



دستورکار آزمایشگاه پایگاه داده‌ها

استاد:

محمد فرجی مهماندار

فهرست مطالب

جلسه اول : نصب MS SQL Server و Power Designer	۴
جلسه دوم : ایجاد مدل مفهومی و فیزیکی در Power Designer	۳۸
جلسه سوم : آشنایی با نرم افزار Microsoft SQL Server	۴۶
جلسه چهارم : آشنایی با Constraint ها در Microsoft SQL Server	۵۷
جلسه پنجم : آشنایی با View ها و ابزار Query Designer	۶۵

فهرست جداول

جدول ۱-۲ انواع داده‌ای عددی	۳۹
جدول ۲-۲ انواع داده‌ای عددی تقریبی	۴۰
جدول ۳-۲ انواع داده‌ای Date & Time	۴۰
جدول ۴-۲ انواع داده‌ای کاراکتری غیر Unicode	۴۰
جدول ۵-۲ انواع داده‌ای کاراکتری Unicode	۴۱
جدول ۶-۲ انواع داده‌ای باینری	۴۱
جدول ۷-۲ صفات موجودیت TITLES	۴۲
جدول ۸-۲ صفات موجودیت WRITERS	۴۲
جدول ۹-۲ صفات موجودیت FCTRS	۴۲
جدول ۱۰-۲ صفات موجودیت CUSTOMERS	۴۲

جلسه اول : نصب MS SQL Server و Power Designer

۱-۱ هدف:

در اولین جلسه از آزمایشگاه پایگاه داده‌ها قصد داریم ابتدا با نرم‌افزار Microsoft SQL Server 2008 R2 و نیازمندیهای سیستمی آن آشنا شده و سپس مراحل نصب آن را به کمک تصویر تشریح نماییم.

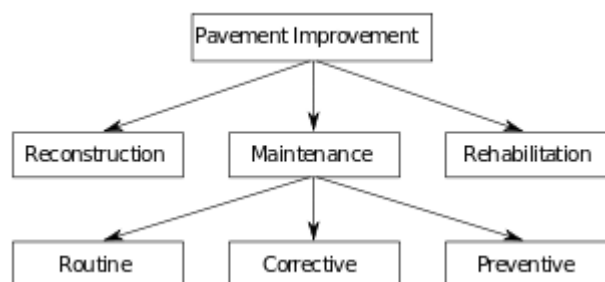
۲-۱ مقدمه:

Microsoft SQL Server یک سیستم مدیریت پایگاه داده رابطه‌ای است که توسط شرکت Microsoft توسعه یافته است و ذخیره، مدیریت و سازماندهی داده‌ها را بر عهده دارد. سازماندهی داده‌ها در پایگاه داده به صورت مدل های زیر می باشد:

۱. مدل سلسله مراتبی
۲. مدل شبکه‌ای
۳. مدل رابطه‌ای

۱. مدل سلسله مراتبی :

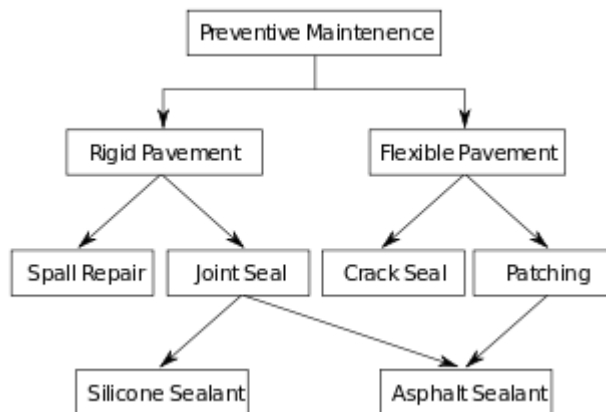
سازماندهی این مدل بصورت درختی است. یعنی هر گره یا Node درخت با گره (Node) پدر خود وابسته است. هر گره می تواند حداکثر یک پدر داشته باشد.



شکل ۱-۱ مثالی از مدل داده سلسله مراتبی

۲. مدل شبکه‌ای :

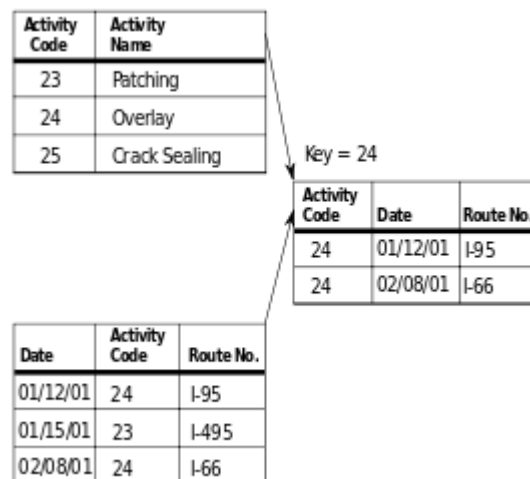
در مدل شبکه‌ای گره‌ها به صورت یک شبکه با یکدیگر در ارتباط هستند.



شکل ۲-۱ مثالی از مدل داده شبکه‌ای

۳. مدل رابطه‌ای :

در این مدل داده‌ها در قالب جداول هستند که هر جدول از تعدادی سطر و ستون تشکیل شده است. زبان‌های SQL و DB2 برای این نوع از پایگاه‌های داده هستند.



شکل ۳-۱ مثالی از مدل داده رابطه‌ای

حال به بررسی معماری‌های نرم‌افزاری به لحاظ لایه‌بندی می‌پردازیم :

۱. تک لایه‌ای (Single Tier) :

نرم‌افزارهایی که مولفه‌های Logic، Presentation و Data Source آن در یک لایه قرار می‌گیرند.

۲. دو لایه‌ای (Two Tier) :

نرم‌افزارهایی هستند که مولفه‌های Logic و Presentation آن در یک لایه و مولفه Data Source آن در لایه‌ای دیگر قرار می‌گیرند.

۳. چند لایه‌ای (Multi Tier) :

هر یک از مولفه‌های Logic، Presentation و Data Source در لایه‌ای جداگانه قرار می‌گیرند.

۱-۳ شرح:

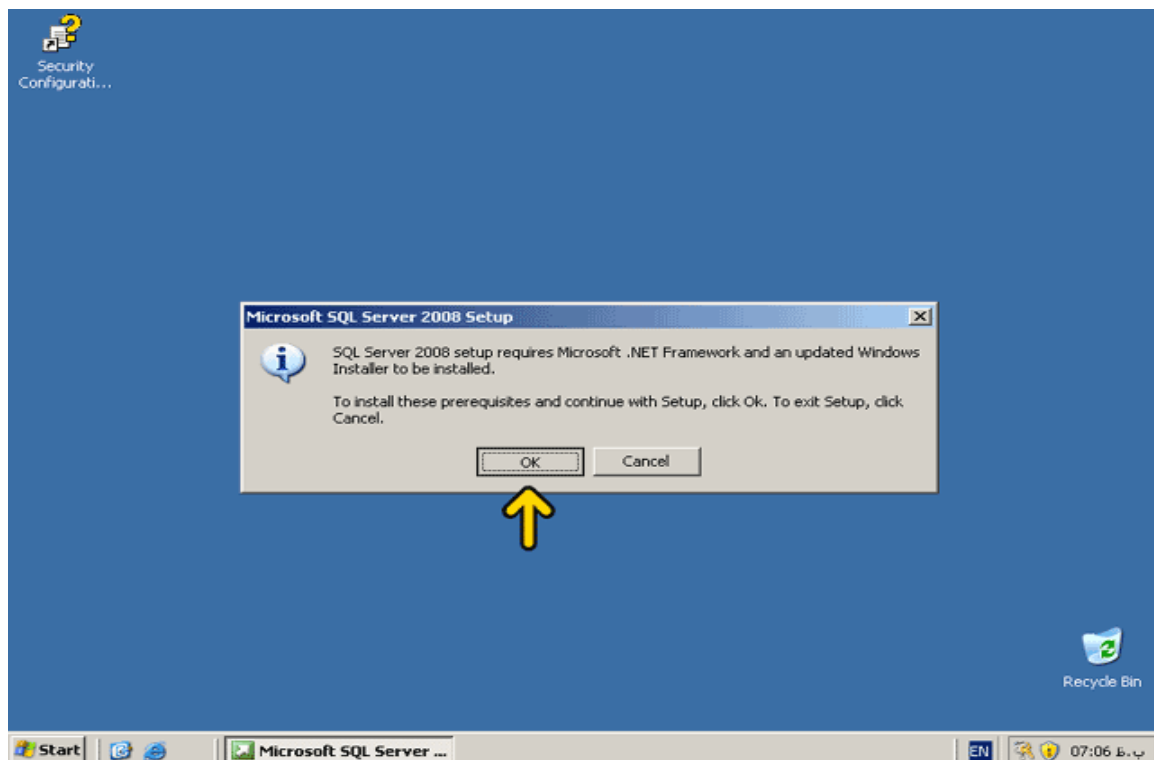
در این جلسه به شرح مراحل نصب نرم‌افزار Microsoft SQL Server 2008 R2 و سپس آموزش نرم‌افزار Power Designer محصول شرکت Sybase می‌پردازیم.

ابتدا به تشریح مرحله به مرحله نصب نرم‌افزار Microsoft SQL Server 2008 R2 می‌پردازیم :

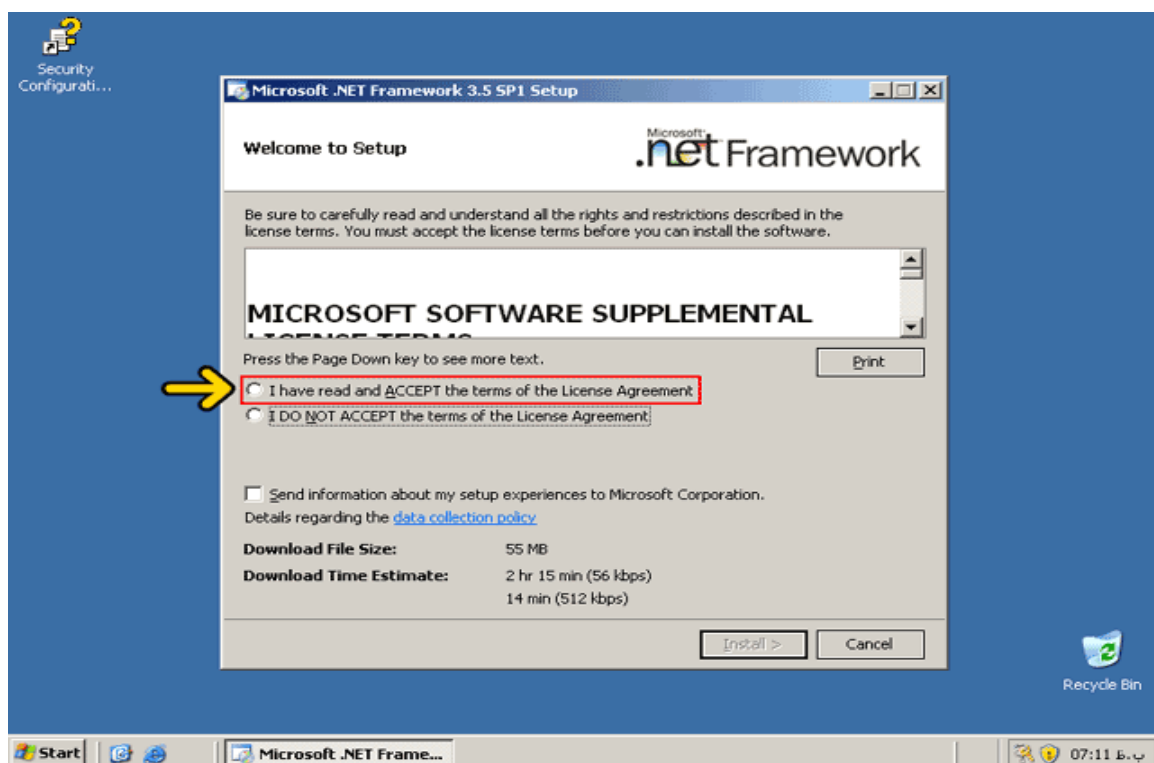
۱. ابتدا بر روی فایل اجرایی نصب برنامه دابل کلیک نمایید.

۲. برای نصب Microsoft SQL Server می‌بایست نرم‌افزارهای Microsoft .Net Framework و Windows Installer بر روی کامپیوتر شما نصب باشند.

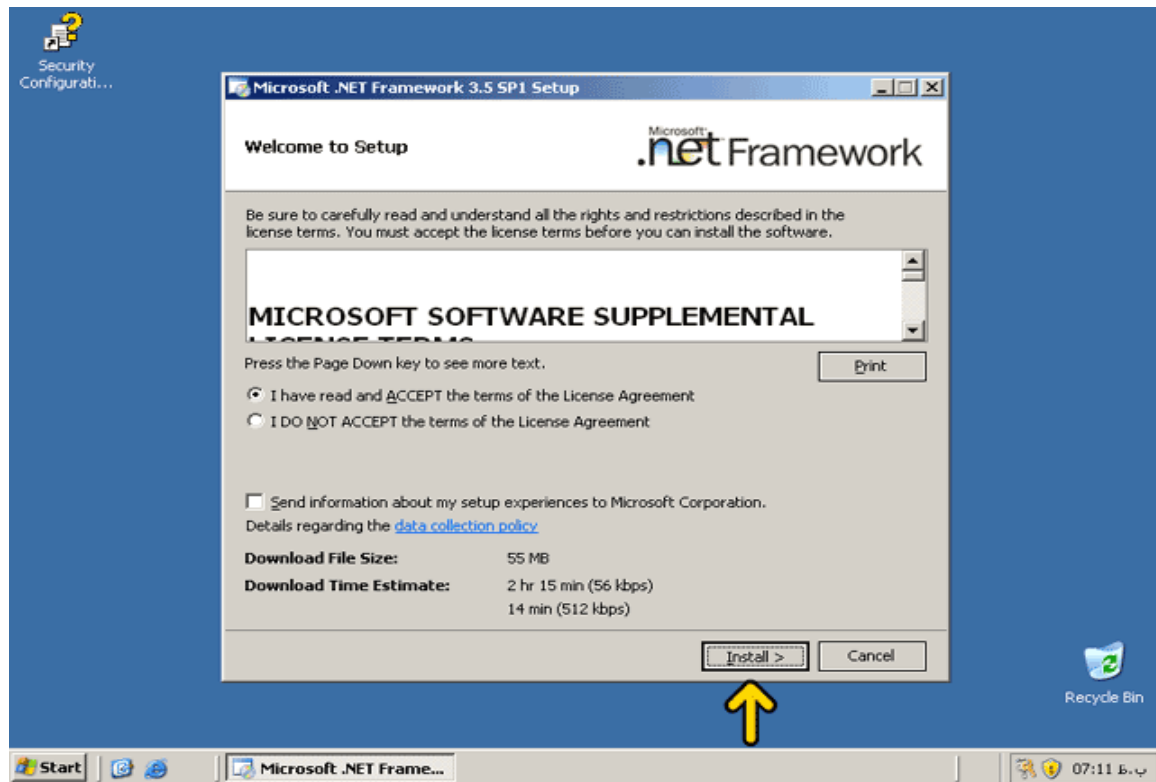
۳. در صورتی که نرم‌افزار Microsoft .Net Framework نصب نباشد ابتدا پنجره نصب این نرم‌افزار باز می‌شود.



