

شاید بعضی از سایت‌ها را دیده باشید که در حین ثبت نام، پس از وارد کردن یک نام کاربری و سپس مشغول شدن به پر کردن فیلد کلمه عبور، در قسمت نام کاربری شروع به جستجو در مورد آزاد بودن نام کاربری درخواستی می‌کنند یا نمونه‌ای دیگر، فرم پرداخت الکترونیکی بانک سامان. پس از اینکه شماره قبض را وارد کردید، بلافاصله بدون ریفرش صفحه به شما پیغام می‌دهد که این شماره معتبر است یا خیر. امروز قصد داریم این قابلیت را با استفاده از کتابخانه Ajax مجموعه jQuery در ASP.Net پیاده سازی کنیم (بدون استفاده از ASP.Net Ajax مایکروسافت). ابتدا سورس کامل را ملاحظه نمایید:

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="AjaxTest.aspx.cs"
Inherits="testWebForms87.AjaxTest" %>
<!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>jQuery Ajax Text</title>

<script src="jquery.min.js" type="text/javascript"></script>
<script type="text/javascript">
    $(document).ready(function() {
        $("#<%= TextBox1.ClientID %>").blur(function(event) {
            $.ajax({
                type: "POST",
                url: "AjaxTest.aspx/IsUserAvailable",
                data: "{ 'username': '" + $('#<%= TextBox1.ClientID %>').val() + "'",
                contentType: "application/json; charset=utf-8",
                dataType: "json",
                success: function(msg) {
                    $('#valid').html("<img src='ajaxImages/waiting.gif' alt='لطفا کمی تامل کنید'>");
                    var delay = function() {
                        AjaxSucceeded(msg);
                    };
                    setTimeout(delay, 2000); //remove this
                },
                error: AjaxFailed
            });
        });
    });
    function AjaxSucceeded(result) {
        if (result.d == true)
            $('#msg').html("<img src='ajaxImages/available.gif' alt='نام کاربری درخواستی موجود است'>");
        else
            $('#msg').html("<img src='ajaxImages/taken.gif' alt='متاسفانه نام کاربری مورد نظر پیشتر شده است'>");
    }
    function AjaxFailed(result) {
        alert(result.status + ' ' + result.statusText);
    }
</script>

</head>
<body>
<form id="form1" runat="server">
<div>
    user name:
    <asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>
    <span id="msg"></span>
    <br />
    pass:
    <asp:TextBox ID="TextBox2" TextMode="Password" runat="server"></asp:TextBox>
</div>
<!-- preload -->
<div style="display: none">
    
    
</div>
```

```

</div>
</form>
</body>
</html>
```

```
using System;
using System.Web.Services;

namespace testWebForms87
{
    public partial class AjaxTest : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
        }

