

ReleaseNotes

Wednesday, February 6, 2019 10:53 AM

1.1.1 - Current Release

- 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)**
 - 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