

Silverlight 4

فهرست مطالب

فصل ۲۵ – تعامل با کدهای HTML صفحه در Silverlight	۵۲۷
مقدمه	۵۲۷
معرفی کلاس BrowserInformation	۵۲۷
تنظیم خواص اشیاء HTML به کمک Silverlight	۵۳۰
عکس العمل نشان دادن به رخدادهای اشیاء HTML در Silverlight	۵۳۱
فراخوانی یک تابع JavaScript از طریق کدهای Silverlight	۵۳۲
فراخوانی یک متد تعریف شده در کدهای Silverlight از طریق کدهای JavaScript	۵۳۳

چاپ عمومی غیر رایگان این مطالب بدون مجوز کتبی از طرف نویسنده به هر نحوی غیرمجاز است.
انتشار این مطالب بر روی اینترنت و یا استفاده از آن به صورت مستقیم و یا غیر مستقیم در نشریات الکترونیکی با ذکر مأخذ بلا مانع است.

فصل ۲۵ – تعامل با کدهای HTML صفحه در Silverlight

مقدمه

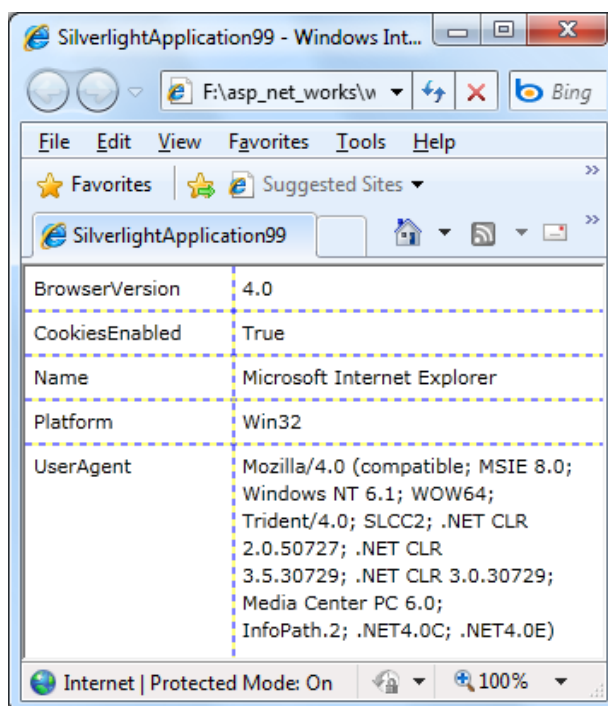
با کمک امکانات مهیای در Silverlight، می‌توان به خواص و اشیاء میزبان آن نیز دسترسی داشت. برای مثال Silverlight با استفاده از کلاس `HtmlDocument` می‌تواند با `HTML Document Object Model (DOM)` تعامل داشته باشد و یا از طریق کلاس `HtmlPage` اطلاعاتی را در اختیار کدهای جاوا اسکریپتی صفحه قرار دهد. این کلاس‌ها در فضای نام `System.Windows.Browser` قرار دارند. به این ترتیب ۷ کلاس مرتبط با صفحات و المان‌های HTML (`HtmlDocument`، `HtmlElement`، `HtmlEventArgs`، `HtmlObject`، `HtmlPage`، `HtmlPopupWindowOptions` و `HtmlWindow`) و ۴ کلاس جهت کار با کدهای اسکریپتی صفحه (`ScriptableObject`، `ScriptableMemberAttribute`، `ScriptableTypeAttribute` و `ScriptObject`) و `ScriptObjectCollection` (به علاوه دو کلاس `HttpUtility` و `BrowserInformation` برای دریافت اطلاعاتی از مرورگر جاری و کار با URLs، در اختیار برنامه نویس‌ها قرار خواهند گرفت.

معرفی کلاس `BrowserInformation`

در طی جدول بعد خواص مفید کلاس `BrowserInformation` را مرور خواهیم کرد. باید خاطر نشان کرد که هیچگاه تصمیمات خود را بر اساس رشته‌ی `UserAgent` ارائه شده‌ی توسط یک مرورگر قرار ندهید؛ زیرا به سادگی قابل تغییر است.

