



# Selenium to Playwright Journey

My Playwright Journey

---

Prepared By: Ashish Dhote

Role: Senior Consultant

Email: [ashish.dhote@ttcglobal.com](mailto:ashish.dhote@ttcglobal.com)

20<sup>th</sup> October 2022



# Agenda

1. My Own Experience

2. What Resources Are Available?

3. Tips to Maximise Learning

4. Reflection

# About Me

# About Me

- ✓ Auckland University of Technology – Information & Computer Science 2016
- ✓ Joined TTC in March 2022
- ✓ Test Automation Engineer for 8 years
- ✓ Scrum Master Certified



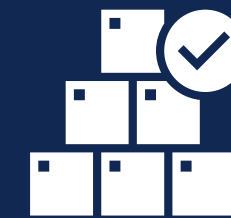
# What is Playwright?

# What is Playwright?

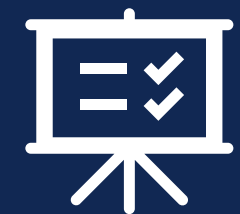
Playwright enables reliable end-to-end testing for modern web apps.



Any Browser - Any Platform - One API



Full Isolation - Fast Execution



Resilient - No Flaky Tests



No Trade-Offs - No Limits



Powerful Tooling

# My Own Experience

# My Own Experience



## Initial Training

Udemy Course/Playwright  
Documentation

Build working knowledge by  
creating POC



## Real-Life Project

Team members present to  
support and assist



## Continuous Growth

Continued practical experience  
=  
Continual growth



# Selenium vs Playwright

## Basic Scenario Comparison Between Selenium and Playwright

#1 Cross Browser Support

### Selenium

```
if(browser.equalsIgnoreCase("firefox")){
    //create firefox instance
    System.setProperty("webdriver.gecko.driver",
        ".\\geckodriver.exe");
    driver = new FirefoxDriver();
}
//Check if parameter passed as 'chrome'
else if(browser.equalsIgnoreCase("chrome")){
    //set path to chromedriver.exe
    System.setProperty("webdriver.chrome.driver", ".\\chromedriver.exe");
    //create chrome instance
    driver = new ChromeDriver();
}
//Check if parameter passed as 'Edge'
else if(browser.equalsIgnoreCase("Edge")){
    //set path to Edge.exe
    System.setProperty("webdriver.edge.driver", ".\\MicrosoftWebDriver.exe");
    //create Edge instance
    driver = new EdgeDriver();
}
else{
    //If no browser passed throw exception
    throw new Exception("Browser is not correct");
}
```

### Playwright

```
// playwright.config.js
// @ts-check
const { devices } = require('@playwright/test');

/** @type {import('@playwright/test').PlaywrightTestConfig} */
const config = {
  projects: [
    {
      name: 'chromium',
      use: { ...devices['Desktop Chrome'] },
    },
    {
      name: 'firefox',
      use: { ...devices['Desktop Firefox'] },
    },
    {
      name: 'webkit',
      use: { ...devices['Desktop Safari'] },
    },
  ],
};

module.exports = config;
```

`npx playwright test --project=chromium`

# Selenium vs Playwright

## Basic Scenario Comparison between Selenium and Playwright

### #2 Handling multiple tabs in browser

#### Selenium

```
String parentWindowHandle = driver.getWindowHandle();
System.out.println("Parent window's handle -> " + parentWindowHandle);
WebElement clickElement = driver.findElement(By.id("button1"));

for(int i = 0; i < 3; i++)
{
clickElement.click();
Thread.sleep(3000);
}

Set<String> allWindowHandles = driver.getWindowHandles();

for(String handle : allWindowHandles)
{
System.out.println("Window handle - > " + handle);
}

}
```

#### Playwright

```
// Get page after a specific action (e.g. clicking a link)
const [newPage] = await Promise.all([
  context.waitForEvent('page'),
  page.locator('a[target="_blank"]').click() // Opens a new tab
])
await newPage.waitForLoadState();
console.log(await newPage.title());
```

