

## معرفی ویژگی ها و قابلیت های تکنولوژی ASP

### معرفی برخی ویژگی های تکنولوژی ASP

- ASP، یک محیط مناسب بر روی سرویس دهنده را بگونه ای فراهم می کند که امکان ایجاد صفحات وب پویا و نرم افزارهای کاربردی تحت وب بسرعت و بسادگی فراهم میگردد.
- صفحات ASP، فایل هائی هستند که شامل دستورات، تگ ها HTML (، متن و دستورات اسکریپت می باشند.)
- صفحات ASP می توانند اجزای ActiveX را جهت انجام عملیات خاصی نظیر اتصال به یک بانک اطلاعاتی و یا محاسبات تجاری خاصی و انجام دهند.
- با استفاده از تکنولوژی ASP، میتوان اقدام به طراحی و پیاده سازی صفحات وب پویا نمود.
- با استفاده از تکنولوژی ASP، میتوان اطلاعات متفاوتی را از طریق فرمهای موجود در صفحات وب جمع آوری و پردازش دلخواه را بر روی آنها انجام داد.
- جهت ایجاد صفحات ASP، می توان از زبانهای اسکریپت متعددی نظیر Vbscript، Jscript، Perl، REXX استفاده نمود.

### مدل عملیاتی ASP

- دستورات و اسکریپت های موجود در صفحات ASP، زمانی اجراء میگردند که مرورگر یک فایل با انشعاب .asp، را از سرویس دهنده وب تقاضا نماید. در ادامه سرویس دهنده وب، ASP مربوطه را صدا زده و پس از اجرای دستورات و اسکریپت های موجود در آن، ماحصل نتایج بصورت یک صفحه وب برای مرورگر ارسال میگردد.
- با توجه به اینکه صفحات ASP بر روی سرویس دهنده اجراء می گردند، بنابراین تمامی عملیات مربوط به اجرای اسکریپت ها و دستورات بر روی سرویس دهنده انجام شده و وابسته به مرورگر نخواهد بود. در ضمن با توجه به اجرای صفحات ASP بر روی سرویس دهنده، امکان مشاهده و تکثیر این نوع فایل ها وجود نداشته و صرفاً ( ماحصل ) نتایج ( اجرای ASP بصورت کدهای استاندارد HTML برای مرورگر ارسال میگردد.

## معرفی فناوری ASP و تاریخچه آن به همراه قابلیت و لوازم شروع آن

شرکت ماکروسافت در سال 1996 تکنولوژی ASP (Active Server Page) را معرفی نمود. تکنولوژی فوق اولین بار به همراه سرویس دهنده اطلاعاتی اینترنت (IIS: Internet Information Server) نسخه 3.0 ارائه گردید. با بکارگیری تکنولوژی فوق امکان طراحی و پیاده سازی نرم افزارهای بزرگ و بانک های اطلاعاتی به همراه صفحات وب فراهم می شود. هسته اولیه ASP زبان Vbscript است. صفحات ASP بکمک کدهای نوشته شده توسط زبان فوق ایجاد می گردند. Vbscript بعنوان زبان استاندارد برای پیاده سازی صفحات ASP در نظر گرفته شده است ولی می توان از زبانهای Perl و Jscript در این راستا نیز استفاده نمود ASP. بر روی سرویس دهنده اجراء می گردند. بدیهی است در این حالت عملکرد صفحات وب مستقل از توانائی های مرورگرها خواهد بود.

بنا به اظهارات شرکت ماکروسافت در ماه مه سال 2000، قریب به 800,000 هزار نفر در این زمینه فعالیت داشته و به طراحی صفحات وب با بهره گیری از تکنولوژی فوق مشغول هستند. آمار فوق نشاندهنده رشد سریع و تصاعدی و استقبال عموم از تکنولوژی فوق در سطح جهان دارد. قریب به 500 شرکت کامپیوتری در سطح جهان بصورت کاملاً تخصصی صرفاً به تولید قطعات و اجزای متفاوت بصورت کیت های آماده نرم افزاری بکمک ASP فعالیت داشته و محصولات ارائه شده توسط آنان بر روی استفاده در وب سایت های متعدد عرضه میگردد.

### ASP چه کارهایی را می تواند انجام دهد؟

تقریباً "همه چیز"! توانائی های تکنولوژی فوق نامحدود است. مثلاً: دستیابی و جستجو در بانک های اطلاعاتی متفاوت مستقل از بانک، شخصی نمودن صفحات وب، نمایش صفحات متفاوت برای مرورگرهای متفاوت، حفاظت رمز عبور، بازیهای کامپیوتری، پردازش فرم ها و نمونه هایی از توانائی های بیشمار تکنولوژی فوق می باشند.

### برای شروع به چه چیزی نیاز است؟

جهت استفاده کامل از توانائی های ASP، می بایست سیستم عامل ویندوز NT 4.0 را به همراه IIS 4.0 استفاده نمود (نسخه IIS 4.0 در زمان نصب ویندوز NT و پس از نصب NT 4.0 Option Pack بر روی سیستم نصب میگردد). از ویندوز Server 2000 به همراه IIS 5.0 نیز میتوان در این زمینه استفاده نمود.

جهت استفاده از ASP بر روی کامپیوترهایی که بر روی آنها ویندوز 98 و یا 95 نصب گردیده است میتوان سرویس دهنده شخصی وب (PWS: Personal Web Server) را بر روی کامپیوتر نصب نمود. جهت نصب PWS، کفایت از CD ویندوز 98 استفاده و در شاخه Add-ons\PWS برنامه Setup را اجراء نمود.

## معرفی فناوری ASP و بستر مورد نیاز آن

### ASP چیست ؟

- asp - مخفف کلمات Active Server Page است.
- asp - برنامه ای است که بکمک IIS اجراء می گردد .
- IIS - مخفف کلمات Internet Information Server است.
- IIS - یکی از عناصری است که به همراه ویندوز 2000 آورده شده است.
- IIS - یکی از اجزای Windows NT 4.0 Option Pack نیز می باشد.
- PWS ( Personal Web Server ) - یک نسخه خاص و کوچکتر از برنامه IIS است .
- PWS - را می توان از روی CD حاوی ویندوز 98 نصب نمود.

### یک فایل ASP شامل چه چیزهایی است ؟

- یک فایل asp در اغلب موارد مشابه یک فایل HTML است.
- یک فایل asp می تواند شامل متن ، کدهای HTML ، کدهای XML و اسکریپت ها باشد.
- اسکریپت ها ی موجود در یک فایل asp بر روی سرویس دهنده ( Server ) اجراء می گردند.
- یک فایل asp دارای انشعاب asp است.

### یک فایل asp چگونه کار می کند؟

- زمانیکه یک مرورگر نظیر ( IE ) درخواست یک فایل HTML را داشته باشد ، سرویس دهنده آن را ارسال میدارد .
- زمانیکه یک مرورگر درخواست یک فایل asp را داشته باشد ، IIS درخواست واصله را به هسته ASP ارسال میدارد .
- هسته asp ، فایل asp را خط به خط خوانده و اسکریپت های موجود در آن را اجراء خواهد کرد.
- در نهایت ، ماحصل اجرای یک فایل asp بصورت کدهای HTML برای مرورگر ارسال خواهد شد.

### با استفاده از asp چه کار می توان کرد؟

- امکان ویرایش ، تغییر و یا اضافه کردن موارد دلخواه بصورت پویا برای یک صفحه وب فراهم می گردد.
- پاسخ به درخواست های واصله از کاربران و یا داده های ارسال شده توسط فرم های HTML
- دستیابی به هر نوع داده و یا بانک اطلاعاتی و برگرداندن نتایج به مرورگر
- سفارشی نمودن یک صفحه وب بمنظور استفاده بهتر برای کاربران خاص
- در مقایسه با تکنولوژی CGI بمراتب ساده تر و سریعتر می باشند.
- تامین امنیت لازم در خصوص عدم نمایش کدهای مربوطه توسط مرورگرها
- با توجه به برگرداندن نتیجه اجرای یک asp بصورت HTML امکان مشاهده آنان توسط هر مرورگر ب راحتی فراهم میگردد.

### چگونه می توان یک فایل asp را بر روی کامپیوتر شخصی خود اجراء نمود؟

با نصب یکی از نرم افزارهای PWS و یا IIS بر روی کامپیوتر خود (با توجه به سیستم عامل نصب شده) می توان بدون نیاز به یک سرویس دهنده کمکی دیگر، اقدام به اجرای فایل های asp نمود.

#### • نحوه نصب PWS و اجرای فایل های asp بر روی ویندوز 98

مرحله اول: برنامه setup.exe را از مسیر \Add-ons\PWS موجود در CD حاوی ویندوز 98 اجراء نمائید.  
 مرحله دوم: یک فولدر با نام Inetpub بر روی هارد دیسک (مسیر مشخص شده در زمان نصب) ایجاد و در آن یک فولدر با نام wwwroot ساخته می گردد. فولدر فوق را باز نمائید.  
 مرحله سوم: یک فولدر جدید با نام دلخواه (نظیر MyfirstWeb) در فولدر wwwroot ایجاد نمائید.  
 مرحله چهارم: با استفاده از یک ادیتور متنی (نظیر Notepad) کدهای دلخواه asp خود را تایپ کرده و فایل فوق را بام دلخواه (Test.asp) ولی با انشعاب asp در فولدر MyfirstWeb ذخیره نمائید.  
 مرحله پنجم: اطمینان حاصل نمائید که سرویس دهنده وب (PWS) در حالت اجراء است.  
 مرحله ششم: مرورگر خود را فعال نموده و در محل مربوطه، آدرس زیر را بمنظور مشاهده فایل asp تایپ نمائید.

