38. How to validate if element is hidden, displayed mode with Expect assertions

39. How to automate Java/JavaScript Alert popups with Playwright

These are called Java pop ups. So that means you cannot spy and get any html element from this.

For pops in playwright we call it as dialog.

There is a method called “on” - So this on method will help you to listen for events so it will emit when that event occurred.

So this js event will be regarding the browser and that a dialog box open.

So from our playwright we are smartly listening for this event to occur.

So when somebody clicks on this confirm what happens, that dialog box will open.

So that means that event occurred. So this step will immediately get notified.

Hey, there is a dialog box that what you want to do now. So the action, what you want to perform when that event occurred that you have to write it in the second argument here.

1st argument – is event

2nd argument – that action it will perfom

40. How to handle & Automate frames with Playwright – Example

How would you switch from parentframe to childframe

when you switch, it will give you a new page object from that.

So that new object you have to store it again and variable, let's call it other frames page.

You can tell playwright to select only that element which is in visible mode.

How to do that?

const {test, expect}  = require('@playwright/test');

test('MoreValidations', async ({page})=>    {

await page.goto("https://www.rahulshettyacademy.com/AutomationPractice/");

//await page.goto("https://www.google.com/");

//await page.goBack();

//await page.goForward();

await expect(page.locator("#displayed-text")).toBeVisible();  // Element Displayed Example // Hide

await page.locator("#hide-textbox").click();

await expect(page.locator("#displayed-text")).toBeHidden();

page.on('dialog',dialog => dialog.accept());   // alert pop click on ok

await page.locator("#confirmbtn").click();   // Switch To Alert Example

await page.locator("#mousehover").hover();

//main parent frame to child frame switch

const framePage = page.frameLocator("#courses-iframe"); // switch to child frame

await framePage.locator("li a[href\*='lifetime-access']:visible").click();

const textCheck = await framePage.locator(".text h2").textContent();

console.log(textCheck.split(" ")[1]);

});