        [WebMethod]
        public static bool IsUserAvailable(string username)
        {
            // این مورد را با خواندن اطلاعات از دیتابیس می‌شود تعویض کرد
            return username != "test";
        }
    }
}
```

توضیحات:

همانطور که ملاحظه می‌کنید صفحه‌ی ASP.Net ما بسیار ساده است و از دو تکست باکس استاندارد تشکیل می‌شود، به همراه تصاویر مربوط به Ajax که یک سری تصاویر ساده چرخان [معروف](#) منتظر بمانید، یافت شد یا موجود نیست می‌باشند. این تصاویر در یک div مخفی (display: none) در صفحه قرار گرفته‌اند و در هنگام بارگذاری صفحه، این‌ها نیز بارگذاری شده و حاضر و آماده خواهند بود. بنابراین هنگام استفاده از آن‌ها، کاربر تاخیری را مشاهده نخواهد کرد. همچنین یک span با id مساوی msg را هم پس از تکست باکس اضافه کرده‌ایم تا تصاویر مربوط به رخدادهای Ajax را با استفاده از توانایی‌های jQuery به آن اضافه کنیم.

اسکرپت Ajax ما با در اختیار گرفتن روال رخداد گردان [blur](#) شیء textBox1 شروع می‌شود. همانطور که در [مقالات پیشین](#) سایت نیز ذکر شد، روش صحیح دریافت ID یک کنترل ASP.Net در کدهای سمت کلاینت جاوا اسکریپتی، بر اساس خاصیت ClientID آن است که در اولین سطر کدهای ما مشخص است (زیرا در ASP.Net نام و ID یک کنترل در هنگام رندر شدن به همراه ID کنترل‌های دربرگیرنده آن نیز خواهد بود، بنابراین بهتر است این مورد را داینامیک کرد).

کار بررسی موجود بودن نام کاربری (یا مثلاً یک شماره قبض و امثال آن) توسط WebMethod ایی به نام IsUserAvailable در code behind صفحه انجام می‌شود که پیاده سازی آن را ملاحظه می‌کنید. بدیهی است در این مثال ساده، تنها نام کاربری از پیش رزرو شده، کلمه‌ی test است و در یک کد واقعی این مورد با مقایسه‌ی نام کاربری با اطلاعات موجود در دیتابیس باید صورت گیرد (و حملات تزریق اس کیوال را هم فراموش نکنید. برای رهایی از آن‌ها " [حتما](#) " باید از پارامترهای ADO.Net استفاده کرد و گر نه کد شما مستعد به این نوع حملات خواهد بود).

سؤال: چرا از web method استفاده شد و همچنین چرا این متد static است؟
زمانیکه یک متد با کلمه کلیدی static مشخص می‌شود حالت [state less](#) پیدا می‌کند یعنی مستقل از وهله‌ی کلاس عمل می‌کند. در این حالت نیازی به ارسال ViewState نبوده (بنابراین در کوئری مورد نظر ما بسیار بهینه و سبک عمل می‌کنند) و همچنین نیازی به ایجاد یک وهله‌ای از کلاس صفحه‌ی ما نیز نخواهد بود. برای توضیحات بیشتر [به این](#) مقاله مراجعه نمائید. (به صورت خلاصه، دلیل اصلی، کارایی بالا و بهینه بودن این روش در این مساله ویژه است و در ASP.Net Ajax مایکروسافت به صورت گسترده‌ای در پشت صحنه مورد استفاده قرار می‌گیرد)

استفاده از ویژگی WebMethod عملکرد صفحه‌ی ما را شبیه به یک وب سرویس خواهد کرد و امکان دسترسی به آن در متدهای استاندارد POST به صورت ارسال دیتا به آدرس WebService.asmx/WebMethodName خواهد بود. [یک مثال](#) ساده و عملی

بررسی تابع Ajax بکار رفته:

این تابع هنگام فراخوانی رخداد blur تکست باکس ما (مطابق کد فوق) فراخوانی می‌شود. ساختار ساده‌ای دارد که به شرح زیر است:

"type: "POST" نحوه‌ی ارسال داده را به متد وب سرویس ما مشخص می‌کند.

url: "AjaxTest.aspx/IsUserAvailable" اطلاعات پست شده، به صفحه‌ی AjaxTest.aspx و وب متد IsUserAvailable ارسال خواهد شد

data: '{username: "' + \$('#<%= TextBox1.ClientID %>').val داده‌ای که به آرگومان username ما ارسال می‌شود، همان مقدار تایپ شده در TextBox1 است (که باز هم دریافت ID آن به صورت داینامیک صورت گرفته تا مشکل را نشود)

contentType: "application/json; charset=utf-8"

dataType: "json"، در این دو سطر از نوع داده‌ی json استفاده شده است که فرمت بسیار سبک و بهینه‌ای برای تبادل اطلاعات در وب به‌شمار می‌آید و توسط کتابخانه‌های جاوا اسکریپتی به سادگی پردازش شده و تبدیل به اشیاء مورد نظر خواهند شد. برای مثال اگر خروجی یک وب سرویس در حالت xml به صورت زیر باشد:

```
<xx yy="nn"></xx>
```

معادل json آن به شرح زیر است:


```
{ "xx": { "yy": "nn" } }
```

success: function(msg) Success ، پس از موفقیت آمیز بودن عملیات ajax در jQuery فراخوانی می‌شود

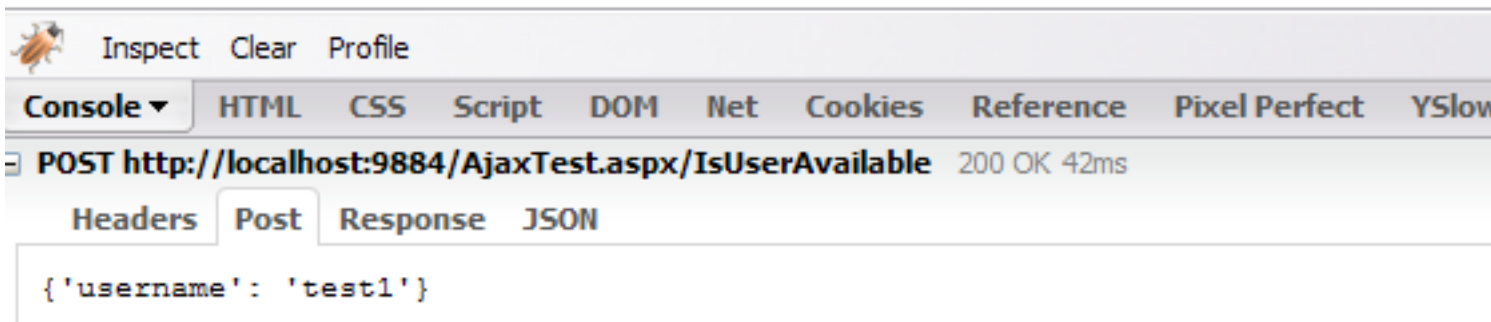
error: AjaxFailed و اگر در این بین خطایی رخ داده باشد، قسمت error فراخوانی می‌شود.


در این مثال برای نمایش بهتر عملیات، یک وقفه‌ی 2 ثانیه‌ای توسط setTimeout ایجاد شده و بدیهی است در یک مثال واقعی باید آن را حذف نمود.

نکته: با استفاده از افزونه‌ی فایرباگ فایرفاکس، می‌توان جزئیات این عملیات را بهتر مشاهده نمود:

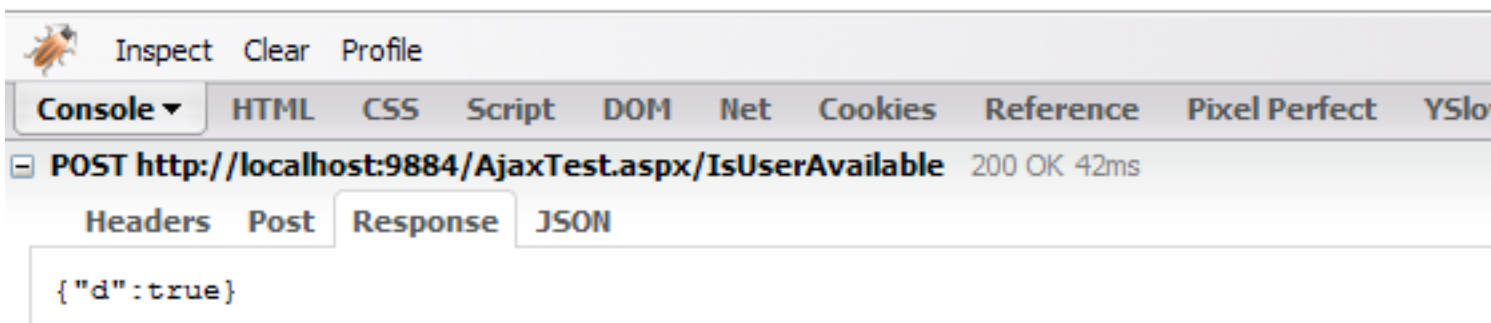
user name: 

pass:



user name: 

pass:



نظرات خوانندگان

نویسنده: مسعود
تاریخ: ۱۳۸۷/۱۱/۲۹ ۲۲:۳۸:۰۰

سلام استاد.

من برام یه سوال پیش اومده ،

من چند وقته ذهنم درگیر این شده که آیا واقعا vb.net متد های Static رو داره یا نه!

من سر این با خیلی ها بحث کردم.

بعضی ها میگن Modual جایگزینش شده ، بعضی ها متد های Shared رو جایگزین می دونن.

نظر شما چیه؟

راستی شما روی jQuery Ajax کار میکنید؟

من خیلی دوست دارم کار کردن باهاش رو.

دارم یه فریم ورک کوچولو برای JQuery Ajax برای دات نت می سازم.البته پر از ارور دی

نویسنده: وحید نصیری
تاریخ: ۱۳۸۷/۱۱/۳۰ ۰۱:۰۴:۰۰

سلام

بله. برنامه reflector معروف که مد نظر هست. اسمبلی برنامه فوق را با آن باز کردم و زبان انتخابی هم VB . حاصل به صورت زیر نمایش داده شد:

Public Shared Function IsUserAvailable

یک سری کار سورس باز دیدم در CodePlex در مورد jQuery Ajax که تلاش‌هایی برای ساده‌تر کردنش کردند. بد نیست برای تکمیل کار یک سری هم به آن‌ها هم بزنید. برای ایده گرفتن خوب است.

نویسنده: میلاد
تاریخ: ۱۳۸۸/۰۵/۰۱ ۱۰:۲۱:۱۳

سلام

اگه بخوایم دو پارامتر ارسال کنیم چی !!؟

نویسنده: وحید نصیری
تاریخ: ۱۳۸۸/۰۵/۰۱ ۱۰:۳۰:۴۶

محدودیتی ندارد. وب متد را اصلاح کنید تا دو پارامتر مورد نظر را قبول کند. همچنین قسمت data در اسکریپت را که با فرمت json باید ارسال شود را نیز اصلاح کنید. مثلا