جدول ۱- خواص کلاس `BrowserInformation`

خاصیت	نوع	توضیح
<code>BrowserVersion</code>	<code>System.Version</code>	شماره نگارش مرورگر را بر می‌گرداند.
<code>CookiesEnabled</code>	<code>Bool</code>	آیا <code>Cookies</code> در مرورگر کاربر فعال است؟
<code>Name</code>	<code>String</code>	نام مرورگر را بر می‌گرداند.
<code>Platform</code>	<code>String</code>	سکوی کاری مرورگر را ارائه می‌دهد.
<code>UserAgent</code>	<code>String</code>	<code>UserAgent</code> متناظر با مرورگر را ارائه می‌دهد.



شکل ۱- یک نمونه خروجی کلاس BrowserInformation در ویندوز ۷ با IE8.

در مثال بعد قصد داریم خواص این کلاس را از طریق Binding نمایش دهیم (شکل ۱). کدهای XAML آنرا در ادامه ملاحظه خواهید نمود.

MainPage.xaml

```
<UserControl x:Class="SilverlightApplication99.MainPage"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    mc:Ignorable="d" d:DesignHeight="300" d:DesignWidth="400">
    <Grid x:Name="LayoutRoot"
        ShowGridLines="True"
        Background="White">
        <Grid.ColumnDefinitions>
            <ColumnDefinition Width="122" />
            <ColumnDefinition Width="*" />
        </Grid.ColumnDefinitions>
        <Grid.RowDefinitions>
            <RowDefinition Height="Auto" />
            <RowDefinition Height="Auto" />
            <RowDefinition Height="Auto" />
        </Grid.RowDefinitions>
    </Grid>
</UserControl>
```

```

        <RowDefinition Height="Auto" />
        <RowDefinition Height="*" />
    </Grid.RowDefinitions>
    <TextBlock Margin="5" Grid.Column="0"
        Grid.Row="0" Text="BrowserVersion" />
    <TextBlock Margin="5" Grid.Column="1"
        Grid.Row="0" Text="{Binding BrowserVersion}" />
    <TextBlock Margin="5" Grid.Column="0"
        Grid.Row="1" Text="CookiesEnabled" />
    <TextBlock Margin="5" Grid.Column="1"
        Grid.Row="1" Text="{Binding CookiesEnabled}" />
    <TextBlock Margin="5" Grid.Column="0"
        Grid.Row="2" Text="Name" />
    <TextBlock Margin="5" Grid.Column="1"
        Grid.Row="2" Text="{Binding Name}" />
    <TextBlock Margin="5" Grid.Column="0"
        Grid.Row="3" Text="Platform" />
    <TextBlock Margin="5" Grid.Column="1"
        Grid.Row="3" Text="{Binding Platform}" />
    <TextBlock Margin="5" Grid.Column="0"
        Grid.Row="4" Text="UserAgent" />
    <TextBlock Margin="5" Grid.Column="1"
        TextWrapping="Wrap"
        Grid.Row="4" Text="{Binding UserAgent}" />
</Grid>
</UserControl>

```

View برنامه اطلاعات خود را از طریق DataContext مقدار دهی شده به صورت ذیل دریافت می‌کند:

MainPage.xaml.cs

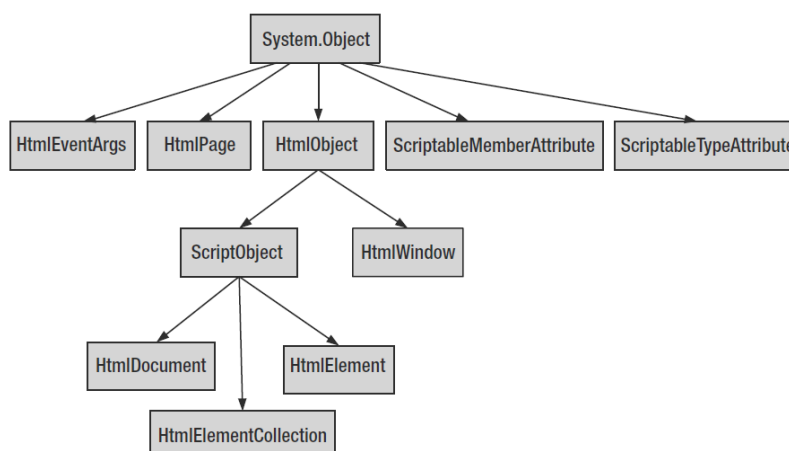
```

using System.Windows.Browser;

namespace SilverlightApplication99
{
    public partial class MainPage
    {
        public MainPage()
        {
            InitializeComponent();
            LayoutRoot.DataContext = HtmlPage.BrowserInformation;
        }
    }
}

```

حدود ۱۰ کلاس جهت تعامل با HTML و مرورگر در Silverlight وجود دارند که سلسله مراتب آن‌ها را در شکل ۲ می‌توان ملاحظه کرد. تمام متدها و خواص این کلاس‌ها static هستند.