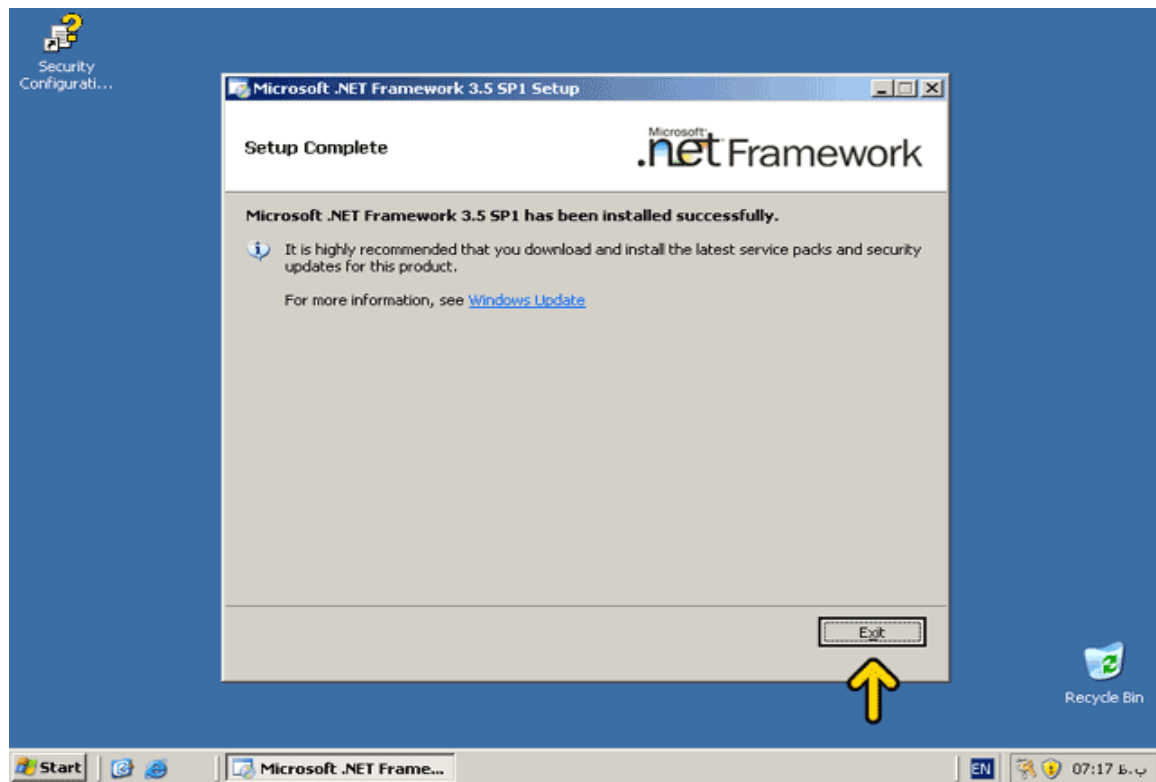
شکل ۴-۱



شکل ۵-۱



شکل ۶-۱



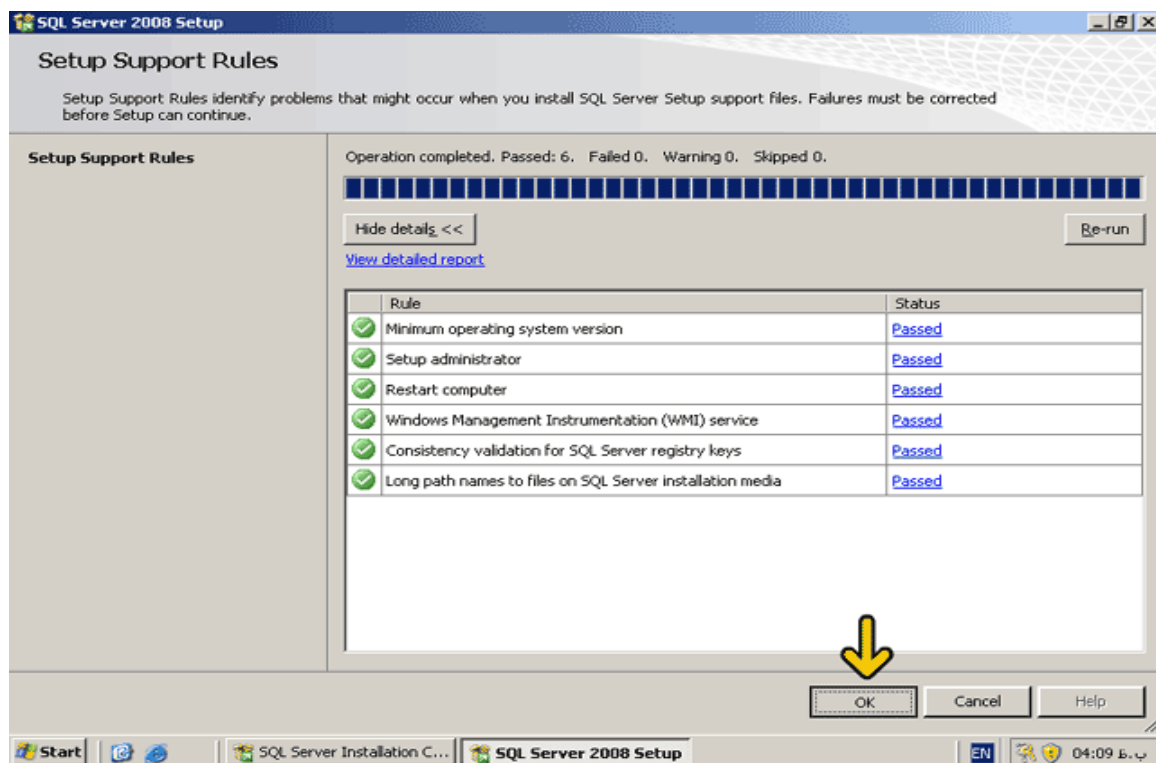
شکل ۷-۱



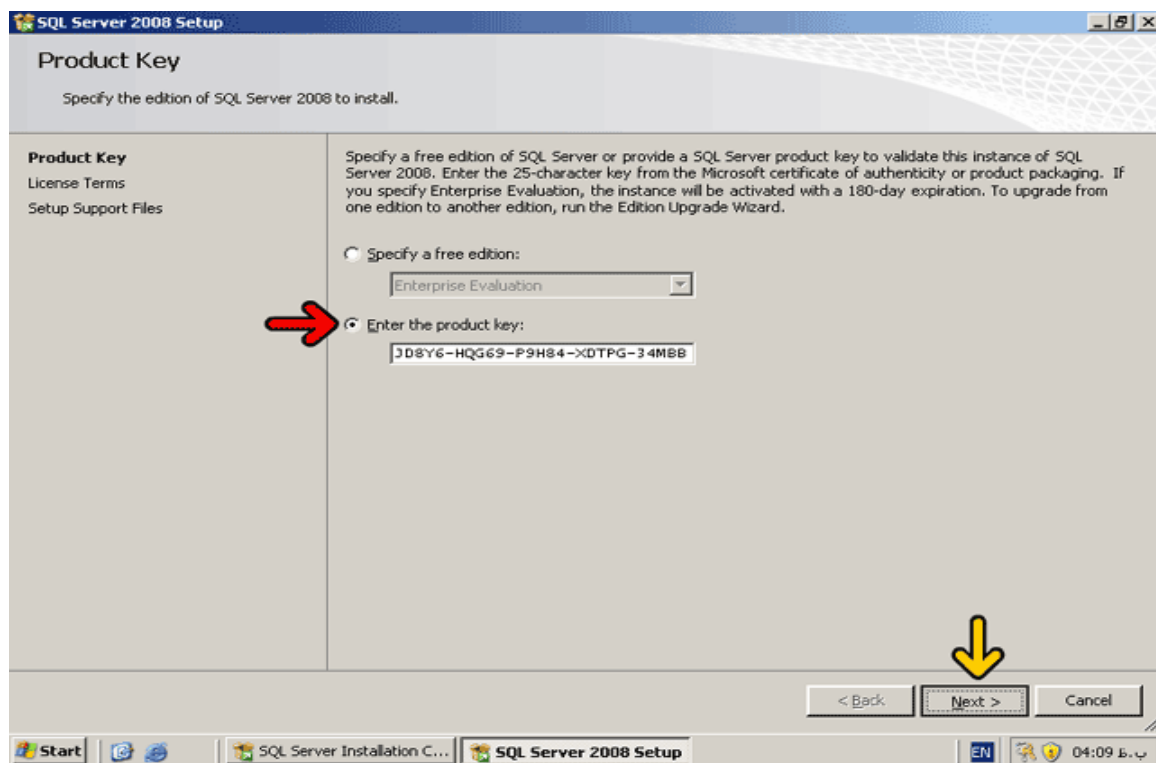
شکل ۸-۱



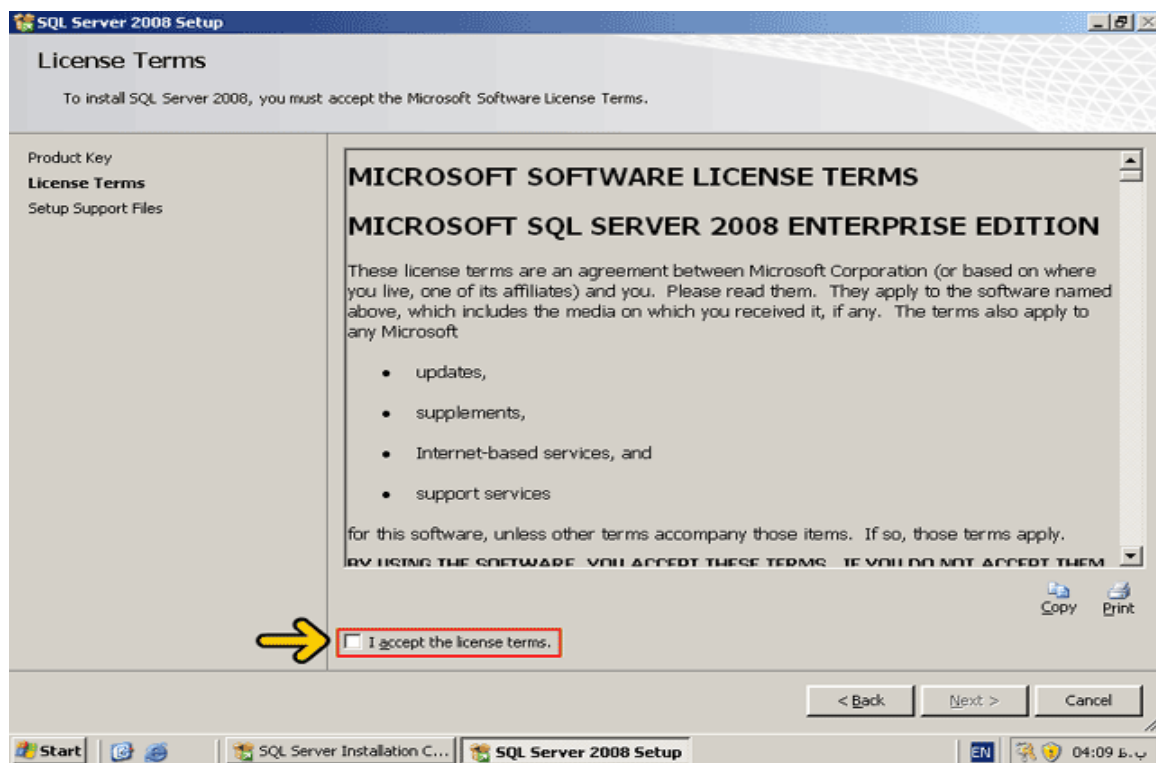
شکل ۹-۱



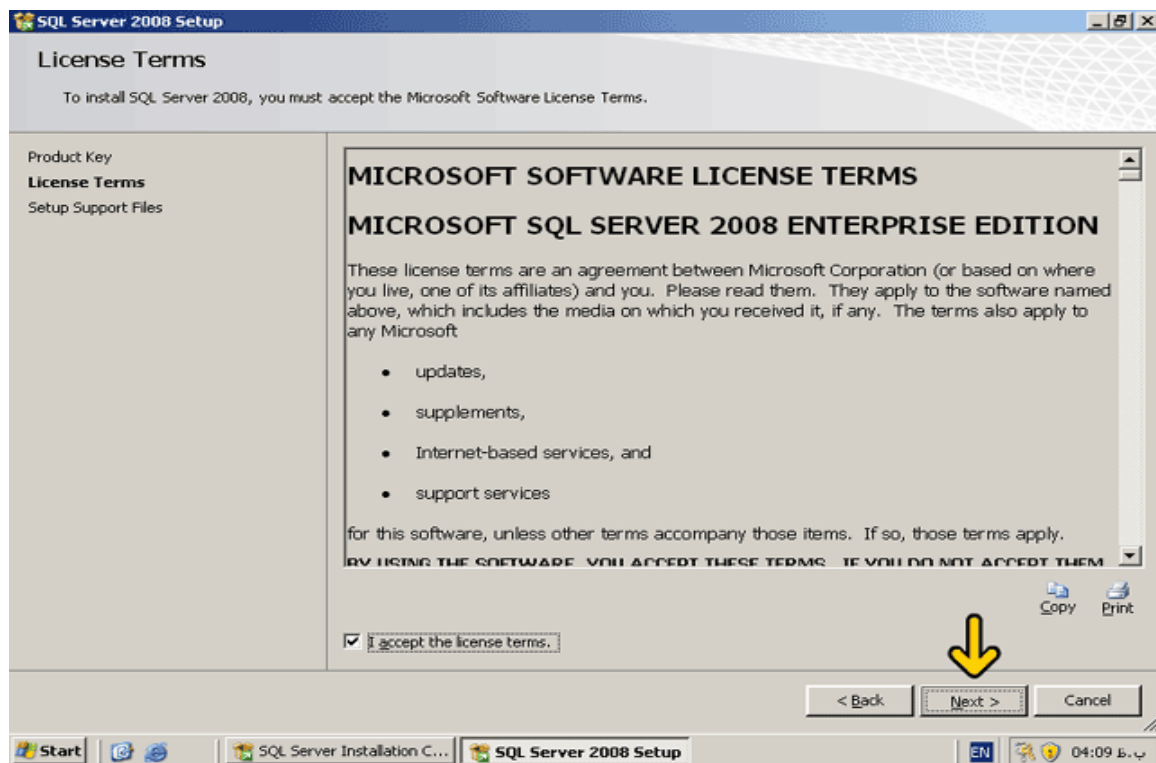
شکل ۱۰-۱



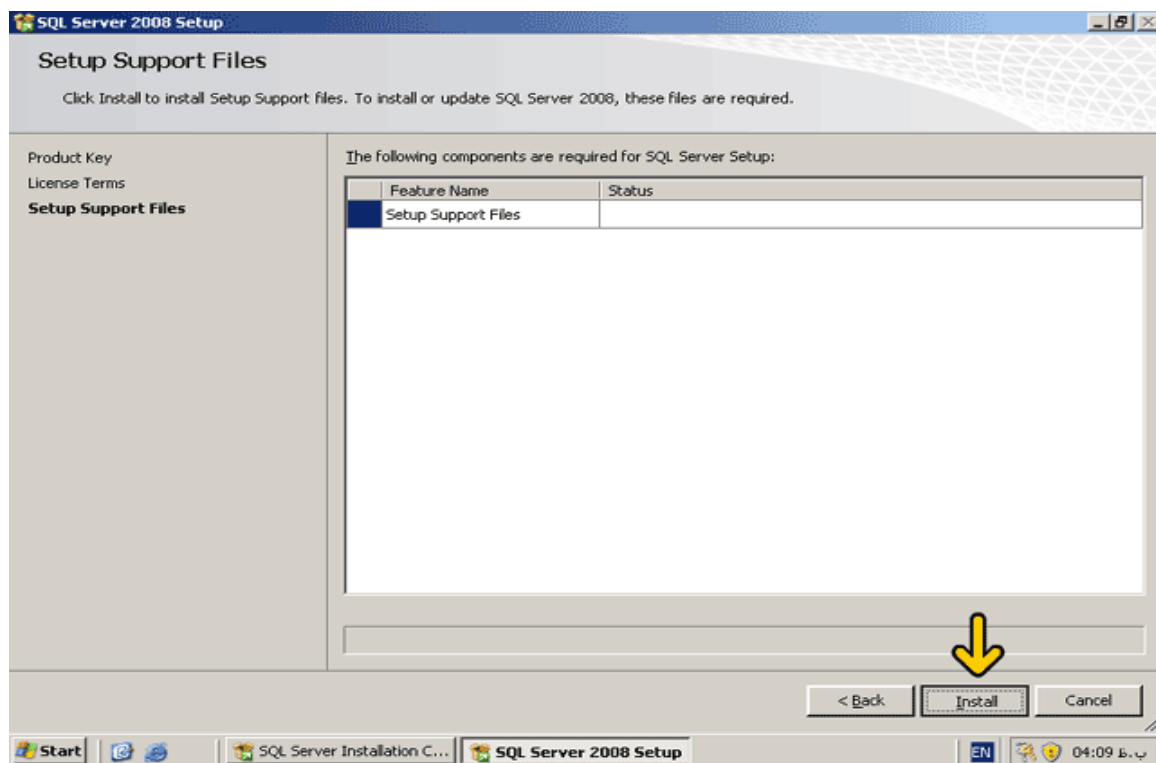
شکل ۱۱-۱



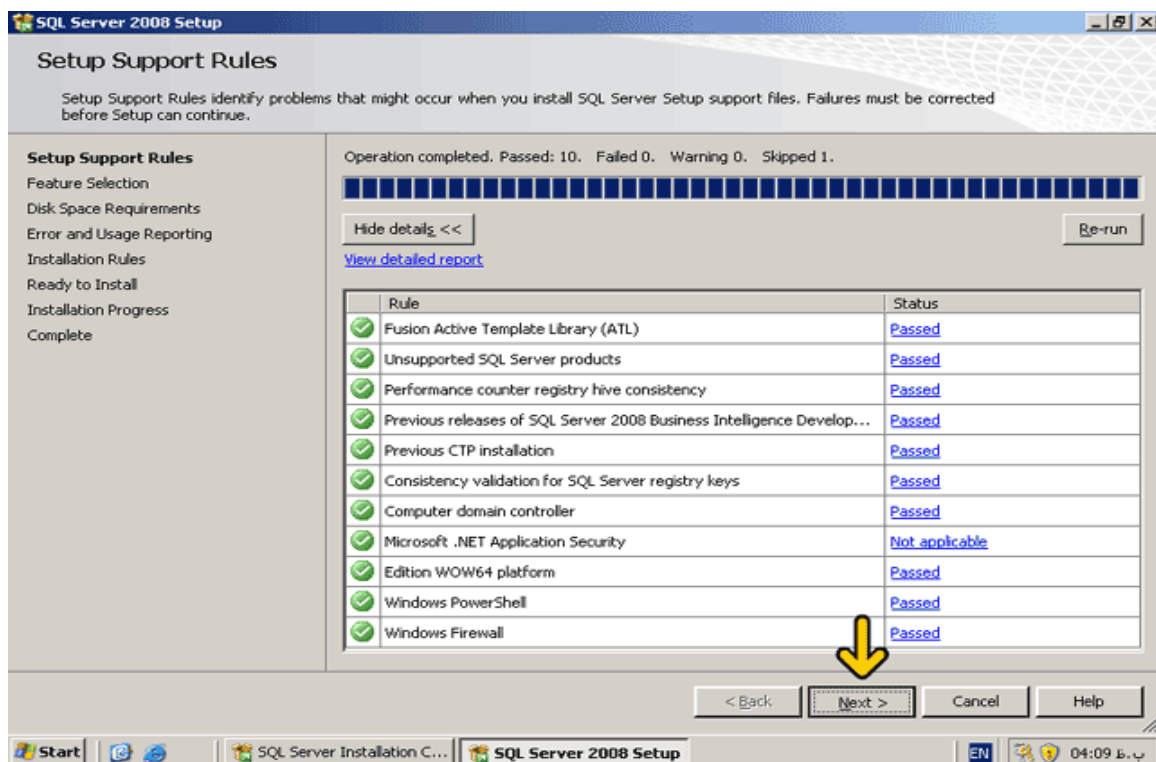
شکل ۱-۱۲



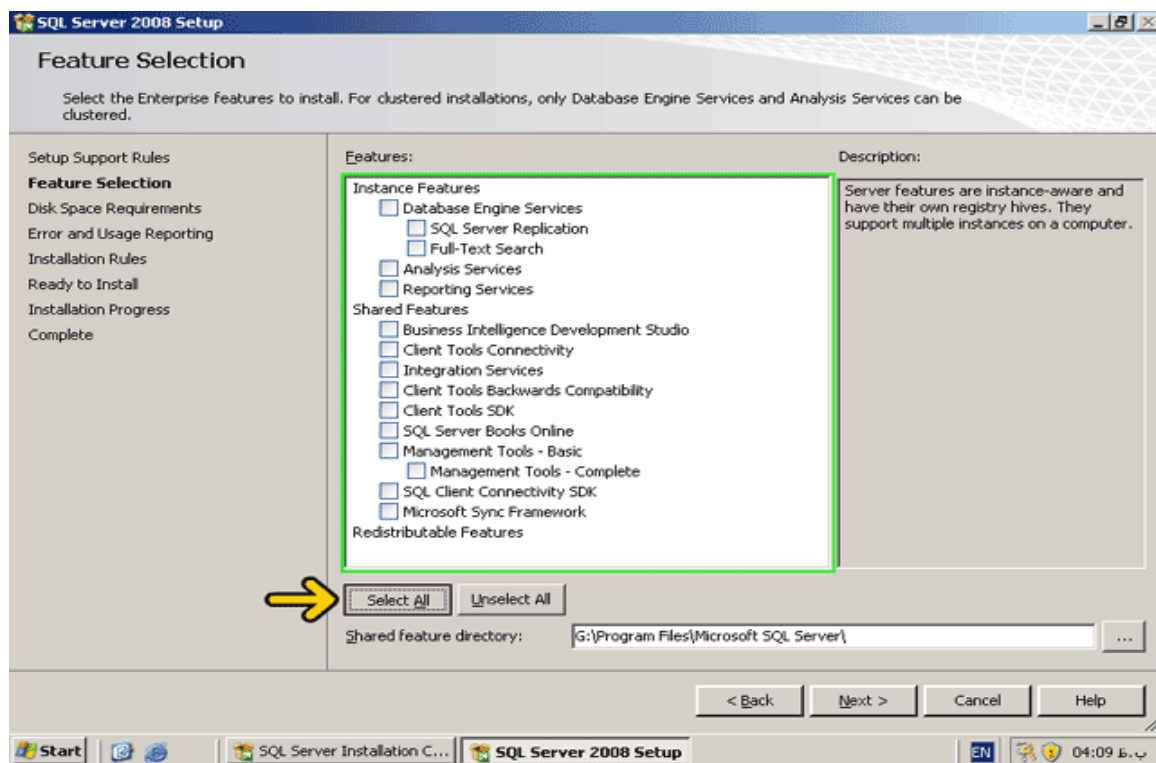
شکل ۱-۱۳



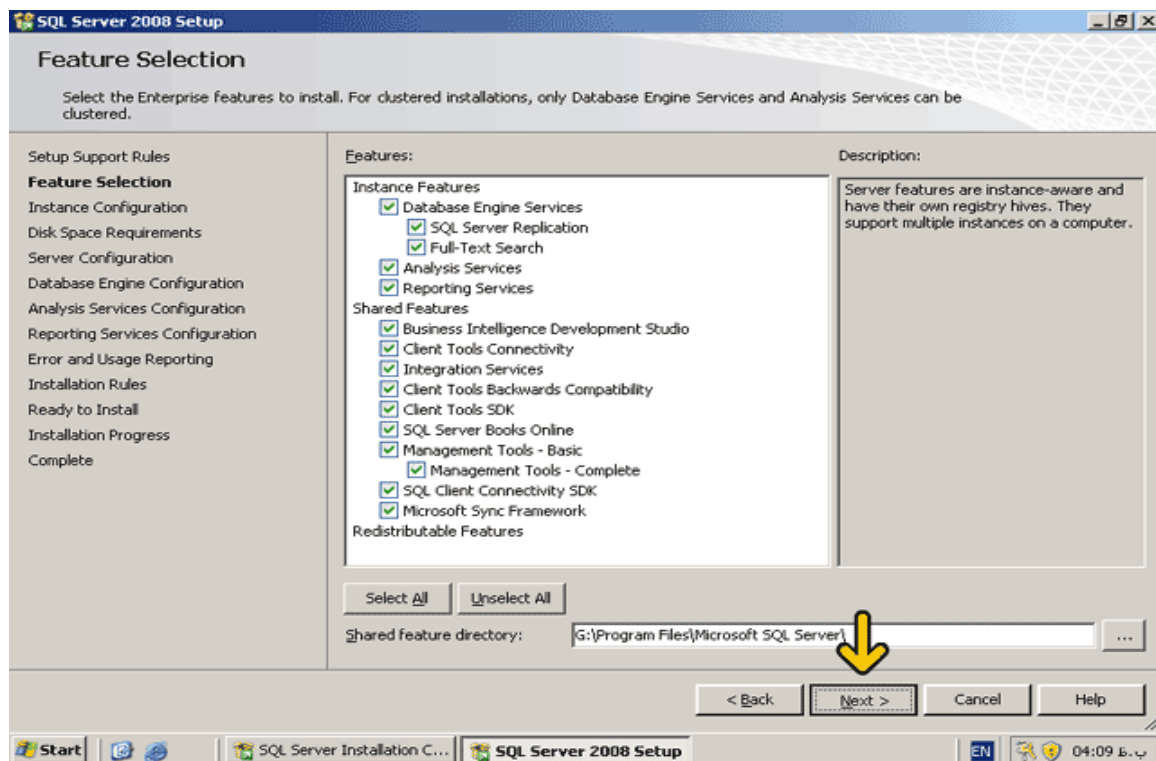
شکل ۱-۱۴



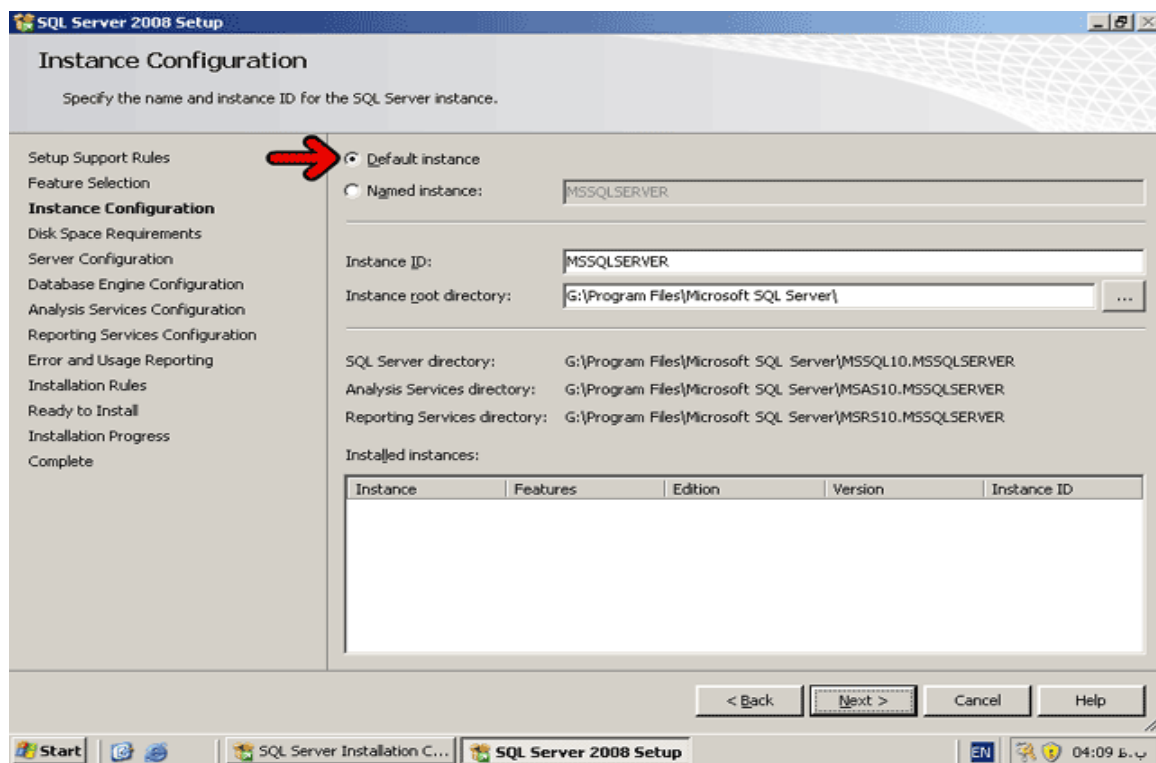
شکل ۱-۱۵



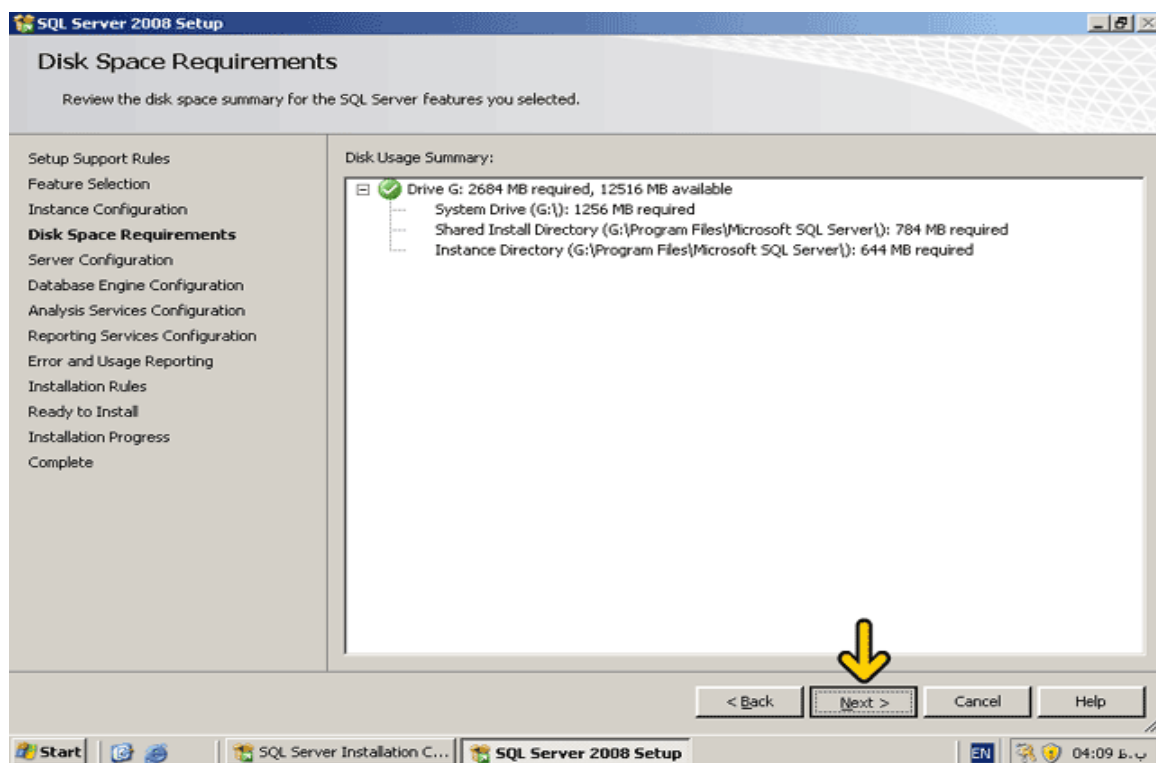
شکل ۱-۱۶



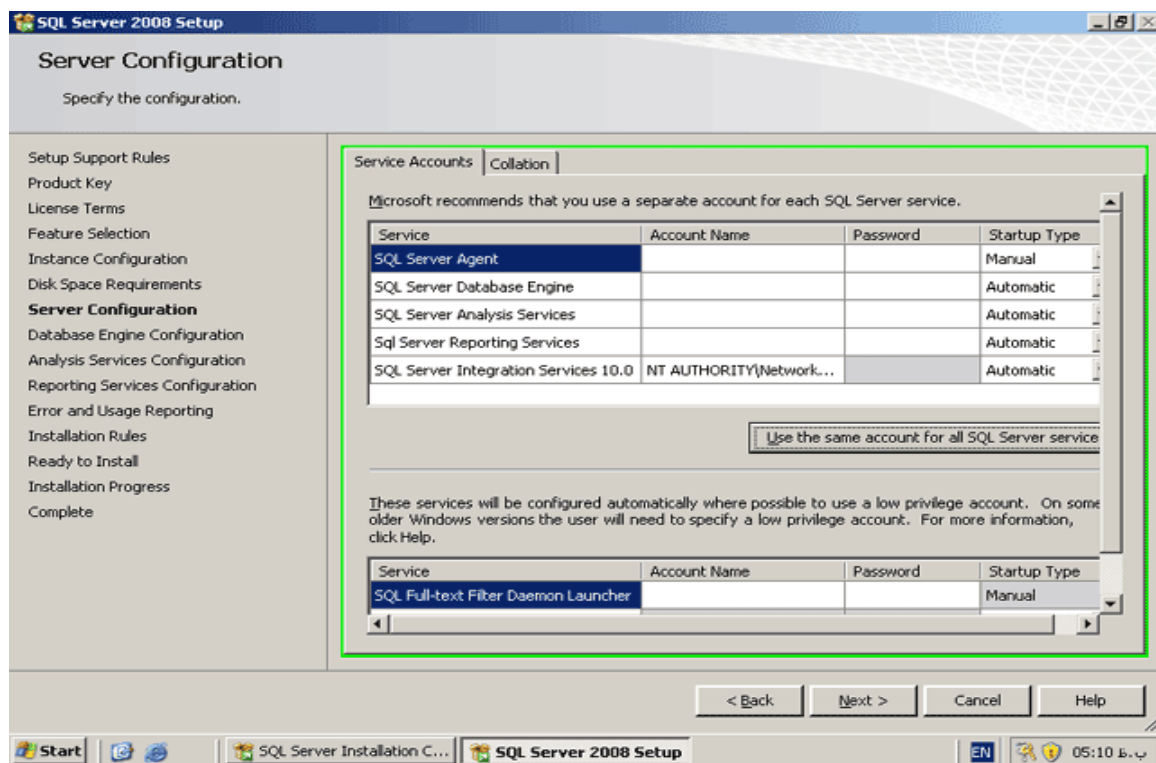
شکل ۱-۱۷



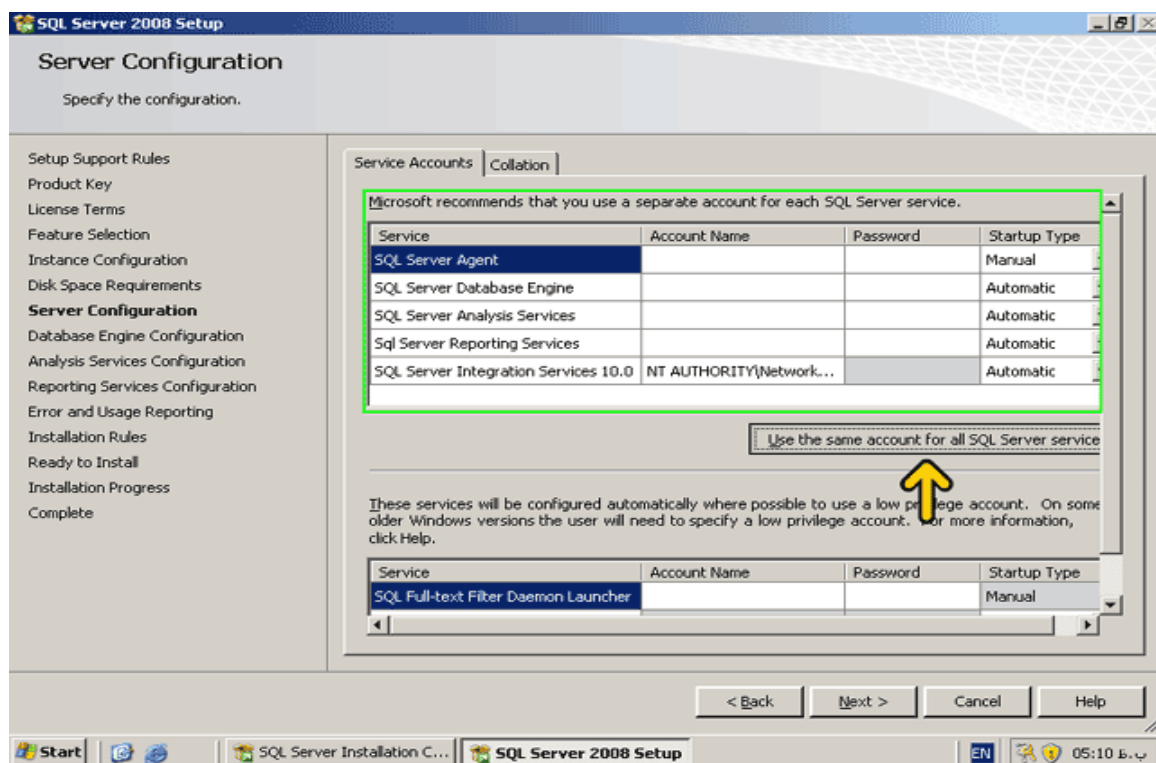
شکل ۱-۱۸



