

معرفی وب

هیچ چیز در دهه ۹۰ به اندازه وب جهانی (Wird Wide Web)،

ارتباطات را تغییر نداده است. وب که از ایده ای در مورد سازماندهی

اطلاعات غیر خطی ریشه گرفته، تا به امروز چنان رشد کرده است که

به عنوان سیستم ارتباطی که دهها میلیون نفر در سراسر جهان از آن

استفاده می کنند، شناخته می شود. در واقع شکی وجود ندارد که هم

اکنون وب راهی برای دریافت اطلاعات، ارتباطات و تداخل کلی روی

اینترنت است.

به همین ترتیب نرم افزاری که برای مشاهده اطلاعات وب (مرورگر یا

Browser وب) به کار می رود، به چنان اهمیتی رسیده است که

شرکت های بزرگ نرم افزاری، کلیدی ترین بخش تولیدات خود را به

آن اختصاص داده اند.

همه این مسائل چگونه رخ داد؟ وب از کجا آمد و چرا به نظر می رسد

که قسمت مهمی از تکنولوژی را تسخیر خواهد؟ آیا وب خصوصی

دارد که به آن می کند جلوتر از زمان حرکت کند؟ آیا وب روزی به

کلی نابود خواهد شد؟ آیا سرنوشت آن، این است که آیندگان، آنرا در

موزه های خود نگهداری نمایند؟

شاید صد سال دیگر، جواب تمام این سوالات روشن شده باشد!

تاریخچه وب

برخی می گویند که وب از یک رؤیا شروع شد. بعضی دیگر معتقدند

که ساختمان و بافت اصلی آن چیز جدیدی نیست و انسانها قرن

هاست که به شیوه کنونی وب با یکدیگر ارتباط برقرار کرده اند.

پایه های تکنیکی سیستمی که امروز به نام وب می شناسیم، در

سوئیس بنا نهاده شد. در ماه مارس سال ۱۹۸۹، تیم برنزی، محقق

مرکز تحقیقات هسته ای اروپا (CERN)، آزمایشگاه اروپایی ذرات

فیزیکی ژنوا، سوئیس، یک سیستم ابرمتن ارائه داد، که قادر بود

اطلاعات را به طور مناسبی در اختیار دانشمندانی که در حال مطالعه

روی فیزیک پرانرژی بودند، قرار دهد. برنزی قبلا در زمینه آرایش و

تنظیم متن و ارتباطات فعالیت کرده بود. او قبلا در سال ۱۹۸۰

سیستم ابرمتنی بنام Equire ابداع کرده بود. اگرچه در آن زمان، از

اصطلاح ابرمتن که تدلسون برای اینگونه متنها وضع کرده بود اطلاعی نداشت). مدل پیشنهادی برنزی در سال ۱۹۸۹ «ابرمتن و CERN» نان داشت. در اواخر سال ۱۹۹۰، مدل اولیه وب روی یک کامپیوتر NEXT به اجرا درآمد و رابطی که روی این سیستم برای کاربران استفاده می شد، WWW نام گرفت. از این پس بود که تمام اجزاء وب شکل گرفتند.

در مارس ۱۹۹۱ رابط WWW در یک شبکه کوچک مورد استفاده قرار گرفت و از ماه می همان سال، روی ایستگاههای مرکزی CERN قرار داده شد. تا پایان آن سال، گروه CERN همه جا به صحبت پیرامون سیستم خود پرداختند. و در ۱۹ اوت سال ۱۹۹۱ اعلام کردند که فایل‌های موجود در گروه خبری alt.hypertext(USENET) در دسترس همه جا قرار دارد.

پایه وب HTML، ابر متن (HyperText) می باشد. تدلسون اولین کسی بود که واژه ابر متن را اختراع کرد. وی آنرا برای توصیف متونی به کار برد که در آنها لازم نبود همانند ساختار سنتی نوشتار،

همه کلمات به دنبال هم و در یک صفحه آورده شوند. او ابرمتن را برای ایجاد سیستمی به نام Xanadu ابداع نمود، که بنا بود در آن بین نوشته های ادبی و حقوق مؤلفین آنها ارتباطی برقرار شود که بوسیله آن به طور اتوماتیک، حق مؤلف از خواننده آنها دریافت شود.

ابعاد تکنیکی وب

تا وقتی که ابعاد تکنیکی و عملی وب روشن نشود، جایگاه آن به عنوان یکی از مهمترین وسائل ارتباطی و اطلاع رسانی درک نخواهد شد. رشد سریع و توسعه تکنیکی وب، آنقدر آنرا معروف کرد که باعث اشتباهاتی در درک مردم از آن شد. خیلی ها فکر می کنند که وب همان اینترنت است. اما باید گفت که وب سیستم کاملاً متفاوتی از اینترنت می باشد. چرا که اولاً وب یک شبکه یا مجموعه ای از شبکه های به هم پیوسته نیست و ثانیاً نرم افزاری که وب بر روی آن سوار می شود، می تواند به گونه ای طراحی شود که بر روی انواع مختلف شبکه ها و یا حتی روی یک کامپیوتر جدا از شبکه استفاده گردد.

در اینجا تعریف ساده ای از وب ارائه می شود:

وب جهانی (World Wide Web یا WWW) یک سیستم اطلاعاتی و ارتباطی بر پایه ابرمتن‌ها، بر روی شبکه کامپیوتری اینترنت می‌باشد که بر اساس مدل سرویسگر/سرویسگر عمل می‌کند. سرویسگر (مرور یا Browser) وب می‌تواند به اطلاعات چند پروتکلی و اطلاعات ابررسانه‌ای (گاه به کمک نرم افزار هیپا یاریگر)، با استفاده از یک سیستم آدرس دهی مخصوصی دسترسی پیدا کند.

چون مطالب وب به وسیله ابرمتن‌ها به هم متصل می‌شوند، اطلاعات آن لازم نیست که بصورت خطی مرتب شده باشد. در اصطلاح ریاضی، وب یک گراف جهت دار است که راسهای آن (گره‌ها) را صفحات وب و یالهای آن را پیوندهای مربوط به ابرمتن‌ها تشکیل می‌دهد.

در واقع صفحات وب حاوی لنگرهای بصورت نقاطی ویژه می‌باشند که کاربر می‌تواند با انتخاب آنها به صفحات جدیدی هدایت شود. شکل ۱-۱ اساس سازماندهی ابرمتن‌ها را مشخص می‌نماید.

پیوندهای روی صفحات وب، که در شکل با پیکانهای جهت دار

مشخص شده اند یک لنگر را در یک صفحه حاوی ابرمتنها، به یک صفحه دیگر از متنها، و یا مکان دیگری در همان صفحه اولیه متصل می نمایند.

همچنین این پیوندها می توانند یک صفحه از ابرمتنها را به سرویسگرهای دیگری روی اینترنت مانند گوفر، FTP و تل نت وصل نمایند. بعلاوه این پیوندها می توانند صفحات مزبور را به مطالب چند رسانه ای یعنی صوت، تصویر و فیلم ارتباط دهند. این لنگرها، همانطور که گفته شد، در مرورگر وب به صورت نقاط ویژه ای نمایش داده می شوند که معمولاً متنهایی با رنگ متفاوت و دارای خطی در زیرشان می باشد و کاربر می تواند با انتخاب این نقاط ویژه، که اکثراً با استفاده از کلیک موس بر روی آنها صورت می گیرد، به صفحاتی هدایت شود که این لنگرها به آنها متصلند.

همانگونه که قبلاً تدنلسون توضیح داده بود، ابر متن، متنی است که می تواند بصورت خطی نباشد. بریا درک بهتر مفهوم متون خطی و غیر خطی، یک کتاب را در نظر بگیرید. تمام مطالب کتاب، با یک نظم

مشخص و واحد و یک ترتیب خاص، بطور متوالی و به اصطلاح به صورت خطی، از ابتدا تا انتهای کتاب نوشته شده اند. ممکن است شما

هنگام مطالعه کتاب به یک کلمه یا اصطلاح برخورد کنید که مفهوم

آنها بدرستی نمی دانید. همچنین ممکن است مفهوم آن کلمه یا

اصطلاح در جای دیگری در همان کتاب و یا یک کتاب دیگر توضیح

داده باشد. تنها کمکی که نویسنده کتاب می تواند در این زمینه برای

شما انجام دهد، این است که آدرسی که آن مطالب در آنجا توضیح

داده شده است را ذکر نماید تا شما راحت تر آنها بیابید. حال این شما

هستید که باید کتاب را به منظور یافتن آن مطلب ورق بزنید، آنها

مطالعه نموده و سپس بخ جای اول برگشته و بقیه کتاب را بخوانید.

اکنون تصور کنید که پیوندهایی وجود داشت که کلمه یا اصطلاح

مورد نظر را مستقیماً به توضیح آن متصل می کرد. بدین ترتیب شما

به سرعت و به راحتی می توانستید به سمت توضیح آن مطلب هدایت

شوید. این دقیقاً همان چیزی است که در متنهای غیر خطی دیده می

شود. یعنی مطالب اینگونه متنها پشت سر هم و بر روی یک خط قرار نگرفته اند. بلکه بصورت شاخه به شاخه و منشعب آرایش شده اند.

