me re-iterate that, only testing of web applications is possible with Selenium. We can neither test any desktop (software) application nor test any mobile application using Selenium.

It' a bummer right? I can feel your pain. But don't worry, there are many tools for testing software and mobile applications like: IBM's RFT, HP's QPT, Appium and many more. But, the focus of this blog is, testing dynamic web applications and why Selenium is the best for that purpose.

What are the advantages of Selenium?

Since Selenium is open-source, there is no licensing cost involved, which is a major advantage over other testing tools. Other reasons behind Selenium's ever growing popularity are:

- Test scripts can be written in any of these programming languages: Java, Python, C#, PHP, Ruby, Perl & .Net
- Tests can be carried out in any of these OS: Windows, Mac or Linux
- Tests can be carried out using any browser: Mozilla Firefox, Internet Explorer, Google Chrome, Safari or Opera
- It can be integrated with tools such as TestNG & JUnit for managing test cases and generating reports
- It can be integrated with Maven, Jenkins & Docker to achieve Continuous Testing

But there surely has to be shortcomings right?

- We can use Selenium only to test web applications. We cannot test desktop applications or any other software
- There is no guaranteed support available for Selenium. We need to leverage on the available customer communities
- It is not possible to perform testing on images. We need to integrate Selenium with Sikuli for image based testing
- There is no native reporting facility. But we can overcome that issue by integrating it with frameworks like TestNG or JUnit

Need For Software Testing

Software testing is where it all boils down to. Today’s world of technology is completely dominated by machines, and their behavior is controlled by the software powering it. Will the machines behave exactly as we want them to? Everytime? Everywhere? The answer to these questions lie in software testing.

At the end of the day, it is the software application’s success rate which is going to control your business growth. The same thing can be said even for web applications because most businesses today are completely reliant on the internet.

Take for example, any e-commerce company. Be it Amazon or E-Bay or Flipkart, they rely on the customer traffic on their web sites and traffic on their web based mobile applications for business.

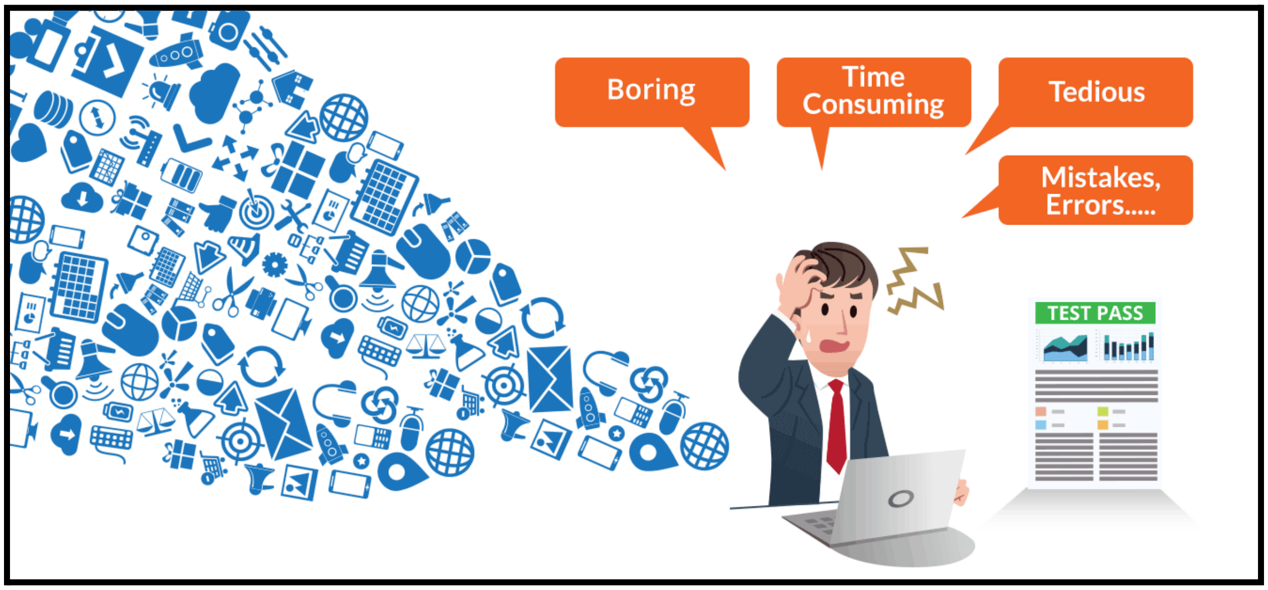
Imagine, if something catastrophic happens like the prices of a number of products being capped off at 10$, all because of a small bug in a "not so easily readable" part of the code. Then what can be done, and how can we prevent it the next time?

By testing the code before deployment right? So, that is the need for software testing. But, what is Selenium? Is it a software testing tool? Well, Selenium is an automation testing tool!

Software testing is of two types: Manual Testing & Automation Testing. Selenium was founded as an automation testing tool to overcome the drawbacks/ limitations of Manual testing.

Challenges With Manual Testing

Manual testing means the (web) application is tested manually by QA testers. Tests need to be performed manually in every environment, using a different data set and the success/ failure rate of every transaction should be recorded.



Look at the above image of a poor chap, who manually verifies the transactions recorded. The challenges he is facing cause fatigue, boredom, delay in work, mistakes and errors because of manual effort. This lead to the invention of Selenium (automation testing tool).

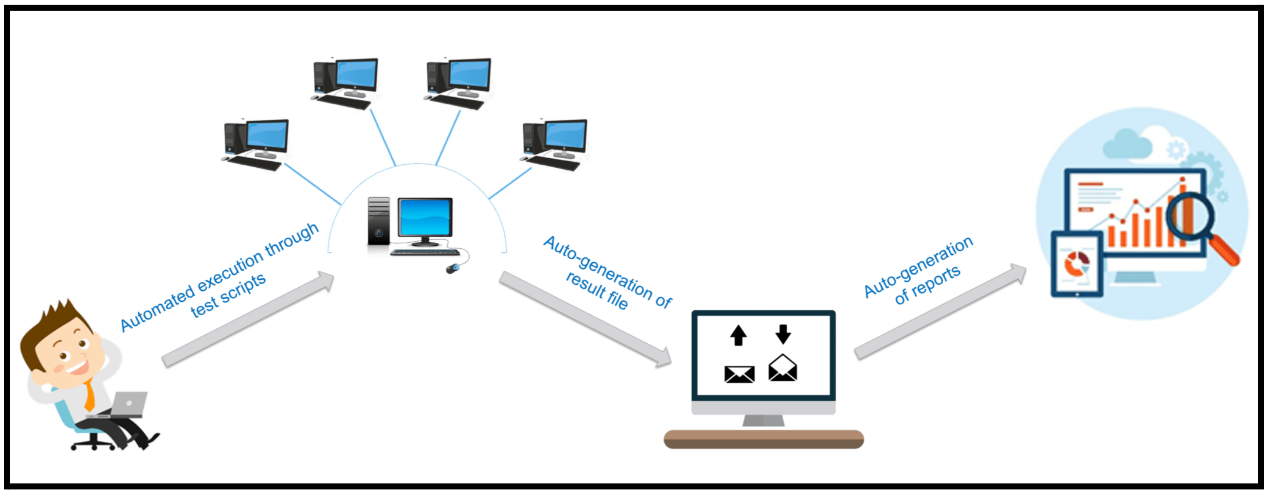
Automation Testing Beats Manual Testing

Automation testing beats manual testing every time. Why? Because it is faster, needs less investment in human resource, it is not prone to errors, frequent execution of tests is possible, supports lights out execution, supports regression testing and also functional testing.

Let's take a similar example to the one mentioned earlier. Suppose there is a login page and we need to verify if all the login attempts are successful, then it will be really easy to write a piece of code which will validate if all the transaction/ login attempts are a success or not (automated test case execution).

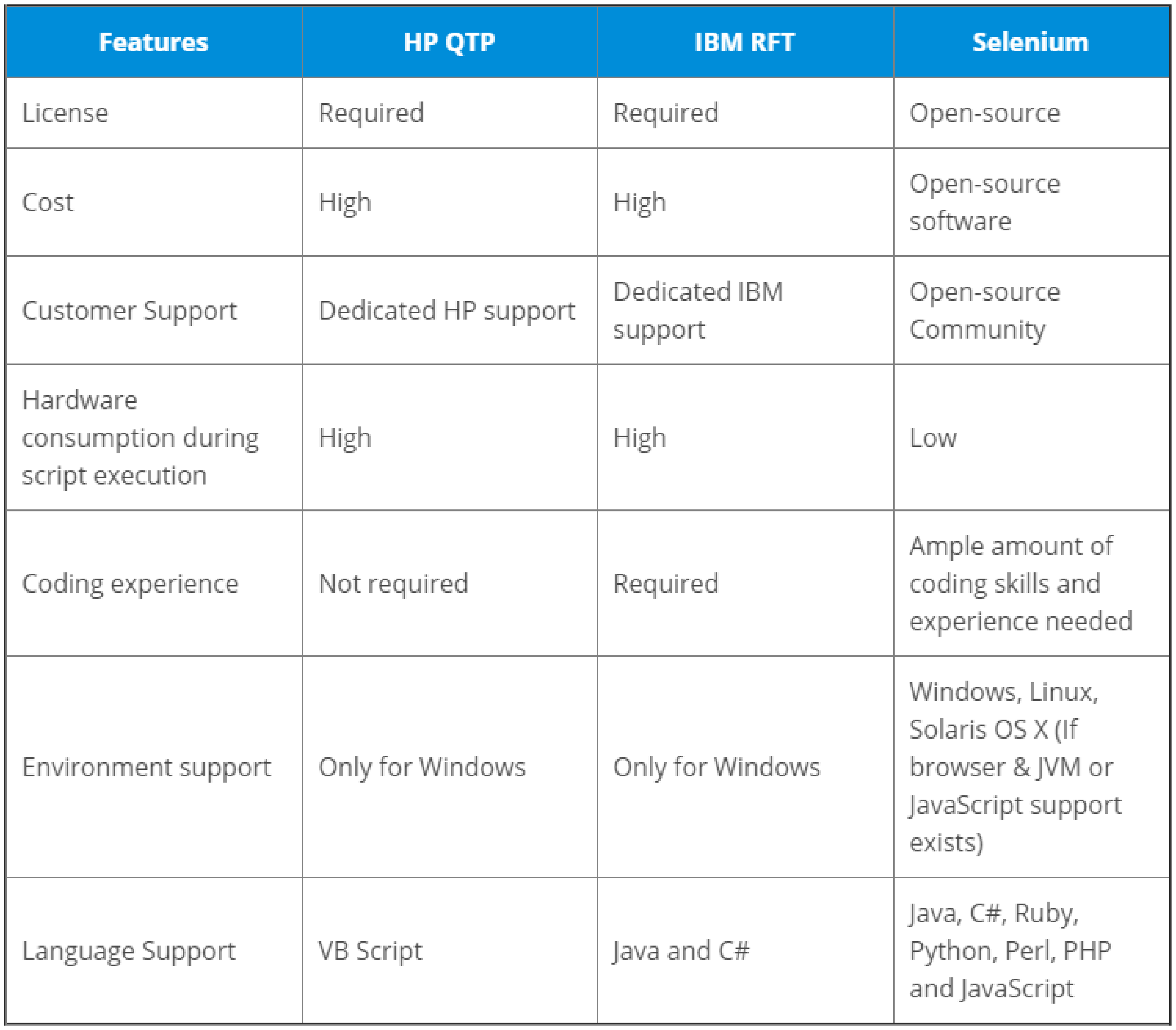
Moreover, these tests can be configured in such a way that they are tested in different environments and web browsers. What else can be done? You can automate the generation of result file, by scheduling it for a particular time during the day. Then you can also automate the generation of reports based on those results and what not.

The key point is that automation testing makes a tester's job a whole lot simpler. Check out the image below which shows a more relaxed environment in which the same tester is working.


Automated execution through test scripts
Auto-generation of result file
Auto-generation of reports

Now let us see where Selenium stands in the market.

Selenium vs QTP vs RFT



Features	HP QTP	IBM RFT	Selenium
License	Required	Required	Open-source
Cost	High	High	Open-source software
Customer Support	Dedicated HP support	Dedicated IBM support	Open-source Community
Hardware consumption during script execution	High	High	Low
Coding experience	Not required	Required	Ample amount of coding skills and experience needed
Environment support	Only for Windows	Only for Windows	Windows, Linux, Solaris OS X (If browser & JVM or JavaScript support exists)
Language Support	VB Script	Java and C#	Java, C#, Ruby, Python, Perl, PHP and JavaScript

It is pretty clear from the above table why Selenium is the most preferred tool. But there are many flavors in Selenium and you should know which is the appropriate right Selenium tool for your need.

Selenium Suite Of Tools

- Selenium RC (Now depreciated)
- Selenium IDE
- Selenium Grid
- Selenium WebDriver

What are the components of Selenium?

Selenium RC (Remote Control)

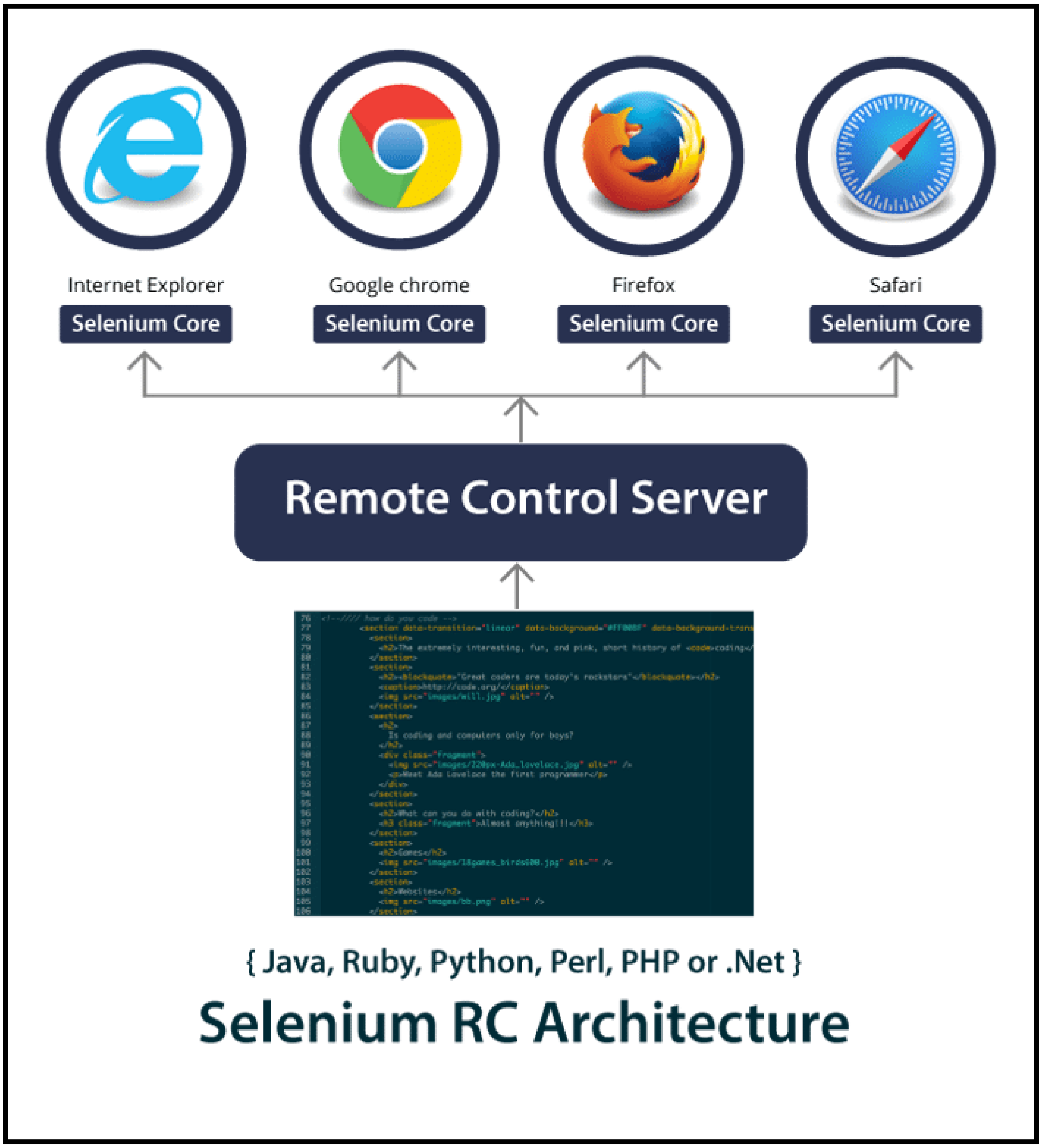
Before I talk about the details of Selenium RC, I would like to go a step back and talk about the first tool in the Selenium project. Selenium Core was the first tool. But, Selenium Core hit a roadblock in terms of cross-domain testing because of the same-origin policy. The same-origin policy prohibits JavaScript code from accessing web elements that are hosted on a different domain compared to where the JavaScript was launched.