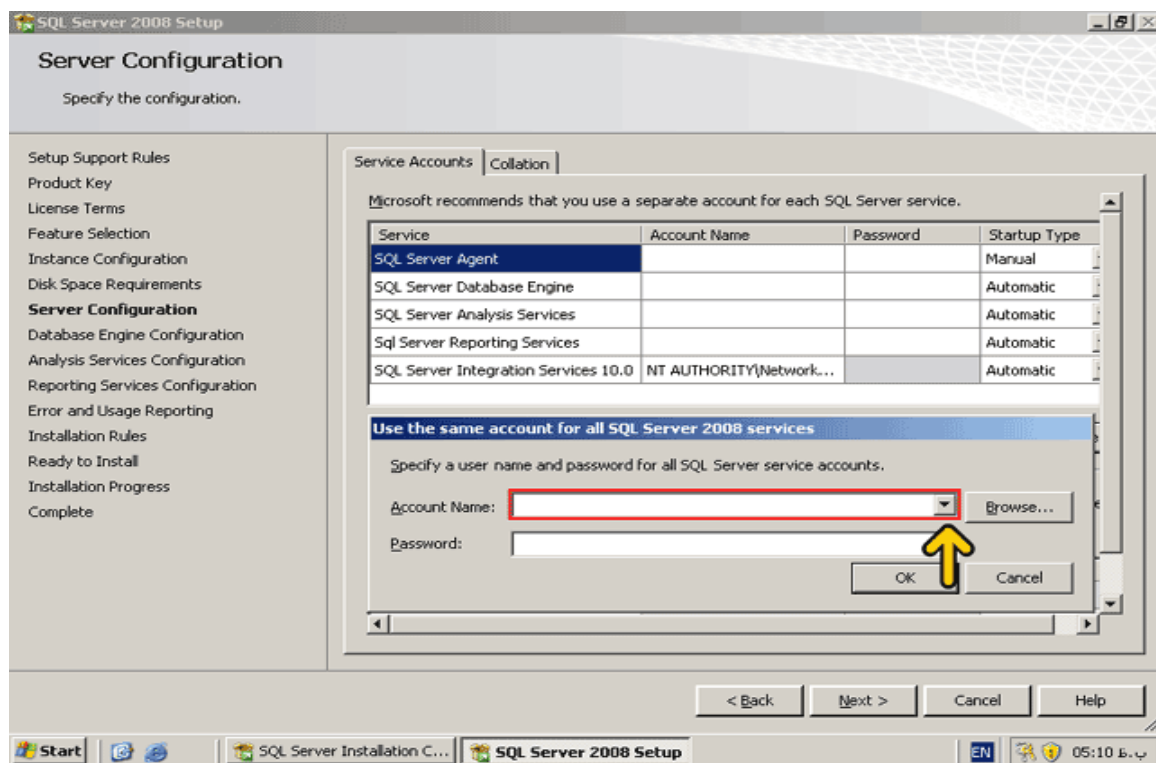
شکل ۱-۱۹



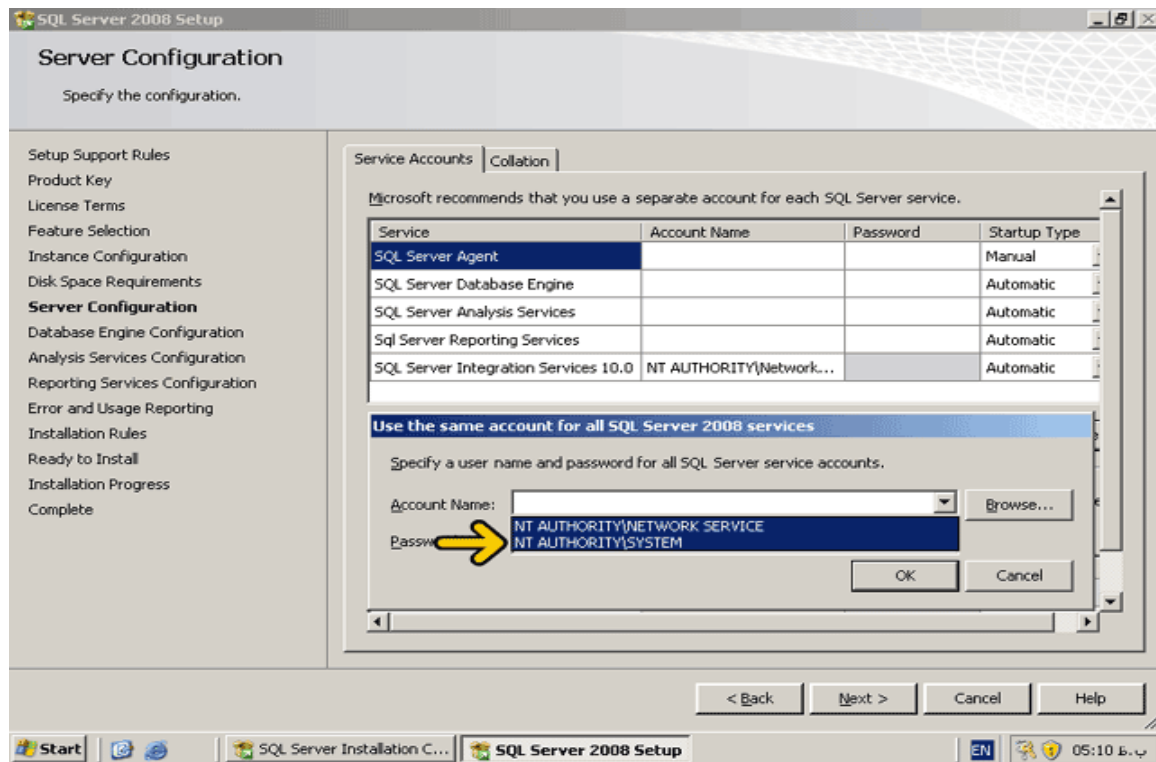
شکل ۲۰-۱



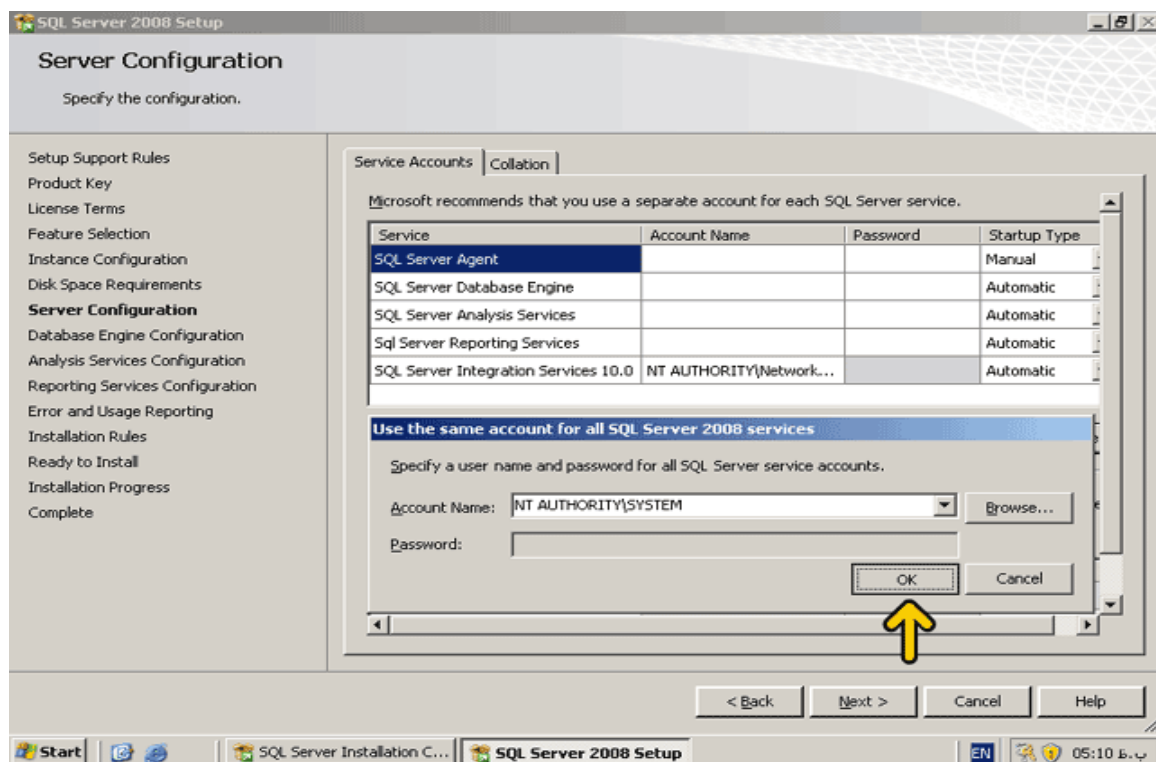
شکل ۲۱-۱



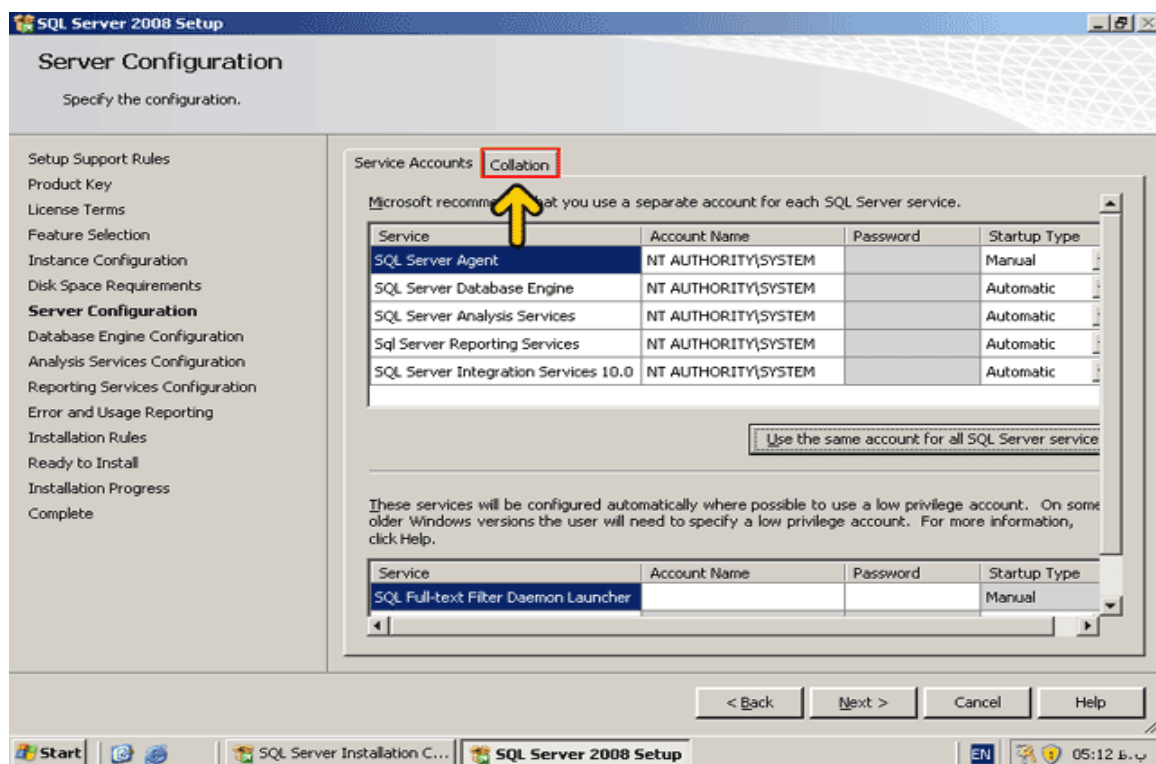
شکل ۲۲-۱



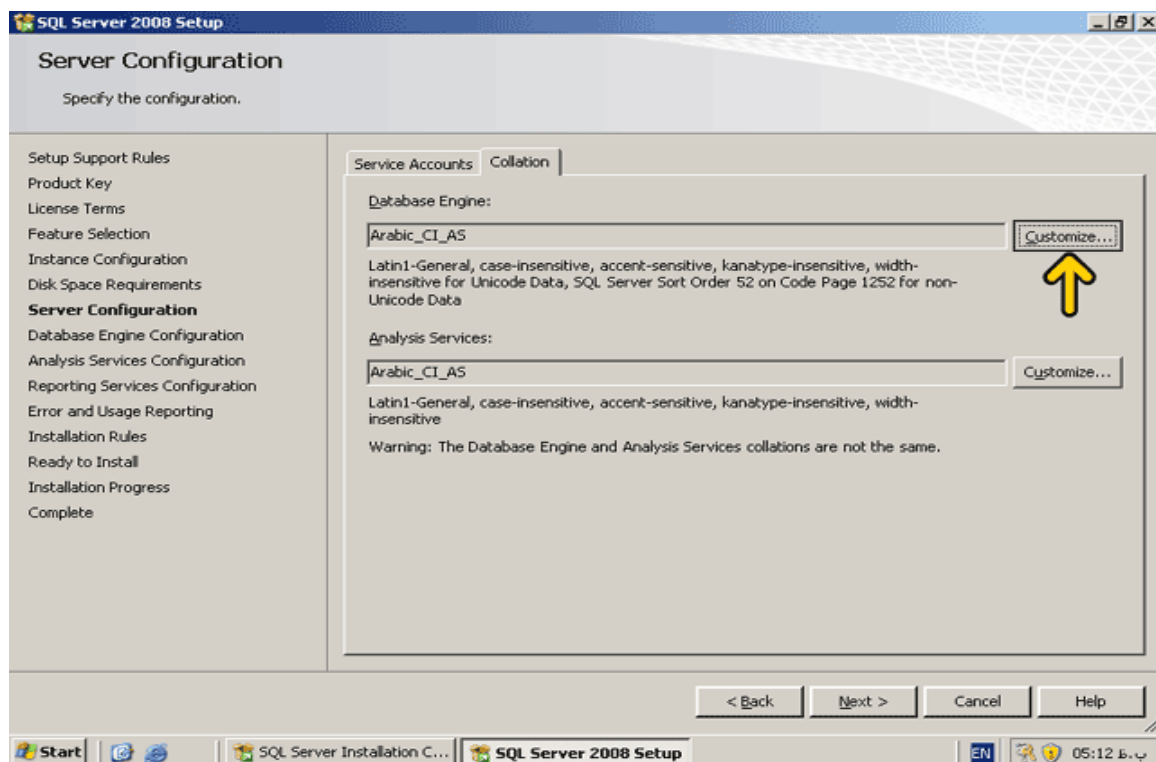
شکل ۲۳-۱



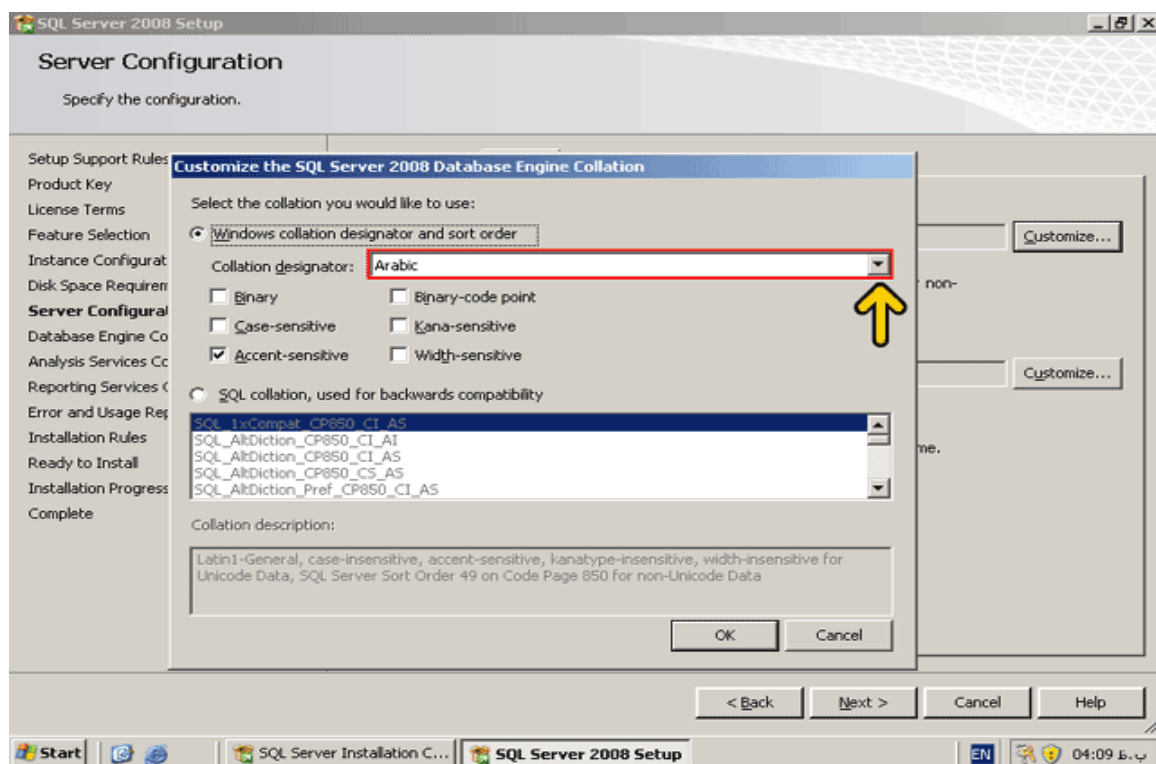
شکل ۱-۲۴



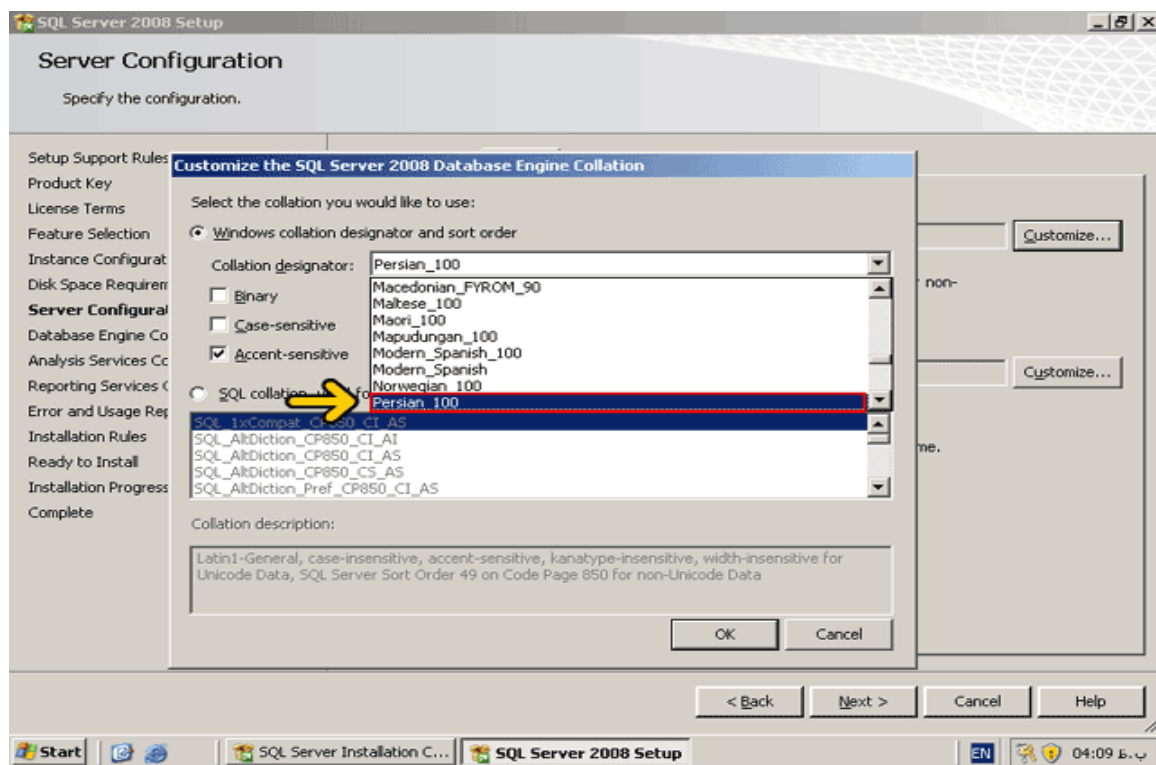
شکل ۱-۲۵



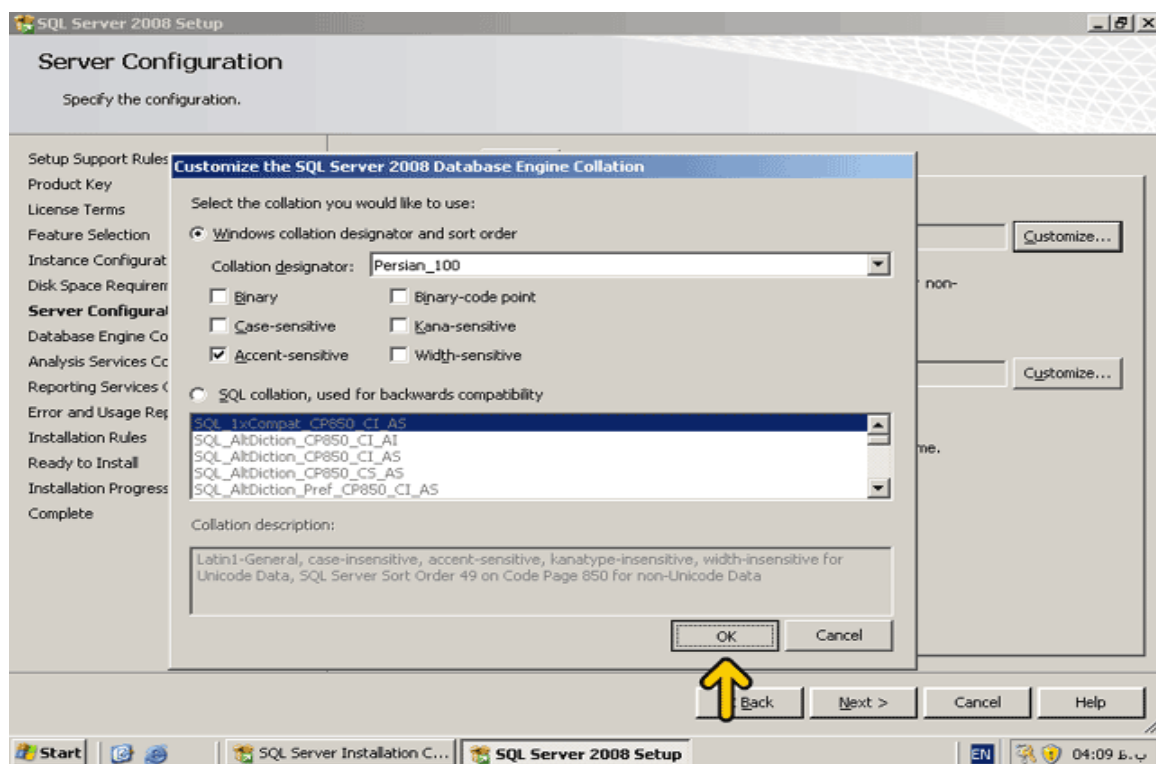
شکل ۲۶-۱



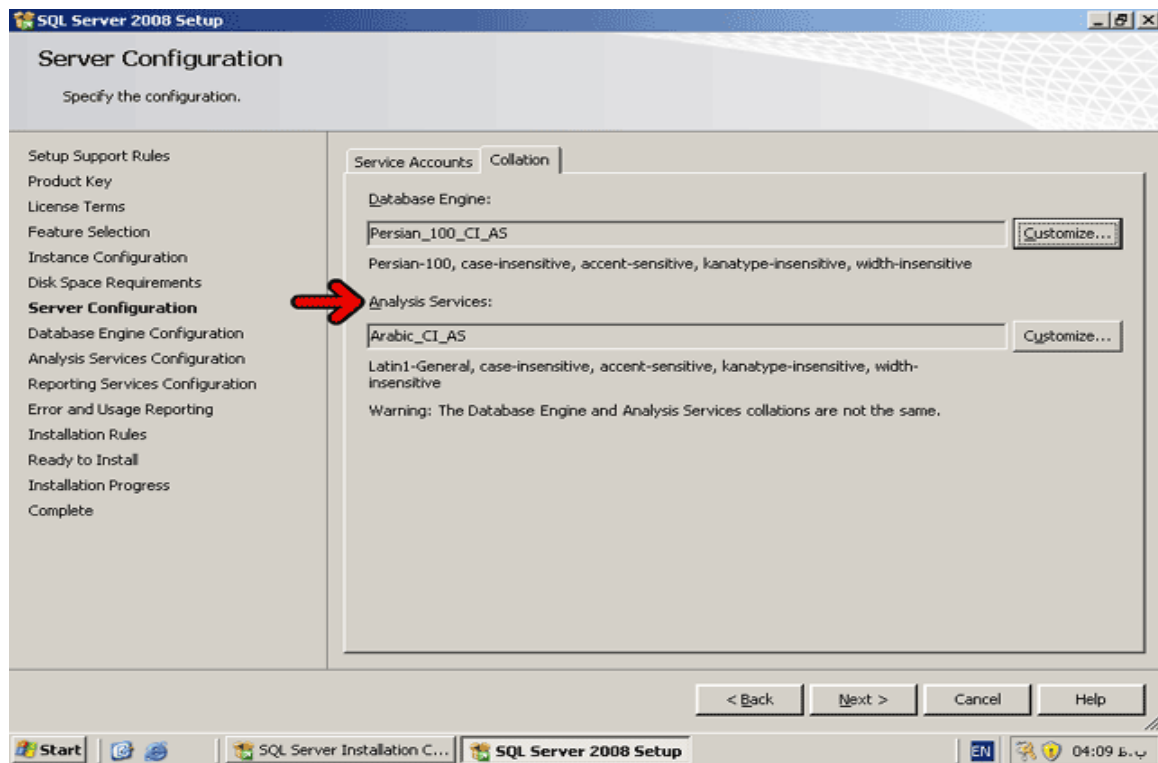
شکل ۲۷-۱



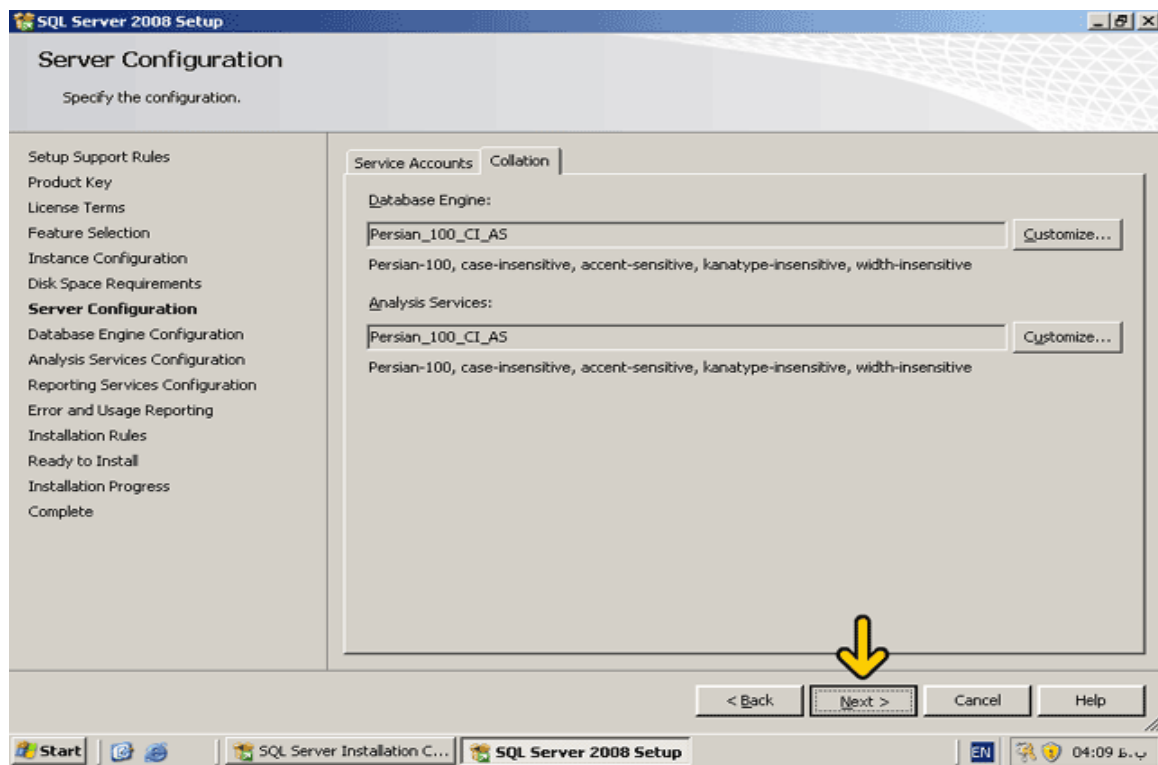
شکل ۲۸-۱



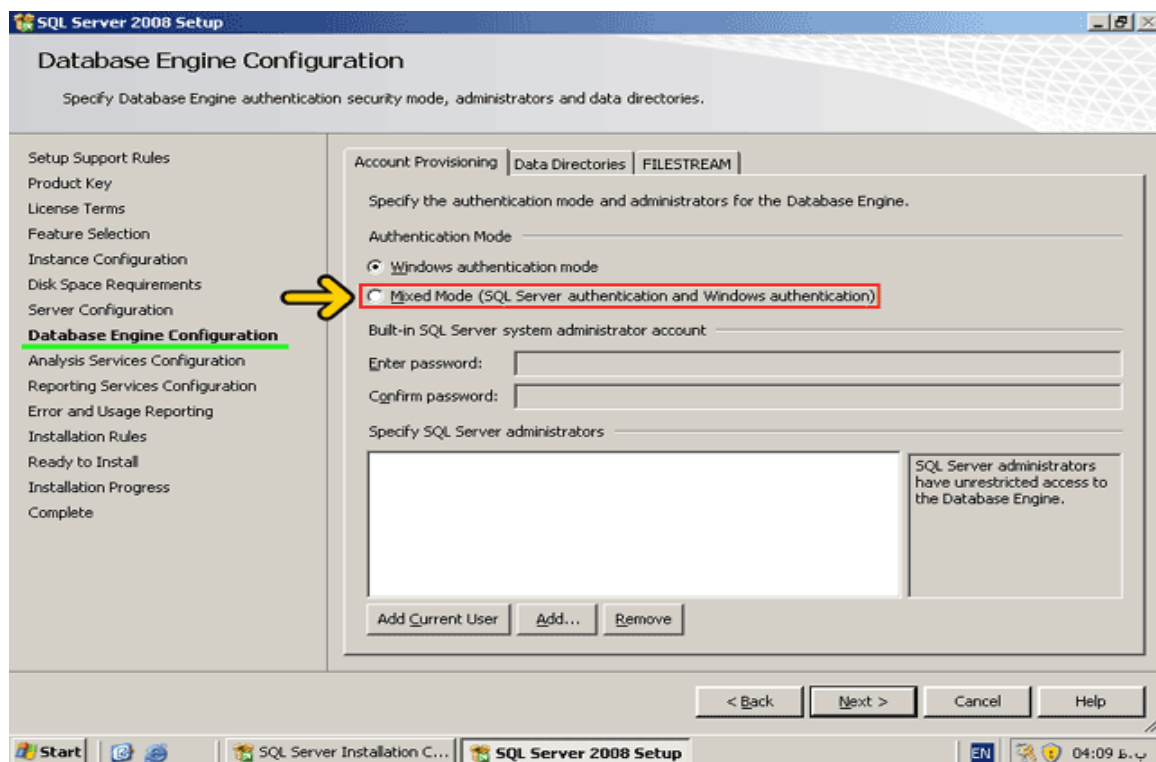
شکل ۲۹-۱



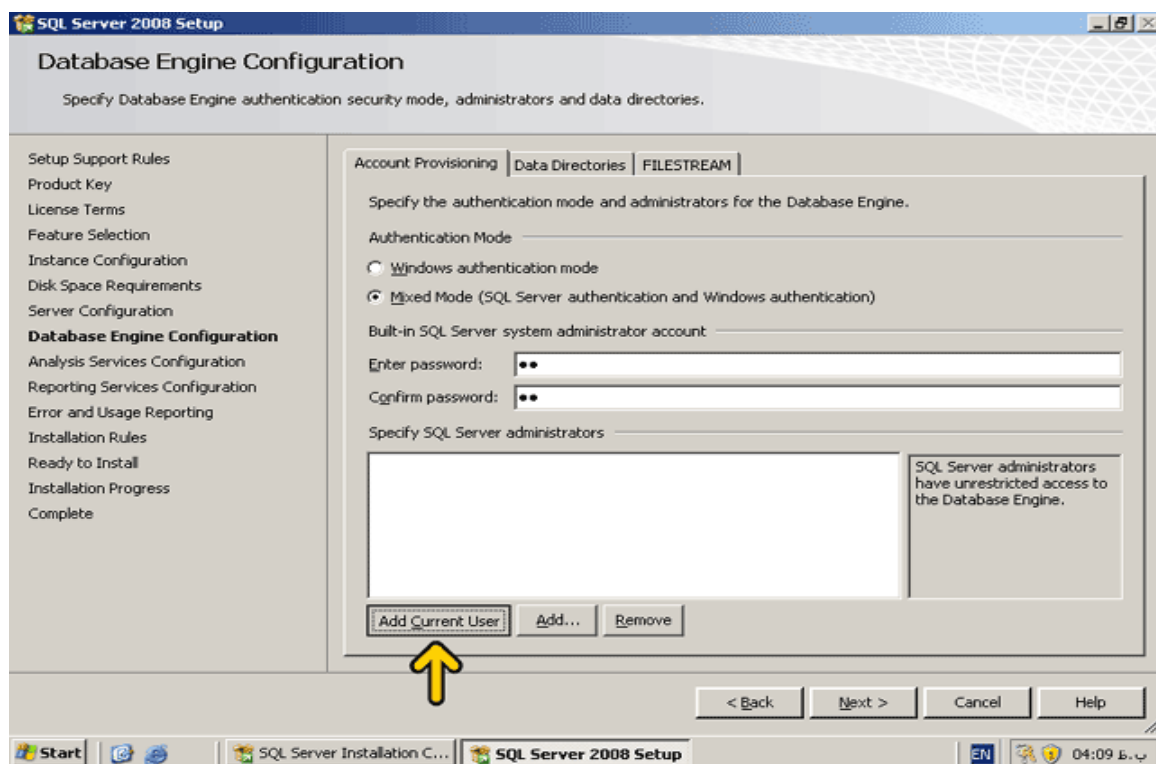
شکل ۳۰-۱



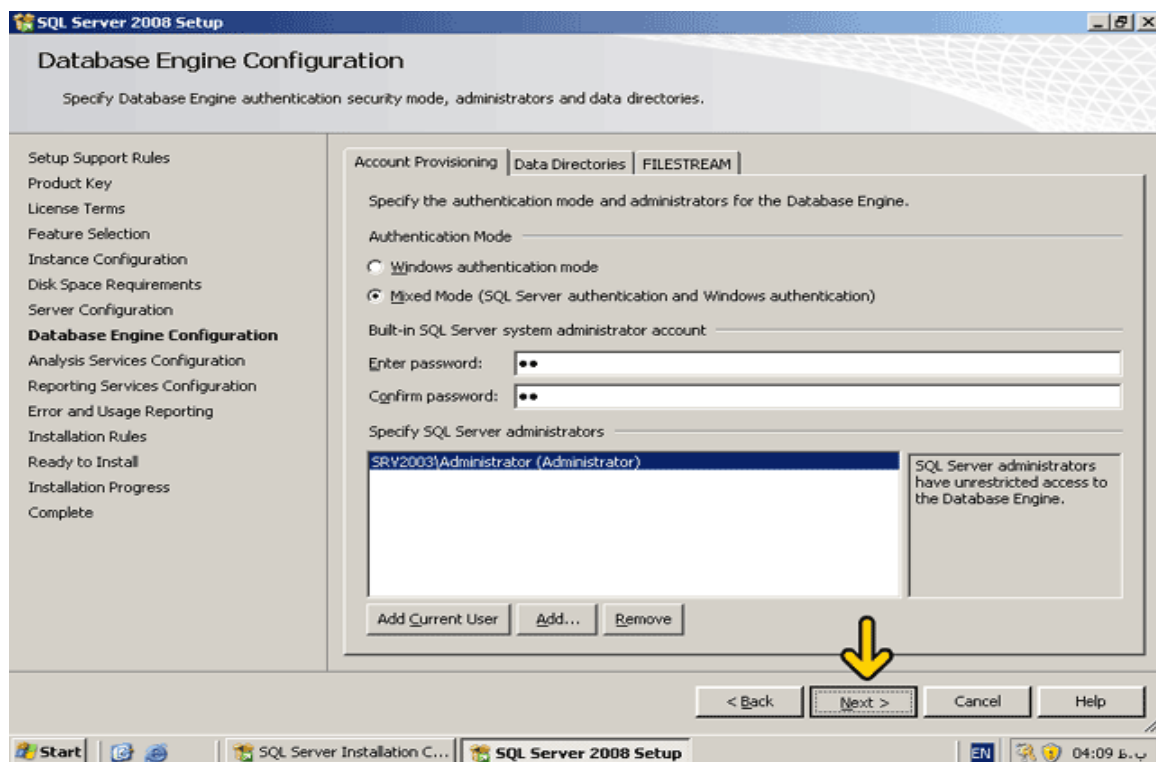
شکل ۳۱-۱



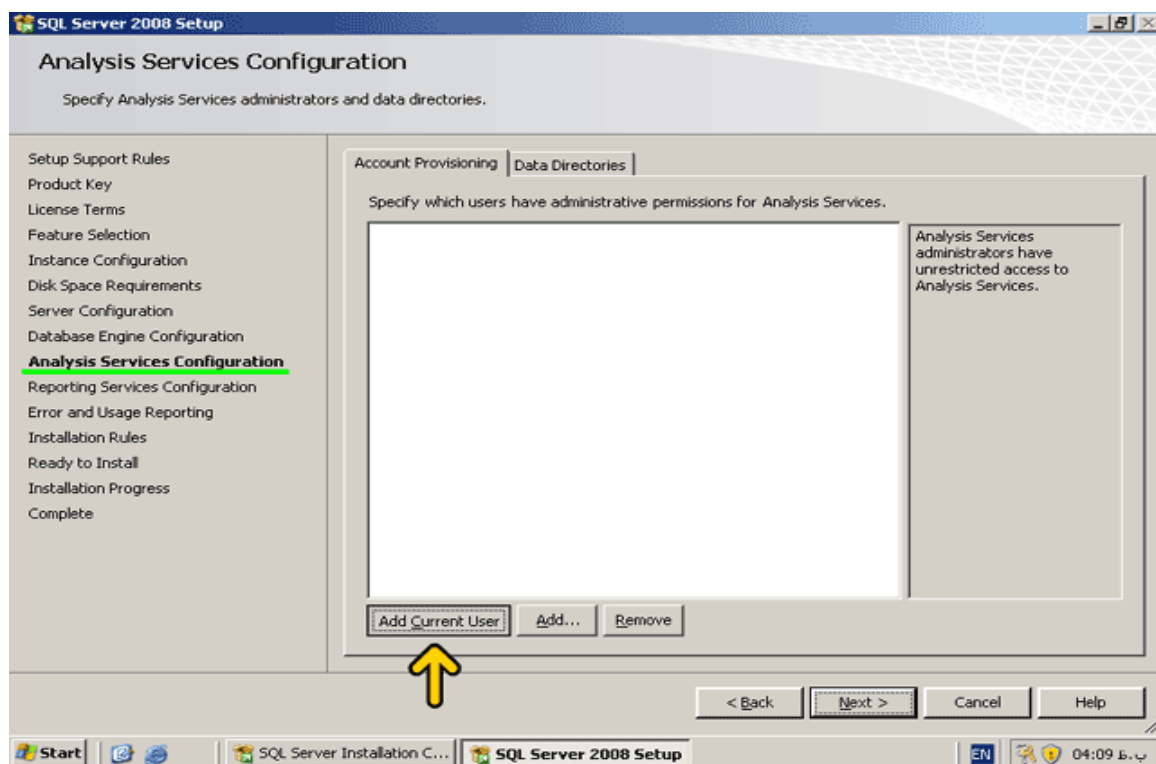
