

Active Server Page چیست؟

با گذشت سالهای متمادی، ما تغییرات مهمی را در موضوع اینترنت مشاهده می نمایم. ابتدا، اینترنت به منظور یک سرویس دنده ارتباطات متوسط برای بخشهای دولتی و مؤسسات دانشگاهی بود. با پیدایش WWW، اینترنت یک محیط دوستانه چند رسانه ای شد. در اصل سرویس اینترنت به عنوان یک مکان برای آدمهای با ذوق به منظور خلق صفحات خانگی بود، اما هرچه افراد بیشتری به اینترنت وارد شدند، اینترنت به یک منبع اطلاعاتی برای همه افراد تبدیل شد. وقتی تعداد افراد online، به یک محدوده بحرانی رسید، شرکتهای که محصولات و خدمات را می فروختند شروع کردن به spring up نمودن. این شرکتهای هیچ حضور فیزیکی نداشتند، فقط یک محضور مجازی داشتند. برای مثال شما می توانید از سایت Amazone کتاب بخرید، اما شما قادر نخواهید بود که کتابفروشی Amazone.com را در همسایگی خودتان پیدا کنید.

همان گونه که اینترنت به صورت یک فروشگاه موفق شد، طراحی وب سایت هم تغییر پیدا کرد. در روزهای اولیه www، HTML به منظور ایجاد صفحات وب ایستاستفاده می شد. اما امروزه، صفحات وب ایستا به سرعت در حال منسوخ شدن می باشند. تصور کنید اگر Amazon.com تنها از صفحات وب ایستا تشکیل شده بود - شما نمی توانستید فهرستهای کتاب را جستجو کنید. شما نمی توانستید یک درخواست را به صورت online صادر کنید، شما نمی توانستید توضیحات کاربران دیگر را بخوانید. این است دلیل یک سرمایه گذاری مطمئن که اگر Amazon.com از صفحات وب پویا استفاده نمی کرد، نمی توانست تعداد زیادی کتاب بفروشد:

شما می توانید صفحات وب را از راههای زیاد ایجاد نمایید. راه حل مایکروسافت برای ساختن صفحات وب پویا استفاده از Active Server Page می باشد که به صورت مختصر بر آن ASP می گویند.

سایت های وب بزرگ زیادی از ASP به منظور سرویس دادن پوسا به اسناد وب استفاده می ک. برای مثال Hotbot.com.Buy.com و

Dell.com از ASP برای ساخت سایت های وب پویای محاوره ای خود استفاده کرده اند.

ASP شامل دو بخش می باشد. کدهای برنامه نویسی و embedded HTML. کدهای برنامه نویسی می توانند به وسیله تعدادی از

زبانهای دست نویس (scripting language) نوشته شوند.

یک زبان دست نویس (scripting language) یک دستور زبان مخصوص است که برای اجرای دستورات بر روی کامپیوتر استفاده می شود.

زبانهای اسکریپت نویسی دست نویس مربوط به وب که محبوب می باشند عبارتند از VBScript و JavaScript. وقتی یک صفحه ASP درست می کنید، شما می توانید از یکی از چهار زبان برنامه نویسی استفاده کنید. شبیه دستور زبان Visual Basic.

* VBScript: معمولترین برنامه نویسی Script برای ASP می

باشد.

* Jscript: شبیه javascript می باشد.

* Perlscript: شبیه به زبان Perl می باشد.

* Python: یک زبان اسکریپت نویسی قدرتمند می باشد که به طور

معمول برای توسعه وب به کار می رود.

بیشتر صفحات ASP با استفاده از Vbscript ایجاد می گردند.

Vbscript از چهار زبان دیگر به زبان انگلیسی نزدیکتر می باشد و

دستور زبان آن شبیه Visual Basic می باشد که تعداد زیادی از

توسعه دهندگان وب از آن استفاده کرده اند.

اجرای صفحات ASP

برای اجرای صفحات ASP بروی کامپیوترتان، احتیاج به اجرای یک

سرویس دهنده web دارید. محصولات رایگان شرکت میکروسافت

در دسترس شما قرار دارند که به شما اجازه می دهند یک سرویس

دهنده وب را روی ویندوز 95 و 98 به همان خوبی windowsNT و windows2000 اجرا کنید.

اگر سرویس دهنده وبی که صفحات ASP را اداره می کند نصب کرده اید، هنگامی که یک درخواست صفحه ASP را داخل مرورگر خود می نمایید، یک پنجره گفتگو را مشاهده می کنید که از شما می پرسد که آیا می خواهید فایل ASP را بر روی دیسک ذخیره نمایید یا

خیر؟

به یاد بیاورید که یک سرویس دهنده وب با توانایی مدیریت صفحات ASP، صفحه وب درخواستی را قبل از اینکه سرویس گیرنده بفرستند پردازش می کند.

در طی این پردازش دو چیز اتفاق می افتد:

۱- کد برنامه نویسی در صفحه ASP به وسیله سرویس دهنده تفسیر می گردد.

۲- سرویس دهنده وب به مرورگر اطلاع می دهد که اطلاعات HTML را فرستاده است و مرورگر این خطوط HTML را دریافت کرده وب رای کاربر اجرا می نماید.

