

Using Typography.js

This guide covers how to use Typography.js in a Gatsby project.

Typography.js

Typography.js is a JavaScript library that allows you to explore the typographic design of your website and define beautiful custom and pre-existing typographic themes. It enables you to change the font on your website with ease. Typography.js currently maintains over 30 themes for you to use. You can also create your own custom font themes if no available themes fit your requirements. To use Typography.js in your project, you will be installing a Gatsby plugin and specifying a configuration object for Typography.js.

Installing the Typography.js plugin

Gatsby has the plugin `gatsby-plugin-typography` to integrate Typography.js into your project.

You can install the plugin and its peer dependencies into your project by running the following command:

```
npm install gatsby-plugin-typography react-typography typography
```

After the installation of the plugin is complete, navigate to your `gatsby-config.js` file located in the root of your project's directory and add the plugin to the configuration:

```
module.exports = {  
  plugins: [  
    // highlight-start  
    {  
      resolve: `gatsby-plugin-typography`,  
      options: {  
        pathToConfigModule: `src/utils/typography`,  
      },  
    },  
    // highlight-end  
  ],  
}
```

`gatsby-plugin-typography` takes two options for you to specify:

- `pathToConfigModule` (string): The path to the file where you export your `Typography.js` configuration.
- `omitGoogleFont` (boolean, default: `false`): By default, `Typography.js` includes a helper that makes a request to Google Font's CDN for fonts you need. You may want to use your own fonts, either by injecting fonts or using a CDN of your choosing. By setting `omitGoogleFont: true`, `gatsby-plugin-typography` will skip adding the font helper. Instead, you will have to include the appropriate fonts yourself - see [Adding a Local Font](#)

Creating the Typography configuration

Now that you have added the plugin, create the directory `src/utils/` if it does not already exist in your project and add a new file named `typography.js`. You will use this file to specify the `Typography.js` configuration and set this file to be the path for the `pathToConfigModule` option.

Inside the `typography.js` file you created, you define your website's typography configuration. A basic `typography.js` configuration looks like this:

```
import Typography from "typography"

const typography = new Typography({
  baseFontSize: "18px",
  baseLineHeight: 1.666,
  headerFontFamily: [
    "Avenir Next",
    "Helvetica Neue",
    "Segoe UI",
    "Helvetica",
    "Arial",
    "sans-serif",
  ],
  bodyFontFamily: ["Georgia", "serif"],
})

export default typography
```

If you're installing `Typography.js` into an existing Gatsby project you've started, you will need to delete all conflicting CSS font styles from your codebase in favor of your new `Typography.js` settings.

Font sizes of all elements in `Typography.js` grow and shrink in relation to the `baseFontSize` defined above. Try playing around with this value and see the visual difference it can make to your website.

To find or create a new typography theme, you can visit [Typography.js](#) to see a list of options.

Installing Typography.js themes

Typography.js has built in themes that can save time when defining your website's font styling. The Funston theme, maintained by Typography.js, is one of the built-in themes. To install the Funston theme from npm, run the command:

```
npm install typography-theme-funston
```

To use the theme, edit the `typography.js` file that you created before and inform Typography.js of the new configuration:

```
import Typography from "typography";  
// highlight-start  
+ import funstonTheme from 'typography-theme-funston'  
// highlight-end  
const typography = new Typography(  
  - {  
  -   baseFontSize: '18px',  
  -   baseLineHeight: 1.666,  
  -   headerFontFamily: ['Avenir Next', 'Helvetica Neue', 'Segoe UI', 'Helvetica', 'Arial'],  
  -   bodyFontFamily: ['Georgia', 'serif'],  
  - },  
  // highlight-start  
  + funstonTheme  
  // highlight-end  
);  
  
export default typography;
```

After completing the above steps, you can start the development server using the command `gatsby develop` and navigate to the local website `http://localhost:8000`. If all went well you should see the text on your website using the Funston typographic theme.

Note: If your fonts remains unchanged, remove all `font-family` calls in your CSS and check again.

To find more themes to install, check out the official [Typography.js](#) website.