

# ReleaseNotes

Wednesday, February 6, 2019 10:53 AM

## 2.0 - Current Release

- IWebDriverManager
  - o Updates to method summaries to align with the IWebDriver interface
  - o Updates to align with the IJavaScriptExecutor interface
    - Added Methods
      - object ExecuteScript(string script, params object[] args)
      - object ExecuteAsyncScript(string script, params object[] args)
    - Properties Added
      - IWebDriver WrappedDriver
        - ◆ This property returns the active IWebDriver object
        - ◆ This should not be utilized for testing and is provided for troubleshooting any issues found while using this package
- WebUiAutomation
  - o Methods removed
    - IJavaScriptExecutor GetJavaScriptExecutor(IWebDriverManager webDriverManager);
      - Replaced by the ExecuteScript and ExecuteScriptAsync methods on the IWebDriverManager Interface
  - o Methods moved to IWebDriverManager
    - All the methods listed below will now need to be called from the IWebDriverManager and their signatures have been updated to exclude the IWebDriverManager parameter.
    - Additionally, any method that was returning the IWebDriverManager object has been updated to void.
      - CheckChildElementExistsAndReturnIt
      - CheckElementExistsReturnCssSelector
      - CheckElementExistsReturnIWebElement
      - CheckElementsExist
      - ClearText
      - Click
      - DoesUrlContain
      - DoesUrlContainUsingRegex
      - GetAllElementsUsingMatchingSelectorData
      - HighlightElement
      - IsElementDisplayed
      - IsElementEnabled
      - IsElementDisplayedAndEnabled
      - JavaScriptClick
      - MoveToElement
      - SendText
      - SendTextWithDelay
      - TakeScreenShot
- Bug Fixes
  - o JavaScriptClick should return the IWebDriverManager object
    - Issue resolved by moving the method onto the IWebDriverManager object now there is no need to return it
  - o Failure info being logged after the closing line in the log
    - Moved the call to log failure info to the correct position

### 1.1.1

- Update Chrome Driver from 76 to 77
- Update Firefox Driver from 24 to 26

- Update IE Driver from 3.150.0 to 3.150.1

## 1.1.0

- Deprecated Methods:
  - o `IWebElement.FindElementWithActiveDriver(SelectorData selectorData)` - meant to assist in troubleshooting issues during the alpha phase
- Added Methods:
  - o `IWebDriverManager`
    - `void SwitchToFrame(SelectorData selectorData)`
      - Switches the driver context to the frame matching the SelectorData provided
      - selectorData -> Object representing the frame to check
  - o `ILogger`
    - `string GetCurrentLogFileName`
      - Returns the name of the failure log file currently in use
    - `string GetCurrentFailureLogFileName`
      - Returns the name of the failure log file currently in use

## 1.0.11-alpha

- Deprecated `HtmlTree` and `HtmlTag` as they were not useful in their current state
- Added `IDomTree` and `IDomNode`(Accessible methods outlined in the general documentation)
- Deprecated the below method:
  - o **`List<By> GetAllBysUsingMatchingSelectorData(SelectorData selectorData, IWebDriverManager webDriverManager)`**
    - It had no practical usage in it's current state as the goal of this tooling was to remove the need it interact with By typed objects
- Added the below method(s)
  - o **`bool IsElementDisplayed(IWebDriverManager webDriverManager, SelectorData selectorData)`**
    - Checks whether or not the provided element is visible(This requires `IWebElement.Displayed` evaluate to true)
    - selectorData -> Object representing the element to check
  - o **`bool IsElementEnabled(IWebDriverManager webDriverManager, SelectorData selectorData)`**
    - Checks whether or not the provided element is enabled(This requires `IWebElement.Enabled` evaluate to true)
    - selectorData -> Object representing the element to check
  - o **`bool IsElementDisplayedAndEnabled(IWebDriverManager webDriverManager, SelectorData selectorData)`**
    - Checks whether or not the provided element is visible(This requires `IWebElement.Displayed` and `IWebElement.Enabled` to both evaluate to true)
    - selectorData -> Object representing the element to check
  - o **`ReadOnlyCollection<IWebElement> GetAllElementsUsingMatchingSelectorData(SelectorData selectorData, IWebDriverManager webDriverManager)`**
    - Finds and returns all elements matching the provided selector data
    - Data to check for in the current DOM instance

## 1.0.9-alpha

- Added method **`bool DoesUrlContainUsingRegex(IWebDriverManager webDriverManager, string pattern)`**
  - o Checks whether or not the current url contains the provided pattern using `regex.IsMatch`

- Added method **void SetDriverOptions(ChromeOptions chromeOptions = null, FirefoxOptions firefoxOptions = null, InternetExplorerOptions internetExplorerOptions = null, EdgeOptions edgeOptions = null)**
  - Sets the options for the associated driver. If the driver is active when this method is called it will be recreated
  - If no options objects are set the default options defined on the driver will be used
- Added **method void SetFirefoxProfile(FirefoxProfile firefoxProfile)**
  - Sets the profile for the firefox driver. If the driver is active when this method is called it will be recreated.
  - If no profile object is set the default profile defined on the browser will be used

## 1.0.8-alpha

- Updated signature of the method CheckChildElementExistsAndReturnIt to take an optional parameter of int nthParentElement
  - This is meant for cases where there are identical elements that could be returned for the parent
  - It is a zero based value used to select a specific parent element and then use it for the starting point of the child search
  - The default value is -1