

JSF pages can be styled using cascading stylesheets (CSS), which define rules for styling page elements.

This involves both:

- **Planning** — this can be done without knowing the syntax of CSS files or how they are incorporated into JSF pages
- **Stylesheet Creation and Use** — this involves placing them into resource folders and accessing them from JSF pages.

### [Style Planning Cascading Style Sheets](#)

Style planning involves determining the visual characteristics of your application's web pages, including:

- Font style, size, and weight
- Foreground and background color
- Borders, etc

Once you know the desired contents of a JSF page you first need to identify and name the *types* of components that need styles.

For example, error messages are often rendered in styles that are different than the rest of the components on a page so they are easily recognized.

Style types should be named according to how they are *used*, not how they *appear*.

For example, a style type name for error messages should be something like **error** rather than **red-font**, even if the error message is in a red font, because an error message's appearance is subject to change.

Once identified, style type names can then be used as values for **class** attributes of plain HTML elements or **styleClass** attributes of JSF HTML elements.

Cascading styles sheet provide a mechanism for separating data in web pages from its presentation, similar to the separation provided by the **Model-View-Controller** design paradigm:

- Web designers can control the presentation of numerous data items with minimal effort
- The same data can be presented in several different visual forms

A cascading style sheet consists of a sequence of **CSS rules**, each defining a set of style attributes that can be applied to elements in a JSF page.

These rules are gathered together into files that can be used in several JSF pages.

The stylesheet file names use a **.css** suffix.

[CSS Rules](#) [Style Attributes](#) [Applying Styles](#) [References](#)

A CSS rule has the following form:

```
selector {  
sequence of declarations  
}
```

- The **selector** part identifies where the rule applies. The simplest and most useful form of a **selector** has the form **.class-name**.
- The **class-name** is then used as the value for the **class** attribute in plain HTML tags or the

**styleClass** attribute in JSF HTML tags.

Each **declaration** has the following form:

**keyword: value;**

- Each **keyword-value** pair specifies a style attribute to be applied at places specified by the **selector**.
- They can specify a variety of attributes such as colors, fonts, sizes, and spacing.

For example, here is a rule that styles the text used to prompt a user for input:

```
.prompt {  
    font-size: larger;  
    color: forestgreen;  
}
```

Here is a partial list of style keywords and legal values for style attributes. This just scratches the surface of what is available in CSS.

Keyword	Values
color	red, orange, yellow, green, blue, gray, black, white, ...
background-color	red, orange, yellow, green, blue, gray, black, white, ...
font-family	monospace, serif, sans-serif, cursive, Times, Helvetica, ...
font-size	small, medium, large, 14px, ...
font-weight	normal, bold, lighter, ...
font-style	normal, italic, oblique
border	thin solid blue, thick dotted green, 1em dashed yellow ...

Page components are styled by adding the following attributes to tags that can be styled.

- For unprefixed (HTML) tags: **class="class-name"**. Example:

```
<h1 class="header"> ... </h1>
```

- For **h:** prefixed tags: **styleClass="class-name"**. Example:

```
<h:outputText styleClass="loginError" ... />
```

- Some **h:** prefixed tags have multiples styles that depend on the values of the tag attributes. For these tags there are multiple style attributes whose names depend on the tag. Example:

```
<h:message errorClass="error" warnClass="warn" ... />
```

The ***class-names*** should be names used in stylesheet rule selectors.

- Eric A. Meyer, "Cascading Style Sheets: The Definitive Guide", O'Reilly, 2004.
- [CSS 2.1 Specification \(W3C\)](#)
- [CSS Tutorial \(www.w3schools.com\)](http://www.w3schools.com)