اگر شما یک سرویس دهنده وب را نصب نکرده باشید، و سعی کنید که یک صفحه ASP را بوسیله مرورگرتان مشاهده نمایید، مرحله دوم اجرا نمی شود. این به آن معنی است که مرورگر مطلع نمی شود که صفحه ASP شامل کد HTML می باشد. مرورگر نمی داند که چه کاری باید با یک فایل ASP انجام دهد و سعی می کند که به وسیله برنامه های دیگر که فایل با پسوند ASP را می شناسند این فایل را ببینند.

اگر یک برنامه بر روی کامپیوترتان نصب باشد که با فایل با پسوند ASP آشنایی داشته باشد آن برنامه اجرا می گردد. برای مثال، Visual Interdev و Microsoft Frontpage پسوندهای ASP را می شناسند. اگر شما یکی از این برنامه ها را بر روی کامپیوترتان نصب کرده باشد و سعی کنید یک صفحه ASP را به وسیله

مرورگرتان مشاهده نمایید (قبل از اینکه سرویس دهنده وب را نصب کرده باشید) Visual UnterDev یا Frontpage به صورت اتوماتیک شروع به کار می کند.

تنظیمات سرویس دهنده اطلاعات اینترنت

(IIS) سرویس دهنده وب حرفه ای مایکروسافت می باشد. آخرین نسخه IIS نسخه 5.0 می باشد که با ویندوز 2000 کار می کند.

ASP 3.0 نیز با IIS 5.0 سازگار می باشد و به صورت خودکار

وقتی IIS 5.0 نصب می گردد آن نیز نصب می گردد. اگر NT

server را اجرا می کنید، به IIS4.0 احتیاج دارید و IIS4.0 را می

توانید در WindowsNT Option pack 4.0 پیدا کنید، یا می توانید

به صورت رایگان از سایت وب مایکروسافت بله آدرس

[http://www.microsoft.com/msdownload/ntoptionpack/as](http://www.microsoft.com/msdownload/ntoptionpack/askwiz.asp)

[kwiz.asp](http://www.microsoft.com/msdownload/ntoptionpack/askwiz.asp)

دریافت نمایید.

احتیاط: IIS 4.0، ASP3.0 را پشتیبانی نمی کند. اگر ویندوز NT

server را اجرا می کنید، شما IIS 4.0 را اجرا خواهید کرد. اگر IIS

4.0 را نصب کرده اید، ASP 2.0 به صورت خودکار نصب خواهد شد. نصب کردن IIS 4.0 به سادگی نصب کردن IIS 5.0 می باشد.

هر دو دارای یک ویزارد ساده می باشند که به شما در طی مراحل نصب کردن کمک می نماید.

IIS 5.0 با ویندوز 2000 ترکیب دشه است، اگر چه هنگامی که شما ویندوز 2000 را نصب می کنید به صورت پیش فرض IIS 5.0 نصب

نمی گردد. برای نصب IIS 5.0 ابتدا باید start و سپس Programs و

Administrative tools و configure your server را انتخاب نمایید.

یک جعبه محاوره ای باز می وشد که دارای چندین گزینه پیکره بندی

در پنجره سمت چپ خود می باشد. از پنجره سمت چپ، پایین ترین

گزینه labeled Advanced می باشد را انتخاب نمایید. بر روی دکمه

Advance یکبار کلیک نمایید. چهار گزینه جدید آشکار می شوند که

عبارتند از Cluster Service، Message Queuing، Support

و Tolls Optional Components. بر روی گزینه Optional

Components کلیک کنید. در پنجره سمت راست، توضیچی در مورد گزینه Components Optional ظاهر می گردد. به جلوتر بروید و برروی فرامتن start کلیک کنید. این کار ویزارد اجزای ویندوز را اجرا می نماید - windows 2000 components wizard اجزای انتخابی ویندوز ۲۰۰۰ را نمایش می دهد. از طریق این ویزارد شما می توانید اجزای انتخابی ویندوز را نصب یا رفع نصب نمایید. گزینه Internet Information Service را بیابید.

اجزای IIS تعدادی از زیر اجزای دیگری از سرویس www را همانند سرویس دهنده FTP و سرویس SMTP شامل می شوند. برای اینکه تعیین کنید که چه اجزایی از IIS نصب شوند، برروی اجزای IIS در ویزارد اجزای ویندوز 2000 کلیک کنید و سپس برروی دکمه Details کلیک نمایید. شما یک لیست از اجزایی که می توانند با IIS نصب شوند را مشاهده می نمایید. اگر شما قصد داشته باشید که از Visual Interdev یا Frontpage شرکا میکروسافت به منظور ویرایش سایت وب خود استفاده نمایید، مطمئن گردید که بسط

های Frontpage 2000 server را نصب می کنید. البته، این به نصب

کردن دیگر اجزای IIS صدمه ای نمی زند.