To overcome the same-origin policy issue, testers needed to install local copies of both Selenium Core (a JavaScript program) and the webserver containing the web application being tested so they would belong to the same domain. This lead to the birth of Selenium RC, which is accredited to then ThoughtWork's engineer, Paul Hammant.

RC overcame the problem by involving an HTTP proxy server to "trick" the browser into believing that Selenium Core and the web application being tested come from the same domain. Thus making RC a two-component tool.

1. Selenium RC Server
2. Selenium RC Client – Library containing your programming language code

RC Server communicates using simple HTTP GET/ POST requests. Look at the below image for understanding the RC architecture.



```
<!--//// how do you code -->
        <section data-transition="linear" data-background="#FF008F" data-background-trans
          <section>
            <h2>The extremely interesting, fun, and pink, short history of <code>coding</
          </section>
          <section>
            <h2><blockquote>"Great coders are today's rockstars"</blockquote></h2>
            <caption>http://code.org/</caption>
            <img src="images/will.jpg" alt="" />
          </section>
          <section>
            <h2>
              Is coding and computers only for boys?
            </h2>
            <div class="fragment">
              <img src="images/220px-Ada_lovelace.jpg" alt="" />
              <p>Meet Ada Lovelace the first programmer</p>
            </div>
          </section>
          <section>
            <h2>What can you do with coding?</h2>
            <h3 class="fragment">Almost anything!!!</h3>
          </section>
          <section>
            <h2>Games</h2>
            <img src="images/18games_birds600.jpg" alt="" />
          </section>
          <section>
            <h2>Websites</h2>
            <img src="images/bb.png" alt="" />
          </section>
```

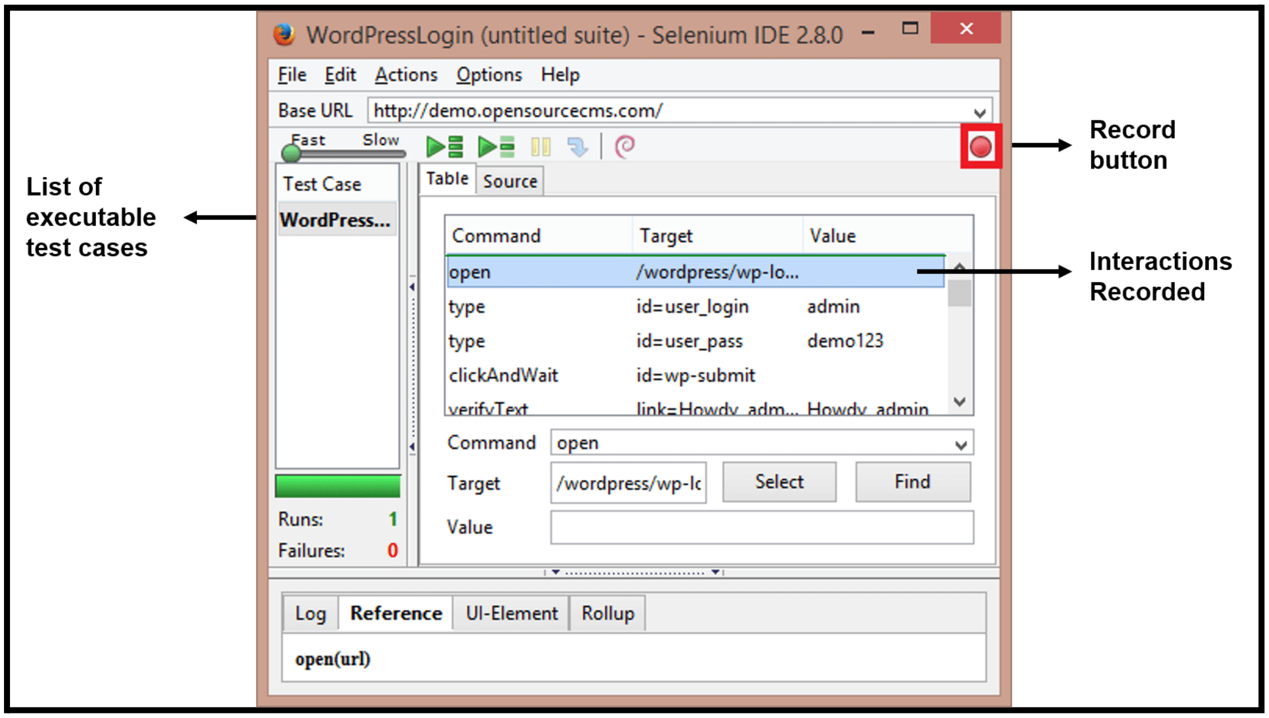
Selenium project's flagship tool was Selenium RC as it was their first tool and it could be used to write test cases in different programming languages. But the drawback with RC is that every communication with the RC server is time consuming and hence RC is very slow. So slow, that it would sometimes take hours to complete single tests.

From Selenium v3 onwards, RC has been depreciated and moved to legacy package. You can however download and work with RC, but unfortunately you cannot avail support for it. But on the flip side, why would you want to use a tool which is outdated, especially when there is a more efficient tool called Selenium WebDriver. Before I talk about WebDriver, let me discuss about IDE and Grid, which are the other tools that make up Selenium v1.

Selenium IDE (Integrated Development Environment)

In 2006, Shinya Kastani from Japan had donated his Selenium IDE prototype to Apache's Selenium project. It was a Firefox plugin for faster creation of test cases. IDE implemented a record and playback model wherein, test cases are created by recording the interactions which the user had with the web browser. These tests can then be played back any number of times.

The advantage with Selenium IDE is that, tests recorded via the plugin can be exported in different programming languages like: Java, Ruby, Python etc. Check out the below screenshot of Firefox's IDE plugin.


WordPressLogin (untitled suite) - Selenium IDE 2.8.0
File Edit Actions Options Help
Fast Slow
Test Case
WordPress...
Table Source
Select
Find
Log Reference UI-Element Rollup
List of executable test cases
Record button
Interactions Recorded

Base URL http://demo.opensourcecms.com/

Command	Target	Value
open	/wordpress/wp-lo...	
type	id=user_login	admin
type	id=user_pass	demo123
clickAndWait	id=wp-submit	
verifyText	link=Howdy, adm...	Howdy, admin

Command open
Target /wordpress/wp-lc
Value

Runs: 1
Failures: 0

open(url)

But the associated shortcomings of IDE are:

- Plug-in only available for Mozilla Firefox; not for other browsers

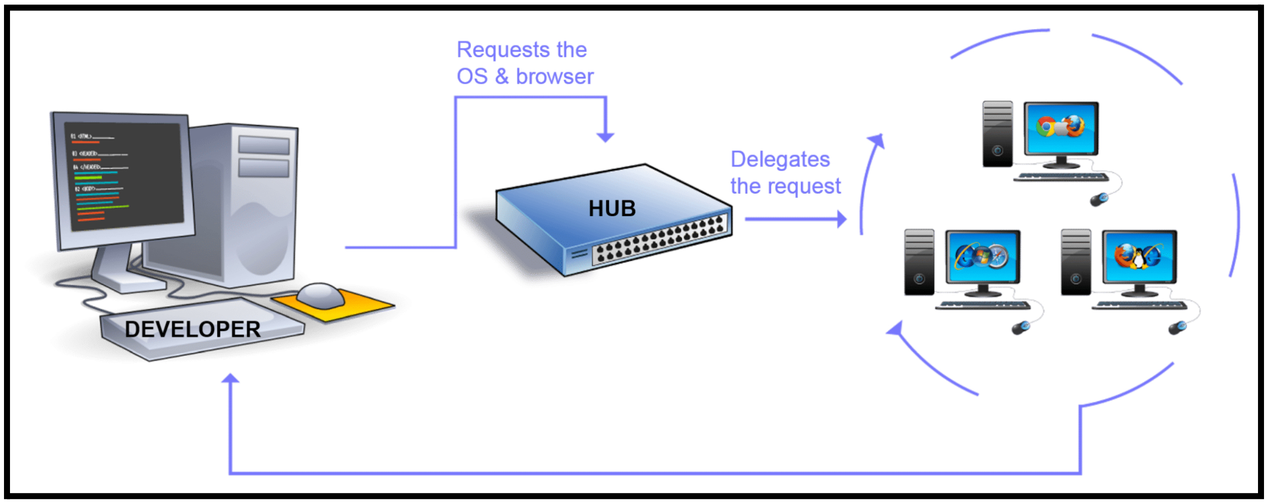
- It is not possible to test dynamic web applications; only simple tests can be recorded
- Test cases cannot be scripted using programming logic
- Does not support Data-Driven testing

These were some of the aspects of Selenium IDE. Let me now talk about Selenium Grid.

What is Selenium Grid

Selenium Grid was developed by Patrick Lightbody and initially called HostedQA (initially a part of Selenium v1) and it was used in combination with RC to run tests on remote machines. In fact, with Grid, multiple test scripts can be executed at the same time on multiple machines.

Parallel execution is achieved with the help of Hub-Node architecture. One machine will assume the role of Hub and the others will be the Nodes. Hub controls the test scripts running on various browsers inside various operating systems. Test scripts being executed on different Nodes can be written in different programming languages.



Grid is still in use and works with both WebDriver and RC. However, maintaining a grid with all required browsers and operating systems is a challenge. For this, there are multiple online platforms that provide an online Selenium Grid that you can access to run your selenium automation scripts. For example, you can use LambdaTest. It has more than 2000 browser environments over which you can run your tests and truly automate cross-browser testing.

What is Selenium WebDriver

Founded by Simon Stewart in 2006, Selenium WebDriver was the first cross-platform testing framework that could control the browser from OS level. In contrast to IDE, Selenium WebDriver provides a programming interface to create and execute test cases. Test cases are written such that, web elements on web pages are identified and then actions are performed on those elements.

WebDriver is an upgrade to RC because it is much faster. It is faster because it makes direct calls to the browser. RC on the other hand needs an RC server to interact with the web browser. Each browser has its own driver on which the application runs. The different WebDrivers are:

- Firefox Driver (Gecko Driver)
- Chrome Driver
- Internet Explorer Driver
- Opera Driver
- Safari Driver and
- HTM Unit Driver

Benefits Of Selenium WebDriver

- Support for 7 programming languages: JAVA, C#, PHP, Ruby, Perl, Python and .Net.
- Supports testing on various browsers like: Firefox, Chrome, IE, Safari
- Tests can be performed on different operating systems like: Windows, Mac, Linux, Android, iOS
- Overcomes limitations of Selenium v1 like file upload, download, pop-ups & dialogs barrier

Short-comings Of Selenium WebDriver

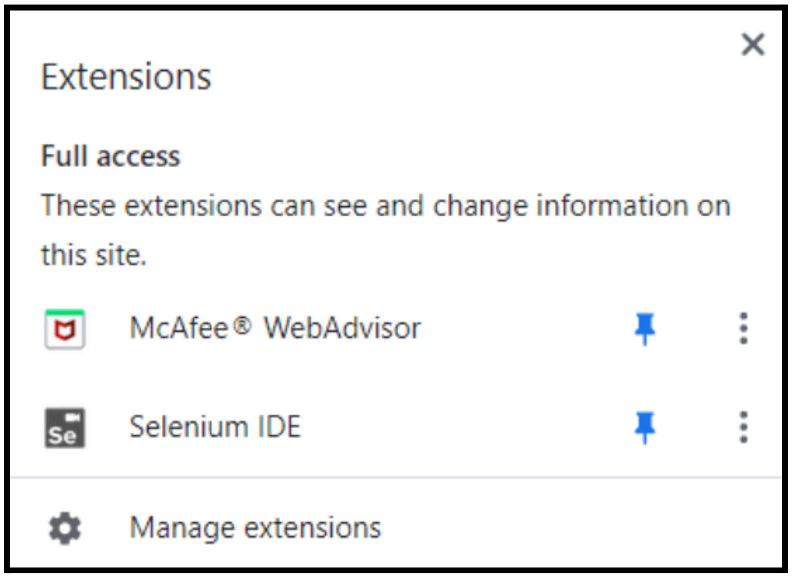
- Detailed test reports cannot be generated
- Testing images is not possible

No matter the challenge, these shortcomings can be overcome by integrations with other frameworks. For testing images, Sikuli can be used, and for generating detailed test reports, TestNG can be used.

Implementation:

Steps in creating a Test case for our project:

1. Click on the Selenium IDE extension:


Extensions

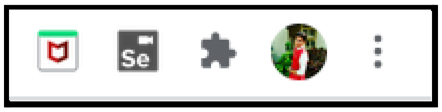
Full access

These extensions can see and change information on this site.

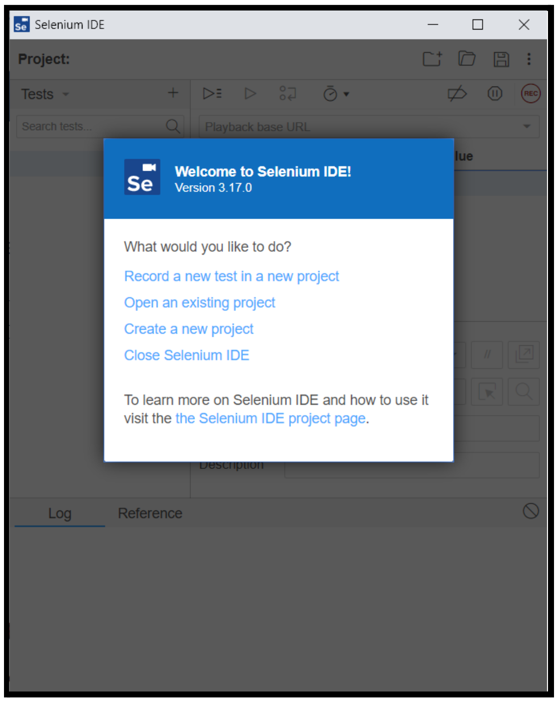
McAfee® WebAdvisor

Selenium IDE

Manage extensions



2. Select "record a new test in a new project":



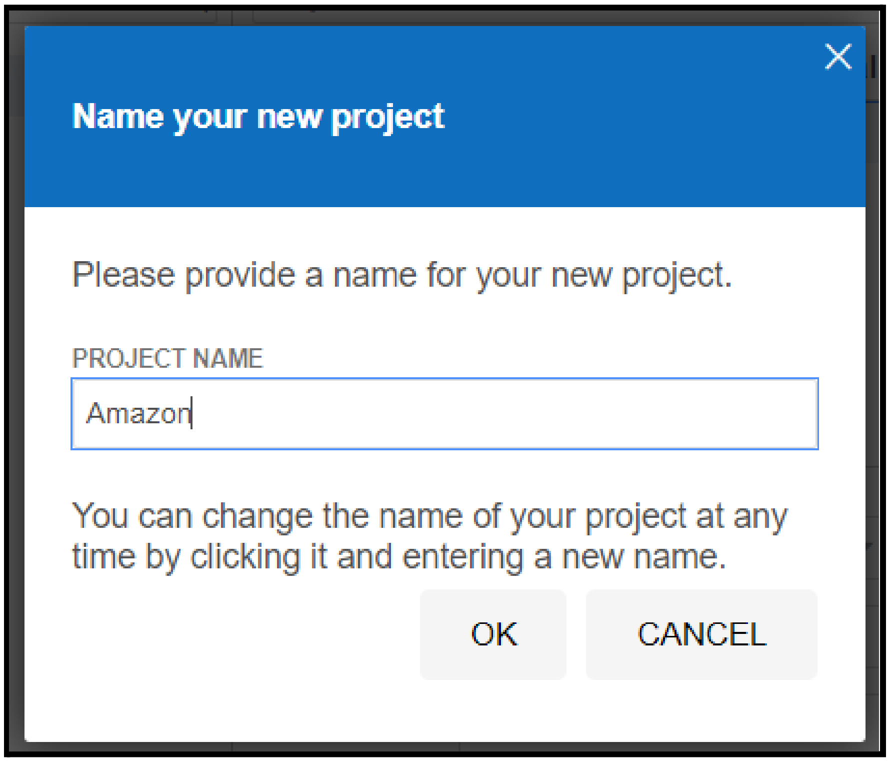
Welcome to Selenium IDE!
Version 3.17.0

What would you like to do?

