

Assignment: Playwright

Submitted By: Neha Sharma

Assignment 1: Launch Browser, Open Website & Take Screenshot

```
import test from '@playwright/test'

test(" Screenshot",async{page})=>{

await page.goto("https://dd-demo-tau.vercel.app/web_elements.html#9-file-
upload");

await page.screenshot({path:'test.png'});

})
```

Assignment 2: Automate Login and Verify Dashboard

```
import { test, expect } from '@playwright/test';

test("Login Test Case", async ({ page }) => {

await page.goto("https://practicetestautomation.com/practice-test-login/");

await expect(page).toHaveTitle("Test Login | Practice Test Automation");

await page.locator('input[name="username"]').fill("student");

await page.locator('input[name="password"]').fill("Password123");

await page.locator('button#submit').click();

await expect(page).toHaveURL(/logged-in-successfully/);

});
```

Assignment 3: Handle Popups and Alerts

```
import {test,expect} from '@playwright/test';

test('01-handling alert',async({page})=>{
    page.once('dialog',dialog=>dialog.accept());
    await page.goto("https://testpages.eviltester.com/pages/basics/alerts-javascript/");
    await page.locator('#alertexamples').click();
});

test('02-handling confirm', async({page})=>{
    page.once('dialog',dialog=>dialog.dismiss());
    await page.goto("https://testpages.eviltester.com/pages/basics/alerts-javascript/");
    await page.locator('#confirmexample').click();
});

test('03-handling prompts', async({page})=>{
    page.once('dialog', async dialog => {
        console.log("Prompt message:", dialog.message());
        await dialog.accept("Done");
    });
});
```

```
});

await page.goto("https://testpages.eviltester.com/pages/basics/alerts-javascript/");

await page.locator('#promptexample').click();

});


```

Assignment 4: Upload a File

```
import {test,expect} from '@playwright/test'

test("File Upload",async({page})=>{

await page.goto("https://dd-demo-tau.vercel.app/web_elements.html#9-file-
upload");

await page.locator('a[href="#9-file-upload"]').click();

await page.locator('input[id="fileUploadField"]').setInputFiles('tests/example.docx');

const messageLocator = page.locator('div[id="fileUploadMsg"]');

await expect(messageLocator).toBeVisible();

const visibleText = await messageLocator.innerText();

console.log('Upload message:', visibleText);

});


```

Assignment 5: Count Hyperlinks on a Page

```
const { test, expect } from '@playwright/test';

test('Count hyperlinks on page', async ({ page }) => {
  await page.goto('https://playwright.dev/');

  const links = await page.locator('a').count();

  console.log('Total hyperlinks:', links);

  expect(links).toBeGreaterThan(0);
});
```