

**چکیده:**

موضوعات مطرح شده در این پایان نامه شامل برنامه نویسی تحت وب با

استفاده از نرم افزار HTML جهت طراحی صفحات استاتیک، ASP

جهت داینامیک کردن صفحات ساخته شده توسط HTML، پایگاه داده

SQLSERVER برای ذخیره سازی داده‌هایی که از طریق صفحات

دریافت می شود، Visual Interdev جهت ایجاد یک پروژه پایگاه داده

می باشد.

در فصل اول به بررسی مطالب آموخته شده پرداخته ایم و در فصل دوم

کار عملی انجام شده عنوان شده است.

صفحه	عنوان
	فهرست مطالب
	فصل اول: مقدمه
۱	۱-۱-۱- مقدمه
	فصل دوم: آموخته‌ها
	بخش اول HTML
۹	۱-۱-۲- HTML چیست
۱۲	۲-۱-۲- طریق آدرس دهی برای صفحه Web
۱۵	۳-۱-۲- HTML tag های
۲۱	۴-۱-۲- ایجاد جدول در یک صفحه وب
۲۳	۵-۱-۲- فرمها در HTML
۲۹	۶-۱-۲- فریمها در HTML
۳۲	۷-۱-۲- Style Sheet
۳۷	۸-۱-۲- متدهای Window در Javascript
۴۳	۹-۱-۲- رویدادهای مهم Button

۴۷ ۲-۱-۱۰- نمایش تاریخ و ساعت در یک صفحه وب

۴۸ ۲-۱-۱۱- نمونه ها و مثالها

### بخش دوم: ASP

۵۷ ۲-۲-۱- محتویات دینامیک و ASP

۵۹ ۲-۲-۲- نوشتن صفحات ASP

۶۲ ۲-۲-۳- مبانی ASP

۶۵ ۲-۲-۴- دیباک کردن برنامه

۶۶ ۲-۲-۵- ODBC و انتخاب پایگاه داده

۶۹ ۲-۲-۶- ایجاد اتصال به پایگاه داده

۷۱ ۲-۲-۷- مبانی SQL

۸۱ ۲-۲-۸- نمونه ها و مثالها

### بخش سوم: SQL Server

۹۲ ۲-۳-۱- ایجاد یک پایگاه داده

۹۴ ۲-۳-۲- حذف پایگاه داده

۹۵ ۲-۳-۳- مجوزها در SQL Server

- ۱۰۳ ۲-۳-۴-انواع داده و ایجاد جداول
- ۱۲۵ ۲-۳-۵-بازیابی داده‌ها با دستور SELECT
- ۱۲۸ ۲-۳-۶-پردازش داده‌ها
- ۱۴۱ ۲-۳-۷-مرتب سازی داده‌ها با عبارت ORDERBY
- ۱۴۳ ۲-۳-۸-دستور SELECT INTO
- ۱۴۷ ۲-۳-۹-درج داده (INSERT)
- ۱۵۱ ۲-۳-۱۰-حذف داده (DELET)
- ۱۵۵ ۲-۳-۱۱-به کارگیری دستور UPDATE
- بخش چهارم: Visual Interdev
- ۱۵۹ ۲-۴-۱-ایجاد یک پروژه ویژوال Interdev
- ۱۶۰ ۲-۴-۲-بررسی واسط ویژوال Interdev برای یک پروژه
- ۱۷۴ ۲-۴-۳-توسعه و تکمیل صفحات وب به وسیله ویژوال Interdev
- ۱۷۸ ۲-۴-۴-طراحی وب سایت با Sitedesigner
- ۱۸۴ ۲-۴-۵-تنظیم گزینه های ASP در ویژوال Interdev
- ۱۸۷ ۲-۴-۶-کاربرد فایل global.asa

۱۸۹

۲-۴-۷- ایجاد یک پروژه پایگاه داده

فصل سوم: کار عملی انجام شده

۱۹۵

شرح پروژه

۲-۳-۱- ایجاد یک پایگاه داده

برای ایجاد یک پایگاه داده جدید 2000 Server S یکی از سه روش زیر

را بکار برید

–ویزارد Database Creation

–برنامه Enterprise Manager

–دستور CREATE DATABASE

برای ایجاد یک پایگاه داده جدید یک کپی از پایگاه داده Model ایجاد

می شود. پس از آن پایگاه داده به اندازه‌ای درخواستی بسط می‌یابد و

فضای اضافی با صفحات خالی تکمیل می‌شود.

داده‌های پایگاه داده به شکل فیزیکی بر روی دیسک نگهداری می‌شوند. هنگام ایجاد یک پایگاه داده جدید لازم است حداقل یک فایل برای ذخیره داده‌ها و جداول سیستم و یک فایل دیگر برای نگهداری فایل تراکنش‌ها مشخص کنید. پایگاه داده و فایل تراکنش می‌توانند چندین فایل را به خود اختصاص دهند. توجه: فایل‌های پایگاه داده نمی‌توانند با فایل‌های سایر پایگاه‌های داده یا گزارش تراکنش به اشتراک گذارده شوند.

برای ایجاد یک پایگاه داده جدید می‌توانید Enterprise Manager را بکار برید. مراحل زیر را دنبال کنید:

۱- برای شروع Enterprise Manager را از منوی Programs Start

و SQL Server 2000 انتخاب کنید.

۲- به نسخه SQL متصل شوید.

۳- پوشه Database را بسط دهید

۴- آیکون پوشه Database یا محلی از ناحیه راست را کلیک راست کنید

و از منو گزینه New Database را انتخاب کنید.

۵- در اینجا لیست Database Properties در برگه General قابل مشاهده است.

نام پایگاه داده را مشخص کنید. حال به برگه Data File بروید تا فایل جدید با اندازه اولیه یک مگابایت را در پوشه پیش گزیده Data\... مشاهده کنید. در بخش FileDatabase در انتهای دیالوگ، گزینه Automatically Grow قرار دارد.

حداکثر اندازه فایل نیز نامحدود (Unrestricted File Growth) مشخص شده است.

شکل زیر ناحیه سمت راست پایگاه‌های داده را نشان میدهد و گزینه منو امکان انجام عملیات را فراهم می‌کند.

۶- برای تغییر خواص فایل‌های پایگاه داده، جعبه مناسب را کلیک کنید و تغییرات را اعمال نمایید. برای ایجاد سایر فایل‌های پایگاه داده می‌توانید به جعبه File Name بروید و خواص سایر فایل‌ها را مشخص کنید.

۷- برگه Transaction را کلیک کنید.

۸- پس از خاتمه Ok را کلیک کنید. اگر پایگاه های ایجاد شده را

مشاهده نمی کنید، بر روی پوشه Database راست کلیک و گزینه

Refresh را انتخاب نمایید.

۲-۳-۲- حذف پایگاه داده

برای حذف یک پایگاه داده از سیستم Transact\_SQL یا Enterprise

بکار می رود. قبل از حذف به نکات زیر توجه کنید:

حذف پایگاه داده اطلاعات آن را از جداول سیستم حذف کرده و کلیه

داده ها و گزارش تراکنشها را از سیستم پاک می کند.

یک پایگاه داده حذف شده را می توان فقط از طریق احیاء آن با نسخه

پشتیبان مجدداً ایجاد کرد.

در هنگام حذف پایگاه داده هیچ کاربری نباید در پایگاه داده باشد.

در هنگام صدور دستور DROP DATABASE باید در پایگاه داده

Master باشید.

حال به نگارش این دستور توجه کنید:

DROP DATABASE database\_name1, database\_name2



برای حذف پایگاه داده از طریق Enterprise Manager مراحل زیر را

دنبال کنید:

۱- Enterprise Manager را اجرا کنید و به پوشه Database بروید.

۲- بر روی پایگاه داده Leap کلیک راست کنید و گزینه Delete را

انتخاب نمایید.

۳- در جعبه تایید عمل حذف Yes را کلیک کنید.

با اینکار پایگاه داده و فایل‌های داده‌ای و تراکنش از سیستم حذف می

شوند.

۲-۳-۳- مجوزها در SQL Server

انواع مجوزها

2000 SQL Server سه واژه اعطا (GRANT) منع (DENY) و باز پس

گیری (Revoke) را در ارتباط با مجوزها بکار می برد.

برای آنکه کاربر بتواند عمل خاصی را انجام دهد باید مجوز آن به وی

اعطا شود. برای جلوگیری از اجرای یک عمل توسط کاربر، کاربر

از اجرای آن منع می شود و برای حذف مجوز اعطا شده قبلی، مجوز باز

پس گرفته می شود.

شما می‌توانید دو نوع مجوز اعطا کنید: مجوزهای سطح دستور و سطح شیء. مجوزهای سطح دستور به کاربر اجازه می‌دهند دستور T-SQL خاصی را اجرا کند، در حالیکه مجوز شیء، مجوز اجرای عملیات خاص نظیر SELECT, INSERT, UPDATE یا DELETE داده‌ها را به کاربر می‌دهد.

تقدیم مجوزها (Permissions Precedence)

درک چگونگی اعمال مجوزها برای آگاهی از انتخاب مجوزهای موثر ضروری است. کلیه مجوزها در SQL Server جمع شونده (Cumulative) هستند. مجوز DENY بر سایر مجوزها غالب است، استثناء محسوب می‌شود.

اگر مجوز SELECT را از عضویت خود در نقش ۱ و مجوز INSERT را از عضویت خود در نقش ۲ دارید در حقیقت هر دو مجوز INSERT و SELECT را در اختیار دارید. اما اگر دستور DENY بکار رود بر کلیه مجوزها غالب شده و آنها را از شما می‌گیرد..

مجوزهای خاص SQL Server

SQL Server 2000 دارای سطوح متعدد مجوزهاست و بسیاری از آنها مختص پایگاه داده می باشند همانطور که قبلاً ذکر شد نقشهای ثابت سرویس دهنده به Login ها بسته می شوند و نه به کاربران پایگاه داده و هر نقش بر مجموعه ای خاص از مجوزها دلالت می کند.

در هر پایگاه داده نقشهای ثابت سرویس دهنده ای وجود دارد که با مجموعه خاصی از مجوزها مرتبط است. هر پایگاه داده دارای کاربری به نام dbo (مالک پایگاه داده) است. اگر چه در SQL Server مستقیماً اطلاعاتی در مورد آن نمی بینید اما مفهومی به نام مالک شیء پایگاه داده Database Object Owner وجود دارد و مجوزهای خاصی به افراد این نقش به ارث می رسد.

#### مجوز CREATE DATABASE

این مجوز به کاربران اجازه می دهد برای خود پایگاه داده ایجاد کنند و dbo آن شوند. مالکیت پایگاه داده را می توان بعداً تغییر داد.

از آنجاییکه مجوزها به کاربران اعطا می شود و نه Login ها بنابراین این این مجوز را فقط در پایگاه Master اعطا کنید. این مجوز در هیچ پایگاه داده دیگری وجود ندارد. مجوز CREATE DATABASE حقوق

استفاده از دستور ALTER DATABASE را نیز به شما می‌دهد به عبارت دیگر نمی‌توانید دستور ALTER را بکار برید مگر آنکه مجوز CREATE DATABASE را داشته باشید.

نکته: بکارگیری نقش سرویس دهنده dbcreator بهتر از اعطای مجوز دستور CREATE DATABASE است زیرا بهر حال به سایر مجوزهای اعطا شده dbcreator نیاز دارید. علاوه بر آن تشخیص حقوق اشخاص از طریق نقش‌های SQL Server ساده‌تر است.

مجوزهای CREATE TABLE, VIEW, FUNCTION, PROCEDURE, DEFAULT, RULE این مجوزها به کاربران امکان می‌دهد دستورات ایجاد اشیاء را در پایگاه داده‌ای که مجوز آن را در اختیار دارند اجزا نمایند. این مجوزها بطور مکرر به برنامه نویسان اعطا می‌شود زیرا ایشان در حین تولید و توسعه پایگاه داده نیاز به ایجاد منابع متفاوت دارند.

نکته: مجوزهای ALL CREATE شامل حقوق حذف و تغییر اشیاء ایجاد شده توسط یک کاربر است. اعطای این مجوز می‌تواند مشکلات جدی را بوجود آورد زیرا به کاربر امکان می‌دهد پس از خاتمه کار با یک

شیء آن را حذف کند. یک کاربر ممکن است شیئی را تغییر دهد و آن را برای سایرین غیر قابل استفاده کند.

تخصیص مجوزهای دستورات

برای اعطاء، رد یا باز پس گیری مجوزهای دستورات می توانید

Enterprise Manager , T-SQL را بکار برید

دستور GRANT

این دستور مجوزهای دستورات را به کاربر اعطا می کند.

GRANT {ALL | Statement\_List} to {Account}

در این نگارش:

ALL- کلیه مجوزهای ممکن است

Statement\_List لیست شمارشی مجوزهای دستوری است که می

خواهید به یک حساب اعطا کنید.

Account- نام کاربر پایگاه داده، نقش پایگاه داده، کاربر ویندوز یا

گروه ویندوز است.

دستور REVOKE

این دستور کلیه مجوزهای اعطا شده قبلی را باز پس می گیرد:

REVOKE {ALL | Statement\_List} to {Account}

در این نگارش:

Statement\_List لیست شمارشی مجوزهای دستوری است که می

خواهید باز پس بگیرید

دستور DENY

فرض کنید کاربری (مثل Joe) عضو نقش پایگاه داده‌ای با مجوز

CREATE DATABASE می باشد بنابراین Joe نیز این مجوز را

داراست. اما اگر مایل نیستید این شخص حتی با این عضویت توانایی

ایجاد جدول داشته باشد، می‌توانید دستور DENY را بکار برید:

DENY {ALL | Statement\_List} to {Account}

در این نگارش کلیه پارامتر مشابه دو دستور قبل می‌باشد.

راهبری مجوزهای دستورات از طریق Enterprise Manager

Enterprise Manager برای پیاده سازی مجوزهای دستورات یک واسط

گرافیکی فراهم آورده است. برای مشاهده یا ویرایش مجوزها، پوشه

Database را بسط دهید و بر روی پایگاه داده مورد نظر کلیک راست

کنید. سپس Propertise را انتخاب کنید. برگه Permissions را کلیک کنید تا مجوزهای پایگاه داده را مشاهده کنید.

برای اعطا یا بازپس گیری مجوزها، جعبه‌ها می توانند حاوی یکی از سه نشانه زیر باشند:

—علامت چک مارک به معنای اعطای مجوز است.

—علامت \* قرمز رنگ به معنای رد مجوز است.

—جعبه خالی نشان دهنده عدم تخصیص صریح هر گونه مجوز است.

برای اعطای یک مجوز علامت چک مارک را در جعبه مناسب هر حساب

قرار دهید. برای رد مجوز بر روی جعبه دوبار کلیک کنید تا علامت \*

قرمز رنگ پدیدار شود. اگر مجوزی قبلاً اعطا شده باشد، کلیک جعبه

سبب نمودار شدن علامت \* قرمز رنگ میشود. برای ارسال دستور

REVOKE جعبه را با کلیک مجدد خالی کنید. برای دائمی شدن

تغییرات OK را کلیک کنید.

هشدار:

توانایی ایجاد اشیاء در پایگاه داده نکته مهمی است بنابراین تنها در

صورت نیاز این مجوز را به کاربران اعطا کنید.

## مجوزهای اشیاء (Object Permissions)

این مجوزها به کاربر، نقش، گروه یا کاربر ویندوز اجازه می دهند تا عملیاتی را بر روی اشیاء خاص پایگاه داده انجام دهند. این مجوزها فقط برای اشیاء مشخص شده در هنگام اعطای مجوز اعمال می شود با اینکار می توان حقوق اجرای دستورات T-SQL بر روی اشیاء را به تک تک کاربران اعطا کرد. این مجوزها متداول ترین نوع مجوزهای اعطایی به کاربران می باشند.

### مجوزهای Object در دسترس عبارتند از:

مشاهده داده در جدول دید یا ستون	SELECT
افزودن داده به جدول یا دید	INSERT
تغییر داده های موجود در جدول، دید یا ستون	UPDATE
حذف داده از جدول یا دید	DELETE
اجرای روال ذخیره شده	EXECUTE
ارجاع به جدول که دارای کلید خارجی است یا ایجاد	Reference:
تابع یا دیدی با گزینه SCHEMABINDING که به یک	



شیء ارجاع می کنند.

توجه:

مجاز REFERENCES در Enterprise Manager به شکل مخفف DRI

بکار می رود و به کاربر (یا برنامه های کاربردی) اجازه می دهد مقداری

را با مقادیر جدول دیگر مقایسه کند، بدون آنکه داده های جدول دیگر

را واقعا مشاهده نماید.

یک کاربرد جدید در SQL Server 2000 , Schema binding نامیده

می شود.

و کار آن ممانعت از تغییر اشیایی است که در دیدها یا توابع بکار برده

اید و به آن وابسته هستید.

تخصیص مجوزهای اشیاء:

برای اعطاء، رد و باز پس گیری مجوزهای اشیاء می توانید Enterprise

Manager T\_SQL را بکار برید.

## اعطای مجوزهای اشیاء از طریق Enterprise Manager

مدیریت مجوزهای اشیاء بخشی از عملیات راهبری سیستم است و اعطا و بازپس گیری این مجوزها از کارهای رایجی است که در طول روز انجام خواهید داد.

Enterprise Manager یک روش سهل، سریع و بصری را برای کنترل مجوزهای اشیاء فراهم آورده است. در این روش مشاهده مجوزها بر

اساس اشیاء یا کاربران امکان پذیر است. این توانایی پیگیری خطاها را

نیز تسهیل می کند

مشاهد مجوزهای یک شیء

برای مشاهده یا تغییر مجوزهای اشیاء در Enterprise Manager مراحل

زیر را دنبال کنید.

۱- پوشه database را برای پایگاه داده مورد نظر بسط دهید و آیکون

نوع شیء را برجسته کنید.

۲- بر روی شیء کلیک راست کنید و گزینه Properties را انتخاب

نمایید.

۳- دکمه Permissions را کلیک کنید.

شما می‌توانید مشاهده کلیه کاربران، گروهها، کاربران و گروههای ویندوز یا گروههای در دسترس پایگاه داده را انتخاب کنید و یا لیست کلیه حسابهایی که بر روی اشیاء قابل مشاهده دارای مجوزند را انتخاب کنید بر که Object Properties درست مشابه بر که Statement Permission عمل می‌کند.

برای اعطای مجوز، جعبه مجاور آن را علامتگذاری کنید، برای DENY یک علامت \* قرمز در جعبه قرار دهید و برای باز پس گیری مجوز جعبه مناسب را پاک کنید. پس از اعمال تغییرات دکمه Apply یا OK را کلیک کنید. برای تنظیم مجوزهای سطح ستون، دکمه Column را کلیک کنید. SQL Server تنها مجوزهای مناسب با نوع شیء انتخاب شده را نمایش می‌دهد.

مشاهده مجوزها برای یک کاربر یا نقش پایگاه داده شما می‌توانید مجوزها را برای یک کاربر یا نقش مشاهده کنید. برای مشاهده یا تغییر مجوزها در Enterprise Manager بر مبنای نقش یا کاربر، مراحل زیر را دنبال کنید:

۱- پوشه Database را برای پایگاه داده مورد نظر بسط دهید و آیکون Database یا Roles Database را بر جسته کنید.

۲- بر روی کاربر یا نقش کلیک راست کرده و گزینه Properties را انتخاب کنید.

۳- دکمه Permissions را کلیک کنید.

۴- جعبه های مناسب را برای اعطاء، یا باز پس گیری مجوزها کلیک کنید. پس از تنظیم مجوزها، Apply یا OK را برای دائمی کردن تغییرات کلیک کنید.

۲-۳-۴- انواع داده و ایجاد جداول

### جداول

جداول کلیه داده های پایگاه داده را به شکل مجموعه ای از سطرها و ستونها (رکوردها و فیلدها) سازماندهی و ذخیره می کنند. هر ستون در جدول قابلیت نگهداری یک نوع داده (نوع خاصی از اطلاعات) را داراست.

هر جدول مجزا در پایگاه داده معرف یک موجودیت (entity) و هر سطر جدول نمودی (ocurance) از آنست. ستون های جدول، مشخصه های

(attributes) موجودیت منطقی را بیان می کنند. برای مثال در جدول فرضی کارمندان (کارمندان موجودیت منطقی هستند) هر سطر نشاندهنده یک کارمند و ستون های تشکیل دهنده جدول، مشخصه های ایشان نظیر شماره کارمندی، نام و نام خانوادگی و شماره تأمین اجتماعی است.

### ستونها

هر ستون (فلید) در جدول دادهای دارای یک نام، نوع داده، طول (اختیاری)، collation (اختیاری) و قابلیت ذخیره سازی مقدار تهی است. ستون ها را می توان با هر ترتیبی در تعریف جدول قرار داد اما لازم است تا نام یکتایی در جدول داشته باشند و از قوانین شناسه های (identifiers) SQL Server تبعیت کنند.

هر پایگاه داده می تواند دو میلیون جدول با ۱۰۲۴ ستون در هر جدول را دار باشد. حداکثر طول هر سطر ۸۱۹۲ بایت است که سربار (overhead) ذخیره سازی فیزیکی سطرها در SQL Server از آن کم می شود. این سربار متغیر است اما بطور کلی حداکثر طول داده های هر سطر ۸۰۶۰ بایت می باشد.

حداکثر طول هر ستون ۸۰۰۰ بایت است و می توان در هر سطر ۱۰۲۴ ستون داشت. در هر حال یک سطر نمی تواند از یک صفحه داده ای (Data Page) برابر با ۸۱۹۲ بایت برای داده و سربار بیشتر شود مگر نوع داده ای متن (text) و تصویر (image) که می تواند تا دو گیگابایت اطلاعاتی متنی یا باینری را ذخیره کند صفحه داده ای برای آنها اعمال نمی شود.

### شناسه های SQL Server

نام جداول ، ستون ها و سایر اشیاء در SQL Sever تابع قوانین زیر است:

- طول شناسه از ۱ تا ۱۲۸ کاراکتر Unicode شامل حروف، اعداد و علائم است.

- نخستین کاراکتر باید حرف یا یکی از علائم @ , # یا -

- (Underscore) باشد. @ , # معانی خاصی در SQL Sever دارند.

- علائم @ , # و - را می توان پس از کاراکتر نخست بکار برد.

- شناسه هایی که با علامت @ شروع می شوند بعنوان متغیرهای

محلی بکار می روند. این علامت می تواند فقط در کاراکتر نخست

بکار رود.

• شناسه هایی که با علامت # شروع می شوند اشیاء موقت

(temporary) را نشان می دهند و در طول نشست (session) توسط

کاربر قابل ارجاع می باشند.

• بکارگیری فاصله (space) در شناسه به شرط قرار گیری آنها در ”

Employee [ ] در زمان ارجاع بلامانع است. برای مثال جدول

Pension را هنگام ارجاع به شکل ”Employee Pension“ یا [

Employee Pension ] بکار برید تا بارشته های ثابت در شناسه ها

اشتباه نشوند.

نکته:

اگر چه بکارگیری فاصله در شناسه ها مجاز است اما بسیاری از برنامه

نویسان بدلیل احتمال بروز اشتباه از آن اجتناب می کنند.

انواع داده

نوع داده مشخص کننده نوع اطلاعات (عددی، کاراکتری و ...)

فضای اشغال شده توسط یک ستون خاص است. برخی انواع داده

طول ثابت و برخی دیگر طول متغیر دارند. انواع داده ای در SQL

Server به گروه های رشته ای (string)، Unicode، باینری، عدد

صحیح (integer)، اعداد دقیق (exact) و تقریبی (approximate)،

تاریخ و ساعت، خاص، پولی اضافه شونده (auto-incrementing)،

مترادف ها (synonyms)، تعریف شده کاربر (user-defined) و

محاسباتی (computed) تقسیم می شوند.



## جدول انواع داده‌ای در SQL Server 2000

گروه	نوع داده	توضیحات
رشته String	Char (n), varchar(n)	رشته های کاراکتری ذخیره می کند
باینری Binary	Binary(n)	اطلاعات باینری را به شکل زوج های دوبایتی ذخیره می کند.
اعداد صحیح Integer	Int, smallint, bigint, tinyint	اعداد صحیح را ذخیره می کند
اعداد تقریبی Approximate	Float, real	اطلاعات اعداد تقریبی را ذخیره می کند
اعداد دقیق Exact	Decimal, Numeric	اطلاعات اعداد دقیق را ذخیره می کند
خاص Special	Bit, text, image, sql-variant, table	یک بیت ، اطلاعات کاراکتری بیش از ۸۰۰۰ بایت یا تصاویر را ذخیره می کند. sql-variant برای ذخیره هر نوع داده و table برای ذخیره جداول بکار می رود.
تاریخ و ساعت Data&Time	Datetime	تاریخ و ساعت ذخیره می کند.
پولی Money	Smallmoney, money	مقادیر پولی را ذخیره می کند.
اضافه شونده خودکار Autoincrementing	Rowversion (timestamp)	مقادیر داده‌ای اضافه شونده یا قابل تنظیم SQL Server را ذخیره می کند.
مترادف Synonyms		نگاشت انواع داده‌ای ANSI با

SQL Server را انجام می دهد

داده unicode Nchar, ntext, nvarchar داده ها را با فرمت دو بایت بازای

هر کاراکتر ذخیره می کند.