شکل ۱-۳۲



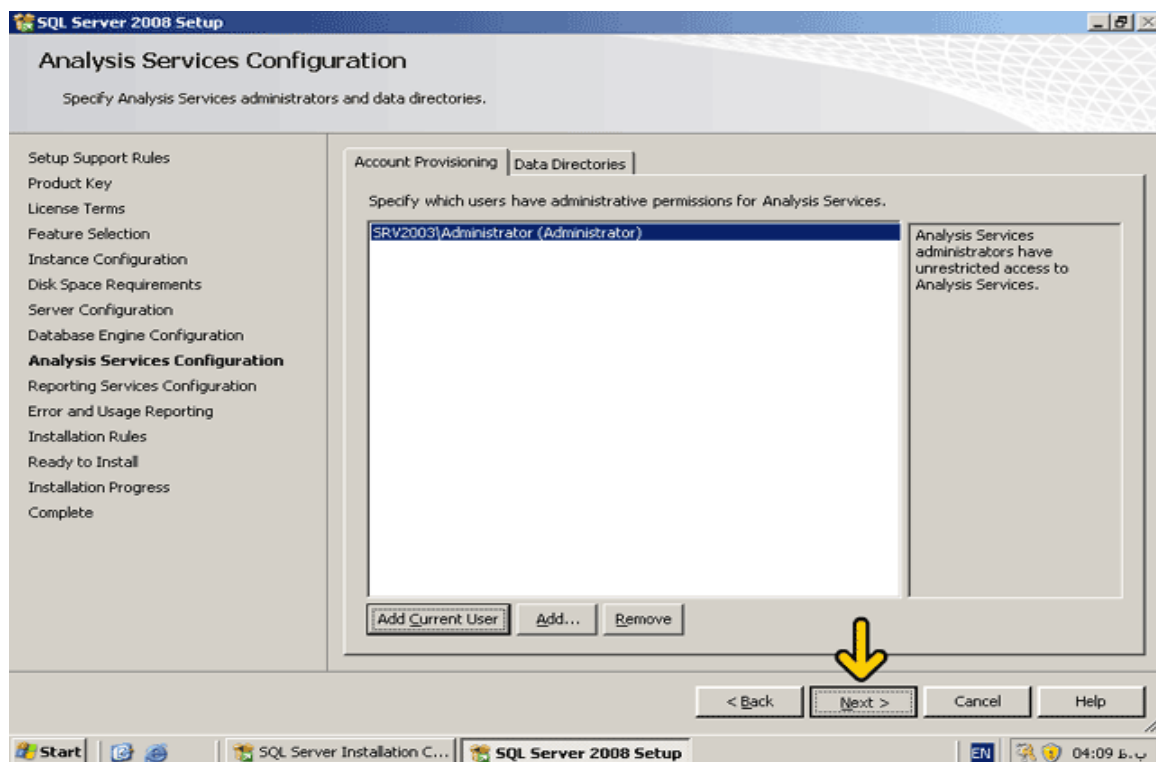
شکل ۱-۳۳



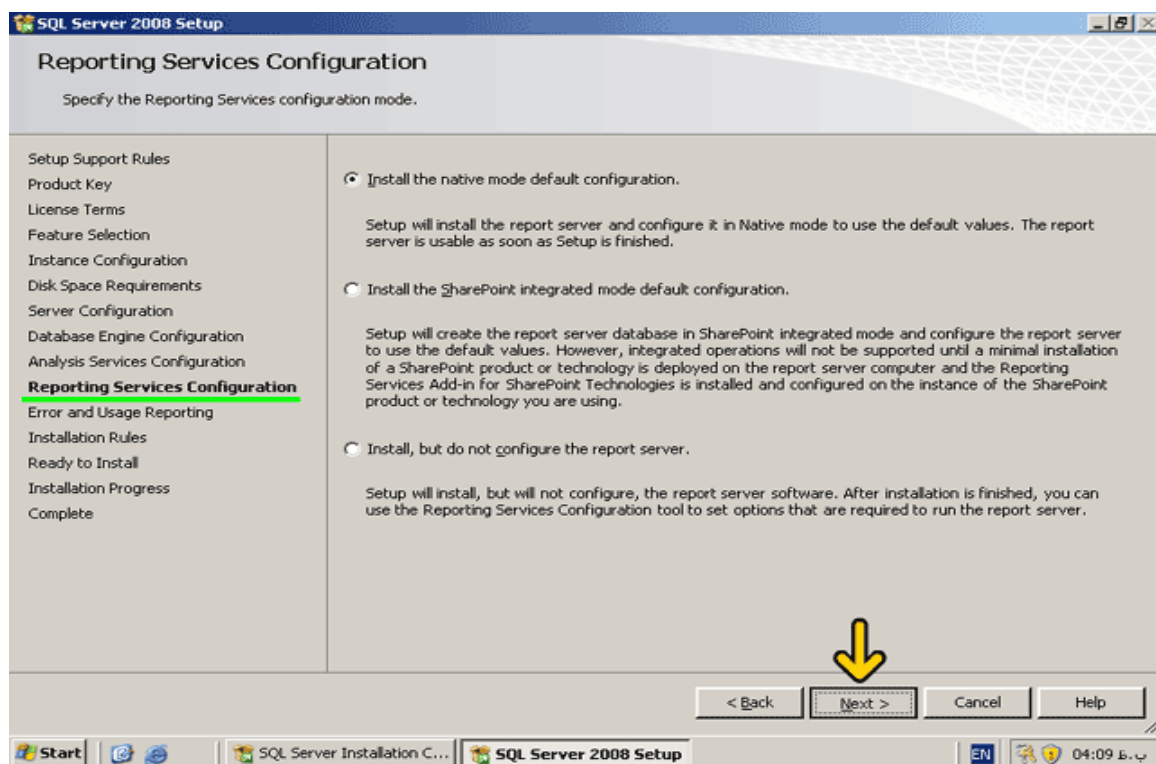
شکل ۱-۳۴



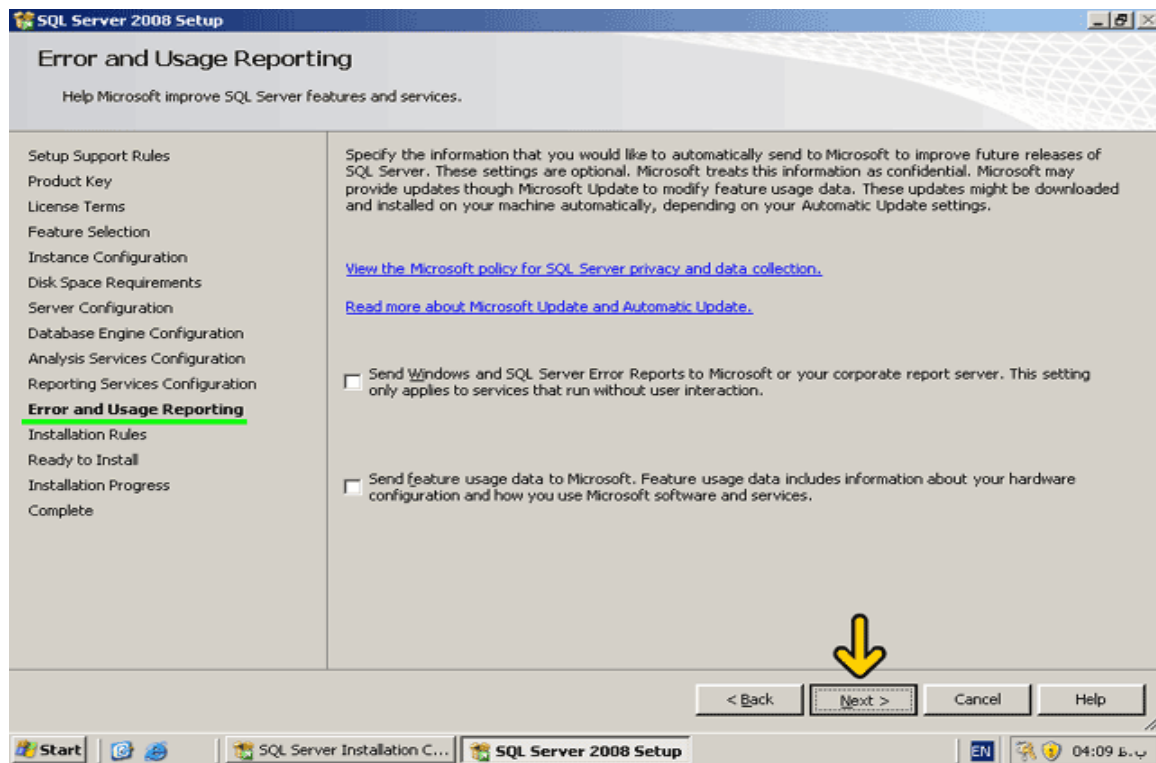
شکل ۱-۳۵



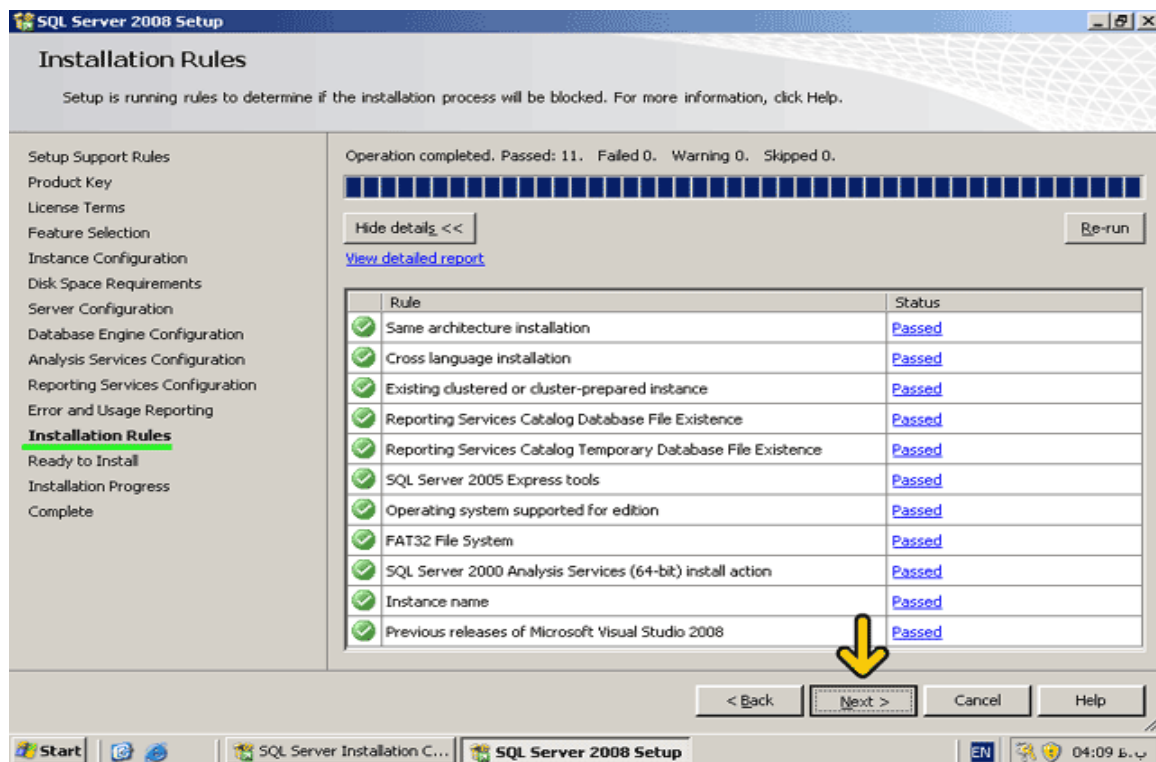
شکل ۱-۳۶



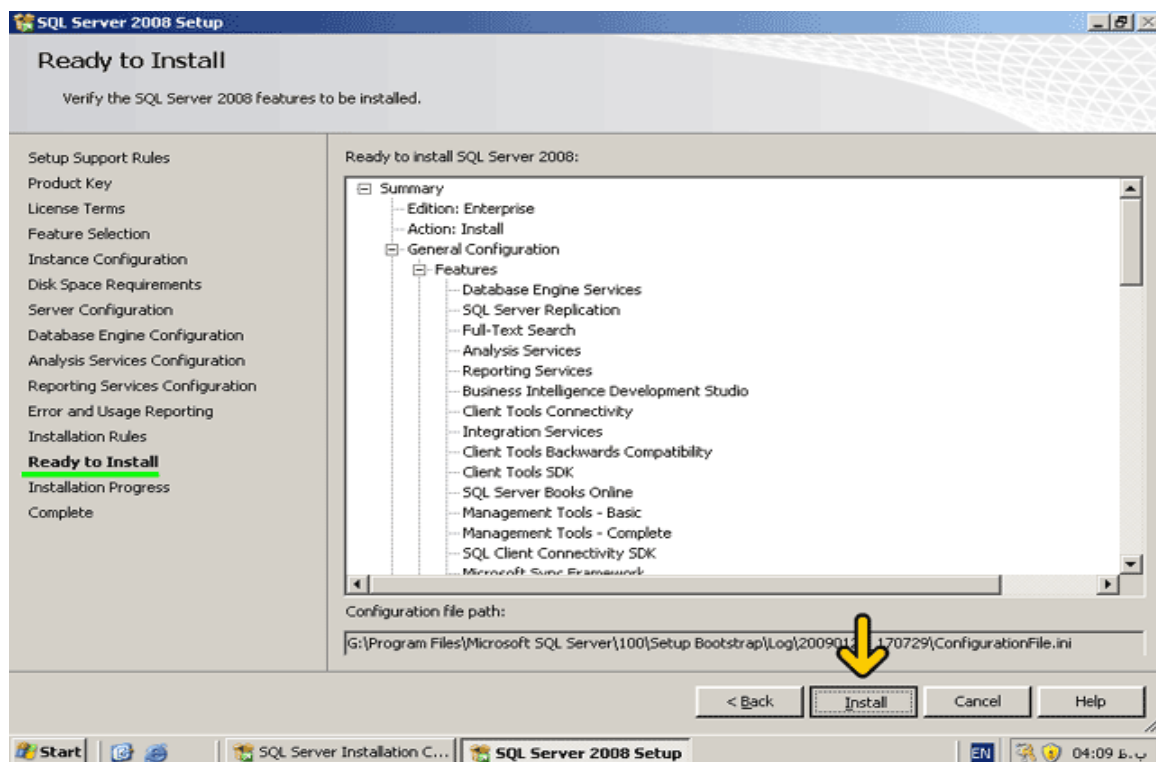
شکل ۱-۳۷



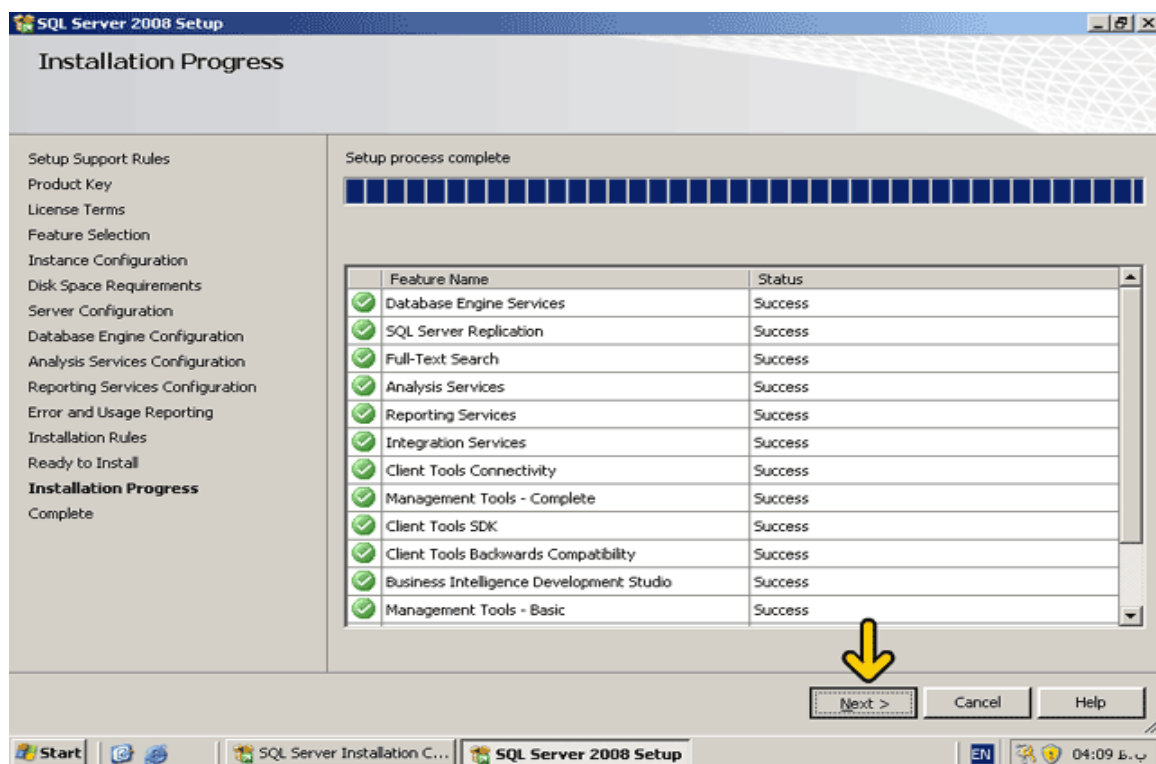
شکل ۳۸-۱



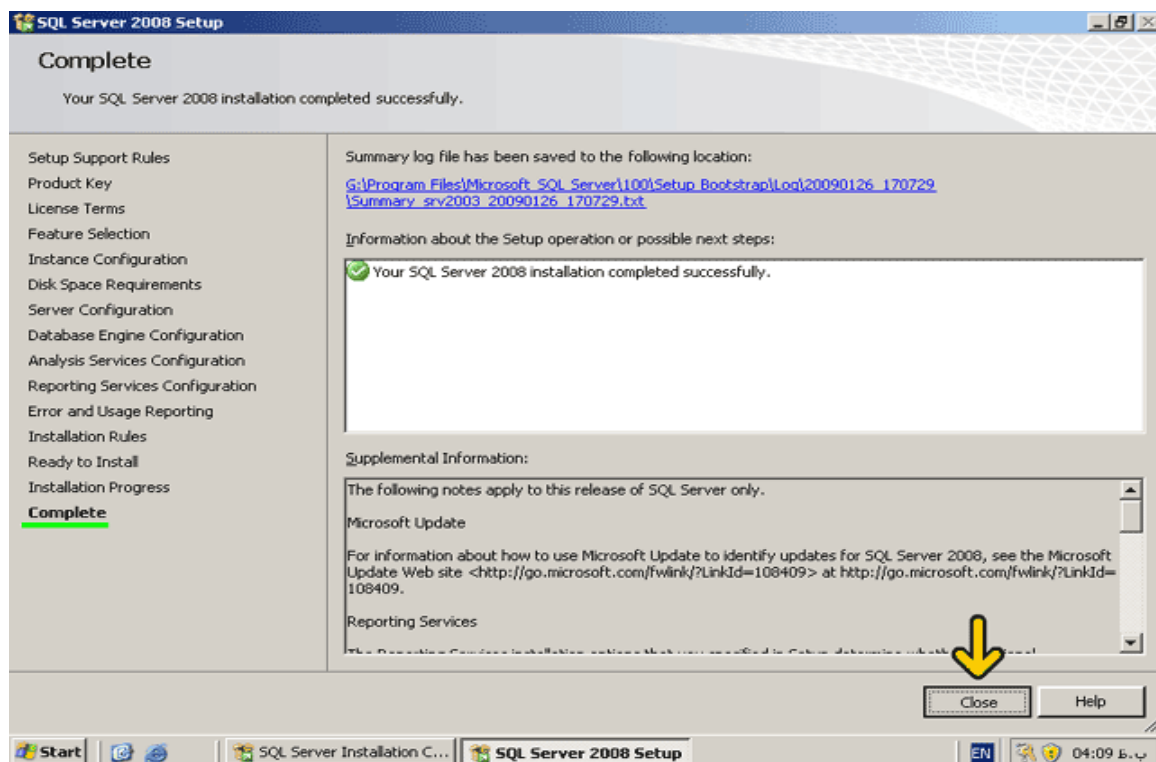
شکل ۳۹-۱



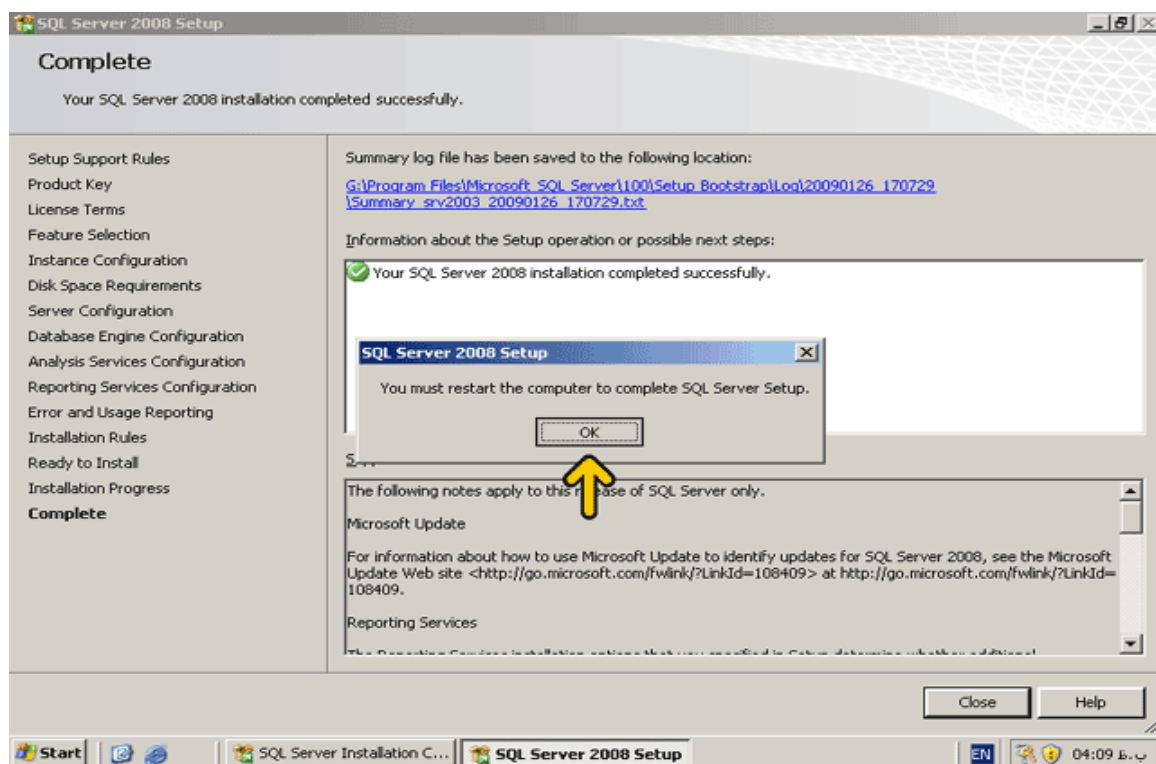
شکل ۴۰-۱



شکل ۴۱-۱



شکل ۴۲-۱

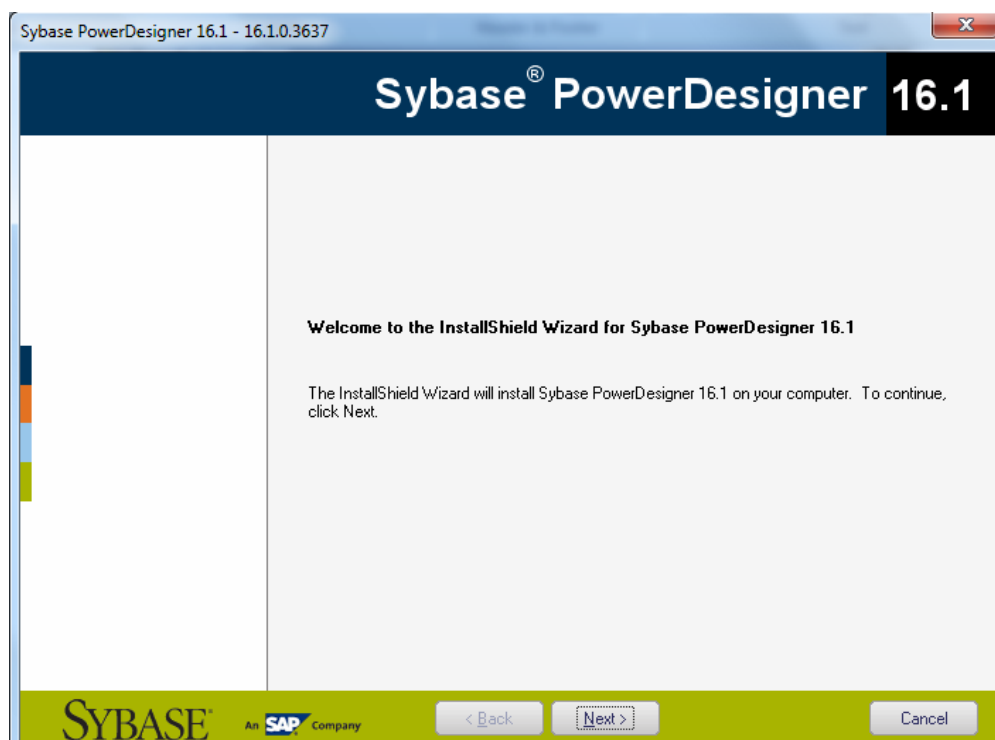


شکل ۴۳-۱

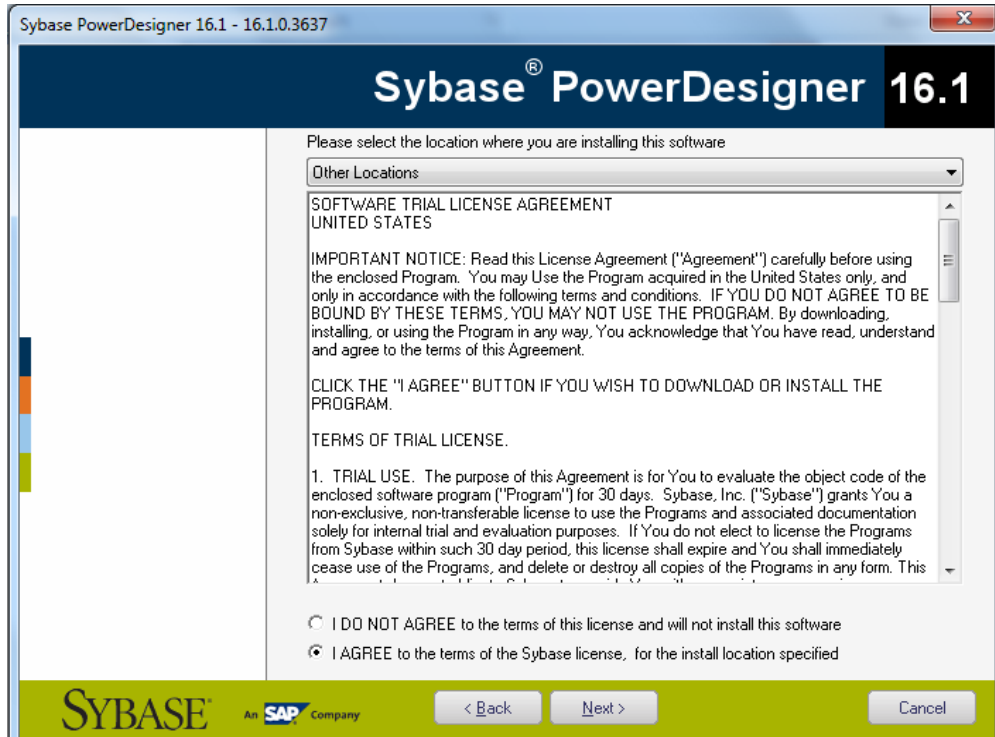


شکل ۱-۴۴

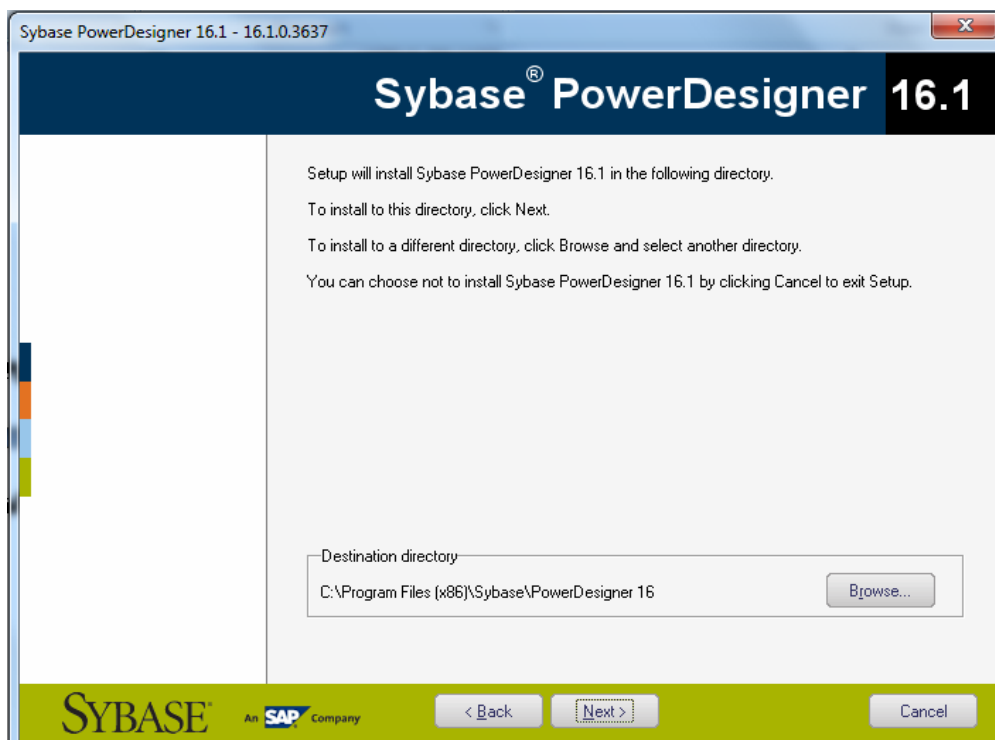
پس از نصب نرم افزار Microsoft SQL Server 2008 R2، نوبت به نصب و فعالسازی برنامه Power Designer می‌رسد. بدین منظور فایل اجرایی Setup.exe را اجرا نمایید.