بعد از اینکه شما تمام اجزای IIS را انتخاب کردید، برای نصب کردن

باید برای دکمه ok کلیک کنید که شما را به ویزارد windows 2000

components برمی گرداند. برای شروع به نصب IIS 5.0 بروی

دکمه Next کلیک کنید. وقتی عملیات نصب کامل شد، شما می توانید

به ISM دسترسی پیدا کنید.

Internet server manager به شما اجازه می دهد که خصوصیات و

تنظیمات IIS را پیکره بندی نمایید.

به منظور اجرای ISM، مراحل زیر را انتخاب کنید.

Start>programs>Administrative tools>Internet Service

Manager

ISM به شما اجازه می دهد که سایت وب تان را و سرویس دهنده

FTP و سرویس SMTP را پیکربندی نمایید.

ISM ساختار دایرکتوری را برای سایت وب شما نشان می دهد. شما

می توانید خصوصیات سایت وبتان را مشاهده و اصلاح نمایید (به

وسیله کلیک دکمه راست ماوس بر روی نام سایت وب (سایت وب پیش فرض) و کلیک بر روی دکمه (Properties) این کار یک کادر به نام web site properties را باز می نماید.

این کادر همانطور که متوجه شده اید دارای 10، tab پر از گزینه می باشد. اما حالا، ما فقط بر روی تعدادی از گزینه های منتخب در داخل local path Home Directory tab متمرکز می شویم. به گزینه

توجه نمایید. دایرکتوری که در اینجا وارد می شود دایرکتوری فیزیکی ریشه می باشد و به صورت پیش فرض در داخل \Inepub\wwwroot بر روی درایوی که ویندوز 2000 نصب شده است قرار دارد. اگر شما تمایل داشته باشید، می توانید این مقدار را

تغییر دهید. اما همانطوری که بیان گردید عرف استاندارد دایرکتوری فیزیکی ریشه \Inetpub\wwwroot می باشد.

در زیر گزینه local path، یک سری از کادرهای انتخاب وجود دارد. دو تا از این کادرهای انتخابی شبیه به گزینه های موجود در کادر Personal web server properties می باشند و عبارتند از

Directory Browsing و log Visits. هر دوی این گزینه ها دقیقاً

مانند گزینه های موجود در PWS می باشند. با انتخاب Directory

Browsing شما به ملاقات کنندگانی که در URL، آدرس شما وارد

می کنند، اجازه می دهید که به اجزای دایرکتوری شما دسترسی پیدا

نمایند.

حالا که شما IIS را نصب کرده اید می توانید سایت وب خود را

مورد بازدید قرار دهید. همین طور با PWS شما می توانید به سایت

وب کامپیوترتان با تایپ کردن عبارات زیر در داخل مرورگر وبتان

دسترسی پیدا نمایید.

<http://localhost>

<http://yourserver>

استفاده از ASP بدون به کارگیری IIS یا PWS

به دلیل اینکه سرویس دهندگان وب، تنها بر روی محصولات ویندوز

شرکت مایکروسافت اجرا می گردند، شما ممکن است فکر کنید برای

استفاده از صفحات ASP حتماً باید از IIS یا PWS بر روی سرویس

دهنده وب خود استفاده نمایید. این درست نیست، زیرا که شرکتهای

زیادی نرم افزارهایی را ایجاد نموده اند که به شما اجازه می دهند صفحات ASP را بر روی سرویس دهنده های متنوع و کامپیوترهای مختلف ایجاد نمایید.

یکی از این محصولات Halycon software's Instant ASP می باشد که به صورت مخفف به آن iASP می گویند یکی دیگر از این محصولات به وسیله شرکت chili!soft ایجاد شده است و chili!ASP نامیده می شود. این محصولات می توانند بر روی سرویس دهنده هایی که IIS ندارند اجرا گردند، به علاوه محصولات زیر

- Apache
- Sun web server
- Java web server
- Netscape Eterprise server

این محصولات همچنین می توانند بر روی کامپیوترهای زیر نصب گردند

- Linux
- Sun solaris

- Apple Mac OS
- IBM/AIX

منظور از [%@language=VBScript%](#) چیست؟

در بالای اولین اسکریپت شما خطی را مشاهده می کنید که به صورت

[%@language=VBScript%](#) خوانده می شود. این خط زبانی که

در ASP استفاده می شود را مشخص می نماید.

شما ممکن است که متوجه نشده باشید که ASP یک زبان برنامه

نویسی نمی باشد. آن یک تکنولوژی سرویس دهنده می باشد. زبان

برنامه نویسی که ما در برنامه نویسی استفاده می کنیم معمولاً

VBScript می باشد.

پایگاه داده های رابطه ای چیستند؟

مفیدترین ویژگی Active server ها توانایی آنها در ارتباط ساده یک

صفحه ASP با یک بانک اطلاعاتی می باشد.

یک بانک اطلاعاتی مجموعه ای از اطلاعات است که می تواند به

راحتی مورد سؤال و تغییری قرار گیرد. وقتی از بانک اطلاعات

استفاده می کنید، می توانید کارهای زیر را انجام دهید:

دریافت داده، درج کردن داده، به روز رسانی داده های موجود یا پاک کردن داده های موجود.

