**A.Write a blog on below questions.**

**1. Explain the difference between Selenium IDE, Selenium WebDriver, and Selenium Grid**

**Selenium IDE:** Selenium Integrated Development Environment (IDE) is a record and playback tool for creating simple test cases in browsers like Chrome and Firefox. It's primarily used for beginners and those who want to quickly generate test cases without writing code.

**Selenium WebDriver:** Selenium WebDriver is the core component of Selenium, offering a programming interface to interact with web browsers programmatically. It supports various programming languages like Java, Python, C#, etc., making it suitable for creating complex and dynamic test scripts.

**Selenium Grid:** Selenium Grid is a tool used for parallel testing. It allows you to execute test cases on multiple machines and browsers simultaneously, saving time in large-scale testing scenarios.

**2. Write a Selenium script in Java to open Google and search for "Selenium Browser Driver."**

**package seldemo;**

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

**import org.openqa.selenium.Keys;**

**import org.openqa.selenium.WebDriver;**

**import org.openqa.selenium.firefox.FirefoxDriver;**

**import io.github.bonigarcia.wdm.WebDriverManager;**

**public class SeleniumScript {**

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

**WebDriverManager.*firefoxdriver*().setup();**

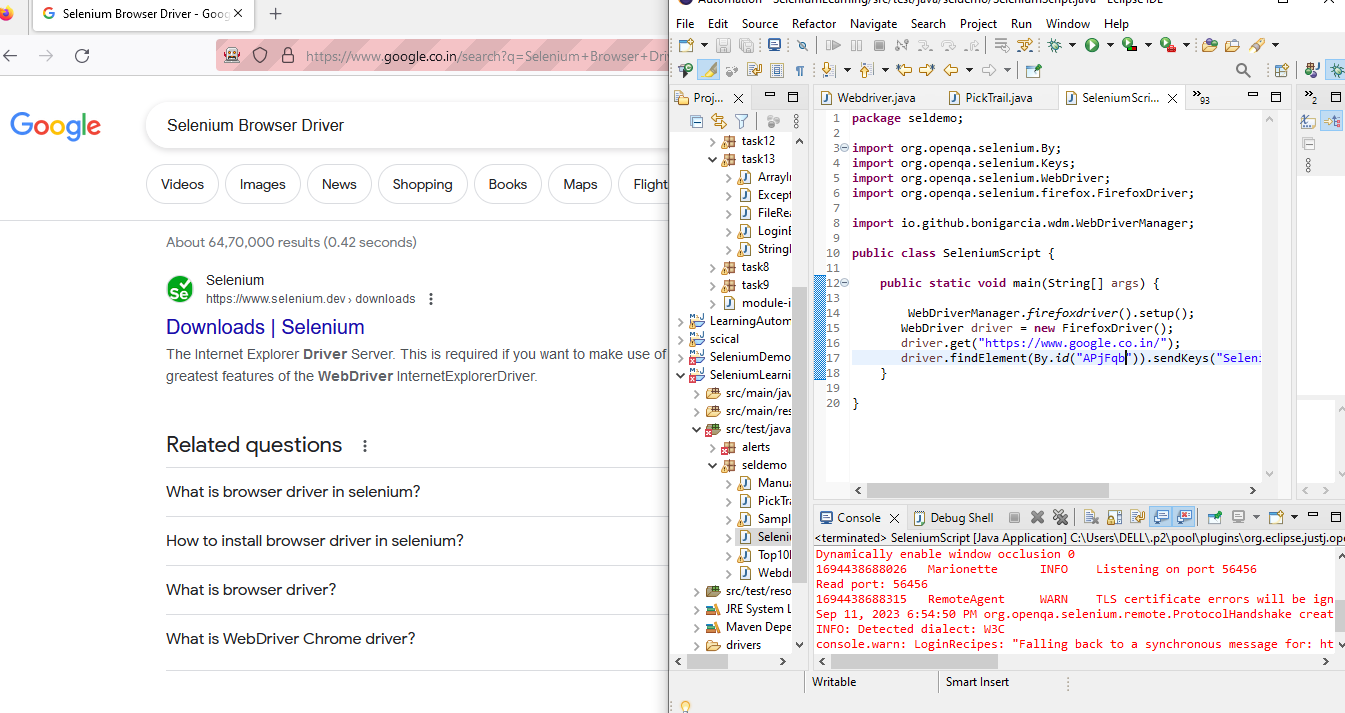
**WebDriver driver = new FirefoxDriver();**

**driver.get("https://www.google.co.in/");**

**driver.findElement(By.*id*("APjFqb")).sendKeys("Selenium Browser Driver"+ Keys.*ENTER*);**

**}**

**}**

****

**3. What is Selenium? How is it useful in Automation Testing?**

Selenium is an open-source framework for automating web browsers. It allows testers and developers to automate repetitive tasks, validate web applications' functionality, and perform regression testing. Selenium is particularly useful in automation testing for the following reasons:

* Cross-Browser Testing: Selenium supports multiple browsers like Chrome, Firefox, Safari, and Edge, enabling testers to ensure compatibility across different platforms.
* Reusability: Test scripts written in Selenium are reusable, saving time and effort when testing similar functionalities across multiple test cases.
* Parallel Testing: Selenium Grid enables parallel execution of test cases on various machines and browsers, reducing test execution time.
* Integration: Selenium can be integrated with various programming languages and test frameworks, making it versatile and adaptable to different testing environments.
* What are all Browser drivers used in Selenium?

**4. What are all Browser drivers used in Selenium?**

Selenium WebDriver supports various browser drivers, including:

* ChromeDriver: For Google Chrome
* GeckoDriver: For Mozilla Firefox
* EdgeDriver: For Microsoft Edge
* SafariDriver: For Apple Safari
* OperaDriver: For Opera Browser

These drivers allow Selenium scripts to interact with and control specific browsers.

**5. What Are The Steps To Create A Simple Web Driver Script?Explain with code.**

Here are the steps to create a simple WebDriver script:

Step 1: Set up your development environment by downloading the required WebDriver executable (e.g., ChromeDriver) and configuring your project.

Step 2: Create a new Java class for your script (e.g., GoogleSearch.java).

Step 3: Write the script using WebDriver commands to automate browser interactions.

Step 4: Run the script to execute the automation.

I've already provided an example script in question 2 that opens Google and searches for "Selenium Browser Driver."

**package** seldemo;

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

**import** org.openqa.selenium.Keys;

**import** org.openqa.selenium.WebDriver;

**import** org.openqa.selenium.firefox.FirefoxDriver;

**import** io.github.bonigarcia.wdm.WebDriverManager;

**public** **class** SeleniumScript {

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

WebDriverManager.*firefoxdriver*().setup();

WebDriver driver = **new** FirefoxDriver();

driver.get("https://www.google.co.in/");

driver.findElement(By.*id*("APjFqb")).sendKeys("Selenium Browser Driver"+ Keys.***ENTER***);

}

}