articles Q&A forums lounge

Search for articles, questions, tips

ASP.NET: Themes

lonewolf3082, 5 Jun 2008

★★★★☆ 3.67 (7 votes) Rate this:

Working wtih themes in ASP.NET.

Download sample - 7.41 KB

Introduction

Using themes is a cool and easy way to create a consistent look and feel across a page or an entire website. By using themes, you can easily customize your server controls with predefined looks that come with the .NET Framework, or you can create your own themes for your own look.

Themes

Themes are a way to counter the problems faced when creating a layout for server controls and giving them the same look and feel throughout the entire application, with as little effort as possible. Default or Global themes are contained in a special folder inside the framework, and can be declared in the source as well as class files. Custom made themes are saved inside the predefined "*App_Themes*" folder inside ASP.NET applications, making them easier to manage and use according to your needs. The essential part of themes are skin files with the *.skin* extension. Besides skin files, a theme can be composed of styles sheet *.css* files as well as images for added support for the layout of the website.

Skins

A skin file has the file name extension *.skin*, and contains property settings for individual controls such as `Button`, `Label`, `TextBox`, or `Calendar`. Control skin settings are like the control markup itself, but contain only the properties you want to set as part of the theme.

Hide Copy Code

```
<asp:button runat="server" BackColor="lightblue" ForeColor="black" />
```

You create *.skin* files in the *Theme* folder. A *.skin* file can contain one or more control skins for one or more control types. You can define skins in a separate file for each control, or define all the skins for a theme in a single file.

There are two types of control skins: default skins and named skins:

- A default skin automatically applies to all controls of the same type when a theme is applied to a page. A control skin is a default skin if it does not have a `SkinID` attribute. For example, if you create a default skin for a `Calendar` control, the control skin applies to all `Calendar` controls on pages that use the theme. (Default skins are matched exactly by control type, so that a `Button` control skin applies to all `Button` controls, but not to `LinkButton` controls or to controls that derive from the `Button` object.)
- A named skin is a control skin with a `SkinID` property set. Named skins do not automatically apply to controls by type. Instead, you explicitly apply a named skin to a control by setting the control's `SkinID` property. Creating named skins allows you to set different skins for different instances of the same control in an application.

Global Themes

Built-in default themes are stored under the installation path of the .NET Framework:

Hide Copy Code

```
%SystemRoot%\Microsoft.NET\Framework\VX.X.XXXX\ ASP.NETClientFiles\Themes\
```

The actual name of the subdirectory labeled *vX.X.XXXX* changes according to the build of ASP.NET. Themes defined in this path are visible to all applications running on the machine. You can also create your own global theme by saving it in a subfolder of the *\Themes* folder in the above directory.

Creating Page Themes

1. In the Solution Explorer, right-click on the web site name and point to **Add ASP.NET** and click **Themes**.
2. Visual Studio will create a *App_Themes* folder automatically.
3. Create a subfolder of the *App_Themes* folder and name it accordingly.
4. Add **Skins, Cascading Style Sheets,** and **Images** as needed.

Adding a Skin file to a Page Theme

1. In the Solution Explorer. right-click the name of the theme and click **Add New Item**.
2. In the **Add New Item** dialog box, click **Skin File**.
3. Type the name of the *.skin* file in the name box.
4. In the *.skin* file, add the control definition using a declarative syntax, only include properties you want to set for the theme. The definition must include the `runat="server"` attribute and must *not* include the `ID=""` attribute.

Hide Copy Code

```
<asp:Button runat="server"

  BackColor="black"

  ForeColor="green"

  Font-Name="Arial"

 Font-Size="10pt" />
```

You can create as many or as few *.skin* files in the theme folder, but typically, you would only create one per control. You can define only one default **Skin** per control. If you want more, use the `SkinID` attribute in the skin's control declaration to create named **Skins** for the same control.

Hide Copy Code

```
<asp:Label runat="server" ForeColor="#585880"

   Font-Size="0.7em" Font-Names="Verdana"

   SkinID="LabelHeader" />

<asp:Label runat="server" ForeColor="#585980"

   Font-Size="0.6em" Font-Names="Arial"

   SkinID="LabelFooter" />
```

Adding Cascading Style Sheets to a theme is the same as adding a skin, except in the **Add New Item** dialog box, you select **Style Sheet**.

Applying a Theme to a Web Site

1. In the application's *web.config* file, set the `<pages>` element to the name of the theme, either page or global.

   Hide Copy Code

   ```
   <configuration>
   <system.web>
   <pages theme="ThemeName" />
   </system.web>
   </configuration>
   ```

   Or:

2. Set a style sheet theme to be subordinate to the local control properties, and set the `styleSheetTheme` attribute instead.

   Hide Copy Code

   ```
   <configuration>
   <system.web>
   <pages styleSheetTheme="ThemeName" />
   </system.web>
   </configuration>
   ```

Applying a Theme to an Individual Page

Set the Theme or StyleSheetTheme attribute of the @Page directive to the name of the theme.

Hide Copy Code

```
<%@ Page Theme="ThemeName" %>
    <%@ Page StyleSheetTheme="ThemeName" %>
```

Applying a Named Skin to a Control

Set the control's SkinID property.

Hide Copy Code

```
<asp:Calendar runat="server" ID="DateSelector" SkinID="LargeCalendar" />
```

Conclusion

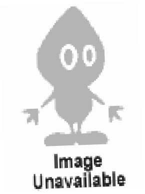
In conclusion, I hope this information was helpful to you. Themes are a nice way to create a consistent look and feel across websites quickly and easily.

License

Share

EMAIL TWITTER

About the Author



lonewolf3082

United States

No Biography provided

You may also be interested in...

Designing For DevOps

SAPrefs - Netscape-like Preferences Dialog

Themes and Skins in ASP.NET 2.0

Generate and add keyword variations using AdWords API

Themes

Window Tabs (WndTabs) Add-In for DevStudio