Record a new test in a new project
Open an existing project
Create a new project
Close Selenium IDE

To learn more on Selenium IDE and how to use it visit the the Selenium IDE project page.

3. Enter name of your Project:



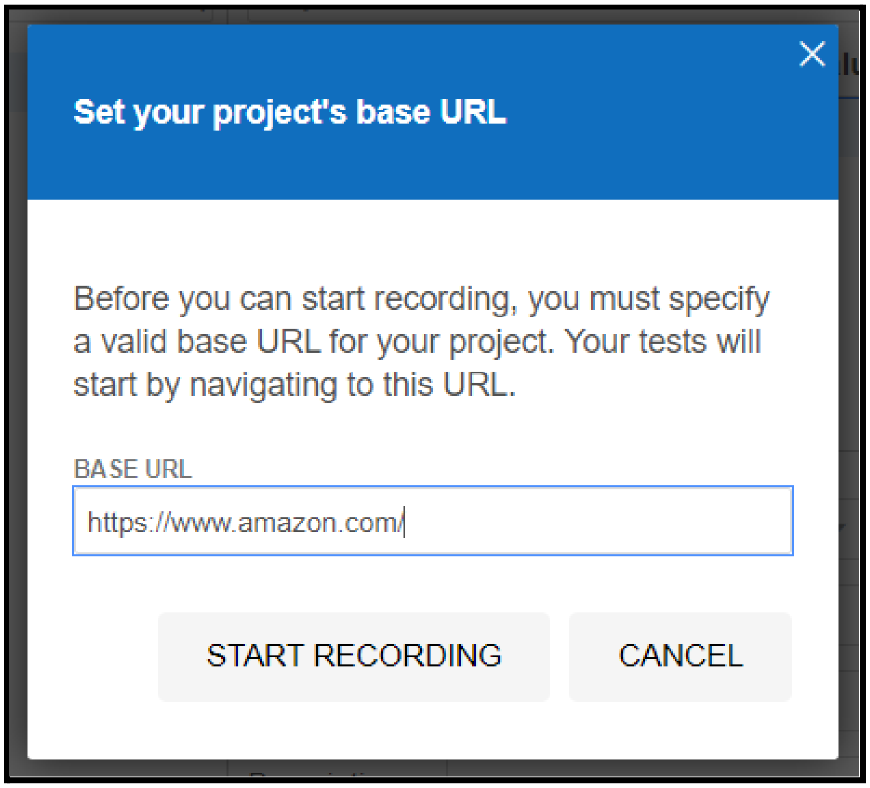
Name your new project

Please provide a name for your new project.

PROJECT NAME: Amazon

You can change the name of your project at any time by clicking it and entering a new name.

4. Enter base url and start recording:

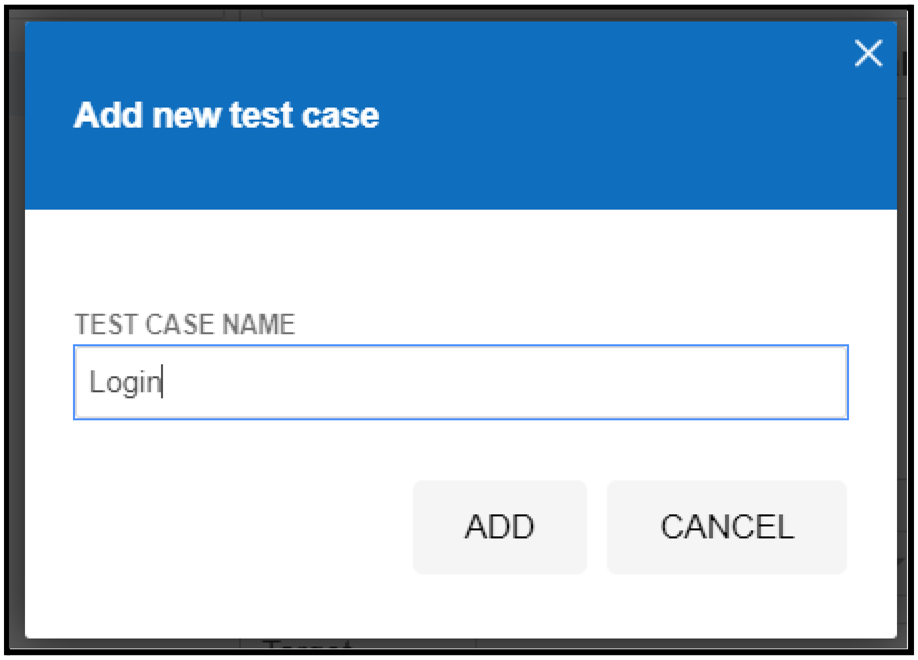


Set your project's base URL

Before you can start recording, you must specify a valid base URL for your project. Your tests will start by navigating to this URL.

BASE URL: https://www.amazon.com/

5. Enter name of test case:

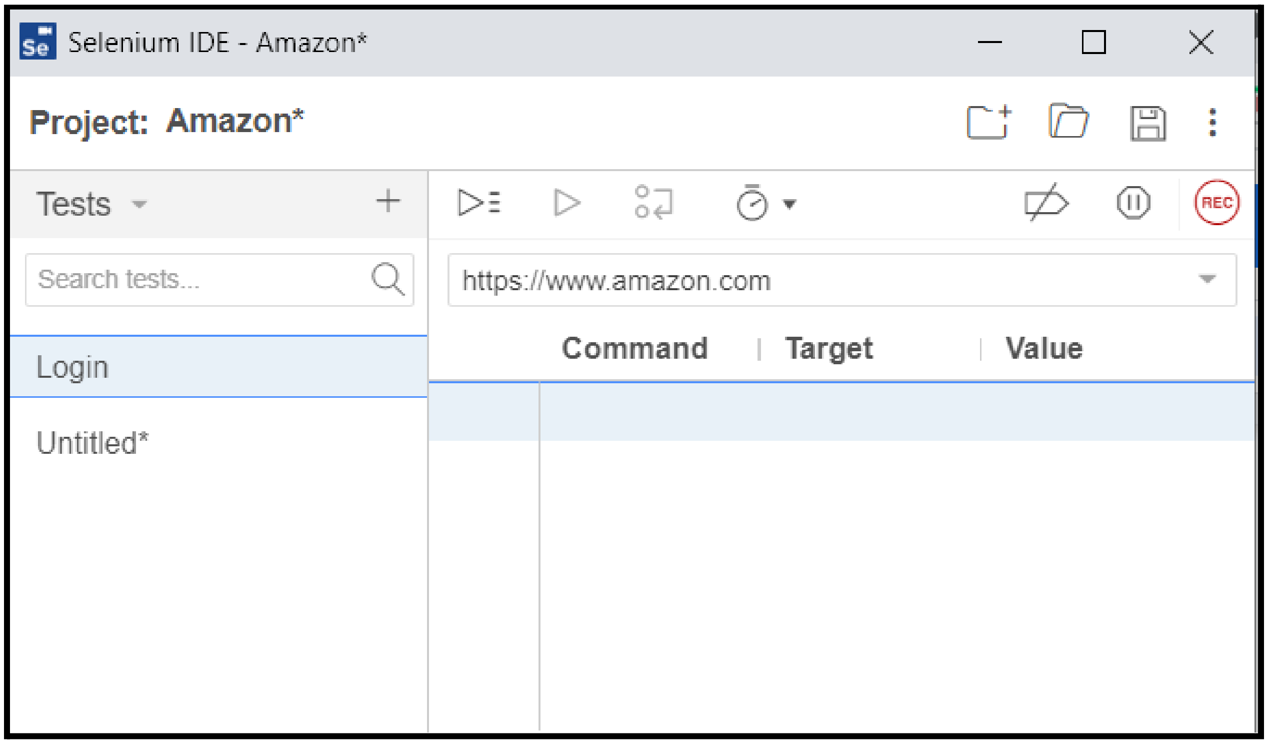


Add new test case

TEST CASE NAME

Login

6. Click on start recording



Selenium IDE - Amazon*

Project: Amazon*

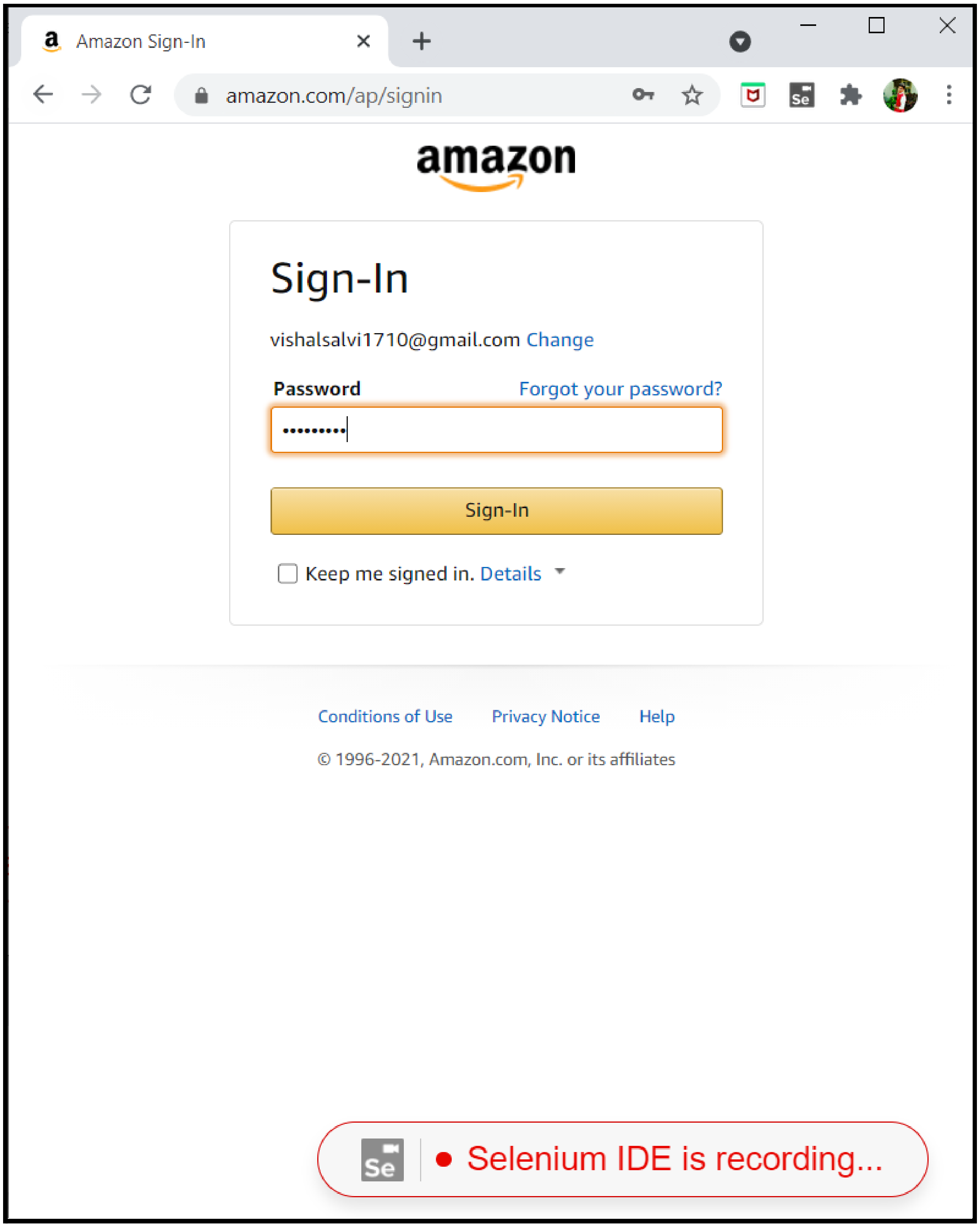
https://www.amazon.com

Command	Target	Value

Login

Untitled*

Log in:



Amazon Sign-In

amazon.com/ap/signin

Sign-In

vishalsalvi1710@gmail.com Change

Password Forgot your password?

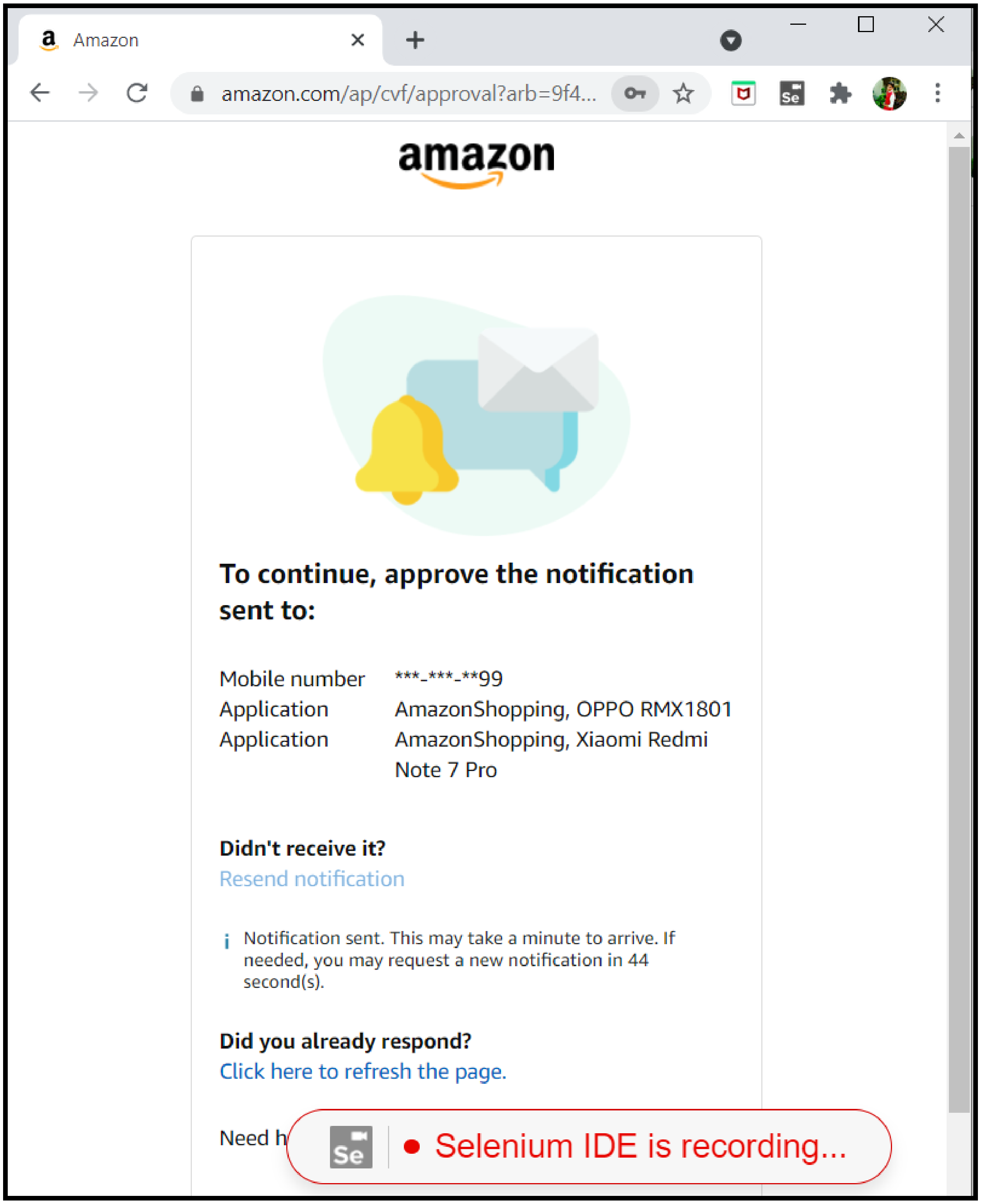
Sign-In

Keep me signed in. Details

Conditions of Use Privacy Notice Help

© 1996-2021, Amazon.com, Inc. or its affiliates

Selenium IDE is recording...