شکل ۲- کلاس‌های مرتبط جهت تعامل با مرورگر

تنظیم خواص اشیاء HTML به کمک Silverlight

قصد داریم خاصیت innerHTML یک span تعریف شده در صفحه‌ی HTML را به ID مساوی SilverlightMessage، توسط یک برنامه‌ی Silverlight مقدار دهی نمائیم. برای این منظور یک برنامه‌ی جدید Silverlight را به همراه ASP.NET Web site آن آغاز کرده و سطر ذیل را در فایل aspx دربرگیرنده‌ی افزونه‌ی Silverlight اضافه کنید:

SilverlightApplicationTestPage.aspx

```

...
<span id="SilverlightMessage" />
...

```

اکنون کدهای روال رخداد گردان یک دکمه‌ی تعریف شده در برنامه‌ی Silverlight ما برای مقدار دهی خاصیت ذکر شده به صورت زیر خواهند بود:

MainPage.xaml.cs

```

using System.Windows;
using System.Windows.Browser;

namespace SilverlightApplication100
{
    public partial class MainPage
    {
        public MainPage()
        {
            InitializeComponent();
        }

        private void btnModifyDOM_Click(object sender,
            RoutedEventArgs e)

```

```

    {
        var span = HtmlPage.Document.GetElementById(
            "SilverlightMessage");

        if (span != null)
            span.SetAttribute("innerHTML",
                "From Silverlight Button!");
    }
}
}

```

در این مثال به کمک کلاس `HtmlPage.Document` و متد `GetElementById` آن، شیء مورد نظر را در صفحه‌ی HTML یافته و سپس ویژگی مورد نظر خود را توسط متد `SetAttribute`، تنظیم خواهیم نمود.

عکس العمل نشان دادن به رخدادهای اشیاء HTML در Silverlight

امکان عکس العمل نشان دادن به رخدادهای اشیاء HTML (برای مثال کلیک بر روی یک دکمه HTML) و گوش فرا دادن به آن‌ها نیز در کدهای Silverlight میسر است. برای مثال دکمه‌ی HTML ذیل را که در صفحه قرار گرفته است در نظر بگیرید:

SilverlightApplicationTestPage.aspx

```

...
<button id="HTMLButtonA">HTMLButtonA</button>
...

```

کدهای Silverlight زیر به رخداد کلیک شدن بر روی این دکمه عکس العمل نشان خواهند داد:

MainPage.xaml.cs

```

using System.Windows;
using System.Windows.Browser;

namespace SilverlightApplication100
{
    public partial class MainPage
    {
        public MainPage()
        {
            InitializeComponent();
            this.Loaded += MainPage_Loaded;
        }

        void MainPage_Loaded(object sender, RoutedEventArgs e)
        {
            HtmlElement button =
                HtmlPage.Document.GetElementById("HTMLButtonA");
            if (button != null)
                button.AttachEvent("onclick",

```

```

        (object o, HtmlEventArgs args) =>
        {
            MessageBox.Show("HTMLButtonA Clicked!");
        }
    );
}
}
}

```

به کمک این کدها، رخداد onclick دکمه را تحت نظر قرار خواهیم داد (با استفاده از متد AttachEvent) و در صورت لزوم می‌توان عکس العمل مناسبی را به کاربر ارائه داد.

فراخوانی یک تابع JavaScript از طریق کدهای Silverlight

فراخوانی یک متد جاوا اسکریپتی از طریق کدهای Silverlight به کمک متد ذیل قابل انجام است:
 HtmlPage.Window.Invoke(name, param object[] args)

برای مثال متد زیر را در یک صفحه‌ی HTML در نظر بگیرید:

SilverlightApplicationTestPage.aspx

```

...
<script type="text/javascript">
    function Foo(arg1, arg2) {
        alert(arg1 + ' - ' + arg2);
    }
</script>
...

```

جهت فراخوانی آن از طریق کدهای برنامه خواهیم داشت:

C#

```
HtmlPage.Window.Invoke("Foo", "param1", "param2");
```

آرگومان اول، نام متد و آرگومان‌های بعدی متناظر هستند با مقادیر پارامترهای متد جاوا اسکریپتی ذکر شده در برنامه.

