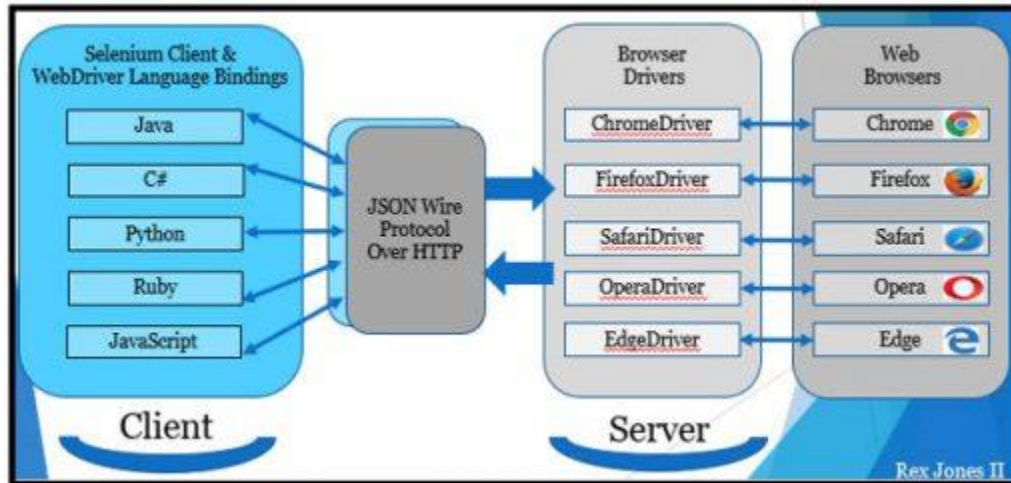# Selenium WebDriver 4

W3C WebDriver Protocol

JSON Wire Protocol Over HTTP has been removed from W3C WebDriver Protocol. That means information is not transferred over HTTP by sending HTTP Requests and receiving HTTP Responses. With Selenium 4, information is transferred directly back and forth from the client to the server without the JSON Wire Protocol.
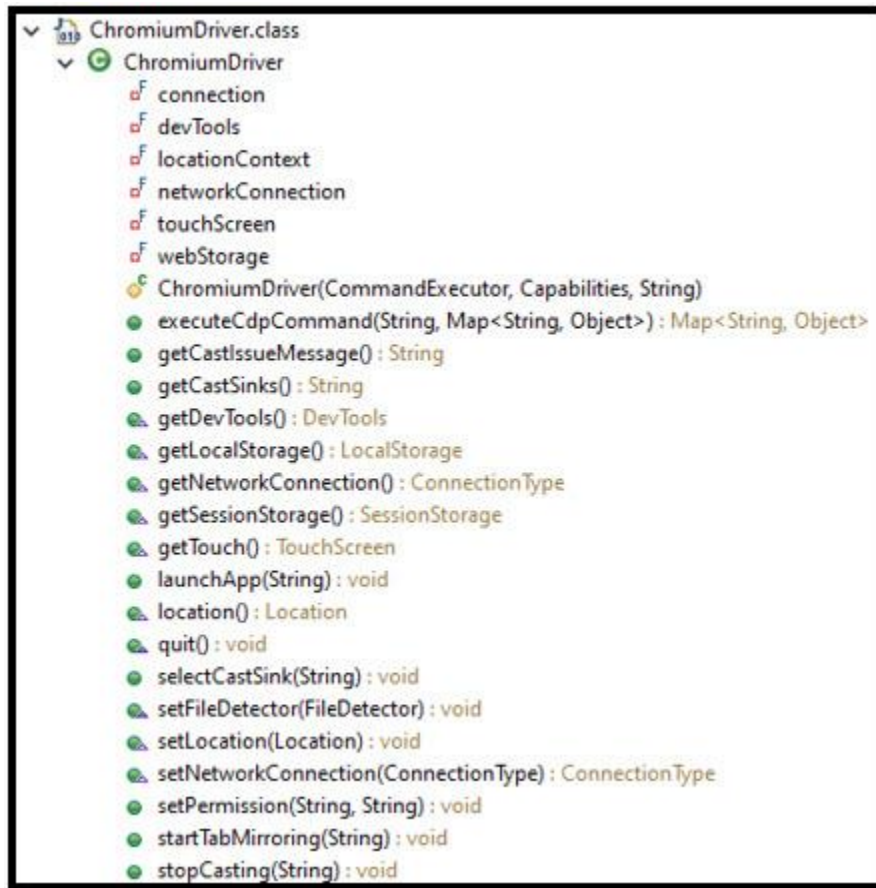
An advantage involves testing applications that will execute more consistently between browsers. Kudos to W3C, an acronym for World Wide Web Consortium for developing web standards. A standardization of W3C promotes compatibility beyond WebDriver API implementations.