<http://localhost/MyfirstWeb/Test.asp>

#### • نحوه نصب و اجرای PWS و اجرای فایل های asp بر روی ویندوز NT

PWS بمنظور نصب بر روی ویندوز NT طراحی نشده است! برای اجرای فایل های asp بر روی ویندوز NT، میبایست "Windows NT 4.0 Option Pack" را تهیه و آن را نصب نمود. پس از نصب pack فوق و نصب IIS زمینه اجرای فایل های asp بر روی ویندوز NT فراهم می گردد.

#### • نحوه نصب PWS و اجرای فایل های asp بر روی ویندوز ME

PWS به همراه ویندوز ME آورده نشده است!

#### • نحوه نصب IIS و اجرای فایل های asp در ویندوز 2000

مرحله اول: فعال کردن Control panel.  
 مرحله دوم: انتخاب Add/Remove Programs  
 مرحله سوم: در پنجره مربوطه، Add/Remove Windows Components را انتخاب نمائید.  
 مرحله چهارم: در پنجره ویزارد مربوطه Internet Information Services را فعال نمائید.  
 مرحله پنجم: یک فولدر با نام Inetpub بر روی هارد دیسک (با توجه به مسیر داده شده) ایجاد می گردد.  
 مرحله ششم: در فولدر فوق یک فولدر دیگر و با نام wwwroot ایجاد می گردد.  
 مرحله هفتم: در فولدر wwwroot یک فولدر دیگر و با نام دلخواه خود (MyFirstWeb) ایجاد نمائید.

<http://localhost/MyfirstWeb/Test.asp>

## نقطه شروع کار با ASP شامل معرفی، کاربرد و مزایای ASP

Active Server Pages فناوری شرکت میکروسافت جهت تولید سایتهای پویای وب می باشد. عبارت دیگر ASP یک روش تولید صفحات وب است که طراحان وب را قادر می سازد تا صفحات HTML را بگونه ای طراحی نمایند که این صفحات برحسب مورد و در زمان فراخوانی بصورت خودکار ایجاد شوند. بسیاری از سایتهای وب بزرگ و معروف از این فناوری استفاده نموده اند، مانند:

- [www.microsoft.com](http://www.microsoft.com) - Microsoft
- [www.dell.com](http://www.dell.com) - Dell online
- [www.bn.com](http://www.bn.com) - Barnes & Noble

در ASP دستورالعمل ها (Script) بر روی سرور اجرا می شوند و سپس حاصل آن بصورت HTML به کامپیوتر کاربر ارسال می گردد. VBScript زبان پیش فرض دستورالعمل نویسی است، اما شما می توانید از دیگر زبانهای دستورالعمل نویسی مانند JScript، Perl استفاده کنید. یک صفحه ASP مانند یک صفحه HTML است با این تفاوت که پسوند نام فایل آن ".asp" می باشد. همچنین یک صفحه ASP می تواند دستورالعملهای نزد سرور و نزد کاربر را همزمان درون خود داشته باشد. عبارت دیگر در ASP معمولاً از VBScript برای دستورالعملهای نزد سرور و از JavaScript برای دستورالعملهای نزد کاربر استفاده می گردد. دقت داشته باشید که JScript و JavaScript دو زبان متفاوت هستند.

## قابلیت های ASP

### • استقلال از مرورگر

از ASP از مرورگر کاملاً مستقل است زیرا تمامی دستورالعملها بر روی سرور اجرا می گردند و مرورگر تنها صفحات HTML معمولی را بعنوان حاصل اجرای دستورالعملها دریافت می کند.

### • استفاده از زبان دستورالعمل نویسی

این بدین معنی است که یک برنامه ASP براحتی قابل تغییر است چرا که زبانهای دستورالعمل نویسی نیاز به کامپایل

• دسترسی آسان و سریع به انواع بانکهای اطلاعاتی  
 اعمال پیچیده مربوط به پردازش بانکهای اطلاعاتی در ASP بسیار ساده است به سرعت می توان برنامه دلخواه را نوشته  
 یا تغییر داد. همان روش مورد استفاده در Visual Basic یا Visual C++ را می توانید در ASP هم استفاده کنید  
 (ADO) و نیاز به یادگیری مجدد روش خاصی ندارید.

### (نزد سرور) ASP بستر مورد نیاز

Windows 2000: شامل ASP 3.0 و IIS 5.0  
 Windows NT: شامل ASP 2.0 و IIS 4.0  
 Windows 95/98: می توان با نصب Personal Web Server (PWS) از ASP 2.0 استفاده نمود.  
 • سایر سیستم عامل ها: می توان با استفاده از نرم افزار Chilisoft - [www.chilisoft.com](http://www.chilisoft.com) تقریباً در هر سیستم  
 عاملی از ASP استفاده نمود.

## راه اندازی ASP

### \*\* چگونه ASP را در رایانه خود اجرا کنید؟

شما می توانید ASP را بدون سرور خارجی در رایانه شخصی خود اجرا کنید. برای انجام این کار باید Microsoft's  
 Personal Web Server (PWS) یا (Internet Information Server (IIS را در رایانه خود نصب کنید.  
 اگر در مورد اجرای ASP جدی هستید، باید حداقل ویندوز 98، نسخه دوم (Second Edition) داشته باشید.  
 اگر در مورد اجرای ASP بسیار جدی هستید، باید از ویندوز 2000 استفاده کنید .

### \*\* چگونه PWS را در ویندوز 95 نصب کنید و ASP اجرا کنید؟

Personal Web Server (PWS) همراه ویندوز 95 نیست!!  
 برای اجرای ASP بر روی ویندوز 95، شما باید "Windows NT 4.0 Option Pack" را از مایکروسافت دان  
 لود کنید.

دان لود "Windows NT 4.0 Option Pack"

### \*\* چگونه PWS را در ویندوز NT نصب کنید و ASP اجرا کنید؟

برای اجرای ASP بر روی ویندوز 95، شما باید "Windows NT 4.0 Option Pack" را از مایکروسافت دان لود کنید.

دان لود "Windows NT 4.0 Option Pack"

### \*\* چگونه PWS را در ویندوز 98 نصب کنید و ASP اجرا کنید؟

- 1) شاخه Add-ons را در سی دی ویندوز 98 خود باز کنید، شاخه PWS را پیدا کرده و فایل setup.exe را اجرا کنید .
- 2) یک شاخه Inetpub در هارد شما ایجاد خواهد شد. آن را باز کنید و شاخه wwwroot را پیدا کنید .
- 3) یک شاخه جدید درست کنید و نام آن را چیزی شبیه به "MyWeb" بگذارید .
- 4) از یک ویرایشگر متن استفاده کنید و کدهای ASP ی را بنویسید، فایل را به نام "test1.asp" را در شاخه "MyWeb" ضبط کنید .
- 5) مطمئن شوید که وب سرور شما اجرا است - برنامه نصب یک آیکن جدید در Task Bar شما اضافه کرده است (این نشانه PWS است). بر روی آن کلیک کنید و دکمه Start را در پنجره ای که آمده بزنید .
- 6) Browser خود را باز کنید و تایپ کنید "<http://localhost/MyWeb/test1.asp>" تا اولین صفحه ASP خود را ببینید .

### \*\* چگونه PWS را در ویندوز Me نصب کنید و ASP اجرا کنید؟

PWS (Personal Web Server) در ویندوز Me وجود ندارد!!

مقاله مربوطه را در سایت مایکروسافت بخوانید

### \*\* چگونه IIS را در ویندوز 2000 نصب کنید و ASP اجرا کنید؟

- 1) از دکمه Start به Settings و سپس Control Panel بروید .
- 2) در پنجره Control Panel آیکن Add/Remove Programs را انتخاب کنید .
- 3) در پنجره Add/Remove Programs گزینه Add/Removes Windows Components را انتخاب کنید .
- 4) در پنجره جدید گزینه Internet Information Services را انتخاب کرده، OK را بزنید .
- 5) یک شاخه Inetpub در هارد شما ایجاد خواهد شد. آن را باز کنید و شاخه wwwroot را پیدا کنید .
- 6) یک شاخه جدید درست کنید و نام آن را چیزی شبیه به "MyWeb" بگذارید .
- 7) از یک ویرایشگر متن استفاده کنید و کدهای ASP را بنویسید، فایل را به نام "test1.asp" را در شاخه "MyWeb" ضبط کنید .

- 8) مطمئن شوید که وب سرور شما اجرا است - برنامه نصب یک آیکن جدید در Task Bar شما اضافه کرده است (این نشانه IIS است). بر روی آن کلیک کنید و دکمه Start را در پنجره ای که آمده بزنید .
- 9) Browser خود را باز کنید و تایپ کنید "<http://localhost/MyWeb/test1.asp>" تا اولین صفحه ASP خود را ببینید .

