Style Variations

- What are style variations?
- Adding custom style variations
- Style variations vs. child themes

Unlike most things here in the Global Settings and Styles documentation, style variations are not things you define within `theme.json`. Instead, they are “variations” to your existing `theme.json` file that you can offer to users.

A more accurate name for this feature might be *Global Settings and Styles Variations*. Or simply `theme.json` variations.

What are style variations?

Style variations are essentially alternative versions of `theme.json` that you can ship with your theme. They are custom-named JSON files that are stored in your theme’s `/styles` folder. Any setting or style that you can add to `theme.json` can also be added to your style variation JSON file.

This lets your users pick and choose which variation they want to use on their site. In a way, they are “skins” for your theme.

For example, suppose you’ve created a restaurant theme and have kept the colors and typography pretty basic so that it covers a lot of different restaurant site designs. Further suppose that you wanted to offer more variety, variations on that initial design. You could create a style variation that caters more toward seafood restaurants with fun fonts and an ocean-oriented color palette. Or maybe you want to set the mood for coffee shops that might be running your theme.

That’s where style variations can really shine. You can bundle each of these alternative designs for your theme and let your users decide which is the best option for their site.

Here is a look at the style variations that are bundled with the default Twenty Twenty-Three theme:

WordPress Site Editor > Styles sub-screen, which is showing a grid of style variations with a red one

When a user selects a style variation, the JSON data is migrated to the site's database and stored as a user customization. This allows the data to overrule the theme's primary `theme.json` settings and styles.

Adding custom style variations

The style variations feature is relatively straightforward if you already understand how `theme.json` works, but there are a couple of differences.

The first difference between `theme.json` and style variations are their names and placement in your theme's folder structure. `theme.json` lives in the root of your theme folder and is considered the default variation. But custom variations must have a unique filename and be placed in the `/styles` folder.

Let's assume you've built that restaurant theme mentioned earlier in this document. Now you want to add a couple of variations named Swashbuckler (for that seafood design) and Latte (for the coffee shop design). This is how your theme files would be organized:

```
/your-theme-folder
├/styles
├├/latte.json
├├/swashbuckler.json
├/theme.json
```

Style variations are simply variations of `theme.json`, so you have full access to everything in the `theme.json` specification at your fingertips.

The second difference between `theme.json` and style variations is the variation title. You can configure this by adding the `title` property to your custom JSON files.

Building off the Latte variation example above, you would open your `/styles/latte.json` file and add it, as shown in this code snippet:

```
{
├"version": 2,
├"title": "Latte",
├"settings": {},
├"styles": {}
}
```

The `title` field is used to represent your variation in the user interface. It is not a required field (WordPress will fall back to your variation), but it does make for a nicer user experience.

Style variations vs. child themes

If you are familiar with the concept of child themes, which are covered in the Advanced Topics documentation, you may be wondering what the differences between them and style variations are.

The most obvious difference is that a style variation is limited to a single JSON file that overrides the primary `theme.json`, whereas a child theme can override anything from its parent theme. So it's probably better to look at the one area they are similar: the JSON file itself.

In a child theme, the `theme.json` simply overrides its parent's `theme.json` file. In a style variation—and this is where the major difference occurs—the variation's JSON file overrides the `theme.json` file and **its data is saved to the database**.

Once a user selects a style variation of a theme, everything in the variation's JSON file is treated as a user customization. Essentially, WordPress stores that **initial** data in the same way as if the user had simply designed the colors, typography, spacing, etc. from the interface. This is an important distinction to make because it means that when you update a style variation in a future theme release, the user will not receive those changes if they have already saved the style variation.

It is possible for users to switch to a variation and switch back to the one they were using to get the update.

Style variations can be a great feature to add to your theme, but they have a specific use case. Sometimes child themes make more sense.

□□□ #2
□ □□□ □□□ 2026-08-29 15:36:32 UTC
□ □□□ □□□ 2026-08-29 15:36:32 UTC