تعداد زیادی برنامه های بانک اطلاعاتی تجارتي محبوب در دسترس می باشند مثل اکسس میکروسافت، که در داخل برنامه آفیس قرار دارد، میکروسافت SQL-server، اوراکل، Informix، DB2 و برنامه های دیگر. از آنجایی که ASP از تکنولوژی میکروسافت استفاده می کند، بانک اطلاعاتی میکروسافت بیشتر با آن مورد استفاده قرار می گیرند. اگرچه، هیچ دلیلی وجود ندارد که نتوانید از بانک اطلاعاتی غیر میکروسافتی مثل اوراکل با آن استفاده نمایید.

هدف اولیه از یک بانک اطلاعاتی ذخیره اطلاعات می باشد. اگرچه هر سیستم بانک اطلاعاتی ممکن است ریزه کاریهای مختص به خود را داشته باشد. اما همه بانک اطلاعاتی از جداول برای ذخیره اطلاعات استفاده می کنند.

جدول یک ماتریس دو بعدی می باشد که برای ذخیره اطلاعات در یک بانک اطلاعاتی مورد استفاده قرار می گیرد.

یک جدول که دارای سطر و ستون به شکل یک ماتریس می باشد، مانند یک انبار برای قرار دادن نمونه هایی از آبجکت ها استفاده می شود. ستونهای یک جدول خصوصیات آبجکت را تشریح می کنند، در حالی که سطر یک نمونه منحصر به فرد از آبجکت می باشد. به سطرها در جدول بانک اطلاعاتی اصطلاحاً رکورد گفته می شود، و به ستونها نیز اصطلاحاً فیلد گفته می شود.

یک رکورد یک نمونه از یک آبجکت می باشد و در یک بانک اطلاعاتی به وسیله سطر نشان داده می شود.

یک فیلد یک خاصیت تنها از یک آبجکت می باشد که در بانک اطلاعاتی به وسیله ستون جدول نشان داده می شود.

برای مثال، تصور کنید که می خواهید اطلاعاتی در باره ماشین ها در داخل بانک های اطلاعاتیتان ذخیره کنید. فکر کردن در مورد اینکه ماشین یک آبجکت می باشد که دارای خصوصیات معینی می باشد به شما کمک زیادی می نماید. (ما از یک مثال مشابه در روز ششم، «کار

با آبجکت ها» استفاده کردیم). برای اینکه قادر باشید که اطلاعات یک

نمونه بخصوصی از ماشین را ذخیره کنید، نیاز به یک جدول ماشین دارید. برای مثال، ممکن است تصمیم بگیرید که کارخانه سازنده،

سال، کیلومترها و خصوصیات رنگ را برای هر آبجت ماشین

ذخیره نمایید. در این مورد، جدول بانک اطلاعاتی ماشین هایتان چهار

ستون دارد. برای هر نمونه ماشین که می خواهید در داخل بانک

اطلاعاتی ذخیره نمایید، نیاز به افزودن یک سطر به جدول ماشین

دارد. اگر شش ماشین منحصر به فرد دارید، باید شش سطر در

جدول ماشین خود داشته باشید..

برای ذخیره کردن این شش نمونه از آبجت ماشین، نیاز به افزودن

شش سطر به جدول ماشین دارید.

بانک اطلاعاتی می توانند شامل چند جدول بانک اطلاعاتی باشند، و

معمولاً هر جدول بانک اطلاعاتی یک آبجت منفرد را نشان می دهد.

در مثال ماشین، یک ماشین را با استفاده از چهار خصوصیت تشریح

کردیم. اگر بخواهید خاصیت پنجمی را - موتور - که خودش یک

آبجت می باشد اضافه کنید، چه کار می کنید؟ آبجت موتور ممکن

است شامل خصوصیات مثل اسب بخار، لیتر و سیلندر باشد. اگر یک خاصیت آبجت، خودش یک آبجت جدید باشد، چگونه آن را در

داخل یک بانک اطلاعاتی نشان می دهید؟

یک راه ایجاد یک جدول جدید به نام، موتور می باشد، که یک سطر برای هر کدام از خاصیت های موتور دارد. با این حال، اجازه دهید به یک راه حل پیشنهادی نگاهی بیاندازیم. در عوض ایجاد یک جدول جدید، اجازه دهید سه ستون جدید در جدول ماشین اضافه نماییم.

این ستونها می توانند Engine Horse Power و Engine Liters و Engine cylinders باشند که اطلاعاتی در باره هر ماشین را در خود دارند. ما اکنون جدول موتور و جدول را با یکدیگر ادغام می

نماییم.

در روزهای اول ایجاد بانک های اطلاعاتی، ترکیب دو جدول در داخل یک جدول، یک راه حل معمولی بود. ادغام دو جدول بانک اطلاعاتی رابطه ای با یکدیگر و در داخل یک جدول تکنیمی که در سیستم بانک

اطلاعاتی flat-file استفاده می شود.

سیستمهای بانک اطلاعاتی تک فایلی، جدولهای وابسته را با یکدیگر و در داخل جدول بزرگتر ادغام می نمایند.

