20-Feb-2020

-----------

CALENDER

--------

Ex-20 : Write script to select 25th of March in Calender

-- Traversing xpath is must for Calender

-- If similar objects use traversing xpath

-- By default it will select 1st element (For e.g. if we need to select Mar 25th, by default it will select Feb 25th as it has the same attribute value)

-- Use Relative and Absolute xpath to locate 25th Mar

//To initialize browser

System.setProperty("webdriver.chrome.driver", "./Drivers//chromedriver.exe");

WebDriver driver= new ChromeDriver();

driver.get("https://www.justdial.com/Travel/Train-Booking");

driver.manage().window().maximize();

//to click on "Departure" field

driver.findElement(By.id("departDate")).click();

//to pause execution

Thread.sleep(4000);

//to select 25th of March

driver.findElement(By.xpath("//div[@class='ui-datepicker-group ui-datepicker-group-last']/table/tbody/tr[4]/td[4]/a")).click();

=======================================================================

Ex-21: Develop automation test script to select June 1st in Calender

Ctrl A (// To select the entire code)

Ctrl+Shift+F -- (// To put the entire code in proper format)

-- For visible text

//span[text()='Feb'] //but after 10 days the value changes to Mar coz of dynamic functionality

-- \*\*\* No need to use @ when using text() attribute

// Will not go for class attribute as it is not unique

//To initialize browser

System.setProperty("webdriver.chrome.driver", "./Drivers//chromedriver.exe");

WebDriver driver= new ChromeDriver();

driver.get("https://www.justdial.com/Travel/Train-Booking");

driver.manage().window().maximize();

//to click on "Departure" field

driver.findElement(By.id("departDate")).click();

//to pause execution

Thread.sleep(4000);

//to read first month name

String month01=driver.findElement(By.xpath("//div[@class='ui-datepicker-group ui-datepicker-group-first']/div/div/span[1]")).getText();

System.out.println(month01);

if (month01.equalsIgnoreCase("June")) {

//to select 1st Date

driver.findElement(By.linkText("1")).click();

}

else {

do{

//to read 2nd month name

String month02=driver.findElement(By.xpath("//div[@class='ui-datepicker-group ui-datepicker-group-last']/div/div/span[1]")).getText();

System.out.println(month02);

if(month02.equalsIgnoreCase("June")) {

driver.findElement(By.xpath("//div[@class='ui-datepicker-group ui-datepicker-group-last']/table/tbody/tr/td[2]/a")).click();

break;

}

else {

//to click on "Next"

driver.findElement(By.xpath("//span[text()='Next']")).click();

Thread.sleep(4000);

}

}while(true);

}

=======================================================================

\*\*\*WebDriver Validation Commands \*\*\*

-----------------------------

WebDriver providing following methods to validate webelements

i. isSelected (Selenium WebDriver commands)

ii. isEnabled ( " )

iii. isDisplayed ( " )

iv. size : ( java related command )

========================================================================

i. isSelected():

------------

This method used to check specified webelement selected or not based on that it will return true/false

Ex. check box, radio button,... etc

ii. isEnabled():

-----------

To check specified webelement is enabled or not in a webpage, based on that it will return true/false

Syntax: WebElement.isEnabled();

\*\*\*iii. isDisplayed():

-------------

Using this command we can check specified webelement visible/invisible in a page, based on that it will return true/false

Syntax: webelement.isDisplayed();

Note: To use isDisplayed() method that webelement code should exist in DOM irrespective of that webelement visible/invisible in a page.

=================================================================================

Ex-22:

Write script to check Telugu language link visible or invisible in a page at www.google.co.in home page

//To initialize browser

System.setProperty("webdriver.chrome.driver", "./Drivers//chromedriver.exe");

WebDriver driver= new ChromeDriver();

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

driver.manage().window().maximize();

//to check Telugu language link availability

if(driver.findElement(By.linkText("??????")).isDisplayed()) {

System.out.println("Telugu link displayed");

}

else {

System.out.println("Telugu link doesn't exist");

}

=================================================================================

iv. size():

-------

To check specified webelement code availibility in a page source code(DOM) we can use size() method

-- DOM -- to check the check source code

Syntax: driver.findElements(By.locator("locator value")).size();

================================================================================

Ex-23: Write script to check "telugu" language link availibility in Google home page before and after click on that link

//To initialize browser

System.setProperty("webdriver.chrome.driver", "./Drivers//chromedriver.exe");

WebDriver driver= new ChromeDriver();

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

driver.manage().window().maximize();

System.out.println("Before click on Telugu link");

//to check Telugu language link availability

if(driver.findElement(By.linkText("??????")).isDisplayed()) {

System.out.println("Telugu link displayed");

driver.findElement(By.linkText("??????")).click();

}

else {

System.out.println("Telugu link doesn't exist");

}

Thread.sleep(4000);

System.out.println("After click on Telugu link");

//to get Telugu language links availability in a page

List<WebElement> teluguLinks=driver.findElements(By.linkText("??????"));

int n= teluguLinks.size();

if(n == 0) {

System.out.println("Telugu links do not exist");

}

else {

System.out.println("No. of telugu links are: "+n);

}

================================================================================

\*\*\*\* Synchronization in WD \*\*\*\*\*\*

---------------------

The time interface between "execution time of tool" to "process completion time in application" is called as "Synchronization"

(or)

It is a concept of specifying delay time between one step to another step during script execution

\*\*\*\* Why we need Synchronization

--------------------------------

To avoid "No such Element exception" and to wait for some time during script execution

"No such element exception" given by WD when specified webelement does not exist in application page to perform operation.

-- In general webelement load time may vary in application during runtime.

When to use Synchronization

---------------------------

- When application loading next page (i.e. Page load time)

- To reach status bar 100% color filled (i.e. uploading file)

- When application performing some back end transactions like (select,insert,delete) like Insert,Update and Delete

- While performing search operations

Synchronization options

-----------------------

There are 2 ways to synchronize test execution

1. Unconditional based Synchronization:

------------------------------------

Using "Thread" class we can pause execution fixed time, irrespective of process in application

Syntax: Thread.sleep(time in milliseconds);

Ex: Thread.sleep(5000);

It will pause execution 5000 milliseconds (i.e. fixed time)

2. Conditional based Synchronization

---------------------------------

Based on particular webelement (i.e. driver.findElement) or elements (i.e. driver.findElements) availibility we can pause execution.

--> In WebDriver we have 2 types of options to synchronize test execution

a. Implicit Wait

b. Explicit Wait

=================================================================END OF CLASS==================================================================