Task 15 – Automation Testing

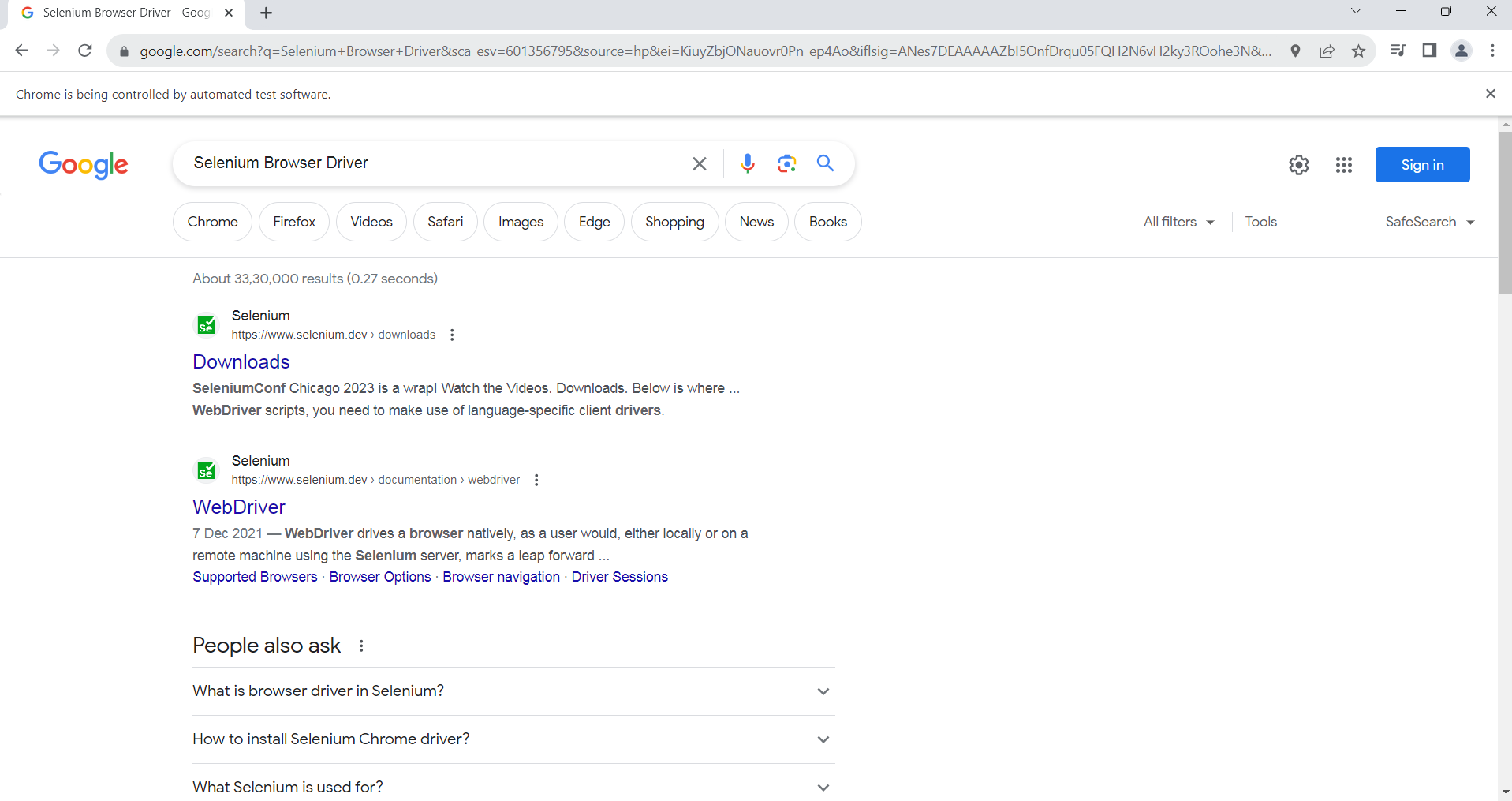
Q1. Selenium IDE, Selenium WebDriver and Selenium Grid:

1. ***Selenium IDE:***
   * 1. Selenium IDE (Integrated Development Environment) is an extension we add to a web browser
     2. It is used to test applications without writing any test scripts
     3. It allows us to record and playback the scripts
     4. It works only with FireFox
     5. It gives result as Selenius commands which can be converted to many programming languages
2. ***Selenium Remote Control (RC):***
   * 1. It’s the very first tool introduced in Selenium
     2. It allows us to write test scripts in multiple languages
     3. It relies on JavaScript for automation
     4. It supports all browsers
     5. It interacts with the Server first then interacts with the browser
3. ***Selenium WebDriver:***
   * 1. Most common and widely used Selenium tool
     2. It allows us to write test scripts in multiple languages
     3. It doesn’t require any remote server
     4. We can directly interact with the browser using the webdriver interface
4. ***Selenium Grid:***
   * 1. Selenium Grid is a tool used together with Selenium RC to run tests on different machines against different browsers in parallel.
     2. It is used for parallel testing or Distributive testing where test scrips are executed in different machines parallelly. It allows to run test scripts on different machines at same time
     3. It uses Hubs and Nodes. Nodes are nothing but machines where test scripts and executed and Hubs controls the execution on various machines.
     4. Eg : Hug (with 150 Test Scripts) 🡪 Divides it and distributes to multiple nodes 🡪 Node 1 executes 100 scripts🡪Node 2 executes 50 scripts. Both executions happen parallelly and hub controls the execution

Q2. Selenium Script in Java:



Adding dependencies required for Selenium and Web Driver Manager



First Browser is opened, it’s maximized, text is entered in search box and searched and results

Q3. Selenium and it’s uses in Automation Testing:

1. Selenium is a open-source automation testing tool that is used to test web applications on browsers
2. It automates browsers and only web applications can be tested
3. Created by Jason Huggins in 2004
4. It is very popular, commonly used open-source testing tool
5. It supports most programming languages such as Java, Python, C#, Ruby etc.,
6. Selenium have a large community support which is swift and provides immediate support for any issues with selenium
7. Selenium provides cross browser compatibility. It can be used in any browser such as Chrome, Firefox, Edge, Safari etc.,
8. Selenium provides cross platform compatibility as well. It can be used in any operating system such as Windows, Mac, Linux etc.,
9. It provides integration with other platforms and frameworks such as TestNG, Apache POI, Junit, ExtentReport Etc.,

Q4. Browser Drivers used in Selenium:

1. ChromeDriver : Driver used for Google Chrome
2. FireFoxDriver or GeckoDriver : Driver used for Mozila FireFox
3. EdgeDriver : Driver used for Microsoft Edge
4. InternetExplorerDriver : Driver used for Internet Explorer
5. SafariDriver : Used for Safari browser
6. OperaDriver : Used for Opera Browser

Q5. Steps to create simple web driver script:

1. Add all required dependencies to the Maven project (Selenium, WebDriverManager)
2. Create a package and a class with main() method
3. Setup the required browser driver by initializing the Driver class
4. Use (drivername).get() method to pass the address of the website that needs to be opened
5. Use manage().window().maximize() method to maximize the browser window
6. Find the name or ID of the web element that is required using the inspect option
7. Use fineElement() and sendKeys() functions to enter the value and to do the “enter” function
8. Use (drivername).close() method to close the browser

Code :

**import** org.openqa.selenium.By;

**import** org.openqa.selenium.Keys; // importing the selenium and webdriver packages

**import** org.openqa.selenium.chrome.ChromeDriver;

**public** **class** BrowserLaunch {

**public** **static** **void** main(String[] args) {

// To open Chrome browser and launch the google url and search for something

ChromeDriver driver = **new** ChromeDriver(); // Setup the driver and launch the browser

driver.get("https://www.google.com/"); // Load the url

driver.manage().window().maximize(); // Maximizing the window

driver.findElement(By.*name*("q")).sendKeys("Selenium Browser Driver", Keys.***ENTER***); // It finds the search bar using the name, enter the value provided and click Search button

//driver.close();

}

}