Amazon

amazon.com/ap/cvf/approval?arb=9f4...

To continue, approve the notification sent to:

Mobile number	***-***-**99
Application	AmazonShopping, OPPO RMX1801
Application	AmazonShopping, Xiaomi Redmi Note 7 Pro

Didn't receive it?
Resend notification

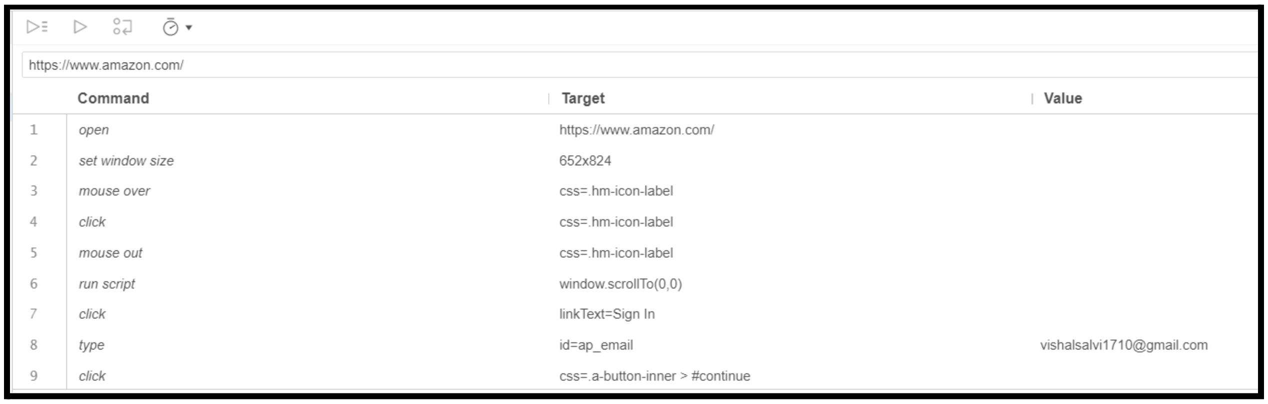
Notification sent. This may take a minute to arrive. If needed, you may request a new notification in 44 second(s).

Did you already respond?
Click here to refresh the page.

Need h

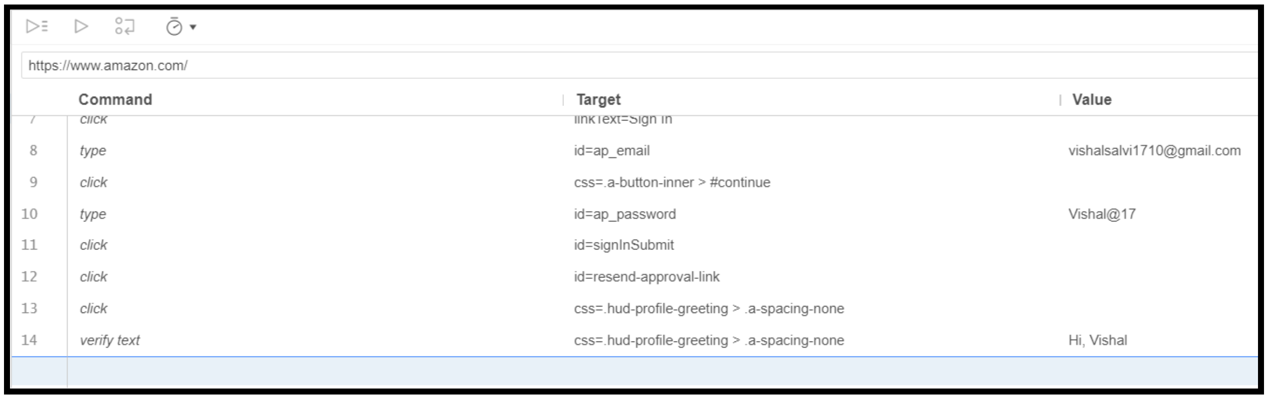
Selenium IDE is recording...

Command:



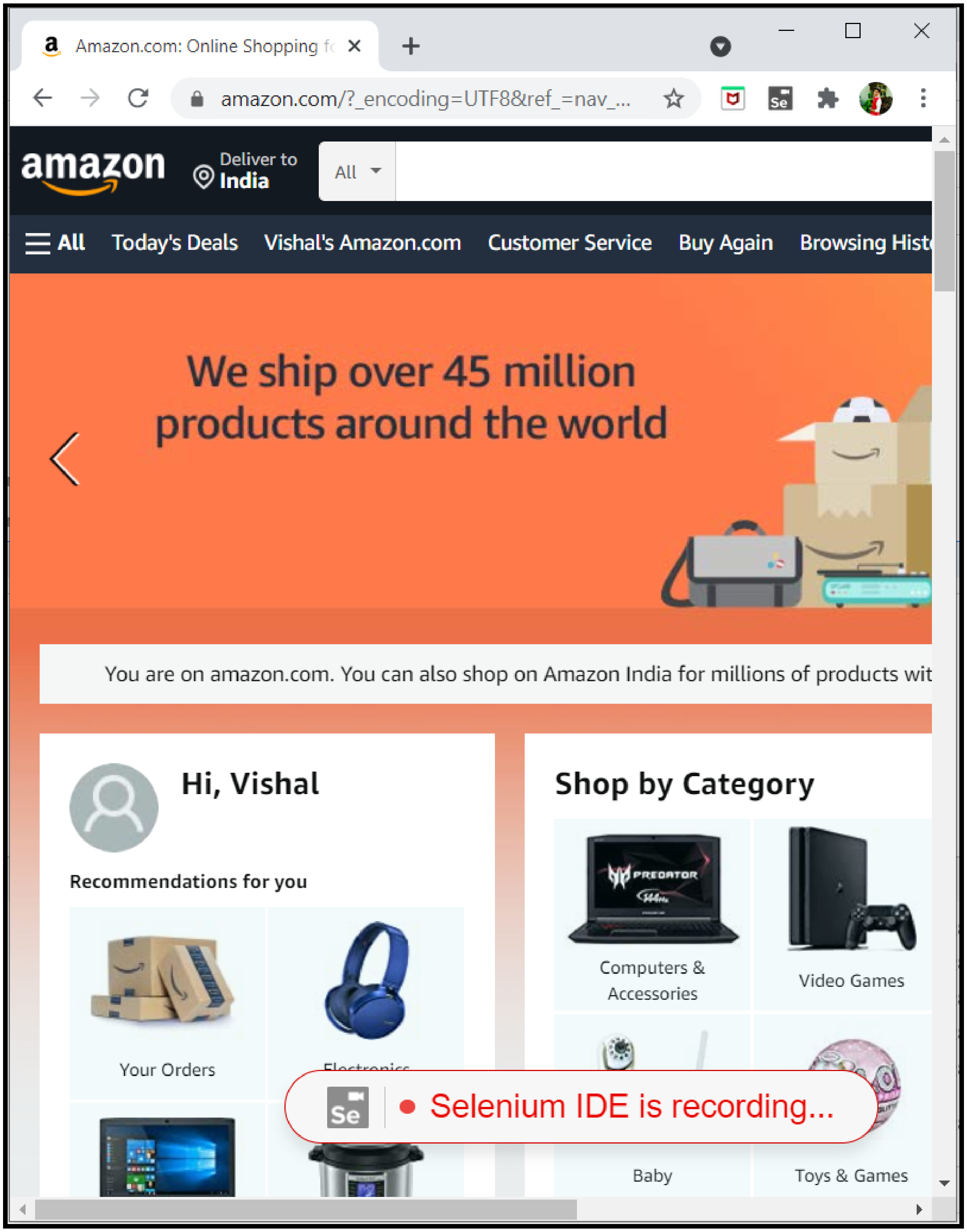
https://www.amazon.com/

	Command	Target	Value
1	open	https://www.amazon.com/	
2	set window size	652x824	
3	mouse over	css=.hm-icon-label	
4	click	css=.hm-icon-label	
5	mouse out	css=.hm-icon-label	
6	run script	window.scrollTo(0,0)	
7	click	linkText=Sign In	
8	type	id=ap_email	vishalsalvi1710@gmail.com
9	click	css=.a-button-inner > #continue	



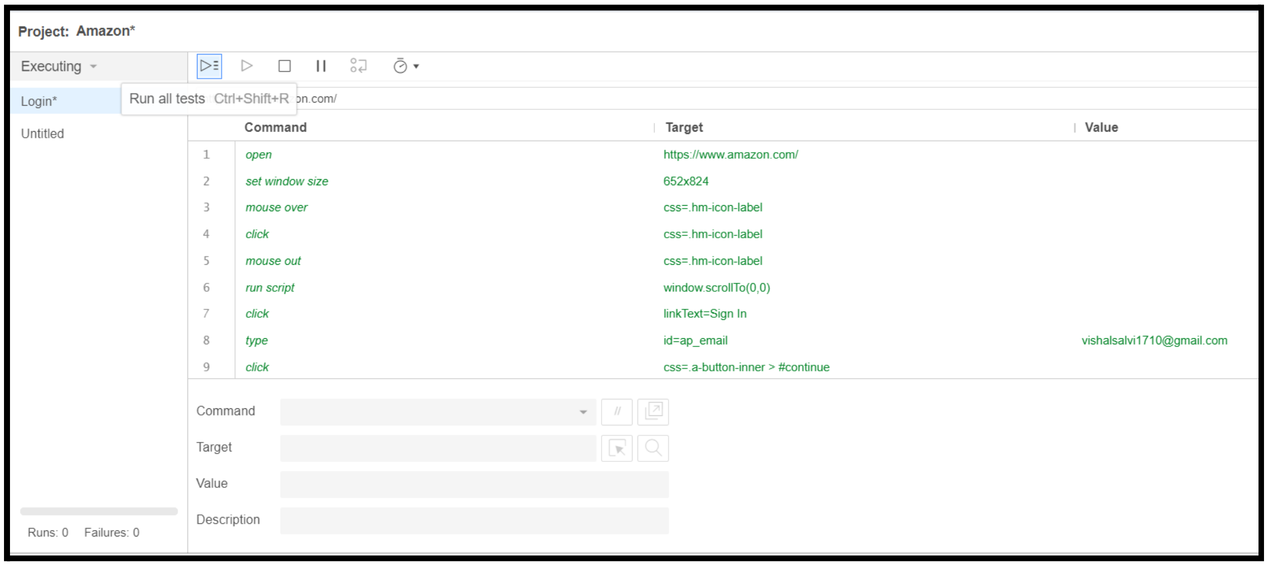
https://www.amazon.com/

	Command	Target	Value
7	click	linkText=Sign In	
8	type	id=ap_email	vishalsalvi1710@gmail.com
9	click	css=.a-button-inner > #continue	
10	type	id=ap_password	Vishal@17
11	click	id=signInSubmit	
12	click	id=resend-approval-link	
13	click	css=.hud-profile-greeting > .a-spacing-none	
14	verify text	css=.hud-profile-greeting > .a-spacing-none	Hi, Vishal



Amazon.com: Online Shopping f
amazon.com/?_encoding=UTF8&ref_=nav_...
amazon
Deliver to
India
All
All
Today's Deals
Vishal's Amazon.com
Customer Service
Buy Again
Browsing Hist
We ship over 45 million products around the world
You are on amazon.com. You can also shop on Amazon India for millions of products wit
Hi, Vishal
Recommendations for you
Your Orders
Electronics
Shop by Category
Computers & Accessories
Video Games
Baby
Toys & Games
Selenium IDE is recording...

Testing login case:


Project: Amazon*

Executing

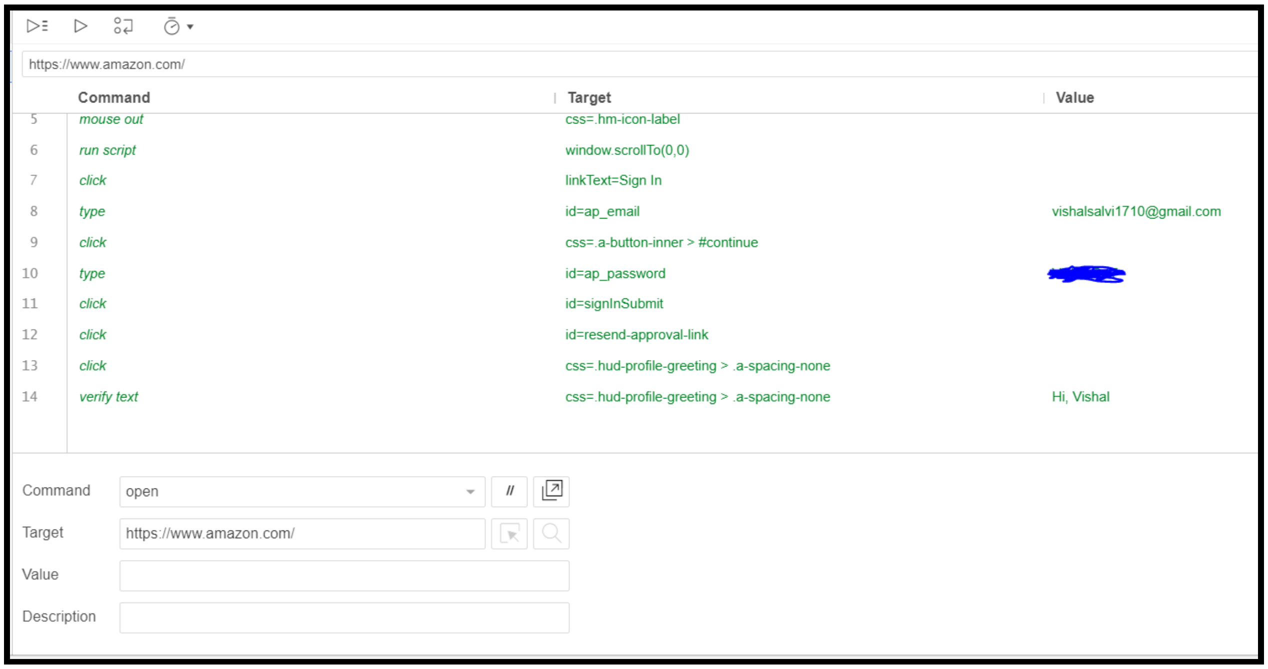
Login*

Untitled

Run all tests Ctrl+Shift+R

Command	Target	Value
open	https://www.amazon.com/	
set window size	652x824	
mouse over	css=.hm-icon-label	
click	css=.hm-icon-label	
mouse out	css=.hm-icon-label	
run script	window.scrollTo(0,0)	
click	linkText=Sign In	
type	id=ap_email	vishalsalvi1710@gmail.com
click	css=.a-button-inner > #continue	

Runs: 0 Failures: 0


https://www.amazon.com/

Command	Target	Value
mouse out	css=.hm-icon-label	
run script	window.scrollTo(0,0)	
click	linkText=Sign In	
type	id=ap_email	vishalsalvi1710@gmail.com
click	css=.a-button-inner > #continue	
type	id=ap_password	[redacted]
click	id=signInSubmit	
click	id=resend-approval-link	
click	css=.hud-profile-greeting > .a-spacing-none	
verify text	css=.hud-profile-greeting > .a-spacing-none	Hi, Vishal