همانطور که توضیح داده شد، صفحات وب حاوی لنگرهای می باشد که کاربر، با انتخاب آنها می تواند به صفحات جدید هدایت شود. به این ترتیب، مطالب وب به صورت غیرخطی سازمان یافته اند.

خصوصیت مهم دیگری که ابر متنهای وب دارند، محدود نبودن

آنهاست. بدین معنی که چون صفحات وب می توانند به صفحات

دیگری که بوسیله نمایندگان دیگر تهیه شده اند و بر روی

سرویسگرهای دیگر قرار دارند، متصل شوند، لذا به سرویسگر و یا

ایستگاه وب منحصری محدود نمی شوند. در واقع، معمولاً صفحات

وب، به صفحات زیاد دیری بر روی اینترنت متصلند و نویسندگان، با

این کار، کیفیت عرضه مطالب خود را بالا می برند. دوباره تذکر مکی

دهیم که گرچه اکثراً وب در ارتباز با اینترنت شناخته می شود، اما

نرم افزارهای وب را می توان روی شبکه های محلی یا اینترنت و یا

حیت روی یک کامپیوتر منفرد قرار داد. بطور مشابهی، هر کاری که

به عنوان وب تھی شده باشد، می تواند مقصدی برای یک لنگر از یک صفحه دیگر باشد.

صفحات ابر متن وب، با استفاده از زبان علامتگذاری ابرمتن (HTML) که گسترشی از SGML می باشد، طراحی می شوند. SGML یک استاندارد بین المللی (ISO 8879) می باشد که برپا پردازش اطلاعات متنی بکار می رود.

فلسفه وجودی SGML این است که به گونه ای اطلاعات را آرایش کند که سیستم های دیگر چاپ برنامه های دیگر، به راحتی بتوانند از آن استفاده نمایند.

HTML که بوسیله SGML تعریف می شود، یک زبان علامت گذاری لغات و متون که روش به نمایش درآمدن آنها را تعیین می کند نه شکل دقیق ظاهر شدن آنها را.

WWW به عنوان یک سیستم اطلاعاتی و ارتباطی بوسیله وب می توان هم اطلاعات را منتشر کرد و هم اطلاعات را دریافت نمود. مورد دوم (دریافت اطلاعات از کاربران) توسط

پرسشنامه هایی انجام می شود که بوسیله HTML ساخته می شوند.

لذا وب فقط سیستمی برای پخش یکطرفه اطلاعات (مانند رادیو و تلوزیون) نمی باشد، بلکه توانائی ایجاد ارتباط متقابل با کاربران را نیز دارا است.

با استفاده از برنامه CGI در کنار پرسشنامه ها، می توان به کاربران اجازه داد تا ساختار ابرمتنها را دستکاری کرده یا تغییر دهند.

چون وب توانایی پخش اطلاعات را دارا می باشد، مشابه هر سیستم پخش کننده اطلاعات، می تواند هر تعداد از مخاطبان را داشته باشد؛ از یک فرد واحد گرفته (هنگامیکه بر روی یک کامپیوتر منفرد سوار شده باشد)، تا یک گروه (وقتی که روی یک شبکه محلی قرار گرفته باشد) تا کل مخاطبان وب جهانی (وقتی روی اینترنت باشد).

WWW و استفاده از آن از طریق شبکه اینترنت گرچه نرم افزارهای وب نیازی به قرارگیری روی اینترنت ندارند و می توانند روی یک رایانه منفرد و یا یک اینترنت محلی هم قرار

بگیرند، معمول ترین شکل استفاده از طریف شبکه اینترنت است. اینترنت یک شبکه منفرد نیست بلکه مجموعه ای از شبکه ها است به یکدیگر متصل شده اند. این اتصال بر اساس یک سری پروتکل ها (ضوابط) که به نام TCP/IP شناخته شده اند، برقرار می باشد. بر اساس این ضوابط، اطلاعات به صورت بسته های کوچکتری درآمده و سپس رد و بدل می شوند.

باید گفت که سیستم اینترنت به گونه ای طراحی شده است که اگر قسمتی از آن از کار بیفتد، اطلاعات (بسته ها) بتوانند از طریق راههای دیگر (البته در صورت وجود) منتقل شوند. به این ترتیب هیچ ضرورتی وجود ندارند که بسته های اطلاعاتی همگی از یک مسیر انتقال یابند.

به شکل ۱-۳ توجه کنید. همانگونه که در شکل مشخص شده، یک مجموعه اطلاعات می تواند به بسته های کوچکتر و مجزا از هم تقسیم شده و از طریق اینترنت ارسال شود. این بسته ها ممکن است از طریق راههای متفاوتی به سمت مقصد هدایت شوند. هنگامیکه به

مقصد رسیدند، دوباره بر اساس اطلاعاتی که در آنها کدگذاری شده است، در کنار هم و ر محل مناسب قرار می گیرند و به این ترتیب اطلاعات بازسازی می شود.

اگر یک بسته اطلاعاتی در طول راه مفقود شده و یا خراب گرد، درخواستی برای ارسال مجدد آن صادر شده و آن بسته مجدداً فرستاده خواهد شد.

گرچه پروتکل TCP/IP برای مدتهای زیادی اینترنت را تغذیه کرده است، اما به تازگی بر روی یک پروتکل جدید بنام IPNG کار می شود.

نام رسمی IPNG، IPV6 است در حالیکه نسخه کنونی (1996) پروتکل اینترنت، IPV4 می باشد. انتظار می رود که IPNG، با حل مشکلاتی که در رابطه با IPV4 وجود دارد، بتواند تا مدتهای مدیری اینترنت را پشتیبانی نماید. یکی از مشکلاتی که با IPV4 وجود دارد این است که فضای برای نامگذاری کامپیوترها روی اینترنت در نظر

گرفته شده به سرعت در حال پر شدن است. این در حالیست که IPNG، اجازه آدرس دهی بسیار بیشتری را خواهد داد.

وب بر اساس مدل سرویسگر / سرویسگیر عمل می کند.

مدل سرویسگر / سرویسگر در سیستم شبکه های کامپیوتری شامل سه جزء است: سرویسگر، سرویسگیر و شبکه. سرویسگیر در واقع یک نرم افزار که بر روی کامپیوتر کاربر اجرا می شود و سرویسگر

نرم افزاری می باشد که بر روی کامپیوتری که عمل تغذیه اطلاعاتی را به عهده دارد اجرا می شود.

نرم افزار سرویسگیر که می تواند متناسب با سیستم سخت افزاری کاربرتهیه شود، در واقع به عنوان یک رابطه بین آن سیستم و

سیستم دیگر عمل می کند. کاربر از طریق این نرم افزار می تواند درخواستی برای دریافت اطلاعات و یا انجام اعمالی ارسال دارد.

این درخواست از طریق شبکه به کامپیوتر سرویسگر می رسد و آن هم به نوبه خود این درخواست را تفسیر کرده و اعمال لازم را انجام

می دهد. این اعمال می تواند شامل جستجو در ایم بانک اطلاعاتی، و یا

ایجاد تغییراتی در اطلاعاتی در اطلاعات ذخیره شده باشد. نتیجه این اعمال (در صورت وجود)، به سوی سرویسگر فرستاده می شود تا برای کاربر نمایش داده شود.

تمام ارتباطاتی که بر اساس مدل سرویسگیر / سرویسگر کار می کنند، از یکسری ضوابط با پروتکل هائی پیروی می نمایند کهب رای آن سیستم تعریف شده اند.

شکل ۴-۱ این روابط را بطور خلاصه بیان می کند. همانگونه که در شکل مشاهده می شود، یک درخواست از سوی سرویسگیر برای سرویسگر فرستاده شده است و در مقابل، سرویسگر، اطلاعات لازم را برای سرویسگیر ارسال داشته است.

این شکل فعالیت فعالیت «درخواست و پاسخ» که از طریق مدل سرویسگر / سرویسگیر اجرا می شود، توانائیهای زیادی دارد. چون رابطه سرویسگیر و سرویسگر بر اساس یک پروتکل از پیش تعریف شده و مشخص برقرار می باشد، نرم افزار سرویسگیر می تواند برای هر نوع سخت افزار خاصی طراحی شود. در واقع سرویسگیر

دیگر نگران این که کاربر از چه نوع کامپیوتری استفاده می کند، نخواهد بود زیرا می داند که زبان مشترکی با سرویسگیر دارد که صرف نشر از سیستم سخت افزاری کاربر، هر دو آنها به این زبان با یکدیگر ارتباط برقرار می کنند. برای مثال یک سرویسگیر وب (مرورگر وب) می تواند برای سیستم مکینتاش طراحی شود و با استفاده از آن به سرویسگرها وب دسترسی پیدا کند. همچنین مرورگری که برای سیستم یونیکس و برای اجرا روی Xwindows طراحی شده نیز می تواند به همه سرویسگرهای وب دسترسی پیدا کند. به این ترتیب مرزهای وظایف سرویسگر و سرویسگر کاملاً روشن است و لذا اطلاعات را می توان به سادگی در اختیار هر کاربر و با هر سیستم سخت افزاری قرار داد و این اطلاعات لازم نیست برای هر کاربر و متناسب با سیستم سخت افزاری وی، به طور جداگانه طراحی شوند.

