

## 2. Automation Testing (playwright with Javascript)

-<https://demoqa.com/>

### ● Elements

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

test('Elements Section Test', async ({ page }) => {
  await page.goto('https://demoqa.com/');

  // Elements
  await page.click("//h5[text()='Elements']");
  await page.click("//div[contains(@class, 'icon')][1]");
  await
page.click("//*[name()='svg'][contains(@stroke, 'currentColor')][2]");
  await page.waitForTimeout(2000);

});
```

### ● Textbox

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

test('Verify Elements(TextBOX) Section ', async ({ page }) => {
  await page.goto('https://demoqa.com/');

  // Elements
  await page.click("//h5[text()='Elements']");

  // Text Box
  await page.click("//li[@id='item-0'][1]");
  await page.waitForTimeout(2000);

  await page.fill("#userName", 'Usrah');
  await page.fill("#userEmail", 'sa@example.com');
  await page.fill("#currentAddress", 'road-1, block-a, Dhaka');
  await page.fill("#permanentAddress", 'Barishal');
```

```

    await page.click("#submit");
    await page.waitForSelector("//button[@id='submit']", { state:
'visible' });
    await page.waitForTimeout(2000);
  });

```

### ● Checkbox

```

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

test('Verify Elements-(Checkbox) Section', async ({ page }) => {
  await page.goto('https://demoqa.com/');

  //Elements
  await page.click("//h5[text()='Elements']");

  await page.click("//span[text()='Check Box']");
  await page.click("//button[@title='Expand all']");
  await page.waitForTimeout(2000);

  await page.check("//span[text()='Desktop']");

  await
page.check("//label[@for='tree-node-angular']//span[@class='rct-ch
eckbox']");
});

```

### ● Radio Buttons

```

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

test('Verify Elements-Radio Button Funtionality', async ({ page })
=> {
  await page.goto('https://demoqa.com/');

  // Elements
  await page.click("//h5[text()='Elements']");

  await page.click("//span[text()='Radio Button']");
  await page.waitForTimeout(2000);

```

```

    await page.check("//label[@for='yesRadio']");
    await expect(await
page.locator("//label[@for='yesRadio']")).toBeChecked();

    await page.waitForTimeout(2000);

    await page.check("//label[normalize-space()='Impressive']");
    await expect(await
page.locator("//label[normalize-space()='Impressive']")).toBeChecked
();
});

```

### ● Web Elements

```

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

test('Verify Elements (Web Tables) Section', async ({ page }) => {
    await page.goto('https://demoqa.com/');

    // Elements
    await page.click("//h5[text()='Elements']");

    await page.click("//li[@id='item-3']")[1];
    await page.click("//button[normalize-space()='Add']")[1];

    await page.fill("//input[@id='firstName']")[1], 'Abcd');
    await page.fill("//input[@id='lastName']")[1], 'Xyz');
    await page.fill("//input[@id='userEmail']")[1],
'Abcd@auto.com');
    await page.fill("//input[@id='age']")[1], '25');
    await page.fill("//input[@id='salary']")[1], '10000');
    await page.fill("//input[@id='department']")[1], 'QA');

    // submit button visible
    await page.waitForSelector("//button[@id='submit']", { state:
'visible' });
    await page.click("//button[@id='submit']");

```

```

    await page.fill("#searchBox" , 'Cierra');
    await page.waitForTimeout(2000);

    // Edit profile
    await page.click("span[id='edit-record-1'] svg path");
    await page.waitForTimeout(2000);
    await page.fill("//input[@id='lastName'] [1]", ' vegas');
    await page.click("//button[@id='submit']");
    await page.waitForTimeout(2000);

    //Delete
    await page.click("(//*[name()='path']) [55]");
    await page.waitForTimeout(2000);

  });

```

## ● Buttons

```

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

test('Verify Elements-Buttons Section', async ({ page }) => {
  await page.goto('https://demoqa.com/');

  await page.click("//h5[text()='Elements']");
  await page.click("//li[@id='item-4'] [1]");
  await
page.locator("//button[@id='doubleClickBtn']").dblclick();
  await
page.locator("//button[@id='rightClickBtn']").click({ button:
'right' });

  await page.click("(//button[normalize-space()='Click
Me']) [1]");
  await page.waitForTimeout(2000);

});

```

## ● Links

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

test('Elements Section Test - Verify All the Links', async ({ page,
context }) => {
    await page.goto('https://demoqa.com/');

    await page.click("//h5[text()='Elements']");
    await page.click("//li[@id='item-5'] [1]");
    await page.waitForTimeout(2000);

    // Following links will open in a new tab and close
    const [newPage1] = await Promise.all([
        context.waitForEvent('page'),
        page.locator('#simpleLink').click()
    ]);
    await newPage1.waitForLoadState();
    console.log('First link opened:', newPage1.url());
    await newPage1.close();

    await page.waitForTimeout(2000);

    const [newPage2] = await Promise.all([
        context.waitForEvent('page'),
        page.locator('#dynamicLink').click()
    ]);

    await newPage2.waitForLoadState();
    console.log('Second link opened:', newPage2.url());
    await newPage2.close();
    await page.waitForTimeout(2000);

    // Following links will send an API call
    await page.click("//a[@id='created']");
    await page.waitForTimeout(1000);
```

```

    await page.click("//a[@id='no-content']");
    await page.waitForTimeout(1000);

    await page.click("//a[@id='moved']");
    await page.waitForTimeout(2000);

    await page.click("//a[@id='bad-request']");
    await page.waitForTimeout(2000);

    await page.click("//a[@id='unauthorized']");
    await page.waitForTimeout(2000);

    await page.click("//a[@id='forbidden']");
    await page.waitForTimeout(2000);

    await page.click("//a[@id='invalid-url']");
    await page.waitForTimeout(2000);

  });

```

- Upload & Download

```

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

test('Verify download & upload functionality', async ({ page }) => {
  // Navigate to the page
  await page.goto('https://demoqa.com/upload-download');

  await page.waitForTimeout(2000);

  //Download File
  await page.click('#downloadButton');
  await page.waitForTimeout(2000);

  //Upload File

```

```
    const handle = page.locator("//input[@id='uploadFile']");  
    await  
handle.setInputFiles("C:/Users/DBA/Downloads/sampleFile.jpeg")  
; //file path  
  
    await page.pause();  
  
    await page.waitForTimeout(2000);  
});
```