

```
from rich.console import Console
from rich.style import Style
console = Console()

base_style = Style.parse("cyan")
console.print("Hello, World", style = base_style + Style(underline=True))
```

You can parse a style definition explicitly with the `parse()` method, which accepts the style definition and returns a `Style` instance. For example, the following two lines are equivalent:

```
style = Style(color="magenta", bgcolor="yellow", italic=True)
style = Style.parse("italic magenta on yellow")
```

## 3.3 Style Themes

If you reuse styles it can be a maintenance headache if you ever want to modify an attribute or color – you would have to change every line where the style is used. Rich provides a `Theme` class which you can use to define custom styles that you can refer to by name. That way you only need to update your styles in one place.

Style themes can make your code more semantic, for instance a style called "warning" better expresses intent than "italic magenta underline".

To use a style theme, construct a `Theme` instance and pass it to the `Console` constructor. Here's an example:

```
from rich.console import Console
from rich.theme import Theme
custom_theme = Theme({
    "info": "dim cyan",
    "warning": "magenta",
    "danger": "bold red"
})
console = Console(theme=custom_theme)
console.print("This is information", style="info")
console.print("[warning]The pod bay doors are locked[/warning]")
console.print("Something terrible happened!", style="danger")
```

### Note

style names must be lower case, start with a letter, and only contain letters or the characters ".","-","\_".

### 3.3.1 Customizing Defaults

The `Theme` class will inherit the default styles built-in to Rich. If your custom theme contains the name of an existing style, it will replace it. This allows you to customize the defaults as easily as you can create your own styles. For instance, here's how you can change how Rich highlights numbers:

```
from rich.console import Console
from rich.theme import Theme
console = Console(theme=Theme({"repr.number": "bold green blink"}))
console.print("The total is 128")
```

You can disable inheriting the default theme by setting `inherit=False` on the `rich.theme.Theme` constructor.