```
{var json = '{"param1":"val1", "param2":"val2", "param3":number3
```

نویسنده: mat21
تاریخ: ۱۳۸۸/۱۰/۰۹ ۱۱:۵۹:۵۷

با سلام و تشکر

من مثال فوق را استفاده کردم اما همیشه قسمت error به اجرا در میاد و error status 200 را بر می گرداند اما زمانی که "datatype: json" را از کد حذف می کنم قسمت success اجرا می شه اما نتیجه درست نیست. من از framework 2.0 استفاده می کنم، ممنون می شم راهنمایم کنید.

نویسنده: وحید نصیری
تاریخ: ۱۳۸۸/۱۰/۰۹ ۱۴:۲۰:۴۴

سلام

در مورد دات نت 2 به آخرین کامنت بنده در این مقاله مراجعه بفرمائید (انتهای صفحه):
http://www.dotnettips.info/2009/07/blog-post_06.html

نویسنده: maryam
تاریخ: ۱۳۸۹/۰۷/۲۰ ۱۲:۳۲:۵۳

من این برنامه رو تست کردم همیشه قسمت error را اجرا می کند و این پیغام می دهد
Internal Server Error 500

نویسنده: وحید نصیری
تاریخ: ۱۳۸۹/۰۷/۲۰ ۱۳:۱۷:۳۸

کمکی نمی تونم بهتون بکنم و من عموماً پیغام هایی مانند پیغام شما رو حذف می کنم به این دلایل:
- نه کدی ارائه شده (کل Solution شما نیاز است برای بررسی)
- نه توضیحات کافی که چه مرحله ای رو طی کردید
- نه توضیحاتی که از چه ابزاری و چه نگارشی از VS.NET دارید استفاده می کنید.

و توقع دارید که من از راه دور کد شما رو در VS.NET ایی که معلوم نیست چه نگارشی است دیباگ کنم، پس از عیب یابی مشخص کنم مشکل چی بوده ...

این ها رو هم نوشتم تا کسانی که پیغام هاشون حذف شده بدونند علت این امر چی بوده.

نویسنده: وحید نصیری
تاریخ: ۱۳۸۹/۰۷/۲۰ ۱۳:۲۰:۴۱

یک مورد کلی را اضافه کنم برای دیباگ Internal Server Error 500 که عرض کردم از راه دور نمی شود آن را دیباگ کرد: افزونه ی فایرباگ مربوط به فایرفاکس را نصب کنید. سپس آن را روی صفحه جاری خودتون فعال کنید. صفحه رو ریفرش کنید و سپس یکبار مراحل رسیدن به خطا رو طی کنید. در قسمت network مربوط به firebug دقیقاً اطلاعات رد و بدل شده و همچنین خطای نهایی که سیستم برگردونده لاگ می شود. این را بررسی کنید تا مشکل را بشود حل کرد.

نویسنده: شاهین کیاست
تاریخ: ۱۳۸۹/۱۱/۱۱ ۲۱:۴۱:۴۳

سلام آقای نصیری.

من عین کد شما را در Page تست کردم و صحیح کار کرد اما در User Control پیام Forbidden را بر می گرداند. آیا علت خاصی دارد؟

نویسنده: وحید نصیری
تاریخ: ۱۳۸۹/۱۱/۱۱ ۲۳:۲۸:۳۶

بله. در یوزر کنترل ها این مورد ساپورت نمی شود. در تمام حالات روش صحیح و قابل نگهداری و قابل استفاده در قسمت های

مختلف برنامه، استفاده از یک WCF Service است (یا یک درجه پایین تر یک وب سرویس معمولی asmx با سرعت کمتر از WCF سرویس).

نویسنده: ایمان اسلامی
تاریخ: ۱۴:۲۶ ۱۳۹۱/۰۴/۲۱

با سلام و خسته نباشید خدمت استاد نصیری

من برنامه ام رو با code first و الگوی UnitOfWork که شما مطرح کردید نوشتم که واقعاً عالی و لذت بخش هست. مشکلی که هست اینه که اگه بخوام از این امکان jqueryAjax در برنامه خودم استفاده کنم میبایست userService تعریف شده رو در یک متد استاتیک (با توجه به مطلب بالا) استفاده کنم که اینکار انجام نمیشه. ممنون میشم کمک کنید.

نویسنده: وحید نصیری
تاریخ: ۱۶:۴ ۱۳۹۱/۰۴/۲۱

- در MVC این مشکلات وجود ندارد و سازگاری کاملی با jQuery در آن در نظر گرفته شده. در آنجا لازم نیست چیزی رو استاتیک تعریف کنید. همچنین چون صفحات دارای ViewState هم نیستند، به سادگی می شود قسمتی از آن را با jQuery Ajax کلا تعویض کرد بدون اینکه پیغام خراب بودن ViewState را در post back به سرور، دریافت کنید. در آنجا مفهومی به نام PartialView تعریف شده که به صورت خودکار یک View را به شکل رشته برای شما رندر می کند و در اختیار jQuery Ajax قرار می دهد.

- در اینجا اگر نمی تونید از وهله سازی خودکار توسط DI Container استفاده کنید، می شود آن را دستی هم انجام داد. اصطلاحاً به آن Service locator گفته می شود. نمونه آن را در استفاده از ObjectFactory.GetInstance در [قسمت 12](#) دیده اید (استفاده از الگوی واحد کار و کلاس های سرویس تهیه شده در یک برنامه کنسول ویندوزی).

نویسنده: ایمان اسلامی
تاریخ: ۱۶:۳۰ ۱۳۹۱/۰۴/۲۱

من فراموش کردم که اشاره کنم من از asp.net web form استفاده میکنم.
راه حل برای asp.net web form چیست؟

نویسنده: وحید نصیری
تاریخ: ۱۶:۳۳ ۱۳۹۱/۰۴/۲۱

همان توضیحات بند دوم نظر قبلی بنده (الگوی service locator مخصوص به فناوری خاصی نیست و در صورت لزوم در همه جا می توان از آن استفاده کرد. یک ضد الگو است. تا حد امکان نباید از آن استفاده کرد؛ اما اگر امکان استفاده صحیح از تزریق وابستگی ها نبود، این روش هم کار می کنه).

نویسنده: ایمان اسلامی
تاریخ: ۱۶:۵۶ ۱۳۹۱/۰۴/۲۱

مهندس فقط میتونم بگم "یه دونه ای!"
واقعاً ممنون

کار کرد. البته در تمامی page ها از base Page ارث بری میکنیم. این استفاده از Object Factory با توجه به استفاده از Object Factory در سازنده کلاس base page، مشکلی ایجاد نمیکند؟

نویسنده: وحید نصیری
تاریخ: ۱۷:۱۱ ۱۳۹۱/۰۴/۲۱

خیر. در پشت صحنه این کتابخانه، Object Factory به صورت خودکار عمل می کند. اگر امکان استفاده خودکار از آن نبود، روشی دستی آن (ObjectFactory.GetInstance) همان عملیات را انجام می دهد.

نویسنده: مهدی غفاری
تاریخ: ۱۵:۴۶ ۱۳۹۱/۰۷/۰۸

من یک پوشه ایجاد کردم و صفحه AjaxTest.aspx رو تو اون قرار دادم
بعد برای اینکه کسی نتونه با وارد کردن آدرس اون به صفحه دسترسی داشته باشه، توی اون پوشه یه web.config ایجاد کردم و
تگ زیر رو به اون اضافه کردم

