

---

PART 5: TEST STRUCTURE & CONFIGURATION

# Test Architecture

Interview Prep Series  
Questions 41 to 50

Abhishesh Mishra

+91 6387782073

## What is playwright.config.ts used for?

It is the central brain of your project. It defines global timeouts, browser projects (Chrome, Firefox, Webkit), reporter settings, and base URLs.

```
export default defineConfig({  
  timeout: 30000,  
  use: { baseURL: 'https://site.com' }  
});
```

## What are retries in Playwright?

Retries allow Playwright to automatically re-run a failed test. This is essential for handling "flaky" tests that might fail due to network glitches but pass on the second try.

```
In config      //  
retries: process.env.CI ? 2 : 0
```

**Q43: TIMEOUTS**

## How do you set timeout for a test?

You can set timeouts globally in the config, or override them for a specific test file or block.

```
test('slow test', async ({ page }) => {
  page.setTimeout(60000);
  // ...
});
```

**Q44: SETUP & TEARDOWN**

## What is global setup & teardown?

Global Setup runs once before any tests start (e.g., logging in once). Global Teardown runs once after all tests finish (e.g., clearing a DB).

```
globalSetup: './global-setup',
globalTeardown: './global-teardown'
```

## How to run tests in parallel?

By default, Playwright runs files in parallel. You can use `fullyParallel: true` to run tests inside the same file in parallel too.

```
export default defineConfig({
  fullyParallel: true,
  workers: 4
});
```

## How to skip tests?

Use `test.skip()` to ignore a test. You can also skip based on conditions (e.g., skip on Firefox).

```
test.skip('not ready', async () => { ... });

test.skip(browserName === 'firefox', 'Broken on FF');
```

## How to tag tests?

Add tags to test titles (e.g., @smoke). You can then run only those tests using the --grep command.

```
test('login @smoke', async () => { ... });

CLI: npx playwright test --grep @smoke
```

## How to run a single test file?

Provide the path to the file in the terminal. You can also use -g to target a specific test name.

```
npx playwright test login.spec.ts
```

## What is expect.soft()?

Standard assertions stop the test. Soft assertions log the failure but allow the rest of the test code to keep running.

```
await expect.soft(page).toHaveTitle('Home');
await page.click('#next'); // Still runs even if title fails
```

## What is beforeEach and afterEach?

These hooks run once per file. beforeEach runs before any test in the file, and afterEach runs after all tests finish.

```
test.beforeAll(async () => {
  console.log('Starting File Suite ... ');
});
```