# Selenium vs Playwright

## Basic Scenario Comparison between Selenium and Playwright

### #3 Handling iFrames

#### Selenium

```
for(int i=0; i<=size; i++){  
    driver.switchTo().frame(i);  
    int total=driver.findElements(By.xpath("html/body/a/img")).size();  
    System.out.println(total);  
    driver.switchTo().defaultContent();}
```

#### Playwright

```
// Locate element inside frame  
const username = await page.frameLocator('.frame-class').getByLabel('User Name');  
await username.fill('John');
```

# Selenium vs Playwright

## Basic Scenario Comparison between Selenium and Playwright

### #4 Screenshots

#### Selenium

```
//Take the screenshot
File screenshot = ((TakesScreenshot) driver).getScreenshotAs(OutputType.FILE);

//Copy the file to a location and use try catch block to handle exception
try {
    FileUtils.copyFile(screenshot, new File("C:\\projectScreenshots\\homePageScreenshot.png"));
} catch (IOException e) {
    System.out.println(e.getMessage());
}
```

#### Playwright

```
/** @type {import('@playwright/test').PlaywrightTestConfig} */
const config = {
  use: {
    screenshot: 'only-on-failure',
  },
};

module.exports = config;
```

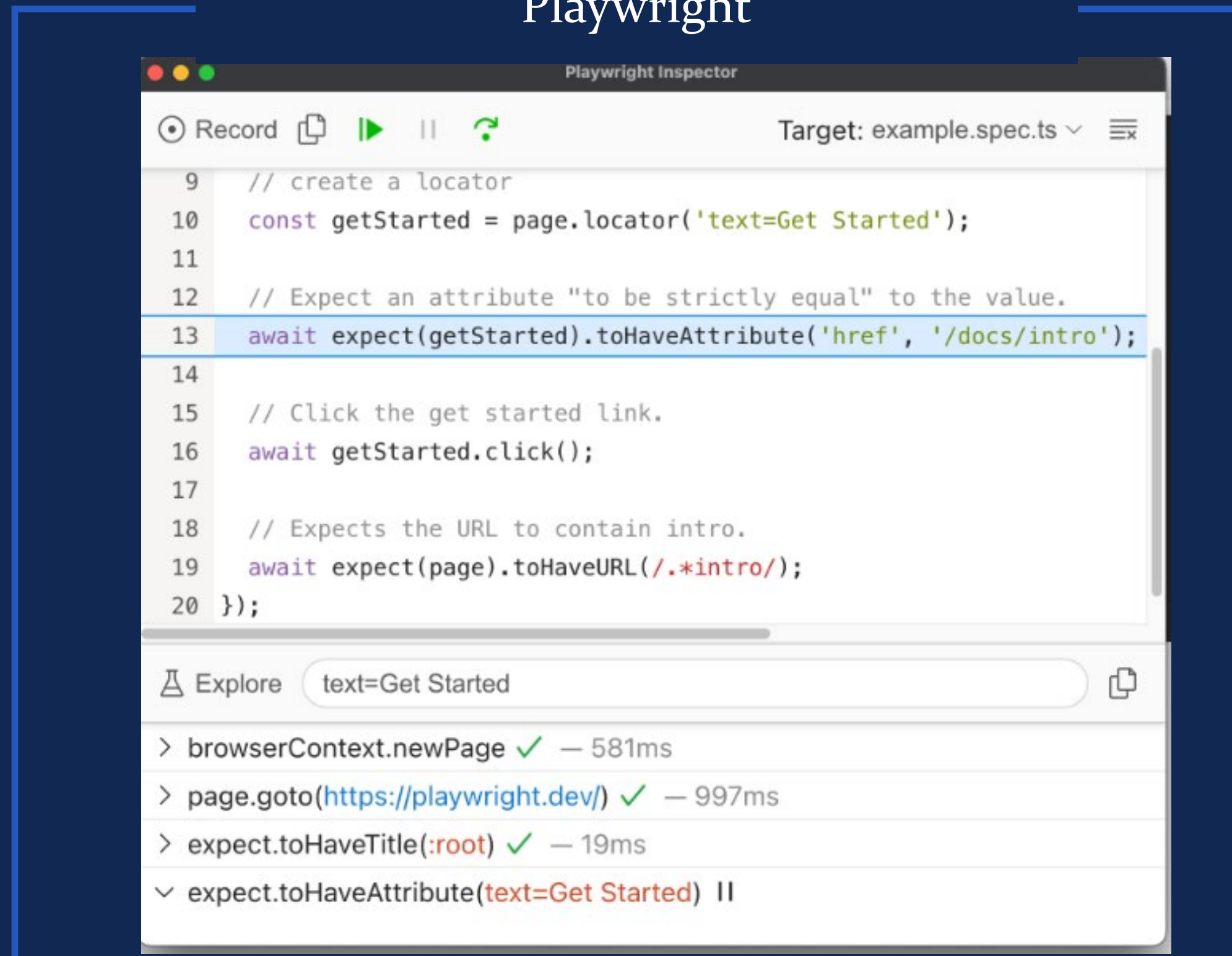
- On – Capture screenshot after each test
- Off – Do not capture screenshot
- Only-On-failure

