

# Using Themes (IIS only)

The information in this chapter applies only to Microsoft Internet Information Services (IIS).

Themes are used to define the output style for the Natural application. Some themes are already delivered in the installation package.

Themes are standard functionality in ASP.NET 2.0. They must be contained in the *App\_Themes* folder. For each theme, the *App\_Themes* folder contains a subfolder. For example, the theme with the name "3270Thema" is contained in the folder *3270Theme*.

Each subfolder which is provided for a theme must contain the following files:

- *SkinFile.skin* (see below)
- *StyleSheet.css* (see below)

This chapter covers the following topics:

- The Skin File
  - The Style Sheet
- 

## The Skin File

The skin file (*SkinFile.skin*) contains entries for Natural elements (such as fields) in the following form:

```
<asp:Label SkinId="Red" runat="server" forecolor="red" />
<asp:Label SkinId="Green" runat="server" forecolor="green" />
<asp:Label SkinId="Blue" runat="server" forecolor="blue" />
...
```

The table below shows the minimum settings that are required in the skin file to map the Natural screen in the Natural Web I/O Interface. This table contains the following columns:

- **Natural Attributes**  
The attributes that can be set in Natural.
- **Output Fields**  
The Natural output fields for text.
- **Input Fields**  
The Natural fields in which you can enter data.
- **Read-only Input Fields**  
When a window is displayed on a Natural screen, the input fields in the main screen are set to read-only. The input fields of the windows that are lying behind the top window are also displayed as read-only.

Natural Attributes	Output Fields	Input Fields	Read-only Input Fields
Red	<code>&lt;asp:Label runat="server" SkinId="Red" /&gt;</code>	<code>&lt;asp:TextBox runat="server" SkinId="Red" /&gt;</code>	<code>&lt;asp:TextBox runat="server" SkinId="RedReadOnly" /&gt;</code>
Green	<code>&lt;asp:Label runat="server" SkinId="Green" /&gt;</code>	<code>&lt;asp:TextBox runat="server" SkinId="Green" /&gt;</code>	<code>&lt;asp:TextBox runat="server" SkinId="GreenReadOnly" /&gt;</code>
Blue	<code>&lt;asp:Label runat="server" SkinId="Blue" /&gt;</code>	<code>&lt;asp:TextBox runat="server" SkinId="Blue" /&gt;</code>	<code>&lt;asp:TextBox runat="server" SkinId="BlueReadOnly" /&gt;</code>
Pink	<code>&lt;asp:Label runat="server" SkinId="Pink" /&gt;</code>	<code>&lt;asp:TextBox runat="server" SkinId="Pink" /&gt;</code>	<code>&lt;asp:TextBox runat="server" SkinId="PinkReadOnly" /&gt;</code>
Yellow	<code>&lt;asp:Label runat="server" SkinId="Yellow" /&gt;</code>	<code>&lt;asp:TextBox runat="server" SkinId="Yellow" /&gt;</code>	<code>&lt;asp:TextBox runat="server" SkinId="YellowReadOnly" /&gt;</code>
Turquoise	<code>&lt;asp:Label runat="server" SkinId="Turquoise" /&gt;</code>	<code>&lt;asp:TextBox runat="server" SkinId="Turquoise" /&gt;</code>	<code>&lt;asp:TextBox runat="server" SkinId="TurquoiseReadOnly" /&gt;</code>
White	<code>&lt;asp:Label runat="server" SkinId="White" /&gt;</code>	<code>&lt;asp:TextBox runat="server" SkinId="White" /&gt;</code>	<code>&lt;asp:TextBox runat="server" SkinId="WhiteReadOnly" /&gt;</code>
Black	<code>&lt;asp:Label runat="server" SkinId="Black" /&gt;</code>	<code>&lt;asp:TextBox runat="server" SkinId="Black" /&gt;</code>	<code>&lt;asp:TextBox runat="server" SkinId="BlackReadOnly" /&gt;</code>
Intensified	<code>&lt;asp:Label runat="server" SkinId="Intensified" /&gt;</code>	<code>&lt;asp:TextBox runat="server" SkinId="Intensified" /&gt;</code>	<code>&lt;asp:TextBox runat="server" SkinId="IntensifiedReadOnly" /&gt;</code>
No attribute	<code>&lt;asp:Label runat="server" SkinId="Normal" /&gt;</code>	<code>&lt;asp:TextBox runat="server" SkinId="Normal" /&gt;</code>	<code>&lt;asp:TextBox runat="server" SkinId="NormalReadOnly" /&gt;</code>

The information in the cells of the above table shows the settings of the ASP.NET web server controls. ASP.NET defines a lot of web server controls. The attribute `runat="server"` indicates that the web controls are interpreted at the server site.

The following controls are used in the default skin files:

Control	Description
<code>asp:Label</code>	A text output field.
<code>asp:TextBox</code>	A text input field.
<code>asp:Table</code>	A table where the field elements are the PF key buttons.
<code>asp:Button</code>	A PF key button.
<code>asp:Image</code>	An image link.
<code>asp:Panel</code>	A container for other web controls.

In addition to defining colors as shown above, the SkinId is also used to define the following:

Natural Element	Description	ASP.NET Skin File
Message line	Information text line in the Natural screen.	<asp:Label runat="server" SkinId="messageline" />
Child window	The child windows of a Natural application.	<asp:Panel runat="server" SkinId="WindowPanel" />
Child window title	The title of a Natural child window.	<asp:TextBox runat="server" SkinId="WindowTitle" />
Child window shadow	The child window is shown with a shadow.	<asp:Panel runat="server" SkinId="WinShadow" />
PF key table	The row of PF key buttons is defined in a table.	<asp:Table runat="server" SkinId="PFKeys" />
PF key button	The definition of the PF keys.	<asp:Button runat="server" SkinId="Pfkey" />
Head picture	Picture displayed at the top of the page.	<asp:Image runat="server" SkinId="headpic" />

## The Style Sheet

In the style sheet (*StyleSheet.css*), you can define styles according to the CSS standard. See <http://www.w3.org/Style/CSS/>.

The following is an example style sheet which defines the background color and font of the web page:

```
body
{
background-color: transparent;
font-family: Courier New, Monospace;
}
```

### Note:

If you define the font size in a CSS file, this has no influence on the screen size. If the font size is changed, only smaller text is displayed.