انواع داده های جدید برای ذخیره

اطلاعات

تعریف شده کاربر

User-define

عبارت محاسباتی ستون ها را

محاسباتی

ذخیره می کند.

Computed

قابلیت ذخیره سازی مقدار تهی (Nullability)

قابلیت ذخیره سازی مقدار تهی در ارتباط با الزام ورود داده در یک

ستون است. اگر می خواهید ستونی بتواند مقدار نامشخص داشته باشد

Null بکار برید. اگر می خواهید کلیه ستونها در هر سطر حتماً مقدار

داده‌ای داشته باشند Not Null را بکار برید. اگر مقدار Null یا Not

Null مشخص نشود، مقدار پیش گزیده پایگاه داده یا اتصال فعلی

SQL Server در نظر گرفته می‌شود. در هنگام نصب اولیه SQL

Server مقدار پیش گزیده برای کلیه ستونها Not Null است، اما مشابه

سایر گزینه ها حتی بر مبنای هر اتصال (connection) قابل تغییر است

بنابراین بهتر است مشخصه Null یا Not Null را در هنگام تعریف هر

ستون مشخص کنید.

Collate

برای هر ستون در جدول می توان Collation تعریف کرد. در غیر

اینصورت پیش گزیده SQL Server بکار می رود.

رشته ها (Strings)

رشته حاوی داده های کاراکتری نظیر حروف، اعداد و علائم است و می تواند با استفاده از Char(n) یا varchar (n) طول ثابت یا متغیر داشته باشد. حداکثر کاراکترهای قابل ذخیره ۸۰۰۰ بایت است.

در هنگام تعریف فیلد با طول ثابت مشخص می کنید این فیلد همواره شامل n بایت اطلاعات است. بنابراین اگر داده ورودی کمتر از n بایت باشد، فیلد با space پر می شود تا به n بایت برسد. همچنین در صورت ورود داده بیش از n بایت، داده های اضافی حذف می شود.

در فیلدهای با طول متغیر حداکثر طول مشخص می شود اما بر خلاف فیلدهای با طول ثابت با فاصله پر نمی شود بکارگیری این نوع داده از نظر استفاده مفید از حافظه مؤثرتر است اما کارایی را تا حدی کاهش

می دهد. اگر فیلدی را طول متغیر تعریف کنید لازم است تا SQL Server نقاط خاتمه فیلد و شروع فیلد جدید را تعیین کند. سربار دیگر مربوط به بایت هایی است که بازای فیلدهای متغیر به سطرهای جدول مناسب است. برای ورود داده های کاراکتری در SQL Server باید داده ها ما

بین علائم ' یا ' ' قرار داده شوند.

علائم بدلیل جلوگیری از بروز اشتباه با شناسه های SQL Server و ثابت

های رشته ای ترجیح دارد. برای ورود مقدار Null برای یک فیلد، کلمه

کلیدی Null را بکار برید

نکته :

پیشوند گذاری نام ستونها خوانایی را بیشتر می کند به طراحان و برنامه نویسان در تشخیص نوع داده هر فیلد کمک می کند. برای مثال فیلد Social Security Number نوع عددی یا کاراکتری را مشخص نمی کند. اما اگر نام فیلد به Social Security Number تغییر کند نوع داده ای کاراکتر را بیان می کند.

داده باینری (Binary)

این نوع داده مجموعه ای از صفرها و یکها را ذخیره می کند و در ورودی و خروجی به شکل زوج هگزادسیمال نمایش داده می شود. این ازواج از کاراکترها ۰ تا ۹ و حروف A تا F تشکیل شده اند. برای مثال فیلدی با تعریف `binSome Data binary(20)` نشان می دهد می خواهید ۲۰ بایت داده باینری ذخیره کنید.

مشابه نوع داده رشته ای می توان تا ۸۰۰۰ بایت را در انواع داده `binary(n)` یا `Varbinary(n)` ذخیره کرد. در حالت طول ثابت، اطلاعات ورودی با فاصله (0X20) پر می شود و اطلاعات بزرگتر از حداکثر طول مجاز قطع می شوند.

در ابتدای اطلاعات داده های باینری از 0X استفاده کنید. برای مثال مقدار ۱۰ در فیلد باینری به شکل 0X10 وارد می شود.

اعداد صحیح (Integer)

چهار نوع داده صحیح tinyint , Smallint , int , bigint هستند و

مقادیر دقیق و قابل اندازه گیری (scalar) ذخیره می کنند. تفاوت آنها

در حافظه مورد نیاز و دامنه اعداد قابل ذخیره است. جدول زیر این

چهار نوع را نشان می ده.

دامنه	طول	نوع داده
0-255	1	Tinyint
+32, 767	2	Smallint
+2, 147 , 483, 657	4	Int
+2 <sup>63</sup>	8	Bigint

توجه: رسوم نامگذاری متغیرها در لیست فوق عبارتند از :

- byt برای tinyint (by مخفف byte است)

- int برای Smallint (عدد صحیح استاندارد دو بایتی)

- lng برای int (چون اعداد صحیح چهار بایتی در زبانهای برنامه سازی

long نامیده می شود از پیشوند lng استفاده شده است)

- big برای bigint

- توجه داشته باشید پیشوند گذاری اجباری نیست. هر چند بنظر می

رسد کار بیشتری را به SQL Server تحمیل می کند اما در عمل

اینطور نیست زیرا اسامی SQL Server ستونها را با tokenهایی با

فرمت داخلی و بهینه شده جایگزین می کند.

نکته:

انواع دادهای integer از نظر ذخیره سازی، بازیابی و محاسبات ریاضی

کارایی بیشتری نسبت به سایر انواع دادهای دارند. بنابراین در صورت

امکان ترجیحاً از آنها استفاده کنید. SQL Server بطور پیش گزیده

حداکثر ۲۸ رقم را به عنوان دقت (Precision) در نظر می گیرد تا تطابق

با ویژگیهای بیسیک و پاور بیلدر میکروسافت تضمین شود. اگر SQL

Server را با گزینه p/ آغاز کنید، دقت به ۳۸ رقم افزایش می یابد.



## اعداد دقیق و تقریبی

واژه جدید در SQL Server هر دو نوع داده دقیق (دسیمال و عددی) و تقریبی (حقیقی real اعشاری Float) وجود دارند. در تعریف اعداد تقریبی، دقت حداکثر تعداد ارقام مجاز در دو قسمت نقطه اعشار است. در تعریف اعداد دقیق، مقیاسی (Scale) نیز تعریف می شود. مقیاس تعداد ارقام مجاز سمت راست نقطه اعشار را تعیین می کند.

## اعداد تقریبی

دو نوع داده اعداد تقریبی، اعداد حقیقی (real) و اعشاری (float) می باشند. اعداد ذخیره شده در این دو نوع داده ای از دو بخش مانتیسی (mantissa) و نما (exponent) تشکیل شده اند.

الگوریتم تولید این دو بخش کاملاً دقیق نیست. به عبارت دیگر آنچه را که وارد می کنید بدست نمی آورید. این مشکل تنها زمانی بروز می کند که دقت عدد ذخیره شده به دقت تعیین شده برای نوع داده ای نزدیک شود. برای مثال دقت مجاز برای اعداد ۳۸ رقم می باشد. اعداد حقیقی و اعشاری در داده های علمی و آماری که دقت مطلق مورد نیاز نیست و دامنه اعداد از بسیار کوچک تا بسیار بزرگ متغیر است مناسب می باشند.

اعداد حقیقی ۷ رقم دقت و چهار بایت فضای ذخیره سازی دارند. اگر داده‌ای از نوع اعشاری با دقت کمتر از ۷ رقم تعریف کنید در اصل نوع عدد حقیقی به شمار می آید. دقت اعداد اعشاری از ۱ تا ۳۸ رقم است و پیش گزیده آن ۱۵ می باشد. کلیه محاسبات بر روی اعداد اعشاری بجز باقیمانده تقسیم صحیح (mod) مجاز است. فرض کنید نوع داده زیر تعریف شده است:

SngSomeval real

اعداد 188,445.2 یا 1,884.452 قابل ذخیره هستند. اما اعداد 188,445.27 یا 1,8844527 مجاز نیستند زیرا از دقت ۷ رقم بیشتر می‌باشند. برای نگهداری مقادیر بزرگتر باید متغیر اعشاری با دقت کافی معرفی شود. به دلیل نیاز به ۸ بایت فضای ذخیره سازی، تعریف نوع داده‌ای به شکل float (8) dblSomeVal انجام می‌شود.

توجه:

پیشوند Sng برای اعداد حقیقی بکاررفته است و مخفف Single-Precision می باشد. برای نمایش اعداد اعشاری از پیشوند dbl مخفف double Precision یا دقت مضاعف استفاده می‌شود. این پیشوندها

مطابق با فضای لازم برای آنها می باشد. در ویژوال بیسیک Single نیاز

به ۴ بایت و double نیاز به ۸ بایت دارد.

## اعداد دقیق (Exact Numeric)

دو نوع اعداد دقیق، دسیمال و عددی (numeric) می باشند. دقت در کمترین رقم با معنی (Significant digit) نگهداری می شود. در تعریف این نوع داده لازم است تا دقت و مقیاس هر دو مشخص شوند در غیر اینصورت مقادیر پیش گزیده ۱۸ و ۰ در نظر گرفته می شود.

توجه:

مقیاس 0 معادل تعریف نوع داده صحیح می باشد زیرا نشاندهنده 0 رقم در سمت راست نقطه اعشار است. بکارگیری مقیاس 0 برای داشتن مقادیر عددی بسیار بزرگ و دقیق بزرگتر از  $2 \times 10^6$  میلیون مفید است. خصیصه IDENTITY که در هنگام اضافه شدن هر رکورد جدید، مقدار جدیدی را بطور خود کار تولید می کند نیاز به نوع داده ای صحیح (integer) دارد. اگر تصمیم دارید بیش از ۲ میلیون رکورد داشته باشید نوع داده عددی حقیقی با مقیاس 0 را بکار برید.

ستونی با تعریف decimal(7-2) می تواند اعداد 1000.55 یا 11000.55 را ذخیره کند اما اعداد 11110000.55 یا 1100.5678 مجاز نیستند.

چنانچه عددی بزرگتر از دقت و مقیاس قابل قبول وارد کنید، عدد

مربوطه قطع می شود.

نکته:

پیشوندهای dec و num برای دسیمال و عددی بکار می روند.

توجه:

در حالت کلی از انواع دادهای حقیقی و اعشاری اجتناب کنید و در

صورت امکان اطلاعات خود را در انواع دادهای دسیمال یا عددی ذخیره

کنید زیرا محاسبات اعداد اعشاری و حقیقی برخی اوقات به نتایج جالب

و گیج کننده ای منتهی می شوند.

نوع دادهای خاص (Special)

برخی انواع دادهای در گروه خاصی نمی گنجد بنابراین در بخش انواع

داده خاص آورده شده اند. این دادهها عبارتند از نوع بیت (bit) اشیاء

بزرگ باینری (blob)، متن (text) و تصویر (image) همچنین نوع

دادهای RowGuid ( بخوانید Row\_goo\_id) از این نوع است.

بیت (bit)

نوع داده بیتی از نوع منطقی برای نگهداری اطلاعات Boolean بکار می رود. این انواع داده برای نمایش وضعیت روشن / خاموش، صحیح / غلط و بلی / خیر بکار می رود و به شکل ۱ یا ۰ ذخیره می شود. این ستونها می توانند NULL (نامشخص) باشند، یک بایت فضا اشغال می کنند ولی ایندکس نمی شوند. چنانچه چندین ستون بیتی در یک جدول تعریف کنید، SQL Server بطور خودکار ۸ ستون را در قالب یک بایت گروه بندی می کند.

نکته:

پیشوند f (مخفف flag) برای داده های Boolean بکار می رود در صورت لزوم می توان flg یا bool بکار برد.

متن و تصویر (Text & Image)

برای فضای بیش از ۸۰۰۰ بایت نوع داده های متن یا تصویر را بکار برید. این نوع داده ها BLOB نامیده می شود و قابلیت ذخیره سازی ۲ گیگابایت داده متنی یا باینری را در هر تعریف دارا هستند.

هنگام تعریف داده متن یا تصویر یک نشانگر (Pointer) ۱۶ بایتی به سطر اضافه می شود و به یک صفحه داده ای مجزا ۸ کیلو بایتی اشاره می کند.

اطلاعات اضافی در این صفحه داده‌ای ذخیره می‌شود. اگر داده از ۸ کیلوبایت بیشتر باشد، نشانگرهای دیگر ساخته می‌شوند که به صفحات دیگری از BLOB اشاره می‌کنند.

ذخیره و بازیابی داده‌های متنی و تصویری کارایی پایگاه داده را بشدت کاهش می‌دهد زیرا در عملیات درج، به روز رسانی یا حذف حجم وسیعی از داده در فایل تراکنش‌ها (transaction logs) ثبت می‌شود. برای مقابله با این امر می‌توانید فرمان WRITETEXT را بکار برید. در این حالت تغییرات داده‌ای بدون ورودی متناظر در فایل تراکنش‌ها اعمال می‌شود.

#### هشدار

هنگام استفاده از دستورات بدون ثبت در تراکنش‌ها بلافاصله از پایگاه داده نسخه پشتیبان تهیه کنید زیرا قابلیت ترمیم (recoverability) در اینحالت بسیار اهمیت دارد.

روش دیگر ذخیره سازی داده‌های بزرگ نگهداری آنها در فایل‌های جداگانه و ذخیره مسیر دستیابی آنها در پایگاه داده است.

نکته:

برای پیشوند داده های `image` و `text` از `img` و `str` استفاده کنید. البته می توانید `text` را برای داده `text` بکار برید اما این پیشوند معمولاً در

زبانهای برنامه سازی برای جعبه متن (`textbox`) بکار می رود.

جدول:

این نوع داده عموماً برای ذخیره نتایج پرس و جو برای استفاده آتی

کاربرد دارد و بکارگیری آن مشابه ایجاد یک جدول موقت است. قدرت

واقعی این نوع داده آن است که هم می تواند مانند یک جدول معمولی

و هم مانند یک متغیر محلی در توابع، عملیات دسته ای (`Batch`) یا روال

های ذخیره شده بکار رود.



## انواع داده تاریخ و ساعت (Data and Time)

تاریخ و ساعت در نوع داده `datetime` یا `Smalldatetime` ذخیره

می شود. توجه داشته باشید تاریخ و ساعت همواره با یکدیگر و به شکل

یک مقدار واحد ذخیره می شوند و دارای فرمتهای متفاوتی می باشد.

توجه:

برای نمایش ماه می توانید مقدار ترتیبی (`Ordinal`) تعیین کنید. مقدار

ترتیبی یک عنصر محل قرارگیری آن در لیست عناصر است.

تاریخ های ذخیره شده در `datetime` با دقت میلی ثانیه و در ۸ بایت

ذخیره می شوند. ۴ بایت برای عدد متناظر با روز نسبت به تاریخ اول

ژانویه ۱۹۰۰ و ۴ بایت برای ثانیه های سپری شده از نیمه شب. (تاریخ

های ما قبل تاریخ فوق به شکل اعداد منفی در دامنه 12/31/9999 تا

1/1/1753 ذخیره می شوند. دقت این تاریخ ها 3.33 میلی ثانیه است.

نکته

برای ذخیره تاریخ ماه، روز یا سال `datetime` بکار نبرید. اگر صرفاً به

سال احتیاج دارید نوع `Smallint` یا `tinyint` موثرتر است. اگر تاریخ را با

این روش ذخیره کنید در هر بار عملیات درج، بروز رسانی و یا سایر کارها با این اطلاعات باید آن را تجزیه (parse) کنید.

نوع داده Smalldatetime ۴ بایت بکار می برد و تاریخ ذخیره شده آن

دقت دقیقه دارد. از نظر داخلی یک Smallint (دو بایت) برای تعداد

روزهای سپری شده از تاریخ اول ژانویه 1900 و Smallint دیگر برای

نگهداری ثانیه‌های سپری شده از نیمه شب بکار می رود و دامنه آن از

1/1/1990 تا 6/6/2076 است.

نکته:

برای تاریخ های جاری در پایگاه داده نوع Smalldatetime بکار برید،

خصوصاً آنهایی که ماهیت زودگذر دارند. برای مثال می‌توانید تاریخ

هایی که نمی‌خواهید بیش از چند سال بکار برید را با این نوع داده

تعریف کنید.

نکته:

پیشوند db برای datetime و Sdt برای Smalldatetime بکار می‌رود.

## پول

دو نوع داده پولی، money و Smallmoney می باشد. هر دو مقیاس ۴ دارند بنابراین ۴ رقم در سمت راست نقطه اعشار ذخیره می کنند. علاوه بر مقادیر دلاری سایر اطلاعات نیز برای کاربرد بین المللی قابل ذخیره است اما توابع تبدیل ارزی در SQL Server وجود ندارند. هنگام ورود داده‌های پولی لازم است علامت \$ در ابتدا وارد شود.

## توجه

پیشوند Cur برای نوع داده پولی بکار می رود. Cur مخفف Currency و معادل نوع داده money در ویژوال بیسیک است.

## کار با ANSI و Unicode

جهت اطمینان از تناظر انواع داده‌های SQL Server و ANSI می‌توانید

از انواع داده ANSI استفاده کنید.

داده Unicode مجموعه کاراکتر چندبایتی -Unicode Ucs

(Multibyte) 2 را بکار می‌برد. هنگام کار با کاراکترهای طبیعی ANSI

برای ذخیره هر کاراکتر یک بایت داده لازم است بنابراین به آن

Narrow نیز گفته می‌شود. در مقابل Unicode یک مجموعه کاراکتر چند

بایتی یا Wide است. مجموعه کاراکتر UCS-2 برای نمایش هر کاراکتر

دو بایت بکار می‌برد که در پایگاه داده‌های چندزبانه بسیار مفید است.

برای مثال زبانهای انگلیسی یا اسپانیایی در یک مجموعه کاراکتر تک

بایتی قرار می‌گیرند. در حالیکه زبانهای مانند ژاپنی حتی نوع

استاندارد آن (kana) بیش از ۱۰۰۰ کاراکتر دارد. یک بایت استاندارد ۸

بیتی تنها می‌تواند ۲۵۶ کاراکتر را نشان دهد. در حالیکه کاراکتر

Unicode دو بایتی 65,536 کاراکتر را نشان می‌دهد.

برای نمایش اطلاعات Unicode می‌توان انواع داده nchar , nvarchar

یا ntext بکار برد. دو نوع اول محدودیت ۸۰۰۰ بایت یا ۴۰۰۰ کاراکتر

دارند. برای مثال (4000) nchar مجاز است اما (6000) nchar نمایانگر ۱۲۰۰۰ کاراکتر است که در یک صفحه داده‌ای قرار نمی‌گیرد. پس لازم است یا به دو نوع داده nchar شکسته شود. و یا به ntext تبدیل شود. این نوع داده ۲/۱۴ گیگا بایت داده را پشتیبانی می‌کند.

نوع داده unicode یکی از خصوصیات بسیار قدرتمند SQL Server 2000 به شمار می‌آید. چنانچه می‌خواهید SQL Server خود را در کشورهای Us English بکار نمی‌برند استفاده کنید، بهترین راه حل ذخیره سازی داده‌ها در نوع داده Unicode است.

ایجاد انواع داده تعریف شده کاربر (User\_Defined Data Types)

برای یک پایگاه داده نمونه یا خاص می‌توانید انواع داده جدیدی تعریف کنید. به یاد داشته باشید یک پایگاه داده نمونه، قالب ایجاد پایگاه های داده جدید است بنابراین انواع داده تعریف شده کاربر در سایر پایگاه داده های بعدی در دسترس می باشد.

برای ایجاد انواع داده تعریف شده کاربر لازم است تا از انواع داده فراهم شده سیستم استفاده کنید. برای مثال می‌توانید نوع داده جدید به

نام ID Employee ایجاد و آن را به عنوان کاراکتر یا عدد معرفی

نمایید اما نمی توانید آن را از نوع ناموجودی مثل Coloumn\_id

تعریف کنید.

تعریف انواع داده‌ای باید قبل از اضافه کردن آنها به جدول انجام شود.

برای اینکار می‌توانید SQL Server Enterprise Manager یا روال

ذخیره شده سیستم به نام sp\_addtype را بکار برید.

پس از معرفی نوع داده می‌توانید آن را بدفعات دلخواه بکار برید

برای افزودن انواع داده تعریف شده کاربر از طریق Enterprise

Manager مراحل زیر را دنبال کنید.

۱- Enterprise Manager را باز کنید.

۲- پایگاه داده مورد را بیابید.

۳- به پوشه Defined Data Type بروید

۴- روی پوشه کلیک راست کنید و New User Defined Data Type

را انتخاب کنید.

۵- نام، نوع داده، طول و سایر قواعد و مقادیر پیش‌گزیده مربوطه را

تکمیل کنید.

۶- OK را کلیک کنید.

برای حذف نوع داده تعریف شده کاربر میتوان Enterprise Manager یا

روال ذخیره شده سیستم به نام Sp\_droptype را بکار برد. در

Enterprise Manager مراحل زیر را دنبال کنید.

۱- پایگاه داده را بسط داده و پوشه User Defined Type را برجسته

کنید.

۲- در پانل راست بر روی نوع داده مورد نظر کلیک راست کنید و سپس

Delete را کلیک کنید.

کاربرد دستور CREATE TABLE

این دستور برای ایجاد جداول جدید در SQL Server 2000 کاربرد

دارد و می تواند با کدنویسی یا Enterprise Manager انجام شود. در

ابتدا با کاربرد کدنویسی جهت تولید جداول آشنا می شوید، سپس روش

دوم را بررسی می کنیم. نگارش (Syntax) دستور CREATE TABLE

به شرح زیر است.

CREATE TABLE [database. [owner] table\_name

(

Column \_ name datatype [identity] Constraint|Null|Not  
Null|Collate]  
[...]  
)

در این نگارش:

**table\_name** نام جدول جدید مطابق با قوانین نامگذاری شناسه‌هاست و باید در پایگاه داده برای مالک (owner) آن یکتا باشد. بدین معنا که اگر دو کاربر مجوز ایجاد جدول در یک پایگاه داده را دارا باشند می‌توانند جدولی با اسامی یکسان داشت زیرا بواسطه نام مالک خود یکتا و منحصر بفرد تلقی می‌شوند و نام مالک بخشی از نام جدول است.

**Column\_name** نام ستون است و از قوانین نامگذاری شناسه‌ها تبعیت می‌کند.

**Datatype** نوع داده ستون است.

بخش آخر اختیاری است در صورت نیاز می‌توان قابلیت ذخیره سازی مقدار تهی، قیود و ... را تعریف نمود.

**Collate** نیز اختیاری است. این خصیصه به شما امکان می‌دهد تا ترتیب

**Collation** این ستون را نادیده بگیرید.



## ایجاد جداول با SQL Server Enterprise Manager

مراحل ایجاد جداول به شکل گرافیکی از طریق Enterprise Manager

عبارتست از:

۱- به SQL Server متصل (connect) شوید و پوشه پایگاه‌های داده را

بسط دهید. سپس پایگاه داده مورد نظر را بسط داده و پوشه Tables را

کلیک کنید.

۲- در سمت راست کلیک راست کنید New Table را انتخاب کنید.

۳- اطلاعات Allow Null, Length, Data Type, Column Name را

اضافه کنید. خواص ستونها نظیر شرح یا مقادیر پیش‌گزیده را تنظیم

کنید. برای ایجاد متغیر identity خصیصه IDENTITY در برگره

Columns را Yes کرده و سپس مقدار اولیه (Seed) و گام افزایش

(increment) آن را انتخاب کنید.

۴- پس از خاتمه، آیکن Save را برای ذخیره جدول کلیک کنید، نام

مناسب را تخصیص داده و دیالوگ را ببندید.

حذف جداول (Dropping Tables)

برای حذف جداول در Enterprise Manager بر روی نام جدول کلیک

راست کنید و سپس از منوی میانبر (Contextmenu) گزینه Delete را

انتخاب کنید. برای حذف جدول tblEmployee با TransaCT\_SQL

دستور زیر را اجرا کنید:

Drop Table tblemployee

توجه:

جداول سیستم قابل حذف نیستند.

برای حذف جدول در Enterprise Manager مراحل زیر را دنبال کنید:

۱- در پوشه Tables نام جدول را انتخاب و کلیک راست کنید.

۲- از منوی میانبر delete را انتخاب کنید.

۳- برای مشاهده وابستگی (dependencies) جدول، دکمه Show

Dependencies را کلیک کنید. پس از آنکه جدول دیگر در هیچ ارتباط

کلید اصلی / کلید خارجی شرکت نداشت می توانید دکمه Show

Dependencies را کلیک کنید. پس از آنکه جدول دیگر در هیچ ارتباط

کلید اصلی / کلید خارجی شرکت نداشت می توانید دکمه Drop All

را کلیک کنید تا حذف شود.

۲-۳-۵- بازیابی داده ها با دستور SELECT

نوشتن دستورات ساده SELECT

دستور Select سه جزء اصلی SELECT، FROM و WHERE دارد.

نگارش پایه آن عبارت است از:

SELECT column – list

FROM table- list

WHERE search – criteria

بخش SELECT ستونهای مورد نظر برای بازیابی را مشخص می کند.

عبارت FROM نام جدولی است که ستونها از آن بازیابی می شوند و

عبارت WHERE تعداد سطرهای بازگشتی از پرس و جو را مشخص می

کند.