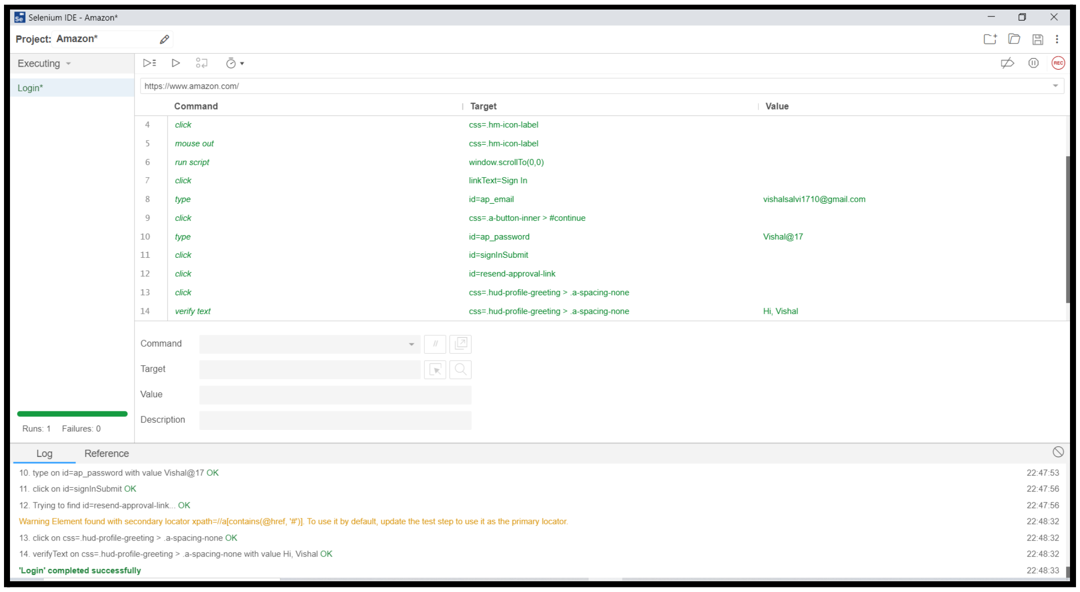
Command: open

Target: https://www.amazon.com/

Value:

Description:

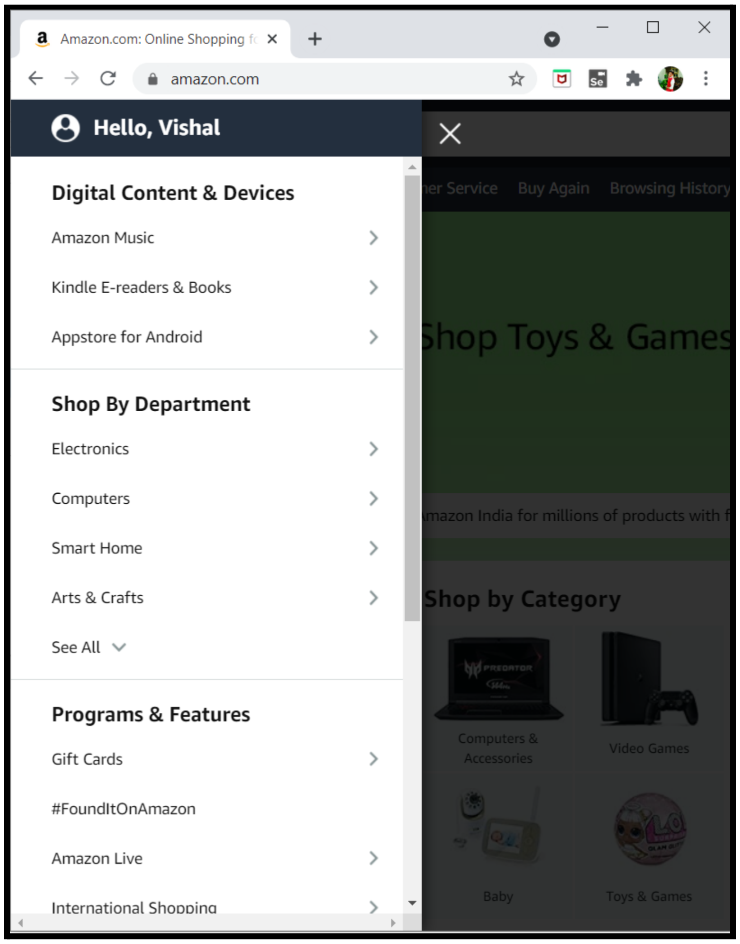
Test case Successful:

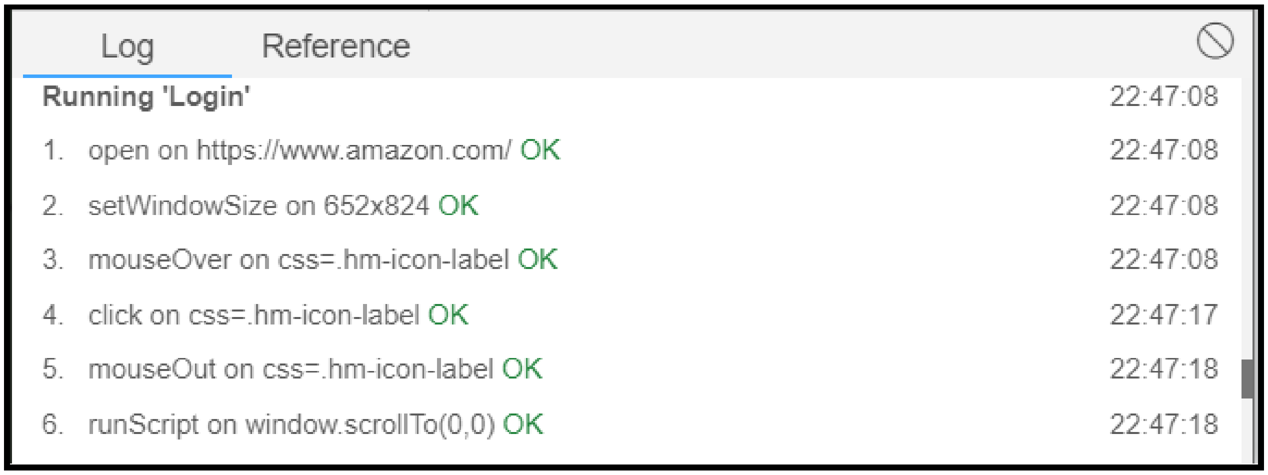


	Command	Target	Value
4	click	css=.hm-icon-label	
5	mouse out	css=.hm-icon-label	
6	run script	window.scrollTo(0,0)	
7	click	linkText=Sign In	
8	type	id=ap_email	vishalsalvi1710@gmail.com
9	click	css=.a-button-inner > #continue	
10	type	id=ap_password	Vishal@17
11	click	id=signInSubmit	
12	click	id=resend-approval-link	
13	click	css=.hud-profile-greeting > .a-spacing-none	
14	verify text	css=.hud-profile-greeting > .a-spacing-none	Hi, Vishal

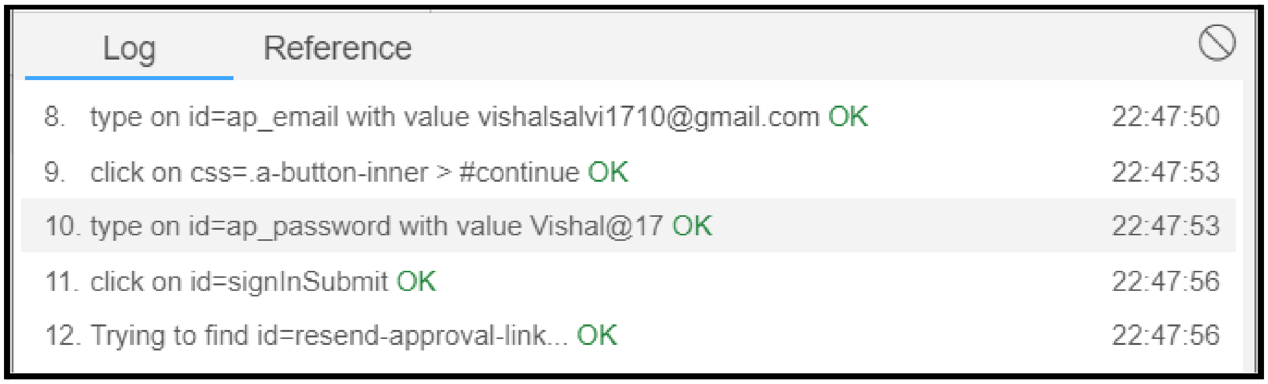
```
10. type on id=ap_password with value Vishal@17 OK                                              22:47:53
11. click on id=signInSubmit OK                                                                 22:47:56
12. Trying to find id=resend-approval-link... OK                                                22:47:56
Warning Element found with secondary locator xpath=//a[contains(@href, '#')]. To use it by default, update the test step to use it as the primary locator.   22:48:32
13. click on css=.hud-profile-greeting > .a-spacing-none OK                                     22:48:32
14. verifyText on css=.hud-profile-greeting > .a-spacing-none with value Hi, Vishal OK          22:48:32
'Login' completed successfully                                                                  22:48:33
```

Result:

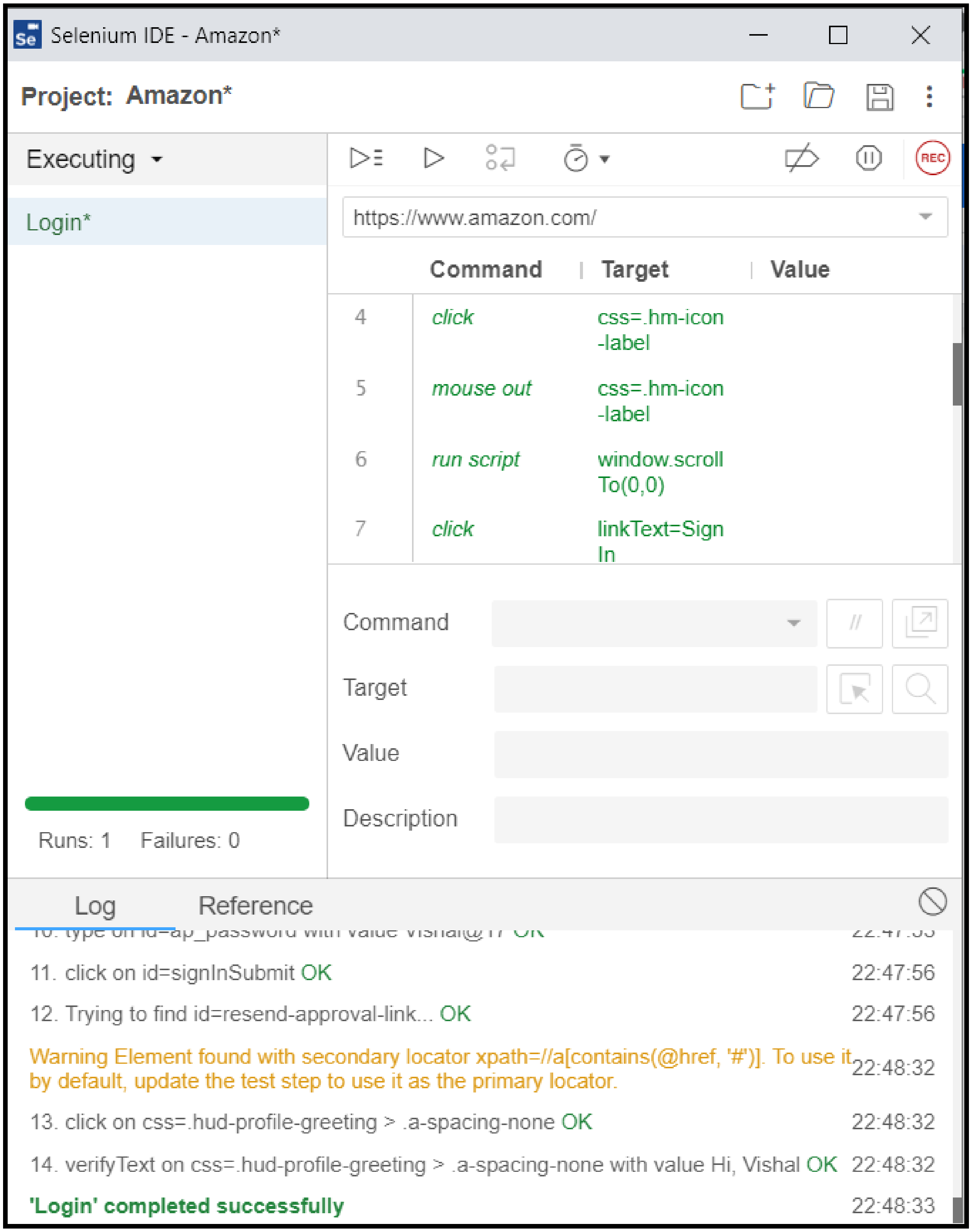




```
Running 'Login'                                        22:47:08
1.  open on https://www.amazon.com/ OK                 22:47:08
2.  setWindowSize on 652x824 OK                        22:47:08
3.  mouseOver on css=.hm-icon-label OK                 22:47:08
4.  click on css=.hm-icon-label OK                     22:47:17
5.  mouseOut on css=.hm-icon-label OK                  22:47:18
6.  runScript on window.scrollTo(0,0) OK               22:47:18
```



```
8.  type on id=ap_email with value vishalsalvi1710@gmail.com OK   22:47:50
9.  click on css=.a-button-inner > #continue OK                   22:47:53
10. type on id=ap_password with value Vishal@17 OK                22:47:53
11. click on id=signInSubmit OK                                   22:47:56
12. Trying to find id=resend-approval-link... OK                  22:47:56
```



https://www.amazon.com/

	Command	Target	Value
4	*click*	css=.hm-icon-label	
5	*mouse out*	css=.hm-icon-label	
6	*run script*	window.scrollTo(0,0)	
7	*click*	linkText=Sign In	