بانک های تک فایلی به ندرت امروزه مورد استفاده قرار می گیرند، زیرا داده هیا غیر ضروری در داخل آنها به صورت تکراری وجود دارد. در مثال موتور، تصور کنید که نیمی از ماشین های داخل جدول بانک اطلاعاتی دارای موتورهای یکسانی باشند. اطلاعات مربوط به موتورهای یکسان به صورت غیرضروری تکرار شده است، همان طور که در شکل ۳-۱۵ مشاهده می کنید. برای پاک کردن این اطلاعات تکراری، از یک فرآیند به نام نرمال سازی استفاده می کنیم. بانک اطلاعاتی نرمال سازی شده شامل هیچ داده تکراری نمی باشد.

در یک بانک اطلاعاتی رابطه ای، اطلاعات مربوط به آبجت موتور در یک جدول بانک اطلاعاتی موتور ذخیره خواهد شد. از آنجایی که آبجت موتور دارای سه خاصیت به نامهای Horse Power و Liters و cylinders می باشد - ممکن است انتظار داشته باشید که جدول موتور دقیقاً سه ستون داشته باشد. در حالی که، جدول موتور

نیاز به چهار ستون دارد. ستون چهارمی ممکن است ID نامیده شود، هر سطر در جدول موتور را به صورت منحصر به فرد تعریف می کند.

سیستم های بانک اطلاعاتی به شما اجازه می دهد که به صورت اتوماتیک فیلدهای ID افزوده شدنی را ایجاد کنید. از آنجایی که هر وقت یک رکورد جدید در داخل بانک اطلاعاتی درج می گردد، فیلد به صورت خودکار اضافه می گردد، این عمل شناسه منحصر به فرد را ایجاد می کند.

یک ستون جدید به نام، Engine ID، نیز باید به جدول ماشین اضافه گردد. ستون Engine ID در جدول ماشین نوع موتور یک نمونه از ماشین بخصوص را نمایش می دهد. Engine ID در جدول ماشین به ID یکسانی در جدول Engine نگاشت پیدا می کند.

اغین همه صحبت از IDها و Engine IDها ممکن است گیج کننده باشد. توجه کنید که وقتی چندین ماشین دارای موتور یکسانی می باشند، ستون Engine ID آنها دارای مقدار یکسانی می باشد.

بانک های اطلاعاتی رابطه ای مرسوم
تعداد زیادی از سیستم های بانک اطلاعاتی رابطه ای تجاری در
دستری می باشند. بعضی از محبوبترین آنها عبارتند از: عبارتند از:
SQL-Server میکروسافت، اکسس میکروسافت، اوراکل و
Informix و با ASP، فقط محدود به استفاده از این چهار سیستم
بانک اطلاعاتی نمی باشید. برعکس. می توانید به هر بانک اطلاعاتی
سازگار با ODBC، به وسیله ASP دسترسی پیدا کنید.

از آنجایی که هر سیستم بانک اطلاعاتی ممکن است دستور زبان
متفاوتی برای دستیابی به داده ها داشته باشد میکروسافت تلاش
کرد که یک استاندارد را ایجاد نماید. این استاندارد تحت عنوان
استاندارد open Database (ODBC) connectivity شناخته می
شود، و یک رابط کاربری عادی برای یک بانک اطلاعاتی فراهم می
کند، به این منظور که مقادیر زیادی از کدهای یکسان بدون توجه به
اینکه چه بانک اطلاعاتی را مورد پرس و جو قرار می دهید، با آن
بانک کار کنند و بانک های اطلاعاتی که این استاندارد را پشتیبانی می

کنند را ODBC-compliant می گویند، و همه سیستم های اصلی بانک اطلاعاتی امروزه ODBC-compliant را پشتیبانی می کنند.

تعدادی از بانک های اطلاعاتی OBC-compliant در زیر فهرست شده اند.

- Delimited text file
- DBASE
- Foxpro
- Access
- Oracle
- Paradox
- SQL-server

همان طوری که این لیست نشان می دهد، می توانید از محدوده

بزرگی از بانک های اطلاعاتی در نرم افزار ASP استفاده نمایید.

اگرچه تعداد زیادی از بانک های اطلاعاتی با ASP کار می کنند،

اما بانک های اطلاعاتی مرسومتر عبارتند از SQL-server

مایکروسافت، اکسس و اوراکل.

چرا از بانک های اطلاعاتی استفاده می شود؟

بانک های اطلاعاتی، مانند فایل های متنی، کوکی ها و آبجکت های session و Application دارای زمان و مکان مختص به

خوئدشان می باشند. اگر نیاز به ذخیره مقدار زیاد اطلاعات در

یک دوره طولانی از زمان را دارید، بانک های اطلاعاتی و چگونگی

ایجاد سیستم های بانک اطلاعاتی کارآمد می باشد. با داشتن

جستجوهای قدرتمند در داخل سیستم های بانک اطلاعاتی، یک

بانک اطلاعاتی مؤثرترین و کارآمدترین متد ذخیره اطلاعات می

باشد.

با این وجود، بانک اطلاعاتی دارای نقاط ضعف نیز می باشد.