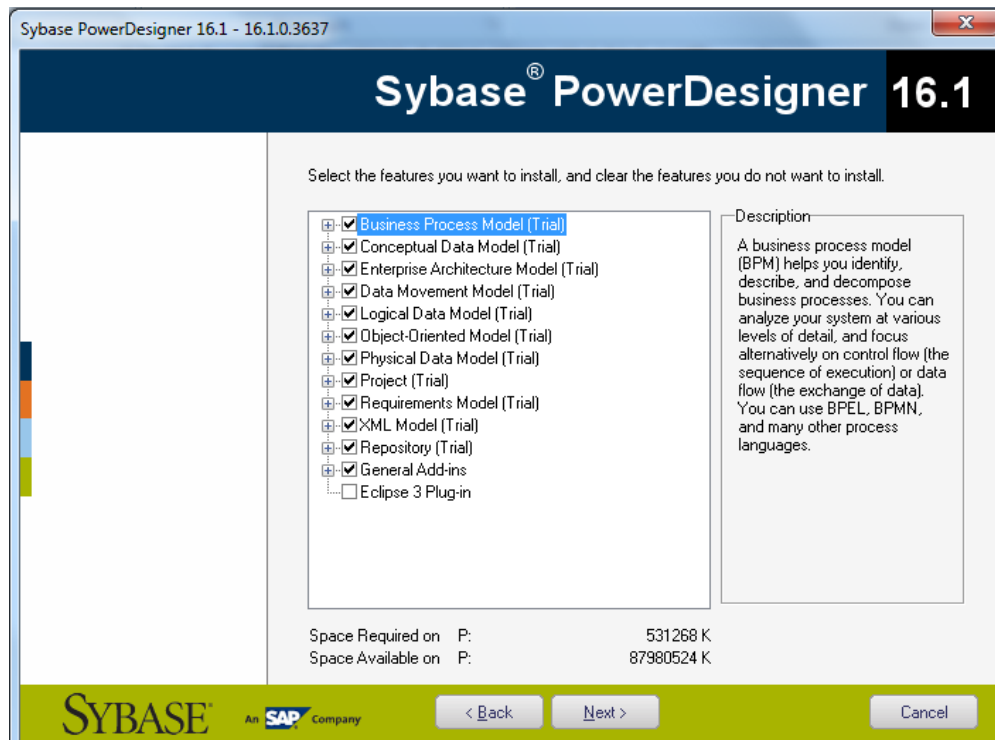
شکل ۱-۴۵



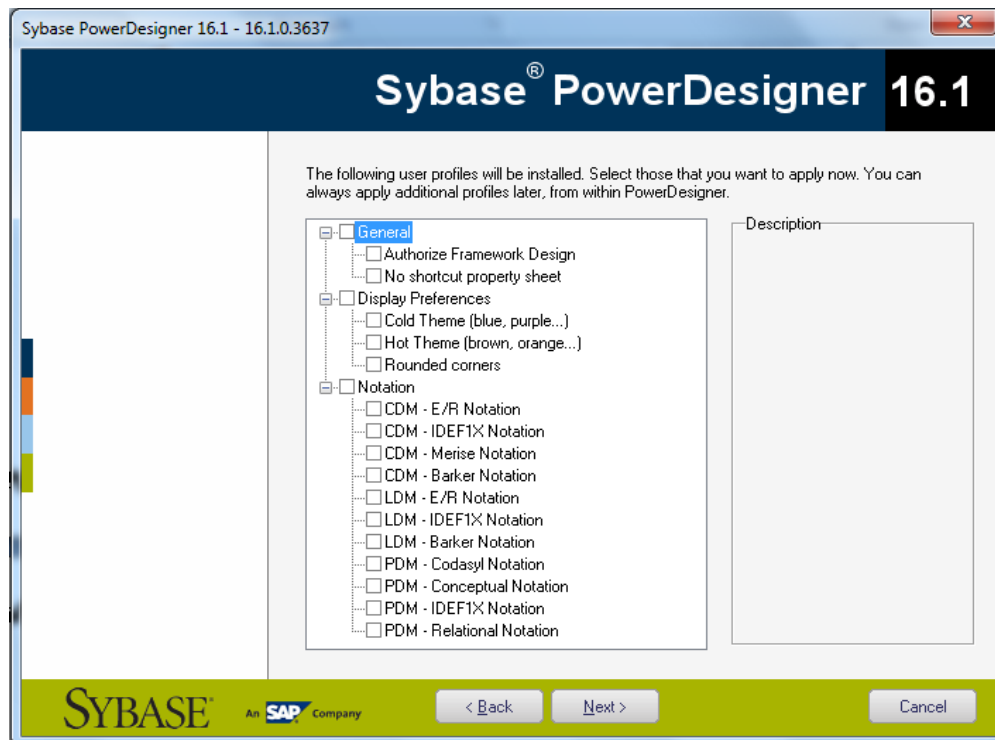
شکل ۱-۴۶



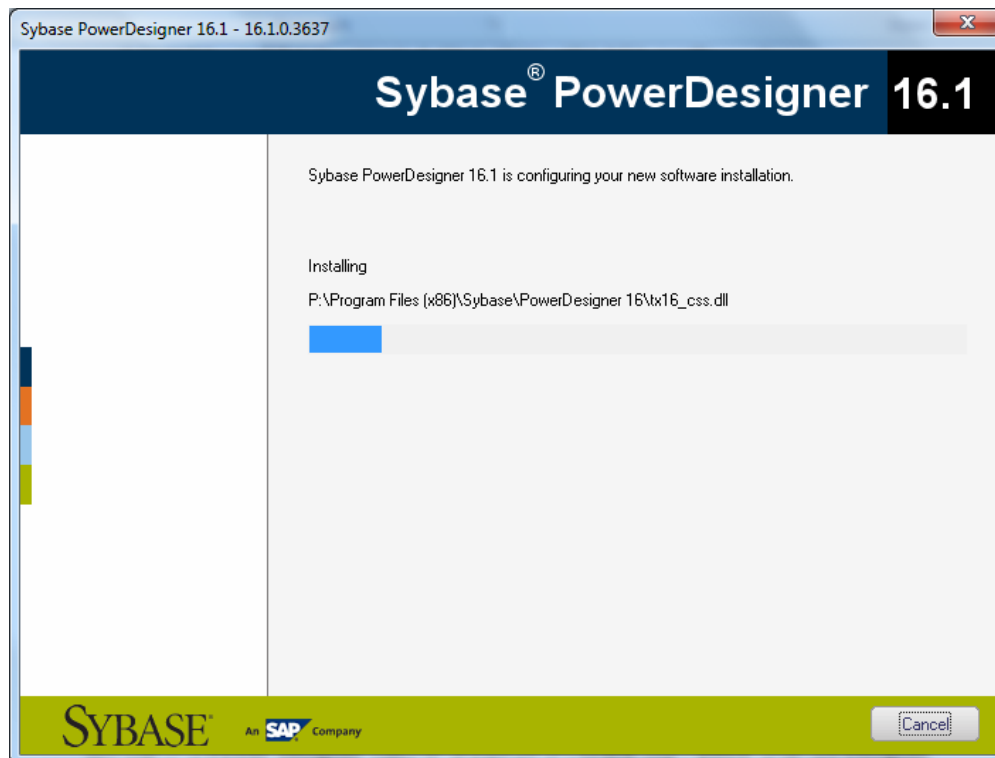
شکل ۱-۴۷



شکل ۱-۴۸



شکل ۱-۴۹



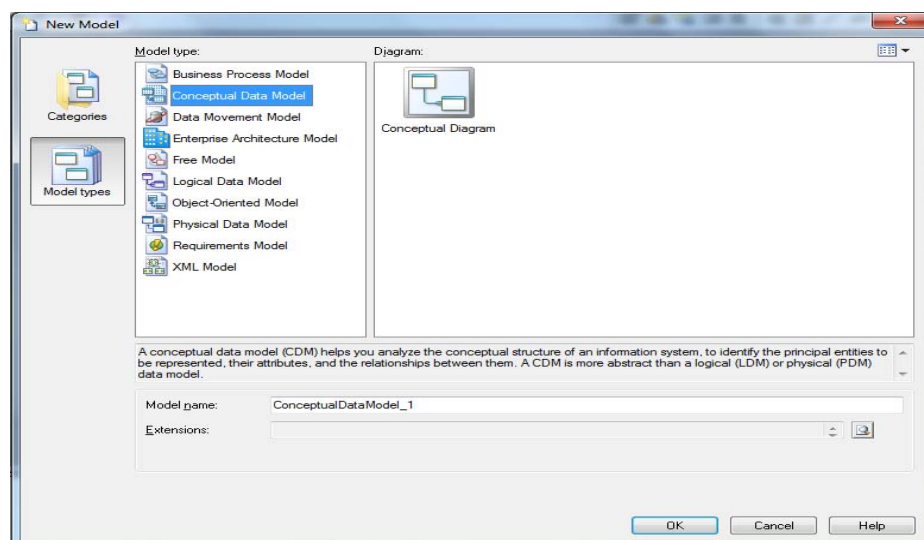
شکل ۱-۵۰

حال به مراحل فعالسازی نرمافزار Power Designer شرکت Sybase می‌پردازیم. پس از نصب نرمافزار Power Designer فایل اجرایی patch از فولدر Crack را در دایرکتوری‌ای که نرمافزار Power Designer در آن نصب شده است کپی نمایید سپس فایل patch را اجرا نمایید. توجه نمایید که چنانچه فایل patch را در جایی بجز دایرکتوری‌ای که نرمافزار Power Designer در آن نصب شده اجرا نمایید فعالسازی نرمافزار به شکست خواهد انجامید. دایرکتوری پیش فرض نصب نرمافزار به صورت ذیل می‌باشد:

C:\Program Files\Sybase\PowerDesigner 15

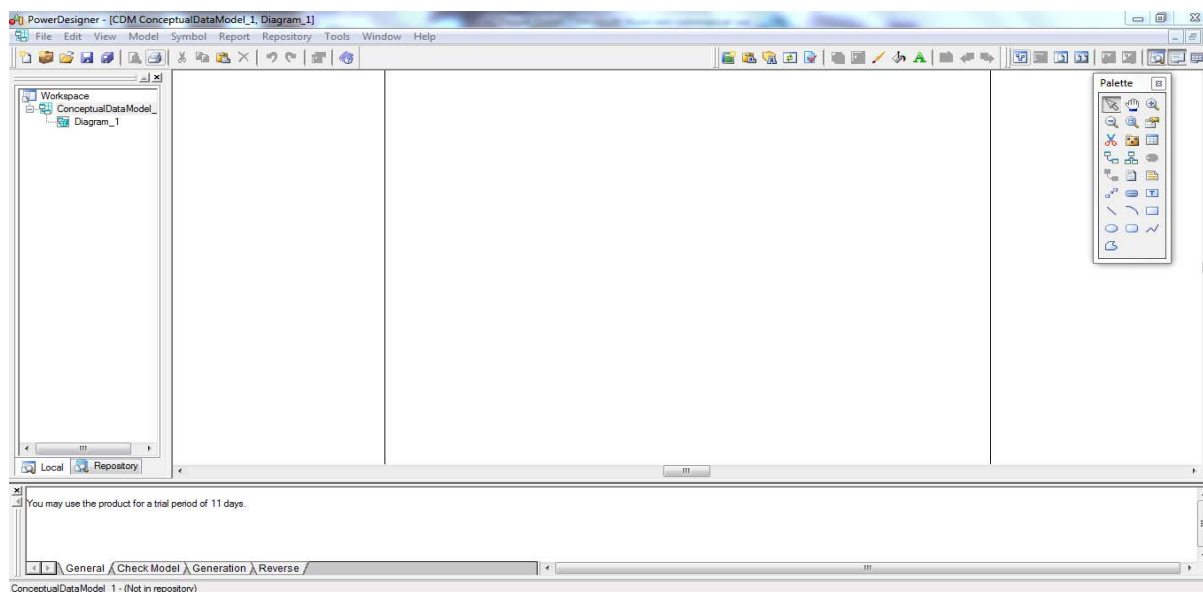
در مورد نسخه ۱۶ این نرمافزار کافیت فایل کتابخانه‌ای DLL بنام pdfm16.dll را در دایرکتوری‌ای که نرمافزار نصب شده است کپی نمایید.

حال که با نحوه فعالسازی نرمافزار Power Designer آشنا شده‌اید به معرفی این نرمافزار می‌پردازیم: پس از باز کردن نرمافزار از منوی File گزینه New Model را انتخاب نمایید.



شکل ۱-۵۱ پنجره مدل جدید در برنامه Power Designer

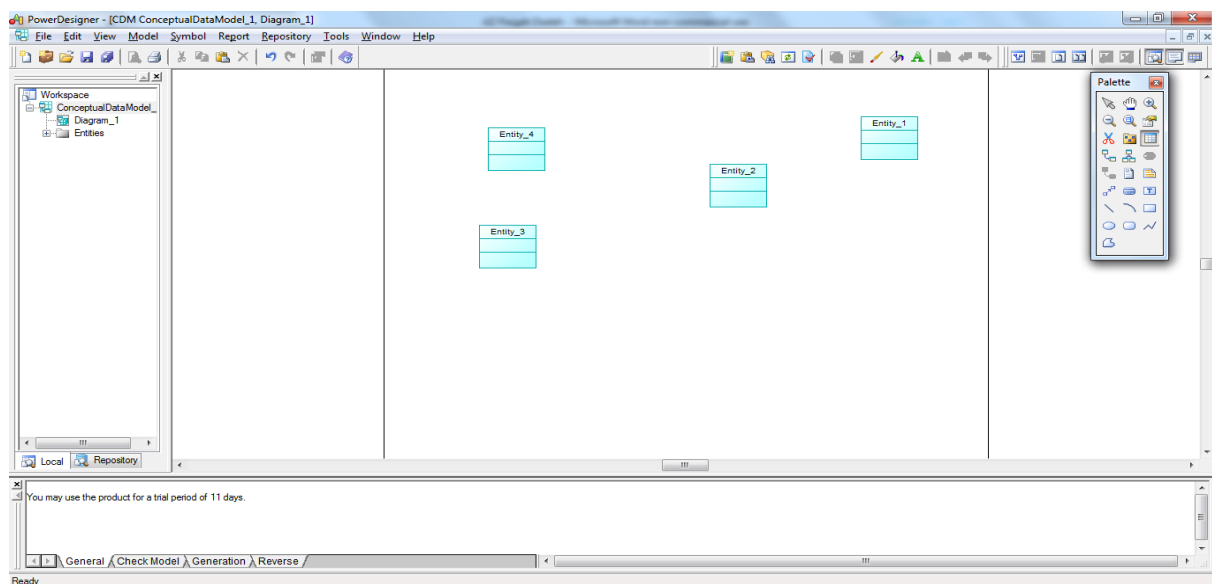
ما در آزمایشگاه پایگاه داده با مدل داده مفهومی (Conceptual Data Model) و مدل داده فیزیکی کار خواهیم نمود. در این جلسه به بررسی مدل داده مفهومی خواهیم پرداخت. بدین منظور ابتدا بر روی نمایه Model types کلیک نمایید سپس در بخش Model type بر روی گزینه Conceptual Data Model کلیک نمایید و در بخش Diagram گزینه Conceptual Diagram را انتخاب نموده و سپس در فیلد Model name یک نام دلخواه را برای مدل مفهومی تان وارد نمایید در انتها بر روی دکمه OK کلیک نمایید.



شکل ۱-۵۲ صفحه طراحی مدل مفهومی در برنامه Power Designer

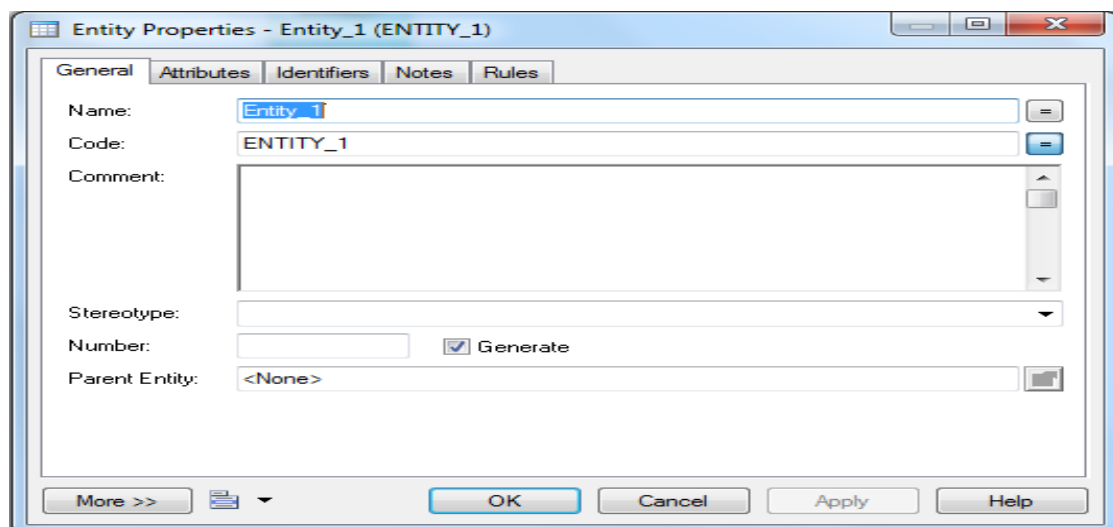
در بخش سمت راست صفحه نوار ابزار Palette قرار دارد. چنانچه به دلیلی این نوار ابزار را مشاهده نمی نمایید یا خودتان آن را بستید ابتدا از منوی Tools گزینه Customize Toolbars... را انتخاب نمایید. سپس در پنجره Toolbars نوار Scrollbar را تا به انتها به پایین آورده و سپس علامت تیک (Check) مربوط به نوار ابزار Palette را بزنید. نوار ابزار Palette نمایش داده خواهد شد.

حال اجازه دهید یک مدل مفهومی ترسیم نماییم. اولین گام افزودن یک یا چند موجودیت (Entity) است. در شکل زیر ۴ موجودیت به مدل اضافه شده است.



شکل ۱-۵۳ افزودن موجودیت به مدل داده مفهومی

پس از افزودن تعداد مورد نظر از موجودیت‌ها بر روی آیکون اشاره‌گر از نوار ابزار Palette کلیک نمایید. حال می‌بایست برای هر یک از موجودیت‌ها صفات خاصه (Attribute) تعریف نماییم بدین منظور بر روی موجودیت مورد نظر دابل کلیک نمایید.

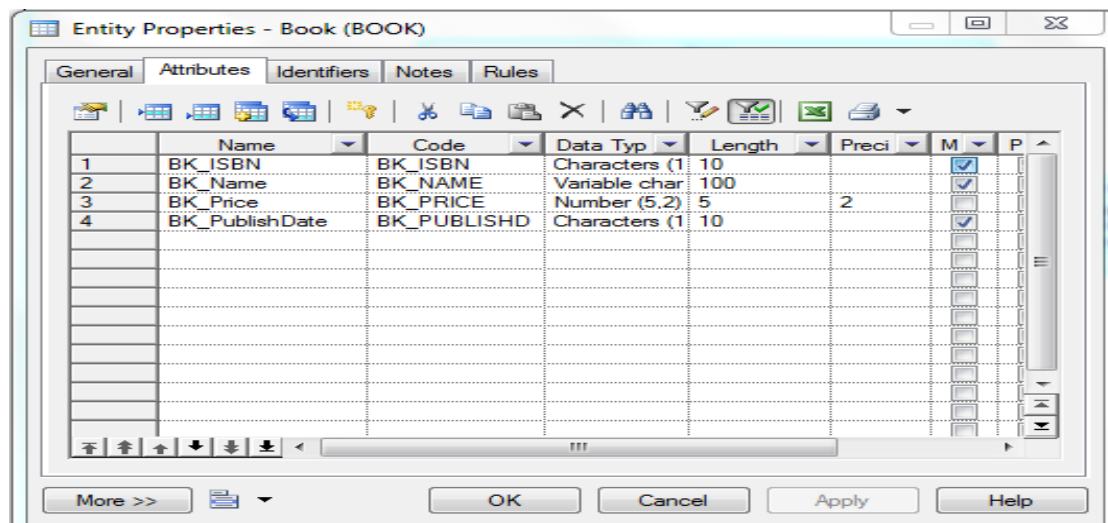


شکل ۱-۵۴ پنجره Entity Properties یک موجودیت نوعی (سربرگ General)

در سربرگ اول (General) می‌توانید نام موجودیت را تغییر دهید.

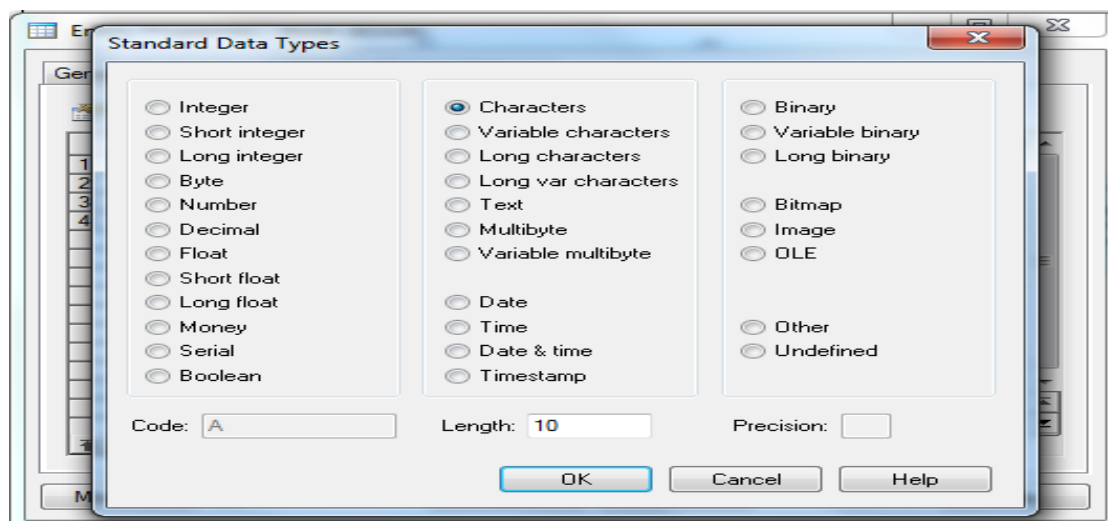
نکته : در زمان کار با نرم‌افزار Power Designer و انتخاب یا تغییر اسامی موجودیت‌ها، صفات خاصه، کلیدها و ... همیشه از روشن بودن کلید Caps Lock اطمینان حاصل نمایید.

در سربرگ Attributes می‌توانید برای این موجودیت صفات خاصه تعریف نمایید.



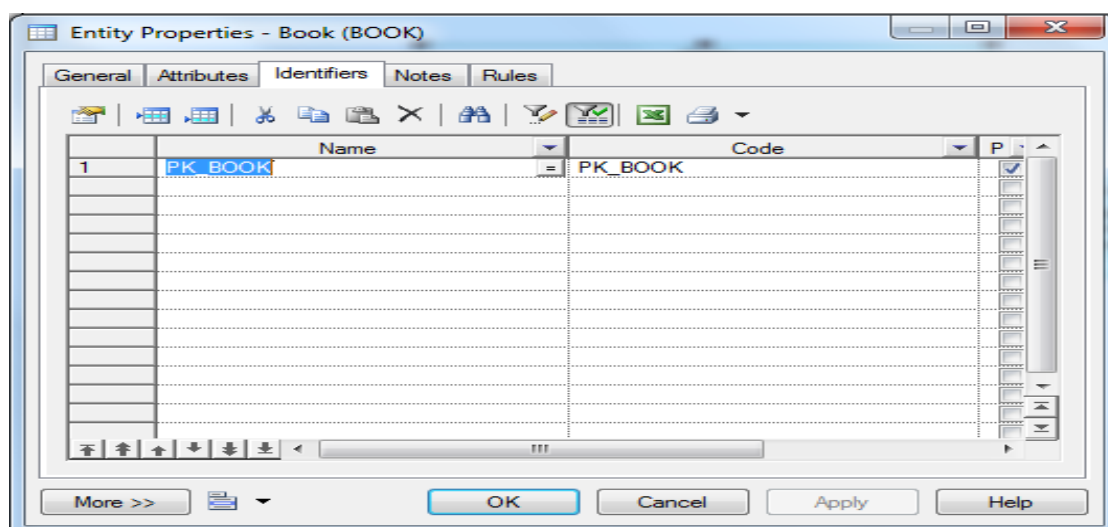
شکل ۵۵-۱ پنجره Entity Properties یک موجودیت نوعی (سربرگ Attributes)

در زمان وارد نمودن صفات خاصه برای موجودیت، در فیلد Data Type کاربر این امکان را خواهد داشت تا مطابق شکل زیر از میان نوع‌های داده‌ای موجود، نوع داده‌ای مورد نظرش را انتخاب نماید.



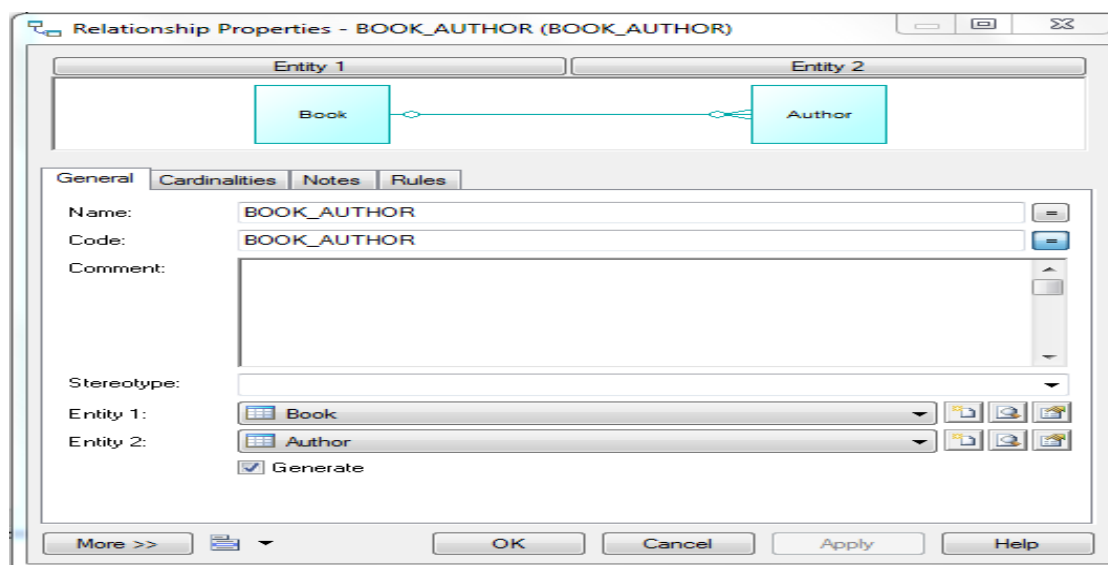
شکل ۵۶-۱ پنجره Standard Data Types

در سربرگ Identifiers می‌توانید برای هر موجودیت کلید اصلی تعریف نمایید.



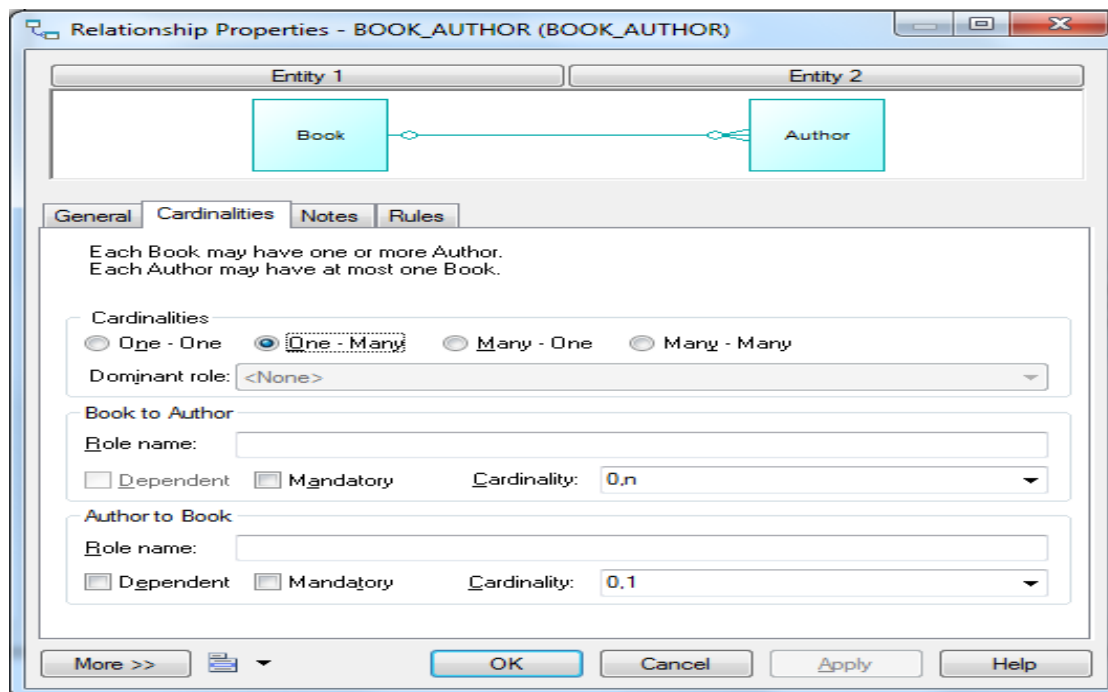
شکل ۱-۵۷ پنجره Entity Properties یک موجودیت نوعی (سربرگ Identifies)

برای مابقی موجودیت‌ها نیز به همین صورت عمل نموده و صفات خاصه و کلید اصلی تعریف می‌نماییم. سپس می‌بایست بین موجودیت‌ها، رابطه (Relationship) تعریف نمود. بدین منظور در نوار ابزار Palette بر روی گزینه Relationship کلیک نمایید. سپس اشاره‌گر را بر روی موجودیت اول قرار داده و کلیک چپ ماوس را فشرده و نگاه دارید در همین حالت اشاره‌گر را بر روی موجودیت دوم برده و رها نمایید. بدین ترتیب یک رابطه بین این دو موجودیت برقرار می‌شود. با دابل کلیک بر روی رابطه پنجره ویژگی‌های رابطه (Relationship Properties) ظاهر می‌گردد. در سربرگ اول (General) نام رابطه و ترتیب موجودیت‌ها در رابطه (موجودیت اول و موجودیت دوم) را تعیین نمایید.



شکل ۱-۵۸ پنجره Relationship Properties (سربرگ General)

در سربرگ دوم (Cardinalities) میتوانید نوع کاردینالیتی رابطه اعم از یک به یک، یک به چند، چند به یک یا چند به چند را تعیین نمایید.



شکل ۱-۵۹ پنجره Relationship Properties (سربرگ Cardinalities)

جلسه دوم : ایجاد مدل مفهومی و فیزیکی در Power Designer**۱-۲ هدف:**

در جلسه اول به صورت مقدماتی با نحوه کار با نرم افزار Power Designer آشنا شدیم. در این جلسه قصد داریم برای یک مجموعه از جداول که نشانگر یک سیستم فروش کتاب است مدل مفهومی ترسیم نماییم و سپس مدل فیزیکی آن را ایجاد نماییم.

۲-۲ مقدمه:

منظور از طراحی پایگاه داده بدست آوردن جداول، فیلدها و ستونهای آنها است. به طور معمول ۳ مدل متفاوت برای پایگاه داده طراحی می شود.

۱. مدل مفهومی (Conceptual Model)

۲. مدل منطقی (Logical Model)

۳. مدل فیزیکی (Physical Model)

از آنجایی که در این جلسه بر روی طراحی مدل مفهومی تمرکز داریم شرح مختصری از این مدل و تکنیکهای مورد استفاده در طراحی آن را ارائه می نماییم.

در طراحی مدل مفهومی نیازمندیهای کاری سیستم پایگاه داده مشخص می شود. این مدل ورودی طراحی منطقی یا فیزیکی می باشد.

برای طراحی مفهومی می توان از یکی از ۴ تکنیک ذیل استفاده نمود :

۱. Object Role Modeling (ORM)

۲. Unified Modeling Language (UML)

۳. Object Oriented System Model (OSM)

۴. Entity-Relationship Model (ER)

روشی که ما از آن استفاده می نماییم، روش ER است. در این روش به مولفه های ذیل نیاز داریم :

۱. **موجودیت (Entity)** : منظور از موجودیت اشیا دنیای واقعی هستند که می خواهیم درباره آنها داده هایی را

نگهداری نماییم. مانند موجودیت محصول، فاکتور و ...