در واقع این طراحی در سطح نرم افزار سرویسگیر صورت گرفته است و همین امر را به صورت یک پدیده غیر وابسته به سیستم در آورده است.

برای درک بهتر این موضوع می توان مدل سرویسگیر / سرویسگر را مشابه سیستم پخش تلوزیونی فرض کرد. بیننده برنامه های تلوزیون می تواند هر نوع دستگاه تلوزیونی که می خواهد (سرویسگیر) تهیه نماید و بوسیله آن برنامه مورد علاقه اس را که از برجهای فرستنده (سرویسگر) پخش می شود، مشاهده نماید. اینکه بیننده از چه نوع دستگاهی استفاده می کند، اهمیتی ندارد. در هر حال، دستگاه وی، اطلاعات را از ایستگاه پخش امواج، بر اساس یک نظم و ساختار استاندارد، دریافت کرده و نمایش می دهد. به این ترتیب، برجهای فرستنده لازم نیست برای هر نوع تلوزیونی، مانند سیاه و سفید و رنگی، و یا تلوزیونهای با اندازه های متفاوت، برنامه های متفاوتی پخش نماید. شکل ۵-۱ خلاصه ای از روابط متقابل

سرویسگر و سرویسگیر را نشان می دهد.

مرورگرهای وب می توانند به اطلاعات چند پروتکلی دسترسی داشته باشند.

مرورگرهای وب، چند پروتکلی (چند ضابطه ای) هستند. این، بدان معناست که مرورگرها می توانند به انواع مختلفی سرویسگرها که بر اساس پروتکل های مختلفی ارتباط برقرار می کنند دسترسی داشته باشند. مهمترین پروتکل هائی با استفاده از آنها، به سرویسگرها متصل شوند، عبارتند از:

HTTP یا پروتکل انتقال ابر متن: این پروتکل مخصوص وب است و برای انتقال ابرمتن از طریق شبکه طراحی شده.

FTP یا پروتکل انتقال فایل: این پروتکل برای این طراحی شده است که به کاربر اجازه دهد تا فایل های متنی یا باینتری را برداشت نماید.

Gopher (گوفر): این پروتکل برای در اختیار قرار دادن اطلاعات، با استفاده از سیستمی از منوها، صفحات و یا اتصالاتی به تل نت طراحی شده است.

News یا NNTP (پروتکل انتقال اخبار شبکه): این پروتکل مربوط به اخبارت یوزنت است. یوزنت سیستمی برای بحث و تبادل اطلاعات پیرامون موضوعاتی تقسیم بندی شده است.

Telnet (تل نت): این پروتکل برای ورود به سیستم یک کامپیوتر میزبان (معمولاً از راه دور) استفاده یم شود. به این ترتیب، مرورگر وب، مثلاً هنگامیکه به یک سرویسگر گوفر متصل است، بعنوان یک سرویسگر گوفر و وقتی که به یک سرویسگر اخبار یوزنت متصل است، مانند یک سرویسگر اخبار، عمل می کند.

آدرسی که به یک صفحه وب و یا هر منبعی بر روی وب جهانی و بطور کلی اینترنت اشاره می نماید URL نامیده یم شود. URL ساختاری دارد که بیان می کند چگونه می توان به یک منبع خاص، دسترسی پیدا کرد. در واقع این ساختار حاوی اطلاعاتی در باره نام کامپیوتر میزبان و راه دسترسی به آن می باشد. برای مثال، در زیر سه نمونه از URL های مختلف آورده شده است.

<http://www/w3.org/hypertext/www/TheProjectr.html>

این URL به یک سرویسگر وب اشاره دارد زیرا حروف http که در ابتدای آن آمده است، مشخص کننده استفاده از پروتکل انتقال ابرمتن می باشد.

برروی این سرویسگر وب که www.w3.org نام دارد، فایلی به نام `hypertext/www/TheProject.html`، در یک دایرکتوری با نام `hypertext/www` قرار گرفته است که مورد اشاره URL مذکور می باشد. اگر به پسوند این فایل (یعنی html) توجه کنیم، در می یابیم که این فایل، یک فایل HTML می باشد.

<ftp://ftp.w3.org/pub/>

این URL، به دایرکتوری `pub`، برروی یک میزبان با نام [ftp.w3.org](ftp://ftp.w3.org) اشاره می کند که با استفاده از پروتکل انتقال فایل ftp می توان به آن دسترسی یافت.

<News:Comp.infosystems.www.misc>

این URL به یک گروه خبری یوزنت اشاره می کند. هنگامیکه کاربر این URL را انتخاب می کند، مرورگر وب، دریزعنوانهای موجود در این URL را استخراج می کند.

در واقع این گروه خبری، گروهی است که مخصوص بحث در مورد مطالب متنوع (mosc) در مورد سیستم های اطلاعاتی (infosystems) کامپیوترها (comp) بر روی وب جهانی (www) می باشد. بر خلاف دو URL فبلیو این URL به یک میزبان مشخص اشاره نمی کند بلکه اشاره آن به یک میزبان سرویسگر اخبار یوزنت می باشد که کاربر هنگام نصب مرورگر خود، آنرا تعریف نموده است.

خلاصه ای از نامگذاری کامپیوترها بر روی اینترنت نامگذار کامپیوترها بر روی اینترنت، بر اساس سیستمی از اعداد سلسله وار انجام می شود. به تمام کامپیوترهای روی اینترنت یک آدرس عددی خاص (IP) تعلق می گیرد.

با استفاده از سیستم نامگذاری حوزه ای اینترنت یا DNS، می توان بین این آدرسهای عددی IP (مثلاً 128.113.1.5) و نام کامپیوترها (مثلاً ftp.rpi.edu) تناظر برقرار کرد و بدین جهت به جای آن اعداد گیج کننده، از اسمهای حرفی که کارکردن با آنها برای دانشسان بسیار ساده تر است، استفاده نمود. نام کامپیوترها از چند قطعه تشکیل شده است که هر قطعه بوسیله یک نقطه، از دیگری جدا شده است. قطعه سمت راست نام بالاترین رده حوزه را مشخص می کند. این قسمت معمولاً مشخص کننده نوع سازمان یا موسسه ای است که کامپیوتر مزبور متعلق به آن می باشد. جدول ۱-۱ تعدادی از این نامها و نوع سازمان مربوطه را مشخص می کند.

علاوه بر این قسمت، هر شرکت یا موسسه برای خود نامی انتخاب می کند که در سمت چپ قسمت مزبور قرار می گیرد. (مانند نام microsoft در microsoft.com). در ضمن هر شرکت می تواند نامهای دیگری، که معمولاً بر اساس تقسیمات سازمانی درون شرکت انتخاب می شوند، به نام خود اضافه کند. برای مثال

در صفحات قبل توضیح داده شد که ابر متنهای وب شامل پیوندهائی چند پروتکلی هستند. از طرف دیگر، صفحات و مطالب ارتباطاتی شبکه ای دارند یعنی به یک سرویسگر خاص محدود نیم شوند و می توانند پیوندهایی با سرویسگرهای دیگر نیز برقرار کنند.

در نتیجه می توان گفت که وب بطور کلی شامل ابر رسانه هائی است. متن، گرافیک، فیلم، صدا) که پیوندهائی چند پروتکلی و ارتباطی شبکه ای دارند.

گاه برای دسترسی به ابر رسانه ها باید از برنامه های یاریگر استفاده شود. برنامه های یاریگر نرم افزارهائی هستند که مرورگر وب به کمک آنها می توانند اطلاعات چند رسانه ای را برای کاربر نمایش دهد. برای مثال نمایش دادن فیلم، مرورگر وب باید نرم افزار نمایش فیلم را در دسترس داشته باشد. و یا برای نمایش دادن تصاویر گرافیکی، مرورگر وب باید مرورگری گرافیکی باشد. که بر روی سیستمی مانند Xwindows، سیستم عامل Macintosh و یا ویندوز میکروسافت نصب شده است). لازم به تذکر است که بعضی

مرورگرها خصوصاً مرورگرهای قدیمی، تنها قادر به نمایش متن هستند اما بیشتر مرورگرهای جدید گرافیکی بوده و بطور گسترده ای بر روی سیستم های مختلف قابل دسترسی می باشند. این مرورگرها، قادر به نمایش تصاویر گرافیکی در پنجره مربوط به خود هستند.

همچنین مفروضات بعضی از زبانهای خاص در دستة برنامه های یاریگر قرار دارند. بریا مثال زبان برنامه نیسی جاوا که بریا ایجاد مطالب متقابل الاقر بکار می رود، می تواند بوسیله مرورگرهایی که قادر به تفسیر برنامه های جاوا می باشند، دریافت و اجرا شود. برای اینکه مرورگر بتواند برنامه های جاوا را تفسیر نماید، در واقع باید یک نسخه از آن که دارای مفسر جاوا می باشد، نوشته شود.

بر اساس مطالب فوق، نکته اساسی در تعریف وب این است که بدانیم:

وب = متنهای ویژه + چند رسانه ای + شبکه:

- متنهای ویژه، پایه پیوندهای مربوطه می باشند

• چند رسانه ای، اطلاعات را به اشکال مختلف و با بکارگیری حواس گوناگون (سمعی و بصری) ارائه می کند.