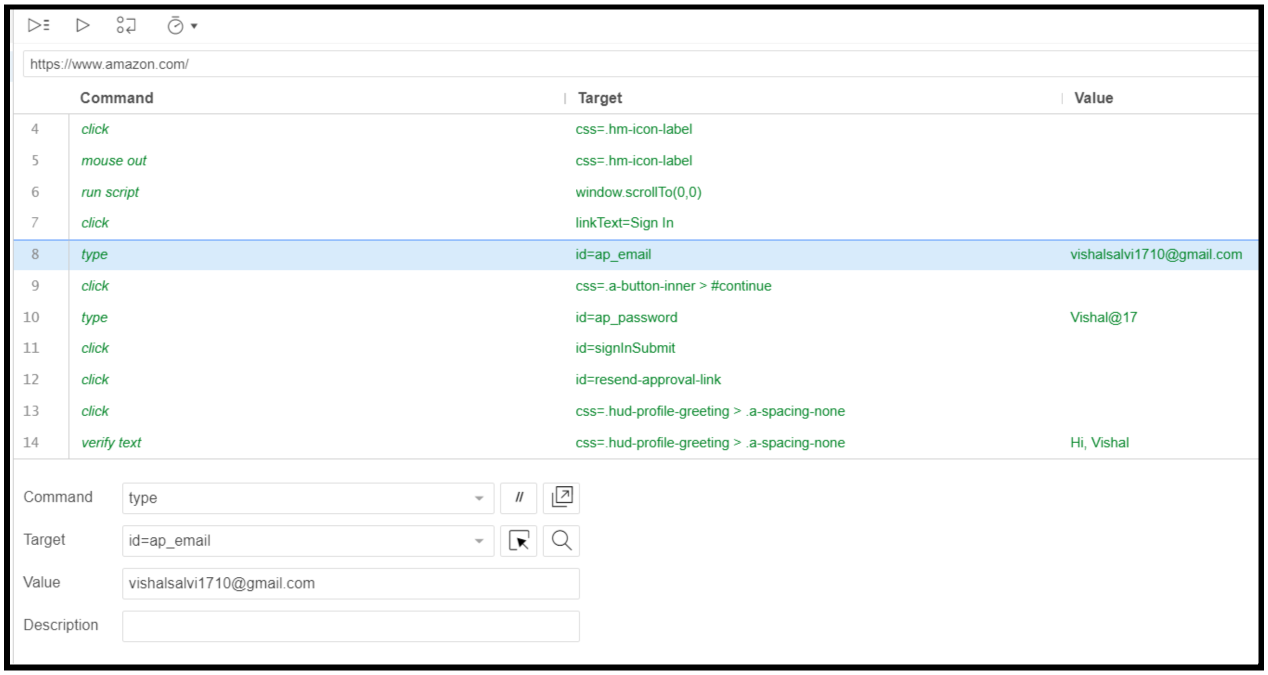
Command

Target

Value

Description

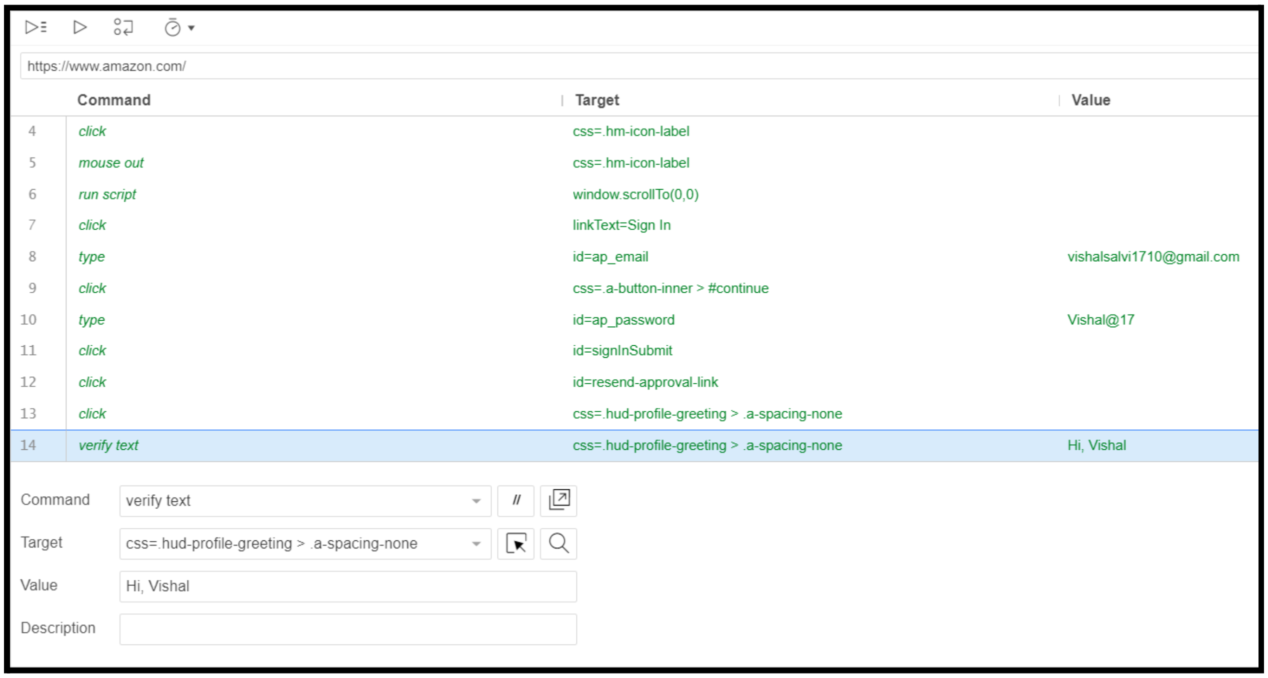
```
11. click on id=signInSubmit OK                                                        22:47:56
12. Trying to find id=resend-approval-link... OK                                       22:47:56
Warning Element found with secondary locator xpath=//a[contains(@href, '#')]. To use it 22:48:32
by default, update the test step to use it as the primary locator.
13. click on css=.hud-profile-greeting > .a-spacing-none OK                            22:48:32
14. verifyText on css=.hud-profile-greeting > .a-spacing-none with value Hi, Vishal OK  22:48:32
'Login' completed successfully                                                         22:48:33
```



https://www.amazon.com/

	Command	Target	Value
4	click	css=.hm-icon-label	
5	mouse out	css=.hm-icon-label	
6	run script	window.scrollTo(0,0)	
7	click	linkText=Sign In	
8	type	id=ap_email	vishalsalvi1710@gmail.com
9	click	css=.a-button-inner > #continue	
10	type	id=ap_password	Vishal@17
11	click	id=signInSubmit	
12	click	id=resend-approval-link	
13	click	css=.hud-profile-greeting > .a-spacing-none	
14	verify text	css=.hud-profile-greeting > .a-spacing-none	Hi, Vishal

Command	type
Target	id=ap_email
Value	vishalsalvi1710@gmail.com
Description	

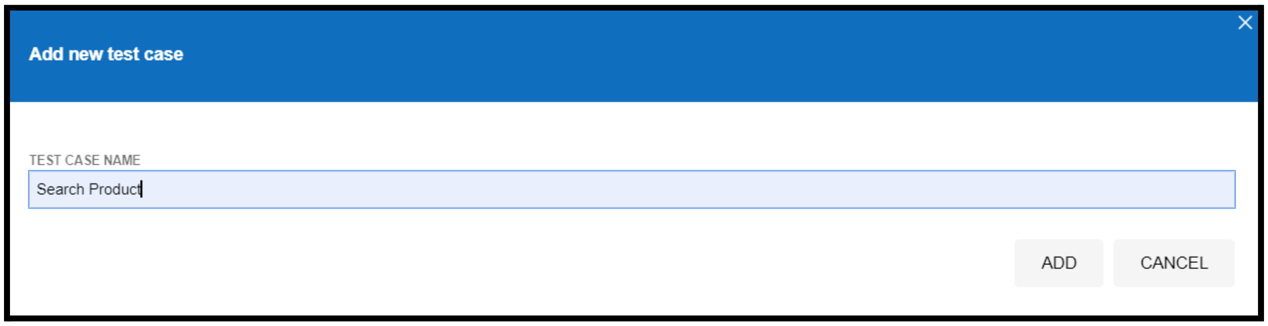


https://www.amazon.com/

	Command	Target	Value
4	click	css=.hm-icon-label	
5	mouse out	css=.hm-icon-label	
6	run script	window.scrollTo(0,0)	
7	click	linkText=Sign In	
8	type	id=ap_email	vishalsalvi1710@gmail.com
9	click	css=.a-button-inner > #continue	
10	type	id=ap_password	Vishal@17
11	click	id=signInSubmit	
12	click	id=resend-approval-link	
13	click	css=.hud-profile-greeting > .a-spacing-none	
14	verify text	css=.hud-profile-greeting > .a-spacing-none	Hi, Vishal

Command	verify text
Target	css=.hud-profile-greeting > .a-spacing-none
Value	Hi, Vishal
Description	

Add new test case: Search Product

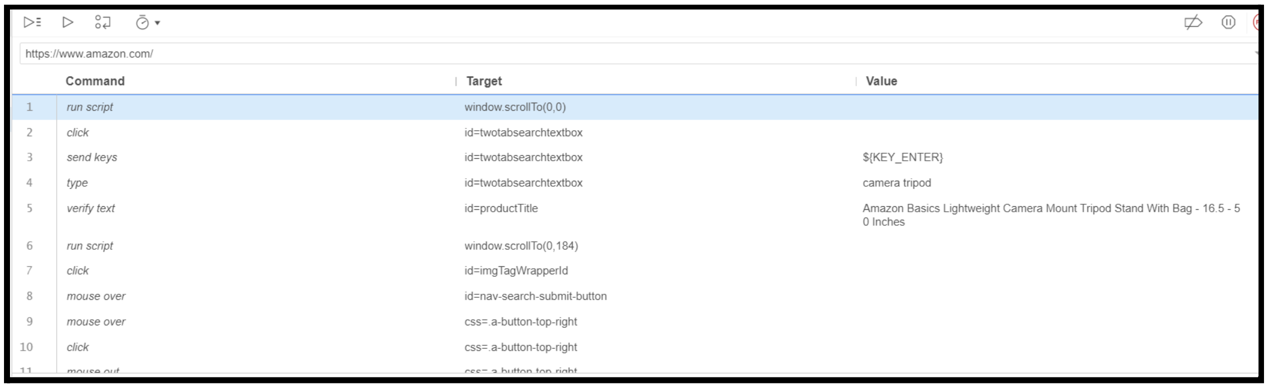


Add new test case

TEST CASE NAME

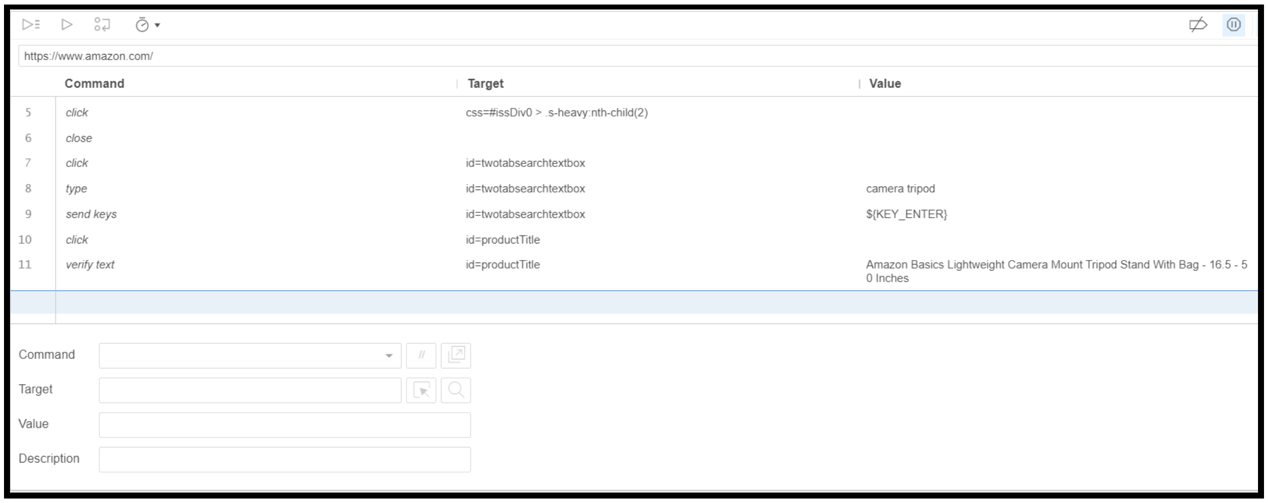
Search Product

ADD CANCEL



https://www.amazon.com/

	Command	Target	Value
1	run script	window.scrollTo(0,0)	
2	click	id=twotabsearchtextbox	
3	send keys	id=twotabsearchtextbox	${KEY_ENTER}
4	type	id=twotabsearchtextbox	camera tripod
5	verify text	id=productTitle	Amazon Basics Lightweight Camera Mount Tripod Stand With Bag - 16.5 - 50 Inches
6	run script	window.scrollTo(0,184)	
7	click	id=imgTagWrapperId	
8	mouse over	id=nav-search-submit-button	
9	mouse over	css=.a-button-top-right	
10	click	css=.a-button-top-right	
11	mouse out	css=.a-button-top-right	

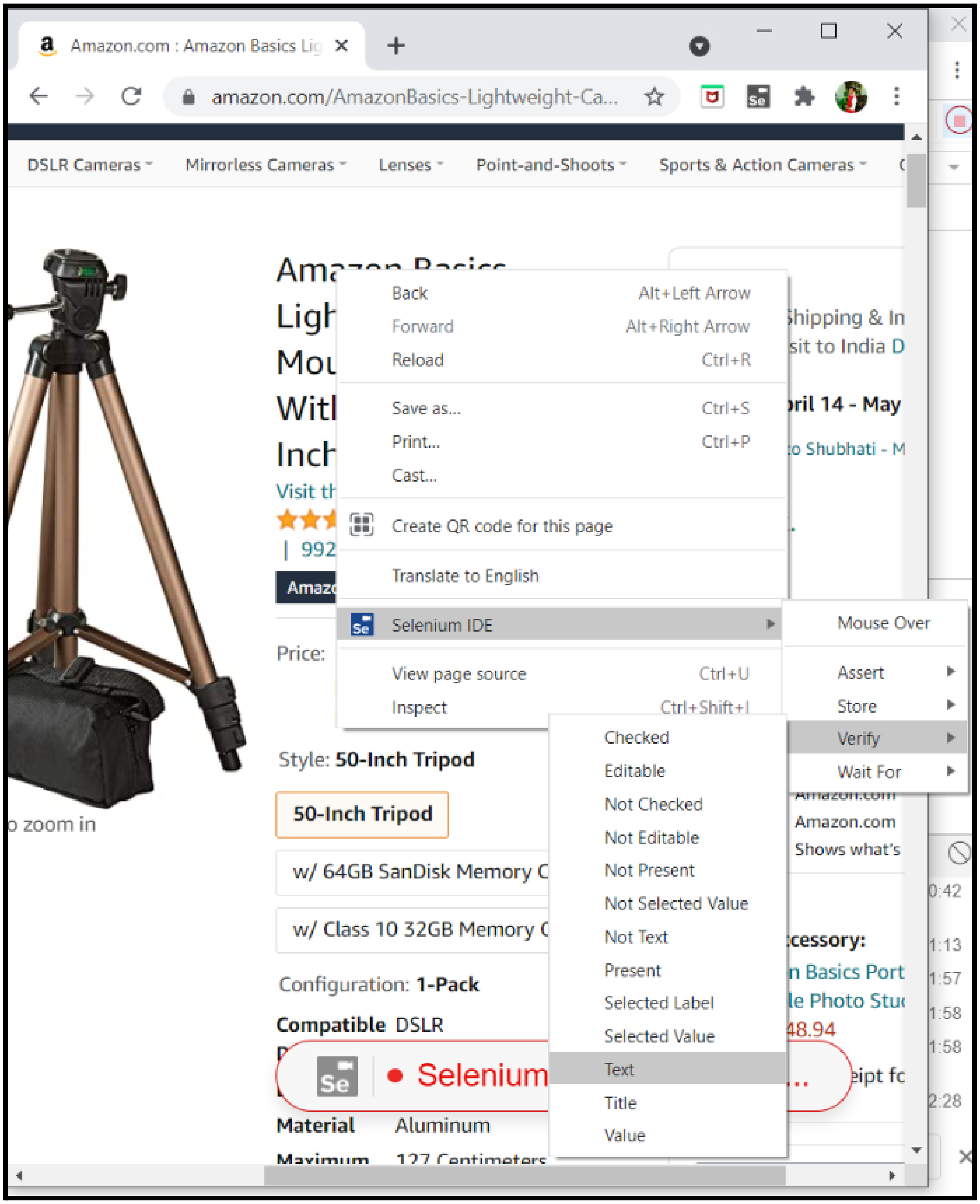


https://www.amazon.com/

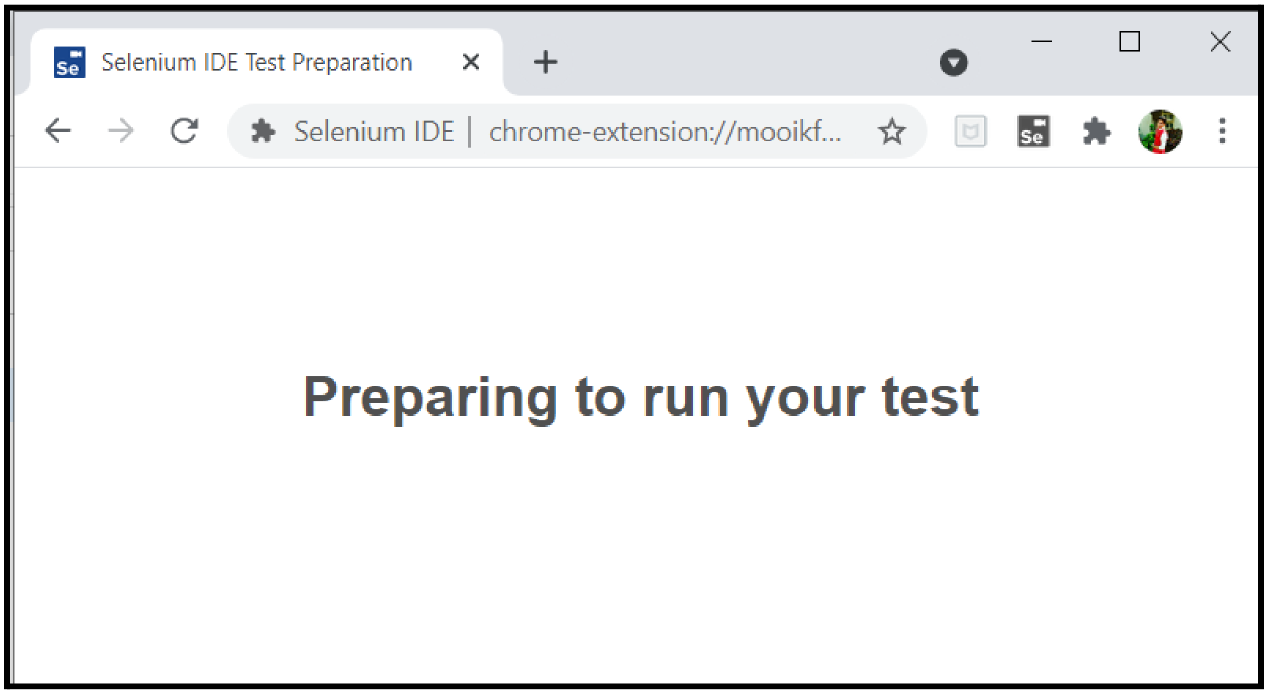
	Command	Target	Value
5	click	css=#issDiv0 > .s-heavy:nth-child(2)	
6	close		
7	click	id=twotabsearchtextbox	
8	type	id=twotabsearchtextbox	camera tripod
9	send keys	id=twotabsearchtextbox	${KEY_ENTER}
10	click	id=productTitle	
11	verify text	id=productTitle	Amazon Basics Lightweight Camera Mount Tripod Stand With Bag - 16.5 - 50 Inches