۲. **صفت (Attribute)** : منظور از صفات، ویژگیهای یک موجودیت هستند. مانند شناسه محصول برای موجودیت محصول

۳. **رابطه (Relationship)** : ارتباطات بین موجودیتها با یکدیگر با کمک رابطهها انجام میگیرد.

حال به بررسی برخی نکات مرتبط با زبان SQL میپردازیم. همانطور که از درس اصول طراحی پایگاه دادهها می دانیم دادهها به دو دسته تقسیم می شوند : ۱. دادههای واقعی ۲. متا دادهها

همانطور که ممکن است بدانید Syntax زبان SQL به حروف بزرگ و کوچک حساس نیست، ولی نکته مهم این است که دادههای موجود در پایگاه داده به بزرگ و کوچک بودن حروف حساس هستند. برای مثال عبارت زیر را در نظر بگیرید:

Select * From TITLES

نحوه بزرگ و نوشتن کلمات رزرو شده عبارت بالا مانند Select و From اهمیتی ندارد ولی در مورد نام جدول Titles می بایست کمی توضیح دهیم. در SQL Server (Microsoft شرکت DBMS) اسامی جداول که جزء متادادهها هستند کاملاً با حروف کوچک (Lowercase) ذخیره می شوند. در ORACLE (DBMS شرکت ORACLE) تمامی مقادیر متادادهها با حروف بزرگ (Uppercase) ذخیره می شوند. در مورد MySQL (DBMS شرکت ORACLE) اسامی متاداده به همان صورتی که کاربر وارد می کند ذخیره شده و در موقع بازیابی نیز می بایست با همان نحوه نوشتار مورد بازیابی قرار گیرند.

اگر بخواهیم به طور خلاصه بگوییم SQL Server متاداده را کاملاً با حروف کوچک ذخیره می نماید، ORACLE متاداده را با حروف بزرگ ذخیره می نماید و در مورد MySQL نیز متادادهها دقیقاً به همان صورتی که کاربر وارد می نماید ذخیره می شوند.

در آخر قبل از شرح آزمایش این جلسه با برخی انواع دادهای متداول در MS SQL Server آشنا می شویم.

▪ انواع دادهای عددی :

جدول ۱-۲ انواع دادهای عددی

نوع	حجم
TINYINT	۱ بایت
SMALLINT	۲ بایت
INT	۴ بایت

BIGINT	۸ بایت
BIT	(True or False)
MONEY	۸ بایت
SMALLMONEY	۴ بایت
DECIMAL	از ۵ تا ۱۷ بایت
NUMERIC	از ۵ تا ۱۷ بایت

▪ انواع داده‌ای عددی تقریبی :

جدول ۲-۲ انواع داده‌ای عددی تقریبی

نوع	حجم
FLOAT	از ۴ تا ۸ بایت
REAL	۴ بایت

▪ انواع داده‌ای Date & Time :

جدول ۳-۲ انواع داده‌ای Date & Time

نوع	حجم
DATE	۳ بایت
SMALLDATETIME	۴ بایت
TIME	۵ بایت
DATETIME	۸ بایت

▪ انواع داده‌ای کاراکتری (غیر Unicode) :

جدول ۴-۲ انواع داده‌ای کاراکتری غیر Unicode

نوع	حجم
CHAR	متناظر با طول رشته از ۱ تا ۸۰۰۰ بایت
VARCHAR	طول واقعی داده‌ها + ۲ بایت (حداکثر ۲ گیگابایت)
TEXT	طول متغیر (حداکثر ۲ گیگابایت)

▪ انواع داده‌ای کاراکتری Unicode :

جدول ۲-۵ انواع داده‌ای کاراکتری Unicode

نوع	حجم
NCHAR	بسته به تعداد کاراکترها (بین ۱ تا ۴۰۰۰ کاراکتر) حجم برابر دو برابر تعداد کاراکترها برحسب بایت است
NVARCHAR	دو برابر (طول واقعی داده‌ها + ۲ بایت) (حداکثر ۲ گیگابایت)
NTEXT	دو برابر (تعداد کاراکترها) بر حسب بایت

▪ انواع داده‌ای باینری :

جدول ۲-۶ انواع داده‌ای باینری

نوع	حجم
BINARY	تعداد کاراکترها بر حسب بایت
VARBINARY	طول واقعی داده + ۲ بایت
IMAGE	داده باینری با صول متغیر از ۰ تا ۲ گیگابایت

۲-۳ شرح:

در ابتدا به ذکر موجودیت‌ها و صفات مورد استفاده برای این مدل مفهومی می‌پردازیم. مدل ما در این جلسه از آزمایشگاه پایگاه داده‌ها شامل موجودیت‌های نویسندگان (WRITERS)، کتب (TITLES)، مشتریان (CUSTOMERS) و فاکتورها (FCTRS) می‌باشد.

حال هر یک از موجودیت‌های مذکور در بخش مقدمه را مورد بررسی قرار می‌دهیم و با صفات هر یک آشنا می‌شویم.

جدول ۷-۲ صفات موجودیت TITLES

TITLES		
نام	نوع فیلد	حداکثر طول
TIT_ISBN	Characters	10
TIT_NAME	Variable characters	50
TIT_DESC	Variable characters	100
TIT_PUB_DT	Characters	10

جدول ۸-۲ صفات موجودیت WRITERS

WRITERS		
نام	نوع فیلد	حداکثر طول
WR_CODE	Integers	---
WR_NAME	Characters	50
WR_DEGREE	Characters	100

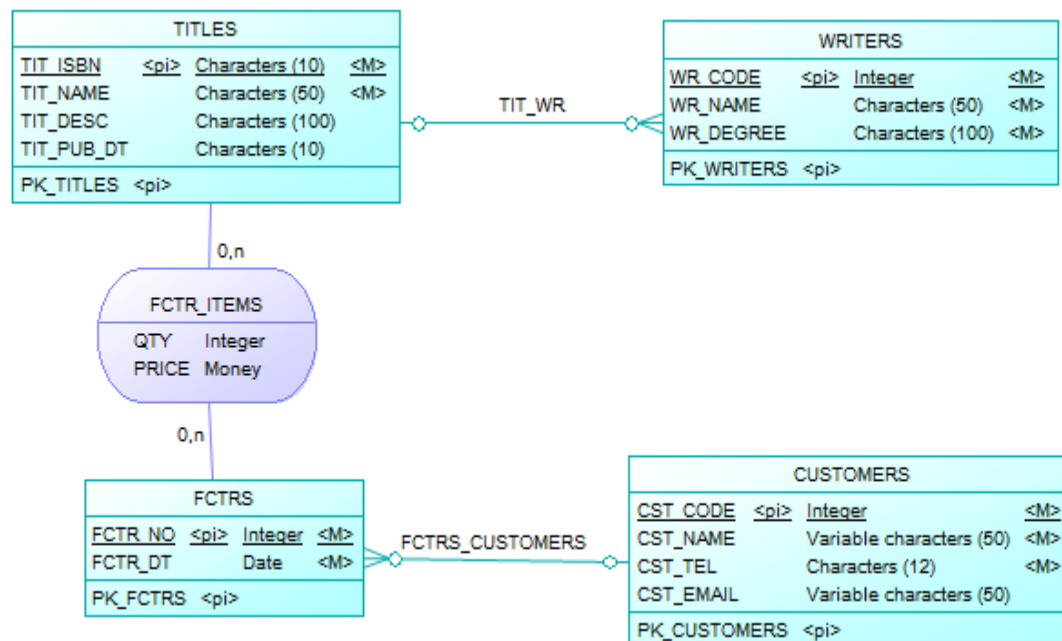
جدول ۹-۲ صفات موجودیت FCTRS

FCTRS		
نام	نوع فیلد	حداکثر طول
FCTR_NO	Integer	---
FCTR_DT	Characters	10

جدول ۱۰-۲ صفات موجودیت CUSTOMERS

CUSTOMERS		
نام	نوع فیلد	حداکثر طول
CST_CODE	Integer	---
CST_NAME	Variable characters	50
CST_TEL	Characters	12
CST_EMAIL	Variable characters	50

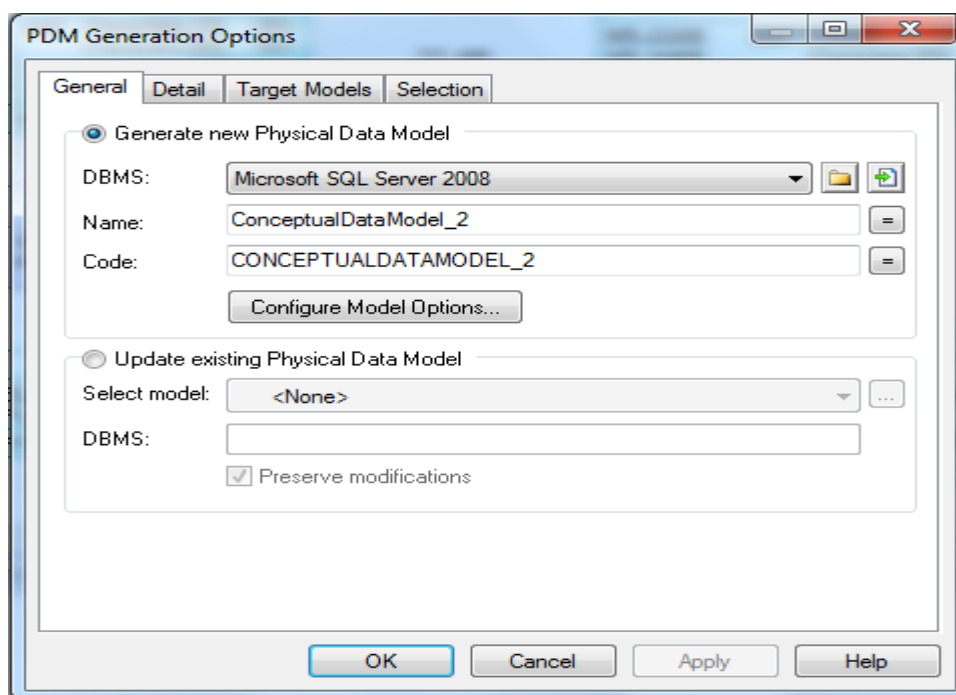
حال که با موجودیت‌ها و صفات آنان آشنا شده‌ایم به ترسیم مدل مفهومی در نرم‌افزار Power Designer می‌پردازیم.



شکل ۱-۲ یک طراحی مدل مفهومی مثالی در نرم افزار Power Designer

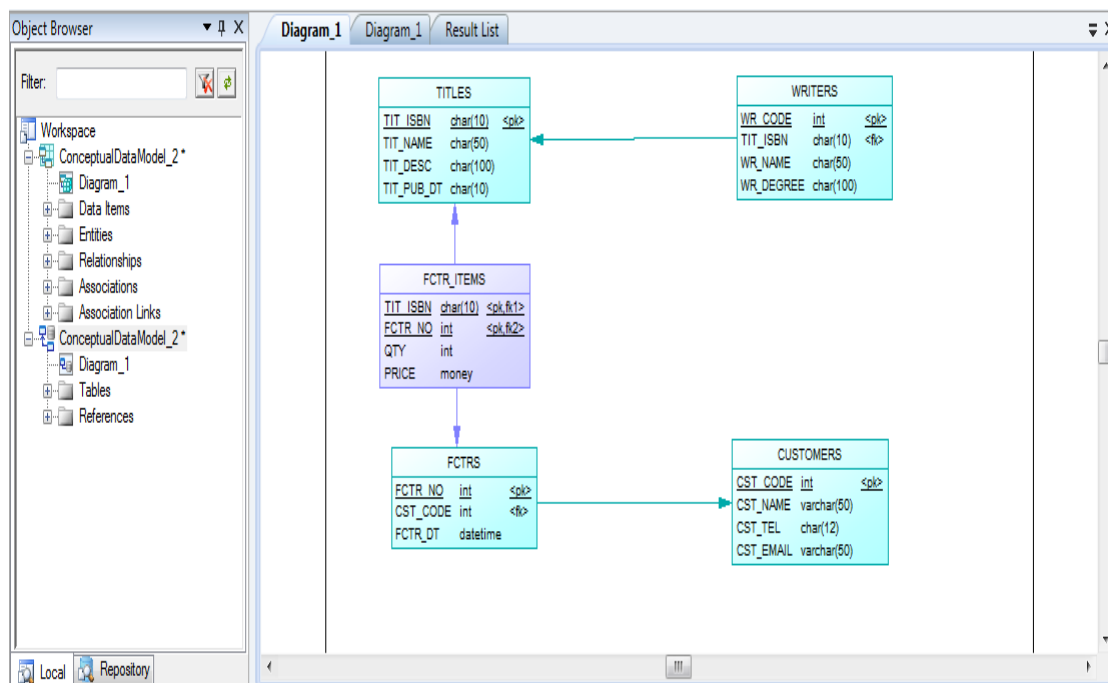
در شکل بالا عنصری جدید بنام Association بکار برده شده است. با کمی تامل درمی یابید که صفات QTY و PRICE را می توان در هر دوی موجودیت های FCTRS و TITLES استفاده نمود در عین حال این دو صفت به هیچ کدام از موجودیت های ذکر شده نیز تعلق ندارند. از این رو با افزودن یک عنصر Association به مدل، صفات QTY و PRICE را برای این Association تعریف می نماییم.

پس از تعریف عنصر Association در مدل مفهومی مان حال آماده ایم تا مدل فیزیکی را تولید نماییم. بدین منظور از منوی Tools گزینه Generate Physical Data Model... را انتخاب می نماییم. در پنجره ای که ظاهر می شود ابتدا از بخش DBMS نوع سیستم مدیریت پایگاه داده ای را که مدل فیزیکی برای آن ایجاد می شود را انتخاب می نماییم. سپس در فیلد Name یک نام دلخواه برای مدل فیزیکی مان انتخاب می نماییم.



شکل ۲-۲ پنجره PDM Generation Options

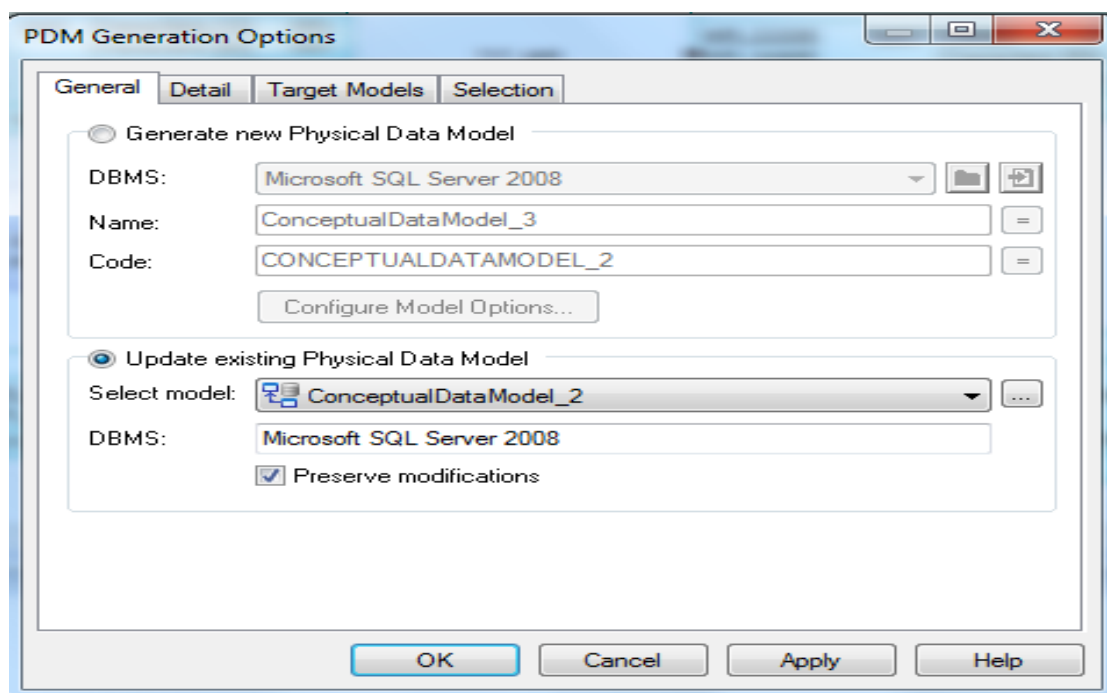
با کلیک بر روی دکمه OK مدل فیزیکی ایجاد می‌شود.



شکل ۲-۳ مدل فیزیکی تولید شده از مدل مفهومی در نرم‌افزار Power Designer

همانطور که مشاهده می‌نمایید کلیدهای اصلی دو موجودیت FCTRS و TITLES بعنوان کلیدهای خارجی در موجودیت Association قرار گرفته‌اند. زیرا عنصر Association یک موجودیت ضعیف (Weak Entity) است که از خودش کلید اصلی ندارد و با استفاده از کلیدهای اصلی موجودیتهای مرتبط که در موجودیت Association کلید خارجی هستند، کلید اصلی تشکیل می‌دهد. در این مورد دو کلید خارجی TIT_ISBN و FCTR_NO با یکدیگر کلید اصلی موجودیت ضعیف FCTR_ITEMS را تشکیل می‌دهند.

حال چنانچه بخواهیم تغییری در مدل مفهومی داشته باشیم پس از اعمال تغییر در مدل مفهومی نیازی به ساخت یک مدل فیزیکی جدید نداریم بلکه می‌توانیم مدل فیزیکی ایجاد شده را بروزرسانی نماییم.



شکل ۴-۲ پنجره PDM Generation Options - بروزرسانی مدل فیزیکی موجود

جلسه سوم : آشنایی با نرم افزار Microsoft SQL Server**۱-۳ هدف:**

در جلسات قبل با طریقه نصب نرم افزار Microsoft SQL Server آشنا شدیم. در این جلسه به کار با این نرم افزار می پردازیم.

۲-۳ مقدمه:

در این جلسه از آزمایشگاه پایگاه داده ها به مبحث نرمال سازی (Normalization) می پردازیم. نرمال سازی فرآیند سازماندهی فیلدها و جداول یک پایگاه داده رابطه ای برای به حداقل رسانیدن افزونگی ها و وابستگی ها است.

نرمال سازی دارای صورتهای مختلفی به شرح ذیل است :

- **صورت نرمال اول (1NF) :**

جداول در صورت نرمال اول قرار دارند در صورتی که تمام فیلدهای آنها تجزیه ناپذیر یا Atomic باشند.

- **صورت نرمال دوم (2NF) :**

جداول در صورت نرمال دوم خود قرار دارند اگر اولاً در صورت نرمال اول (1NF) باشند و ثانیاً تمامی فیلدهای جدول با کلید اصلی آن جدول وابستگی تابعی کامل داشته باشند.

- **صورت نرمال سوم (3NF) :**

جداول در صورت نرمال سوم (3NF) خود قرار دارند اگر اولاً در صورت نرمال دوم (2NF) قرار داشته باشند و اینکه تمامی فیلدهای جدول فقط با کلید اصلی وابستگی تابعی داشته باشند، یعنی فیلدهای جدول دو به دو با یکدیگر رابطه ای نداشته باشند.

- **صورت نرمال چهارم (4NF) :**

جداول در صورت نرمال چهارم (4NF) هستند اگر اولاً در صورت نرمال سوم (3NF) و BCNF (صورت نرمال Boyce-Codd) خود قرار داشته باشند و فاقد هرگونه وابستگی چندمقداری باشند.

حال برای تفهیم بهتر مفهوم نرمال سازی یک مثال ارائه می نماییم. فاکتور فروش زیر را در نظر بگیرید.

Sales Order

*Fiction Company
202 N. Main
Mahattan, KS 66502*

CustomerNumber:	1001	Sales Order Number:	405
Customer Name:	ABC Company	Sales Order Date:	2/1/2000
Customer Address:	100 Points	Clerk Number:	210
	Manhattan, KS 66502	Clerk Name:	Martin Lawrence

Item Ordered	Description	Quantity	Unit Price	Total
800	widgit small	40	60.00	2,400.00
801	tingimajigger	20	20.00	400.00
805	thingibob	10	100.00	1,000.00
Order Total				3,800.00

فیلدهایی که می بایست در مورد این فاکتور فروش ذخیره شوند به صورت زیر هستند. در ابتدا تمامی فیلدهایی که می بایست ذخیره شوند را بدون هیچ گونه جدول بندی بصورت زیر در کنار هم قرار می دهیم.

SalesOrderNumber, Date, CustomerNumber, CustomerName, CustomerAddress, ClerkNumber, ClerkName, ItemNumber, Description, Qty, UnitPrice

■ صورت نرمال اول:

در صورت اول نرمال سازی فیلدهایی که برای یک موجودیت چندین بار تکرار می شوند را جدا نموده و در جداولی جدید قرار می دهیم. کلید اصلی جدول جدیدی (جدولی که فیلدهای تکرارشونده در آن قرار داده می شوند) که ساخته می شود یک کلید مرکب خواهد بود که در مورد فاکتور فروش بالا کلید مرکب می تواند شامل شماره سند (SalesOrderNumber) بعلاوه ی فیلدی دیگر که به طور منحصر بفرد ردیفهای جدول جدید را متمایز می نماید باشد. در مورد این فاکتور فروش مثلاً فیلد ItemNumber.

در مورد فاکتور فروش بالا می توان فیلدها را به دو جدول تقسیم نمود.

فیلدهای زیر فیلدهایی هستند که برای هر ردیف تکرار می شوند پس می بایست در جدولی جداگانه قرار داده شوند.

SalesOrderNumber, ItemNumber, Description, Qty, UnitPrice

همانطور که در توضیح صورت نرمال سازی اول آمد کلید اصلی این جدول جدید شامل شماره سند بعلاوه ی شماره آیتم باشد که با یکدیگر یک کلید اصلی مرکب را تشکیل می دهند. نحوه تشخیص فیلدهای تکرارشونده در مورد فاکتور فروش این است که فیلدهای ItemNumber, Description, Qty, UnitPrice برای هر آیتم شامل در فاکتور تکرار می شوند. پس این فیلدها می بایست در جدولی جدید قرار داده شوند. فیلد SalesOrderNumber نیز بعنوان فیلدی که بصورت منحصر بفرد فاکتور فروش را شناسایی می نماید در این جدول قرار داده می شود تا به همراه فیلد ItemNumber یک کلید اصلی مرکب را تشکیل دهند.

با جداسازی فیلدهای تکرارشونده فیلدهای ذیل باقی می ماند.

SalesOrderNumber, Date, CustomerNumber, CustomerName, CustomerAddress, ClerkNumber, ClerkName

اگر صورت اول نرمال سازی را انجام نمی دادیم فیلدهای شماره سفارش فروش، تاریخ، شماره مشتری، نام مشتری، آدرس مشتری، شماره کارمند و نام کارمند برای هر آیتم در فاکتور تکرار می شدند.

■ صورت نرمال دوم :

در صورت دوم نرمال سازی وابستگی های تابعی جزئی را حذف می نماییم.

وابستگی تابعی به این معناست که مقدار صفتی در جدول کاملاً بواسطه صفتی دیگر تعیین شود.

وابستگی تابعی جزئی نوعی از وابستگی تابعی است که صفت فقط با بخشی و نه کل کلید اصلی وابستگی تابعی دارد. (در این حالت کلید اصلی مرکب است و از بیش از یک بخش تشکیل شده است). برای حذف وابستگی تابعی جزئی از جداول، فیلدهایی را که با بخشی و نه کل کلید اصلی وابستگی دارند را همراه با آن بخش از کلید اصلی که با آن وابستگی تابعی دارند به جدولی جدید منتقل می نماییم.

می توانیم فیلد Description را از جدول آن جدا نماییم زیرا با کلید اصلی وابستگی تابعی کامل ندارد بلکه فقط با فیلد ItemNumber وابستگی تابعی جزئی دارد. پس جدولی جدید شامل این دو فیلد

ایجاد می نماییم. البته فیلد **ItemNumber** را از جدول اصلی حذف نمی نماییم زیرا برای تمایز کلیدها در آن جدول لازم است. جدول جدید شامل فیلدهای زیر خواهد بود :

ItemNumber, Description

ممکن است فکر کنید که فیلد **UnitPrice** را هم می توان جدا نمود ولی دقت داشته باشید که فیلد قیمت واحد می تواند برای هر فاکتور فروش متفاوت باشد (بر حسب تخفیف، مشتریان خاص و ...)

با این حساب تا به اینجا کار و در پایان صورت دوم نرمال سازی سه جدول به شرح ذیل خواهیم داشت :

- 1 SalesOrderNumber, Date, CustomerNumber, CustomerName, CustomerAddress, ClerkNumber, ClerkName
- 2 SalesOrderNumber, ItemNumber, Qty, UnitPrice
- 3 ItemNumber, Description

اگر صورت دوم نرمال سازی را انجام نمی دادیم با مشکلات زیر مواجه می شدیم :

تکرار داده ها - فیلد **Description** در هر سفارش آن آیتم ظاهر می شد.

آنومالی درج - برای درج یک آیتم جدید در انبار می بایست برای فیلد شماره سفارش فروش نیز مقدار وارد نمایید و در واقع برای درج یک آیتم جدید در انبار می بایست برای آن سفارش فروش ثبت نمایید.

آنومالی بروزرسانی - برای تغییر فیلد **Description** می بایست این تغییر را در هر سفارش فروش اعمال نمایید.

آنومالی حذف - تمامی اطلاعات آیتم های انبار در جدول جزئیات سفارش فروش ذخیره شده اند و چنانچه سفارش فروشی حذف شود آیتم های آن نیز حذف می شوند.

▪ صورت نرمال سوم :

وابستگی های انتقالی را حذف می نماییم.

وابستگی انتقالی نوعی از وابستگی تابعی است که یک صفت با صفتی به غیر از کلید اصلی وابستگی تابعی داشته باشد. بنابراین مقدار آن صفت به صورت غیر مستقیم توسط کلید اصلی تعیین می شود. در این صورت آن صفت و فیلدهایی که با آن وابستگی تابعی دارند را در جدولی جدید قرار می دهیم البته یک کپی از صفت کلید را در جدول اولیه اش باقی می گذاریم.

