

بیشتر برنامه‌های کامپیوتری تنظیمات خاص خود را در قالب مفهومی بنام Setup نگهداری می‌کنند. های ASP.NET هم از این قاعده مستثنی نیستند و می‌توان کلیه تنظیمات وب سایت را در قالب یک فایل از جنس XML Web.Config نگهداری کرد.

بیشتر برنامه‌های کاربردی کامپیوتری دارای بخشی Setup هستند که برای انجام تنظیمات متناسب با استفاده‌های مختلف کاربران مورد استفاده قرار می‌گیرد. برنامه‌های کاربردی مبتنی بر Web نیز که دارای کاربران زیادی هستند از این قاعده مستثنی نیستند و احتیاج به انجام تنظیماتی متناسب با کاربردهای مختلف خود دارند.

ASP.NET اینگونه تنظیمات که مربوط به منابع مورد استفاده برنامه کاربردی، اطلاعات محلی، اطلاعات امنیتی و... می‌شوند در داخل یک فایل XML ذخیره می‌شوند که نام آن Web.Config است. XML شامل تعدادی تگ (Tag HTML است و به کوچک و بزرگ بودن حروف نیز حساس است. (Case Sensitive).

در این مقاله به اختصار به توضیح در مورد تگهای مهم این فایل و چگونگی تنظیم خصوصیات مهم این تگها و کاربردهای آنها می‌پردازیم. پیش از ادامه مطلب این نکته را متذکر می‌شویم که این فایل زیادی تگ با خصوصیات مختلف است که هر یک به جنبه‌ای از یک برنامه کاربردی تحت Web هستند. مقاله در مورد تگهای پرکاربرد و مهم این فایل و خصوصیات مهم آنها اطلاعاتی ارائه است برای به دست آوردن اطلاعات بیشتر در مورد هر یک از تگها و خواص آنها باید به منابع تخصصی مرتبط با هر بخش رجوع کرد.

محل و محدوده عملکرد فایل Web.Config

به طور طبیعی هر برنامه کاربردی Web ASP.NET دارای حداقل یک فایل Web.Config در داخل دایرکتوری ریشه خود می باشد که به طور خودکار توسط Visual Studio .NET Web Application ایجاد شده و با مقادیر پیش فرض در دایرکتوری ریشه برنامه کاربردی قرار می گیرد. ولی برنامه نویسان می بر حسب نیازها و کاربردهای برنامه های خود تعداد بیشتری از این فایل XML دایرکتوریهای مختلف برنامه ایجاد کنند و برای تنظیمات فایلها داخل همان دایرکتوری یا زیر دایرکتوریهای آن به کار گیرند.

تنظیمات هر فایل Web.Config فایلها داخل زیر دایرکتوری ای که در آن قرار دارد و همچنین زیر دایرکتوریهای آن اعمال می شود. همچنین برنامه نویس می تواند از طریق کدهای داخل هر فایل Web.Config منابع و دایرکتوریهای مورد نظر را تغییر دهد. برای درک بهتر نحوه ارتباط فایلها Web.Config در یک برنامه کاربردی Web ، مثال زیر را بیان می کنیم.

اگر یک دایرکتوری به نام main و دو زیر دایرکتوری به نام Sub1 Sub2 در برنامه کاربردی موجود باشند که در درون هر کدام از این دایرکتوریها یک فایل Web.Config باشد تنظیمات هر فایل Web.Config بر فایلها داخل زیر دایرکتوری آن اعمال می شود اما اگر در یک فایل Web.Config که داخل زیر دایرکتوری Sub1 Sub2 قرار دارد یکی از تنظیماتی که در فایل Web.Config که داخل main تنظیم شده در داخل فایل موجود در main به زیردایرکتوریها به ارث می

نکته ای که باید در اینجا به آن اشاره کرد این است که کاربران برنامه وب به محتوای فایلها Web.Config دسترسی ندارند.

Web.Config

هر فایل Web.Config شامل یک بخش پایه است که همان دو تگ <configuration> </configuration> هستند و بقیه محتوا درون این دو تگ قرار می گیرد. پس ساختار کلی هر فایل Web.Config

:

```
<configuration>
<!--Some Configurations-->
</configuration>
```

محتوای هر فایل Web.Config شامل دو بخش اساسی است: معرفی و تنظیمات. بخش معرفی داخل دو `<configsections/>` `<configsections>` قرار می‌گیرد و بخش تنظیمات، تنظیمات مربوط به منابع معرفی شده در این تگها را شامل می‌شود.

ابتدای فایل Web.Config `<configuration>` `<configsections>` تگ جفت آن یعنی `<configsections/>` تگهایی جهت معرفی منابع و اداره کننده قسمتها قرار دارد. حال که با ساختار کلی فایل Web.Config آشنا شدیم در ادامه به توضیحاتی در مورد تگهای مهم این فایل و خصوصیات آنها می‌پردازیم.

