Linking Between Pages

This guide covers how to link between pages in a Gatsby site.

The Gatsby link component

The Gatsby `<Link />` component is for linking between pages within your site. For external links to pages not handled by your Gatsby site, use the regular HTML `<a>` tag.

Using the `<Link />` component for internal links

Here's an example of creating a link between two pages in a Gatsby site.

Open a page component (e.g. `src/pages/index.js`) in your Gatsby site. Import the `Link` component from Gatsby, which makes it available in the component. Add a `<Link />` component below the header, and give it a `to` property with the value of `"/contact/"` for the pathname:

```
import React from "react"
import { Link } from "gatsby"

export default function Home() {
  return (
    <div>
      <Link to="/contact/">Contact</Link>
    </div>
  )
}
```

The above code will add a link to the contact page, automatically rendered in HTML as `<a href="/contact/">` but with added performance benefits. The link's value is based off of the page's filename which in this case would be `contact.js`.

> **Note:** the value `"/"` for the `to` property will take users to the home page.

Using relative links in the `<Link />` component

Relative links are ones where the `to` property doesn't start with a `/`. These behave slightly differently from relative links in `<a>` tags, and instead follow the behavior of @reach/router. This avoids confusion with trailing slashes by ignoring them entirely and treating every page as if it were a directory. For example, if you are on either `/blog/my-great-page` or `/blog/my-great-page/` (note the trailing slash), a link to `../second-page` will take you to `/blog/second-page`. Similarly, if you are on `/blog` or `/blog/` a link to `hello-world` will take you to `/blog/hello-world`.

Using `<a>` for external links

If you are linking to pages not handled by your Gatsby site (such as on a different domain), you should use the native HTML `<a>` tag instead of Gatsby Link.

Additionally, if you wish to have an external link open in new window using the `target` attribute, use the `rel` attribute as seen below to avoid a vulnerability in which the referrer page can be replaced dynamically in JavaScript with a different page:

```
import React from "react"

export default function Home() {
  return (
    <div>
      <a href="https://example.com" target="_blank" rel="noopener noreferrer">
        External link
      </a>
    </div>
  )
}
```

It is also recommended to include a visual icon or some kind of indicator differentiating external links from internal ones.

Other resources

- For the complete example of how to link between pages, see Part One in the Tutorial
- Check out more detail on routing in Gatsby in the API doc for Gatsby Link.