### \*\* چگونه IIS را در ویندوز XP نصب کنید و ASP اجرا کنید؟

**نکته:** شما نمی توانید ASP را روی ویندوز XP نسخه خانگی (Home Edition) اجرا کنید .

- 1) از دکمه Start به Settings و سپس Control Panel بروید .
- 2) در پنجره Control Panel آیکن Add/Remove Programs را انتخاب کنید .
- 3) در پنجره Add/Remove Programs گزینه Add/Removes Windows Components را انتخاب کنید .
- 4) در پنجره جدید گزینه Internet Information Services را انتخاب کرده، OK را بزنید .
- 5) یک شاخه Inetpub در هارد شما ایجاد خواهد شد. آن را باز کنید و شاخه wwwroot را پیدا کنید .
- 6) یک شاخه جدید درست کنید و نام آن را چیزی شبیه به "MyWeb" بگذارید .
- 7) از یک ویرایشگر متن استفاده کنید و کدهای ASP ی را بنویسید، فایل را به نام "test1.asp" را در شاخه "MyWeb" ضبط کنید .
- 8) مطمئن شوید که وب سرور شما اجرا است - برنامه نصب یک آیکن جدید در Task Bar شما اضافه کرده است (این نشانه IIS است). بر روی آن کلیک کنید و دکمه Start را در پنجره ای که آمده بزنید .
- 9) Browser خود را باز کنید و تایپ کنید "<http://localhost/MyWeb/test1.asp>" تا اولین صفحه ASP خود را ببینید .
- 10) یک فایل ASP می تواند حاوی متون، Tag های HTML و اسکریپت ها باشد. اسکریپت ها در یک فایل ASP بر روی سرور اجرا می شوند.
- 11) **به عنوان پیش نیاز چه چیزی را باید بدانید؟**  
قبل از شروع، باید با مسائل زیر را تا حدودی آشنا باشید
- 12) WWW و HTML و مسائل اولیه ساخت صفحات Web  
یک زبان اسکریپت نویسی همچون JavaScript یا VBScript

**ASP چیست؟**

- \* ASP حروف اختصاری Active Server Pages است .
- \* ASP برنامه ای است که درون IIS اجرا می شود .
- \* IIS حروف اختصاری Internet Information Services است .
- \* IIS به عنوان یک مولفه رایگان با ویندوز 2000 عرضه می شود .
- \* IIS همچنین به عنوان جزئی از Windows NT 4.0 Option Pack است .
- \* Option Pack می تواند از سایت مایکروسافت دان لود شود .
- \* PWS نسخه کوچکتر - ولی کاملا عمل کننده - از IIS است .
- \* PWS را می توانید در سی دی ویندوز 95 یا 98 خود پیدا کنید .

**سازگاری ASP**

- ASP تکنولوژی مایکروسافت است .
- برای اجرای IIS باید ویندوز NT نسخه 4.0 یا بالاتر را داشته باشید .
- برای اجرای PWS باید ویندوز 95 یا بالاتر را داشته باشید .
- Chili ASP تکنولوژی است که ASP را بدون نیاز سیستم عامل ویندوز اجرا می کند .
- Instant ASP تکنولوژی دیگری است که ASP را بدون نیاز به سیستم عامل ویندوز اجرا می کند .

**یک فایل ASP چیست؟**

- یک فایل ASP تقریبا شبیه به یک فایل HTML است .
- یک فایل ASP می تواند حاوی متن، XML، HTML، و اسکریپت باشد .
- اسکریپت های درون یک فایل ASP بر روی Server اجرا می شود .
- یک فایل ASP پسوند ".asp" دارد .

**فرق ASP با HTML چیست؟**

- وقتی که Browser (کاشگر اینترنت) درخواست یک فایل HTML را بکند، Server آن فایل را بر می گرداند .
- وقتی که Browser درخواست یک فایل ASP را بکند، IIS درخواست مربوطه را به موتور هسته ای که وظیفه

## ASP چه کاری می تواند برای شما انجام دهد؟

به طور دینامیکی محتویات یک Web page را ویرایش یا اضافه کند. به درخواست های کاربر که از فرم های HTML ارسال شده اند پاسخ دهد. به هر گونه اطلاعات با بانک اطلاعاتی دسترسی پیدا کرده و نتایج را به Browser برگرداند. سفارشی نمودن (Customize) یک Web page تا برای کاربران مختلف مفیدتر باشد. مزایای استفاده از ASP به جای CGI و Perl سرعت و سادگی است. فراهم کردن امنیت بیشتر به گونه ای که کد ASP شما از طریق Browser قابل مشاهده نیست. از آن جایی که فایل های ASP به شکل HTML برگردانده می شوند، میتوانند بوسیله هر Browser مشاهده گردند. برنامه نویسی درست ASP می تواند ترافیک شبکه را به حداقل برساند.

شما می توانید ASP را بدون نیاز به یک سرور خارجی اجرا کنید. برای انجام این کار شما باید Microsoft's Personal Web Server (PWS) یا Internet Information Server (IIS) را در رایانه خود نصب کنید.

## آشنائی با ساختار برنامه در ASP به همراه چند مثال

کدهای نوشته شده در یک ASP را نمی توان در یک مرورگر مشاهده نمود. در چنین حالتی صرفا حاصل اجرای کدهای ASP بصورت کدهای HTML توسط مرورگرها قابل نمایش خواهد بود.

### قانون اولیه در رابطه با گرامر ASP

یک فایل ASP اغلب شامل کدهای HTML نظیر آنچه که در یک فایل HTML استاندارد قرار می گیرد، است. علاوه بر کدهای HTML که در یک فایل ASP قرار می گیرد، می توان از مجموعه ای اسکریپت مخصوص سرویس دهنده نیز استفاده کرد. این اسکریپت ها بین  $<0\%$  و  $>0\%$  قرار می گیرند. اسکریپت های استفاده شده بر روی سرویس دهنده اجراء شده و می تواند شامل عبارات، عملگرها، جملات، برنامه های فرعی و سایر امکانات مجاز مربوط به زبانهای اسکریپت باشد.

تمامی علاقه مندان به دنیای نرم افزار بخصوص برنامه نویسی بخوبی آگاه هستند که در زمان آشنائی با یک زبان برنامه

**معرفی شیء Response**

متد Write مربوط به شیء Response ، مسئول ارسال اطلاعات ( محتویات ) به مرورگر است. مثلا در صورتیکه بخواهیم عبارت معروف فوق، توسط مرورگر نمایش داده شود، کافی است از دستور `Response.Write("Hello World")` استفاده گردد .

در ASP از زبانهای اسکریپت متفاوتی استفاده می شود. زبان پیش فرض در این زمینه VBScript است.

```
<%
Response.write("Hello World!")
%>
```

در صورتیکه بخواهیم از زبان جاوا اسکریپت بعنوان زبان پیش فرض استفاده کنیم ، کافی است در ابتدای صفحه زبان فوق را معرفی نمائیم.

```
<%
Response.write("Hello World!");
%>
```

جاوا اسکریپت بر خلاف VBScript به حروف بزرگ و کوچک حساسیت داشته و می بایست این امر مورد توجه قرار گیرد.

ASP امکان نوشتن کدهای مورد نظر را بکمک زبانهای اسکریپت جاوا اسکریپت و VBScript فراهم می کند. در صورتیکه قصد استفاده از سایر زبانهای اسکریپت نظیر Python , Perl را داشته باشیم، می بایست هسته ( engines ) مربوط به آن زبان بطور جداگانه نصب گردد. در پایان این بخش به بررسی دو مثال ساده خواهیم پرداخت.

**مثال** در این مثال با نحوه شکل دهی در یک ASP و بکمک امکانات موجود در HTML آشنا می شویم.

```
<%
Response.Write("<h2>Hello World!<br>This sentence uses HTML<br>tags to
format the text!</h2>")
%>
```

## نحوه تعریف متغیر و بکارگیری آن در ASP

استفاده از متغیر در یک برنامه بسیار حائز اهمیت است. برنامه نویسان از متغیرها برای ذخیره سازی داده های مربوطه استفاده و زمینه پردازش داده های ذخیره شده با امکان دستیابی به متغیرها در طول یک برنامه فراهم می گردد. یک برنامه کامپیوتری خود می تواند شامل چندین برنامه جانبی بوده که هر کدام مسئول انجام بخشی از عملیات در کل برنامه می باشند. متغیرها را می توان در درون یک تابع ( برنامه جانبی ) و هم بیرون از یک تابع تعریف نمود. نوع تعریف یک متغیر ( محلی و یا سراسری ) محدوده مجاز دستیابی به متغیر را تبیین خواهد ساخت.

در برنامه های ASP، زمانیکه یک متغیر در خارج از یک تابع تعریف می گردد، امکان دستیابی به آن در طول برنامه فراهم شده و از این بابت محدودیت در دسترسی را نخواهد داشت. در صورتیکه یک متغیر در یک تابع تعریف گردد، در زمان صدا زدن تابع متغیر ایجاد و پس از خروج از تابع متغیر از بین خواهد رفت. بدیهی است در چنین حالتی اسکریپت های موجود در سایر بخش های یک برنامه ASP، قادر به دستیابی و تغییر محتویات اینچنین متغیرهایی نخواهند بود. تا کنون بحث ما مربوط به محدوده مجاز یک متغیر در یک فایل ( برنامه ASP ) ( بود، در صورتیکه بخواهیم امکان دستیابی به یک متغیر را در طول چندین فایل ASP فراهم کنیم، می بایست آن را بصورت یک متغیر Session و یا یک متغیر Application تعریف نماییم.