SELECT [ All | DISTINGCT ] [ Top n [PENSEVT ] [ WITH  
TIEST ]]

Select – list

[ INTO new – table ]

[ FROME table – sources]

[ HERE search – condition ]

[ GROUP – BY [ ALL] group – by – expression [...n]

[ WITH { CUBE | ROLLUP } ] ]

[ HAVING search – condition ]

[ ORDER BY { colum – name [ ASC | DESC ] } [ ,...n ]

[ COMPUTE { { AVG | COUNT | MAX | MIN | SUM }  
(expression) }

[ ,...n ] [ BY expression [ ,...n ] ]

[ FOR BROWSE ] [ OPTION (query – hint [ ,...n ]

**دستور SELECT \* FROM table – name ابتدایی ترین نوع پرس و**

**جواست. بکارگیری علامت (\*) به عنوان colum – list کلیه ستونها**

**جدول را بازیابی می کند در داخل SQL Server این علامت به فهرست**

**ستونهای جدول تبدیل می شود.**

**برای بازیابی ستونهای مشخص لازم است ستونها با ( ) از یکدیگر جدا**

**شوند اما پس از آخرین ستون نیازی به کاما نیست.**

**هنگام اجرای پرس و جوی \* SELECT ترتیب ستونها مشابه ترتیب آنها**

**در دستور TABLE CREATE است اما وقتی ستونهای خاصی از جدول**

**را انتخاب می کنید این ترتیب نباید مشابه ترتیب ایجاد آن باشد. شما می**

توانید با تغییر ترتیب ستونها در column – list ترتیب آنها در خروجی

پرس و جو را تغییر دهید.

**تغییر عنوان ستونها**

هنگام نمایش نتایج پرس و جو، عنوان ستونها مشابه اسامی آنها در

column – list است اما می توانید جای آنها اسامی خواناتری بکار برد.

از طریق بکارگیری اسامی مستعار در عنوان ستونها انجام می شود. برای

ایجاد اسامی مستعار در عناوین ستونها هر دو نگارش ، SQL Server

ANSI SQL می روند. دو روش برای این کار وجود دارد.

SELECT colum – name ‘column – heading ‘

FROM table – name

**توجه:**

اگر نام مستعار بکار رفته دارای فاصله است و یا یکی از کلمات کلیدی

SQL server است آن را مابین علائم “ [] قرار دهید. به مثال زیر

**توجه کنید.:**

Select Iname As ‘Last Name ‘ , fname As [First Name]

FROM employee

مثال زیر یک کلمه کلیدی SQL را به کار می برد:

```
SELECT 'Count' = Count (*)
```

```
FROM employee
```

با دستورات زیر می توانید پرس و جوی خود را بازنویسی کنید:

نگارش SQL Server 2000

```
SELECT Employee ID= emp - id, Last Name = Lnme,
```

```
FirsName = Fnma
```

## نگارش ANSI SQL

```
SELECT emp – id As Employeid , Lname As Last Name,  
fname As FistName
```

توجه:

کلمه کلیدی As ضروری نیست. برای مثال دستور زیر اطلاعات مشابه

پرس و جو قبل را باز می گرداند:

```
SELECT emp – id Employeid, Lname LasName, fame  
firstName
```

## بکار گیری حروف (Literals)

برای خوانایی بیشتر خروجی پرس و جو می توانید از لیترالها استفاده

کنید. یک لیترال رشته ای است مالین علائم ‘ یا ‘ ‘ محصور شده

در -list column قرار می گیرد و در خروجی پرس و جو نمایش داده

می شود. در نتیجه یک لیبل در ستون مجاور ستون نتایج نمایش داده

میشود.

نگارش بکار گیری لیترال ها به شکل زیر است:

```
SELECT ‘literal’ [, ‘literal’ ....]
```

۲-۳-۶ پردازش داده ها (Manipulating Data)

شما می توانید با پردازش نتایج پرس و جوها، ستونهای جدیدی را برای نمایش مقادیر محاسباتی، مقادیر رشته ای جدید، تبدیلات تاریخ و ... تولید کنید. اینکار با عملگرهای (Operators) حسابی، توابع ریاضی. توابع رشته ای، توابع تاریخ و توابع سیستم انجام می شود. همچنین می توانید تابع CONVERT را در تبدیل یک نوع داده به نوع دیگری برای پردازش ساده تر بکار برید.

### عملگرهای حسابی (Arithmetic Operators)

عملگرهای حسابی بر روی انواع داده , real , money , bigint , float , decimal , numeric , tintnt , smalint , int یا smallmoney

بکار می رود.

### ترتیب تقدیم عملگرها (Operator Precedence)

واژه جدید: در عملیات حسابی دو سطح تقدیم در ارتباط با تقدم نوع داده و تقدم عملگر وجود دارد.

تقدم نوع داده زمانی به کار می رود که عملیات حسابی بر روی انواع داده متفاوت انجام می شود. در این حالت نوع داده کوچکتر به نوع



داده بزرگتر تبدیل می شود. برای مثال اگر یک Smalint را در یک int

ضرب کنید int خواهد بود. تنها استثناء موجود در مورد نوع money

است که نتیجه همواره از نوع money خواهد بود.

تقدیم عملگر زمانی در صورت وجود چندین عملگر به کار می رود. این

ترتیب هنگام خواندن از چپ به راست به ترتیب مدول، ضرب و تقسیم و

بالاخره جمع و تفریق می باشد.

مشابه سایر عملیات حسابی می توان ترتیب تقدم عملگر را با پرانتز تغییر

داد. در این صورت داخلی ترین عبارت ابتدا محاسبه می شوند. همچنین

بکارگیری پرانتز خوانایی عبارات محاسباتی را بیشتر می کند. برای مثال

در:

$5+5*5=30$  ابتدا ضرب انجام می شود، اما در  $5+5*(5+5)=30$  عبارت

داخل پرانتز ابتدا محاسبه می شود عموماً با قرار دادن عملیات حسابی

در پرانتز، خوانایی و وضوح کد Transact – SQL را بهبود می بخشد.

توابع ریاضی (Mathematical Functions)

توابع داده ریاضی امکان عملیات رایج بر روی داده های ریاضی را

فراهم می آورند. با نگارش زیر می توانید داده های ریاضی را

بازگردانید:

SELECT function – name (parameters)

## جدول ۱۰-۲ توابع ریاضی

تابع	نتیجه
ABS (numeric – expr)	مقدار مطلق
ACOS ASIN  ATAN(float – expr)	سینوس، کسینوس و تانژانت زاویه به رادیان با مقدار شناور
COS SIN COT	سینوس، کسینوس و تانژانت زاویه به رادیان
TAN(float – expr)	بزرگترین عدد صحیح کوچکتر یا مساوی مشخص شده
FLOOR (numeric – expr)	لگاریتم طبیعی
LOG(float – expr)	لگاریتم بر مبنای ۱۰
LOG10(float – expr)	عدد ثابت ۳/۱۴۱۵۹۲۶۵۳۵۸۹۹۳
PI()	مقدار پارامتر به توان y
POWER (numeric – expr, y)	

RADIANS (numeric –  
تبدیل درجه به رادیان  
expr)

ROUND(numeric – expr)  
گرد کردن عدد به طول مشخص شده  
len

تابع	نتیجه
------	-------

SQURE((float – expr)  
مربع مقدار مشخص شده

SQRT (float – expr)  
جذر مقدار مشخص شده

نکته:

هنگام استفاده از توابع ریاضی با انواع داده پولی، همواره قبل از داده  
ها علامت دلار (\$) قرار دهید در غیر این صورت مقدر مشخص شده نوع

داده numeric مبنای ۴ در نظر گرفته می شود.

توجه:

هنگام کار با نوع داده float بعضاً نتایج غیر قابل انتظاری از Query

Analyzer یا سایر ابزار به دست می آید. برای مثال دستور

SELECT ROUND (12.345678E+5,2)  
برابر

1234567.889999999 Open (ODBC) کار این نتیجه خواهد بود.

database connectivity است. SQL Server کماکان گرد کردن را

ادامه می دهد. اما بنا به تعریف داده های نوع float مبهم هستند، در

حالت کلی بهتر است از این داده اجتناب کنید.

توابع رشته ای ((String Functions))

هنگام کار با اطلاعات کاراکتری، توابع رشته ای زیادی برای پردازش

وجود دارد. اکثر توابع رشته ای بر داده های نوع , nchar , varchar

char و nvarchar کار می کنند بنابراین لازم است سایر انواع داده ای را

قبل از کار تبدیل کرد. داده های کاراکتری با تابع زیر بازگردانده می

شوند:

SELECT function – name (parameters)

توابع تاریخ

پردازش مقادیر datetime با توابع date انجام می شود. این توابع در

Column – list عبارت WHERE یا هر عبارت دیگر قابل استفاده اند.

برای این منظور، نگارش زیر را به کار برید:

SELECT date – function (parameters)

مقادیر پارامتر datetime را بین علائم ، ، یا “ “ قرار دهید. برخی توابع پارامتری به عنوان derepart دریافت می کنند.

تابع	نتیجه
Day(date)	مقدار عددی نمایش دهنده روز را بر می گرداند
Getdate()	تاریخ و ساعت جاری را بر می گرداند
Month (date)	مقدار عددی نمایش دهنده ماه را بر می گرداند
Year(date)	مقدار عددی نمایش دهنده سال را بر می گرداند.

## توابع سیستم (System Functions)

برای دریافت اطلاعات از جداول سیستم، توابع سیستمی از پیش تعریف

شده وجود دارند. برای این منظور بگارش زیر را به کار برید Select

function – name (parameters) این توابع در Colum – list عبارت

WHERE یا سایر عبارت کاربرد دارند.

### جدول ۹-۱۰ توابع سیستم

تابع	نتیجه
COL – NAME (table – id , colum – id)	نام ستون را بر میگردداند
COL – LENGTH ('table – name' Column – name)	طول ستون را بر می گرداند
DB – ID ([ 'database – name'])	شناسه پایگاه داده را بر می گرداند
DB – NAME ( [database – id])	نام پایگاه داده را بر می گرداند

[www.kandoo.cn.com](http://www.kandoo.cn.com)

[www.kandoo.cn.com](http://www.kandoo.cn.com)

[www.kandoo.cn.com](http://www.kandoo.cn.com)

[www.kandoo.cn.com](http://www.kandoo.cn.com)

[www.kandoo.cn.com](http://www.kandoo.cn.com)



## عملگر های مقایسه ای (comparison Operators)

عملگر های مقایسه ای در پیاده سازی شرایط جستجو کاربرد دارند.

سطر های خاص با مقایسه مقادیر یک ستون یا عبارت یا مقدار خاص انجام

می شود. این عبارات شامل مقادیر ثابت نام ستونهای دیگر، توابع یا زیر

پرس و جوهای متداخل (nested subqueries) می باشند. اگر دو نوع

داده های متفاوت را با هم مقایسه کنیم (برای مثال char و varchar) و یا

نوع داده تاریخ را مقایسه می کنید.

(datetime , samlldatetime) ، حتماً آنها را مابین علائم ' ' قرار دهید.

علائم " " نیز قابل قبول است اما نوع اول با استاندارد ANSI تطابق

دارد.

نگارش عبارت WHERE با عملگر های مقایسه ای به شرح زیر است:

SELECT Column – list

FROM table – list

WHERE Column – name Comparison – operator expression

پرس و جوی زیر شماره کارمندی، نام و نام خانوادگی کلیه کارمندان

استخدام شده توسط ناشر کد 0877 را بر می گرداند.

```
SELECT emp – id, Iname , fame  
FROM employee  
WHERE pub – id = ‘877’
```

نکته:

اگر عملگرهای حسابی و منطقی را با هم به کار می برید توجه کنید

عملگرهای حسابی نخست پردازش می شوند اما با استفاده از پرانتز می

توانید ترتیب تقدم را تغییر دهید.

دامنه ها (Ranges)

با کاربرد کلمه کلیدی BETWEEN می توانید سطرهایی را بر اساس

مقادیر یک دامنه بازیابی کنید. مشابه عملگرهای مقایسه ای لازم است تا

داده های نوع کاراکتری یا تاریخ را مابین علائم ' ' قرار دهید.

نگارش عبارت WHERE در این حالت به شرح زیر است:

SELECT Column – list

From table – list

WHERE Column – name [Not] BETWEEN expression And

expression

پرس و جوی زیر مقادیر شماره کارمندی و نام خانوادگی کلیه کارمندان

استخدام شده بین 10/1/92 تا 12/31/92 را بازیابی می کند:

SELECT Iname, emp – id

FROM employed

WHERE fire – date BETWEEN ‘10/1/92 AND ‘12/31/92’

در این مثال دو مقدار مشخص شده نیز قابل پذیرش هستند. توجه کنید

مقدار کوچکتر باید در ابتدا آورده شود.

لیست ها (List)

با کلمه کلیدی IN می توانید سطرهای مطابق با مقادیر یک لیست را

بازیابی کنید. مطابق معمول داده های نوع کاراکتری و تاریخ را مابین

علائم ' ' قرار دهید. نگارش عبارت WHERE در این حالت به شرح زیر

است:

SE:LECT Column – list

FROM table – list

WHERE Column – name [ Not ]IN (valu – list)

برای کارمندانی که با ناشر کد 0877 یا 9999 کار می کنند پرس و جوی

زیر را اجرا کنید:

ورودی

SELECT emp – id . Iname , fname

FROM employee

WHERE pub – id IN ('0877', '9999')

برای بازیابی سطرهایی که در لیست نیستند می توان عملگر نفی

(NOT) بکار برید. برای یافتن کلیه کارمندانی که با ناشر کد 0877 یا

9999 کار نمی کنند، پرس و جوی زیر را اجرا کنید:

SELECT emp – id, name, faname

FROM employee

WHERE pub – id NOT IN ('0877', '9999')

نکته:

در صورت امکان شرطهای مثبت بکار برید و از عملگر NOT اجتناب

کنید. زیرا بهینه ساز پرس و جوها همواره شرایط جستجوی منفی را

تشخیص نمی دهد. به عبارت دیگر SQL Server باید کار زیادی برای

بازگرداندن نتایج با عملگرهای نفی را انجام دهد.

رشته های کاراکتری (Character String)

با کلمه کلیدی LIKE می توانید سطرها را بر اساس بخشی از رشته

های کاراکتری بازیابی کنید. LIKE با انواع داده , text , datetime

char , varchar , nchar , nvarchar و Smalldatetime به کار می

رود. همچنین می توانید از چهار نوع کاراکتر جانشین (Wildcaed) نیز

استفاده کنید.

نگارش عبارت Where با کلمه کلیدی Like به شرح زیر است

SELECT Column – list

FROM table – list

WHERE Column – name [NOT] LIKE ‘string’

عملگرهای جانشین در دسترس عبارتند از:

% رشته ای از هیچ یا چند کاراکتر

- یک کاراکتر

[] یک کاراکتر در محدوده خاص

[^] یک کاراکتر خارج از محدوده خاص

هنگام کار حتماً توجه کنید عملگرهای جانشین مابین علائم ' ' قرار داده

شوند.

با اجرای پرس و جوی زیر title و id - title کلیه کتبی که عنوان آنها

شامل Computer است از جدول titles بازیابی می شود.

```
SELECT title - id , title
```

```
FROM titles
```

```
WHERE title LIKE '%computer %'
```

مقادیر نامشخص (Unknown Values)

مقدار تهی یا Null مشابه رشته حرفی خالی (Blank) و یا صفر در داده

های عددی نیست. Null زمانی واقع می شود که فیلد مقدار داده نشود.

به عبارت دیگر Null معادل نامشخص می باشد. در این شرایط مقایسه

Null با صفر، Blank و یا حتی سایر مقادیر Null منفی باشد. بنابراین

چگونه میتوان سطرها را بر اساس مقادیر Null بازیابی کرد؟ اینکار با

تفکیک سطرهایی از جدول که حاوی مقادیر Null هستند و از طریق

کلمات کلیدی IS NULL و IS NOT NULL انجام می شود.

نگارش عبارات WHERE با استفاده از عملگرهای IS NULL و IS

NOT NULL به شرح زیر است:

```
SELECT Column – list
```

```
FROM table – list
```

```
WHERE column – name Is [NOT] NULL
```

برای مثال پرس و جوی زیر را بازیابی کتبی که فروش نداشته اند اجرا

**کنید:**

```
SELECT title – id
```

```
FROM title
```

```
WHERE ytd- sales IS NULL
```

برعکس پرس و جوی فوق، می توانید کلید کتبی که فروش داشته اند با

**پرس و جوی زیر بازیابی کنید:**

```
SELECT title – id
```



FROM title

WHERE ytd – sales ISNOT NULL

### بکارگیری شرایط چند گانه در بازیابی سطرها

تا کنون بازیابی سطرها از طریق مقادیر مشخص، لیستها، دامنه، مقایسه

رشته ای و ... را دیدید. حال بهتر است نگاهی به بازیابی سطرها از

طریق شرایط چندگانه داشته باشید. برای ایجاد شرایط چندگانه می

توانید عملگرهای منطقی (logical)، نفی (NOT) و (AND) و یا (OR)

را به کار برید. AND و OR امکان تلفیق دو یا چند عبارت را فراهم می

آورند. شرط AND در صورت کلیه شرایط مقدار بر می گرداند در

حالیکه OR در صورت صحت هر یک از شرایط مقدار بر می گرداند.

در استفاده از چندین عملگر منطقی ترتیب تقدم NOT و AND و OR

است. نگارش عبارت WHERE در شرایط چندگانه به شرح زیر است:

SELECT column – list

FROM table – list

WHERE [NOT] expression {AND|OR} [NOT] expression

حذف اطلاعات تکراری (Duplicate)

هنگام انتخاب برخی از اطلاعات ممکن است با سطرهای تکراری مواجه شوید. برای حذف آنها عبارت DISTINCT در دستور SELECT را به

کار برید. در غیر این صورت کلیه شرایطی که شرط عبارت WHERE را برقرار می کند در نتیجه نمایش داد می شوند.

نگارش عبارت DISTINCT به شرح زیر است:

SELECT DISTINCT column – list

FROM table – name

WHERE Search – condition

DISTINCT بر روی ترکیب ستون ها در column – list اعمال می

شود. از آنجاییکه مقادیر null به عنوان تکراری در نظر گرفته می شود

بنابراین فقط یکی از آنها برگردانده می شود.

۲-۳-۷ مرتب سازی داده ها با عبارت ORDER BY

برای مرتب سازی داده ها در دستور SELECT عبارت ORDER BY

بکار برید. نگارش پایه این عبارت به شرح زیر است:

SELECT column –list

FROM table –list

[ORDER By Column – name | column – list – number [ASC |  
DESC]

تعداد ستونهای لیست ORDER BY نامحدود است به شرط آنکه از ۹۰۰

بایت بیشتر نشود. همچنین می توانید نام ستونها یا عدد ترتیبی (ordinal)

آنها را در column – list بکار برید.

بکارگیری توأم نام و شماره ترتیبی نیز مجاز است. همچنین نوع مرتب

سازی صعودی یا نزولی داده ها نیز قابل تعیین است. مقدار ASC

صعودی و DESC نزولی و پیش گزیده صعودی است. اگر نوع مرتب

سازی ASC باشد و ستون مشخص شده حاوی مقدار NULL باشد،

سطرهای حاوی NULL ابتدای لیست قرار می گیرند.

هنگام کار با عبارت ORDER BY، مرتب سازی SQLServer ممکن

است در نتایج تاثیر بگذارد ترتیب پیش گزیده در SQLServer ترتیب

دیکشنری است و نسبت به حروف حساس نیست. ترتیب حساس به حروف

نتایج بازگشتی را تغییر می دهد زیرا در این حالت حروف بزرگ و

کوچک معادل نیستند.

توجه:

ORDER BY با نوع داده text و image بکار نمی رود. فیلدهای نوع

text در محل متفاوتی ذخیره نمی شوند و طولشان از ۰ تا ۲۰ گیگا بایت

است و SQL Server اجازه مرتب سازی این نوع فیلد را نمی دهد. نوع

داده image نیز قابل مرتب سازی نمی باشد. در صورت لزوم میتوانید

یک جدول مرتبط حاوی اطلاعات خلاصه شده را از داده های متنی و

تصویری ایجاد کنید.

۲-۳-۸- دستور SELECT INTO

این دستور امکان می دهد جدول جدیدی را بر اساس نتایج یک پرس و

جو ایجاد کنید. این جدول جدید بر اساس ستونهای مشخص شده در

لیست select جداول نامبرده در عبارت FROM و سطرهای منتخب

عبارت WHERE ایجاد می شود. با دستور Select into دو نوع جدول

دائم و موقت می توان ساخت. نگارش این دستور به شرح زیر است:

SELECT column – list

INTO new – table – list

FROM table – list

WHERE search – criteria

SELECT INTO/ در ایجاد جدول دائمی لازم است گزینه

SELECT INTO BULKCOPY را تنظیم کنید. با به کارگیری دستور

می توان بدون فرایند (process) معمول تعریف داده، جدول جدیدی را

ایجاد و داده هایی را که در آن قرار داد. نام جدید تابع قوانین نام

گذاری SQL Server است و باید در سطح پایگاه داده منحصر بفرد و

یکتا باشد. اگر ستون های column – list دستور SELECT بدون نام

باشند (ستونهای حاصل از توابع جمعی) آنگاه ستونهای جدول جدید نیز

بدون نام خواهند بود. این امر دو مشکل بوجود می آورد:

- اسامی ستونها باید منحصر به فرد باشد، بنابراین اگر دو ستون بدون

نامی وجود داشته باشد دستور SELECT INTO به شکست می

انجامد.

- اگر جدول جدید حاوی ستون بدون نام باشد آنگاه تنها روش بازیابی

این ستون بکارگیری دستور Select است.

نظر به دلایل فوق بهتر است اسامی مستعار برای ستونهای مشتق شده (derived) ایجاد شود.

همچنین چون بکارگیری دستور SELECT INTO در فایل تراکنشها ثبت نمی شود لازم است تا بلافاصله از پایگاه داده نسخه پشتیبان تهیه شود. همچنین می توانید این دستور را در ایجاد جداول موقت به کار برید. دو نوع جدول موقت عبارت است از:

- جدول موقت محلی (local) که تنها در نشست (session) جاری کاربرد با SQL Server در دسترس است و فضای تخصیصی به آن بلافاصله پس از خاتمه نشست آزاد می شود. برای ایجاد جدول موقت محلی قبل از نام جدول علامت # قرار دهید.

- جدول موقت عمومی (global) که در کلیه نشستهای کاربران با SQL Server در دسترس است و فضای تخصیصی به آن پس از خاتمه نشست آخرین کاربر با جداول آزاد می شود. برای ایجاد این نوع جدول قبل از آن علامت ## قرار دهید.

جدول موقت در پایگاه داده tempdb قرار می گیرند. مثال زیر جدول

موقتی را با نام #tmptitles و با سه فیلد ایجاد می کنند:

```
SELECT title – id, title, price
```

```
INTO # tmp Titles
```

```
From title
```

```
GO
```

```
SELECT * FROM #tmp Titles
```

```
GO
```

**نکته:**

دستور SELECT INTO جدول جدیدی ایجاد می کند. اگر مایلید به

یک جدول موجود سطرهایی اضافه کنید دستورات INSERT یا

INSERT INTO را بکار برید.

**عملگر UNION**

با استفاده از عملگر UNION می توانید نتایج حاصل از دو یا چند پرس

و جو را در یک مجموعه جواب ترکیب کنید. به طور پیش گزیده سطرهای

تکراری حذف می شوند، اما اگر کلمه کلیدی ALL را با این عملگر بکار

ببرید کلیه سطرها حتی تکراری ها نیز بازگردانده می شوند. نگارش این

عملگر به شرح زیر است:

SELECT column – list [INTO column]

[FROM clause]

[Where clause]

[GROUP BY clause]

[HAVING clause]

SELECT column – list

[From clause]

[WHERE clause]

[GROUP BY clause]

{HAVING clause}

[ORDER BY clause]

[COMPUTE clause]

قوانین عملگر UNION عبارتند از:

• در کلیه column – list ها باید تعداد، ترتیب نوع داده ستون ها

مشابه باشند.

• عبارت INTO در صورت وجود باید حتماً در پرس و جوی نخست

باشد.



• عبارات GROUP BY و COMPUTE جهت تعیین ترتیب نتایج

نهایی و یا محاسبه مقادیر خلاصه در انتهای دستور UNION مجاز

هستند.

• اسامی ستونها از list – column نخستین SELECT گرفته می

شود.

## ۲-۳-۹-درج داده (INSERT)

دستور پایه INSERT برای درج یک سطر در جدول تغییر جداول یا

درج داده در جدول پایه از طریق دید (view) به کار می رود. با

تغییراتی در این دستور پایه می‌توانید چندین سطر را با انتخاب داده از

سایر جداول یا دیدها و یا با اجرای روال های ذخیره شده یا توابع

درج کنید. در هر یک از حالت ها لازم است تا از ساختار جداولی که

درج آن انجام می شود اطلاع داشته باشید. آگاهی از اطلاعات زیر

مفید می باشد:

- تعداد ستون های جدول

- نوع هر ستون

- نام ستونها برای برخی از دستور INSERT

- محدودیتها و خواص ستونها نظیر Uniqueidentifier , identity

نگارش دستور INSERT به شرح زیر است:

```
INSERT [INTO]
```

```
{table – or – view}
```

```
{ { [column – list ] }
```

VALUE

({DEFAULT |  
constant – expression } [,...n]) |

select – statement |

execute – statement } |

DEFULT VALUE}

ساده ترین راه برای تعیین تعداد، نام و نوع ستونها، بکارگیری دیالوگ

Table Properties در Enterprise Manager است. برای باز کردن

دیالوگ مراحل زیر را دنبال کنید:

۱- پایگاه داده را از پوشه Databases انتخاب کنید.

