Animating in Response To User Interaction

Christian Wenz

Overview

The Animation control in the ASP.NET AJAX Control Toolkit is not just a control but a whole framework to add animations to a control. The animations can start automatically or may be triggered by user interaction, e.g. by clicking with the mouse.

Steps

First of all, include the **ScriptManager** in the page; then, the ASP.NET AJAX library is loaded, making it possible to use the Control Toolkit:

```
<asp:ScriptManager ID="asm" runat="server" />
```

The animation will be applied to a panel of text which looks like this:

```
<asp:Panel ID="panelShadow" runat="server" CssClass="panelClass">
  ASP.NET AJAX is a free framework for quickly creating a new
   generation of more efficient,
  more interactive and highly-personalized Web experiences that
   work across all the
  most popular browsers.<br />
  ASP.NET AJAX is a free framework for quickly creating a new
   generation of more efficient,
  more interactive and highly-personalized Web experiences that
   work across all the
  most popular browsers.<br />
  ASP.NET AJAX is a free framework for quickly creating a new
   generation of more efficient,
  more interactive and highly-personalized Web experiences that
   work across all the
  most popular browsers.<br />
</asp:Panel>
```

In the associated CSS class for the panel, define a nice background color and also set a fixed width for the panel:

```
<style type="text/css">
  .panelClass {background-color: lime; width: 300px;}
</style>
```

Then, add the **AnimationExtender** to the page, providing an **ID**, the **TargetControlID** attribute and the obligatory **runat="server"**:
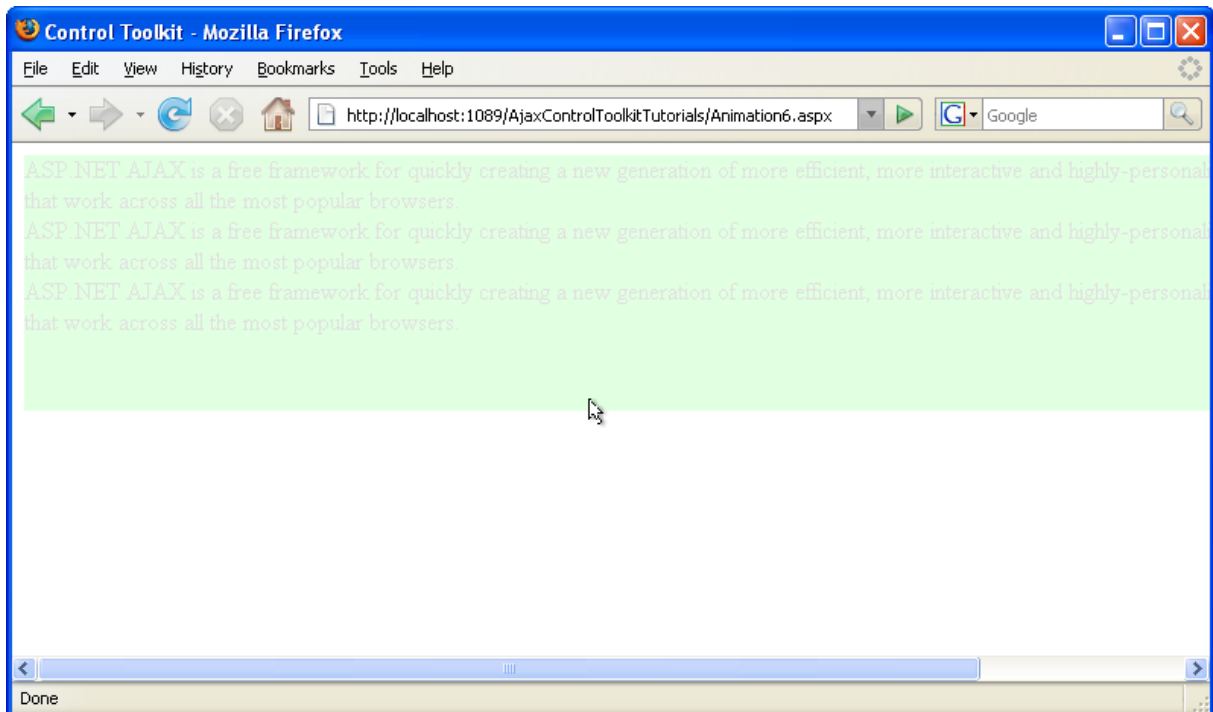
```
<ajaxToolkit:AnimationExtender ID="ae" runat="server"
  TargetControlID="Panel1">
```

Within the **<Animations>** node, there are five ways to start the animation via user interaction (the missing element is **<OnLoad>** which is executed once the whole page has been fully loaded):

- **<OnClick>** (mouse click on the control)
- **<OnHoverOut>** (mouse leaves the control)
- **<OnHoverOver>** (mouse hovers over a control, stopping the <**OnHoverOut**> animation)
- **<OnMouseOut>** (mouse leaves a control)
- **<OnMouseOver>** (mouse hovers over a control, not stopping the <**OnMouseOut**> animation)

In this scenario, **<OnClick>** is used. When the user clicks on the panel, it is resized and fades out at the same time.

```
<ajaxToolkit:AnimationExtender ID="ae" runat="server"
  TargetControlID="Panel1">
  <Animations>
    <OnClick>
      <Parallel>
        <FadeOut Duration="1.5" Fps="24" />
        <Resize Width="1000" Height="150" Unit="px" />
      </Parallel>
    </OnClick>
  </Animations>
</ajaxToolkit:AnimationExtender>
```


A mouse click starts the animation