Command	
Target	
Value	
Description	

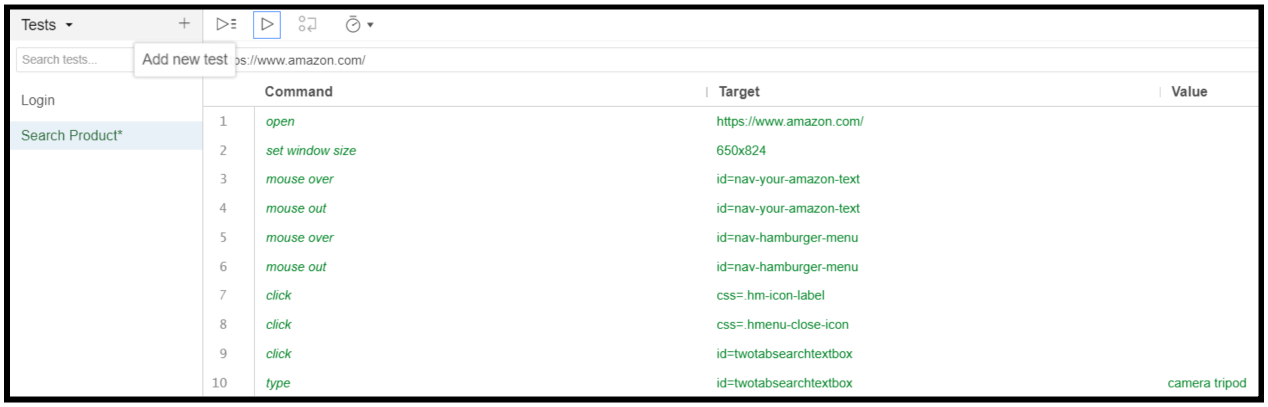
Verify text:


Back Alt+Left Arrow
Forward Alt+Right Arrow
Reload Ctrl+R
Save as... Ctrl+S
Print... Ctrl+P
Cast...
Create QR code for this page
Translate to English
Selenium IDE
View page source Ctrl+U
Inspect Ctrl+Shift+I
Mouse Over
Assert
Store
Verify
Wait For
Checked
Editable
Not Checked
Not Editable
Not Present
Not Selected Value
Not Text
Present
Selected Label
Selected Value
Text
Title
Value

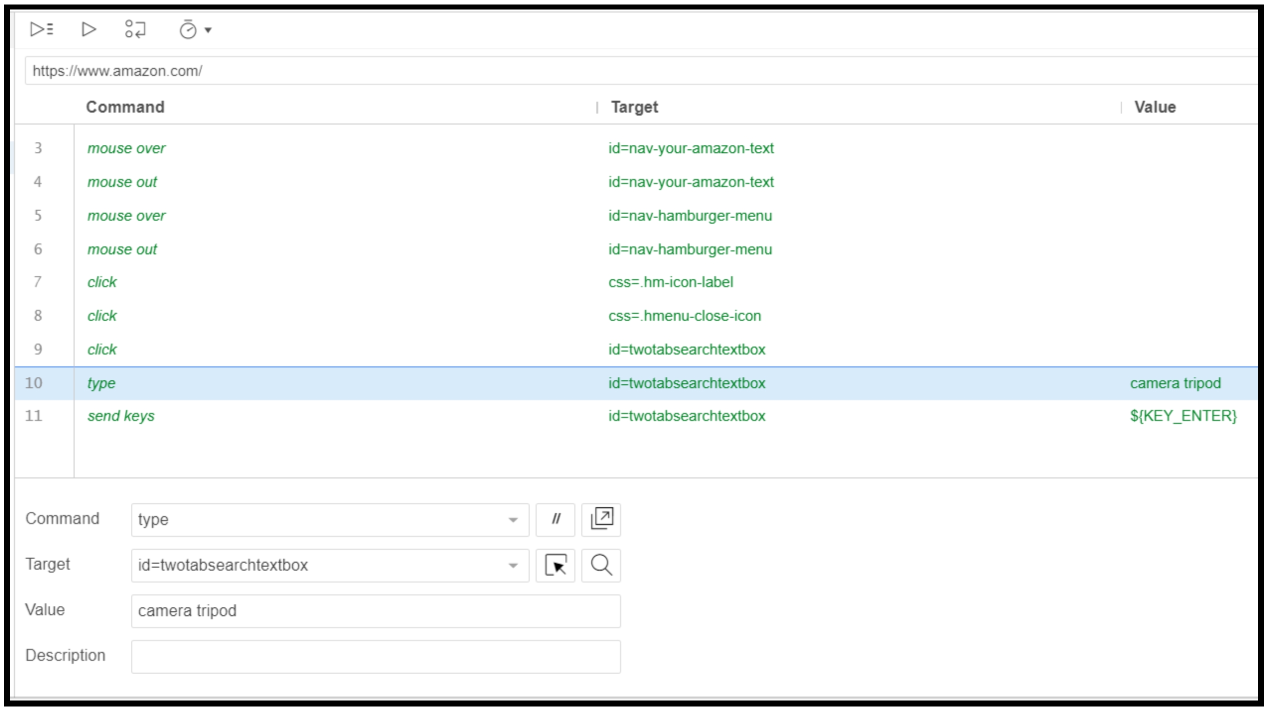
Running test case:


Selenium IDE Test Preparation
Selenium IDE | chrome-extension://mooikf...

Preparing to run your test



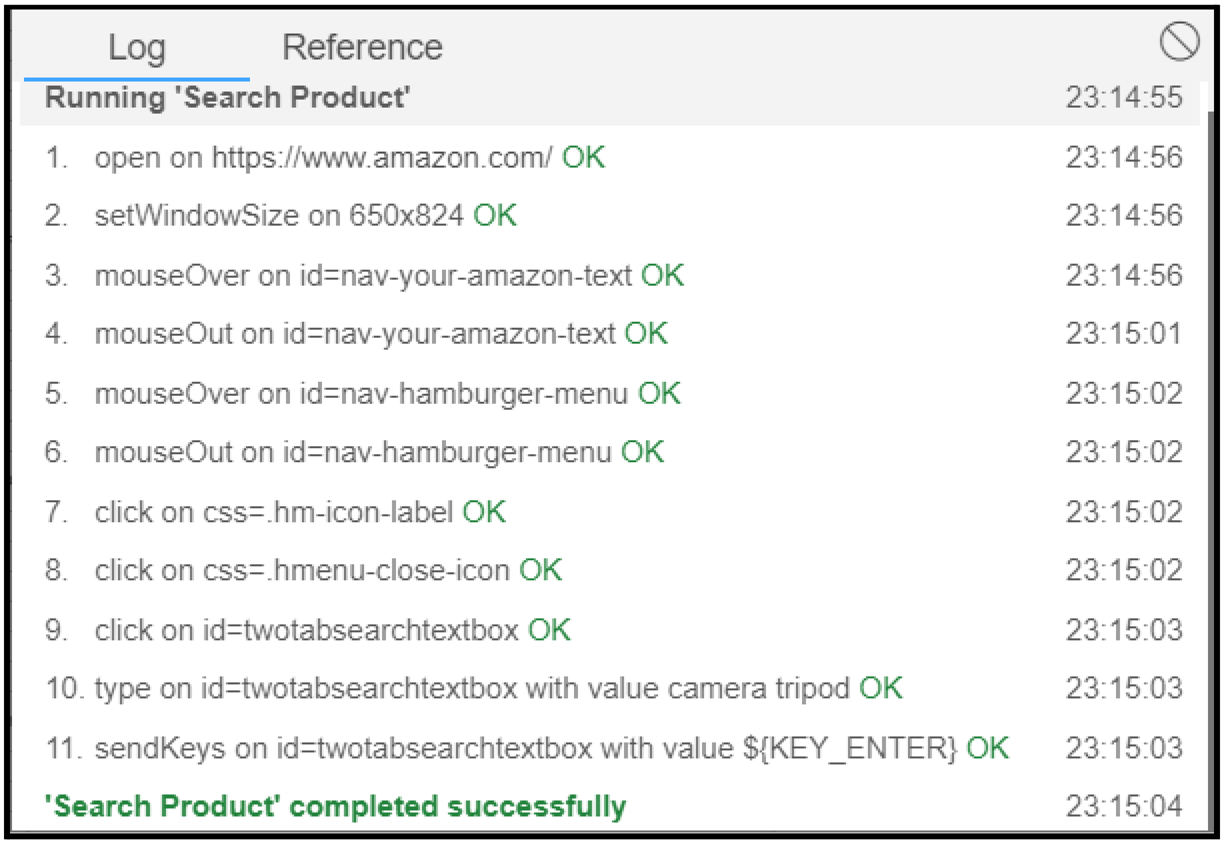
	Command	Target	Value
1	*open*	https://www.amazon.com/	
2	*set window size*	650x824	
3	*mouse over*	id=nav-your-amazon-text	
4	*mouse out*	id=nav-your-amazon-text	
5	*mouse over*	id=nav-hamburger-menu	
6	*mouse out*	id=nav-hamburger-menu	
7	*click*	css=.hm-icon-label	
8	*click*	css=.hmenu-close-icon	
9	*click*	id=twotabsearchtextbox	
10	*type*	id=twotabsearchtextbox	camera tripod



https://www.amazon.com/

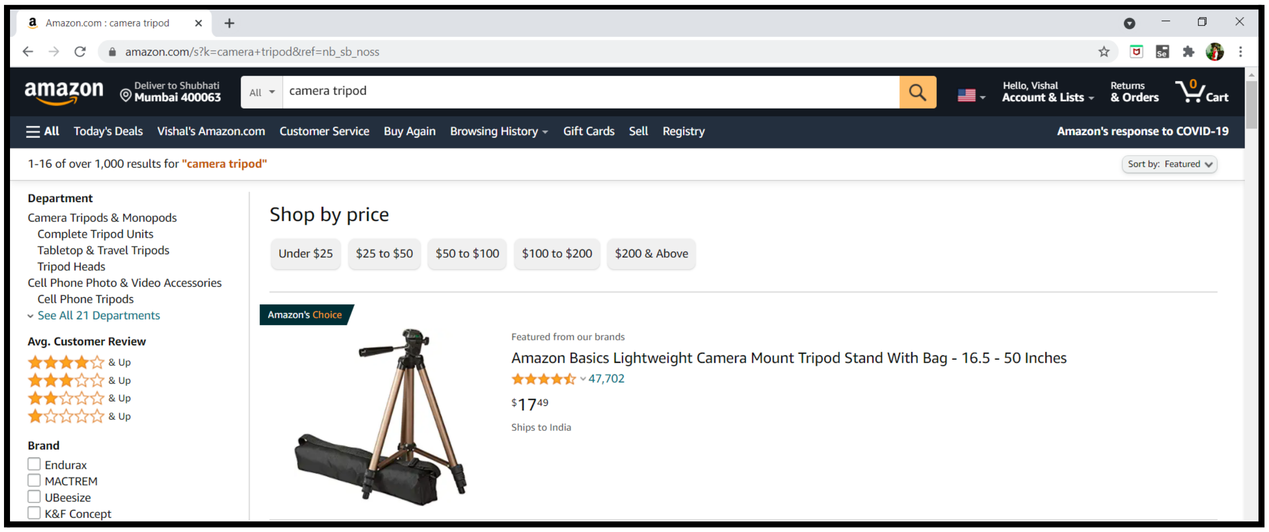
	Command	Target	Value
3	mouse over	id=nav-your-amazon-text	
4	mouse out	id=nav-your-amazon-text	
5	mouse over	id=nav-hamburger-menu	
6	mouse out	id=nav-hamburger-menu	
7	click	css=.hm-icon-label	
8	click	css=.hmenu-close-icon	
9	click	id=twotabsearchtextbox	
10	type	id=twotabsearchtextbox	camera tripod
11	send keys	id=twotabsearchtextbox	${KEY_ENTER}

Command	type
Target	id=twotabsearchtextbox
Value	camera tripod
Description	



```
Running 'Search Product'                                                    23:14:55
1.  open on https://www.amazon.com/ OK                                       23:14:56
2.  setWindowSize on 650x824 OK                                              23:14:56
3.  mouseOver on id=nav-your-amazon-text OK                                  23:14:56
4.  mouseOut on id=nav-your-amazon-text OK                                   23:15:01
5.  mouseOver on id=nav-hamburger-menu OK                                    23:15:02
6.  mouseOut on id=nav-hamburger-menu OK                                     23:15:02
7.  click on css=.hm-icon-label OK                                           23:15:02
8.  click on css=.hmenu-close-icon OK                                        23:15:02
9.  click on id=twotabsearchtextbox OK                                       23:15:03
10. type on id=twotabsearchtextbox with value camera tripod OK               23:15:03
11. sendKeys on id=twotabsearchtextbox with value ${KEY_ENTER} OK            23:15:03
'Search Product' completed successfully                                      23:15:04
```

Result of test case: Search Product



amazon.com/s?k=camera+tripod&ref=nb_sb_noss

1-16 of over 1,000 results for "camera tripod"

Shop by price

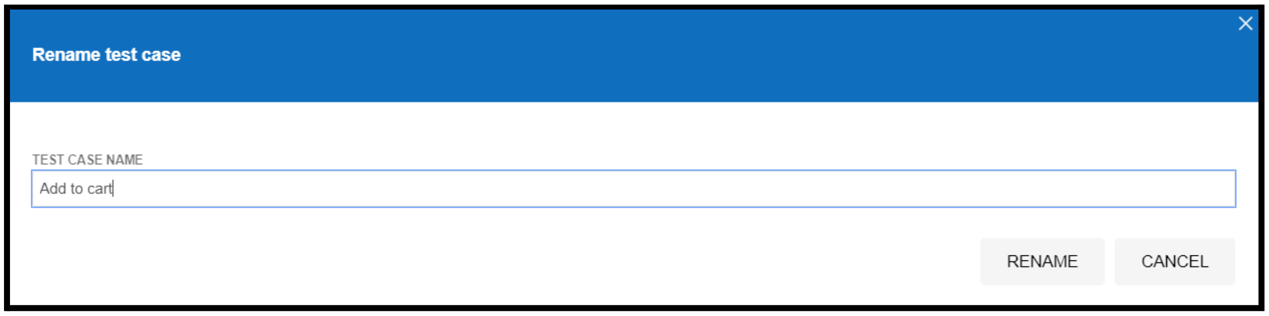
Amazon Basics Lightweight Camera Mount Tripod Stand With Bag - 16.5 - 50 Inches

