Playwright Browser test runner

CONCEPTS

Test filters are regexes on the file path, so "login" runs every file with login in its path.

Config projects, browsers
↓ `test`
Workers parallel files
↓ `reporter`
Report HTML, traces
↓ `show-trace`
Trace viewer timeline, DOM

Each project in the config is one browser or device setup. Select it with --project.

SETUP

`npm init playwright@latest`
Scaffold config, example tests and CI file

`npx playwright install`
Install the default browsers

`npx playwright install --with-deps`
Install browsers and system libraries

RUNNING TESTS

`npx playwright test`
Run every test headless

`npx playwright test <file>`
Run files matching a path regex

`npx playwright test <file>:<line>`
Run the test at a line

`npx playwright test -g "<title>"`
Run tests whose title matches

`npx playwright test --project=<name>`
Run one project (browser)

`npx playwright test --headed`
Show the browser window

`npx playwright test --last-failed`
Re-run only the failures

DEBUGGING

`npx playwright test --ui`
Interactive UI mode with time travel

`npx playwright test --debug`
Run with the Inspector, headed, 1 worker

`npx playwright test --trace on`
Record a trace for every test

`npx playwright show-trace <trace.zip>`
Open a trace in the viewer

PARALLELISM AND CI

`npx playwright test --workers <N>`
Number of workers or a percent of cores

`npx playwright test --workers=1`
Run serially in one worker

`npx playwright test --retries <N>`
Retry failed tests N times

`npx playwright test --shard <n>/<total>`
Run one slice, e.g. 2/4

`npx playwright test --forbid-only`
Fail if test.only is left in

REPORTS AND OUTPUT

`npx playwright show-report`
Open the last HTML report

`npx playwright test --reporter <name>`
Choose reporters: list, line, dot, html, json, junit

SNAPSHOTS

`npx playwright test -u`
Update snapshots that changed

CODEGEN

`npx playwright codegen <url>`
Record actions into test code

`npx playwright codegen --save-storage <file>`
Save logged-in state