• شبکه، برای دسترسی به منابع گوناگون ضروری است.

بررسی وب به عنوان یک رسانه جدید

شاید یکی از مهمترین دلایلی که وب را تبدیل به یک رسانه جذاب و پرترفدار کرده است، توانائی آن در ایجاد رابطه متقابل با کاربران

می باشد.

تا این اواخر، تمامی رسانه ها یک رابطه یک سویه با مخاطبان خود برقرار می کردند. تلوزیون، رادیو، روزنامه ها و مجلات، همگی، اطلاعات را به یک شیوه خطی و از پیش تعیین شده، که قبلاً تهیه

کنندگان به طرق مختلف مثل صوت، متن و تصویر تنظیم کرده اند، به افراد دیگر انتقال می دهند.

شما کنترلی روی برنامه هائی که از طریق تلوزیون پخش می شود، ندارید. البته می توانید صدای آن را کم و زیاد کنید، کانال آنرا عوض

کنید و یا در صورتیکه ترجیح می دهید، تلوزیون خود را خاموش
نمائید.

همچنین شما می توانید با روزنامه ها یا مجلات مشترک شوید یا
نشوید. و یا می توانید فقط مقاله مورد علاقه خود را مطالعه کنید.
حتی می توانید برای نویسنده منامه بنویسید و نظرات خود را به او
منتقل نمائید. اما تمام اینها در مقایسه با دنیای مهیج و متقابل الاثر
وب جهانی خیلی محدودند.

رسانه جدید

علاقه کاربران برای رهائی از رسانه های سنتی و قدیمی و رسیدن به
نوعی دیگر از رسانه، که بیشتر بتوانند با آن رابطه برقرار کنند، باعث
شد که فکر ایجاد چیزی که گاه به آن «رسانه جدید» می گویند،
بوجود بیاید. توانائی ایجاد رابطه متقابل با مخاطبان، اساسی ترین
جنبه این رسانه به حساب می آید. به این ترتیب، مخاطبان اختیارات و
قدرت انتخابی، بیشتر از آنچه تلوزیون، رادیو و روزنامه به آنها می
دهد، خواهند داشت. همچنین رسانه جدید نیازی ندارد که اطلاعات و

مطالب خود را به صورت خطی سازماندهی کند. بنابراین بجای ارائه مطالبی که باید خط به خط و از ابتدا تا انتها خواننده شوند، با ارائه یک سازماندهی غیر خطی اطلاعات، به مخاطب خود اجازه می دهد تا از مطلبی به سراغ مطلب دیگر برود و به این ترتیب بتواند از زوایای مختلف به اطلاعات دسترسی پیدا کند. چیزی که این شکل از سازماندهی غیر خطی اطلاعات را جذاب تر می کند، این است که این ساختار، شباهت بسیار نزدیکی به طرز تفکر بشر دارد.

برای درک بهتر این موضوع فرض کنید که یکی از دوستانتان مشغول تعریف کردن یک فیلم سینمایی برای شما است. ممکن است هنگامیکه او از قهرمان فیلم صحبت می کند، ناگهان این سؤال برای شما ایجاد شود که کدام هنرپیشه نقش او را بازی کرده است. در چنین موقعیتی شما معمولاً این سؤال را در ذهن خود نگاه نمی دارید و از دوست خود در باره آن پرسش خواهید کرد.

همچنین وقتی جکه دوست شما، به این سوال پاسخ داد، ممکن است این سوال برایتان ایجاد شود که این هنرپیشه در چه فیلمهای دیگری بازی کرده است.

ممکن است این «از این شاخه به آن شاخه پریدن» آنقدر ادامه یابد که کاملاً شما را از بحث اصلی دور کند. در واقع «از موضوعی به موضوع دیگر پریدن» یک روند کاملاً طبیعی در تفکر آدمی است و وب، به عنوان یک رسانه جدید، این روند را دارا بوده و به آن شکل می دهد.

درک این نکته که رسانه جدید، انتخابی و آزاد عمل می کند و بجای ساختاری بر اساس خطوط مستقیم، ساختاری غیرخطی را دنبال می کند، برای طراحی صفحات وب، کمک زیادی به شما می کند.

باید توجه داشت که این ارتباطات متقابل، تنها با استفاده از یک نوع رسانه انجام نمی شود. این خصومصیت، استفاده از چند رسانه ای را می توان در حد کوچکتري در روزنامه که از متن و تصویر، همزمان استفاده می کند و در حد وسیعتری در تلویزیون، که از

متن، تصویر، صوت، تصاویر متحرک و فیلم، در کنار یکدیگر بهره می گیرد، مشاهده کرد.

تمام این موارد را که البته با قدرت و سرعت رایاتنه ها نیز ترکیب می شوند، می توان برای خلق صفحات وب بکار برد. به این ترتیب می توان محیطهایی را خلق کرد که تمامی آنها جزئی از یک محیط بزرگتر هستند. این محیطهای کوچکتر، صفحات خانگی (Home Page) و

ایستگاههای وب (web sites)، و محیط بزرگتر، وب جهانی (World Wide Web یا WWW) نام دارد. نه تنها هر یک از این محیطهای کوچک روابط متقابل و انتخابهای مختلفی در درون خود دارند، بلکه خود آنها، با استفاده از پیوندهای ویژه (Hyperlink) که در واقع خطوط لوله کشی بین ایستگاههای وب گوناگون به شمار می روند، با یکدیگر مرتبطند و بدین شکل وب جهانی را می سازند.

به منظور طراحی یک وب قوی، باید به تمام اجزاء آن از متنها و تصاویر ساده گرفته تا برنامه های پیشرفته مانند جاوا و VRML

(واقعیت مجازی) توجه کرد. یک وب خوب، شامل متنهائی مناسب،
تصاویری با کیفیت قابل قبول و ارتباطاتی جذاب می باشد.

وب از دیدگاه کاربران

اگر شما یک کاربر وب هستید، احتمالاً نیازی به دانستن مطالبی از قبیل روش کد شدن اطلاعات در بسته های ارسالی اینترنت ندارید. در عوض باید هنر استفاده از وب را کسب نمائید و بتوانید از منابعی که شما در استفاده از وب راهنمائی می کنند، بهره بگیرید. در مرحله بعد باید روشن ساختن و توسعه صفحات وب و قرار دادن اطلاعات، در آنها فرا بگیرد.

زمینه های ارتباطی وب

ارتباط از طریق وب می توان به اشکال مختلف و طرق مختلفی انجام شود. در واقع انواع مختلفی از روشهای ارتباطی بر روی وب توسعه یافته و استفاده می شود. این روشها به طور عمده عبارتند از:

- ترتیبات بین فردی: وب به کاربران اجازه ایجاد یک صفحه خانگی مخصوص به خود را می دهد تا اطلاعات فردی یا شغلی خود را از طریق آن عرضه نمایند.

ایده صفحه خانگی از آنجا منشاء گرفت که ایستگاههای وب می خواستند فایلی داشته باشند که اگر کاربر بدون ذکر نام فایل خاصی تنها به آدرس یک کامپیوتر میزبان و یا یک دایرکتوری خاص در داخل آن کامپیوتر اشاره کرد، (به عنوان صفحه خانگی آن دایرکتوری) برای کاربر نمایش داده شود.

به همین دلیل صفحه خانگی که ایستگاههای وب، سازمانها و یا افراد برای خود طراحی می کنند، اولیت صفح مرتبط به آنها محسوب می شود و کاربرانی که می خواهند صفحات وب آنها را مشاهده نمایند، معمولاً ابتدا وارد صفحه شده و سپس از آنجا و از طریق پیوندهائی که به صفحات دیگر مربوط است، به سایر صفحات راهنمائی می شتند.

البته باید گفت که ایستگاههای وب معمولاً صفحات خانگی کاربران را در یک دایرکتوری مخصوص قرار می دهند و صفحه ای که بطور اولیه پس از ذکر نام آن دایرکتوری ظاهر می شود و در واقع صفحه خانگی آن دایرکتوری محسوب می شود، حاوی لیستی از نام کاربران دارای صفحه خانگی است که نام آنها پویندی برای صفحه مربوط به خودشان می باشد.

معمولاً کاربران پویندهائی در صفحات خانگی خود، برای صفحات خانگی همکاران و دوستان خود قرار داده و بدین ترتیب «قبایل الکترونیکی» ایجاد می نمایند!

ذکر این نکته ضروری است که گرچه وب مانند هر سیستم ارتباطی دیگر، قابلیت پخش اطلاعات را برای توده مردنم دارا می باشد، اما این صفحات خانگی معمولاً به عنوان وسیله ای برای ارتباطات بین فردی و یک به یک در نظر گرفته می شوند.

- ارتباطات گروهی: مردم یم توانند از طریق وب روابطی شکل دهند

که بدون توجه به مرزهای جغرافیایی. روی عناوین مورد علاقه)

مشترک متمرکز شده باشد. طبقه بندیهای موضوعی اطلاعات و عناوین معمولاً روش مناسبی برای مشخص کردن گروههای مزبور است.