47,702

$17.49

Ships to India

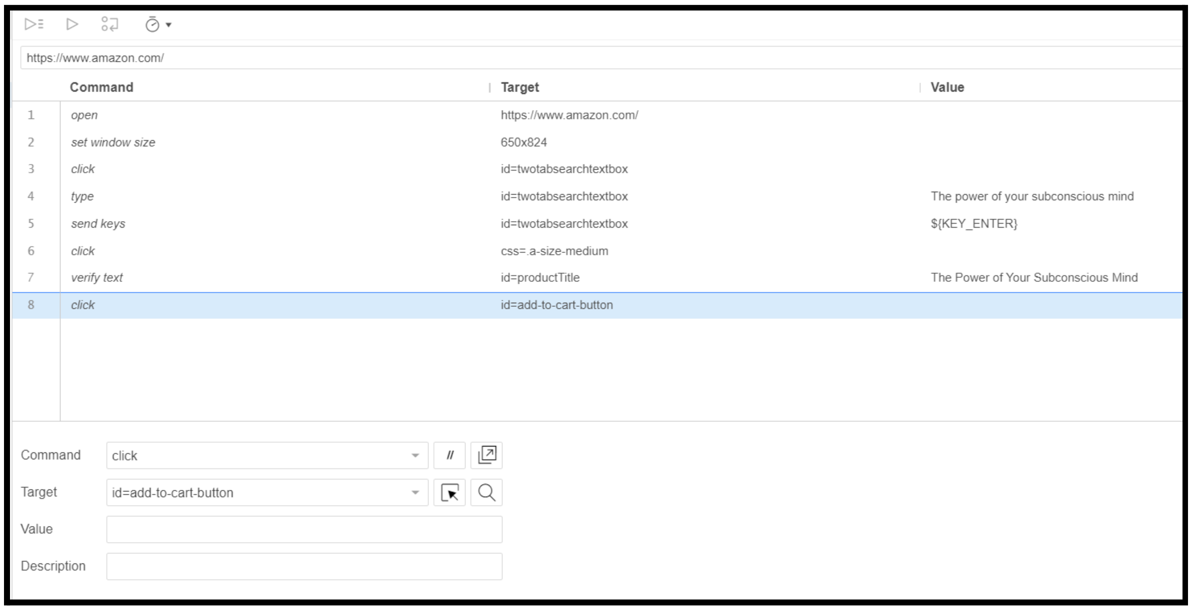
Add New Test Case: Add to cart



Rename test case

TEST CASE NAME

Add to cart

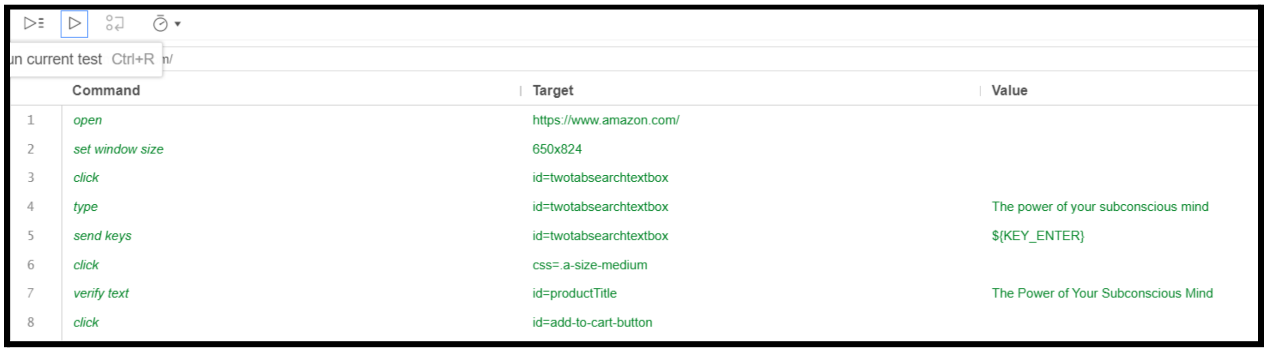


https://www.amazon.com/

	Command	Target	Value
1	open	https://www.amazon.com/	
2	set window size	650x824	
3	click	id=twotabsearchtextbox	
4	type	id=twotabsearchtextbox	The power of your subconscious mind
5	send keys	id=twotabsearchtextbox	${KEY_ENTER}
6	click	css=.a-size-medium	
7	verify text	id=productTitle	The Power of Your Subconscious Mind
8	click	id=add-to-cart-button	

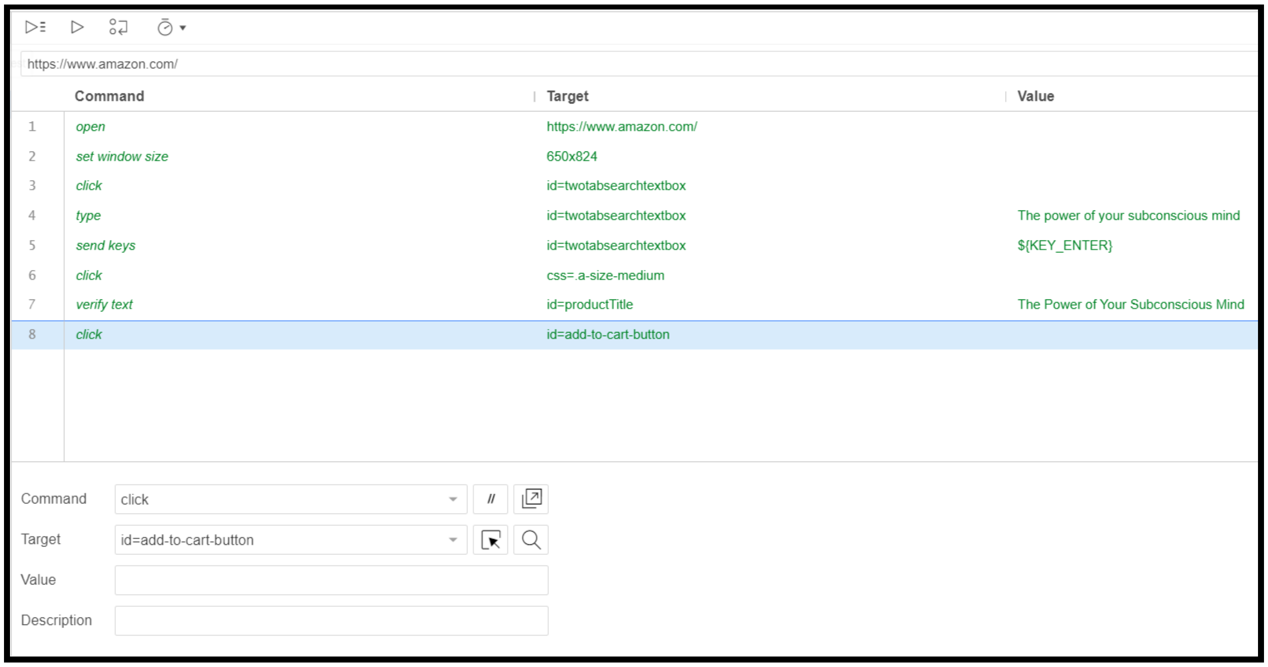
Command	click
Target	id=add-to-cart-button
Value	
Description	

Run test case:



un current test Ctrl+R n/

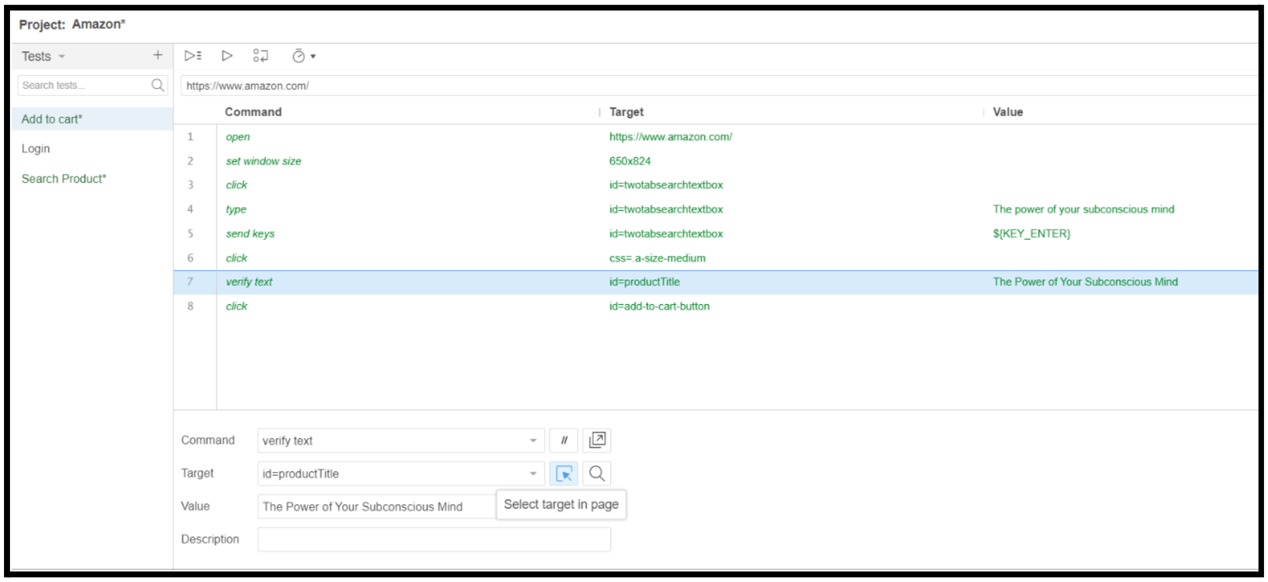
	Command	Target	Value
1	open	https://www.amazon.com/	
2	set window size	650x824	
3	click	id=twotabsearchtextbox	
4	type	id=twotabsearchtextbox	The power of your subconscious mind
5	send keys	id=twotabsearchtextbox	${KEY_ENTER}
6	click	css=.a-size-medium	
7	verify text	id=productTitle	The Power of Your Subconscious Mind
8	click	id=add-to-cart-button	



https://www.amazon.com/

	Command	Target	Value
1	open	https://www.amazon.com/	
2	set window size	650x824	
3	click	id=twotabsearchtextbox	
4	type	id=twotabsearchtextbox	The power of your subconscious mind
5	send keys	id=twotabsearchtextbox	${KEY_ENTER}
6	click	css=.a-size-medium	
7	verify text	id=productTitle	The Power of Your Subconscious Mind
8	click	id=add-to-cart-button	

Command	click
Target	id=add-to-cart-button
Value	
Description	



Project: Amazon*

Tests

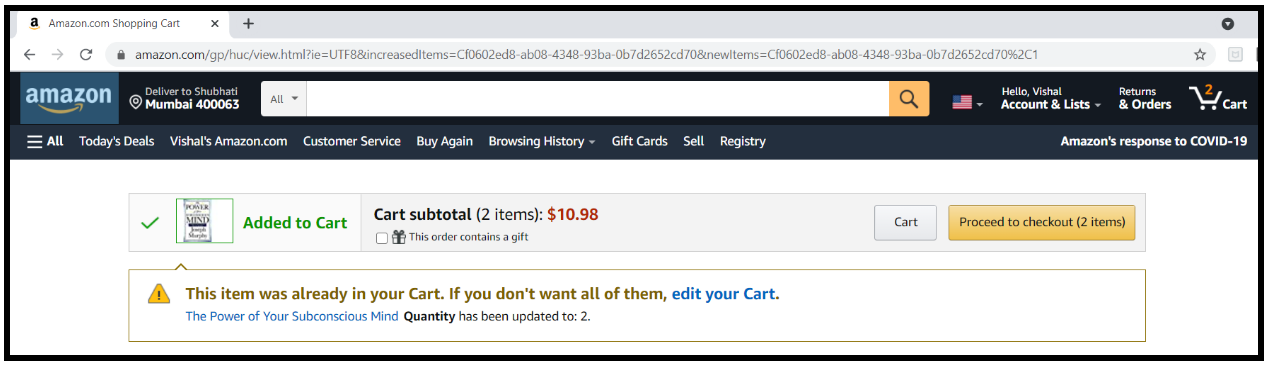
Add to cart*
Login
Search Product*

https://www.amazon.com/

	Command	Target	Value
1	open	https://www.amazon.com/	
2	set window size	650x824	
3	click	id=twotabsearchtextbox	
4	type	id=twotabsearchtextbox	The power of your subconscious mind
5	send keys	id=twotabsearchtextbox	${KEY_ENTER}
6	click	css=.a-size-medium	
7	verify text	id=productTitle	The Power of Your Subconscious Mind
8	click	id=add-to-cart-button	

Command	verify text
Target	id=productTitle
Value	The Power of Your Subconscious Mind
Description	

Result of test case: Add to cart


Amazon.com Shopping Cart
amazon.com/gp/huc/view.html?ie=UTF8&increasedItems=Cf0602ed8-ab08-4348-93ba-0b7d2652cd70&newItems=Cf0602ed8-ab08-4348-93ba-0b7d2652cd70%2C1
Deliver to Shubhati Mumbai 400063
Hello, Vishal Account & Lists
Returns & Orders
Cart
All Today's Deals Vishal's Amazon.com Customer Service Buy Again Browsing History Gift Cards Sell Registry
Amazon's response to COVID-19

Added to Cart

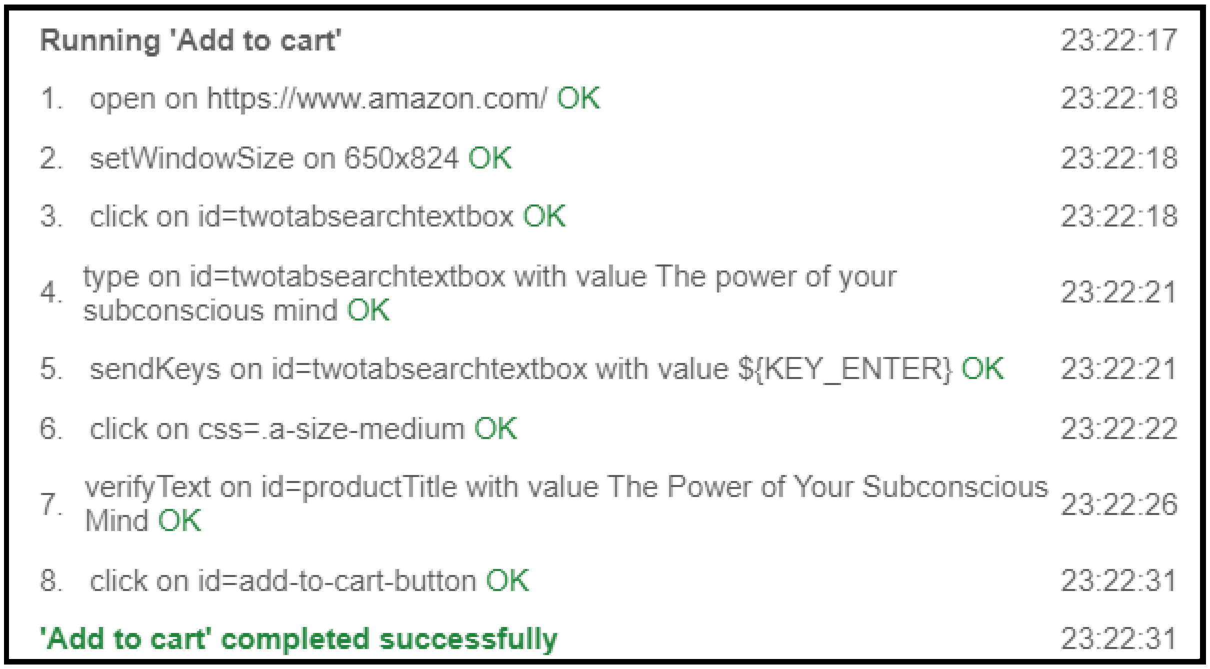
Cart subtotal (2 items): $10.98

This order contains a gift

Cart   Proceed to checkout (2 items)

This item was already in your Cart. If you don't want all of them, edit your Cart.

The Power of Your Subconscious Mind **Quantity** has been updated to: 2.



```
Running 'Add to cart'                                                              23:22:17
1. open on https://www.amazon.com/ OK                                              23:22:18
2. setWindowSize on 650x824 OK                                                     23:22:18
3. click on id=twotabsearchtextbox OK                                              23:22:18
4. type on id=twotabsearchtextbox with value The power of your
   subconscious mind OK                                                            23:22:21
5. sendKeys on id=twotabsearchtextbox with value ${KEY_ENTER} OK                   23:22:21
6. click on css=.a-size-medium OK                                                  23:22:22
7. verifyText on id=productTitle with value The Power of Your Subconscious
   Mind OK                                                                         23:22:26
8. click on id=add-to-cart-button OK                                               23:22:31
'Add to cart' completed successfully                                               23:22:31
```

Conclusion:

In this experiment, we understood the concept of automation testing in a project. We implemented automated testing for our project amazon.com.Selenium IDE does have some limitations as a testing framework, but is a valuable tool in its own right and deserves a place in the toolbox of any testing team.

References:

1)https://www.edureka.co/blog/what-is-selenium/