


Gatsby v2.1.0 introduces `useStaticQuery`, a new Gatsby feature that provides the ability to use a [React Hook](#) to query with GraphQL at *build time*.

Just like the [StaticQuery](#) component, it allows your React components to retrieve data via a GraphQL query that will be parsed, evaluated, and injected into the component. However, `useStaticQuery` is a hook rather than a component that takes a render prop!

In this guide, you will walk through an example using `useStaticQuery`. If you're not familiar with static queries in Gatsby, you might want to check out [the difference between a static query and a page query](#).

How to use useStaticQuery in components

 You'll need React and ReactDOM 16.8.0 or later to use `useStaticQuery`.

 `npm install react@^16.8.0 react-dom@^16.8.0`

`useStaticQuery` is a React Hook. All the [Rules of Hooks](#) apply.

It takes your GraphQL query and returns the requested data. That's it!

Basic example

Let's create a `Header` component that queries for the site title from `gatsby-config.js`:

```
import React from "react"
import { useStaticQuery, graphql } from "gatsby"

export default function Header() {
  const data = useStaticQuery(graphql`
    query HeaderQuery {
      site {
        siteMetadata {
          title
        }
      }
    }
  `)

  return (
    <header>
      <h1>{data.site.siteMetadata.title}</h1>
    </header>
  )
}
```

Composing custom `useStaticQuery` hooks

One of the most compelling features of hooks is the ability to compose and re-use these blocks of functionality.

`useStaticQuery` is a hook. Therefore, using `useStaticQuery` allows us to compose and re-use blocks of reusable functionality. Perfect!

A classic example is to create a `useSiteMetadata` hook which will provide the `siteMetadata` to be re-used in any component. It looks something like:

```
import { useStaticQuery, graphql } from "gatsby"

export const useSiteMetadata = () => {
  const { site } = useStaticQuery(
    graphql`
      query SiteMetaData {
        site {
          siteMetadata {
            title
            siteUrl
            headline
            description
            image
            video
            twitter
            name
            logo
          }
        }
      }
    `
  )
  return site.siteMetadata
}
```

Then import your newly created hook, like so:

```
import React from "react"
import { useSiteMetadata } from "../hooks/use-site-metadata"

export default function Home() {
  const { title, siteUrl } = useSiteMetadata()
  return <h1>welcome to {title}</h1>
}
```

Known Limitations

- `useStaticQuery` does not accept variables (hence the name "static"), but can be used in *any* component, including pages
- Because of how queries currently work in Gatsby, we support only a single instance of `useStaticQuery` in a file