- همچنین گروههای مختلفی می توانند تشکیل شوند که با ایجاد کنفرانسها و گفتگوها از طریق وب، بر روی یک موضوع اجتماعی و یا عملی به بحث و تبادل نظر می پردازند.

- ارتباطات سازمانی: بیشتر سرویسگرهای اولیه وب که در واقع وب را ظاهر کردند، متعلق به سازمانها بودند نه افراد. و طبیعتاً صفحات خانگی هم مخصوص همان سازمان طراحی می شد.

- به این ترتیب سیستم های اطلاع رسانی دانشگاهها راه انهدازی شد. از این طریق دانشگاهها. مؤسسات آموزشی اطلاعات خود، از قبیل برنامه های درسی و تحقیقاتی و برنامه اساتید و دروس آنها و شرایط ثبت نام را به اطلاع علاقه مندان می رسانند.

در کنار آنها، سازمانهای تجاری دولتی و یا غیر دولتی نیز به پیروی از این الگو، در این زمینه شروع به فعالیت نمودند.

- ارتباطات توده ای: درست به همان ترتیبی که وسایل دیگر

ارتباطی (روزنامه، رادیو، تلویزیون)، برای انتشار اطلاعات به توده

مردم بکار می روند، وب نیز در جهت ارتباطات توده ای استفاده

می شود. بسیاری از مجلات تجاریر یا غیر تجاری، و همچنین

نشریات زیادی دیگری از طریق وب ارائه می شوند. بعلاوه،

همانگونه که قبلاً نیز تذکر داده شد، تمام صفحات عمومی وب می

توانند توسط هر کاربری مشاهده شوند و بدین ترتیب پتانسیل

ارتباط با توده مردم را دارا می باشند.

- نکته کلیدی در این مبحث این است که وب همانند هر سیستم

ارتباطی می توان اطلاعات را به هر تعداد از اشخاص منتقل نماید.

لذا طبقه بندی انواع ارتباطات ۰ که در بالا در قالب ۴ گروه ذکر

شد) بستگی به این دارد که چه کسی یا چه کسانی در این ارتباط

شرکت می کنند. به عبارت دیگر، گرچه صفحات وب، پتانسیل

اطلاع رسانی و ارتباط جهانی را دارا هستند، اما بسته به اینکه

چگونه و توسط چه کسانی استفاده می شوند، جزء یکی از دسته

های فوق جای می یگرند.

HTML چیست؟

شکل پیشین وب، گوفر بوده است. گوفر، روشی برای نمایش

اطلاعات بوسیله متنهای اسکی معرفی کرد. متنهای اسکی، اجازه می

دهد تا اطلاعات، با استفاده از tabها، حروف فاصله (Spaces) و

خطوط جدید مرتب شوند و این، یک روش ساده برای پخش و

نمایش گسترده اطلاعات محسوب می شود.

در واقع اگر وب نیز فقط از متنهای ساده استفاده می کرد، هرگز

چنین محبوب و معروف نمی گشت زیرا متنهای ساده از لحاظ

بصری جذاب نمی باشد. به همین دلیل بود که زبانهای علامتگذاری

روی کار آمدند.

با استفاده از زبانهای علامتگذاری، می توان به سادگی، اطلاعات را به

روشهای جذاب تری آرایش کرد. با کمک اینها، قادر خواهیم بود تا

متنهای ساده را با ساختار پیچیده تری مانند جداول و لیستها در هم

بیامیزیم. زبانهای علامتگذاری به ما اجازه می دهد تا ساختار صفحات را به کمک عبارات و دستوراتی ساده تعریف، و یا اصطلاحاً

«علامتگذاری» کنیم. (بدون اینکه نیاز باشد تا آنها را کمپایل کرده و یا

تبدیلی به زبان قابل فهم برای کامپیوتر بنمائیم)

این قابلیت، از دو جهت قابل توجه است:

ائلاً، چون این دستورات بهد زبان هیچ کامپیوتری ترجمه نشده اند، از

اسنادی که بویسأ آنها ساخته شده است، می توان به طور وسیعی،

برروی هر نوع کامپیوتری استفاده کرد. تنها باید برای هر نوع

کامپیوتر یا سیستم عامل خاصی، یک مفسر نوشته شود تا بتواند

اسناد مزبور را ترجمه یا اصطلاً تفسیر نماید.

ثانِباً، چون عبارت و دستورات آنها را از کاراکترهای استاندارد

تشکیل شده است، برای انسان قابل فهم می باشد و لذا به راحتی می

توان صفحات دلخواه را به کمک این زبانها خلق نمود. در واقع، برای

ایجاد یک سند HTML تنها بهد یک ویرایشگر متن، مانند Notepad

ویندوز نیاز دارید. هنگامیکه یک سند HTML خلق شد، کافی است

با یک مرورگر وب، آن فایل را باز (Open) کنید تا بتوانید صفحه وب مربوطه را مشاهده نمایید.

HTML معروفترین و شایعترین زبان علامتگذاری است. معروفیت آن بیشتر به دلیل سهولت استفاده از آن می باشد. به کمک HTML، می توان به سرعت و به سادگی صفحات وب دلخواه را خلق کرده و به مخاطبان عرضه نمود.

هنگامی که تیم برنزی مشغول طراحی وب بود، راهی را برای طراحی صفحات آن در نظر گرفته بود تا همگان به راحتی بتوانند به نشر وب پردازند. به همین منظور، او و همکارانش در آزمایشگاه اروپائی ذرات فیزیکی (CERN) HTML طرح ریزی کردند. چون SGML ثابت بود که می تواند یک راه حل مناسب و عمومی بر روی سیستم های مختلف کامپیوتری باشد، یک پایه مناسب برای HTML به حساب می آمد به این ترتیب زبانی ساده برای ایجاد صفحات وب بوجود آمد که در عین سادگی، می توانست در آینده، به میزان لازم

رشد پیدا کند.

HTML از هنگام تولد تا کنون راه زیادی را پیموده است. در حال حاضر، حداقل چهار نسخه از آن شناخته شده است: HTML 1.0،

HTML 2.0، HTML 3.2، HTML 4. هر نسخه جدید، علاوه بر

اینکه بطور کامل با نسخه های قبلی سازگار است، اجزاء زیاد دیگری را نیز درون خود جای داده است.

همچنین، باید گفت که توسعه دهندگان اینترنت، اجزاء دیگری را نمیز

به این نسخه های اصلی HTML اضافه کرده اند. گرچه این اجزاء

استاندارد نمی باشد، اما بسیاری از ناشرین وب، آنها را پذیرفته اند و

از آنها استفاده می نمایند. بعضی دیگر از این اجزاء، خصوصاً آنهایی

که بوسیله مروگر Netscape و Internet Explorer تعریف شده

اند، آنچنان معروفیتی یافته اند که گوئی جزئی از HTML استاندارد

می باشد.

افزودن مطالب چند رسانه ای (مولتی مدیا) به صفحات وب

پدیده چند رسانه ای (مولتی مدیا) به سرعت محبوبیت وب جهانی را

افزایش می دهد. شما با استفاده از پدیده چند رسانه ای می توانید

صفحاتی صفحاتی هیجان انگیز و زیبا خلق کنید که بینندگان را به خود جذب کرده و آنها را ترغیب می نماید تا صفحات شما را پشت سر هم مشاهده کنند.

تصاویر، مهمترین جزء این پدیده به شمار می روند. در واقع استفاده از تصاویر، وبرا بسیار قدرتمندتر می کند. در صورتی که تا کنون صفحات وب را دیده باشید، حتماً می دانید که در اکثر صفحات وب، تصاویری وجود دارد. طراحان وب، به بهره گیری از این تصاویر، صفحات خود را زیباتر کرده و با توجه بینندگان را به این صفحات جلب می نمایند.

علاوه بر تصاویر، مولتی مدیا شامل صوت و ویدئو نیز می باشد. بسیاری از طراحان وب، از صدا و ویدئو نیز در صفحات خود استفاده می کنند زیرا دریافته اند که بینندگان به این ترتیب، بیشتر مجذوب این صفحات می شوند. در واقع وقتی که در صفحات وب، از صدا و ویدئو نیز استفاده شود، کاربران انگیزه بیشتری برای مشاهده آنها خواهند شد.

افزودن متنهای متحرک به صفحات وب
مرورگر Internet Explorer، قابلیت منحصر به فردی ارائه داده
است که بوسیله آن می توان متنهایی را در عرض صفحه نمایش به
حرکت درآورد.
با استفاده از این قابلیت که Internet Explorer و نسخه های بعدی،
از آن پشتیبانی می کنند، می توان برای بینندگان صفحات وب،
اطلاعاتی مثل گزارشات، نتایج مسابقات، اخبار جدید و انواع آگهی ها
را به نمایش درآورد.
انواع مختلفی از این متنهای متحرک وجود دارد:
متنهای لغزان، متنهایی هستند که از یک طرف صفحه وارد شده و
وقتی به طرف دیگر می رسند، همانجا می ایستند.
متنهای چرخان، از یک طرف صفحه وارد شده، عرض صفحه را طی
می کنند و بطور کامل از یک طرف دیگر خارج می شوند. و سپس
دوباره از طرف اول وارد صفحه می شوند.

