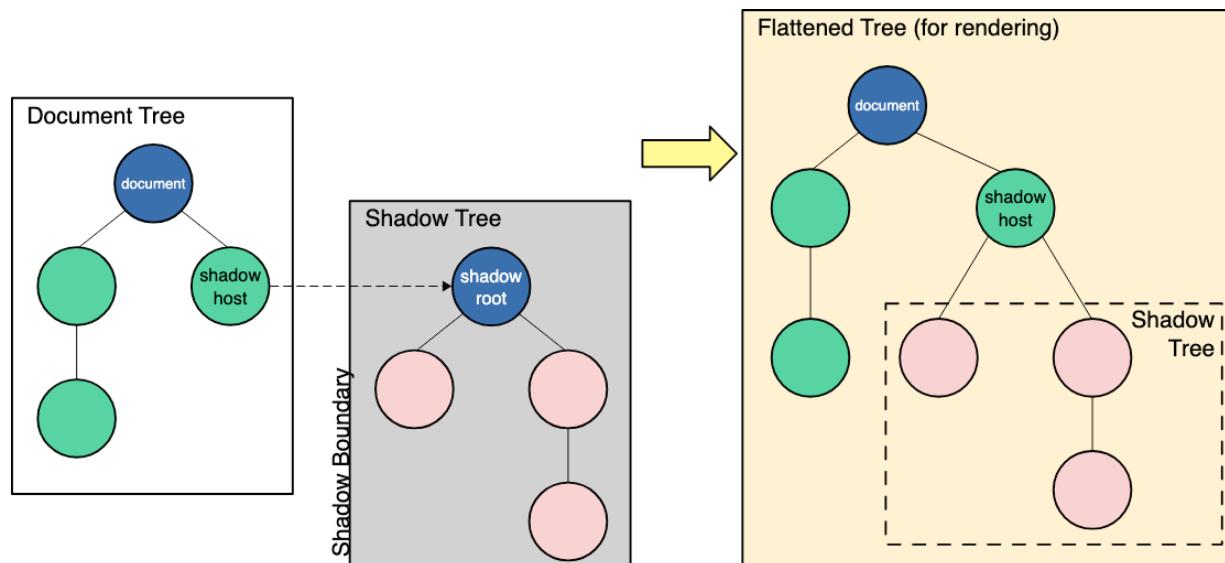


## Shadow DOM

The Shadow DOM is a hidden and encapsulated part of the DOM tree, typically used within web components to isolate internal structure and styling from the main document.



### Shadow Host

- A regular DOM element to which a Shadow DOM is attached.
- It acts as the entry point or anchor for the shadow tree.

### Shadow Root

- The root node of the Shadow DOM.
- It's where the shadow tree begins.

### Shadow Tree

- A collection of DOM elements encapsulated within the shadow root.
- It represents the internal structure of the Shadow DOM.

**Shadow Host + Shadow Tree = Shadow DOM**

### Shadow DOM in Playwright

- Playwright natively supports working with elements inside the Shadow DOM.
- You can use CSS selectors to interact with shadow elements.
- XPath selectors **do not work** with elements inside the Shadow DOM in Playwright.

# Cookies in Playwright

## What Are Cookies?

Cookies are small pieces of data stored on the client side (browser). They are mainly used for:

- Session management (e.g., login sessions)
- Personalization (e.g., themes, preferences)

## Playwright and Cookies

In Playwright, we can **add**, **get**, and **delete** cookies using its powerful API, which allows us to simulate real-world browser behavior.

### Get Cookies

You can retrieve cookies using **context.cookies()** or **page.context().cookies()**.

```
const cookies = await context.cookies();
console.log(cookies);
```

If you want cookies for a specific URL:

```
const cookies = await context.cookies('https://example.com');
```

### Set/Add Cookies

You can set cookies using **context.addCookies()**.

```
await context.addCookies([
  {
    name: 'token',
    value: 'abc123',
    domain: 'example.com',
    path: '/',
    httpOnly: true,
    secure: true,
    sameSite: 'Lax',
  },
]);
```

**Note:** Set cookies **before** navigating to the page if required for authentication or tracking.

## Delete Cookies

Playwright doesn't have a direct "deleteCookie" function, but you can **clear all cookies** using:

```
await context.clearCookies();
```

### Notes:

- Use browser tools (DevTools → Application → Cookies) to inspect live cookies.
- Use cookies to simulate authenticated sessions during testing.