# Selenium vs Playwright

Basic Scenario Comparison between Selenium and Playwright

#5 Debug Window

## Playwright



The screenshot displays the Playwright Inspector interface. At the top, there are control buttons for Record, Copy, Play, Pause, and Refresh, along with the target file 'example.spec.ts'. The main area shows a TypeScript test script with the following code:

```
9 // create a locator
10 const getStarted = page.locator('text=Get Started');
11
12 // Expect an attribute "to be strictly equal" to the value.
13 await expect(getStarted).toHaveAttribute('href', '/docs/intro');
14
15 // Click the get started link.
16 await getStarted.click();
17
18 // Expects the URL to contain intro.
19 await expect(page).toHaveURL(/.*intro/);
20 });
```

Below the code editor, the 'Explore' section shows the selected locator 'text=Get Started'. The execution log at the bottom shows the following steps:

- > browserContext.newPage ✓ — 581ms
- > page.goto(https://playwright.dev/) ✓ — 997ms
- > expect.toHaveTitle(:root) ✓ — 19ms
- ∨ expect.toHaveAttribute(text=Get Started) II

# What Resources are Available for Learning Playwright?

# What Resources are Available?



Udemy courses on how to learn and use Playwright



Playwright documentation has a lot of useful information



Playwright



Experienced colleagues within or outside your organisation



# Tips to Maximise Learning



# Tips to Maximise Learning



- Gain a solid understanding of the concepts from each exercise in the course.



- Experiment & create own examples using concepts gained



- Practice makes perfect

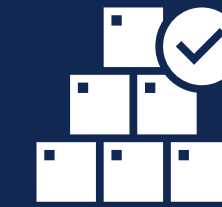
# Reflection

# Reflection



## Ask Questions

Not hesitate to reach out for support at initial stages



## Practical Experience

Gained most traction in upskilling process through practical experience



## Explore & Experiment

Explore and experiment with the tool



## Enjoy the learning process

Try not to pressure yourself and enjoy your learning!

Thank You + Questions?

# ✓ttc | Contact Us

## New Zealand

Level 6, Cathedral House  
48-52 Wyndham St.  
Auckland 1010, New Zealand  
+64 9 948 2225  
info@ttcglobal.com

## United Arab Emirates

14<sup>th</sup> Floor, Al Khatem Tower  
Wework Hub 71 Abu Dhabi  
Global Market Square, Al  
Maryah Island Abu Dhabi, UAE  
+971 58 5233912  
UAE@ttcglobal.com

## United States

25211 Grogans Mill Rd #450  
The Woodlands,  
Texas 77380  
(832) 813-8063  
sales.us@ttcglobal.com

## India

6 Floor Westport S.No.  
32/1A/1/30 to 38 & 54 Pan  
Card Club Rd, Baner, Pune,  
Maharashtra 411045  
india@ttcglobal.com

## Europe | UK

25 Crossharbour Plaza #2304  
London E14 9SS  
United Kingdom  
+44 7348 719098  
uk@ttcglobal.com

## Australia

154 Lawrence St  
Alexandria  
Sydney  
NSW 2015  
+61 2 8999 1965  
australia@ttcglobal.com

## Singapore

Hong Leong Building  
6 Raffles Quay, #33-03  
Singapore 048581  
+65 9822 6679  
singapore@ttcglobal.com