متنهای متناوب، متنهایی هستند که در یک طرف صفحه ظاهر می شوند و سپس به طرف دیگر حرکت می کنند و وقتی به حاشیه مقابل رسیدند، منعکس شده و در منطقه ای که به آنها اختصاص داده شده، جلو و عقب می روند.

استفاده از جداول برای سازماندهی اطلاعات جداول، بطور گسترده ای در صنعت چاپ و نر بکار می روند، جداول نیز مانند لیستها، برای سازماندهی و تنظیم اطلاعات بسیار مفیدند. می توان گفت که تقریباً در تمام مقالات علمی، پایان نامه های تحصیلی و مطالب حرفه ای و تخصصی، جداول به کار گرفته می شوند.

به همین ترتیب، امروزه در صفحات وب نیز از جداول استفاده زیادی می شود.

اولین مرورگری که جداول را ابداع کرد Netscape بود. در حال حاضر، بیشتر مرورگرها از جمله Netscape، Internet Explorer، NCSA Mosaic از جداول پشتیبانی می کنند.

برنامه های یاری دهنده و اتصالی

صفحات اولیه وب، چیزی بیش از مقداری متن آرایش شده که گاه

با تصویری همراه می شدند، در خود نداشتند. گرچه این صفحات می

توانستند حاوی پیوندهائی (Links) به سایر صفحات باشند اما

بسیاری از افراد آنرا انتخابی بهتر از گوفر به حساب نیم آوردند.

باید گفت از هنگام عرض مرورگرهای اولیه، تا کنون، پیشرفتهای

زیادی در آنها صورت گرفته است. امروزه، مرورگرها یک رابطه مهم

و حیاتی برای ایجاد ارتباط با وب جهانی محسوب می شوند.

وب نیز از روزهای اولیه عمرش که شبیه به گوفر بوده، تا کنون

پیشرفت قابل ملاحظه ای داشته است و این پیشرفت همچنان ادامه

دارد.

اما با وجود تمام اینها، باید گفت که گرچه مرورگرها، در طول

سالهای اخیر با توانائی نمایش دادن تصاویر JPEG، توانائی قرار

دادن زمینه برای صفحات و همچنین قابلیت ایجاد زیر صفحه ها، گام

مهمی را به جلو برداشته اند، اما هنوز هم نمی توانند تمامی انواع

اطلاعات را پردازش کرده و نمایش دهند. در واقع، چنین انتظاری هم از آنها نمی رود. اگر مرورگرها می خواستند تمام نیازهای شما را برآورده کنند، تبدیل به برنامه هائی بی نهایت حجیم و دست و پاگیر می شدند. در عوض آنها سعی می کنند در چارچوبه وظایف ۶ خود، به بهترین نحو عمل نمایند.

بریا مثال اگر بخواهید برنامه ای را که بر روی اینترنت قرار دارد اجرا کنید، مرورگر می تواند به شما کمک کرده و آنرا برداشت نماید، اما اگر آن فایل، فشرده سازی شده باشد، مرورگر آنرا باز نخواهد کرد و این کار را به عهده برنامه هائی می گذارد که دقیقاً برای همین منظور ساخته شده اند؛ این برنامه ها نرم افزارهای «کمکی» یا «یاری دهنده» نامیده می شوند.

هنگامیکه مرورگر به فیلی برمی خورد که به خودی خود آن را نمی شناسد، سعی می کند تا آن را دریافت کرده و اجرای آنرا به یک برنامه کمکی واگذار نماید. در مورد مثال برداشت فایل از روی اینترنت، اگر برنامه ای مانند winZip یا stuffIt Explorer داشته

باشد، در صورتیکه مرورگر به درستی تنظیم شده باشد، آن برنامه را اجرا کرده و فایل برداشت شده را به آن تحویل می دهد.

انواع مختلفی از اطلاعات را می توان به این ترتیب، در HTML کار گذاشت.

دستورات ساختاری سند HTML

اگر تا کنون یک سند HTML را مشاهده کرده باشید، کمابیش با

شکل کلی اسناد HTML و ساختار آنها آشنایی دارید. هر سند

HTML باید دارای یک ساختار باشد و برای اینکار باید یکسری

قوانین خاص در آن رعایت شوند. این قوانین چگونگی شروع و پایان

یک سند و چگونگی مرتب شدن اطلاعات موجود در سند را تعیین می

کنند.

دستور `<HTML>` مشخص کننده شروع و پایان یک سند می باشد.

دستور آغازین آن یعنی `<HTML>`، باید در ابتدای هر سند

HTML و قبل از هر چیز دیگری قرار داده شود.

به این ترتیب ابتدای تمامی اسناد HTML به این شکل خواهد

بود: <HTML>

و دستور پایینی آن با </HTML> نیز باید در انتهای هر سند

HTML قرار داده شود: </HTML> سایر مطالب و اطلاعات

موجود در یک سند HTML بین این دو دستور قرار خواهند گرفت.

هر سند HTML شامل دو قسمت «سر» و «تنه» می باشد.

در قسمت «سرب» اطلاعاتی از قبیل نوع سند، تیترا آن و روش ساخته

شدن آن قرار می گیرند، که البته مرورگر آنها را نمایش نخواهد داد.

همچنین، این قسمت مکانی مناسبی برای قرار دادن توابع جاوا و

اسکریپت می باشد. این قسمت به وسله دو دستور آغاگزین و پایانی

<HEAD> و </HEAD> مشخص می شود. مهمترین چیزی که در

قسمت «سر» قرار می گیرد، «تیترا» صفحه وب است. این «تیترا» در

قسمت بالیا پنجره مرورگر به نمایش درخواهد آمد.

برای تعیین تیترا، باید از دستورات <TITLE> و </TITLE>

استفاده کرد.

```
<HTML>
```

```
<HEAD>
```

```
<TITLE>
```

```
This is my web page!
```

```
</TITLE>
```

```
</HEAD>
```

```
.
```

```
.
```

```
.
```

```
</HTML>
```

باید دانست که دستوران HTML را می توان پشت سر هم نوشت.

در واقع رعایت اصول پاراگراف بندی در HTML ضروری نمی

باشد. به عنوان مثال، دستورات بالا را می توان به این شکل نوشت.

```
<HTML>
```

```
<HEAD> <TITLE> This is my web page </TITLE>
```

```
</HEAD>
```

```
.
```

```
.
```

```
.
```

```
</HTML>
```

```
</BODY> , <BODY>
```

پس از قسمت «سر» نوبت به «تنه» می رسد. این قسمت به وسیله دو دستور `</BODY>` , `<BODY>` احاطه می شود.

تمامی متنها و دستوراتی که صفحه وب را خواهند ساخت، باید در این قسمت قرار بگیرند.

به این ترتیب، ساختار کلی یک سند HTML به این صورت خواهد بود.

`<HTML>`

`<HEAD>`

`<TITLE>`

...

`</TITLE>`

`</HEAD>`

`<BODY>`

.

.

.

`</BODY>`

`</HTML>`

استفاده از پیوندها (LINKS) در صفحات وب

در صورتی که آشنایی مختصری با وب داشته باشید، حتماً مفهوم پیوندهای ویژه یا ابر پیوندها (Hyperlinks) را می دانید. بطور معمول شما در هنگام مشاهده صفحات وب، متنهای متمایز شده ای (معمولاً زیر آنها خط کشیده شده) می بینید که با کلیک کردن موی بر روی آنها به یک صفحه دیگر و یا به قسمت دیگری از همان صفحه هدایت می شوید. تمام این کارها را مرورگر انجام می دهد. شما تنها روی یک متن و یا یک تصویر گرافیکی دکمه موس را فشار داده اید. اما این کار چگونه انجام می پذیرد؟ در واقع، نوسنده سند آء، درون آن یک دستور ابر پیوند قرار داده است و بدین ترتیب به مرورگر می فهماند که وقتی کاربر، آن متن یا تصویر خاص را انتخاب کرد، باید به کجا هدایت شود. به عبارت دیگر، هر پیوند دو سر دارد. یک سر آن متنها یا تصاویر ویژه و یا فعالی روی صفحه وب است و سر دیگر آن مقصدی است که کاربر باید به آنجا هدایت شود.

این مقصد می تواند روی همان کامپیوتر سرور (server) که صفحه اول هم روی آن بوده، قرار داشته باشد و یا روی کامپیوتر دیگری که کیلومترها با کامپیوتر اول فاصله دارد و به اینترنت متصل است، باشد.

یک سایت (ایستگاه وب) به کمک این پویندها که در صفحات وب خود قرار می دهد، کاربران را به سوی صفحات دلخواه هدایت می نماید.

در HTML برای ایجاد پیوند، باید از دستور A استفاده کنید. بریا تعیین مقصدی که این پویند، به آنجا اشاره می کند، از ویژگی HREF این دستور استفاده نمائید:

``

تمام متنها با تصاویر گرافیکی که بین دو دستور آغازین و پایانی `<A...>` و `` قرار دهید، در صفحه وب بصورت یک پیوند عمل خواهد کرد و وقتی دکمه موس را بر روی آنها فشار دهید، مرورگر، شما را به مقصد مورد نظر هدایت خوانهد کرد.