ChromiumDriver

Selenium WebDriver uses a driver to manage each browser. ChromeDriver is an executable that Selenium WebDriver uses to control Google Chrome and EdgeDriver controls Microsoft Edge. Both drivers extend ChromiumDriver as an update in Selenium 4. The ChromiumDriver class has methods to create a connection with DevTools.

```
public class ChromeDriver extends ChromiumDriver

public class EdgeDriver extends ChromiumDriver
```

ChromiumDriver.class
  ChromiumDriver
    connection
    devTools
    locationContext
    networkConnection
    touchScreen
    webStorage
    ChromiumDriver(CommandExecutor, Capabilities, String)
    executeCdpCommand(String, Map<String, Object>) : Map<String, Object>
    getCastIssueMessage() : String
    getCastSinks() : String
    getDevTools() : DevTools
    getLocalStorage() : LocalStorage
    getNetworkConnection() : ConnectionType
    getSessionStorage() : SessionStorage
    getTouch() : TouchScreen
    launchApp(String) : void
    location() : Location
    quit() : void
    selectCastSink(String) : void
    setFileDetector(FileDetector) : void
    setLocation(Location) : void
    setNetworkConnection(ConnectionType) : ConnectionType
    setPermission(String, String) : void
    startTabMirroring(String) : void
    stopCasting(String) : void

Selenium IDE: New and refreshed Selenium IDE

- Selenium IDE is a tool for record and playback. Additionally, it will have advanced and rich features in Selenium 4.0.

- Moreover, the limitation of a lack of ability for parallel execution will overcome in the new version of IDE. In other words, a new version is coming with improved browser support. Additionally, parallel execution will be possible along with the passed/failed status of test cases and execution time.

- New IDE is W3C compatible and completely dependent on WebDriver.

- Moreover, the new version will have a new Node JSbased CLI runner.

Selenium 4 Relative Locator – Methods

above	Web element to be searched/located appears above the specified element
below	Web element to be searched/located appears below the specified element.
toLeftOf	Web element located appears to the left of the specified element
toRightOf	Web element located appears to the right of the specified element
near	Web element located is at most 50 pixels away from the specified element

```
@Test
public void test_login_using_relative_locators_2(){
    String txt_name = driver.findElement(withTagName("input")
            .toLeftOf(By.id("addbutton"))
            .|
        .b
        .g
    assertEqua
    driver.man                                              CONDS);
```

- ● above(By locator) : RelativeBy - RelativeBy
- ● above(WebElement element) : RelativeBy - RelativeBy
- ● below(By locator) : RelativeBy - RelativeBy
- ● below(WebElement element) : RelativeBy - RelativeBy
- ● near(By locator) : RelativeBy - RelativeBy
- ● near(WebElement element) : RelativeBy - RelativeBy
- ● near(By locator, int atMostDistanceInPixels) : RelativeBy - Relativ
- ● near(WebElement element, int atMostDistanceInPixels) : Relative
- ● toLeftOf(By locator) : RelativeBy - RelativeBy
- ● toLeftOf(WebElement element) : RelativeBy - RelativeBy
- ● toRightOf(By locator) : RelativeBy - RelativeBy

```
ems   @ Javadoc
ted> RelativeLocators [Te                                      17:33 AM)

===============
t suite
```