

بسم الله الرحمن الرحيم

## آموزشگاه تحلیل داده

تخصصی ترین مرکز برنامه نویسی و دیتابیس در ایران

## آموزش کدنویسی سمت سرویس گیرنده – ASP.NET

مدرس : مهندس افشین رفوآ

### آموزش کدنویسی سمت سرویس گیرنده – ASP.NET

کدنویسی سمت سرویس گیرنده در ASP.NET دارای دو جنبه می باشد:

**Client side scripts** – روی مرورگر اجرا شده و متعاقباً اجرای صفحه را تسریع می بخشد. برای مثال می توان به اعتبارسنجی داده سمت سرور (**client side data validation**) اشاره کرد که قادر است داده های نامعتبر را گرفته (catch) و بر اساس آن به کاربر هشدار دهد.

**Client side source code** - صفحات ASP.NET سوره کد سمت سرویس گیرنده را تولید می کنند. برای مثال، سورس کد HTML صفحه ی ASP.NET شامل تعدادی فیلد پنهان و مجموعه (بلوک) کدهای جاوا اسکریپت (که به صورت خودکار تزریق می شوند) می باشد که اطلاعاتی مانند **view-state** را نگه داشته یا کارهای دیگری انجام می دهد که به عملکرد صفحه کمک می کند.

### اسکریپت های سمت سرویس گیرنده (client side scripts)

کلیه ی کنترل های سرور ASP.NET امکان فراخوانی کدهای سمت سرویس گیرنده که توسط زبان های javascript و VBscript نوشته شده باشند را فراهم می کنند. برخی از کنترل های سرور ASP.NET با استفاده از برنامه های تحت وب که در سمت سرویس گیرنده و روی مرورگر کلاینت اجرا می شوند (client-side scripting)، پاسخ (response) را مستقیماً (بدون ارسال / بازگرداندن آن به سرور) به کاربرها ارائه می دهند، به عنوان نمونه می توان به کنترل های validation (اعتبارسنجی) اشاره کرد که در سمت سرویس گیرنده اجرا می شوند.

جدا از اسکریپت های ذکر شده، کنترل **button** دارای خاصیتی (**property**) به نام **OnClick** می باشد که به مجرد فشردن دکمه اجازه ی اجرا اسکریپت سمت کلاینت را می دهد.

کنترل های **HTML server** دارای رخدادهای (**event**) زیر می باشند. به محض فعال شدن این رخدادها اسکریپت های مربوطه نیز اجرا می شوند:

Event	Description
onblur	رخداد onblur زمانی اجرا می شود که کاربر این کنترل را رها کرده و کنترل دیگری را انتخاب کند.
onfocus	هنگامی فعال می گردد که کنترل مورد نظر توسط کاربر انتخاب شود.
onclick	این رخداد به محض اینکه کنترل مربوطه کلیک شود، اجرا می گردد.
onchange	زمانی فعال می گردد که مقدار کنترل مورد نظر تغییر پیدا کند.
onkeydown	هنگامی که کاربر یک دکمه را فشار می دهد، به مجرد آن رخداد onkeydown فعال می گردد.
onkeypress	این رخداد زمانی فعال می شود که کاربر یک کلید الفبا عددی ( alphanumeric key ) را فشار دهد.
onkeyup	این رخداد زمانی فعال می گردد که کاربر کلید مورد نظر را رها کند.
onmouseover	این رخداد هنگامی فعال می شود که کاربر مکان نمای موس را روی کنترل مربوطه قرار دهد.
onserverclick	به مجرد اینکه کنترل مورد نظر کلیک می شود، رخداد ServerClick کنترل مزبور فعال می گردد.

## Client Side Source Code (سورس کد سمت سرویس گیرنده)

همان طور که پیش تر توضیح داده شد صفحات ASP.NET عموماً در دو فایل نوشته می شوند:

markup file یا Content file

Code behind file

فایل Content دربردارنده ی **control tag** ها و **literal** های (به نشانه هایی در کد منبع برنامه گفته می شود که در زمان کامپایل به مقادیر مشخصی تفسیر می شوند) HTML و ASP.NET است که ساختار کلی صفحه را تشکیل (شکل) می دهند، فایل **code-behind** نیز دربردارنده ی تعریف کلاس (**class definition**) می باشد. به هنگام **run-time** (در زمان اجرا)، فایل **content** تجزیه (**parse**) شده و به یک کلاس **page** تبدیل می شود.

کلاس نام برده به همراه تعریف کلاس (**class definition**) موجود در فایل **code-behind** و همچنین کد تولید شده توسط سیستم (**system generated code**)، کد اجرایی (**executable code**) را بوجود می آورد. کد اجرایی مزبور در واقع تمامی داده های ارسال (**post**) شده را پردازش کرده و بر اساس آن پاسخ (**response**) را تولید می کند، سپس پاسخ را به سرویس گیرنده (کلاینت) می فرستد (**send back**).

کد نوشته شده برای یک صفحه ی ساده:

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Default.aspx.cs"
Inherits="WebApplication1._Default" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
  <title>Untitled Page
  </title>
</head>
<body>
  <form id="form1" runat="server">
    <div>
      <asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>
      <asp:Button ID="Button1" runat="server" OnClick="Button1_Click" Text="Click" />
    </div>
    <hr />
    <h3>
      <asp:Label ID="Msg" runat="server" Text=""> </asp:Label>
    </h3>
  </form>
</body>
</html>
```

هنگامی که صفحه ی فوق روی صفحه ی مرورگر بار گذاری می شود، شما می توانید با راست کلیک و انتخاب گزینه ی

**View Source** صفحه ی HTML که توسط ASP.NET runtime به مرورگر ارسال شده را مشاهده کنید:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head><title>
    Untitled Page
</title></head>
<body>
    <form name="form1" method="post" action="Default.aspx" id="form1">
<div>
<input type="hidden" name="__VIEWSTATE" id="__VIEWSTATE"
value="/wEPDwUKLTixOTA3MjcyNWRkuhm7BZdSLAsPsRlEOfamUeGZotA=" />
</div>
<div>
<input type="hidden" name="__VIEWSTATEGENERATOR" id="__VIEWSTATEGENERATOR" value="CA0B0334" />
<input type="hidden" name="__EVENTVALIDATION" id="__EVENTVALIDATION"
value="/wEWAwKSjJLJDwLs0bLrBgKM54rGBm8B3L+ojLrWeq/2/MAi0iEYxmgj" />
</div>
        <div>
            <input name="TextBox1" type="text" id="TextBox1" />
            <input type="submit" name="Button1" value="Click" id="Button1" />
        </div>
        <hr />
        <h3>
            <span id="Msg"></span>
        </h3>
    </form>
```

اگر با دقت به کد فوق نظاره کنید متوجه می شوید که دو تگ **<<div>** اول دربردارنده ی فیلدهای پنهان هستند. این فیلدهای پنهان اطلاعات **view state** و **validation** را در خود ذخیره می کنند.