برای مثال، به متن زیر که در یک سند HTML نوشته شده توجه کنید:

I have a very beautiful

 Home page .

دستورات لازم برای قرار دادن تصاویر در صفحه وب

برای قرار دادن یک تصویر در صفحه وب، باید از دستور IMG

استفاده کنید. ویژگی SRC این دستور، URL فایل گرافیکی مورد

نظر را مشخص می کند.

بوسیله ویژگی ALT این دستور، می توان متنی را مشخص کرد تا

کاربرانی که نمی توانند تصویر مزبور را دریافت کنند، آن متن را به

جای آن مشاهده نمایند:

<IMG SRC="My_graphic.gif" ALT="This is a graphical
Image">

به علاوه، با استفاده از ویژگی HEIGHT و WIDTH این دستور،

می توان تصویر را به اندازه دلخواه نمایش داد.

<IMG SRC="My_graphic.gif" HEIGHT = 50 WITH =
60>

استفاده از جداول برای سازماندهی اطلاعات

قبل از اینکه جداول وارد HTML شوند، کنترل طراحان صفحات وب بر روی صفحاتی که می ساختند، بسیار کمتر از امروزه بود. شاید به

همین دلیل بود که گرافیسرها تا مدتها از وب بیزار بودند.

در واقع نسخه های جدید HTML، با ارائه جداول، امکان کنترل

بیشتری نسبت به گذشته، به طراحان صفحات وب اهدا کردند.

با استفاده از جداول، نه تنها می توان اطلاعات را طبقه بندی و منظم

نمود، بلکه همانطور که توضیح خواهیم داد، می توان شکل کلی

صفحه و جای قرار گرفتن اجزاء مختلف آنرا نیز تعیین کرد.

یک جدول، از تعدادی سطر و ستون تشکیل شده است. محل تلاقی

یک سطر با یک ستون، یک خانه (cell) جدول نامیده می شود.

در خانه های جدول، علاوه بر متنهای عادی، می توان متنهای ویژه

(Hypertext)، تصاویر گرافیکی و یا هر چیز دیگری که در صفحات

عادی وب، قرار می گیرد، جای داد. همچنین می توان آنها را خالی،

رها کرد.

طرز تعریف جداول به کمک HTML

تعریف جداول در HTML، در ابتدا ممکن است کمی گیج کننده باشد اما با کمی تمرین، بزودی در این کار مهارت خواهید یافت.

به طور خلاصه، برای رسم یک جدول، باید مراحل زیر را به ترتیب انجام داد:

در ابتدا باید مطالب درون جدول را بین دو دستور `<TABLE>` و `</TABLE>` قرار دهید.

سپس مطالب هر سطر از جدول را بقین دو دستور `<TR>` و `</TR>` جای دهید و در نهایت مطالب هر خانه جدول را بین دستورات `<TD>` و `</TD>` بنویسید.

Active Server Page چیست؟

با گذشت ساتلهای متمادی، ما تغییرات مهمی را در موضوع اینترنت مشاهده می نمائیم. 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 ذئق 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

Amazone.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 embeded 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 بر روی 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 نکرده 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 به آدرس

<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

configure your server و Administrative tools و Programs

را انتخاب نمایید.

یک جعبه محاوره ای باز می شود 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

Tolls و Optional Components. بر روی گزینه Optional

Components کلیک کنید. در پنجره سمت راست، توضیحی در

مورد گزینه Optional Components ظاهر می گردد. به جلوتر

بروید و بر روی فرامتن start کلیک کنید. این کار ویزارد اجزای

windows 2000 components را اجرا می نماید -

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 اجزاهای 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، بر روی اجزای IIS در ویزارد اجزای ویندوز 2000 کلیک کنید و سپس

بر روی 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 بر روی 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 و بتان 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

Home Directory tab متمرکز می شویم. به گزینه local path

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 و بتان

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 کردند، به علاوه محصولات

زیر

Apache

Sun web server

Java web server

Netscape Eterprise server

این محصولات همچنین می توانند بر روی کامپیوترهای زیر نصب

کردند

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 مایکروسافتی 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 آجکت می باشد و در یک بانک اطلاعاتی به وسیله سطر نشان داده می شود.

یک فیلد یک خاصیت تنها از یک آجکت 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 آجکت می باشد که دارای خصوصیات معینی می باشد به شما کمک زیادی می نماید. (ما از یک مثال مشابه در روز ششم، «کار

با آجکت 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 آجکت 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 آبجکت 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 آبجکت 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

آبجکت می باشد اضافه کنید، چه کار می کنید؟ آبجکت موتور ممکن است شامل خصوصیتی مثل اسب بخار، لیتر و سیلندر باشد. اگر یک

خاصیت آبجکت، خودش یک آبجکت 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 آبجکت موتور در یک جدول بانک اطلاعاتی موتور ذخیره خواهد شد. از آنجایی که آبجکت 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 کاربر 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 ODBC-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 میکروسافت،

اکسسو 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 خودشان

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 پرجرجی 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 لز 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 |
|---|--------------------------|
| <p>کوکی ها</p> <p>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 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</p> | <p>متغیرهای Session</p> |

| | |
|---|--------------------------------------|
| <p>می نماید، اگر قصد دارید اطلاعات را برای مدت زمانی طولانی نگهداری نمایید، از متغیرهای session استفاده نکنید. همچنین متغیرهای 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 کاربران سایت وب شما وجود دارند. مقایسه 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>متغیرهای Applicati on</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، اسباب بازی می فروشید و لیستی از شماره کارت های اعتباری دارید که نمی خواهید آنها را بپذیرید. 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>فایل های متن</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 مورد استفاده قرار می گیرند. اگر شما فقط نیاز به ذخیره</p> | <p>بانک های اطلاعاتی</p> |

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

مایکروسافت DAO با Data Access objects بود. DAO به وسیله

RDO (Remote Data object) که برپایه 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

مایکروسافت، «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 langage) ذخیره شده باشند. در پاتسخ 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 (برای 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 اطلاعات ODBC

سازگاری پیدا نماید.

ارتباط با بانک اطلاعاتی با استفاده از ActiveX Data (ADO)

Objects

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 Recordset 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 onjConn = 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 مکتبوع 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"

استفاده از یک ارتباط DSN-less

در اینجا یک پیشنهاد برای استفاده از سیستم DSN وجود دارد. به

جای قرار دادن اطلاعات اتصال در داخل DSN می توانید آن را در

داخل یک رشته ارتباطی قرار دهید. یک مثال از این نوع اتصال را

دیروز دیدید. در اینجا ما از رشته ارتباطی به صورت زیر استفاده

می کنیم:

```
objConn. ConnectionString="DRIVER={Microsoft  
Access Driver (*.mdb)};" &-"DBQ=C:\My Document\  
Widget World. Mdb"
```

خط = DRIVER به آبجکت connection می گوید که با چه نوع

بانک اطلاعاتی ارتباط برقرار می نماید.

خط DBQ تعیین می کند که در کجای سرور دهنده بانک اطلاعاتی

قرار داده شده است. این باید مسیر کامل فیزیکی باشد. اگر مسیر

کامل فیزیکی را نمی دانید، از server.MapPath استفاده نمایید. می

توانید مقادیر دیگری را در داخل رشته ارتباطی تعیین نمایید. اگر به

یک بانک اطلاعاتی SQL متصل شده‌اید، ممکن است نیاز به

Uasename و password بریا دسترسی به آن داشته باشید. می

توانید اینها را با PWD,UID معین نمایید. همچنین نیاز به تعیین

اطلاعات SERVER, DRIVER,DATABASW دارید.

باز کردن یک ارتباط

تا اینجا، همه کاری که انجام داده اید ایجاد یک آجکت connection و

گفتن به آن درباره بانک اطلاعاتیتان بود. هنوز به صورت واقعی

ارتباط را باز نکرده اند. برپا انجم این کار، از متد open آجکت

connection استفاده می شود:

```
objConn.open
```

پس اجازه دهید تمام چیزها را اکنون در کنار یکدیگر قرار دهیم:

```
<%
```

```
Dim objconn
```

```
Set objconn=server.createobject("ADODB.
```

```
Connection")
```

```
Objconn. Connectionstring="DSN = widgetworld.dsn"
```

```
Objconn. Opne
```

```
%>
```

این کدها آجکت connection را می سازند، رشته ارتباطی را تنظیم

می کنند و ارتباط را باز می کنند. در اینجا به نظر می رسد که از یک

سیستم DSN استفاده شده است. اگر دوست دارید، می توانید رشته

ارتباطی را برای یک اتصال DSN-less تغییر دهید.

در سایت های وبی که چندین صفحه از یک بانک اطلاعاتی را مورد بررسی قرار دهند. بهتر است که این کدها رادر داخل یک فایل مندرج قرار دهید.

