

Writing tests for Next.js

Getting Started

You can set-up a new test using `yarn new-test` which will start from a template related to the test type.

Test Types in Next.js

- e2e: These tests will run against `next dev` and `next start`
- development: These tests only run against `next dev`
- production: These tests will run against `next start`.
- integration: These tests run misc checks and modes and is where tests used to be added before we added more specific folders. Ideally we don't add new test suites here as tests here are not isolated from the monorepo.
- unit: These are very fast tests that should run without a browser or running next and should be testing a specific utility.

For the e2e, production, and development tests the `createNext` utility should be used and an example is available [here](#). This creates an isolated Next.js install to ensure nothing in the monorepo is relied on accidentally causing incorrect tests.

All new test suites should be written in TypeScript either `.ts` (or `.tsx` for unit tests). This will help ensure we catch smaller issues in tests that could cause flakey or incorrect tests.

If a test suite already exists that relates closely to the item being tested (e.g. hash navigation relates to existing navigation test suites) the new checks can be added in the existing test suite.

Best Practices

- When checking for a condition that might take time, ensure it is waited for either using the browser `waitForElement` or using the `check` util in `next-test-utils`.
- When applying a fix, ensure the test fails without the fix. This makes sure the test will properly catch regressions.