

# Appearance

Exploring the various ways to customize the button's appearance.

Let's face it, buttons are essential to today's user interfaces. As a result, developers often desire flexibility in how their buttons appear. A toolbar button should and behave slightly differently than a push button. Sometimes we want buttons to stand out, other times we want them subtle. The SRP Button Control gives users far more flexibility in how their button's appear than the standard OpenInsight button.

## Style

The [Style](#) property plays the largest part in how your button looks and behaves. The goal of this property is to establish the button's look and feel in one call. Note that this property does affect behavior. For instance, any toolbar style prevents the button from getting focus. See the [Style](#) property for details.

## Content

After establishing the button's style, the next step is to add content. For most purposes, the [Caption](#) and [Icon](#) properties will do the job. Using these properties you can have one caption, one icon, or both centered on the button face. You can customize the caption's appearance by altering the [Font](#) and [ForeColor](#) properties. The [TransparentColor](#) property gives you fine tuned control over the icon rendering.

For more complex content, see the [CaptionList](#) and [IconList](#) properties, which allows you to place as many icons and captions as you please.

## Background

If the [Style](#) property doesn't get you all the way there, you can use the [Background](#) property to set images for the button's background. The property allows you to add an image to be used as the button's background for each state of the button.

You can also opt to have no background for your button by setting the [BackVisible](#) property. This is useful for subtle buttons that are composed only of an icon or caption.