### متغیرهای Session

این نوع متغیرها اطلاعاتی را در رابطه با یک کاربر ذخیره کرده و امکان دستیابی به آن برای تمامی صفحات موجود در یک برنامه فراهم می گردد. اغلب اطلاعات عمومی نظیر نام کاربر و مشخصه آن در چنین متغیرهایی ذخیره می گردد. پس از ایجاد یک متغیر Session، از شی Session برای ذخیره سازی آن استفاده می گردد.

### متغیرهای Application

این نوع متغیرها نیز قابل دستیابی در تمامی صفحات مربوط به یک برنامه ASP می باشند. از این متغیرها اغلب جهت ذخیره سازی اطلاعات در رابطه با تمامی کاربران یک برنامه خاص، استفاده می گردد. پس از ایجاد متغیری از این نوع، از شی Application برای ذخیره سازی آن استفاده می گردد.

در ادامه به بررسی چندین مثال کاربردی از نحوه تعریف و بکارگیری متغیرها در برنامه های ASP خواهیم پرداخت.

**مثال ۱** - ایجاد یک متغیر. در این مثال با نحوه تعریف، مقداردهی و استفاده از یک متغیر، آشنا خواهیم شد.

```
<%
Dim name
name="IranASP.NET"
Response.write("My favorite site is:"& name)
%>
```

**مثال ۲** - ایجاد یک آرایه ( بردار ) . از ساختمان داده بردار برای ذخیره سازی مجموعه ای محدود و یکسان داده استفاده می‌گردد. در این مثال با نحوه تعریف و استفاده از یک بردار آشنا خواهیم شد.

```
<%
Dim myaim(4)
myaim(0) = "I"
myaim(1) = "Like"
myaim(2) = "To"
myaim(3) = "Learn"
myaim(4) = "ASP"
For i = 0 to 4
response.write(myaim(i) & "<br>")
Next
%>
```

**مثال ۳** - حلقه تکرار . در این مثال با نحوه بکارگیری متغیرها و استفاده از آنان در حلقه های تکرار آشنا خواهیم شد

```
<%
Dim i
for i = 1 to 6

response.write("<h" & i & ">This is header " & i & "
Next
%>
```

**مثال ۴** - در این مثال با توجه به زمان مربوط به سرویس دهنده، پیامهای متفاوتی به کاربر داده خواهد شد.

```
<%
Dim h
h = hour(now)
response.write("<p>" & now)
response.write(" (Local Time) </p>")
If h < 12 then
response.write("Good Morning!")
else
response.write("Good Night!")
end if
%>
```

## بررسی روال ها و زیربرنامه ها در ASP کلاسیک

یک روال یا تابع مجموعه ای از یک یا چند دستور است که بصورت گروهی اجرا می شوند. یک تابع مانند یک روال است با این تفاوت که مقداری را برمی گرداند.

در VBScript برای تعریف یک روال از عبارت Sub و برای تعریف یک تابع از Function استفاده می کنیم. در VBScript جهت برگرداندن یک مقدار از تابع، باید مقدار مورد نظر را مانند زیر به نام تابع نسبت دهیم.

```
myFunction = myValue
```

در جاوااسکریپت برای نوشتن یک روال یا تابع تنها از عبارت function استفاده می کنیم. بعبارت دیگر یک روال تابعی است که مقداری را برنمی گرداند. جهت برگرداندن یک مقدار در توابع جاوااسکریپت از دستور return استفاده می کنیم.

```
return myValue;
```

در برنامه های ASP، می توان یک روال را از طریق VBScript و یا بالعکس صدا زد. برای صدا زدن یک روال از طریق یک برنامه ASP که با زبان VBScript نوشته شده است، از دستور Call که بدنال آن نام روال می آید، استفاده می گردد. در صورتیکه یک روال نیازمند پارامتر باشد، می بایست پارامترهای مربوطه را بین پرانتز قرار داده و آنها را در اختیار روال قرار داد. در صورتیکه از فرمان call برای صدا زدن یک روال استفاده نگردد، پارامترهای مربوطه را نباید بین پرانتز قرار داد. قرار دادن پرانتز در صورتیکه یک روال دارای پارامتر نباشد، اختیاری است.

در زمانیکه اسکریپت های موجود در یک برنامه ASP به زبان جاوااسکریپت نوشته شده باشند و قصد صدا زدن یک روال را داشته باشیم که با زبان VBScript و یا جاوااسکریپت نوشته شده باشد، قرار دادن پرانتز بعد از نام روال ضروری است.

در مثال زیر با نحوه صدا زدن یک روال (تابع Zarb نوشته شده با VBScript) توسط یک برنامه ASP آشنا خواهیم شد.

```
<html>
<head>
<%
Sub Zarb(num1,num2)
response.write(num1*num2)
End Sub
%>
</head>
```

```

<body>
<p>Calling a Subroutine with Call statement: <%call Zarb(7,6)%>
  <p>Calling a Subroutine without Call statement: <%Zarb 6,7%>
</body>
</html>

```

در مثال زیر با نحوه صدا زدن دو روال ( یکی با جاوا اسکریپت و دیگری با VBScript نوشته شده است) در یک برنامه ASP آشنا می شویم.

```

<html>
<head>
  <%
    Sub ZarbVBScript(num1,num2)
      Response.Write(num1*num2)
    End Sub
  %>

  <script language="javascript" runat="server">
    function ZarbJavascript(num1,num2)
      {
        Response.Write(num1*num2)
      }
  </script>
</head>
<body>
  <p>The Result of the VBScript Subroutine: <%call ZarbVBScript(6,7)%>
  <p>The Result of the JavaScript Subroutine: <%call ZarbJavascript(6,7)%>
</body>
</html>

```

## طرز ساخت کوکی ها

یک Cookie اغلب برای شناسایی کاربر استفاده می شود .

مثالها

Cookie خوش آمدگویی

چگونه یک Cookie خوش آمدگویی درست کنیم.

**یک Cookie چیست؟**

یک Cookie اغلب برای شناسدن کاربر استفاده می‌شود. یک Cookie یک فایل کوچک است که سرور آن را در کامپیوتر کاربر جاسازی می‌کند. هر بار که همان کامپیوتر یک صفحه را درخواست می‌کند، Cookie نیز فرستاده می‌شود. با ASP شما می‌توانید هم مقادیر Cookie را بسازید و هم آنها را استخراج نمایید.

**چگونه یک Cookie بسازید؟**

Response.Cookie برای ساختن Cookie استفاده می‌شود.

**توجه** Response.Cookie: باید قبل از تگ درج شود.

در مثال زیر، ما یک Cookie درست خواهیم کردم و اسم آن را "firstname" می‌گذاریم و مقدار "Alex" را به آن می‌دهیم:

همچنین امکان دارد که خصوصیتی را مثل تاریخ انقضای cookie را به آن بدهید:

چگونه مقدار cookie ها را استخراج کنیم

دستور "Request.Cookies" برای استخراج اطلاعات از cookie ها استفاده می‌شود.

در مثال زیر، ما مقدار cookie ی با نام "firstname" را استخراج کرده و در صفحه نشان دادیم:

خروجی:

Firstname=Alex

**یک Cookie با کلید**

اگر یک cookie حاوی مجموعه‌ای از مقادیر باشد، می‌گوییم آن cookie دارای کلیدهایی است.

در مثال زیر، ما یک مجموعه cookie با نام "user" درست می‌کنیم cookie. به نام "user" دارای کلیدهایی است که حاوی اطلاعاتی راجع به کاربر است.

```

<%
Response.Cookies("user")("firstname")="John"
Response.Cookies("user")("lastname")="Smith"
Response.Cookies("user")("country")="Norway"
Response.Cookies("user")("age")="25"
%>

```

خواندن همه Cookie ها به کد زیر نگاه کنید:

```

<%
Response.Cookies("firstname")="Alex"
Response.Cookies("user")("firstname")="John"
Response.Cookies("user")("lastname")="Smith"
Response.Cookies("user")("country")="Norway"
Response.Cookies("user")("age")="25"
%>

```

فرض کنیم که سرور شما همه cookie های بالا را به کاربر فرستاده است

حالا ما می‌خواهیم تمام cookie ها را که به کاربر فرستاده شده است را بخوانیم. مثال زیر نشان می‌دهد که چگونه این کار را انجام دهیم (توجه داشته باشید که در مثال پایین با استفاده از HasKeys چک می‌کند که آیا cookie کلید دارد یا نه؟

```

<html>
<body>
<%
dim x,y
for each x in Request.Cookies
response.write("<p>")
if Request.Cookies(x).HasKeys then
for each y in Request.Cookies(x)
response.write(x & ":" & y & "=" & Request.Cookies(x)(y))
response.write("<br />")
next
else
Response.Write(x & "=" & Request.Cookies(x) & "<br />")
end if
response.write "</p>"
next
%>
</body>
</html>

```

خروجی :

```

firstname=Alex user:firstname=John user:lastname=Smith user: country=Norway
user: age=25

