Gutenberg and web fonts

> ## Note:
>
> This is another example of expectation management when working with Gutenberg and the full site editor.
>
> As currently implemented in the Gutenberg plugin, the API will only work with the Full Site Editor… it will not allow you to use the fonts you defined in `theme.json` to change the fonts for content inside the templates.
>
> I am still researching if it's possible to change the font for inline elements inside blocks or specific blocks in a page.

Gutenberg 12.8 introduced the web fonts API intro the Gutenberg plugin to give theme creators and customizers the ability to use web fonts in their themes.

It does so by introducing the `fontFace` property in the `typography` object. This is similar to how you define `@font-face` in CSS.

The `@font-face` attributes that I've seen used most often are:

`"fontFamily"` : The stack we will use for the font, including any fallback options.

`"fontWeight"` : One or two numbers indicating the numeric weight of the font. : If one number is used then it represents the absolute value of the font weight (for example 700) : If two numbers are used they represent the range of acceptable values of the font-weight attribute. This is commonly used in [variable fonts](#) : The values used in the two number value are font dependent. Not all fonts have the same ranges

`"fontStyle"` : Either `italic` or `normal` to indicate if the font is italics or not. : If the font makes available a `slnt` custom axis then you can use code like this: `oblique 14deg` and change the degree angle to what works with your font : Some fonts may use custom axes for italics including `slnt`. Those are not covered in the `@font-face` CSS attribute or the `fontFace` attribute we're discussing here.

"fontStretch" : The property uses single keyword value or a single percentage value : Numerical values are limited to 50% to 200% (inclusive). **Negative values are not allowed** : If you use a keyword value then the following values are allowed: `ultra-condensed`, `extra-condensed`, `condensed`, `semi-condensed` `normal`, `semi-expanded`, `expanded`, `extra-expanded`, `ultra-expanded` : It also allows for the `normal` value for the property

"src" : Gives the source to the file relative to the `theme.json` file and wrapped as a JSON object. : You can use any font format supported by your target browsers but see performance notes below

```json
{
  "typography": {
  "fontFamilies": [
    {
      "fontFamily": "-apple-system,BlinkMacSystemFont,\"Segoe UI\",Roboto
      "fontFamily": "Genev": "ahoma, Verdana, sans-serif",
      "slug": "gene": ""slug""
    },
    {
      "fontFamily": "Cambria, Ge": ""slug"if",
      "slug": "cambria-georgia"
 ": ""fontFamily"  "fontFamily": "Recursive, sans-se": ""slug"   "name": "
      "slug": "r": ""fontFamily": "Recursive, sans-serif",
      "name"tFamily": "Recursive"": ""slug"  "fontWeight": "300 ": ""font
   ": ""fontWeight"ay": "swap",
         "": ""fontStyle"     "file:./assets/fonts/recursive-latin-subse
  "file:./assets/fonts/recursive-latin-subset.woff2""fontDisplay": "swap",
         "src": [
            "file:./assets/fonts/recursive-latin-subset.woff2"
         ]
      }
    ]
  }
  ],
}
```

Performance Notes

These notes cover some basic aspects of web fonts, particularly format and size reduction.

What formats to use?

The four most used formats for fonts are:

- [TTF](#) (True Type Fonts)
- [OTF](#) (Open Type Fonts)
- [WOFF](#) (Web Open Font Format 1.0)
- [WOFF2](#) (Web Open Font Format 2.0)

The decision will depend on what browsers your users have on their devices. Most modern browsers support WOFF2, which is smaller than the other alternatives

subset your fonts

You can further reduce the site by subsetting a WOFF2 font. While the gains may be marginal, it may still be a good idea to do so.

I've documented the font compression and subsetting process in [Font formats for the web and converting from one to another](#)