۲- جدولی را از پوشه Tables انتخاب کنید.

۳- بر روی نام جدول کلیک راست کنید و Properties را

انتخاب کنید.

با اجرای روال ذخیره شده سیستم help – sp اطلاعات یکسانی به

دست خواهید آورد:

USE pubse

Go

EXEc sp – help publishers

Go

خروجی به دست آمده نام، نوع و محل قرارگیری هر ستون را نشان می دهد. این اطلاعات در ساخت دستورات INSERT کاربرد دارند.

همچنین می توانید با دیدهای INFORMATION SCHEMA اطلاعات مورد نظر را به دست آورید.

## کاربرد دستور INSERT VALLUES

ساده ترین فرم دستور INSERT نیاز به تعیین مقدار هر ستون مطابق

با ترتیب تعریف آنها دارد. این ترتیب در مثالهای قبل نشان داده شده است.

اگر مجوز INSERT جدولی را دارا باشید و هیچ قیدی را نقض نکنید،

SQL Server پیامی مشابه زیر باز می گرداند:

1row (s) affected

توجه:

به دلیل کارکتری بودن نوع ستون ها داده ها ما بین علائم “ قرار گرفته اند. مقادیر عددی نیاز به “ ندارند.

توجه: کنترلهای شبکه مانند (gride – type Controls) که در ابزارهای

بصری توسط کاربران به کار می روند به پایگاه داده بسته نشده اند و

بطور خودکار داده به پایگاه داده نمی افزایند. بنابراین لازم است با

نوشتن کد، داده های تغییر یافته در شبکه را بررسی کنید و دستورات

INSERT / VALUES مناسب برای افزودن سطرها به پایگاه داده

اجرا کنید.

ساده ترین فرم دستور INSERT نیازمند ارائه مقدار ستون با ترتیب

صحیح است. اگر نمی خواهید ترتیب ستونها را رعایت کنید و یا مقدار

آنها را مشخص کنید می توانید نوع دیگر دستور INSERT را به کار

برید.

دستور INSERT زیر لیستی از اسامی ستونها را قبل از عبارت

VALUE مشخص می کند و این لیست تنها شامل زیر مجموعه ای از

ستونهای جدول است، بنابراین لیست مقادیری که پس از آن می آید نیز

تنها برای همین ستونها به کار می رود.

```
INSERT INTO tmpPublishers (sate, pub – id)
```

```
VALUES ('AK' , '9932')
```

نکته:

برنامه های کاربردی و یژوال بسیک یا پاور بیلدر هنگام اتصال به پایگاه داده و افزودن داده ها عملاً دستور INSERT / values را به کار می

برند.

حال برای ستونهایی که در لیست مشخص نشده اند چه اتفاق می افتد؟ هنگام درج یک سطر در جدول، همه ستون ها باید مقدار داشته باشند و اگر شما این مقدار را مشخص نکنید SQL Server باید بتواند این کار را انجام دهد. بنابراین لازم است تا این ستونها یکی از شرایط زیر را دارا باشند:

- ستون دارای مقدار پیش گزیده متصل به آن باشند.
  - ستون از نوع identity باشد.
  - مقدار null برای ستون مجاز باشد.
- دستور INSERT قبل از هیچ مقداری برای ستونها مشخص نشده است. در جدول ستونها مجاز به ذخیره مقادیر null هستند و پس از اجرای دستور INSERT می توانید پرس و جو با دستور select را اجرا کنید.

در صورت اجرای دستور INSERT و مشخص نکردن مقدار برای ستونها بجز موارد استثنا پیام خطا دریافت خواهید کرد.

مقادیر پیش گزیده (DEFAULT VALUE)

یکی دیگر از انواع دستور INSERT حالتی است که می خواهید SQL Server مقادیر پیش گزیده ستونها را در صورت وجود به کار برد. (این مقادیر پیش گزیده شامل null و identify نیز می شوند.)

برای این منظور لیست مقادیر واقعی کلمه کلیدی DEFAULT را به کار برید تا SQL Server بتواند مقدار صحیح را مشخص کند مثال زیر را در نظر بگیرید:

در حالتی که همه ستونها مقدار پیش گزیده قابل استفاده SQL

Server را دارا هستند، دستور INSERT زیر را به کار برید:

```
INSERT INTO [table – name] DEFAULT VALUES
```

۲-۳-۱۰- حذف داده (Delete)

با استفاده از دستور Delete می توانید یک یا چند سطر را از جدول حذف کنید:



DELETE FROM {table – name}

[Where Clause]

**توجه:**

**FROM و WHERE اختیاری می باشند.**

**دستور زیر کلیه سطرهای جدول tmpSales را حذف می کند:**

DELETE tmpSales

**برای حذف زیر مجموعه ای از سطرهای یک جدول عبارت WHERE را**

**به کار برید. شرط WHERE میتواند شامل کلیه شرایط نظیر عملگرهای**

**رابطه ای (= و >، <) کلمات کلیدی IN، LIKE و BETWEEN باشد.**

**مثال زیر کلیه کتابهایی که pub – name آنها "Ne Moon Books"**

**است را از جدول tmppublishers حذف می کند:**

DELETE tmppublishers

WHERE pub – name = 'New Moon Books'

**دستور DELETE و بکار گیری جدول جستجو (Lookup Table)**

**دستور DELETE می تواند سطرهای یک جدول را حذف کند، اما**

**SQL Server این امکان را فراهم می کند که جدول دیگری به عنوان**

**جستجو برای این دستور مشخص شود. این کار باید اغلب با یک زیر پرس**

و جو مشخص شود. در مثال بعد کلیه عناوین منتشر شده توسط New Moon Books حذف می شود. جدول tmpTitles تغییر خواهد کرد اما این جدول صرفاً کد ناشرین (Publisher ID) را نگهداری می کند و نام ناشرین در آن قرار ندارد. بنابراین باید در جدول Publisher جستجو کرده و کد ناشر New Moon Books را پیدا کنید تا سطرهای قابل حذف از جدول tmpTitles مشخص شوند.

```
DELETE FROM tpIPublisherList
```

```
WHERE strPubName=
```

```
(SELECT pub -name FROM tmppublishers
```

```
WHERE pub - id ='9956')
```

زیر پرس و جوی مثال فوق به جدول tmppublishers دسترسی یافته و یک

مقدار مجزا برای pub -name باز می گرداند. سپس این مقدار برای

تعیین سطرهای قابل حذف از publisher - list به کار می رود. برای این

منظور کلیه سطرهای دارای pub -name برابر با مقدار بازگشتی از زیر

پرس و جو تنها یک سطر است زیرا pub - id در جدول publishers کلید

اصلی (Primary key) است. اگر زیر پرس و جو بتواند بیش از یک

مقدار باز گرداند لازم است کلمه کلیدی IN را بجای علامت تساوی (=) به کار برید. زیر پرس و جوی فوق می تواند با دستور تلفیق به نحو مؤثرتری باز نویسی شود. هنگام به کار گیری برای نام جدول حاوی سطرهای حذف شونده و دیگری برای جدول تلفیق شونده. این مثال کلیه سطرهای tmp Sales با نوع کتاب business را حذف می کند.

```
DELETE tmpSales
```

```
WHERE title – id IN
```

```
(SELECT title – id FROM tmpTitles
```

```
WHERE type = ‘business’
```

زیر پرس و جوی فوق لیستی از مقادیر title – id را از جدول tmpTitles

باز می گرداند. این مقادیر برای تعیین سطرهای قابل حذف از جدول

tmpSales با مقدار title – id برابر با یکی از مقادیر فوق به کار می

رود.

با استفاده از SQL – Transact می توانید دستور DELETE جدیدی

بنویسید که در عبارت FROM آنها چندین جدول نام برده شده اند. این

روش دستور DELETE را به شکل JOIN نمایان می سازد گرچه عمل حذف تنها از یک جدول انجام می شود. این قابلیت همانند به کارگیری زیر پرس و جوهاست و جدول دوم تنها به عنوان جدول جستجو به کار می رود.

بکارگیری دستور TRUNCATE TABLE tmpSales

با نحوه حذف تمام سطرهای یک جدول آشنا شدید. اگر مایلید تنها

ساختار جدول باقی بماند روش زیر را به کار برید:

TRUNCATE TABLE tmpSales

بر خلاف دستور DELETE این دستور پیامی مبنی بر تعداد سطرهای

حذف شده باز نمی گرداند. موارد زیر تفاوتهای دستور DELETE بدون

عبارت WHERE و دستور TRUNCATE TABLE را بیان می کنند:

• DELETE هر سطر حذف شده را در فایل گزارش تراکنشها (log)

ثبت می کند در حالیکه TRUNCATE TABLE نگهداری کلید

ایندکس ها را در یک مرحله انجام می دهد و تنها آزاد سازی صفحه

را ثبت می کند.

• TRUNCATE TABLE تنها توسط دارنده جدول عضوی از نقش

پایگاه داده db – owner و یا نقش سرویس دهنده Sysadmin قابل

اجراست در حالی که دستور delete توسط هر کاربری با مجوزهای

مناسب بر روی جدول به کار می رود.

• تریگر DELETE هنگام اجرای دستور TRUNCATE TABLE

فعال نمی شود اما هنگام حذف سطرها با دستور DELETE انجام می

شود.

• دستور TRUNCATE TABLE مقدار identity را به هسته

(seed) بر می گرداند در حالیکه دستور DELETE تأثیری در مقدار

identity بعدی ندارد.

به روز رسانی داده (UPDATE)

دستور سوم تغییر داده، دستور UPDATE است که امکان تغییر

مقادیر ستونها در سطرهای موجود را فراهم می آورد.

۲-۳-۱۱- بکارگیری دستور UPDATE

نگارش این دستور به شرح زیر است:

UPDATE

{table – name | view – name}

SET

Column – name1 = {expression 1 | NULL | (select – statement)}

[, column – name 2 =...]

[WHERE search – conditions]

**عبارت SET ستونهای به روز شونده را مشخص می کند. در این دستور**

**عبارت WHERE اختیاری است.**

**دستور زیر مقدار ستون ytd – sales جدول tmpTitles را در هر**

**سطر به صفر تغییر می دهد. ممکن است بخواهید این مقدار را در**

**ابتدای هر سال تغییر دهید.**

UPDATE tmpTitles

SET ytd – sales = 0

**دستور فوق به بدون عبارت WHERE مقدار این ستون در کلید**

**سطرها را صفر می کند. مثال زیر ستون city ناشر Algodate**

**Infosystems را به روز رسانی می کند:**

UPDATE tmpPublishers

SET city 'EL Cerrito'

Where pub – name = 'Algodata Infoystem'

با دستور UPDATE می توانید مقدار جدیدی را بر حسب مقدار

موجود ستون ایجاد کنید. مثال زیر قیمت کتابهای روانشناسی

(psychology) را ده درصد کاهش می دهد:

UPDATE tmpTitles

SET price = price \* 0.90

WHERE type = 'psychology'

دستور UPDATE میتواند بیش از یک ستون را به بروز رسانی کند.

برای این منظور نام ستونها در عبارت SET با کاما یکدیگر جدا کنید.

مثال زیر مبلغ کلیه کتابهای Popular Computing را ۲۰ درصد

افزایش داده ورشته (Price Increase) را به فیلد notes آنها می

افزاید:

UPDATE tmpTitles

SET price = price \* 1.2, notes = notes + '(price increases)'

WHERE type = 'popular – comp'

دستور UPDATE و جدول جستجو

دستور UPDATE می تواند سطرهایی از یک جدول را به روز رسانی کند. اما SQLServer این امکان را فراهم می آورد که جدول دیگری به عنوان جدول جستجو برای آن مشخص شود. این جدول معمولاً با یک زیر پرس و جو در عبارت WHERE یا SET ظاهر می شود مشخص می گردد.

همچنین می توانید با استفاده از SQL – Transact دستورات UPDATE جدیدی بنویسید که در عبارت FROM آنها چندین

جدول به کار رفته باشد. این روش شکل تلفیق به دستور UPDATE می دهد. اما تنها یک جدول تغییر می یابد.

قابلیت فراهم شده مشابه زیر پرس و جوهاست و جدول دوم صرفاً به عنوان جدول جستجو به کار می رود.

برخی دستورات UPDATE با تلفیق پیچیده تر می شود. یکی مورد زمانی است که دستور UPDATE در هر دو عبارت WHERE و SET از زیر پرس و جو استفاده کند.



## ۲-۴-۱- ایجاد یک پروژه ویژوال InterDev

نخستین مرحله در ایجاد یک وب سایت با استفاده از ویژوال InterDev

ایجاد یک پروژه جدید بستگی به این دارد که شما در حال حاضر برنامه

کابردی ویژوال InterDev را اجرا کرده اید. یا خیر. به طور پیش گزیده

دیالوگ پروژه ساخت نمایش داده می شود. اگر در ویژوال InterDev

هستید از منوی اصلی گزینه های File و New را انتخاب کنید.

به جای دیالوگ File Open دیالوگ پرونده جدید از نوع چند برگی

است و به شما امکان می دهد ایجاد پروژه های جدید و یا باز کردن

پروژه های موجود را از یک دیالوگ انجام دهید.

در این مرحله مطمئن شوید که شما New web Project از برگه New

انتخاب شده است. در کنترل ویرایشی Name نام AddrBook را وارد

کنید. توجه کنید کنترل ویرایشی Location مطابق با نام کامل پروژه ای

که تایپ می کنید به روز در می آید. سپس کلید Open را کلیک نمایید.

در این لحظه ویزارد پروژه وب را می بینید. ویزارد پروژه وب مجموعه

های از دیالوگ هاست که به شما امکان می دهد تنظیمات اولیه برای

پروژه خود را تعیین و به پایان برید.

در پاسخ به پرسش می خواهید از چه سرویس دهنده ای استفاده کنید؟ لازم است تا نام یک سرویس دهنده فعال وب را مشخص نمایید. چنانچه ایجاد پروژه خود را بر روی دستگاهی انجام می دهید که سرویس دهنده وب نیز هست می توانید مقدار localhost را مشخص نمایید. توجه داشته باشید که پیشوند http و https را مشخص نکنید.

نکته قابل توجه دیگر مربوط به انتخاب Secure Socket Layer (SSL) است. SSL یک پروتکل امنیتی مربوط به Netscape است که برای رمز گشایی HTTP در هنگام ارسال و دریافت به / از سرویس دهنده وب به کار می رود در این مرحله SSL انتخاب نشده قرار دهید. درخاتمه لازم است تا مواد عملیاتی را انتخاب کنید. اساساً مود محلی به معنای آن است که می خواهید در یک محیط ایزوله کار کنید به طوریکه تغییراتی که در پروژه خود می دهید در سرویس دهنده وب انعکاس نداشته باشد. برای این پروژه مود Master را انتخاب کنید و کلید Next را کلیک نمایید.

مرحله دوم به شما امکان می دهد تا مشخص نمایید این پروژه یک برنامه کاربردی وب کاملاً جدید است و یا به یک برنامه موجود اضافه خواهد

شد. در حال حاضر مقدار موجود را بدون تغییر به کار برید. این دیالوگ همچنین امکان می دهد تا ایجاد اتوماتیک پرونده `Search.htm` را مشخص نماید. این پرونده جستجوی متنی را روی کلید صفحات مربوط به برنامه وب را برای کاربران آن فراهم می سازد. در مرحله بعدی ویزارد وب در ارتباط با انتخاب طرح و زمینه برنامه کاربردی است.

پس از آنکه دکمه `Finish` را کلیک کردید، ویژوال `InterDev` فرایند ایجاد پرونده های جدید پروژه شما را آغاز می نماید. ۲-۴-۲- بررسی واسط ویژوال `InterDev` برای یک پروژه بعد از اینکه یک پروژه ویژوال `InterDev` ایجاد کرده اید، لازم است تا بررسی واسط کاربر ویژوال `InterDev` را شروع نماییم.

نوار عنوان (Title Bar)

در هنگام شروع ویژوال `InterDev` پس از ایجاد یک پروژه جدید نخستین چیزی که احتمالاً به آن توجه خواهید کرد آنست که نوار عنوان مربوط به برنامه ویژوال `InterDev` تغییر کرده و شامل نام کامل راه حل

شماست. این نام قالب Web server/ Solution name و مقادیری که در

زمان ایجاد راه حل یا پروژه وارد کرده اید را داراست.

[www.kandoo.cn.com](http://www.kandoo.cn.com)

[www.kandoo.cn.com](http://www.kandoo.cn.com)

[www.kandoo.cn.com](http://www.kandoo.cn.com)

[www.kandoo.cn.com](http://www.kandoo.cn.com)

## کاوشگر پروژه (Project Explorer)

کاوشگر پروژه پرونده های مربوط به راه حل را کاملاً مشابه ساختار یک وب سایت و به شکل سلسله مراتبی نمایش می دهد. فهرست پرونده ها شامل صفحات HTML , ASP , تصاویر، اپلت ها و غیره ایجاد شده برای هر پروژه می باشد.

این پنجره از آن جهت کاوشگر پروژه نامیده می شود که در حقیقت از کلیه توابع کاوشگر نظیر کشیدن و رها کردن (Drag & Drop) ایجاد پوشه های جدید برای سازماندهی پرونده ها و تغییر نام عناصر و منوهای پدیدار شونده حساس به متن (Context Menus) پشتیبانی می نماید. در این مرحله بر روی پوشه های مختلف کلیک کنید تا ایده ای از محتویات و توابع قابل انجام بر روی هر آیت بدست آوریم. این نکته را در ذهن داشته باشید که مشابه سایر برنامه های ۳۲ بیتی ویندوز می توان تنها با کلیک کردن دگمه راست ماوس منوی پدیدار شونده حساس به متن که در آن کلیه انتخابهای خاص هر عنصر لیست شده اند را نمایش دهید اگر به عنصر دوم در کاوشگر پروژه نگاه کنید ورودی مربوط به سرویس دهنده وب AddrBook را خواهید دید. این ورودی نشاندهنده

ریشه مجازی پروژه شماست و نقطه برنامه است. چنانچه یک راه حل شامل چندین پروژه باشد هر کدام از پروژه ها ریشه مجازی مربوط به خود را تحت نام کامل راه حل خواهند داشت.

بنا به دلایلی، زمانیکه شما یک پروژه ویژوال InterDev را ایجاد می کنید، یک علامت + قبل از نام هر پوشه قرار داده می شود. در حالت عادی این + نشاندهنده ساختار درختواره قابل گسترش برنامه است. در مواردی اگر بر روی بعضی از آنها کلیک کنید خواهید دید که علامت “+” ناپدید می شود بنابراین در قسمت بالای کاوشگر پروژه نوار ابزاری را شامل یک تا ۵ دکمه بسته به عنصر انتخاب شده فعلی مشاهده خواهید کرد. اگر نام راه حل انتخاب شده باشد، تنها یک دکمه در نوار ابزار وجود خواهد داشت که امکان می دهد خواص مربوط به راه حل را نمایش دهید. اگر آیتم دیگری انتخاب شده باشد دکمه های زیر را قابل دیدن می باشد:

• Open این دکمه برای باز کردن یک پرونده نظیر صفحه HTML یا

ASP جهت ویرایش به کار می رود.

• Refresh Project Viwe کلیک کردن این دکمه کاوشگر پروژه را

Refresh می کند تا منعکس کننده هر گونه تغییرات خارجی بر روی

پرونده های پروژه باشد.

• Synchronize Files این گزینه به شما امکان میدهد تا پرونده

های محلی را با نسخه های موجود در سرویس دهنده و یا بالعکس

همزمانی نمایید این امر در مواردیکه به شکل offline کار کرده و می

خواهید تا سرویس دهنده را به روز در آورید و یا نسخ پرونده های

محلی شما با پرونده های سرویس دهنده همزمان نیستند بسیار مفید

است.

• Copy Webserver این گزینه برای کپی یک پروژه بر روی یک

سرویس دهنده وب دیگر و یا ایجاد نسخه دوم پروژه موجود بر روی

همان سرویس دهنده وب با نام جدید به کار می رود.

• Properties از این گزینه برای نمایش صفحه خواص یک شیء

استفاده می شود. بسته به نوع شیء و صفحه معمولاً نکاتی مانند نام

پرونده محلی مربوط به شیء نام پرونده سرویس دهنده مربوط به شیء و سایر اطلاعات در سطح پرونده را نشان می دهد. چنانچه نسخه ای از سیستم کنترل ویرایش مشابه با ویژوال SourceSafe نصب کرده باشید صفحه خواص اطلاعات مرتبط با ویرایش را نیز نمایش می دهد.

نکته دیگر قابل توجه شمایی است که در سمت چپ هر آیتیم از کاوشگر پروژه قرار دارد. شمایل کوچک به شکل مداد نشاندهنده آن است که یک نسخه خواندنی / نوشتنی از پرونده در اختیار دارید. در حالیکه شمایل قفل کوچک به این معنا است که یک کپی محلی از پرونده را به شکل فقط خواندنی در اختیار دارید.

• Open این گزینه برای باز کردن یک پرونده جهت ویرایش به کار می رود. در صورتیکه پرونده فقط خواندنی باشد، ویژوال InterDev طی پیامی سؤال می کند آیا می خواهید ویرایش فعلی از پرونده را از سرویس دهنده بازیابی کنید و یا پرونده مورد نظر را فقط خواندنی باز کنید.



• **Open With** این گزینه به شما امکان می دهد که تا پرونده ای را

توسط برنامه دیگر که می تواند این نوع پرونده را پشتیبانی کند باز

نمایید. در حالیکه پرونده محلی فقط خواندنی باشد ویزوال

InterDev طی پیامی انتخاب های یکسانی را مشابه حالت "Open"

برای یک پرونده فقط خواندنی نمایش دهد.

• **Get Latest Version** این گزینه برای همزمان سازی کپی محلی

یک پرونده با کپی موجود در سرویس دهنده وب به کار می رود.

برای مثال فرض کنیم شما می دانید ویرایش جدیدی از یک پرونده

یک گروهی از پرونده ها در سرویس دهنده وب موجود است. با

انتخاب پرونده های مورد نظر گزینه فوق کارایی بیشتری نسبت به

همزمان سازی کلیه پروژه ها یا راه حل دارد.

**Get Working copy** این گزینه صریحاً به ویزوال InterDev می

گوید ویرایش خواندنی / نوشتنی یک پرونده را از سرویس دهنده وب

بازیابی کند.

پنجره خواص (Properties Winvos)

به طور پیش گزیده این پنجره درست در زیر کاوشگر پروژه قرار دارد.  
پنجره خواص برای نمایش تنظیمات زمان طراحی عنصری که در حال حاضر در ویرایشگر پروژه انتخاب شده است به کار می رود نام عنصر مربوطه در کادر کومبر در بالای پنجره خواص قابل مشاهده است.

پنجره خواص همچنین دارای یک نوار ابزار با سه کلید می باشد و برای نمایش هر چه ساده تر خواص عناصر انتخاب شده به کار می رود.

• **Alphabetic** این گزینه امکان می دهد تا خواص را بر اساس نام

خصوصیت مرتب نماییم. زمانیکه با عنصری با سه خصوصیت کار می

کنیم، این انتخاب تأثیر چندانی ندارد ولی هنگام کار با عناصر دارای

۲۰ تا ۳۰ خصوصیت ارزش این انتخاب را قطعاً در خواهید یافت.

• **Categorized** برخلاف فهرست الفبایی که کلیه خواص را در یک

سطح نشان می دهد این گزینه امکان می دهد تا خواص را دسته

بندی نمایید. برای مثال فرض می کنیم بدنبال خواصی از یک عنصر

هستید که دارای ۱۰ خصوصیت نمایشی و ۱۰ خصوصیت متفرقه باشد.

در صورت استفاده از دکمه **Alphabetic** کلیه این خواص به صورت

درهم خواهید دید زیرا آنها صرفاً بر اساس ترتیب الفبایی و بدون

توجه به مفهوم منطقی فهرست بندی می شود. اما در حالت تنظیم  
Categorized هر یک از خواص تحت سرفصل طبقه مربوط لیست  
خواهد بود.

• Property Pages این کلید مشابه کلیک کردن کلید Properties  
Toolbar در کاوشگر پروژه عمل می کند.

### افزودن صفحه HTML

چنانچه کاربری بخواهد به صفحه ای از سرویس دهنده وب شما  
دسترسی داشته باشد میتواند صریحاً URL کامل مربوط را تایپ نماید.  
اما اگر Default.htm را ایجاد کرده باشد، کافیس نام سرویس دهنده  
وب و شاخه مجازی را تایپ نماید. بنابراین نخستین صفحه ای که شما  
ایجاد خواهید کرد صفحه پیش گزیده پروژه خواهد بود.