```

### اگر Browser دارای پشتیبانی Cookie نباشد چه اتفاقی می افتد؟

اگر برنامه شما با Browser ی سر و کار داشته باشد که cookie را پشتیبانی نمی کند، باید از روش های دیگری برای انتقال اطلاعات از یک صفحه به صفحات دیگر در برنامه خود استفاده کنید. دو راه برای انجام این کار هست:

#### 1- اضافه کردن پارامترها به URL

شما می توانید پارامترها را به URL اضافه کنید:

```

<a href="welcome.asp?fname=John&lname=Smith">
Go to Welcome Page</a>

```

و مقادیر موجود در "welcome.asp" را همچون مثال زیر استخراج کنید:

#### 2- استفاده از form

شما می توانید از فرمها استفاده کنید. وقتی که کاربر روی دکمه Submit کلیک کند فرم ورودی های کاربر را به "welcome.asp" رد می کند:

```

<form method="post" action="welcome.asp">
First Name: <input type="text" name="fname" value="">
Last Name: <input type="text" name="lname" value="">
<input type="submit" value="Submit">
</form>

```

مقادیر موجود در "welcome.asp" را همچون این مثال استخراج کنید:

```

<%
fname=Request.form("fname")
lname=Request.form("lname")

```

```
response.write("<p>Hello " & fname & " " & lname & "!</p>")
response.write("<p>Welcome to my Web site!</p>")
%>
```

## فرم ها در ASP

مثالها

### یک فرم با متد "get"

چگونه با استفاده از دستور Request.QueryString به کاربر عکس العمل نشان دهید.

### یک فرم با متد "post"

چگونه با استفاده از دستور Request.Form به کاربر عکس العمل نشان دهید.

### یک فرم با Radio Button ها

چگونه با استفاده از دستور Request.Form و از طریق Radio Button ها به کاربر عکس العمل نشان دهید.

### ورودی کاربر

آبجکت Request می تواند برای گرفتن اطلاعات از کاربر از فرمها استفاده گردد.

```
<form method="get" action="simpleform.asp">
First Name: <input type="text" name="fname">
<br />
Last Name: <input type="text" name="lname">
<br /><br />
<input type="submit" value="Submit">
</form>
```

ورودی های کاربر می توانند از دو طریق استفاده گردد: با Request.QueryString یا Request.Form

**Request.QueryString**

دستور Request.QueryString برای جمع کردن مقادیر موجود در یک فرم با متد "get" استفاده می شود. اطلاعات فرستاده شده از فرمی با متد "get" برای هر کسی قابل مشاهده است (این اطلاعات در قسمت آدرس Browser نشان داده می شود) و در مورد مقدار اطلاعات محدودیت دارد.

اگر کاربر در فرمی با متد "get" مقدار "Bill" و "Gates" را وارد کنید، URLی که به سرور فرستاده می شود شبیه به این خواهد بود:

<http://www.irandevlopers.com/simpleform.asp?fname=Bill&lname=Gates>

فرض کنید فایل "simpleform.asp" حاوی اسکریپت زیر است:

```
<body>
Welcome
<%
response.write(request.querystring("fname"))
response.write(" " & request.querystring("lname"))
%>
</body>
```

Browser خط زیر در قسمت Body سند نشان خواهد داد:

Welcome Bill Gates

Request.Form

دستور Request.Form برای جمع کردن اطلاعات از یک فرم با متد "post" استفاده می شود. اطلاعات فرستاده شده از فرمی با متد POST برای بقیه غیرقابل مشاهده است و هیچ محدودیتی در مقدار اطلاعات فرستاده شده ندارد.

اگر کاربر در فرمی با متد "post" کلمه "Bill" و "Gates" را تایپ کند، URLی که به سرور فرستاده می شود شبیه به این خواهد بود:

<http://www.w3schools.com/simpleform.asp>

فرض کنید که فایل "simpleform.asp" حاوی اسکریپت زیر باشد:

```

<body>
Welcome
<%
response.write(request.form("fname"))
response.write(" " & request.form("lname"))
%>
</body>

```

Browser خط زیر در قسمت Body سند نشان خواهد داد:

Welcome Bill Gates

### تصدیق فرم یا تست معتبر بودن (درست بودن) مقادیر ورودی

ورودی کاربر باید در Browser در هر جایی که امکان داشت (با استفاده از اسکریپت‌های client تصدیق گردد. تصدیق Browserی سریعتر است و زمان load شدن سرور را نیز کاهش می‌دهد

در صورتی که مقادیر ورودی کاربر وارد Database می‌شود لازم است که معتبرسازی یا تصدیق ورودی‌های کاربر بر روی سرور انجام شود. یک راه خوب برای تصدیق ورودی‌های کاربر، به جای فرستادن اطلاعات به یک صفحه دیگر، پست کردن آنها به خود فایل است. در آن صورت کاربر می‌تواند خطاها را در خود صفحه‌ای که فرم در آن قرار دارد مشاهده نماید. این پیدا کردن خطا را برای وی ساده‌تر می‌کند.

### برنامه کاربردی تحت وب چیست ؟

یک برنامه کاربردی تحت وب بایک وب سایت تفاوت دارد . یک وب سایت شامل اطلاعات ثابتی است که از پیش آماده شده و به صورت فایل های HTML ذخیره شده اند . در این حالت جهت حرکت اطلاعات از سرویس دهنده به سرویس گیرنده می باشد . در واقع وب سایت اطلاعات را به کاربر ارائه می دهد .

بین درخواست ها ، سرویس دهنده از عملکرد سرویس گیرنده اطلاعی ندارد . سرویس گیرنده می تواند از یک محل به محل دیگر ببرد و این کار روی وب سایت اولیه هیچ تاثیری ندارد ، چون هر یک از صفحات به صورت یک واحد کاملاً جدا می باشند . در واقع وب سایت شامل گروهی از فایل های HTML از هم جدا می باشد .

در مقابل ، یک برنامه کاربردی تحت وب ، اطلاعات را برای یک کاربر یا گروهی از کاربران به شکلی خاص فرمت کرده و ارائه می دهد . در این حالت ، جهت حرکت اطلاعات دو طرفه می باشد ؛ محتویاتی که توسط مرور گر نمایش داده می شوند ، بر حسب اطلاعات ورودی کاربر یا هویت وی تعیین می گردند .

بنابر این یک برنامه کاربردی تحت وب، اطلاعات را به صورت دینامیکی ارائه می دهد نه به صورت فایل های HTML ثابت. در واقع وظیفه برنامه، فراهم کردن اطلاعات بر حسب شرایط است که این اطلاعات معمولاً از پایگاه داده (Database) بدست می آید. برنامه کاربردی نه تنها اطلاعات می دهد، بلکه می تواند از کاربر اطلاعات بگیرد و بکارهای مختلف پاسخ مناسب دهد. به عبارت دیگر، یک برنامه کاربردی، عملکرد کاربر را از یک درخواست به درخواست دیگر ردیابی می کند. بنابراین کاربر می تواند آن گونه که در یک وب سایت به جستجو و گردش می پردازد، به شکل دلخواه تمام صفحات یک برنامه کاربردی را بررسی و مشاهده کند. بلکه این برنامه کاربردی است که صفحات مناسب را در شرایط مقتضی به طور دینامیکی تولید کرده و به کاربر ارائه می دهد.

برای نوشتن برنامه های کاربردی تحت وب از زبان های مختلفی می توان استفاده کرد، مثل Perl, Python, Visual Basic, C, C++, Java, Quick Basic یا Script های Unix.

در هر صورت مواردی که در زیر به آنها اشاره می شود جزء ضروریات یک برنامه کاربردی هستند:

-ارتباط با یک پایگاه داده

-سرعت

-کار با رشته ها

-امنیت (Security)

-تراکنش ها (Transactions)

### تراکنش چیست؟

یک برنامه کاربردی تحت وب اغلب باید کارهای مختلفی انجام دهد که هر یک از آنها برای تداوم اعتبار اطلاعات باید به طور کامل انجام شوند. هنگام کار با یک پایگاه داده، معمولاً این کارها را بصورت یک تراکنش در نظر می گیریم. یک تراکنش قرار دادی است که موفقیت یا عدم موفقیت مجموعه ای از عملیات را تضمین می کند که در صورت عدم موفقیت، اطلاعات باید دست نخورده باقی بمانند. به عنوان یک نمونه کلاسیک، می توانیم یک حساب چک (Checking Account) را در نظر بگیریم.

هنگامی که یک چک را به حساب خود می گذارید، بانک مبلغ مشخص شده را از حساب مربوطه برداشته و به حساب شما واریز می کند. کل این عملیات در صورتی موفقیت آمیز خواهد بود که برداشت و واریز پول به طور کامل انجام شود. در این حالت عمل دو گانه برداشت از یک حساب و واریز به یک حساب دیگر را یک تراکنش می گوئیم.