بستن یک ارتباط

همانند همه آجکت ها، وقتی وقتی کارتتان تمام شد باید حافظه را از آجکت های مربوط آزاد نمایید. هر چند، قبل از اینکه این کار را انجام دهید، باید ارتباط را ببندید. آزاد کردن حافظه از آجکت ها connection بدون بستن آن شبیه به گذاشتن گوشی تلفن قبل از قطع کردن مکالمه می باشد. یک ارتباط را می بندید و سپس حافظه را از آجکت آزاد می نماید مانند زیر:

```
objconn. Close
```

```
set objconn= Nothing
```

همیشه بستن و آزاد کردن آجکت های recordset مربوط به یک

ارتباط ، قبل از بستن و آزاد کردن آجکت connection صورت می

گیرد.

اگر کار با یک آبجکت connection را تمام کردید، اما می خواهید یکی دیگر را باز کنید، می توانید از آبجکت connection ی که کار با آن را تمام کرده‌اید. دوباره استفاده نمایید. فقط آن را ببندید، اطلاعات ارتباطی را تنظیم نمایید و سپس آن را دوباره باز کنید.

استفاده از adovbs.inc

در مباحثمان از چندین ثابت موجود در آبجکت Recordset استفاده نموده ایم. این ثابتها به صورت پیش ساخته در ASP وجود ندارند. شما می توانید آنها را هر زمانی که به آنها احتیاج دارید، تعریف نمایید یا می توانید از فایل adovbs.inc که شامل همه ثابتهای تعریف شده برای ADO می باشد، استفاده نمایید. این فایل به همراه PWS, IIS وجود دارد و در مسیر C:\program Files\ common File\system\ado\ به صورت پیش فرض نصب می شود. این فایل را در دایرکتوری ریشه وب خود کپی کنید و آنرا به صفحه ای که می

خواهید از ثابتها در آن استفاده می نمایید. به صورت زیر ملحق

نمایید::

اگر نتوانستید فایل `adovbs.inc` را پیدا کنید، در اینجا ثابتهای که ما

مورد استفاده قرار خواهیم داد، تعریف شده اند:

```
const adopn Forwardonly=0
```

```
const adlockoptimistic=3
```

```
const adlockReadonly=1
```

```
const adcmdtable=2
```

شما می توانید به سادگی این اعداد را هر جایی که نیاز دارید عوض

کنید و استفاده از نام ثابتها را کاملاً فراموش کنید. البته این ایده

خوبی نمی باشد. زیرا این کار خوانایی کدها را از بین می برد. کدام

یک از کدهای زیر قابل فهم تر می باشند. هر دوی اینها ممکن است

برای شما در این لحظه عجیب به نظر برسند، اما احتمالاً متوجه شده

اید که حروف بهتر به نظر می رسند.

```
ObjRS.open "tblusers",objeconn, 0, 1, 2
```

یا

ObjRS.open “tblusers”,objeconn, adopennForwardonly,
adlockReadonly, adcmdtable

Adovs.inc به عنوان یک راه پیشنهادی به جای METADATA

تگ METADATA ممکن است برای وارد کردن ثابتهای نوع

کتابخانه ای به داخل صفحاتتان مورد استفاده قرار گیرد. بنابراین به

عنوان یک پیشنهاد به جای الحاق adovbs,inc می توانید در بالای

صفحاتتان عبارت زیر را قرار دهید:

```
<!--METADATA TYPE = “typelib”  
FILE = “C:\ program Files \ common  
Files\sustem\ado\msado15.dll-->
```

Msado15.dll یک فایل کتابخانه ای ADO می باشد. به صورت پیش

فرض این فایل در داخل دایرکتوری C:\ program Files \

common Files\sustem\ado قرار دارد. منوط به اینکه از چه

نسخه ای از ADO استفاده می کنید. شماره فایل ممکن تفاوت داشته

باشد. برای پیدا کردن این فایل از عبارت msado*.dll استفاده

نمایید.

استفاده از تگ METADATA در یکی از صفحاتتان، ممکن است باعث ارث بردن ثابتهای تعریف شده در فایل کتابخانه‌ای گردد. همچنین می‌توانید این عبارت را در داخل فایل `global.asa` قرار دهید، که در این صورت تمام صفحاتتان به این ثابتها دسترسی خواهند داشت.

انواع قفل گذاری

وقتی شما یک بانک اطلاعاتی را بر روی وب ایجاد می‌نمایید، ممکن است صدها یا هزاران کاربر را داشته باشید که در یک لحظه به آن دسترسی پیدا می‌نمایند. این کار تنها در صورتی خوب می‌باشد که آنها فقط قصد خواندن اطلاعات از داخل بانک را داشته باشند. اگر دو نفر سعی کنند یک رکورد را در یک لحظه تغییر دهند، فکر می‌کنید که چه اتفاقی می‌افتد؟ این عمل می‌تواند باعث ایجاد انواع مشکلات گردد. برای اجتناب از این کار، اولین نفری که سعی می‌کند رکورد را تغییر دهد یک «قفل» بر روی آن رکورد می‌گذارد. هنگامی که رکورد

قفل گذاری شد، هیچ کس نمی تواند بر روی رکورد تغییر را اعمال نماید. هنگامی که کاربر کارش را تمام می کند، قفط از بین می رود.

از آنجایی که شما نیازی به ایجاد تغییراتی بر روی بانک اطلاعاتی نداشتید. رکوردست را به صورت فقط خواندنی بارگذاری کردید. اگر

قصد ندارید تغییری بر روی رکورد انجام دهید، هیچ اهمیتی ندارد که رکوردها را قفل گذاری نمایید. ولی ، امروز نیاز به نوشتن بر روی

بانک اطلاعاتی دارید. این به معنی آن است که نوع قفلی که دیروز مورد استفاده قرار دارید و "adlockReadonly" نام داشت برای

کار ما کفایت نمی کند. چندین نوع قفل گذاری وجود دارند که شما می توانید از آنها استفاده نمایید. تفاوت بین آنها مربوط به مواقعی است

که ما عمل قفل گذاری را انجام می دهیم.

ثابتی که شما اکنون مورد استفاده می دهید adlockoptimistic نام دارد که مقدار آن 3 می باشد.

اضافه کردن و به روز رسانی (ADDNEW and update)

آبجکت Recordset دو متد دارد که شما می توانید از آنها برای تغییر

بر روی بانک اطلاعاتی استفاده نمایید: متد update و متد AddNew

متد AddNew یک رکورد جدید را در داخل رکوردست ایجاد می

نماید. رکورد جدید ا تا زمانی که متد update فراخوانی نشود، به

بانک اطلاعاتی اضافه نمی گردد. بعد از فراخوانی متد AddNew

رکورد جدید به عنوان رکورد فعلی قلمداد می گردد. و حتی بعد از

فراخوانی متد update به عنوان رکورد فعلی باقی می ماند.

در اینجا چگونگی استفاده از متدهای update و AddNew را به شما

نشان می دهیم. فرض کنید که رکوردست objRS به درستی باز شده

است:

```
objRS.AddNew
```

```
objRS("Name")="Bill"
```

```
objRS("Email")="whois @ Nothing.net"
```

```
objRS.Update
```

متد AddNew یک رکورد جدید خالی ایجاد می نماید و آن را به

عنوان رکورد فعلی تنظیم می نماید. شما می توانید مقادیری را به

راحتی در داخل فیلدهای آن قرار دهید. وقتی شما کار قرار دادن

مقادیر را به پایان رساندید، برای اضافه کردن رکورد به جدول بانک اطلاعاتی از متد update استفاده نمایید.

اگر هنگامی که شما در انتهای یک رکورد قرار دارید متد AddNew را فراخوانی نمایید، متد update به صورت خودکار فراخوانی می‌گردد. بنابراین دستورات زیر تاثیر ایجاد یک رکورد جدید و افزودن آن به جدول، و ایجاد رکورد جدید دوم و افزودن آن به جدول را دارند:

```
objRS AddNew
```

```
objRS("Name")="Bill"
```

```
objRS("Email")="whois @ Nothinng.net"
```

```
objRS. AddNew
```

```
objRS("Name")="jim"
```

```
objRS("Email")="jl @ Nothinng.net"
```

```
objRS Update
```

این کار مقداری به خوانایی کدهایتان لطمه می‌زند، پس انجام این کار توصیه نمی‌کنیم. به عنوان یک پیشنهاد به جای استفاده از عملگر

مساوی برای دادن مقدار به فیلد، شما می توانید فیلدها را به وسیله
متد AddNew Ilnhv nid مقدار دهی نمایید.

objRS AddNew files , values

فیلدها و مقادیرشان، مقادیر منفرد یا آرایه هایی با همان تعداد عنصر
می باشند.

به روز رسانی رکوردها

شما همچنین می توانید از متد Update برای انجام تغییرات در

رکوردهای موجود استفاده نمایید. به جای فراخوانی متد AddNew

به سراغ رکوردهای که قصد تغییر دادن آن را دارید بروید. مقادیر

فیلد را با استفاده از عملگر مساوی، همان طوری که در بخش قبلی

مشاهده نمودید تنظیم نمایید. وقتی که کار شما تمام شد، متد

Update را فراخوانی نمایید:

```
objRS("username")="Fred"
```

```
objRS("email")="fr2 @ whatever.net"
```

```
objRS Update
```

دو خط اول مقادیر فیلدهای email, username رکوردهای فعلی را

تغییر می دهند. سپس، با فراخوانی متد Update این تغییرات در داخل

بانک اطلاعات ذخیره می گردند.