با توجه به امکاناتی که ویژوال InterDev مدعی است به عنوان  
محیط توسعه برنامه های کاربردی وب داراست، انتظار خواهید داشت  
تا ایجاد یک صفحه HTML بسیار ساده باشد. ایجاد یک صفحه  
HTML بسادگی کلیک کردن دکمه راست ماوس در ریشه پروژه و  
انتخاب گزینه Add از منوی HTML Page می باشد. انجام این کار

در این مرحله سبب نمایش دیالوگ Add Item می گردد. ابتدا کنترل کنید آیتم HTML Page در برگه New انتخاب شده باشد. سپس در کنترل ویرایشی Name کلمه Default را وارد و دکمه Open را کلیک نمایید. ویژوال InterDev کد منبع برای پرونده Default. Htm را ایجاد خواهد کرد. توجه داشته باشید نیازی به تعیین پسوند برای صفحه HTML نمی باشد. این پسوند به شکل پیش گزیده html خواهد بود.

بسیاری از سرویس دهنده های وب نظیر سرویس دهنده اینترنت مایکروسافت و سرویس دهنده وب شخصی مایکروسافت، نام Default.htm را به عنوان نام سند پیش گزیده در برنامه های وب به کار می برند. بنابراین اگر می خواهید نام متفاوتی (نظیر Default.htm) انتخاب نمایید لازم است تا پیکر بندی سرویس دهنده وب خود را متناسب با آن تغییر دهید.

پس از تمام عملیات فوق ویژوال InterDev به طور اتوماتیک پرونده جدید را باز می کند. به وضعیت سه برگه های Source, Design و Quick View در پایین صفحه برگه Design امکان می دهد تا به

شکل بصری صفحه خود را ویرایش کنید. این کار از طریق کشیدن و رها کردن عناصر بر روی صفحه انتخاب متن و تغییر فونت تعیین فشردگی و سایر مشخصه های مربوط انجام می شود. نکته مهم در برگه Design آن است که امکان می دهد تا صفحات HTML را بدون دانش HTML ایجاد نمایید. برگه Design امکان می دهد تا کارهای اولیه عادی نظیر افزودن متن و پیوند را به روشی سریعتر از تایپ دستی در کد HTML اجرا نمایید. برگه Source امکان می دهد تا کد مبلغ HTML هر صفحه را مشاهده کنید. این امر زمانی سودمند است که بخواهید به شکل دستی که HTML را در صفحه وارد کنید برای نمونه می توان از اتصال اسکریپت CGI جهت افزودن شمارنده صفحه برای یک وب سایت و پیگیری تعداد افرادی را که در هر روز از سایت بازدید کرده اند، نام برد. تعدادی شمارنده آزاد در اینترنت موجود است و فراهم آورنده این سرویس می گوید کدام HTML را در صفحه خود قرار دهید تا بتوانید برنامه CGI آنها را فراخوانی کنید. پس از آن برنامه CGI مسئول نگهداری و پیگیری تعداد دفعات سرزدن به صفحه وب در یک مقطع زمانی مشخص خواهد بود.

آخرین برنامه برگه مربوط به Quick View است و امکان می دهد تا صفحه طراحی شده را دقیقاً مشابه آنچه یک مرورگر نمایش می دهد، ببینید مزیت این برگه این است که ناچار نیستید جهت تست صفحه مرتباً برنامه خود را تعویض نمایید.

در این مرحله بر روی یک برگه Design کلیک کنید و سپس وارد ناحیه آزاد صفحه پروند Default.him شوید. حال متنی را تایپ کنید. این متن را انتخاب و با استفاده از نوار ابزار HTML فونت آن را از حالت پیش گزیده Times eb Roman به Comic Sans Ms تغییر دهید. همچنین می توانید اندازه فونت را بسادگی با انتخاب اندازه جدید از کادر کومبو که در همین نوار ابزار گرفته است تغییر دهید. در این مرحله اندازه فونت را به ۶ تغییر دهید. در انتها متن را با استفاده از کلید Center در همین نوار ابزار در مرکز صفحه قرار دهید. چنانچه در یافتن این دکمه ها مشکل دارید. این امر می تواند بواسطه دو دلیل باشد. اولین دلیل آنکه نوار ابزار قابل مشاهده نباشد. نوار ابزار با HTML دارای دو کادر کومبو بزرگ است که شناسایی آن را ساده می سازد. نخستین آنها برای تعیین سبک و شیوه متن ( /...)

InterDev کلیک راست نمایید و منوی HTML را از آن انتخاب

نمایید. این کار سبب می شود تا نوار ابزار HTML قابل مشاهده

گردد. چنانچه نوار ابزار وجود دارد اما دکمه ها را بر روی آن نمی

یابید، کافست موس را حدود ۳ ثانیه بر روی دکمه نگه دارید تا نام

دکمه مربوطه را مشاهده کنید. انجام این کار بر روی دکمه ها راه

مناسبی جهت آگاهی یافتن از مسئولیت هر یک از دکمه هاست.

حال که برای این صفحه یک عنوان انتخاب کردید نشانگر را به انتهای

خط حرکت داده و کلید Enter را دو بار فشار دهید. پس از آن دکمه

Align Left از نوار ابزار را انتخاب و اندازه و فونت را به ۳ تغییر

دهید. سپس کلمات Address Book را وارد کنید. حال برای آنکه

این کلمات یک ارتباط فرامتنی با صفحه دیگر داشته باشند، خط تایپ

شده را به طور کامل انتخاب کنید و بر روی LINK از نوار ابزار

HTML کلیک نمایید. در آخرین کلید در سمت راست به شکل زنجیر.

که دیالوگ ارتباط فرامتنی را نمایش می دهد.



در کنترل ویرایشی URL نام صفحه ای را که عهده دار نمایش اطلاعات نشانی می باشد وارد نمایید. در اینجا می توانید با کلیک کردن روی برگه Source ببینید تا ویژوال InterDev چه مقدار کار را تنها بازای چندین ماوس کلیک شما انجام داده است. نکته دیگر قابل توجه آن است که ویژوال InterDev به اندازه کافی هوشمندی دارد که بداند با انتخاب هر برگه چه امکاناتی کاربر دارد. بنابراین سایر امکانات و انتخابها نامعتبر خواهد بود.

حال که صفحه پیش گزیده پروژه خود را دارید صفحه دیگری را قبل از تست پروژه توسط مرورگر ایجاد نمایید. در این بخش تنها یک ASP خیلی ساده ایجاد خواهید کرد.

ASP ترکیبی از HTML و کد اسکریپت است. همانطور که از نام آن بر می آید این صفحات در سرویس دهنده قرار می گیرند و توانایی نمایش محتویات فعال یا پویا را فراهم می سازد. نمونه قابل توجه از ASP وب سایت میکروسافت است. زمانیکه شما وب سایت میکرو سافت را مرور می کنید متوجه می شوید که یک به دفعات یک ASP را مشاهده می کنید. این امر بدین دلیل است که میکروسافت



استفاده گسترده ای از قابلیت‌های ASP برای ایجاد محتویات پویا بر اساس فاکتورهای نظیر تاریخ و نمودار کاربران انجام داده است.

برای ایجاد یک ASP جدید، در کاوشگر پروژه کلید راست ماوس را بر روی ریشه پروژه کلیک کنید و بخش Add ASP نمایش داده می‌شود و آیتم صفحه ASP به طور پیش‌گزیده ای انتخاب می‌شود. در کنترل ویرایشی تمام URL نوشته شده در قبل را تایپ کنید. سپس دکمه Open را کلیک نمایید.

همانگونه که در بخش قبل انجام دادید، روی برگه Design کلیک کرده و سپس دکمه Align Center نوار ابزار را کلیک نمایید. فونت Comic Sans Ms و اندازه فونت انتخاب کنید. کلماتی را تایپ کنید. سپس کلید Enter را دو بار فشار دهید. در این مقطع لازم است تا در ارتباط با نحوه بازیابی اطلاعات تصمیم‌گیری نمایید.

در اینجا جدولی حاوی داده‌های متنی ایستا ایجاد خواهید کرد، به طوری که این صفحه در زمان نمایش اطلاعات مفیدی را نشان دهد.

برای اضافه کردن یک جدول به صفحه در منوی اصلی ویزوال  
InterDev گزینه های table و Insert table دیالوگ مربوط به  
افزودن جدول را نمایش خواهد داد.

همانگونه که می بینید این دیالوگ امکان می دهد تا گزینه های  
متعددی را برای جدول جدید تعیین کنید. در این مرحله مقادیر پیش  
گزیده را تأیید و دکمه OK را فشار دهید.

حال شما می توانید یک جدول  $3 \times 3$  را ببینید. اندازه فونت را به ۳  
تغییر داده و در ردیف بالا عناوین ستونها را وارد کنید. سپس سطر اول  
اطلاعات را انتخاب و دکمه Bold از نوار ابزار HTML را کلیک  
نمایید.

پس از آنکه عناوین ستونها را به پایان رساندید، اطلاعات را در دو  
سطر جدول وارد کنید.

چنانچه داده ها به طور کامل در سلولهای مختلف قرار نگرفت ممکن  
است نیاز به تغییر اندازه جدول داشته باشید. بر روی حاشیه جدول  
کلیک کنید و نشانگر ماوس را در محلی قرار دهید که به شکل نشانگر  
دو سر تعیین اندازه تبدیل شود. در این مرحله اندازه جدول و ستون

ها را به نوعی تغییر دهید که داده هر سلول تنها در یک خط قرار

گیرد.

## مشاهده ASP

چنانچه مبادرت به دیدن ASP از برگه Quick View در ویژوال

InterDev نمایید پیام هشدار با این مضمون که ویژوال InterDev

نمی تواند اسکریپتهای بخش سرویس دهنده یا کنترل های مبتنی بر

سرویس دهنده را پردازش نماید دریافت خواهید کرد. این امر به این

دلیل است که این اسکریپتها و کنترلها باید بر روی سرویس دهنده

اجرا شوند، اما کماکان می توانید نتایج HTML را مشاهده نمایید.

زیرا این بخش از ASP برای اجرا در بخش سرویس گیرنده مد نظر

است. اگر می خواهید صفحه مربوطه را مشاهده کنید و اسکریپت آن

را نیز اجرا کنید می توانید گزینه های View و View in Browser

را انتخاب کنید.

تست برنامه وب نمونه

برای تست در دیالوگ Run از منوی Start ویندوز یا در فیلد URL

در مرورگر خود آدرس مربوطه را تایپ می کنید.

توجه داشته باشید که آدرس های پست الکترونیکی ارتباطات فرامتنی را

نشان می دهند. زمانیکه بر روی آنها کلیک می نمایید بطور اتوماتیک

برنامه پست الکترونیکی پیش گزیده با ((To Field)) مقدار دهی شده  
نمایش داده می شود. چنانچه به عقب و ویژوال InterDev بازگردید و  
بر روی برگه Source پرونده کلیک نمایید خواهید دید ویژال InterDev  
مقادیر پست الکترونیکی را تشخیص می دهد. و بطور اتوماتیک HTML  
لازم برای نمایش آدرسها را به شکل ارتباطات فرامتنی ایجاد و برنامه  
پست الکترونیکی شما را در زمانیکه بر روی آدرسها کلیک می نماید  
آغاز می کند. این امر گواه دیگری از زمان و انرژی است که در هنگام  
کار با ادیتور ویژوال InterDev ذخیره می شود.

۲-۴-۳- توسعه و تکمیل صفحات وب به وسیله ویژال InterDev

### ایجاد عناوین صفحات وب

همانطور که حدس زدید، طراحی صفحات خوب و تاثیرگذار با ویژوال  
InterDev نسبتا ساده است. حال بینیم با چه سرعتی می توان تغییرات  
در هر یک از صفحات وب با استفاده از ادیتور WYSIWYG ویژوال  
InterDev انجام داد.

در این مقطع یک پروژه جدید ایجاد کنید. سپس یک صفحه HTML به  
آن اضافه کنید. در بالای صفحه عنوانی را وارد کنید. سپس با استفاده از

دگمه‌های مربوط به تعیین رنگ پیش زمینه (ForeGround Color) و پس زمینه (BackGroundColor) از همان نوار ابزار، رنگ دلخواه را انتخاب کنید چنانچه مایل باشید بتوانید فونت عنوان فوق را نیز تغییر دهید.

### ایجاد فهرستهای بولت دار

از فهرستهای بولت دار می توان برای نمایش مجموعه ای مرتب شده از دستورات استفاده کرد. برای مثال در مستند سازی یک ماجول نرم افزاری، مراحل نصب سیستم با این فهرست قابل نمایش است. همچنین این فهرست برای بر شمردن لیست عناصر کاربرد دارد.

حال دگمه چپ چین (Align Left) را انتخاب و لیست نمونه زیر را تایپ نمایید.

- How to create Web page
- How to create an Active Server Page
- How to apply Styles

پس از آنکه عناصر فوق را وارد کردید، آنها را انتخاب و از نوار ابزار دگمه Bulleted List را کلیک نمایید.

در صورتیکه بخواهید لیست فوق شماره گذاری شده باشد، می توانید دگمه Numbered List را کلیک کنید.

## تعریف اتصال (LINK) به سایر صفحات

برای افزودن یک ارتباط فرامتنی به یک صفحه، کفیسیت از منوی

HTML گزینه LINK را انتخاب کنید. پس از آن ویژوال InterDev

دیالوگ را به شما نشان می دهد که در آن نوع اتصال قابل تعریف

است. این ارتباط فرامتنی می تواند به هر منبع اینترنتی باشد. دیالوگ

ارتباط فرامتنی دارای یک کادر کومبو برای تعیین نوع ارتباط است.

این لیست شامل News , http , ftp , mailto, gopher, telnet می

باشد. پس از آنکه نوع ارتباط دلخواه را تعیین کردید، این دیالوگ

پیشوند مناسب با سرویس دهنده اینترنت انتخابی را در URL قرار می

دهد.

مشخص کردن صفحاتی از سایر پروژه های ویژوال InterDev

در تعریف یک ارتباط فرامتنی هم می توانید صریحاً URL دلخواه را

وارد کنید و هم اینکه با انتخاب دکمه مرور (Browse) از سایر پروژه

های ویژوال InterDev موجود در Solution فعلی صفحه ای را انتخاب

کنید. در این حالت دیالوگ (Create URL) بر روی صفحه نمای داده

می شود. پروژه های این دیالوگ را مرور و منبع مورد نظر را برای اتصال انتخاب کنید.

افزودن تصاویر گرافیکی به صفحات وب

بطور قطع هیچ چیز در اتصال یک پیام بهتر از ترکیب متن مناسب با تصویر یا تصاویر پشتیبان نیست در این دنیای شتابنده که بنظر می رسد هر کسی در آن وقت کمتری برای انجام کارهای روز افزون دارد، دیگر فرصت و تمایلی برای خواندن کامل یک متن جهت درک محصولات شما وجود ندارد. به همین دلیل است که امروزه شرکتها وجوه عمده ای را صرف ایجاد جملات و تصاویر جذاب می کنند تا توجه مشتریان را به خود جلب نمایند. آنها امیدوارند اگر تصاویر توانستند توجه شما را جلب کند، محصولاتشان باقی کارها را انجام خواهد داد.

برای افزودن تصاویر ابتدا به محلی از صفحه بروید که می خواهید

شکل مربوطه در آنجا درج شود. سپس از منوی HTML گزینه Image

را انتخاب کنید. اینکار دیالوگ Insert Image را فعال می کند.



در این دیالوگ امکانات گسترده ای جهت کنترل نحوه قرارگیری تصویر در صفحه وجود دارد. نمونه ای از امکانات قابل دسترسی برای تولید تاثیرات دلخواه به شرح زیر است:

Picture Source در این بخش مسیر کامل تصویر مورد نظر را وارد کنید و با دکمه Browse تصویری را جستجو نمایید که در پروژه ویژوال InterDev جاری قابل انتخاب باشد. برای آشنایی با نحوه در دسترس قرار دادن تصاویر برای انتخاب به بخش افزودن تصاویر به پروژه ویژوال InterDev مراجعه نمایید.

Aternate Text هر آنچه در این بخش وارد شود به عنوان ToolTip در نظر گرفته شده و با چند ثانیه نگهداری ماوس بر روی تصویر نمایش داده می شود.

Alignment گزینه های این بخش به شما امکان می دهد تا تصویر را در محلی که می خواهد قرار دهید برای مثال چنانچه می خواهید تصویر در سمت راست و متن در سمت چپ قرار گیرد گزینه Right را انتخاب کنید.

Border Thickness در صورتیکه بخواهید تمایز بین تصویر درج شده و پس زمینه صفحه وب ایجاد کنید، می توانید از این گزینه استفاده کنید.

Horizontal and Vertical Spacing با استفاده از این گزینه می توانید تعداد پیکسلهای حد فاصل تصویر و متن اطراف آن را مشخص کنید. با این روش از ایجاد صفحات در هم و بر هم و غیر حرفه ای اجتناب می کنید.

افزودن تصاویر به پروژه ویزوال InterDev

برای اینکه بتوان تصویری را در صفحه وب درج کرد، لازم است تا تصویر متعلق به پروژه باشد. این کار بسادگی و با استفاده از کاوشگر پروژه انجام می شود. با کلیک راست روی پوشه Images و انتخاب گزینه های Add و Add Item دیالوگی نمایش داده میشود، برگه Existing از این دیالوگ را انتخاب کنید.

سپس در کادر کومبو Files of Type نوع Image را وارد کنید. با این روش می توانید به محل تصاویر ذخیره شده رفته و آنها را مرور کنید.

پس از انتخاب تصویر مورد نظر دکمه Open را فشار دهید. یک کپی از

پرونده این تصویر در پوشه تصاویر پروژه شما ایجاد خواهد شد. این تغییرات در کاوشگر پروژه قابل مشاهده می باشد.

۲-۴-۴- طراحی وب سایت با Sitedesigner

در آمدی بر Site Desiner

طراحی یک برنامه کاربردی شامل مراحل بررسی، تجزیه و تحلیل و تست تمام و کمال کلیه ایده های مرتبط با موضوع پیشنهادی است. این امر برای وب سایتها نیز صادق است. برخلاف برنامه های کاربردی تحت ویندوز که به نوعی کاربران را در مورد نحوه انجام کار محدود می سازند، کاربران وب سایتها به محیط آزادی عادت کرده اند که در آن از صفحه ای به صفحه دیگر بروند و تنها با یک کلیک ماوس به هر نقطه ای که مایلند برگردند. بنابراین قبل از آنکه به فکر طراحی صفحات وب برای سایت خود باشید، لازم است تا در مورد سازماندهی اطلاعات در سایت بگونه ای تصمیم گیری نمایید که کاربران براحتی بتوانند سایت شما را پیمایش کنند.

## نمودار سایت

نمودار سایت نمایش گرافیکی از صفحات وب است که در آن چگونگی پیمایش و حرکت از صفحه ای به صفحه دیگر مشخص شده است. در یک نمودار، صفحاتی که منطقی با یکدیگر مرتبط هستند. در یک ساختار سلسله مراتبی به نام درخت نگهداری می شوند. سپس از این درختها برای تعریف اتصالات پیمایشی استفاده می شود. ارتباطات سلسله مراتبی بین صفحات به راحتی قابل تغییر است و تنها با جابجایی آنها در نمودار سایت انجام می شود.

## مزایای استفاده از نمودار سایت

یکی از مزایای عمده استفاده از نمودار سایت امکان تولید سهل و سریع الگوی نمونه (Prototype) سایت بدون کد کردن صفحات HTML و ASP مربوطه است. علاوه بر آن نمودار سایت نحوه پیمایش صفحات در سایت و چگونگی تعامل (Interaction) کاربران با سایت را نشان می دهد.

نمودار سایت امکان طراحی سریع برنامه های وب را فراهم می سازد ایجاد صفحات جدید تنها با یک کلیک ماوس انجام می شود. همچنین

می توان سایر برنامه ها مانند کاوشگر اینترنت را باز کرد و ارتباطات فرامتنی دلخواه را در این نمودار کشاند این کار به معنای افزودن صفحه مربوطه به وب سایت جاری است. هنگامی که نمودار سایت را ذخیره می کنیم کلیه صفحاتی که وجود ندارند بطور اتوماتیک ایجاد و به پروژه اضافه می شوند.

اگر از Layout در هنگام ایجاد پروژه استفاده کرده اید، کلیه صفحات جدید ایجاد شده درست مشابه ایجاد صفحات توسط کاوشگر پروژه خواهد بود. همچنین اگر ساختار پیمایشی برای سایت تعریف کرده باشید، این ساختار به ازای هر صفحه جدید اتوماتیک به هنگام می شود.

### ایجاد نمودار سایت

در این مقطع یک پروژه نمونه ایجاد می کنیم مراحل کار عبارتند از

۱- پروژه جدیدی به نام SiteDesign بسازید

۲- کاوشگر پروژه را باز کنید.

۳- کلید راست ماوس را بر روی نام پروژه کلیک کرده و گزینه Add

از منوی Site Diagram را انتخاب کنید.

۴- پس از آنکه دیالوگ Add Item نمایش داده شد، نام پرونده سایت را مطابق SiteDesign.wdm قرار دهید. هر چند می توانید برای یک پروژه چندین نمودار سایت تعریف کنید اما تنها یک پرونده ساختار سایت وجود دارد به همین دلیل نام این پرونده مشابه نام پروژه انتخاب می شود.

**افزودن صفحات به نمودار سایت**

پس از آنکه نمودار سایت را ایجاد کردید، راههای مختلفی برای

**افزودن صفحات HTML , ASP به آن وجود دارد:**

- ایجاد صفحات جدید در نمودار سایت
- افزودن صفحات یک پروژه به نمودار سایت
- افزودن صفحات سایر برنامه های کاربردی به نمودار سایت

**ایجاد صفحات جدید در نمودار سایت**

برای ایجاد یک صفحه جدید در نمودار سایت از منوی اصلی گزینه

Diagram و سپس New HTML Page را انتخاب کنید. اگر مایلید

این صفحه ASP باشد

New ASP Page را انتخاب نمایید. مستطیلی که نشان دهنده صفحه

جدید است در نمودار سایت نشان داده می شود. ویژوال InterDev نام منحصر بفردی به هر صفحه اختصاص می دهد. از آنجائیکه این نام به عنوان نام واقعی پرونده در نظر گرفته می شود، در هنگام ذخیره نمودار سایت می توانید آن را به دلخواه تغییر دهید. در هنگام افزودن صفحات به نمودار سایت، هر صفحه جدید به عنوان فرزند صفحه انتخاب شده فعلی ایجاد می شود. به عبارت دیگر اگر شما روی صفحه home page باشید و سپس گزینه Diagram را انتخاب کنید صفحه بعدی به عنوان فرزند آن ایجاد و ارتباط با آن برقرار می شود. به همین طریق است که می توان با سادگی هر چه تمامتر صفحات و ارتباطات مابین آنها مشخص کرد.

در هنگام افزودن صفحات جدید به نمودار سایت توجه داشته باشید که این صفحات ایجاد شده تا قبل از ذخیره نمودار سایت در کاوشگر پروژه قابل مشاهده نیستند.

اگر می خواهید صفحه ای ایجاد کنید که دارای صفحه پدر نباشد، ابتدا بر روی پس زمینه کلیک کرده و سپس گزینه Diagram را انتخاب

کنید. در اینصورت صفحه جدید بدون ارتباط با صفحات دیگر ایجاد می شود.

روش دیگر برای افزودن صفحات به نمودار سایت، استفاده از صفحات ایجاد شده و موجود از پروژه فعلی است. برای این منظور مراحل زیر را انجام دهید:

(۱) نمودار سایت را ایجاد و یا آن را باز کنید.

(۲) کلید ماوس را بر روی صفحه پدر کلیک کنید. اگر نمی خواهید صفحه جدید دارای صفحه پدر باشد بر روی پس زمینه کلیک کنید.

(۳) از منوی پدیدار شونده حساس به متن گزینه Add Existing File را انتخاب کنید. اینکار سبب می شود تا دیالوگ CHOOSE URL

نمایش داده شود.

(۴) از این دیالوگ، صفحه دلخواه را انتخاب، پارامترهای مورد نظر را تنظیم و دکمه OK را فشار دهید.

افزودن صفحات سایر برنامه های کاربردی به نمودار سایت



یکی از امکانات جالب توجه Site Designer امکان کشیده و رها کردن ارتباطات فرامتنی از سایر برنامه‌های کاربردی به وب سایت جاری است. براین اینکار مراحل زیر را انجام دهید.

(۱) نمودار سایت را ایجاد و یا آن را باز کنید.

(۲) برنامه ای را که حاوی ارتباط فرامتنی مورد نظر جهت افزودن به نمودار سایت است باز کنید (برای مثال کاوشگر اینترنت)

(۳) برنامه ها را به شکل بصری بر روی صفحه طوری تنظیم کنید که بتوان در آن واحد هر دو آنها را مشاهده کرد.

(۴) اتصال مورد نظر در برنامه اصلی را انتخاب کنید.

(۵) اتصال انتخاب شده را بر روی نمودار سایت بکشید (Drag)

(۶) اتصال انتخاب شده را بر روی نمودار سایت رها کنید. (Drop)

برداشتن صفحات از نمودار سایت

برای برداشتن صفحات از نمودار سایت کافیست صفحات دلخواه را

انتخاب و سپس کلید Delete را فشار دهید. (در آن واحد می توانید

چندین صفحه و یا با Ctrl+A تمام صفحات را انتخاب کنید) پس از

اینکار دیالوگ تایید نمایش داده می شود. این دیالوگ را با دقت

بخوانید زیرا دیالوگ معادل « آیا مطمئن هستید؟ » نمی باشد. بلکه به شما امکان می دهد نه تنها صفحه را از روی نمودار سایت حذف کنید بلکه آن را از کل پروژه برداشته و فایل فیزیکی متناظر با آن را نیز حذف نمایید.