### تفاوت میان برنامه های IIS و DHTML

برنامه های IIS، برنامه هایی هستند که بر روی سرویس دهنده و تحت Microsoft Internet Information Server اجرا می شوند. در مقابل، برنامه های DHTML برنامه Microsoft Internet Explorer استفاده می کنند و به این ترتیب به ورودی های کاربر پاسخ سریع می دهند. این برنامه ها برای نوشتن بازی ها و شبیه سازی محیط و گرفتن اطلاعات از کاربر مناسب هستند. متأسفانه این نوع برنامه

با توجه به این که برنامه های IIS بر روی سرورس دهنده اجرا می شوند ، برای به روز در آوردن آنها تنها کافی است کد موجود بر روی سرورس دهنده را تغییر داد. این نوع برنامه ها از هر محلی قابل دسترسی می باشند . هنگامی که یک برنامه IIS بر روی سرورس دهنده نصب شده و به اجرا در آمد ، هر کامپیوتری که بتواند به آن سرورس دهنده متصل شود می تواند از آن برنامه استفاده کند .

شما می توانید برای شناسایی کاربران ، به آنها نام و کلمه عبور اختصاص دهید و هنگام برقراری ارتباط با برنامه کاربردی ، نام و کلمه عبور را در خواست کنید . با توجه به این که تمام اطلاعات بر روی یک کامپیوتر ذخیره می شوند ، هنگامی که کاربر برنامه کاربردی را ترک کند ، هیچ نوع اطلاعاتی را از دست نمی دهد .

در قسمت بعد ، در مورد چگونگی کارکرد برنامه های IIS بحث خواهیم کرد .

### چگونگی در خواست فایل ها توسط مرورگر :

هنگامی که شما یک آدرس URL در فیلد آدرس مرورگر خود تایپ می کنید ، چند چیز اتفاق می افتد . مرورگر ، آدرس وارد شده را تحلیل کرده و پیغامی به سرورس دهنده نام ( Name Server ) می فرستد تا نام مشخص شده (به عنوان مثال ، ( Microsoft . com را به یک آدرس ) – IP ( Internet Protocol ) به عنوان مثال ، ( 207.84.25.32 ترجمه کند . سپس مرورگر با استفاده از آن آدرس IP به سرورس دهنده مربوطه متصل شده و فایل را درخواست می کند . سرورس دهنده فایل را می خواند و محتویات آن را به مرورگر بر می گرداند . در این مرحله مرورگر ، دستورات HTML آن فایل را تحلیل و فرمت فایل را تعیین می کند . بیشتر فایل های HTML درون خود به تصاویر گرافیکی اشاره می کنند. این اشاره ها به صورت رفرنس ( Reference ) بوده که هر یک از آنها نیز بایک آدرس URL مشخص می شوند .

بنابراین کل این عملیات برای تک تک تصاویر و یا فایل های دیگری که با رفرنس به آنها اشاره شده است تکرار می شود.

به این ترتیب پروسه نمایش یک فایل HTML شامل تعدادی تراکنش کوچک بین سرورس گیرنده ( یعنی مرورگر ) و سرورس دهنده وب می باشد .

برنامه های IIS نیز تقریباً به همین صورت عمل می کنند. ابتدا یک مرورگر یک صفحه را از سرویس دهنده درخواست می کند. این درخواست همیشه برای یک فایل خاص می باشد. پاسخ سرویس دهنده به نوع فایل درخواستی بستگی دارد. اگر فایل مورد نظر از نوع (HTML با پسوند .htm یا .html) باشد، سرویس دهنده به سادگی محتویات فایل را می خواند و محتویات آنرا به صورت یک رشته در آورده و براساس قوانین URL کد گذاری می کند و سپس رشته حاصله را به مرورگر برمی گرداند. کل این پروسه، از درخواست تا پاسخ، یک تراکنش مابین سرویس گیرنده و سرویس دهنده تلقی می شود. سرویس گیرنده همواره تراکنش را شروع می کند و سپس تا زمان رسیدن پاسخ از سوی سرویس دهنده منتظر می ماند که در این زمان تراکنش کامل شده و خاتمه می یابد.

درخواست فایل از سرویس دهنده وب مشابه هنگامی است که در Windows Explorer روی یک فایل شبکه ای (فایلی که بر روی کامپیوتر سرویس دهنده قرار دارد) دوبار - کلیک می کنید؛ البته با دو تفاوت:

- سرویس دهنده وب هیچگاه اجازه نمی دهد که برنامه محلی شما (یعنی مرورگر) فایل درخواستی را باز کرده یا در آن چیزی بنویسید؛ در عوض خودش فایل را باز می کند و محتویات آن را برمیگرداند.

- ارتباط مابین سرویس گیرنده و سرویس دهنده گذرا (Transient) است. برای برقراری ارتباط با سرویس دهنده وب نیازی به مشخص کردن نام درایوی که فایل بر روی آن قرار دارد نیست. پس از این که سرویس دهنده وب به درخواست شما پاسخ داد، ارتباط را قطع می کند.

هنگامی که تراکنش کامل شد، سرویس دهنده وب، سرویس گیرنده را به طور کلی از یاد می برد. اگر بلا فاصله کلید (Refresh) موجود بر روی نوار ابزار (IE) را کلیک کنید. سرویس دهنده وب، آن تراکنش را تکرار می کند و اصلاً به خاطر نمی آورد که 5 ثانیه قبل این فایل توسط شما درخواست شده است.

اگر ارتباط براساس پروتکل استاندارد HTTP برقرار شود، بیشتر فایل ها به چندین تراکنش نیاز دارند، یکی برای فایل Html اصلی و بقیه برای یکایک فایل های گرافیکی که در فایل اصلی به آنها اشاره شده است. بنابراین برای نمایش یک فایل که شامل 5 تصویر گرافیکی است، مرورگر درخواست جداگانه به سرویس دهنده وب می فرستد.

شیء RESPONSE و استفاده از فایل های include

### شیء RESPONSE و استفاده از فایل های include

برای پاسخ گویی به مرورگر، از متدهای شیء Response استفاده می شود. در واقع شیء Response راهی برای برقراری ارتباط با سرویس گیرنده است.

### شیء: Scripting Context

این شیء امکان دسترسی اشیاء ActiveX خارجی به اشیاء درونی ASP را فراهم می کند. البته بهتر است به جای این شیء از شیء Object Context استفاده کنید. برای گرفتن شیء Object Context نیز باید متد Object

هر یک از این 5 ویژگی یک اشاره گر ( Pointer ) به یکی از اشیاء درونی ASP بر می گردانند .  
 • توجه Microsoft : توصیه می کند که به جای استفاده از شیء Scripting Context از متد Object Context استفاده شود چه امکان استفاده از شیء Scripting Context کمکان وجود دارد، ولی باید بدانید که این شیء از رده خارج شده و در برنامه های جدید خود نباید از آن استفاده کنید.

### شیء: Object Context

شیء Object Context به عنوان پل ارتباطی با ( Microsoft Transaction Server ) MTS تلقی می شود. با MTS شما می توانید کاری کنید که اشیاء ActiveX در تراکنش هایی که صفحه ASP آغاز شده اند شرکت کنند. همچنین با استفاده از شیء Object Context در هنگام نوشتن اشیاء ActiveX می توانید به اشیاء درونی ASP دسترسی پیدا کنید .

### آشنایی با ساختار یک برنامه ASP

از نظر موتور ASP ، یک برنامه کاربردی عبارت است از مجموعه ای از تمام فایل و زیر دایرکتوری های ( Subdirectory موجود در یک دایرکتوری که شامل فایل Global . asa باشد . بیشتر برنامه های کاربردی ASP شامل فایل های ASP و فایل های ( Include با پسوند .inc هستند که هر دوی آنها می توانند ترکیبی از HTML، کد و فایل های گرافیکی باشند. البته شما می توانید فایل های ASP را با فایل های HTML یا هر فایل دیگری که برای سرویس دهنده قابل شناسایی باشد ترکیب کنید .

بالاترین دایرکتوری در این ساختار شامل فایل Global . asa است . این فایل دایرکتوری ریشه ( Root Directory ) برنامه کاربردی را مشخص می کند .

اولین باری که یک کاربر یکی از فایل های ASP موجود در یکی از دایرکتوری های برنامه را درخواست می کند ، موتور ASP ساختار درختی درختی دایرکتوری ها را به سمت بالا طی میکند تا فایل Global . asa را پیدا کند و یا به ریشه وب ( Root Web Directory ) برسد . این نکته از اهمیت زیادی برخوردار است ، چون اگر شما چند برنامه کاربردی را درون هم قرار دهید ( که به آنها برنامه های کاربردی تودرتو - Nested Applications - می گویند ) ، فایل Global . asa که برای هر درخواست کاربر اجرا می شود بستگی دارد به این که کاربر ابتدا کدام فایل را درخواست نماید . چون نمی توان اولین درخواست کاربر را کنترل نمود ، بهتر است برنامه های ASP را درون هم قرار ندهید ؛ مگر این که دلیل موجهی برای این کار داشته باشید .

## آبجکت Response در ASP

شما نمی توانید سورس کد های ASP را بوسیله انتخاب گزینه View Source در Browser مشاهده کنید، شما فقط خروجی فایل ASP را مشاهده می کنید که HTML ساده است. زیرا که اسکریپت ها قبل از این که نتایج به Browser فرستاده شود بر روی سرور اجرا می گردد.