اتصال به یک بانک اطلاعاتی از طریق ASP کار پرخرجی می

باشد. خواندن محتویات فایل متنی و یا یک کوکی، زمان خیلی

کمتری را از اتصال به یک بانک اطلاعاتی و دریافت اطلاعات از

جداول بانک اطلاعاتی می گیرد. به همین دلیل، بانک های اطلاعاتی

همیشه بهترین راه حل نیستند.

نکته: اجرای یک عمل پرس و جو در بانک اطلاعاتی کارآمد می باشد. ناکارآمدی زمانی اتفاق می افتد که قصد برقراری ارتباط با یک بانک اطلاعاتی از طریق صفحه ASP را دارید. اتصال به بانک های اطلاعاتی فردار مورد بحث قرار خواهد گرفت. برای کمک به اینکه اصمیم بگیرید که از یک بانک اطلاعاتی برای ذخیره کردن اطلاعات استفاده نمایید و یا متدهای دیگر ذخیره اطلاعات را به کار ببرد، با جدول زیر مشورت نمایید.

| متد | هنگامی که استفاده می شود |
|---------|--|
| کوکی ها | زمانی که قصد دارید تکه های کوچک اطلاعات را بر پایه user-by-user برای مدتهای طولانی نگهداری نمایید، تر کوکی ها استفاده نمایید. از آنجایی که کوکی ها بر روی ماشین سرویس گیرنده ذخیره می وند هیچ تضمینی وجود ندارد که کوکی های کاربران حذف نشود یا تغییر داده نشوند. اگر اطلاعاتی که که می خواهید ذخیره |

| | |
|---|-------------------------|
| <p>نمایید، برای شما حیاتی می باشند، کوکی ها بهترین انتخاب نمی باشند.</p> <p>یک موتور جستجو مانند YAHOO!، از کوکی ها برای ذخیره آخرین جستجوهای انجام شده برای هر کاربر استفاده می نماید. هنگامی که کاربر صفحه جستجو را مورد بازدید قرار می دهد، آخرین جستجوی کاربر به صورت خودکار به داخل text box وی وارد می شود.</p> | |
| <p>متغیرهای Session برای ذخیره اطلاعات بر مبنای مدل user-by-user برای مدت زمانی که کاربر از سایت شما بازدید می نماید، ایده آل می باشند. از آنجایی که متغیرهای session را فقط در طی زمانی که کاربر از سایت بازدید می نماید، نگهداری می نماید، اگر قصد دارید اطلاعات را برای مدت زمانی طولانی نگهداری نمایید، از</p> | <p>متغیرهای Session</p> |

| | |
|--|-----------------------------|
| <p>متغیرهای session استفاده نکنید. همچنین متغیرهایی session بر رنی کارایی نیز تاثیر می گذارند. زیرا مجموعه ای از متغیرهای session برای هر کاربر همزمان بر روی سایت شما ساخته می شود. یک متغیر session را می توان برای ردیابی هر صفحه وب یک کاربر ویژه که از سایت وب شما بازدید کرده است، مورد استفاده قرار می گیرد. این لیست صفحات را بعداً می توان در یک ساختار سلسله مراتبی سازماندهی کرد و به عنوان یک وسیله راهنما بکار برد.</p> | |
| <p>متغیرهای Application اطلاعات را از زمانی که سایت وب شروع به کار می کند تا زمانی که به کارش پایان می دهد، نگهداری می نمایند. تنها یک مجموعه از متغیرهای Application برای تمامی</p> | <p>متغیرهای Application</p> |

| | |
|---|----------------------|
| <p>کاربران سایت وب شما وجود دارند. مقایدر ذخیره شده در متغیرهای Application زمانی که سرویس دهنده وب شما راه اندازی مجدد می گردد از بین می روند. متغیرهای Application برای ذخیره اطلاعات ایستا و عمومی بر روی سرویس دهنده وب مناسب می باشند.</p> | |
| <p>فایل های متنی برای ذخیره تکه های کوچک اطلاعاتی که شما برای نگهداری نیاز دارید مورد استفاده قرار می گیرند. فایل های متنی کارا می باشند، اما ذخیره اطلاعات در مدل user-by-user با استفاده از فایل های متنی مشکل می باشند. فایل های متنی، زمانی بهترین کارکرد را دارند که اطلاعات ذخیره شده برای هر کار، معین نشده باشند. فرض کنید که شما بر روی سایت وب،</p> | <p>فایل های متنی</p> |

| | |
|--|--------------------------|
| <p>اسباب بازی می فروشید و لیستی از شماره کارت های اعتباری دارید که نمی خواهید آنها را بپذیرید.</p> <p>شما می توانید یک فایل متنی داشته باشید که شامل لیستی از شماره های کارت های اعتباری باشند. وقتی برای اطمینان از اینکه شماره کارت اعتباری جزو شماره هایی که در فایل متنی وجود دارد یا نه، فایل متنی را بررسی نمایید.</p> | |
| <p>بانک های اطلاعاتی بهترین کارایی را برای ذخیره قطعات اطلاعاتی بزرگ ایجاد می کنند. بانک های اطلاعاتی برای ساده کردن پردازش ذخیره و بازیابی اطلاعات طراحی می شوند. اتصال به یک بانک اطلاعاتی، و انتقال اطلاعات هزینه سنگینی دارد. بانک های اطلاعات فقط برای ذخیره مقادیر عظیم اطلاعات برای پریودهای زمانی طولانی</p> | <p>بانک های اطلاعاتی</p> |

مورد استفاده قرار می گیرند. اگر شما فقط نیاز به ذخیره اطلاعات موقتی دارید، از کوکی ها، یا متغیرهای session استفاده نمایید. اگر شما نیاز به ذخیره قطعات کوچک از اطلاعات عمومی دارید، مانند آدرس پست الکترونیکی web master، از فایل های الحاقی یا متغیرهای application استفاده کنید.

