▲ / NEXT.Js   🔍
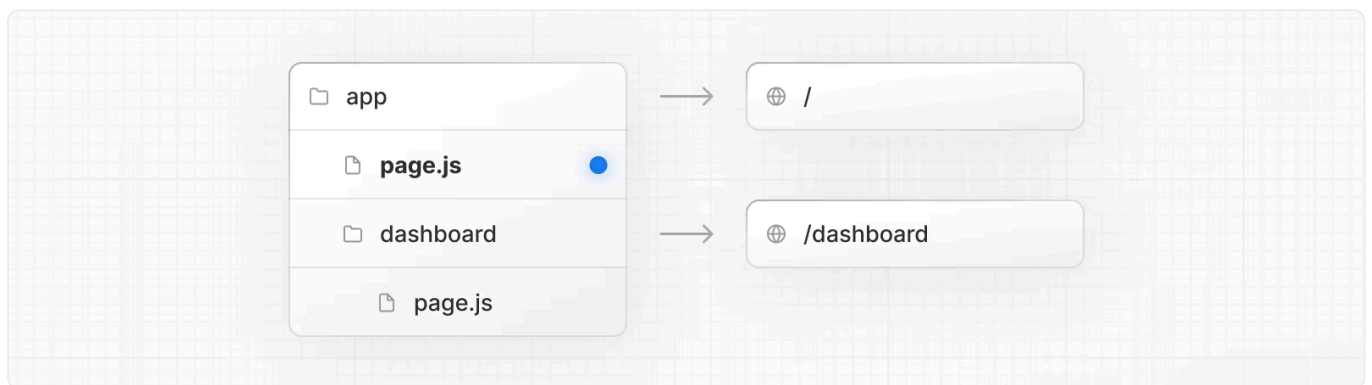
> Menu

App Router > … > Routing > Pages

Pages

A page is UI that is **unique** to a route. You can define a page by default exporting a
component from a `page.js` file.

For example, to create your `index` page, add the `page.js` file inside the `app` directory:

TS app/page.tsx	TypeScript ⌄ ⧉

```tsx
1  // `app/page.tsx` is the UI for the `/` URL
2  export default function Page() {
3    return <h1>Hello, Home page!</h1>
4  }
```

Then, to create further pages, create a new folder and add the `page.js` file inside it. For
example, to create a page for the `/dashboard` route, create a new folder called `dashboard`,
and add the `page.js` file inside it:

TS app/dashboard/page.tsx	TypeScript ⌄ ⧉

```tsx
1  // `app/dashboard/page.tsx` is the UI for the `/dashboard` URL
2  export default function Page() {
```

```
3      return <h1>Hello, Dashboard Page!</h1>
4    }
```

Good to know:

- The `.js`, `.jsx`, or `.tsx` file extensions can be used for Pages.

- A page is always the leaf of the route subtree.

- A `page.js` file is required to make a route segment publicly accessible.

- Pages are Server Components by default, but can be set to a Client Component.

- Pages can fetch data. View the Data Fetching section for more information.

Next Steps

App Router › ... › Routing

Layouts and Templates

Create your first shared layout in Next.js.

App Router › ... › Routing

Linking and Navigating

Learn how navigation works in Next.js, and how to use the Link Component and `useRouter` hook.

Previous
‹ **Defining Routes**

Next
Layouts and Templates ›

Was this helpful? 😆 🙂 🙁 😭

▲**Vercel**

Resources

Docs

Learn

Showcase

Blog

More

Next.js Commerce

Contact Sales

GitHub

Releases

About Vercel

Next.js + Vercel

Open Source Software

GitHub

X

Legal

Privacy Policy

Analytics   Telemetry

Next.js Conf   Governance

Previews

Subscribe to our newsletter

Stay updated on new releases and
features, guides, and case studies.

you@domain.com   Subscribe

© 2024 Vercel, Inc.