### قواعد اساسی نحو

یک فایل ASP به طور عادی دقیقا همانند HTML حاوی Tag های HTML است. اگر چه، یک فایل ASP می تواند همچنین حاوی Script ها نیز باشد. این اسکریپت ها به وسیله علامت های `<%` و `%>` احاطه شده اند. اسکریپت های سرور بر روی سرور اجرا می شود و می تواند حاوی هر گونه expressions ، دستورات (Statements) ، توابع، یا عملگرهایی باشد .

### آبجکت Response

متد Write مربوط به آبجکت Response برای فرستادن محتویات به Browser استفاده می شود. به عنوان مثال، دستورات زیر برای فرستادن متن "سلام دنیا!" به Browser استفاده می شود .

```
<%
response.write("سلام دنیا")
%>
```

### VBScript

این امکان وجود دارد که از زبان های مختلف اسکریپت نویسی در فایل های ASP استفاده کنید. اگرچه زبان پیش فرض، VBScript است .

```
<html>
<body>
<%
response.write("سلام دنیا!") %>
</body>
</html>
```

مثال بالا "سلام دنیا!" را در قسمت body متن می نویسد .

## JavaScript

برای این که JavaScript را به عنوان زبان پیش فرض برای یک صفحه خاص مشخص کنید می بایست از راهنماهای زبانی در ابتدای صفحه خود استفاده کنید :

```
<%@ language="javascript"%>
<html>
<body>
<%
Response.Write("Hello World!")
%>
</body>
</html>
```

**نکته** JavaScript: برعکس VBScript به بزرگ یا کوچک بودن حروف حساس است. شما مجبور خواهید بود که وقتی زبان احتیاج دارد، کد ASP خود را با حروف بزرگ یا کوچک بنویسید .

## ساختن یک برنامه کاربردی وب تحت IIS 4.0 به یک مرحله دیگر نیز دارد:

شما باید به IIS بگویید که دایرکتوری مجازی ( Virtual Directory ) در برگیرنده فایل Global . asa ، دایرکتوری ریشه برنامه کاربردی است .

صرف نظر از این که کدام فایل درخواست شود، فایل Global . asa همواره اولین فایلی خواهد بود که اجرا میشود. بنابراین اگر در نظر دارید کاربر برنامه کاربردی شما را با صفحه خاصی که مورد نظر شماست آغاز کند ، این فایل بهترین جایی است که میتوانید مرورگر را به آن صفحه هدایت کنید.

برنامه کاربردی که در شکل قبل ملاحظه فرمودید ، شامل چندین فایل ASP و دو زیردایرکتوری به نام های images و include می باشد. البته از نظر تعیین ساختار دایرکتوری های یک برنامه کاربردی هیچ محدودیتی وجود ندارد ؛ شما می

• **توجه :** دایرکتوری ریشه وب و دایرکتوری ریشه برنامه ASP لزوما نباید یکسان باشند. به عنوان مثال ، می توانید برای چندبرنامه کاربردی ASP از یک فایل Global . asa استفاده کنید . در این شرایط می توانید فایل Global . asa را درون یک دایرکتوری قرار داده و سپس هر یک از زیر دایرکتوری های درون آن را به عنوان یک دایرکتوری مجازی تعریف کنید که بر اساس برنامه کاربردی مربوطه نامگذاری شده اند . ساختار دایرکتوری موجود در شکل زیر ، شامل 4 برنامه کاربردی است : 401 . k , Paycheck , Retirement . Timesheet

بالاترین دایرکتوری ، که HR Applications نام دارد، شامل فایل Global . asa است . هر زمان که یک کاربر به یکی از 4 برنامه کاربردی موجود متصل شود ، موتور ASP ساختار درختی دایرکتوری هارا بالا می رود تا به دایرکتوری HR Applications که در بر گیرنده فایل که Global . asa است برسد . یکی از دلایلی که می توان برای مرتب کردن برنامه ها به این شکل در نظر گرفت ، این است که ممکن است هر چهار برنامه از یک روش امنیتی استفاده کنند و یا ممکن است تمام برنامه ها از یک نوع روش برای برقراری ارتباط با پایگاه داده و یا از یک پایگاه داده مشترک استفاده کنند و شما بخواهید اطلاعات اولیه را در فایل Global . asa تنظیم کنید .

### استفاده از فایل های : Include

برای جلوگیری از تکرار کد یا دستورات HTML در صفحات ASP ، می توانید از فایل های خارجی استفاده کرده و آنها را در صفحات ASP خود بگنجانید . یک فایل Include در یک فایل ASP از دستور INCLUDE استفاده می شود . دستور INCLUDE مشابه این است که محتویات فایل include مورد نظر را درون یک صفحه دیگر قرار دهید. در این جا مثالی از یک دستور INCLUDE را مشاهده می کنید :

```
<!--# INCLUDE = "C:\include\my Include .inc" -->
```

این دستور سبب می شود که سرویس دهنده آن را با محتویات فایل مشخص شده جایگزین کند. دو نوع دستور INCLUDE موجود است : یکی INCLUDE FILE # و دیگری INCLUDE VIRTUAL # نوع اول برای مشخص کردن فایل به یک آدرس فیزیکی نیاز دارد در حالی که نوع دوم به آدرس مجازی به فایل اشاره می کند . به هر حال از هر روشی که استفاده کنید، موتور ASP قبل از شروع پردازش دستورات ، آن را با محتویات فایل مورد نظر

```

< %
if my Var = True than
% >
< ! -- # INCLUDE FILE = " C: \ include \ my Include . inc" -- >
< %
else
% >
< ! -- # INCLUDE FILE = " C: \ include \ your Include . inc" -- >
< %
end if
% >

```

هنگامی که موتور ASP این فایل را تحلیل می کند ، دو دستور INCLUDE را با فایل های نامبرده جایگزین می کند و این کار را قبل از اجرای دستور if...end if انجام می دهد . پس باز هم تاکید می کنم : شما با استفاده از کد نمی توانید برای موتور ASP تعیین کنید که کدام فایل را بگنجانند و یا این که اصلا یک فایل را بگنجانند یا خیر

### فرستادن فایل یا Upload از صفحه وب

طریق صفحه وب ( Upload ) به یک سرویس دهنده می پردازیم . این کدها از مجموعه موجود در سایت گرفته شده اند . در ابتدا باید هر دو فایل زیر را بسازید :

**upload\_page.asp** :

```

<!-- #include file="upload_class.asp" -->
<%
Server.ScriptTimeout = 300 'now i can upload and s
Dim intLevel, intUpload, intSave, strError, strContenT
Dim objUpload
Dim lngTime, lngUploadTime, lngSaveTime
intLevel = Request.QueryString("level")

```

**upload\_class.asp** :

```
<%
+-----+
|'Class:  FileUpload |
|'Date:      11:01 PM 7/23/2002|
|'By:        M.Meijer |
|'Version: 1.0 |
+-----+

```

حالا کافیسست در فایل اول در خط 15 مسیر را برای ذخیره فایلها روی سرورس دهنده خود مشخص کنید :

```
;"Path = "D:\inetpub\wwwroot\test.
```

مسیر تعیین شده باید حتماً موجود باشد

### ارسال ایمیل با کمک ASP

استفاده از آبجکت CDONTS که یکی از کامپوننتهای ویندوز سرور 2000 و ان تی می باشد به راحتی می توان در

اسکرپتتهای سمت سرور

ایمیل ارسال کرد.

طریقه استفاده به شکل زیر می باشد

```
<%
Dim objNewMail
Set objNewMail = Server.CreateObject("CDONTS.NewMail")
%>
```

مقادیر قابل تنظیم برای این آبجکت به شرح جدول زیر می باشد.

یک مقدار رشته ای که آدرس پست الکترونیکی فرستنده را مشخص می کند (به طور مثال <a href="mailto:Me@somewhere.com">Me@somewhere.com</a> )	From
---	------

<p>کند یک مقدار رشته ای که آدرس پست الکترونیکی گیرنده را مشخص می کند.</p> <p>(به طور مثال, <a href="mailto:someone@somewhere.com">someone@somewhere.com</a>)</p> <p>از علامت روبرو امکان پذیر میباشد آدرس گیرنده های مختلف با استفاده ";"</p>	To
عنوان پیام	Subject
متن پیام	Body
آدرس پست الکترونیکی شخصی که قرار است یک کپی از نامه را دریافت کند	Cc
آدرس الکترونیکی شخصی که قرار است یک کپی از نامه را دریافت کند بدون اطلاع دریافت کننده نامه	Bcc
یک مقدار عددی که نشان دهنده درجه اهمیت نامه می باشد (مثال خیلی مهم , مهم , عادی برای)	Importance
<p>باشد یک مقدار عددی که نشان دهنده فرمت اطلاعات متن نامه می</p> <p>ObjMail.BodyFormat = 0 (HTML format)</p> <p>ObjMail.BodyFormat = 1 (عادی متن)</p>	BodyFormat
<p>باشد یک مقدار عددی که نشان دهنده فرمت اطلاعات نامه می</p> <p>ObjMail.MailFormat = 0 (Mime format)</p>	MailFormat

--	--

### متودهای این آبجکت

این متود یک فایل را به نامه پیوست می کند	AttachFile
این متود آدرس فایل پیوست شده را تخصیص می دهد	AttachURL
این متود نامه را ارسال می کند	Send

بعد از ارسال ایمیل مقادیر در حافظه باقی می ماند و شما باید حافظه را خالی کنید. همچنین برای ارسال یک ایمیل جدید باید یک آبجکت جدید

