En ocasiones no ha terminado de refrescar la pagina, y direcciona al element anterior, primero se le dice que espere, no hasta que es visible o encontrado, sino dele un wait hasta que la pagina se halla refrescado para continuar

La pagina se regenera con los mismos objetos, pero los nuevos objetos tienen diferentes atributos.

Se le da un wait largo, o se le dice “Stale”

* Se da click a button
* Se dice, espere hasta que se desconectaron de DOM los objetos actuales
* Luego se le dice, traiga informacion de los objetos regenerados

// ESPERE HASTA QUE REFERENCIA AL OBJETO DESAPAREZCA PARA CONTINUAR (SE ESTA REFRESCANDO)

driver.WaitUntilElementIsStale(splitWindowList);

// ESPERE HASTA QUE LA REFERENCIA ES NUEVAMENTE VISIBLE, SE HA REGENERADO

elementId = "splitWindowList";

splitWindowList = driver.WaitUntilElementIsVisible(By.Id(elementId), Wait4);

tbody = splitWindowList.FindElements(By.TagName("tbody"))[0];

trow = tbody.FindElements(By.TagName("tr"))[0];

tcells = trow.FindElements(By.TagName("td"));

. . . . . . . .

public static void WaitUntilElementIsStale(this IWebDriver driver, IWebElement element, int timeout = 1000)

{

try

{

if (timeout > 0)

{

var wait = new WebDriverWait(driver, TimeSpan.FromMilliseconds(timeout));

wait.Until(ExpectedConditions.StalenessOf(element));

}

}

catch (NoSuchElementException)

{ }

catch (Exception)

{ }

finally

{ }

}

public static IWebElement WaitUntilElementIsVisible(this IWebDriver driver, By by, int timeout = 1000)

{

IWebElement element = null;

try

{

if (timeout > 0)

{

var wait = new WebDriverWait(driver, TimeSpan.FromMilliseconds(timeout));

element = wait.Until(ExpectedConditions.ElementIsVisible(by));

}

}

catch (NoSuchElementException)

{ }

catch (Exception)

{ }

finally

{ }

return element;

}