اگر بر روی یک صفحه پدر کلیک کنید و کلید Delete را فشار دهید آنگاه دیالوگ Delete Pages Children نمایش داده می شود. این دیالوگ به شما نشان می دهد که عمل شما نه تنها سبب حذف صفحات انتخاب شده بلکه شامل صفحات فرزند آنها نیز می شود.

وصل و فصل صفحات (Attach and Detach)

در طول مراحل طراحی یک وب سایت، ممکن است بارها نظر خود را نسبت به نحوه قرارگیری صفحات در ارتباط با هم تغییر دهید. همانطور که قبلاً دیدید، تعیین صفحه پدر برای صفحه جدید در هنگام افزودن صفحه به نمودار سایت انجام می شود، امام در عین حال می توانید این تغییرات را بعداً نیز بسادگی انجام دهید. برای وصل یک صفحه فرزند به صفحه پدر تنها کافیست صفحه فرزند را به سمت صفحه پدر بکشید در همین حال مشاهده می کنید که یک صفحه نقطه چین را

حرکت می دهید. پس از اینکه این صفحه تقریباً به مجاورت صفحه مورد نظر رسید یک خط ارتباطی بین آن دو نمایش داده میشود. در این مرحله کلید ماوس را آزاد کنید تا صفحه مورد نظر به عنوان صفحه فرزند با پدر خود مرتبط شود. مجموعه ای از صفحات را به شکل شاخه ای از درخت به سمت دیگر نمودار سایت بکشید. فصل یک صفحه از صفحه پدر نیز به همین راحتی انجام می شود. فقط کافیست صفحات مورد نظر را بکشید و در ناحیه با از نمودار سایت رها کنید.

#### ۲-۴-۵- تنظیم گزینه های ASP در ویژوال InterDev

قبل از کار با ASP لازم است تا از تنظیم صحیح گزینه های آن در ویژوال InterDev مطمئن شوید. برای این منظور مراحل زیر را دنبال کنید:

- ۱- یک پروژه را در ویژوال InterDev باز کنید.
- ۲- یک صفحه ASP موجود در این پروژه را باز کنید. اگر صفحه ای از نوع ASP ندارید برای این مثال فایل `global.asa` را باز کنید.

۳ پس از باز کردن یک صفحه ASP یا فایل [golobal.asa](http://golobal.asa) گزینه Viwe

و Property را انتخاب کنید تا دیالوگ Properties نمایش داده شود.

## زبان پیش گزیده

همانطور که قبلا دیدید در هنگام اسکریپت نویسی می توانید زبان

پیش گزیده ای را انتخاب نمایید. مقدار پیش گزیده اسکریپت نویسی

در بخش سرویس دهنده در ویژوال InterDev ویژوال بیسیک انتخاب

شده است. بنابراین در ابتدای هر صفحه ASP کد زیر وجود دارد:

```
<%@language = VBScript%>
```

اگر زبان پیش گزیده (ECMA) JavaScript باشد در ابتدای هر

صفحه ASP کد زیر را مشاهده خواهید کرد:

```
<%@language = JavaScript%>
```

فعال سازی تراکنش ها (transaction)

تراکنش ها به طراحان امکان می دهند تا مجموعه ای از عملیات را به

شکل منطقی گروه بندی کرده و به عنوان یک عمل منطقی منفرد

تعریف نمایند. بطوریکه اگر یکی از عملیات منجر به شکست شد نتیجه

مجموعه عملیات موفقیت امیز تلقی نگردد. برای مثال فرض کنید می

خواهید یک برنامه بانکداری بنویسید. اگر یک مشتری بخواهد مبلغی

پول را از حساب جاری خود برداشت و به حساب پس انداز خود واریز

نمایند، باید نخست حساب جاری او را بدهکار و سپس حساب پس انداز وی را بستانکار نمایند. حال فرض کنید پس از کاستن موجودی از حساب جاری، افزودن مبلغ به حساب پس انداز به هر دلیلی دچار مشکل شود. حال چه اتفاقی می افتد؟ در این شرایط مبلغ برآشته شده، نه در حساب جاری و نه در حساب پس انداز قرار دارد مسلماً باید راهی داشته باشید تا مبلغ برآشت شده را به حساب او بازگردانید. در این شرایط است که مفهوم تراکنش معنا می یابد. با گروه بندی عملیات برداشت و واریز به صورت یک عمل واحد مشخص می کنید که یا هر دو با هم انجام شود و یا هیچکدام انجام نشوند و بدین طریق صحت انجام عملیات را کنترل می کنید گزینه فعال سازی تراکنشها تعیین می کند که آیا سرویس دهنده تراکنش های مایکروسافت (MTS) باید تغییرات انجام شده در صفحات ASP را به عنوان تراکنش در نظر بگیرد یا خیر. در حقیقت MTS اینکار را از طریق Distributed(DTC) transaction Coordinator مایکروسافت انجام می دهد. و به همین دلیل است که در هنگام نصب IIS و DTC بطور اتوماتیک نصب می گردند. توجه داشته باشید این

قابلیت در سرویس دهنده وب شخصی (PWS) وجود ندارد. اگر این گزینه را انتخاب کنید، کد زیر در ابتدای صفحه ASP درج شود.

```
<%@Trasaction =required%>
```

### صفحات ASP بدون نشست

بازدید کاربر از یک سایت نشست (Session) نامیده می شود. شی Session جهت ذخیره اطلاعات مربوط به نشست کاربر با یک وب سایت به کار می رود. بنابراین این گزینه به معنای آن است که نگهداری هیچگونه اطلاعاتی جهت نشست کاربر از طرف سرویس دهنده مجاز نیست. با انتخاب این گزینه، کد زیر در بالای صفحه ASP درج می شود

```
<%EnableSessionState=False%>
```

### فعال سازی مدل اشیاء اسکریپت نویسی (Scripting Objrect Model)

این گزینه مشخص کننده آن است که شما مایل به پشتیبانی از مدل اشیاء اسکریپت نویسی نمی باشید این مدل با ویژوال InterDev6 معرفی شده و به طراحان وب امکان می دهد تا با صفحات وب مطابق اشیاء رفتار کنند بنابراین می توانید به هر صفحه وب در سایت مشابه

سایر اشیاء در اسکریپت ASP رجوع کنید. علاوه بر آن با این مدل اشیاء مفهوی به نام Remote Scripting نیز امکان پذیر می شود این مفهوم به شما امکان می دهد تا متدهای یک صفحه ASP را از صفحه سرویس گیرنده فراخوانی و اجرا نمایید.

## ۲-۴-۶- کاربرد فایل global.asa

پرونده global.asa پرونده ای است که با هر بار ایجاد یک پروژه جدید در ویژوال InterDev بطور اتوماتیک ایجاد می شود. این پرونده جهت اداره رویدادهای عمومی و اطلاعات اشیاء یک برنامه بکار می رود. برای مثال از طریق کاربرد فایل global.asa می توانید متدی را با هر بار شروع یک نشست جدید فراخوانی کنید. از این طریق می توانید عملیاتی نظیر پیگیری کاربران و کنترل Log را انجام دهید.

این پرونده شامل عناصر زیر است:

• رویدادهای Application

• رویدادهای Session

• تعاریف <OBJECT>



## • تعاریف TypeLibrary

احتمالاً بارها خواسته اید پیمایش کاربران در وب سایت خود را کنترل کنید. برای مثال اگر سایت شما نیاز به گذراندن مرحله Logon دارد، مایل نیستید کاربران بتوانند قبل از انجام اینکار مستقیماً به صفحات دیگر سایت شما بروند.

با استفاده از فایل `golobal.asa` اداره این شرایط بسیار ساده است.

کافیست یک روال برای رویداد `Session - Onstart` بنویسید سپس با

هر بار شروع یک نشست جدید این رویداد فعال و روال مربوط به آن

اجرا می شود. از این طریق می توانید مطمئن شوید نخستین صفحه ای

که کاربر آن را ملاحظه می کند صفحه `Login` است و در غیر اینصورت

کاربر را به این صفحه هدایت نمایید.

ایجاد توابع در سطح `Application`

هر یک از توابعی که در فایل `golobal.asa` معرفی کرده اید تنها می

توانند در یکی از اسکریپتهای مرتبط با رویدادهای `Session - OnEnd`,

`Session-OnStart`, `Application Onstart` , `Applivation-OnEnd`

فراخوانی شوند. به عبارت دیگر این توابع در تمام صفحات ASP برنامه وب در دسترس نیستند.

جهت ایجاد توابعی که بتوانند در سطح Application بکار روند،

مراحل زیر را دنبال کنید:

۱- یک صفحه در اسکریپت ایجاد کنید.

۲- نام پرونده را با پسوند .inc ذخیره کنید.

۳- از جمله Server Side Include (SSI) جهت Inckude کردن

فایل در هر صفحه ASP که نیاز به فراخوانی این توابع دارد استفاده نمایید.

۲-۴-۷- ایجاد یک پروژه پایگاه داده

برای ایجاد یک پروژه پایگاه داده گزینه File و New را از منوی اصلی

انتخاب کنید تا دیالوگ Newproject را ببینید در این مرحله بر روی

ورودی Visul Studio در سمت چپ دیالوگ دوبار کلیک کنید. با

اینکار سه زیر شاخه جهت معرفی سه نوع پروژه جدید بر روی صفحه

نمایش داده می شود. این تاپها عبارتند از , Utiily Projects

Database Projects, Distribution Units, بر روی عنصر Database

Project کلیک کرده و نام پروژه را وارد نمایید. سپس دکمه Open را انتخاب کنید.

پیکر بندی ارتباط داده‌ای

پس از آنکه پروژه پایگاه داده ایجاد شد. دیالوگ Data Link Property نمایش داده می شود. این دیالوگ شامل ۴ برگه متفاوت است که کلیه امور از فراهم سازی OLE DB تا تعیین مشخصه‌های ارتباط نظیر نام کاربر و رمز عبور را امکان پذیر می سازد.

چنانچه از یک پایگاه داده رابطه ای واقعی نظیر SQL Server یا اوراکل استفاده می کنید. مطمئن شوید که قبل از اتصال با آن، پایگاه داده در حال اجراست. در مورد 7 SQL SERVER برنامه SQL SERVER SERVICE Maneger را اجرا سرویس MS SQL SERVER را انتخاب و دکمه Start را کلیک کنید.

انتخاب OLE DB Provider

همانطور که می بینید اولین برگه دیالوگ Provider نام دارد که امکان انتخاب نوع OLEDB Provider جهت ارتباط با منبع داده ای را فراهم می سازد. به سطر سوم که ارتباط ODBC را ایجاد می کند

**دقت کنید اینکار برای پایگاه داده‌هایی که دارای ODBC Driver**

**هستند ولی OLE DB Provider ندارند بسیار مناسب است.**

### **تعریف ارتباط OLE DB**

**این برگه امکان تعریف مشخصه های ارتباطی خاص پایگاه داده ها را**

**فراهم می سازد. لازم است تا نام سرویس دهنده و نام پایگاه داده ای را**

**که این ارتباط با آن برقرار خواهد شد تعیین نمایید. عناصر متفاوتی که**

**در این دیالوگ وارد می کنید به سه بخش مجزا تقسیم شده است.**

**نخستین تعیین نام سرویس دهنده پایگاه داده است. در این مرحله نام**

**سرویس دهنده ای را انتخاب کنید که پایگاه داده شما برای روی آن**

**قرار گرفته است.**

**مرحله بعدی نحوه ورود Login به سرویس دهنده را مشخص می کند.**

**در اینجا دو گزینه وجود دارد. گزینه اول Windows NT Intergrated**

**Security به معنای استفاده از User ID و رمز عبور نشت جاری**

**(Session) شما جهت ورود به سرویس دهنده پایگاه داده است. گزینه**

**دوم به شما امکان می دهد تا یک User ID و Password جدید برای**

**اینکار تعریف کنید، در این مثال Username معادل sa بدون Password**

انتخاب شده است) و مقادیر پیش گزیده ای SQL-Server ارائه می گردد. اما در محیط های واقعی لازم است مقادیری را انتخاب نماید که تضمین کننده امنیت سیستم باشد.

مرحله آخر در این برگه به معرفی پایگاه داده مورد استفاده اختصاص دارد. چنانچه نام سرویس دهنده پایگاه داده و رمز عبور و User ID مناسب را انتخاب کرده باشید در این کادر کومبو نام چندین پایگاداده را خواهید دید.

پس از اینکه اطلاعات برگه Connection را تکمیل کردید، می توانید ارتباط خود را تست کنید برای اینکا دکمه Test Connection را کلیک نمایید. یک ارتباط کامل و موفق با پیامی نشان داده می شود.

#### تعریف خواص پیشرفته OLE DB

برگه Advanced اختصاص به خواصی نظیر Connect Timeout و مجوزهای دسترسی (Access Permission) دارد. Connect Timeout که بر حسب واحد ثانیه بیان می شود نشاندهنده زمان لازم جهت برقراری ارتباط با پایگاه داده است مجوزهای دسترسی نیز امکان تعریف

مجوزهای دسترسی برنامه پس از اتصال به منبع داده ای را فراهم می سازد.

### تعیین خواص منبع داده

با استفاده از برگه ALL می توانید مجموعه وسیعی از خواص مربوط به منبع داده برای یک ارتباط جدید را تعیین نمایید. از این خواص، آنهایی هستند که در برگه های قبل تنظیم کرده و یا نیازی به تغییر آنها ندارد. (برای مثال سایز (Local Identifier , Packet) بهر حال اگر بخواهید هر یک از آنها را تغییر دهید کفایت خصیصه مورد نظر را انتخاب و دکمه Edit Value را کلیک کنید. این کار سبب می شود تا دیالوگ کوچکی جهت درج مقادیر جدید نمایش داده شود.

### کار با طراح پایگاه داده (Database Designer)

پس از برقراری ارتباط با پایگاه داده و ویروال (InterDev) پنجره جدیدی را نمایش می دهد. پنجره Data Viwe تنها زمانی در دسترس است که قبلاً به یک پایگاه داده مرتبط شده باشد. این پنجره یک ابزار گرافیکی جهت مدیریت و کار با پایگاه داده و همچنین ابزاری جهت تغییر شمای فیزیکی پایگاه داده و داده های آن است. کاربرد نمودارهایی پایگاه داد

اولین عنصری که در زیر نام پایگاه داده ها در پنجره InterDev قرار دارد مربوط به نمودارهای پایگاه داده (Data base Diagram) است این نمودارها به شما امکان می دهد به شکل گرافیکی عناصر پایگاه داده نظیر جداول، ستونها، ارتباطات ایندکس ها و محدودیت ها را ایجاد کنید.

## ایجاد نمودار پایگاه داده

برای ایجاد یک نمودار پایگاه داده مراحل زیر را دنبال کنید:

۱- بر روی پنجره Data Viwe قرار بگیرید. اگر این پنجره باز نیست

با انتخاب گزینه های Other View Windows و Data View آنرا

باز کنید.

۲- کلید راست ماوس را بر روی Database Diagram کلیک کرده

وگزینه New Diagram را از منوی پدیدار شونده حساس به متن

انتخاب کنید. اینکار از طریق انتخاب گزینه های Add project

Diagram , Database Item نیز قابل انجام است.

۳- در این مرحله ویژوال InterDev یک پنجره خاصی را نشان می

دهد که برای تغییرات گرافیکی در شمای پایگاه داده بکار میرود.

۴- ویژوال InterDev بطور اتوماتیک یک نام پیش گزیده به نام

Database Diagram (n یک عدد ترتیبی است که برای تضمین

یکتایی نمودار بکار می رود) به این دیاگرام جدید اختصاص می

دهد.



در زمان ضبط دیاگرام می توانید نام مناسبتری برای آن ایجاد کنید.  
Order قرار دهید.

چنانچه کار خود را ذخیره نمایید، هیچ شی جدید در Data Viwe نخواهید دید حتی اگر از ویژوال InterDev خارج و مجدداً خارج و مجدداً آن را باز کنید باز هم هیچ ورودی جدیدی در Data Viwe نخواهید دید این امر بدین دلیل است که ویژوال InterDev نمودارهای خالی را ذخیره نمی کند.

افزودن جداول به نمودار پایگاه داده

افزودن جداول به نمودار پایگاه داده کار بسیار ساده ای است. برای اینکار جداول دلخواه را از Data Viwe بر نمودار بکشید و ویژوال InterDev هر گونه ارتباط ما بین این جدول و سایر جداول مرتبط موجود در نمودار را به شکل اتوماتیک نشان خواهد داد برای مثال دو جدول نمونه را روی نمودار بکشید. در این حالت خطی مابین این دو جدول خواهید دید که نشان دهنده یک ارتباط بین این دو جدول است.

## شرح صورت پروژه:

پروژه مورد نظر عبارت است از طراحی صفحات وب که توسط آنها شخصی که در سازمان تامین اجتماعی مشغول به کار است بتواند به عنوان یک کارمند جدید ثبت نام کرده و با وارد کردن مشخصات خود، اطلاعاتش در پایگاه داده طراحی شده قرار گیرد و پس از آنکه ثبت نام کرد بتواند بخش نامه های مربوط به واحد خود را مطالعه کند.

همچنین طراحی صفحات وبی که توسط آن بخش نامه های مربوط به هر واحد صادر شود. و در لیست بخش نامه ها قرار گیرد. و طراحی صفحاتی که از طریق آن بتوان در ارتباط با تعداد کاربران و مشخصات کاربران ثبت نام شده Quarry گرفت.

## بررسی و تحلیل نیازها:

یکی از دلایل اصلی ایجاد یک سیستم وجود پاره ای از نیازمندیهاست. به همین خاطر لازم است که از ابتدا از این نیازها به دقت شناسایی شده و ثبت گردد. نکته مهم این است که نیازها به دو گروه نیازهای کارکردی

(Functional) و نیازهای غیر کارکردی (nonfunctional) تفکیک

میشوند.

یک نیاز Functional عبارت از نیازی است که در ارتباط با

Functionality یا وظیفه مندی است که در حال حاضر در سیستم وجود

ندارد و سیستم نیازمند داشتن آن است.

اما یک نیاز غیر کارکردی به نیازی اطلاق می گردد که در حال حاضر در

سیستم وجود دارد اما بنا به دلایلی کار کردن آن مطلوب نیست مثلاً

سرعت آن پایین است یا از لحاظ Security دارای مشکل است

نیازها جهت ایجاد سیستم مور نظر ما عبارت بوده از فراهم آورد امکان

دسترسی سریع توسط کاربران به بخش نامه های صادر شده. و فراهم

آوردن قابلیت ثبت نام سهل و آسان کاربران از طریق صفحات وب و

همچنین امکان گرفتن Query های لحظه ای در ارتباط با تعداد کاربران

ومشخصات آنها که خود باعث افزایش سرعت و کیفیت در برنامه ریزی و

مدیریت سیستم می شود.

امکان سنجی:

برای طراحی و پیاده سازی این سیستم به نرم افزارهای زیر نیاز داریم. و همچنین سخت افزار نیز باید از لحاظ ظرفیت هارد و میزان RAM و سرعت CPU به گونه ای باشد که بتواند نرم افزارهای زیر را ساپورت کند

### نرم افزارها

1-Windows 2000 Server یا Windows NT Server

و یا استفاده از personal Web Server در Windows98

(۱) Internet Explorer 5.0 یا بالاتر

(۲) SQL Server 2000

(۳) Visual Interdev

امکانات سیستم طراحی شده:

سیستم ایجاد شده قابلیت های زیر را تحت وب دارا می باشد

ثبت نام از یک User جدید با دریافت مشخصات آن از طریق صفحه وب

صدور بخش نامه برای واحدهای مختلف سیستم

امکان دیدن و مطالعه بخش نامه های هر دو واحد توسط کاربران همان

واحد

دریافت پاسخ برای Quarry مختلف در ارتباط با کاربران ثبت نام شده در

سیستم

ERD(توصیف موجودیتهای داده‌ای)

جهت طراحی این سیستم به چند table نیاز داریم تا اطلاعات مورد نیاز

را در خود نگهداری کنند در اینجا ما سه table داریم:

(1) table – user : این table شامل فیلدهای زیر است و اطلاعات مربوط

به کاربران سیستم را در خود نگهداری می کند ضمن اینکه اطلاعات و

مشخصات کاربر جدیدی که از طریق صفحه وب ثبت نام می کند نیز در

این table قرار می گیرد:

فیلدهای آن عبارتند از:

User ID : که با وارد شدن هر رکوردی یک User ID منحصر به فرد

برای آن رکورد توسط خود SQL Server در نظر گرفته می شود.

این فیلد Primery Key این table می باشد.

PassWord : توسط خود User در صفحه وب وارد شده و در جدول

ذخیره می گردد.

Name : نام کاربر Family : نام خانوادگی کاربر، age : سن کاربر،

Favoritecolor رنگ مورد علاقه کاربر که تمام این اطلاعات توسط

کاربر در صفحه وب وارد شده و در این Table ذخیره میشود.

همچنین فیلد Code - unit برای اینکه مشخص شود این کاربر در کدام

واحد مشغول است.

(۲) Table - Unit : این table اطلاعات مربوط به هر واحد و بخشی از

این سیستم را در خود نگهداری می کند و شامل فیلدهای زیر است.

Unit - Code که واحد در این فیلد قرار می گیرد ضمن اینکه این فیلد

Primery Key این table نیز هست

Description: نام و عنوان واحد در این فیلد قرار دارد.

(۳) table Letter : این table مربوط به بخش نامه های سازمان می باشد.

که ضمن نگهداری اطلاعات مربوط به بخش نامه باید مشخص شود که هر

بخش نامه مربوط به کدام واحد سازمان است. فیلدهای این table توسط

فرد صادر کننده بخش نامه از طریق صفحه وب پر می شود، این فیلدها

عبارتند از:

Letter - Code : که بخش نامه از Primery این table است.

Subject: موضوع و عنوان بخش نامه

Description: توضیحات و متن بخش نامه

Unit-Code کد واحدی که این بخش نامه برای آن صادر شده

از آنجائیکه گاهی بخش نامه هایی صادر می شود که مربوط به یک واحد

خاص نیست و کلیه واحدهای سازمان باید آن را دریافت کنند.

یک فیلد دیگر نیز تحت نام (Permit) برای این table قرار دهیم که می

تواند محتوی مقدار صفر یا یک باشد که در صورت ۱ بودن آن یعنی این

بخش نامه عمومی است و تمام واحدها باید آن را دریافت کنند.

این سه table Letter , Unit , User در یک data base به نام

tblaccount در SQL Server قرار دارند.

برای مشخص شدن relation بین این سه table نمودار LDS زیر را

خواهیم داشت:

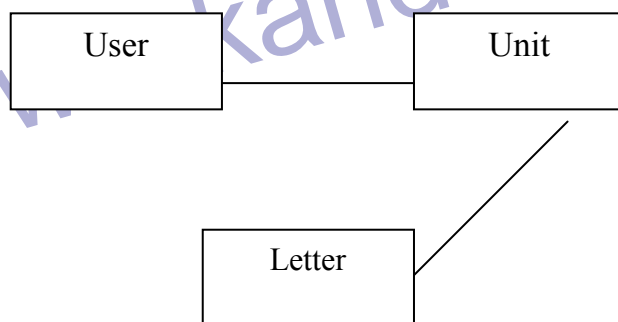
موجودیت ها عبارتند از: Letter , Unit , User

هر واحد چند user دار اما هر User فقط مربوط به یک واحد است. و

فقط در یک واحد می تواند مشغول باشد پس ارتباط User و Unit یک

ارتباط یک به چند است:

برای هر واحد چندین بخش نامه صادر می شود. اما یک بخش نامه بخصوص برای چند واحد فرستاده نمی شود بلکه فقط به یک واحد مربوط به خود می رود پس ارتباط بین Letter , Unit نیز یک ارتباط یک به چند است.



همانطور که در نمودار مشاهده میشود این ارتباطات یک به چند هستند و دیگر نیازی به نرمال سازی نیست. ضمن اینکه User و Letter نیز هیچ ارتباط مستقیمی با یکدیگر ندارند و رابطه آنها از طریق Unit می باشد. تعریف کار انجام شده

برای تشکیل یک دیتابیس جدید و قرار دادن این سه table در آن، در SQL Server از قسمت Enterprise manager استفاده کرده و از طریق New database و Newtable این جداول را می سازیم با همان فیلدهای



ذکر شده در ضمن جدول User دارای یک فیلد data نیز می باشد که این فیلد توسط کاربر پر نمی شود بلکه در صفحه وب تاریخ جاری از سیستم گرفته می شود و در Table ذخیره می شود.

برای کار با SQL Server باید بتوانیم یک UserID و Password برای خود در جدول سیستم تعریف کنیم یک رکورد از پیش تعریف شده در این جدول وجود دارد که Userid آن sa می باشد. ما می توانیم با استفاده از دستورات SQL رکورد دیگری را به این جدول اضافه کنیم که در این صورت Userid , Passord مورد نظر خود را وارد می کنیم. البته در ابتدا محدودیت هایی وجود دارد که می توانیم با استفاده از دستور grant و ... مجوزهایی مثل Create table , Edit table و ... را برای کاربر جدید قرار دهیم.