برای مثال، اگر شما نیاز به ذخیره اطلاعاتی در باره سرویس گیرنده های شرکتهایی که با آنها تجارت می کنید دارد، از بانک اطلاعاتی استفاده کنید.

بانک اطلاعاتی و ASP

سابقاً، اتصال به بانک اطلاعاتی کاری مشکل بود. بانک های

اطلاعاتی با فرمت های گوناگون به وجود آمدند، و مجبور بودند

ASP های سطحی پایین را برای هر بانک اطلاعاتی که می
خواستید از آن استفاده کنید بدانید. نیاز به یک ASP بین المللی که
بتواند با بانک های اطلاعاتی گوناگونی کار کند وجود داشت.
ODBC یا open Database connectivity به عنوان ASPی
بین المللی ساخته شد. تعداد زیادی از بانک های اطلاعاتی از این
استاندارد پیروی نمودند و به عنوان بانک های اطلاعاتی سازگار
با ODBC شناخته شدند مانند: اکسس، MS-SQL server،
اوراکل، Informix و غیره.
ODBC دارای چند عیب می باشد. ODBC همچنان دارای تعداد
زیادی فراخوانی های سطح پایین می باشد که توسعه آن را
مشکل کرده است. توسعه دهندگان هنوز مجبورند که بر روی
ارتباطات سطح پایین با بانک اطلاعاتی تمرکز نمایند. افراد روشی
را می خواستند، که اطلاعات را آن گونه که می بینند بدست
بیاورند. راه حل میکروسافت DAO با Data Access objects
بود. DAO به وسیله RDO (Remote Data object) که بریا

معماری بانک اطلاعاتی توزیعی ساخته شده) و سپس ADO (Activex Data objects) مورد متابعت قرار گرفت.

اگرچه، همه اینها دارای نقاط ضعفی می باشند. بر طبق نظر میکروسافت، «ODBC دسترسی محلی به اطلاعات SQL را فراهم می کند» و «DAO آبجکت های سطح بالا را برای اطلاعات فراهم می کند». DAO و RDO نیاز به اطلاعاتی دارند که به فرمت SQL (structured Quer language) ذخیره شده باشند.

در پاسخ به این نقاط ضعف، میکروسافت OLEDB را معرفی نمود، که یک آبجکت دسترسی به اطلاعات بر مبنای com می باشد، که دسترسی به انواعی از اطلاعات حتی دسترسی به اطلاعات ذخیره شده به صورت منفصل را فراهم می نماید (بر ای مثال، اگر اطلاعات را بر روی laptop داشته باشید، می توانید به راحتی یک تصویر لحظه ای از بانک اطلاعاتی را از آخرین زمانی که آن را هماهنگ کرده اید بدست آورید).

OLEDB تا اندازه ای ODBC را پشت سر گذاشته است و شامل یک درایور ODBC می باشد تا بتواند با همه منابع اطلاعاتی ODBC سازگاری پیدا نماید.

ارتباط با بانک اطلاعاتی با استفاده از ActiveX Data Objects

(ADO) ActiveX Data Objects (ADO) ره همراه ASP آمد و به

صفحاتتان اجازه می دهد که به راحتی به بانک های اطلاعاتی

اتصال برقرار نمایند. ADO با هر منبع OLEDB که شامل منابع

سازگار با ODBC می باشند کار می کند. بنابراین، با بیشتر بانک

های اطلاعاتی فعلی کار خواهد کرد.

مدل ADO دارای شش آبجکت می باشد. از بین این آبجکت ها،

امروزه فقط آبجکت Recirdset و connection مورد استفاده

قرار می گیرند.

آبجکت connection شما را به منبع اطلاعات متصل می نماید.

برقراری اتصال، اولین مرحله برای کار کردن با بانک اطلاعاتی

می باشد.

آبجکت Recordset به شما اجازه می دهد که با داده های موجود در جدول کار کنید. آبجکت Recordset دارای مجموعه ای از سطرهای یک جدول می باشد. این آبجکت را می توان برای خواندن سطرهای یک جدول، تغییر دادن سطرهای یک جدول، یا اضافه کردن مجموعه ای از داده های جدید به جدول استفاده نمود.

آبجکت Error یک خطای تولید شده به وسیله منبع داده ها را نشان می دهد. مجموعه ای یک خطای تولید شده به وسیله منبع داده ها را نشان می دهد. مجموعه فوق وقتی مورد استفاده قرار می گیرد که یک متد FAILED فراخوانده شده اجازه تولید چندین خطا را داشته باشد.

