Here's the **shortened step-by-step process** for setting up and running Playwright using **PowerShell**:

Step 1: Set Execution Policy

1. Open PowerShell as Administrator.
2. Set execution policy:

   Set-ExecutionPolicy RemoteSigned -Scope CurrentUser

   - Type Y to confirm.
3. Verify:

   Get-ExecutionPolicy -Scope CurrentUser

   It should return RemoteSigned.

Step 2: Install Node.js

1. Download and install Node.js **LTS version** from [Node.js official website](#).
2. Verify installation:

   node -v

   npm -v

Step 3: Create Playwright Project

1. Create a folder for the project:

   mkdir playwright-automation

   cd playwright-automation

2. Initialize the project:

   npm init -y

3. Install Playwright and browsers:

```
npm install --save-dev @playwright/test

npx playwright install
```

Step 4: Create a Test File

1. Create a folder and test file:

```
mkdir tests

New-Item -Path tests/example.spec.js -ItemType File
```

2. Add this code to the test file:

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

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

    await page.goto('https://example.com');

    const title = await page.title();

    console.log(title);

    expect(title).toBe('Example Domain');

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

});
```

Step 5: Run the Test

1. Run the test:

```
npx playwright test
```

2. Run in headed mode (non-headless):

```
npx playwright test --headed
```

Optional: Reset Execution Policy

After completing, reset the execution policy:

Set-ExecutionPolicy Restricted -Scope CurrentUser

Folder Structure

Your folder should now look like this:

playwright-automation/

├── tests/

│   ├── example.spec.js

├── node_modules/

├── package.json

├── package-lock.json

This condensed process ensures a clean setup while keeping it simple and efficient! 🚀