پس از اینکه این table ها ساخته شدند. با استفاده از Visaul Interdev و استفاده از دستورات برنامه نویسی HTML و asp صفحات وب مورد نظر طراحی شد:

و نحوه استفاده از سیستم Page Format, Interface

اولین صفحه ای که کاربر وارد آن می شود صفحه [home.htm](http://home.htm) است.

که این صفحه فقط دارای چند Link به صفحات دیگر می باشد.

(۱) ثبت نام: با کلیک بر روی این link کاربر جهت ثبت نام به عنوان یک کاربر جدید وارد صفحه First.htm می شود که در آن از طریق یک فرم مشخصات خود و کد واحدی که در آن مشغول خواهد شد را وارد می کند. و با زدن دکمه تایید فرم به صفحه SeCond.asp ارسال می شود که این صفحه عمل انتقال اطلاعات را به User – table به عهده دارد. که برای این منظور لازم است یک Connection از این صفحه asp به دیتابیس مورد نظر ایجاد شود و قبل از آن ما باید یک منبع ایجاد کنیم.

مزیت ایجاد یک منبع داده فایل این است که اطلاعات ارتباطی در یک فایل واقعی ذخیره می شوند و بیش از یک کاربرد به این فایل دسترسی خواهند داشت. قبل از ایجاد یک منبع اطلاعاتی جدید یک سرویس دهنده SQL فعال باشد. از SQL servis manager در برنامه SQL server برای این منظور استفاده می کنیم. برای ایجاد یک منبع داده فایل مراحل زیر را انجام می دهیم:

(۱) control panel

(۲) ODBC

۳) Ado، با کلیک بر روی آن جعبه محاوره Create new data source

باز می شود.

۴) انتخاب نام و توضیح برای منبع فایل جدید

۵) انتخاب نام پایگاه داده ای که جداولمان روی آن قرار دارد.

۶) تست منبع داده فایل ایجاد شده.

پس از ایجاد منبع داده فایل لازم است یک Connection جهت برقراری

ارتباط بین صفحات asp با پایگاه داده ایجاد کنیم.

برای این کار از نرم افزار Visual Interdev استفاده می کنیم که پس از

اجرای new project و ایجاد یک project جدید و انتقال فایل های

مربوطه به این project با کلیک راست بر روی پروژه و انتخاب گزینه

Add new Connection یک Connection جدید ایجاد می کنیم که پس

از انجام این کار اطلاعات مربوط به آن از جمله نام منبع داده فایل

ایجاد شده به طور اتوماتیک در فایل global.asa قرار می گیرد. چون

اطلاعات موجود در این فایل در متغیر Application ذخیره شده

بنابراین از طریق تمام صفحات وب موجود در آن پروژه ایجاد شده قابل

شناسای می باشد. بنابراین می توانیم با نوشتن دو دستور در تمام صفحات

asp ارتباط تمامی آنها را با پایگاه داده مورد نظر برقرار نمائیم:

```
set cn=server.Createobject (AdoDB.Connetion)
```

```
set cn=open Application (Connection 1- Connection string)
```

1-Connection نام Connection ای است که توسط Interdev ایجاد

کرده ایم.

پس از ارسال موفقیت آمیز اطلاعات پر شده توسط کاربر در فرم

به user,table کاربر صفحه ای به رنگ دلخواه خود پیامی در ارتباط با

انجام موفقیت آمیز ثبت نام دریافت می کند.

اما اگر کاربری قبلا در سیستم ثبت نام شده باشد. با ورود به اولین صفحه

نیازی ندارد که Link ثبت نام را کلیک کند و برای او Link های دیگری

وجود دارد. از جمله Link، Query که با کلیک بر روی آن کاربر وارد

صفحه Qury.htm می شود در این صفحه لیست Query ها قرار دارد که با

کلیک بر روی Query مورد نظر خود می تواند پاسخ آن را در صفحه

دیگری که برایش بازی می شود مشاهده کند Query عبارتند از:

۱) تعداد کاربرانی که تا بحال در سیستم ثبت نام کرده اند  $\Leftarrow$  پاسخ در

صفحه ای تحت نام Number. Asp به نمایش در می آید.

۲) تعداد کاربرانی که فیلد name یا Favoritecolor را خالی گذاشته

اند  $\Leftarrow$  پاسخ در صفحه ای به نام Field.asp به نمایش در می آید.

۳) مشخصات کاربرانی که نام خانوادگی آنه به «ی» ختم می شود  $\Leftarrow$

پاسخ در صفحه Family.asp به نمایش در می آید که در این صفحه

تمام اطلاعات مربوط به این کاربرانی در یک table نشان داده

می شود.

۴) مشخصات کاربرانی که در یک محدوده سنی خاص قرار دارند  $\Leftarrow$  این

Link به صفحه age1.htm می باشد که در این صفحه توسط دو text

box و یک کلید (submit) محدوده سنی به صورت از سن  تا سن

وارد می شود. این اطلاعات توسط کلید تایید به صفحه age2.asp

ارسال می شود. و در نهایت پاسخ این Quarry به صورت یک table

حاوی تمام اطلاعات افرادی که در این محدوده سنی قرار دارند به

نمایش در می آید.

۵) مشخصات کاربرانی که در یک محدوده زمانی خاص ثبت نام کرده

اند ← این Link هم مثل age می باشد فقط در اینجا صفحه date

1.htm را داریم که محدوده زمانی را وارد می کنیم و بعد پاسخ در

صفحه date2.asp به نمایش در می آید.

Link دیگری که در صفحه home موجود است برای فردی است که بخش

نامه ها را صادر می کند تحت نام صدور بخش نامه. با کلیک بر روی آن

صفحه ای باز می شوئد که در آن کد بخش نامه، عنوان و متن بخش نامه

همچنین کد واحدی که این بخش نامه مربوط به آن است و نوع بخش

نامه که عمومی یا خصوصی است وارد می شود و باز زدن کلید تایید این

اطلاعات توسط یک صفحه asp به جدول letter ارسال می شود.

(Letter.htm , Letter .asp)

و آخرین Link که در صفحه home وجود دارد مربوط به خواندن بخش

نامه هاست. با کلیک بر روی این Link صفحه ای باز می شود که در آن

لازم است user ای که می خواهد بخش نامه ها را بخواند. Unicode

خود را وارد کند. بعد با زدن کلید تایید بخش نامه های مربوط به همان

واحد بر روی صفحه لیست می شود.

## نوشتن دستورات ساده SELECT

دستور Select سه جزء اصلی SELECT، FROM و WHERE دارد.

نگارش پایه آن عبارت است از:

SELECT column – list

FROM table- list

WHERE search – criteria

بخش SELECT ستونهای مورد نظر برای بازیابی را مشخص می کند.

عبارت FROM نام جدولی است که ستونها از آن بازیابی می شوند و

عبارت WHERE تعداد سطرهای بازگشتی از پرس و جو را مشخص می

کند.

SELECT [ All | DISTINGCT ] [ Top n [PENSEVT] [ WITH  
TIEST ]]

Select – list

[ INTO new – table ]

[ FROME table – sources]

[ HERE search – condition ]

[ GROUP – BY [ ALL] group – by – expression [,...n]

[ WITH { CUBE | ROLLUP } ] ]

[ HAVING search – condition ]

[ ORDER BY { colum – name [ ASC | DESC ] } [ ,...n ]

[ COMPUTE { { AVG | COUNT | MAX | MIN | SUM }  
(expression) }

[ ,...n ] [ BY expression [ ,...n ] ]

[ FOR BROWSE ] [ OPTION (query – hint [ ,...n ]

**دستور SELECT \* FROM table – name ابتدایی ترین نوع پرس و**

**جواست. بکارگیری علامت (\*) به عنوان colum – list کلیه ستونها**

**جدول را بازیابی می کند در داخل SQL Server این علامت به فهرست**

**ستونهای جدول تبدیل می شود.**

**برای بازیابی ستونهای مشخص لازم است ستونها با ( ) از یکدیگر جدا**

**شوند اما پس از آخرین ستون نیازی به کاما نیست.**

**هنگام اجرای پرس و جوی \* SELECT ترتیب ستونها مشابه ترتیب آنها**

**در دستور TABLE CREATE است اما وقتی ستونهای خاصی از جدول**

**را انتخاب می کنید این ترتیب نباید مشابه ترتیب ایجاد آن باشد. شما می**



توانید با تغییر ترتیب ستونها در column – list ترتیب آنها در خروجی

پرس و جو را تغییر دهید.

**تغییر عنوان ستونها**

هنگام نمایش نتایج پرس و جو، عنوان ستونها مشابه اسامی آنها در

column – list است اما می توانید جای آنها اسامی خواناتری بکار برد.

از طریق بکارگیری اسامی مستعار در عنوان ستونها انتخاب می شود.

برای ایجاد اسامی مستعار در عناونی ستونها هر دو نگارش SQL Server

ANSI SQL, می روند. دو روش برای این کار وجود دارد.

SELECT colum – name ‘column – heading ‘

FROM table – name

**توجه:**

اگر نام مستعار بکار رفته دارای فاصله است و یا یکی از کلمات کلیدی

SQL server است آن را مابین علائم “ یا [] قرار دهید. به مثال زیر

**توجه کنید.:**

Select Iname As ‘Last Name ‘ , fname As [First Name]

FROM employee

مثال زیر یک کلمه کلیدی SQL را به کار می برد:

```
SELECT 'Count' = Count (*)
```

```
FROM employee
```

با دستورات زیر می توانید پرس و جوی خود را باز نویسی کنید:

نگارش SQL Server 2000

```
SELECT Employee |D= emp - id, Last Name = Lnme,  
FirsName = Fnma
```

SQL AMSI نگارش

```
SELECT emp - id As Employeed , Lname As Last Name,  
fname As FistName
```

توجه:

کلمه کلیدی As ضروری نیست. برای مثال دستور زیر اطلاعات مشابه

پرس و جو قبل را باز می گرداند:

```
SELECT emp - id Employeed, Lname LasName, fame  
firstName
```

بکار گیری حروف (Literals)

واژه جدید: برای خوانایی بیشتر خروجی پرس و جو می توانید از لیترها

استفاده کنید. یک لیترال رشته ای است مالین علائم ' یا " محصور

شده در list column- قرار می گیرد و در خروجی پرس و جو نمایش

داده می شود. در نتیجه یک لیبِل در ستون مجاور ستون نتایج نمایش

داده میشود.

نگارش بکارگیری لیترال ها به شکل زیر است:

```
SELECT 'literal' [, 'literal' ...]
```

**پردازش داده ها (Manipulating Data)**

شما می توانید با پردازش نتایج پرس و جوها، ستونهای جدیدی را

برای نمایش مقادیر محاسباتی، مقادیر رشته ای جدید، تبدیلات تاریخ و

... تولید کنید. اینکار با عملگرهای (Operators) حسابی، توابع

ریاضی. توابع رشته ای، توابع تاریخ و توابع سیستم انجام می شود.

همچنین می توانید تابع CONVERT را در تبدیل یک نوع داده به نوع

دیگری برای پردازش ساده تر بکار برید.

**عملگرهای حسابی (Arithmetic Operators)**

bingint, money , real , عملگرهای حسابی بر روی انواع داده

smallmoney یا float, decimal , nameric , tintnt , smalint , int

بکار می رود.

ترتیب تقدیم عملگرها (Operator Precednce)

واژه جدید: در عملیات حسابی دو سطح تقدیم در ارتباط با تقدیم نوع

داده و تقدیم عملگر وجود دارد.

تقدیم نوع داده زمانی به کار می رود که عملیات حسابی بر روی انواع

داده متفاوت انجام می شود. در این حالت نوع داده کوچکتر به نوع

داده بزرگتر تبدیل می شود. برای مثال اگر یک Smalint را در یک int

ضرب کنید int خواهد بود. تنها استثناء موجود در مورد نوع money

است که نتیجه همواره از نوع money خواهد بود.

تقدیم عملگر زمانی در صورت وجود چندین عملگر به کار می رود. این

ترتیب هنگام خواندن از چپ به راست به ترتیب مدول، ضرب و تقسیم و

بالاخره جمع و تفریق می باشد.

مشابه سایر عملیات حسابی می توان ترتیب تقدم عملگر را با پرانتز تغییر

داد. در این صورت داخلی ترین عبارت ابتدا محاسبه می شوند. همچنین

بکارگترین پرانتز خوانایی عبارات محاسباتی را بیشتر می کند. برای مثال

در:

$5+5*5=30$  ابتدا ضرب انجام می شود، اما در  $30=(5+5)*5$  عبارت

داخل پرانتز ابتدا محاسبه می شود عموماً با قرار دادن عملیات حسابی

در پرانتز، خوانایی و وضوح کد SQL – Tradsact را بهبود می بخشد.

توابع ریاضی (Mathematical Functions)

توابع داده ریاضی امکان عملیات رایج بر روی داده های ریاضی را

فراهم می آورند. با نگارش زیر می توانید داده های ریاضی را

بازگردانید:

SELECT function – name (parameters)

## جدول ۲-۱۰ توابع ریاضی

تابع	نتیجه
ABS (numeric – expr)	مقدار مطلق
ACOS ASIN  ATAN(float – expr)	سینوس، کسینوس و تانژانت زاویه به رادیان با مقدار شناور
COS SIN COT	سینوس، کسینوس و تانژانت زاویه به رادیان
TAN(float – expr)	
FLOOR (numeric – expr)	بزرگترین عدد صحیح کوچکتر یا مساوی مشخص شده
LOG(float – expr )	لگاریتم طبیعی
LOG10(float – expr)	لگاریتم بر مبنای ۱۰

عدد ثابت ۳/۱۴۱۵۹۲۶۵۳۵۸۹۹۳

PI()

مقدار پارامتر به توان  $y$  POWER (numeric – expr,  
 $y$ )

تبدیل درجه به رادیان RADIANS (numeric –  
expr)

گرد کردن عدد به طول مشخص شده ROUND(numeric – expr)  
len

تابع	نتیجه
------	-------

مربع مقدار مشخص شده SQUIRE((float – expr)

جذر مقدار مشخص شده SQR (float – expr)

نکته:

هنگام استفاده از توابع ریاضی با انواع داده پولی، همواره قبل از داده

ها علامت دلار (\$) قرار دهید در غیر این صورت مقدر مشخص شده نوع

داده numeric مبنای ۴ در نظر گرفته می شود.

هشدار:

هنگام کار با نوع داده float بعضاً نتایج غیر قابل انتظاری از Query Analyzer یا سایر ابزار به دست می آید. برای مثال دستور SELECT ROUND (12.345678E+5,2) خواهد بود. این نتیجه کار Open database connectivity (ODBC) است. SQL Server کماکان گرد کردن را ادامه می دهد. اما بنا به تعریف داده های نوع float مبهم هستند، در حالت کلی بهتر است از این داده اجتناب کنید.

### توابع رشته ای ((String Functions))

هنگام کار با اطلاعات کاراکتری، توابع رشته ای زیادی برای پردازش وجود دارد. اکثر توابع رشته ای بر داده های نوع varchar , nchar , char و nvarchar کار می کنند بنابراین لازم است سایر انواع داده ای را قبل از کار تبدیل کرد. داده های کاراکتری با تابع زیر بازگردانده می شوند:

SELECT fuction – name //9parametrs)



## توابع تاریخ

پردازش مقادیر datetime با توابع data انجام می شود. این توابع در

Column – list عبارت WHERE یا هر عبارت دیگر قابل استفاده اند.

برای این منظور، نگارش زیر را به کار برید:

SELECT data – function (parameters)

مقادیر پارامتر datetime را بین علائم ‘ ‘ یا “ “ قرار دهید. برخی توابع

پارامتری به عنوان derepart دریافت می کنند.

توابع سیستم (System Functions)

برای دریافت در Column – list عبارت WHERE یا سایر عبارت

کاربرد دارند.

جدول ۹-۱۰ توابع سیستم

تابع	نتیجه
COL – NAME (table – id , column – id)	نام ستون را بر میگرداند
Column – name)COL – LENGTH (‘table – name’ c	طول ستون را بر می گرداند

شناسه پایگاه داده را بر می  
گرداند DB – ID ([‘database – name’])

نام پایگاه داده را بر می گرداند DB – NAME ( [database – id])

### عملگر های مقایسه ای (comparison Operators)

عملگرهای مقایسه ای در پیاده سازی شرایط جستجو کاربرد دارند.

سطرهای خاص با مقایسه مقادیر یک ستون یا عبارت یا مقدار خاص انجام

می شود. این عبارات شامل مقادیر ثابت نام ستونهای دیگر، توابع یا زیر

پرس و جوهای متداخل (nested subqueries) می باشند. اگر دو نوع

داده های متفاوت را با هم مقایسه کنیم (برای مثال char و varchar) و یا

نوع داده تاریخ را مقایسه می کنید. (samlldatetime , datatime) ،

حتماً آنها را مابین علائم ‘ ‘ قرار دهید. علائم “ “ نیز قابل قبول است اما

نوع اول با استاندارد ANSI تطابق دارد.

**نگارش عبارت WHERE با عملگرهای مقایسه ای به شرح زیر است:**

SELECT Column – list

FROM table – list

WHERE Column – name Comparison – operator expression

**پرس و جوی زیر شماره کارمندی، نام و نام خانوادگی کلیه کارمندان**

**استخدام شده توسط ناشر کد 0877 را بر می گرداند.**

SELECT emp – id, Iname , fame

FROM employee

WHERE pub – id = ‘877’

**نکته:**

**اگر عملگرهای حسابی و منطقی را با هم به کار می برید توجه کنید**

**عملگرهای حسابی نخست پردازش می شوند اما با استفاده از پرانتز می**

**توانید ترتیب تقدم را تغییر دهید.**

**دامنه ها Ranges**

**با کاربرد کلمه کلیدی BETWEEN می توانید سطرهایی را بر اساس**

**مقادیر یک دامنه بازیابی کنید. مشابه عملگرهای مقایسه ای لازم است تا**

**داده های نوع کاراکتری یا تاریخ را مابین علائم ‘ ‘ قرار دهید.**

**نگارش عبارت WHERE در این حالت به شرح زیر است:**

SELECT Column – list

WHERE table – list

WHERE Column – name [Not] BETWEEN expression And  
expression

**پرس و جوی زیر مقادیر شماره کارمندی و نام خانوادگی کلیه کارمندان**

**استخدام شده بین 10/1/92 تا 12/31/92 را بازیابی می کند:**

SELECT Iname, emp – id

FROM employed

WHERE fire – date BETWEEN ‘10/1/92 AND ‘12/31/92’

**در این مثال دو مقدار مشخص شده نیز قابل پذیرش هستند. توجه کنید**

**مقدار کوچکتر باید در ابتدا آورده شود.**

**لیست ها (List)**

**با کلمه کلیدی IN می توانید سطرهای مطابق با مقادیر یک لیست را**

**بازیابی کنید. مطابق معمول داده های نوع کاراکتری و تاریخ را مابین**

**علائم ' ' قرار دهید. نگارش عبارت WHERE در این حالت به شرح زیر**

**است:**

SE:LECT Column – list

FROM table – list

WHERE Column – name [ Not ]IN (valu – list)

**برای کارمندانی که با ناشر کد 0877 یا 9999 کار می کنند و پرس و**

**جوی زیر را اجرا کنید:**

**ورودی خروجی**

SELECT emp – id . Iname , fname

FROM employee

WHERE pub – id IN ('0877', '9999')

**برای بازیابی سطرهایی که در لیست نیستند می توان عملگر نفی**

**(NOT) بکار برید. برای یافتن کلیه کارمندانی که با ناشر کد 0877 یا**

**9999 کار نمی کنند، پرس و جوی زیر را اجرا کنید:**

SELECT emp – id, name, faname

FROM employee

WHERE pub – id NOT IN ('0877', '9999')

**نکته:**

در صورتی که امکان شرایط مثبت بکار برید و از عملگر NOT اجتناب کنید. زیرا بهینه ساز پرس و جو ها همواره شرایط جستجوی منفی را تشخیص نمی دهد. به عبارت دیگر SQL Server باید کار زیادی برای بازگرداندن نتایج با عملگرهای نفی را انجام دهد. پرس و جوی بالا را با AND و DETWEEN باز نویسی کنید.

### رشته های کاراکتری (Character String)

با کلمه کلیدی LIKE می توانید سطرها را بر اساس بخشی از رشته های کاراکتری بازیابی کنید. LIKE با انواع داده ، text ، datetime ، char ، varchar ، nchar ، nvarchar و Smalldatetime به کار می رود. همچنین می توانید با چهار نوع کاراکتر جانشین (WHERE) با کلمه کلیدی LIKE به شرح زیر است:

SELECT Column – list

FROM table – list

WHERE Column – name [NOT] LIKE ‘string’

عملگرهای جانشین در دسترس عبارتند از:

% رشته ای از هیچ یا چند کاراکتر

- یک کاراکتر

[] یک کاراکتر در محدوده خاص

[^] یک کاراکتر خارج از محدوده خاص

هنگام کار حتماً توجه کنید عملگرهای جانشین مابین علائم ' ' قرار داده

شوند.

با اجرای پرس و جوی زیر title و id - title کلیه کتبی که عنوان آنها

شامل Computer است از جدول titles بازیابی می شود.

ورودی

خروجی

```
SELECT title - id , title  
FROM titles  
WHERE title LIKE '%computer %'
```

مقادیر نامشخص (Unknown Values)

مقدار تهی یا Null مشابه رشته حرفی خالی (Blank) و یا صفر در داده های عددی نیست. Null زمانی واقع می شود که فیلد مقدار داده نشود.

به عبارت دیگر Null معادل نامشخص می باشد. در این شرایط مقایسه

Null با صفر، Blank و یا حتی سایر مقادیر Null منفی باشد. بنابراین

چگونه میتوان سطرها را بر اساس مقادیر Null بازیابی کرد؟ اینکار با

تفکیک سطرهایی از جدول که حاوی مقادیر Null هستند و از طریق

کلمات کلیدی IS NULL و IS NOT NUL انجام می شود.

نگارش عبارات WHERE با استفاده از عملگرهای IS NULL و IS

NOT NUL به شرح زیر است:

```
SELECT Column – list
```

```
FROM table – list
```

```
WHERE column – name Is [NOT] NULL
```

برای مثال پرس و جوی زیر را بازیابی کتبی که فروی نداشته اند اجرا

کنید:

```
SELECT title – id
```

```
FROM title
```

```
WHERE ytd- sales IS NULL
```



برعکس پرس و جوی فوق، می توانید کلید کتبی که فروش داشته اند با

پرس و جوی زیر بازیابی کنید:

```
SELECT title – id
```

```
FROM title
```

```
WHERE ytd – sales ISNOI NULL
```

بکارگیری شرایط چند گانه در بازیابی سطرها

تا کنون بازیابی سطرها از طریق مقادیر مشخص، لیستها، دامنه، مقایسه

رشته ای و ... را دیدید. حال بهتر است نگاهی به بازیابی سطرها از

طریق شرایط چندگانه می توانید عملگرهای منطقی (logical)، نفی

(NOT) و (AND) و یا (OR) را به کار برید. AND و OR امکان تلفیق

دو یا چند عبارت را فراهم می آورند. شرط AND در صورت کلیه

شرایط مقدار بر می گرداند در حالیکه OR در صورت صحت هر یک از

شرایط مقدار بر می گرداند.

در استفاده از چندین عملگر منطقی ترتیب تقدم NOT و AND و OR

است. نگارش عبارت HERE در شرایط چندگانه به شرح زیر است:

SELECT column – list

FROM table – list

WHERE [NOT] expression {AND|OR} [NOT] expression

### حذف اطلاعات تکراری (Duplicate)

هنگام انتخاب برخی از اطلاعات ممکن است با سطرهای تکراری مواجه

شوید. برای حذف آنها عبارت DISTINCT در دستور SELECT را به

کار برید. در غیر این صورت کلیه شرایطی که شرط عبارت HERE را

برقرار می کند در نتیجه نمایش داده م شوند.

نگارش عبارت DISTINCT به شرح زیر است:

SELECT DISTINCT column – list

FROM table – name

WHERE Search – condition

DISTING بر روی ترکیب ستون ها در column – list اعمال می شود.

از آنجاییکه مقادیر null تکراری در نظر گرفته می شود بنابراین فقط

یکی از آنها برگردانده می شود.

مرتب سازی داده ها با عبارت ORDER BY

برای مرتب سازی داده ها در دستور SELECT عبارت ORDER BY

بکار برید. نگارش پایه این عبارت به شرح زیر است:

SELECT column –list

FROM table –list

[ORDER By Column – name | column – list – number [ASC |  
DESC]

تعداد ستونهای لیست ORDER BY نامحدود است به شرط آنکه از ۹۰۰

بایت بیشتر نشود. همچنین می توانید نام ستونها یا عدد ترتیبی (ordinal)

آنها را در column – list بکار برید. دو پرس و جوی زیر نتایج یکسانی

را باز می گرداند.

بکارگیری توأم نام و شماره ترتیبی نیز مجاز است. همچنین نوع مرتب

سازی صعودی یا نزولی داده ها نیز قابل تعیین است. مقدار ASC باشد و

ستون مشخص شده حاوی مقدار NULL باشد، سطرهای حاوی NULL

ابتدای لیست قرار می گیرند.

هنگام کار با عبارت ORDER BY، ترتیب دیکشنری است و نسبت به حروف حساس نیست. ترتیب حساس به حروف نتایج بازگشتی را تغییر می دهد زیرا در این حالت حروف بزرگ و کوچک معادل نیستند.