```
<deny users="*" />
```

چون میخواستم هیچ کس نتونه اون صفحه رو ببینه. ولی خوب کدهای نوشته شده توسط من باید بتونن به اون صفحه دسترسی داشته باشن (فقط برنامه)
لطفا بفرمائید این رو چطور ادیت کنم تا این کار ممکن بشه؟
ممنونم از شما

نویسنده: وحید نصیری
تاریخ: ۱۶:۳۱ ۱۳۹۱/۰۷/۰۸

jQuery یا کلا JavaScript یعنی کدهای سمت کاربر. بنابراین این کدها باید دسترسی رسیدن به مدخل سمت سرور را داشته باشند و اگر عموم از آن منع شده باشند، یک اسکریپت معمولی هم بدیهی است که دسترسی خاصی نخواهد داشت چون سمت سرور اجرا نمی‌شود. فقط نتیجه عملیات آن به سمت سرور Post می‌شود.
یکی از کارهای متداول سمت سرور جهت منع عموم و فقط دسترسی دادن به jQuery مورد زیر است:
« [بررسی امنیتی، حین استفاده از jQuery Ajax](#) »

نویسنده: مهدی غفاری
تاریخ: ۲۲:۴۶ ۱۳۹۱/۰۷/۰۸

ممنونم از توجه شما؛
آیا همیشه از هندلرها استفاده کرد تو همچنین مواقعی؟

نویسنده: وحید نصیری
تاریخ: ۲۲:۵۲ ۱۳۹۱/۰۷/۰۸

بله امکان پذیر است. مثل این مثال [اسلایدشو](#) می‌تونید JSON بازگشت دهید.