Location

بیشتر اشاره شد که تنظیمات هر فایل Web.Config بر فایل‌های داخل ریشه این فایل و زیردایرکتوری‌های آن اعمال می‌شوند، حال اگر بخواهیم تنظیمات یک فایل Web.Config را بر دایرکتوری خاصی اعمال کنیم تنها لازم است این تنظیمات را داخل یک جفت تگ `<location>` `<location/>` دهیم یعنی شکل کلی :

```
<location path="url">
<!--Configurations-->
</location>
```

اطلاعات محلی

<globalization> برای تعریف خصوصیات فرهنگی و زبانی برنامه کاربردی استفاده می شود.

مهم این تگ عبارتند :

• culture: فرهنگ برنامه کاربردی را تعیین می کند .

• requestencoding: شکل رشته های درخواست شده را تعیین می کند (برای مثال، Unicode).

• responseencoding: شکل رشته های پاسخ را تعیین می کند .

تگهای امنیتی

Web.Config تگهایی جهت تنظیم مقادیر و خصوصیات امنیتی برنامه کاربردی قابل تعریف هستند.

جفت تگهای <authentication> <authentication/> <authorization> <authorization/>

ساختار امنیتی درون فایل Web.Config را می سازند که دارای خصوصیات و مقادیر متفاوتی هستند که توضیح در مورد آنان به مباحث مقدماتی امنیت در ASP.NET بر می گردد ولی در زیر توضیح کوتاهی از هر یک از این جفت تگها ارائه می .

<authentication>: این جفت تگ به تعیین شکل تایید اعتبار در برنامه کاربردی تحت Web می پردازد.

از طریق خصوصیت mode که چهار مقدار Form Passport Windows None را می پذیرد می توان شکل تایید اعتبار را به ترتیب بر مبنای ساختار امنیتی IIS ، سرویس تایید اعتبار Passport شرکت مایکروسافت، cookie ها و بدون استفاده از سیستم امنیتی تعیین کرد .

<authorization>: این جفت تگ برای تعیین کاربران واجد شرایط و یا فاقد شرایط

کنترل فایل Web.Config استفاده می شود. <allow> <deny> می توان کاربرانی

که اجازه ورود به محدوده تحت حفاظت را دارند و فاقد این اجازه هستند استفاده کرد .

خطای زمان

<customerrors> جهت تعیین شکل عملکرد برنامه در زمان وقوع یک خطای پیش بینی نشده در زمان درخواست یک صفحه توسط کاربر به کار می رود. دارای دو صفت مهم است که در زیر توضیح داده

mode: این صفت جهت تعیین نوع عملکرد برنامه کاربردی در مواجهه با خطا به کار می رود و یکی از سه RemoteOnly On Off می گیرد. با تنظیم این صفت با مقدار RemoteOnly بروز خطای زمان اجرا صفحه از پیش تعیین شده در قسمت defaultredirect برای کاربران نشان داده خواهد شد ولی کاربرانی که بصورت محلی از برنامه استفاده می کنند (صفحه خودکار ASP.NET) و جزئیات خطا را خواهند دید. با تنظیم On در صورت بروز خطای زمان اجرا همواره صفحه از پیش تعیین شده در قسمت defaultredirect نشان داده خواهد شد. با تنظیم این صفت با مقدار Off صفحه خطای خودکار ASP.NET به نمایش در می آید که از لحاظ امنیتی گزینه مناسبی نیست.

defaultredirect: • Url فایلی را که برای نمایش خطا در زمان تنظیم خصوصیت mode On RemoteOnly بکار می رود تعیین می کند .

وضعیت جلسه کاری

<sessionState> برای تعیین وضعیت جلسه کاری مورد استفاده قرار می گیرد و مدت زمان یک جلسه کاری و همچنین شکل تعیین درخواست کاربران را تعیین می کند. از صفات مهم این تگ یکی Cookiless است که تعیین می کند جلسه کاری کاربران Cookie ها یا SessionID ها نسبت داده شوند که دو true false را می گیرد true: برای استفاده از SessionID false برای است Cookie.

خصوصیه مهم دیگر این تگ mode است که یکی از چهار مقدار زیر را می گیرد:

Off: جلسه کاری غیر فعال

Inproc: • In-process ذخیره می شوند .

SQLServer: • اطلاعات جلسه کاری توسط پایگاه داده SQL Server مدیریت می شوند .

StateServer: • اطلاعات توسط یک سرویس Out-Of-Process می شوند .

خصوصیت مهم دیگر این تگ timeout است که زمان اعتبار جلسه کاری هر کاربر را بر حسب دقیقه تعیین می کند .

کامپایل برنامه کاربردی

<compilation> برای تعیین خصوصیات مربوط به شکل کامپایل برنامه کاربردی مورد استفاده قرار می گیرد که دو خصیصه مهم آن عبارتند :

debug: • یکی از دو مقدار true false را می گیرد (false) پیش فرض است (و تعیین می کند که اطلاعات اشکال زدایی در اسمبلی های بعد از کامپایل قرار بگیرند یا خیر. با تنظیم این خصیصه با مقدار true این کار رت می گیرد. و با تنظیم این خصیصه با مقدار false این کار صورت نمی گیرد .

defaultlanguage: • نویسی مورد استفاده جهت کامپایل برنامه کاربردی (Dynamic) را تعیین می کند .