ایجاد کرده و از آن استفاده کنید. کد زیر نحوه استفاده از این آبجکت را در asp نشان می دهد.

```
<%
```

```
Option Explicit
```

```
Dim objNewMail
```

```
' ابتدا یک آبجکت از نوع cdots ایجاد می کنیم
```

---

---

### درج تاریخ شمسی

تا به حال الگوریتم‌های زیادی برای نمایش تاریخ جاری بر اساس تقویم جلالی ارائه شده‌اند . در زیر یک نمونه ساده ولی کارآمد آن ارائه می شود . این نمونه تاریخ Server را نشان می دهد . برای اینکه تاریخ از روی کامپیوتر دیدکننده محاسبه شود کافیست `runat=server` را از کد حذف کنید .

```
<script language="vbscript" runat="server">
Function IranDate()
Dim if day,if month,if year,if dayof year
Dim iyear,idayof year
Dim inumdayof year
Dim aif monthdays
aif monthdays = Array(31,31,31,31,31,31,31,31,31,31)
inumdayof year = 365
iyear =year(date)

```

## ارسال ایمیل در ASP ساده و کارآمد

با این کد ساده می توان صفحه‌ای برای ارسال ایمیل ایجاد کرد. این کد را یک هندی به نام مستعار Prince Joseph نوشته است و آنطور که گفته ملزومات سیستمی آن برای اجرا عبارتند از: یک ویندوز نسخه سرویس دهنده و SMTP Service بر روی IIS. این دو صفحه با استفاده از کامپوننت یا ابزار CDO که توسط شرکت مایکروسافت ارائه شده کار می کنند.

### : mailSend.htm

```
<html><body>
<form id=frmMail method=post action=mailSend.asp>
TO:<input type=text name=txtTo size=30><br>
FROM:<input type=text name=txtFrom size=30><br>
SUBJECT:<input type=text name=txtSubject size=30><br>
BODY:<textarea rows="5" name=txtBody cols="28">
</textarea><br>
<input type=submit value="Send the Mail">
</form>
</body></html>
```

mailSend.asp :

```
<%@ Language=VBScript %>
<%
Dim objMail
Set objMail = CreateObject("CDONTS.NewMail")
objMail.From = Request.Form("txtFrom")
objMail.To= Request.Form("txtTo")
objMail.Subject= Request.Form("txtSubject")
objMail.Body= Request.Form("txtBody")
objMail.Send
Set objMail = Nothing
```

Response.Write "Mail should have been sent"

%&gt;

### ورود اطلاعات فارسی با استفاده از جاوااسکریپت

اگر قبلا اقدام به استفاده از زبان فارسی در کدهای خود و مخصوصا هنگام کار با Microsoft Access کرده باشید به احتمال زیاد با مشکلات موجود بر سر راه آشنا هستید. یکی از مشکلات اساسی بر سر راه استفاده از زبان فارسی در بانکهای اطلاعاتی Microsoft Access در نحوه ورود اطلاعات می باشد. برای درک بهتر این موضوع اینگونه عمل نمایید:

در ویندوز خود قابلیت فارسی نویسی را فعال نمایید و پس از آن در Access یک Table با نام test و با یک فیلد به نام test از نوع text ایجاد نمایید. سپس با استفاده از خود Access اطلاعاتی به زبان فارسی وارد نمایید. سپس از Access خارج شده و با استفاده از ASP به بانک اطلاعاتی Access متصل شده و اطلاعات وارد شده را خوانده و نمایش دهید. مشاهده خواهید نمود که اطلاعات نمایش داده شده داخل مرورگر با اطلاعات وارد شده توسط شما در داخل Access متفاوت می باشد.

### برای حل این مشکل چه باید کرد؟

برای حل این مشکل باید اطلاعات از خارج از Access و با استفاده از ASP و یک تابع Javascript که صفحه کلید را در همه ویندوزها اعم از فارسی و انگلیسی، فارسی نموده و امکان تایپ فارسی را فراهم میکند، وارد شوند. روش کار به این صورت است که در یک فرم HTML و در داخل یک Input Box هنگام فشردن هر کلیدی از صفحه کلید، تابع Javascript فرا خوانده شده و معادل فارسی حرف وارد شده را نمایش میدهد .

برای نمایش حروف فارسی لازم است که از charset=windows-1256 استفاده نمایید. فونت صفحه را نیز یکی از دو فونت Times New Roman و یا Tahoma انتخاب نمایید. این دو فونت فونتهای استاندارد ویندوز بوده و برای دیده شدن متون فارسی نیازی به نصب فونت جدید روی سیستم کاربر نخواهد بود Javascript. استفاده شده در زیر آمده است:

پس از طراحی فایل HTML نوبت به ایجاد فایل ASP می رسد. در فایل ASP باید اطلاعات وارد شده در فرم HTML را دریافت نموده و در بانک اطلاعاتی Access توسط برنامه خود بنویسید. نکته جالب این است که اگر وارد Microsoft Access شوید و اطلاعات وارد شده توسط برنامه خود را مشاهده نمایید اثری از حروف فارسی مشاهده نخواهید کرد.

حال اگر دوباره توسط برنامه ASP اولیه به بانک اطلاعاتی متصل شوید و اطلاعات دریافت شده را داخل مرورگر نمایش دهید اطلاعات به صورت فارسی نمایش داده میشوند. لازم به ذکر است که باید از `charset=windows-1256` در فایل ASP که اطلاعات فارسی را نمایش میدهد استفاده نمایید.

### یک شمارنده ساده یا کونتور

این برنامه یک مثال خوب از نحوه باز کردن ، خواندن و نوشتن داخل یک فایل است که نیاز به هیچ نوع بانک اطلاعاتی این فایل را با اسم `counter.asp` ذخیره کنید و در کنارش یک فایل متنی ساده به کمک `NotePad` و به نام `database.inc` ایجاد کنید و در سطر اولش هم عدد صفر یا هر عدد دیگری را که مایلید پیش فرض شمارنده شما باشد بنویسید. فراموش نکنید که سطح دسترسی لازم برای نوشتن را نیز به این فایل بدهید. با کمی ابتکار می توان این شمارنده را به نوع گرافیکی تبدیل کرد. حتی می توانید به تعداد صفحات سایت ، فایل ایجاد کنید تا تعداد مشاهده هر صفحه را به تنهایی ردیابی کنید .

```
<%
Dim objFSO , objFile
set objFSO = createobject("scripting.filesystemobject")
set objFile = objFSO.opentextfile(server.mappath("database.inc"))
'get the current count and store in counter
counter = cLng(objFile.readline)
'add one to the current count
counter = counter + 1
'close file object
objFile.close
'write a new text object with the same name and new count
Set objFile = objFSO.CreateTextFile(server.mappath("database.inc"), true)
objFile.writeline(counter)
'close file object
objFile.close
'

counter_length = len(counter)
'loop through the hit count and display the image
for i = 1 to counter_length
%>
<%= mid(counter,i,1) %>
<%
next
%>
```

## آشنائی با روش کار صفحات ASP و مقایسه آن با HTML به همراه مثال

بهترین راه درک روش کار ASP مقایسه آن با یک صفحه HTML است. تصور کنید که شما می خواهید یک صفحه HTML واقع در آدرس `http://www.somewhere.com/page.html` را ببینید. مراحل زیر را خواهیم داشت:

- ۱- ابتدا آدرس فوق را در محل آدرس مرورگرتان وارد می کنید.
- ۲- مرورگرتان درخواست شما را به وب سرور مربوطه می فرستد.
- ۳- وب سرور صفحه مورد نظر را از دیسک سخت (یا حافظه) خود خوانده و آن را به مرورگر شما می فرستد.
- ۴- مرورگر صفحه دریافتی را نمایش می دهد.

همانگونه که مشاهده کردید مراحل فوق اعمال ساده ای هستند و پردازش خاصی صورت نمی گیرد. حال فرض کنید می خواهید یک صفحه ASP به آدرس `http://www.somewhere.com/page.asp` را ببینید:

- ۱- ابتدا آدرس فوق را در محل آدرس مرورگرتان وارد می کنید.
- ۲- مرورگرتان درخواست شما را به وب سرور مربوطه می فرستد.
- ۳- وب سرور صفحه مورد نظر را از دیسک سخت (یا حافظه) خود می خواند.
- ۴- صفحه یافت شده توسط وب سرور از ابتدا تا انتها پردازش شده و هر دستورالعملی که در صفحه یافت شود اجرا می گردد و معادل آن کد HTML تولید شده و به سوی مرورگر شما ارسال می گردد.
- ۵- مرورگر صفحه دریافتی را نمایش می دهد.

مرحله اضافی در اینجا نسبت به حالت قبل مرحله شماره ۴ می باشد و تفاوت اساسی میان صفحات ASP و HTML را بیان می کند. لازم به ذکر است که پردازشگر صفحات ASP در وب سرور فایل یا برنامه `ASP.dll` در می باشد و کلیه دستورالعملهای ASP بین دو علامت `<` و `>` قرار می گیرند. جهت درک بهتر مطالب فوق به مثال ساده زیر توجه نمائید:

```
<%For I = 1 To 5%>
<Font Size="<%=I%">Hello World </Font><br>
<%Next%>
```