لازم به ذکر است که متد Invoke، یک خروجی از نوع object نیز دارد که دقیقاً با خروجی تابع جاوا اسکریپتی فراخوانی شده، مساوی است. برای استفاده از این خروجی باید تبدیلات لازم صورت گیرد. برای مثال:

C#

```

// Returning a Double
double doubleValue = (double)HtmlPage.Window.Invoke("myJSMethod");

```



```
// Returning a String
string stringValue = (string)HtmlPage.Window.Invoke("myJSMethod");
```

حال اگر خروجی متد جاوا اسکریپتی ما یک object بود چکار باید کرد؟ در این حالت خروجی دریافتی را به نوع ScriptObject تبدیل کنید. سپس از متد GetProperty آن جهت دریافت مقادیر متناظر خواص آن شیء استفاده نمائید:

C#

```
ScriptObject myScriptObject =
    (ScriptObject)HtmlPage.Window.Invoke("myJSMethod");
double doubleValue = (double)myScriptObject.GetProperty("prop1");
double stringValue = (string)myScriptObject.GetProperty("prop2");
```

علاوه بر امکان فراخوانی متدهای سفارشی جاوا اسکریپتی، تعدادی متد پیش فرض نیز در کلاس HtmlPage.Window برای انجام یک سری امور متداول پیش بینی شده است؛ برای مثال متدهای آشنای Alert، Confirm، Navigate و NavigateToBookmark جهت نمایش پیغامی به کاربر و یا گشودن یک صفحه‌ی HTML جدید یا هدایت کاربر به قسمت مشخصی از صفحه‌ی جاری.

فراخوانی یک متد تعریف شده در کدهای Silverlight از طریق کدهای JavaScript

امکان فراخوانی متدهای Silverlight تحت شرایط ویژه‌ای در کدهای JavaScript صفحه وجود دارند. مراحل این عملیات به شرح بعد هستند:

۱. کلاس‌ها و متدهایی که قرار است در کدهای جاوا اسکریپتی فراخوانی شوند باید به ویژگی‌های ScriptableType و ScriptableMember همانند کلاس بعد مزین شوند:

FooClass.cs

```
using System.Windows;
using System.Windows.Browser;
namespace SilverlightApplication100
{
    [ScriptableType]
    public class FooClass
    {
        [ScriptableMember]
        public void Foo(string data)
        {
            MessageBox.Show("FromJavascript: " + data);
        }
    }
}
```

۲. سپس باید نام مستعاری را جهت دسترسی به این کلاس و متدهای آن مشخص نمائیم:

MainPage.xaml.cs

```
...
public MainPage()
{
    InitializeComponent();
    HtmlPage.RegisterScriptableObject("MyFooClass", new FooClass());
}
...
```

۳. به محلی که شیء افزونه‌ی Silverlight معرفی شده است مراجعه کرده و یک id را به آن نسبت دهید. برای مثال:

SilverlightApplicationTestPage.aspx

```
...
<object data="data:application/x-silverlight-2,"
        id="silverlightControl1"
        type="application/x-silverlight-2"
...

```

۴. اکنون کدهای اسکریپت و دکمه‌ی فراخوان آن به شرح بعد خواهند بود. ابتدا شیء افزونه‌ی Silverlight یافت شده و سپس از طریق آن نسبت به فراخوانی متدهای ذکر شده از طریق نام مستعار کلاس آن‌ها اقدام خواهیم کرد:

SilverlightApplicationTestPage.aspx

```
...
<script type="text/javascript">
    function CallFooFromJavascript() {
        var slObject =
            document.getElementById("silverlightControl1");
        slObject.Content.MyFooClass.Foo("Test1");
    }
</script>
...
<input type="button"
        value="CallFooFromJavascript"
        onclick="CallFooFromJavascript();" />
...
```

امکان استفاده از خروجی متدهای Silverlight نیز در کدهای JavaScript ایی صفحه به سادگی میسر است. اگر خروجی، از نوع‌های پایه‌ای مانند int و string و امثال آن است، مشکلی در استفاده از آن‌ها نخواهد بود. اما اگر خروجی متد فراخوانی شده یک شیء تعریف شده در کدهای .NET است، باید به طریق زیر عمل نمود:

C#

```
// Call the method and get the returned object
var obj = plugin.Content.myObject.Method1();
```

```
// Get the value of a property of the returned object  
var strName = obj.Name;
```

ابتدا متد مورد نظر فراخوانی شده و شیء حاصل دریافت می‌گردد. سپس از طریق این شیء می‌توان به خواص آن دسترسی یافت.