توجه:

ORDER BY با نوع داده text و image بکار نمی رود. فیلدهای نوع text در محل متفاوتی ذخیره نمی شوند و طولشان از ۰ تا ۲۰ گیگا بایت است و SQL Server اجازه مرتب سازی این نوع فیلد را نمی دهد. نوع داده image نیز قابل مرتب سازی نمی باشد. در صورت لزوم میتوانید یک جدول مرتبط حاوی اطلاعات خلاصه شده را از داده های متنی و تصویری ایجاد کنید.

دستور SELECT INTO

این دستور امکان می دهد جدول جدیدی را بر اساس نتایج یک پرس و جو ایجاد کنید. این جدول جدید بر اساس ستونهای مشخص شده در لیست select جداول نامبرده در عبارت FROM و سطرهای منتخب

عبارت WHERE ایجاد می شود. با دستور Select into دو نوع جدول

دائم و موقت می توان ساخت. نگارش این دستور به شرح زیر است:

SELECT column – list

INTO new – table – list

FROM table – list

WHERE search – criteria

در ایجاد جدول دائمی لازم است گزینه SELECT INTO/

BULKCOPY را تنظیم کنید. با به کارگیری دستور SELECT INTO

می توان بدون فرایند (process) بعضی معمول تعریف داده، جدول

جدیدی را ایجاد و داده هایی را که در آن قرار داد. نام جدید تابع

قوانین نام گذاری SQL Server است و باید در سطح پایگاه داده منحصر

بفرد و یکتا باشد. اگر ستون های column – list دستور SELECT بدون

نام باشند (ستونهای حاصل از توابع جمعی) آنگاه ستونهای جدول جدید

نیز بدون نام خواهند بود. این امر دو مشکل بوجود می آورد:

- اسامی ستونها باید منحصر به فرد باشد، بنابراین اگر دو ستون بدون

نامی وجود داشته باشد دستور SELECT INTO شکست می انجامد.

- اگر جدول جدید حاوی ستون بدون نام باشد آنگاه تنها روش بازیابی این ستون بکارگیری دستور Select است.

نظر به دلایل فوق بهتر است اسامی مستعار برای ستونهای مشتق شده (derived) ایجاد شود.

همچنین چون بکارگیری دستور SELECT INTO در فایل تراکنشها ثبت نمی شود لازم است تا بلافاصله از پایگاه داده نسخه پشتیبان تهیه شود.

همچنین می توانید این دستور را در ایجاد جداول موقت به کار برید. دو نوع جدول موقت عبارت است از:

- جدول موقت محلی (local) که تنها در نشست (session) جاری

کاربرد با SQL Server در دسترس است و فضای تخصیصی به آن

بلافاصله پس از خاتمه نشست آزاد می شود. برای ایجاد جدول موقت

محلی قبل از نام جدول علامت # قرار دهید.

- جدول موقت عمومی (global) که در کلیه نشستهای کاربران با SQL

Server در دسترس است و فضای تخصیصی به آن پس از خاتمه

نشست آخرین کاربر با جداول آزاد می شود. برای ایجاد این نوع جدول قبل از آن علامت ## قرار دهید.

جدول موقت در پایگاه داده tempdu قرار می گیرند. مثال زیر جدول موقتی را با نام #tmptitles و با سه فیلد ایجاد می کنند:

```
SELECT title – id, title, price  
INTO # tmp Titles  
From title  
GO  
SELECT * FROME #tmp Titles
```

نکته:

دستور SELECT INTO جدول جدیدی ایجاد می کند. اگر مایلید به

یک جدول موجود سطرهایی اضافه کنید دستورات INSERT یا

INSERT INTO را بکار برید. (این مباحث در درس فردا مطرح می

شوند.)

عملگر UNION

با استفاده از عملگر UNION می توانید نتایج حاصل از دو یا چند پرس

و جو را در یک مجموعه جواب ترکیب کنید. به طور پیش گزیده سطرهای

تکراری حذف می شوند، اما اگر کلمه کلیدی ALL را با این عملگر بکار  
ببرید کلیه سطرها حتی تکراری ها نیز بازگردانده می شوند. نگارش این

عملگر به شرح زیر است:

SELECT column – list [INTO column]

[FROM clause]

[Where clause]

[GROUP BY clause]

{HAVING clause}

SELECT column – list

[From clause]

[WHERE clause]

[GROUP BY clause]

{HAVING clause}

[ORDER BY clause]

[COMPUTE clause]

قوانین عملگر UNION عبارتند از:

• در کلیه column – list ها باید تعداد، ترتیب نوع داده ستون ها

مشابه باشند.



• عبارت INTO در صورت وجود باید حتماً در پرس و جوی نخست باشد.

• عبارات GROUP BY و COMPUTE جهت تعیین ترتیب نتایج

نهایی و یا محاسبه مقادیر خلاصه در انتهای دستور UNION مجاز

هستند.

• اسامی ستونها از list - column نخستین SELECT گرفته می

شود.

## درج داده (INSERT)

دستور پایه INSERT برای درج یک سطر در جدول تغییر جداول یا

درج داده در جدول پایه از طریق دید (view) به کار می رود. با

تغییراتی در این دستور پایه می‌توانید چندین سطر را با انتخاب داده از

سایر جداول یا دیدها و یا با اجرای روال های ذخیره شده یا توابع

درج کنید. در هر یک از حالت ها لازم است تا از ساختار جداولی که

درج آن انجام می شود اطلاع داشته باشید. آگاهی از اطلاعات زیر

مفید می باشد:

- تعداد ستون های جدول

- نوع هر ستون

- نام ستونها برای برخی از دستور INSERT

- محدودیتها و خواص ستونها نظیر Uniqueridentifier , idevntity

نگارش دستور INSERT به شرح زیر است:

```
INSERT [INTO]
```

```
{table – or – view}
```

```
{ { [column – list ] }
```

VALUE

({DEFAULT |  
constant – expression } [,...n]) |

select – statement |

execute – statement } |

DEFULT VALUE

ساده ترین راه برای تعیین تعداد، نام و نوع ستونها، بکارگیری دیالوگ

Table Properties در Enterprise Manager است. برای باز کردن

دیالوگ مراحل زیر را دنبال کنید:

۴- پایگاه داده را از پوشه Databases انتخاب کنید.

۵- جدولی را از پوشه Tables انتخاب کنید.

۶- بر روی نام جدول کلیک راست کنید و Properties را انتخاب

کنید.

یا اجرای روال ذخیره شده سیستم sp – help اطلاعاتی یکسانی به

دست خواهید آورد:

ورودی

USE pubse

Go

EXEc sp – help publishers

Go

شکل ۱-۱۱

دیالوگ table Properties

بخشی از نتایج در زیر آمده است.

خروجی

خروجی به دست آمده نام، نوع و محل قرارگیری هر ستون را نشان می

دهد. این اطلاعات در ساخت دستورات INSERT کاربرد دارند.

همچنین می توانید با دیدهای INFORMATION SCHEMA

اطلاعات مورد نظر را به دست آورید.

ورودی

کاربرد دستور INSERT VALLUES

ساده ترین فرم دستور INSERT نیاز به تعیین مقدار هر ستون مطابق با ترتیب تعریف آنها دارد. این ترتیب در مثالهای قبل نشان داده شده است.

ورودی

اگر مجوز INSERT جدولی را دارا باشید و هیچ قیدی را نقض نکنید، SQL Server پیامی مشابه زیر باز می گرداند:

خروجی

4) row (s) affected

توجه: کنترل‌های شبکه مانند (grid – type Controls) که در ابزارهای

بصری توسط کاربران به کار می روند به پایگاه داده بسته نشده اند و

بطور خودکار داده به پایگاه داده نمی افزایند. بنابراین لازم است با

نوشتن کد، داده های تغییر یافته در شبکه را بررسی کنید و دستورات

INSERT / VALUES مناسب برای افزودن سطرها به پایگاه داده

اجرا کنید.

ساده ترین فرم دستور INSERT نیازمند ارائه مقدار ستون با ترتیب صحیح است. اگر نمی خواهید ترتیب ستونها را رعایت کنید و یا مقدار آنها را مشخص کنید می توانید نوع دیگر دستور INSERT را به کار برید.

دستور INSERT زیر لیستی از اسامی ستونها را قبل از عبارت VALUE مشخص می کند و این لیست تنها شامل زیر مجموعه ای از ستونهای جدول است، بنابراین لیست مقادیری که پس از آن می آید نیز تنها برای همین ستونها به کار می رود.

```
INSERT INTO tmpPublishers (sate, pub – id)
VALUES ('AK' , '9932')
```

**نکته:**

برنامه های کاربردی و پژوهی یا پاوربیلدر هنگام اتصال به پایگاه داده و افزودن داده ها عملاً دستور INSERT / values را به کار می برند.

حال برای ستونهایی که در لیست مشخص نشده اند چه اتفاق می افتد؟  
هنگام درج یک سطر در جدول، همه ستون ها باید مقدار داشته باشند و  
اگر شما این مقدار را مشخص نکنید SQL Server باید بتواند این کار را  
انجام دهد. بنابراین لازم است تا این ستونهایی یکی از شرایط زیر را  
دارا باشند:

- ستون دارای مقدار پیش گزیده متصل به آن باشند.
  - ستون از نوع identify باشد.
  - مقدار null برای ستون مجاز باشد.
- دستور INSERT قبل از هیچ مقداری برای ستونها مشخص نشده  
است. در جدول ستونها مجاز به ذخیره مقادیر null هستند و پس از  
اجرای دستور INSERT می توانید پرس و جو با دستور select را  
اجرا کنید.
- در صورت اجرای دستور INSERT و مشخص نکردن مقدار برای  
ستونها بجز موارد استثنا پیام خطا دریافت خواهید کرد.

توجه داشته باشید پیام خطا کاملاً تکمیل نیست. مشکل نه تنها عدم قابلیت ستون id – pub در نگهداری مقادیر null بلکه عدم وجود مقدار پیش گزیده است. همچنین این ستون از نوع identify نیز نیست.

**مقادیر پیش گزیده (DEFAULT VALUE)**

یکی دیگر از انواع دستور INSERT حالتی است که می خواهید

SQL Server مقادیر پیش گزیده ستونها را در صورت وجود به کار

برد. (این مقادیر پیش گزیده شامل null و identify نیز می شوند.)

برای این منظور لیست مقادیر واقعی کلمه کلیدی DEFAULT را به

کار برید تا SQL Server بتواند مقدار صحیح را مشخص کند مثال زیر

را در نظر بگیرید:

در حالتی که همه ستونها مقدار پیش گزیده قابل استفاده SQL

Server را دارا هستند، دستور INSERT زیر را به کار برید:

INSERT INTO [table – name] DELAULT VALUES

حذف داده (Delete)



با استفاده از دستور Delete می توانید یک یا چند سطر را از جدول

حذف کنید:

DELETE FROM {table – name}

[Where Clause]

توجه:

FROM و WHERE اختیاری می باشند.

دستور زیر کلیدهای سطرهای جدول tmpSales را حذف می کند:

DELETE tmpSales

برای حذف زیر مجموعه ای از سطرهای یک جدول عبارت WHERE را

به کار برید. شرط WHERE میتواند شامل کلیدهای شرایط نظیر عملگرهای

رابطه ای (= و >، <) کلمات کلیدی IN, LIKE و BETWEEN باشد.

مثال زیر کلید کتابهایی که pub – name آنها "New Moon Books"

است را از جدول tmpPublishers حذف می کند:

ورودی

DELETE tmpPublishers

WHERE pub – name = 'New Moon Books'

دستور DELETE و بکارگیری جدول جستجو (Lookup Table)

دستور DELETE می تواند سطرهای یک جدول را حذف کند، اما SQL Server این امکان را فراهم می کند که جدول دیگری به عنوان جستجو برای این دستور مشخص شود. این کار باید اغلب با یک زیر پرس و جو مشخص شود. در مثال بعد کلیه عناوین منتشر شده توسط New Moon Books حذف شود. جدول tmpTitles تغییر خواهد کرد اما این جدول صرفاً کد ناشرین (Publisher ID) را نگهداری می کند و نام ناشرین در آن قرار ندارد. بنابراین باید در جدول Publisher جستجو کرده و کد ناشر New Moon Books را پیدا کنید تا سطرهای قابل حذف از جدول tmpTitles مشخص شوند.

#### ورودی

```
DELETE FROM tpIPublisherList
WHERE strPubName=
(SELECT pub -name FROM tmppublishers
WHERE pub -id ='9956')
```

زیر پرس و جو مثال فوق به جدول tmppublishers دسترسی یافته و یک مقدار مجزا برای pub -name باز می گرداند. سپس این مقدار برای

تعیین سطرهای قابل حذف از list – publisher به کار می رود. برای این

منظور کلیه سطرهای دارای pub – name برابر با مقدار بازگشتی از زیر

پرس و جو تنها یک سطر است زیرا id – pub در جدول publishers کلید

اصلی (Primary key) است. اگر زیر پرس و جو بتواند بیش از یک مقدار

باز گرداند لازم است کلمه کلیدی IN را بجای علامت تساوی (=) به کار

برید. زیر پرس و جو فوق می تواند با دستور تلفیق به نحو مؤثرتری

بازنویسی شود. هنگام به کار گیری برای نام جدول حاوی سطرهای

حذف شونده و دیگری برای جدول تلفیق شونده.

این مثال کلیه سطرهای tmp Sales با نوع کتاب business را حذف می

کند.

```
DELETE tmpSales
```

```
WHERE title – id IN
```

```
(SELECT title – id FROM tmpTitles
```

```
WHERE type = ‘business’
```

زیر پرس و جو فوق لیستی از مقادیر title – id را از جدول tmpTitles

باز می گرداند. این مقادیر برای تعیین سطرهای قابل حذف از جدول

tmpSales با مقدار id – title برابر با یکی از مقادیر فوق به کار می

رود.

با استفاده از SQL – Transact می توانید دستور DELETE جدیدی

بنویسید که در عبارت FROM آنها چندین جدول نام برده شده اند. این

روش دستور DELETE را به شکل JOIN نمایان می سازد گرچه عمل

حذف تنها از یک جدول انجام می شود. این قابلیت همانند به کارگیری

زیر پرس و جوهاست و جدول دوم تنها به عنوان جدول جستجو به کار

می رود.

بکارگیری دستور TRNCARTE TABLE tmpSales

با نحوه حذف تمام سطرهای یک جدول آشنا شدید. اگر مایلید تنها

ساختار جدولی باقیمانده بماند روش زیر را به کار برید:

TURNCURTe TABLE tmpSales

بر خلاف دستور DELETE این دستور پیامی مبنی بر تعداد سطرهای

حذف شده باز نمی گرداند. موارد زیر تفاوتهای دستور DELETE بدون

عبارت WHERE و دستور TURNCURTe TABLE را بیان می کنند:

• DELETE هر سطر حذف شده در فایل گزارش تراکنشها (log)

ثبت می کند در حالیکه TURNCURTe TABLE نگهداری کلیه

ایندکس ها را در یک مرحله انجام می دهد و تنها آزاد سازی صفحه

را ثبت می کند.

• TURNCURTe TABLE تنها توسط دارنده جدول عضوی از

نقش پایگاه داده db – owner و یا نقش مجوزهای مناسب بر روی

جدول به کار می رود.

• تریگر DELETE هنگام اجرای دستور TURNCURTe TABLE

فعال نمی شود اما هنگام حذف سطرها با دستور DELETE انجام می

شود.

• دستور TURNCURTe TABLE مقدار identity را به هسته

(seed) بر می گرداند در حالیکه دستور DELETE تأثیری در مقدار

identity بعدی ندارد.

به روز رسانی داده (UPDATE)

دستور سوم تغییر داده دستور UPDATE است که امکان تغییر

مقادیر ستونها در سطرهای موجود را فراهم می آورد.

بکارگیری دستور UPDATE

نگارش این دستور به شرح زیر است:

UPDATE

{table – name | view – name}

SET

Column – name1 = {expression 1 | NULL | (select – statement)}

[, column – name 2 =...]

[WHERE search – conditions]

عبارت SET ستونهای به روز شونده را مشخص می کند. در این دستور

عبارت WHERE اختیاری است.

دستور زیر مقدار ستون sales – ytd جدول tmpTitles را در هر

سطر به صفر تغییر می دهد. ممکن است بخواهید این مقدار را در

ابتدای هر سال تغییر دهید.

ورودی

خروجی

```
UPDATE tmpTitles
```

```
SET ytd – sales = 0
```

دستور فوق به بدون عبارت WHERE مقدار این ستون در کلیه

سطرها را صفر می کند. مثال زیر ستون city ناشر Algodate

Infosystems را به روز رسانی می کند:

```
UPDATE tmpPublishers
```

```
SET city ‘EL Cerrito’
```

```
Where pub – name = ‘Algodata Infoystem’
```

ورودی

خروجی

با دستور UPDATE می توانید مقدار جدیدی را بر حسب مقدار موجود ستون ایجاد کنید. مثال زیر قیمت کتابهای روانشناسی (psychology) را ده درصد کاهش می دهد:

```
UPDATE tmoTitles
```

```
SET price * 0.92
```

```
WHERE type = 'psychology'
```

دستور UPDATE میتواند بیش از یک ستون را به بروز رسانی کند.

برای این منظور نام ستونها در عبارت SET با کاما یکدیگر جدا کنید.

مثل زیر مبلغ کلیه کتابهای Popular Computing را ۲۰ درصد

افزایش داده ورشته (Price Increase) را به فیلد notes آنها می

افزاید:

```
UPDATE tmpTitles
```

```
SET price * 1.2, notes = notes + '(price increases)'
```

```
WHERE type = 'popular - comp'
```

دستور UPDATE و جدول جستجو

دستور UPDATE می تواند سطرهایی از یک جدول را به

ورزرسانی کند. اما SQLServer این امکان را فراهم می آورد که



جدول دیگری به عنوان جدول جستجو برای آن مشخص شود. این

جدول معمولاً با یک زیر پرس و جو در عبارت WHERE یا SET

ظاهر می شود مشخص می گردد.

همچنین می توانید با استفاده از SQL – Transact دستورات

UPDATE جدیدی بنویسید که در عبارت FROM آنها چندین

جدول به کار رفته باشد. این روش شکل تلفیق به دستور UPDATE

می دهد. اما تنها یک جدول تغییر می یابد.

قابلیت فراهمه شده مشابه زیر پرس و جوهاست و جدول دوم صرفاً به

عنوان جدول جستجو به کار می رود.

انتخاب روش زیر پرس و جو یا روش تلفیق سلیقه شخص کاربر است.

مشابه دستور DELETE روش زیر پرس و جو از نقطه نظر تشخیص

جدولی که تغییر یابد مقدار جدید ستون id – pub و سطرهایی که

تغییر یابند واضحتر است. همچنین توجه داشته باشید روش تلفیق یک

استاندارد غیر ANSI است.

برخی دستورات UPDATE با تلفیق پیچیده تر می شود. یکی مورد

زمانی است که دستور UPDATE در هر دو عبارت WHERE و SET

از زیر پرس و جو استفاده کند.

نخستین مرحله در ایجاد یک وب سایت با استفاده از ویژوال InterDav

ایجاد یک پروژه جدید بستگی به این دارد که شما در حال حاضر برنامه

کابردی ویژوال InterDav را اجرا کرده اید. یا خیر. به طور پیش گزیده

دیالوگ پروژه ساخت نمایش داده می شود. اگر در ویژوال InterDav

هستید از منوی اصلی گزینه های File و New را انتخاب کنید.

به جای دیالوگ File Open دیالوگ پرونده جدید از نوع چند برگی

است و به شما امکان می دهد ایجاد پروژه های جدید و یا باز کردن

پروژه های موجود را از یک دیالوگ انجام دهید.

(شکل زیر را ببینید).

در این مرحله مطمئن شوید که شما **New eb Prohct** از **برگه New** انتخاب شده است. در کنترل ویرایشی **Name** نام **AddrBook** را وارد کنید. توجه کنید کنترل ویرایشی **Location** مطابق با نام کامل پروژه ای که تایپ می کنید به روز در می آید. سپس کلید **Open** را کلیک نمایید. در این لحظه ویزارد پروژه وب را می بینید. ویزارد پروژه وب مجموعه های از دیالوگ هاست که به شما امکان می دهد تنظیمات اولیه برای پروژه خود را تعیین و به پایان برید.

در پاسخ به پرسش می خواهید از چه سرویس دهنده ای استفاده کنید؟ لازم است تا نام یک سرویس دهنده فعال وب را مشخص نمایید. چنانچه ایجاد پروژه خود را بر روی دستگاهی انجام می دهید که سرویس دهنده وب نیز هست می توانید مقدار localhost را مشخص نمایید. توجه داشته باشید که پیشوند http و https را مشخص نکنید.

نکته قابل توجه دیگر مربوط به انتخاب Secure Socket Layer (SSL) است. SSL یک پروتکل امنیتی مربوط به Netscape است که برای رمز گشایی HTTP در هنگام ارسال و دریافت به / از سرویس دهنده وب به کار می رود در این مرحله SSL انتخاب نشده قرار دهید. درخاتمه لازم است تا مواد عملیاتی را انتخاب کنید. اساساً مود محلی به معنای آن است که می خواهید در یک محیط ایزوله کار کنید به طوریکه تغییراتی که در پروژه خود می دهید در سرویس دهنده وب انعکاس نداده باشد. برای این پروژه مود Master را انتخاب کنید و کلید Next را کلیک نمایید.

(شکل زیر ویزارد پروژه وب - مرحله دوم: تعیین وب

مرحله دوم به شما امکان می دهد تا مشخص نمایید این پروژه یک برنامه کاربردی وب کاملاً جدید است و با به یک برنامه موجود اضافه خواهد شد. در حال حاضر مقدار موجود را بدون تغییر به کار برید. این دیالوگ همچنین امکان می دهد تا ایجاد اتوماتیک پرونده `Search . htm` را مشخص نماید. این پرونده جستجوی متنی را روی کلید صفحات مربوط به برنامه وب را برای کاربران آن فراهم می سازد.

در مرحله بعدی ویزارد وب در ارتباط با انتخاب طرح و زمینه برنامه کاربردی است.

پس از آنکه دکمه `Finish` را کلیک کردید، ویژوال `InterDev` فرایند ایجاد پرونده های جدید پروژه شما را آغاز می نماید.

بررسی واسط ویژوال `InterDev` برای یک پروژه

حال که یک پروژه ویژوال InterDew ایجاد کرده اید، لازم است تا بررسی واسط کاربر ویژوال InterDew شروع نماییم.

نوار عنوان (Title Bar)

در هنگام شروع ویژوال InterDew پس از ایجاد یک پروژه جدید نخستین چیزی که احتمالاً به آن توجه خواهید کرد آنست که نوار عنوان مربوط به برنامه ویژوال InterDew تغییر کرده و شامل نام کامل راه حل شماست. این نام قالب Web server/ Solution name و مقادیری که در زمان ایجاد راه حل یا پروژه وارد کرده اید را داراست.

کاوشگر پروژه (Project Explorer)

کاوشگر پروژه پرونده های مربوط به راه حل را کاملاً مشابه ساختار یک

وب سایت و به شکل سلسله مرتبی نمایش می دهد. فهرست پرونده ها

شامل صفحات HTML, ASP, تصاویر، اپلت ها و غیره ایجاد شده برای

هر پروژه می باشد.

این پنجره از آن جهت کاوشگر پروژه نامیده می شود که در حقیقت از

کلیه توابع کاوشگر نظیر کشیدن و رها کردن (Drag & Drop) ایجاد

پوشه های جدید برای سازماندهی پرونده ها و تغییر نا عناصر و منوهای

پدیدار شونده حساس به متن (Context Menus) پشتیبانی م نماید. در این مرحله بر روی پوشه های مختلف کلیک کنید تا ایده ای از محتویات و توابع قابل انجام بر روی هر آیتم بدست آوریم. این نکته را در ذهن داشته باشید که مشابه سایر برنامه های ۳۲ بیتی ویندوز می توان تنها با کلیک کردن دکمه راست موشواره منوی پدیدار شونده حساس به متن که در آن کلیه انتخابهای خاص هر عنصر لیست شده اند را اگر به عنصر دوم در کاوشگر پروژه نگاه کنید و ورودی مربوط به سرویس دهنده وب AddrBook را خواهید دید. این ورودی نشاندهنده ریشه مجازی پروژه شماست (قبلاً توضیح داده شد) و نقطه برنامه است. چنانچه یک راه حل شامل چندین پروژه باشد هر کدام از پروژه ها ریشه مجازی مربوط به خود را تحت نام کامل راه حل خواهند داشت.

بنا به دلایلی، زمانیکه شما یک پروژه ویژوال InterDev را ایجاد می کنید، یک علامت + قبل از نام هر پوشه قرار داده می شود. در حالت عادی این + نشاندهنده ساختار درختواره قابل گسترش برنامه است. در مواردی اگر بر روی بعضی از آنها کلیک خواهید دید که علامت “+” ناپدید می شود بنابراین در قسمت بالای کاوشگر پروژه نوار ابزاری را

شامل یک تا ۵ دکمه بسته به عنصر انتخاب شده فعلی مشاهده خواهید کرد. اگر نام راه حل انتخاب شده باشد، تنها یک دکمه در نوار ابزار وجود خواهد داشت که امکان می دهد خواص مربوط به راه حل را نمایش دهید. اگر آیتم دیگری انتخاب شده باشد دکمه های زیر را