exportPathMap

▶ **Examples**

`exportPathMap` allows you to specify a mapping of request paths to page destinations, to be used during export. Paths defined in `exportPathMap` will also be available when using `next dev` .

Let's start with an example, to create a custom `exportPathMap` for an app with the following pages:

- `pages/index.js`
- `pages/about.js`
- `pages/post.js`

Open `next.config.js` and add the following `exportPathMap` config:

```js
module.exports = {
  exportPathMap: async function (
    defaultPathMap,
    { dev, dir, outDir, distDir, buildId }
  ) {
    return {
      '/': { page: '/' },
      '/about': { page: '/about' },
      '/p/hello-nextjs': { page: '/post', query: { title: 'hello-nextjs' } },
      '/p/learn-nextjs': { page: '/post', query: { title: 'learn-nextjs' } },
      '/p/deploy-nextjs': { page: '/post', query: { title: 'deploy-nextjs' } },
    }
  },
}
```

Note: the `query` field in `exportPathMap` cannot be used with automatically statically optimized pages or getStaticProps pages as they are rendered to HTML files at build-time and additional query information cannot be provided during `next export` .

The pages will then be exported as HTML files, for example, `/about` will become `/about.html` .

`exportPathMap` is an `async` function that receives 2 arguments: the first one is `defaultPathMap` , which is the default map used by Next.js. The second argument is an object with:

- `dev` - `true` when `exportPathMap` is being called in development. `false` when running `next export` . In development `exportPathMap` is used to define routes.
- `dir` - Absolute path to the project directory
- `outDir` - Absolute path to the `out/` directory (configurable with `-o`). When `dev` is `true` the value of `outDir` will be `null` .
- `distDir` - Absolute path to the `.next/` directory (configurable with the `distDir` config)
- `buildId` - The generated build id

The returned object is a map of pages where the `key` is the `pathname` and the `value` is an object that accepts the following fields:

- `page` : `String` - the page inside the `pages` directory to render
- `query` : `Object` - the `query` object passed to `getInitialProps` when prerendering. Defaults to `{}`

> The exported `pathname` can also be a filename (for example, `/readme.md`), but you may need to set the `Content-Type` header to `text/html` when serving its content if it is different than `.html` .

Adding a trailing slash

It is possible to configure Next.js to export pages as `index.html` files and require trailing slashes, `/about` becomes `/about/index.html` and is routable via `/about/` . This was the default behavior prior to Next.js 9.

To switch back and add a trailing slash, open `next.config.js` and enable the `trailingSlash` config:

```
module.exports = {
  trailingSlash: true,
}
```

Customizing the output directory

`next export` will use `out` as the default output directory, you can customize this using the `-o` argument, like so:

```
next export -o outdir
```

Related

Introduction to next.config.js: Learn more about the configuration file used by Next.js.
Static HTML Export: Export your Next.js app to static HTML.