ارتباط با یک بانک اطلاعاتی

قبل از اینکه بتوانید کاری با یک بانک اطلاعاتی انجام دهید، باید با آن ارتباط برقرار نمایید. این ارتباط مثل برقرار کردن تماس تلفنی با یک شخص می باشد. شما نمی توانید تنها با برداشتن

گوشی تلفن شروع به مکالمه نمایید. بلکه نیاز به گرفتن شماره مورد نظر خود دارید. سپس کامپیوترهای شرکت مخابرات همه ارتباطات صحیح با جایی که قصد تلفن زدن به آنجا را دارید برقرار می نمایند. در پایان دوست شما صدای زنگ زدن تلفن را می شنود و گوشی تلفن را برمی دارد. اکنون می توانید با او مکالمه نمایید. این تشابه کاملی با ارتباط ASP با بانک اطلاعاتی دارد. شما نیازمند برقراری یک ارتباط می باشد.

آبجکت CONNECTION

آبجکت connection برای نگهداری اطلاعاتی در مورد بانک اطلاعاتی که می خواهید به آن دسترسی پیدا نمایی، استفاده می شود. این آبجکت به همان شیوه ای که کامپوننت هایتان را می ساختید با استفاده از SERVER.CREATEOBJECT ایجاد می گردد:

```
Dim objConn
```

```
Set objConn = Server.CreateObject
```

```
("ADODB.Connection")
```

مواقع زیادی پیش می آید که مایلید بخ چندین بانک اطلاعاتی به صورت همزمان ارتباط برقرار نمایید. ممکن است از چندین بانک

اطلاعاتی Access یا مخلطی از بانک های اطلاعاتی اکسس،

اوراکل و SQL استفاده نمایید. در این موارد باید از چندین نمونه

آبجکت connection استفاده نمایید. استفاده از چندین منبع

اطلاعاتی کمی باعث پیچیدگی می گردد، زیرا منابع گوناگون دارای

تواناییهای گوناگون می باشند. خوشبختانه، نیاز به داشتن مکتب

چند گانه اطلاعات در یک صفحه ASP به ندرت اتفاق می افتد.

استفاده از سیستم DSN

یک سیستم DSN، فایلی است که اطلاعاتی در باره اینکه بانک

اطلاعاتی در کجا قرار دارد و چه نوع از بانک اطلاعاتی می باشد

را نگهداری می نماید. DSN نام منبع اطلاعاتی را نگهداری می

نماید. ایجاد یک سیستم DSN به راحتی صورت می گیرد:

۱- Access را ببندید.

۲- در ویندوز ۲۰۰۰، بر روی start کلیک نمایید، و سپس مراحل زیر را انجام دهید.

Start>setting>control panel>administrative Tools>
Data sources (ODBS)

یا

Start>Administrative Tools>Data source.

۳- سراغ system DSN Tab بروید. در این جا فهرستی از منابع

اطلاعاتی سیستم فعلی که پیکره بندی شده است وجود دارد. در

سمت راست سه دکمه وجود دارد که به شما اجازه می دهند یک

سیستم DSN جدی را اضافه نمایید، یا سیستم انتخاب شده را

حذف کنید، یا تنظیمات سیستم انتخاب شده فعلی را تغییر دهید.

۴- شما قصد اضافه کردن یک DSN جدید را دارید، پس دکمه

Add را کلیک کنید.

۵- درایو اکسس میکروسافت را انتخاب کنید (*.mdb) و روی

دکمه Finish کلیک نمایید.

۶- اکنون کادر setup ظاهر می گردد. به عنوان منبع اطلاعات widgetword.dsn را وارد کنید. این نامی است برای ارجاع به DSN در صفحات ASP خود از آن استفاده می نمایید. ممکن است توضیحاتی را نیز وارد نمایید. این توضیحات به شما کمک می کند که به یاد بیاورید که این بانک اطلاعاتی و DSN برای چه کاری می باشند.

۷- بر روی دکمه select کلیک کنید. اکنون فایل mdb را برای بانک اطلاعاتی انتخاب نمایید. ما آن را Widgetworld.mdb نامیدیم.

۸- بر روی دکمه ok کلیک کرده تا بانک اطلاعاتی انتخاب گردد. صفحه شما باید شبیه به شکل ۵-۱۶ باشد. بر روی دکمه ok دوباره کلیک کنید تا ایجاد DSN خاتمه پذیرد. باید DSNی که ایجاد کرده اید در زیر منابع اطلاعاتی سیستم مشاهده نمایید.

۹- بر روی دکمه ok کلیک نمایید تا از این مرحله خارج گردید.

اگر از نسخه های پیشین ویندوز استفاده می نمایید، می توانید به منابع اطلاعاتی ODBC از طریق زیر دسترسی پیدا نمایید:

Start>setting>control panel>ODBC Data Sources

اکنون یک سیستم DSN دارید. می توانید آن را به وسیله دستور

زیر در اختیار آبجکت connection قرار دهید:

ObjConn.ConnectionString="DSN=WidgeWorld.dsn"