با توجه به جداول حال حاضر جداول جدید بصورت زیر ایجاد می شوند :

CustomerNumber, CustomerName, CustomerAddress

ClerkNumber, ClerkName

تمامی فیلدهای دو جدول بالا به استثنای فیلدهایی که در این جداول کلید اصلی هستند را از جدول اولیه شان منتقل می نماییم ولی در مورد کلیدهای اصلی جداول جدید برای امکانپذیر بودن ارتباط داده ها با یکدیگر آنها را در جدول اولیه شان باقی می گذاریم. جدول اولیه مربوط به دو جدول بالا بصورت زیر درمی آید :

SalesOrderNumber, Date, CustomerNumber, ClerkNumber

در پایان صورت سوم نرمال سازی جداول پایگاه داده بصورت ذیل خواهند بود :

1	Customer	<u>CustomerNumber</u> , CustomerName, CustomerAddress
2	Clerk	<u>ClerkNumber</u> , ClerkName
3	Inventory Items	ItemNumber, Description
4	Sales Orders	<u>SalesOrderNumber</u> , Date, CustomerNumber, ClerkNumber
5	Sales Order Detail	<u>SalesOrderNo</u> , <u>ItemNo</u> , Qty, UnitPrice

اگر صورت سوم نرمال سازی را انجام نمی دادیم با مشکلات زیر مواجه می شدیم :

تکرار داده ها – جزئیات مشتری و کارمند بر روی هر سفارش فروش ظاهر می شدند.

آنومالی حذف – حذف یک سفارش فروش باعث حذف کارمند و مشتری می شد.

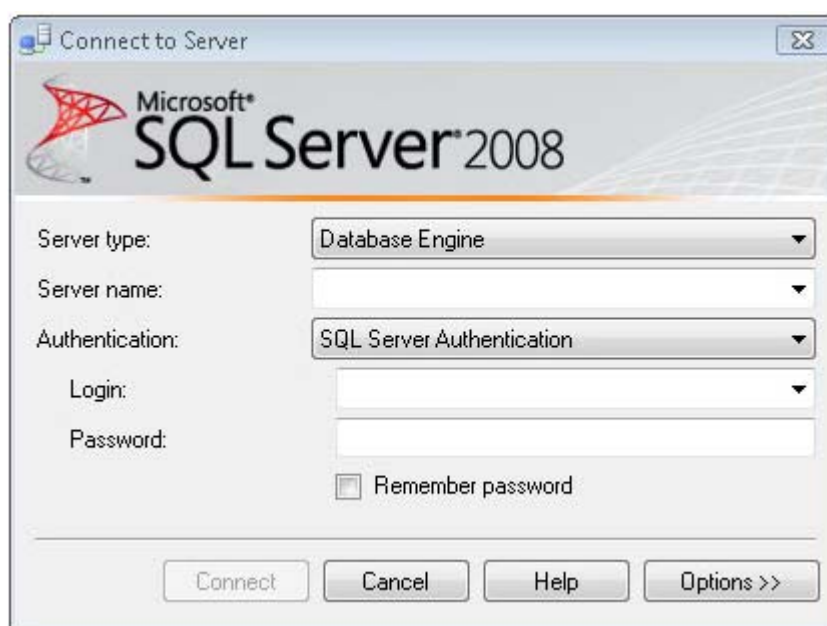
آنومالی درج – برای درج یک مشتری یا کارمند جدید می بایست یک سفارش فروش ثبت می نمودیم.

آنومالی بروزرسانی – برای تغییر نام، آدرس و ... باید این فیلدها را بر روی تمامی سفارش های فروش تغییر می دادیم.

۳-۲ شرح:

در این جلسه ابتدا مراحل اتصال به سرویسگر پایگاه داده و سپس پایگاههای داده که بصورت پیش فرض در نرم افزار Microsoft SQL Server موجود هستند را تشریح می نماییم. پس از آشنایی با پایگاههای داده پیش فرض، یک پایگاه داده جدید با نام دلخواهمان ایجاد می نماییم. سپس مراحل پشتیبان گیری از پایگاه داده ایجاد شده را تشریح می نماییم.

پس از باز کردن نرم افزار Microsoft SQL Server پنجره Connect to Server برای اتصال به سرویسگر پایگاه داده ظاهر می شود:



شکل ۳-۱ پنجره اتصال به سرویسگر پایگاه داده

در فیلد Server type گزینه Database Engine را انتخاب نمایید. در فیلد Server name می بایست نام سرویسگری را که قصد دارید به آن متصل شوید را وارد نمایید. در صورتی که سرویسگر همان کامپیوتر خودتان باشد می توانید در فیلد سرویسگر یکی از سه مورد زیر را وارد نمایید:

۱. نقطه (.)

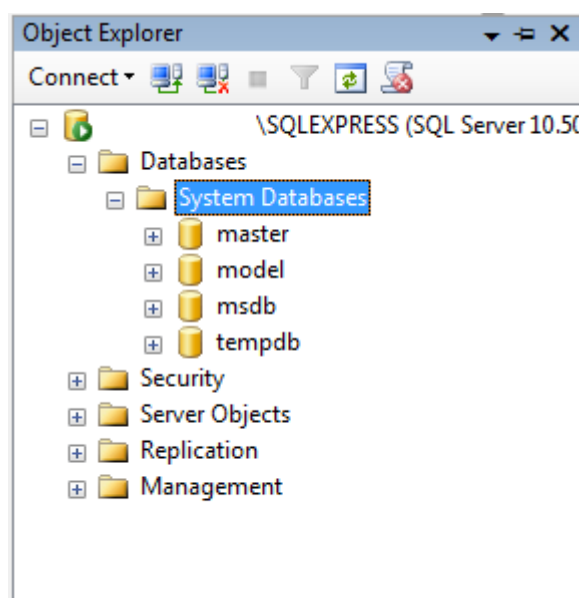
۲. localhost

۳. 127.0.0.1

تمامی موارد ذکر شده در بالا دارای معانی یکسان هستند، که به معنی اتصال به کامپیوتر جاری می باشند. همچنین می توانید در فیلد نام سرویسگر نام کامپیوتر را وارد نمایید.

در مورد فیلد Authentication (هویت شناسی) می بایست نحوه تایید هویت را تعیین نمایید. در صورتی که در زمان نصب نرم افزار Microsoft SQL Server در تعیین نحوه هویت شناسی گزینه Mixed Mode را انتخاب نموده باشید در زمان استفاده از نرم افزار می توانید به دو شیوه وارد محیط کاربری گردید. در شیوه اول تایید هویت می تواند بواسطه نام کاربری و رمز عبور حساب کاربری ویندوز انجام گیرد، در مورد شیوه دوم ورود به محیط کاربری با استفاده از نام کاربری و رمز عبوری که مختص پایگاه داده است انجام گیرد. (نام کاربری پیش فرض در SQL Server نام کاربری sa است).

پس از ورود به محیط کاربری نرم افزار Microsoft SQL Server در پنل Object Explorer با باز کردن فولدر Databases و سپس فولدر System Databases می توانید پایگاههای داده پیش فرض را مشاهده نمایید.



شکل ۳-۲ پنل Object Explorer

در شکل قبل پایگاههای داده سیستمی پیش فرض نشان داده شده است. در ادامه به صورت خلاصه عملکرد هر کدام از این پایگاههای داده پیش فرض را تشریح می نماییم.

▪ پایگاه داده master :

این پایگاه داده محتوی اطلاعات متاداده ها (داده های درباره داده های واقعی) و لغت نامه داده (Data Dictionary) می باشد. اسامی فیلدها و جداول، نوع های داده ای، طول های نوع های داده ای، اسامی کاربران و سطوح دسترسی آنان و ... که داده های واقعی ما نیستند همگی متاداده هستند و در پایگاه داده master ذخیره می شوند.

▪ پایگاه داده model :

زمانی که در نرم افزار Microsoft SQL Server پایگاه داده جدیدی ایجاد می نماییم این پایگاه داده جدید با الگوبرداری از پایگاه داده model ایجاد می شود.

▪ پایگاه داده msdb :

برای انجام Job ها یا کارهای گروهی از پایگاه داده msdb استفاده می شود.

▪ پایگاه داده tempdb :

این پایگاه داده یک پایگاه داده موقتی است که برای نگهداری object های کاربری موقتی و نیز object های موقتی خود نرم افزار SQL Server مورد استفاده قرار می گیرد. این پایگاه داده یک پایگاه داده میانی است. همچنین برای مرتب سازی موقت داده ها از این پایگاه داده استفاده می شود.

حال که با پایگاههای داده پیش فرض آشنا شدیم می خواهیم تا یک پایگاه داده جدید ایجاد نماییم. به این منظور بر روی فولدر Databases کلیک راست نموده و گزینه New Database... را انتخاب نمایید. در پنجره New Database مطابق شکل زیر با رعایت قوانین نامگذاری نامی دلخواه برای پایگاه داده انتخاب می نماییم.

Logical Name	File Type	Filegroup	Initial Size (MB)	Autogrowth
MyDB	Rows ...	PRIMARY	3	By 1 MB, unrestricted growth
MyDB_log	Log	Not Applicable	1	By 10 percent, unrestricted growth

شکل ۳-۳ ایجاد پایگاه داده جدید

حال می خواهیم ببینیم پایگاه داده ایجاد شده در کدام بخش از هارد دیسک سیستم ذخیره شده است. بدین منظور از بخش Computer ویندوز مسیر زیر را طی نمایید:

C:\Program Files\Microsoft SQL Server\MSSQL10_50.SQLEXPRESS\MSSQL\DATA

با طی نمودن مسیر بالا به فولدري می رسید که تمامی پایگاههای داده در آنجا ذخیره شده اند. همانطور که مشاهده می نمایید برای هر پایگاه داده دو نوع فایل ذخیره شده است. فایل اول با پسوند .mdf و فایل دیگری با پسوند .ldf.

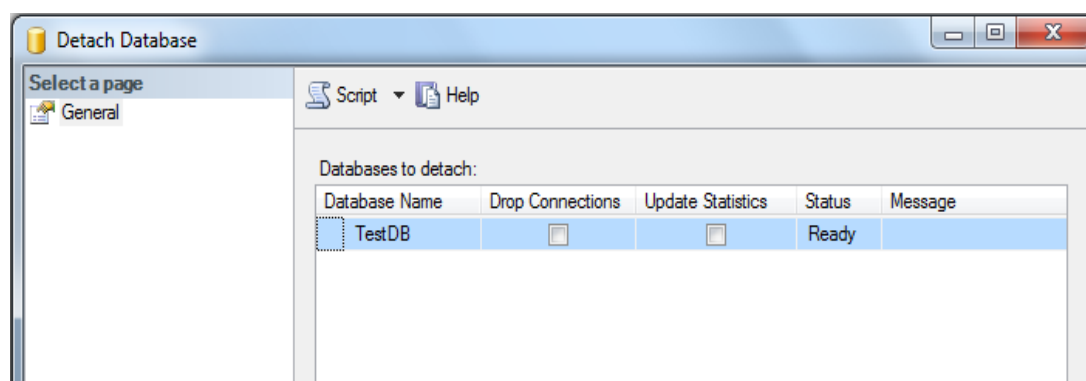
نکته : چنانچه پسوندهای فایلها قابل مشاهده نیستند در یکی از پنجره های ویندوز و از منوی Tools گزینه Folder options... را انتخاب نمایید سپس در پنجره ظاهر شده از سربرگ View چک گزینه Hide extensions for known file types را پاک کنید و بر روی دکمه Ok کلیک نمایید.

برای نمونه چنانچه پایگاه داده ای با نام Test ایجاد نماییم دو فایل با نامهای Test.mdf و Test_log.ldf برای آن ساخته خواهد شد.

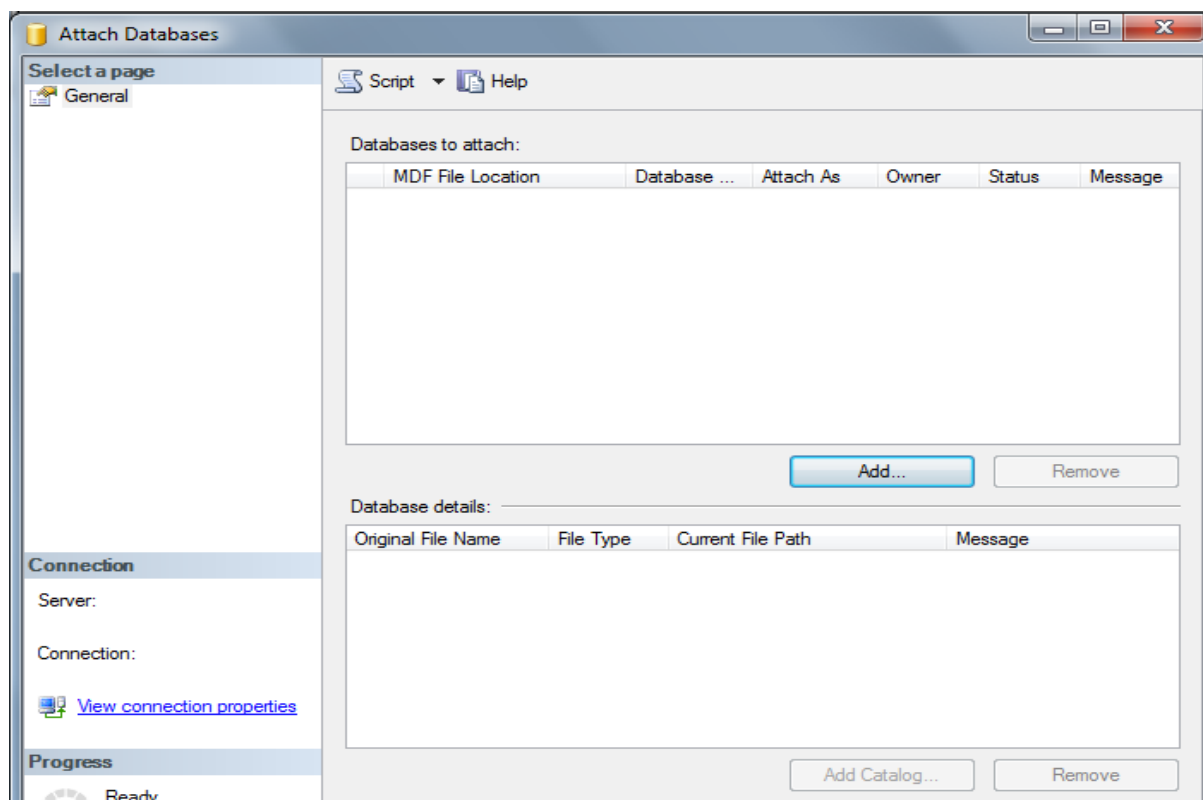
نکته : پسوند .mdf مخفف Master Database File است و پسوند .ldf مخفف Log Database File می باشد. فایل با پسوند .ldf فایل ثبت وقایع است یعنی تمامی تغییرات انجام شده بر روی پایگاه داده در این فایل ذخیره می شود. فایل با پسوند .mdf محتوی داده های متاداده (تعاریف اشیاء پایگاه داده) و داده های واقعی می باشد.

چنانچه سعی کنید در فولدري که فایل های پایگاه داده ذخیره شده اند اقدام به کپی کردن فایلها نمایید خطایی مبنی بر اینکه SQL Server در حال استفاده از فایل می باشد دریافت خواهید نمود. برای حل این مشکل ابتدا می بایست کاری کنیم تا SQL Server دیگر از آن پایگاه داده استفاده ننماید، بدین منظور در محیط SQL Server Management Studio بر روی نام پایگاه داده مورد نظر کلیک راست نموده و از منوی Tasks گزینه Detach را انتخاب نمایید.

پنجره ای مطابق شکل زیر ظاهر می شود، چک های مربوط به ستونهای Drop Connections و Update Statistics را بزنید و بر روی دکمه Ok کلیک نمایید.



چنانچه پنل Object Explorer را مشاهده نمایید متوجه می شوید که پایگاه داده مورد نظر دیگر در لیست پایگاههای داده در زیر فولدر Databases لیست نشده است. این به معنی حذف شدن پایگاه داده از سیستم شما



نیست بلکه فقط به این معنی است که نرم افزار SQL Server دیگر با این پایگاه داده کاری نخواهد داشت و حالا شما می توانید فایل های آن پایگاه داده بر روی هارد دیسک کامپیوترتان را برای مثال کپی نمایید.

چنانچه زمانی تصمیم بگیرید که مجدداً از این پایگاه داده در محیط SQL Server Management Studio استفاده نمایید می بایست با کلیک راست بر روی فولدر Databases از پنل Object Explorer، بر روی گزینه Attach... کلیک نمایید. پنجره ای مطابق شکل زیر ظاهر می شود:

بر روی دکمه Add کلیک می نماییم و فایل پایگاه داده مورد نظرمان (با پسوند .mdf) را انتخاب می نماییم. فایل انتخابی ما در بخش Databases to attach نمایش داده می شود، سپس SQL Server خودش فایل با پسوند .ldf متناظر با فایل انتخابی ما را یافته و در بخش Database details نمایش می دهد. با کلیک بر روی دکمه Ok پایگاه داده به لیست پایگاههای داده در زیر فولدر Databases در پنل Object Explorer افزوده خواهد شد و مجدداً می توانید توسط SQL Server با این پایگاه داده کار کنید.

چنانچه بر روی علامت + کنار نام پایگاه داده کلیک نمایید تعدادی فولدر در زیر آن پایگاه داده نمایش داده می شوند. این فولدرها محتوی Object های پایگاه داده هستند. Object ها برای ذخیره داده ها یا ارجاع به داده ها مورد استفاده قرار می گیرند. برای مثال Table ها شیء ها (Object هایی) هستند که برای ذخیره داده ها مورد استفاده قرار می گیرند.

در مورد فولدر Views می بایست ذکر گردد که این فولدر محتوی Select های ذخیره شده در پایگاه داده که جهت ارجاع به داده ها مورد استفاده قرار می گیرند.

برای ایجاد جدول در پایگاه داده بر روی فولدر Tables کلیک راست نموده و گزینه New Table... را انتخاب نمایید.

در پنجره ظاهر شده مطابق شکل زیر در ستون Column Name می بایست نام ستون مورد نظر را وارد نمایید. در ستون Data Type می بایست نوع داده ای برای ستون مورد نظر را وارد نمایید. ستون Allow Nulls هم مربوط به این است که آیا این ستون می تواند بدون هیچ مقداری، خالی رها شود یا خیر.

Column Name	Data Type	Allow Nulls
<input type="text"/>		<input type="checkbox"/>

بعد از تعریف ستونهای مورد نظر برای جدول آن را ذخیره نمایید. چنانچه علامت + در کنار فولدر Tables در زیر پایگاه داده مورد نظر را مشاهده نمایید متوجه می شوید که جدولی که به تازگی ایجاد نمودید در زیر فولدر Tables پایگاه داده تان لیست شده است.

جلسه چهارم : آشنایی با Constraint ها در Microsoft SQL Server

۴-۱ هدف:

در این جلسه از آزمایشگاه پایگاه داده‌ها به بررسی قوانین یا همان Constraint های موجود در Microsoft SQL Server می‌پردازیم.

۴-۲ مقدمه:

در ابتدا شرحی مختصر از انواع قوانین جامعیت ارایه می‌نماییم. این قوانین حفظ صحت، امنیت و جامعیت پایگاه داده را تضمین می‌نمایند. این قوانین بر دو نوع هستند :

۱. قوانین جامعیت عام
۲. قوانین جامعیت خاص

۱. قوانین جامعیت عام :

قوانینی هستند که در تمامی پایگاه‌های داده صادق هستند. این قوانین به دو گروه تقسیم می‌شوند:

۱. قوانین جامعیت موجودیتی (Entity Integrity Rules) :

هیچ بخشی از کلید اصلی نمی‌بایست Null باشد.

۲. قوانین جامعیت ارجاعی (Referential Integrity Rules) :

اگر دو جدول Table1 و Table2 را داشته باشیم و فیلد f1 در جدول Table2 کلید خارجی باشد حال هر مقداری که این فیلد می‌تواند داشته باشد می‌بایست در جدول Table1 موجود باشد در غیر این حالت مقدار فیلد f1 در جدول Table2 می‌بایست Null باشد.

۲. قوانین جامعیت خاص :

قوانینی هستند که حوزه اعمال آنها به یک پایگاه داده خاص محدود است.

در ادامه به تشریح انواع Constraint های موجود در Microsoft SQL Server می پردازیم، سپس هر یک از قوانین را به جزئیات تشریح نموده و برای آن مثال ارائه می نماییم. از Constraint ها برای حفظ جامعیت پایگاه داده استفاده می نماییم.

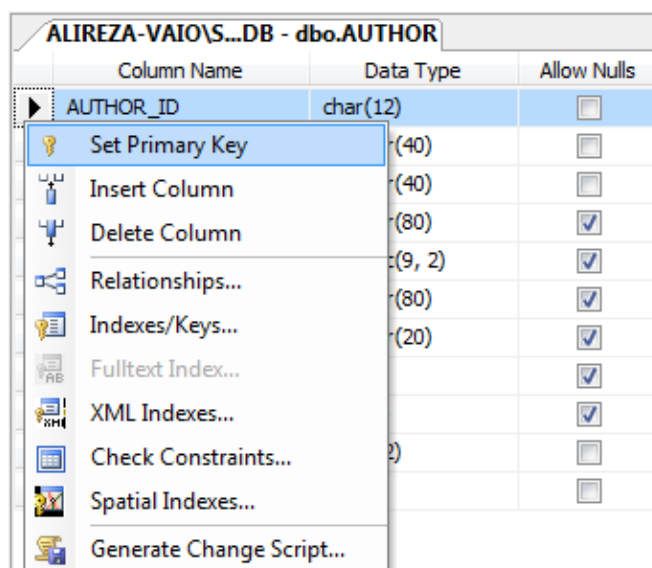
۴-۳ شرح:

انواع قوانین یا Constraint های موجود در Microsoft SQL Server عبارتند از:

- Primary Key
- Foreign Key
- Unique Key
- Check Constraints

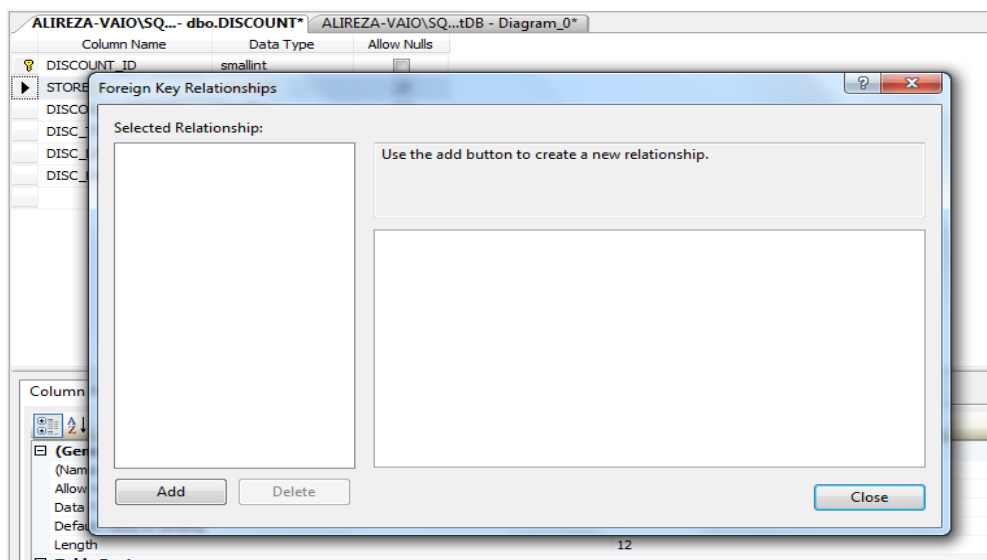
▪ Primary Key :

از جلسات قبلی آزمایشگاه پایگاه داده ها با نحوه تعریف کلید اصلی آشنا هستیم. با این حال مجدداً نحوه ایجاد کلید اصلی را تشریح می نماییم.
برای ایجاد کلید اصلی در یک جدول بر روی فیلد موردنظر کلیک راست نموده و گزینه Set Primary Key را انتخاب می نماییم.

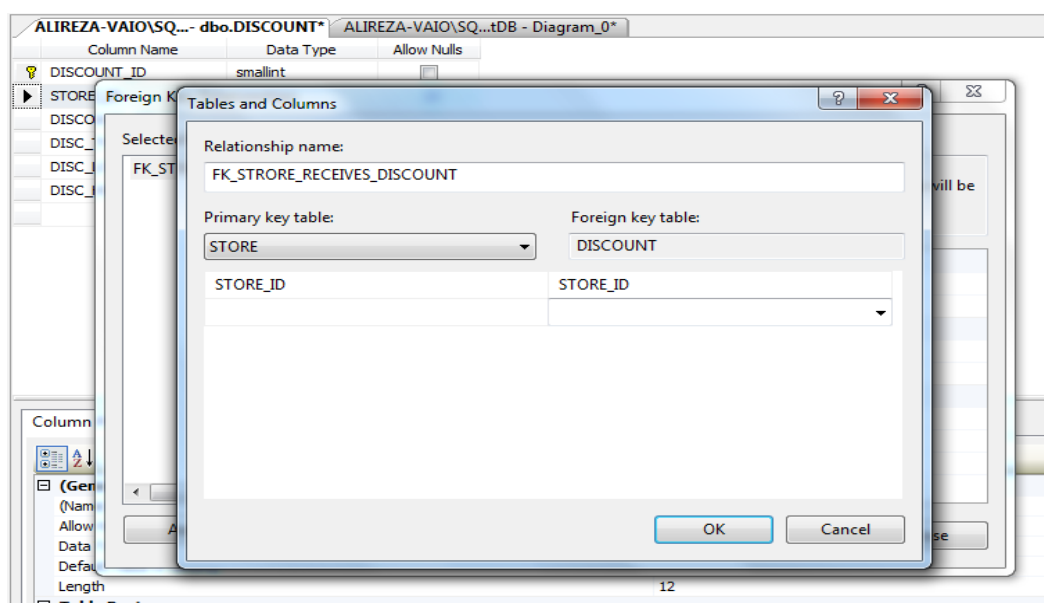


Foreign Key

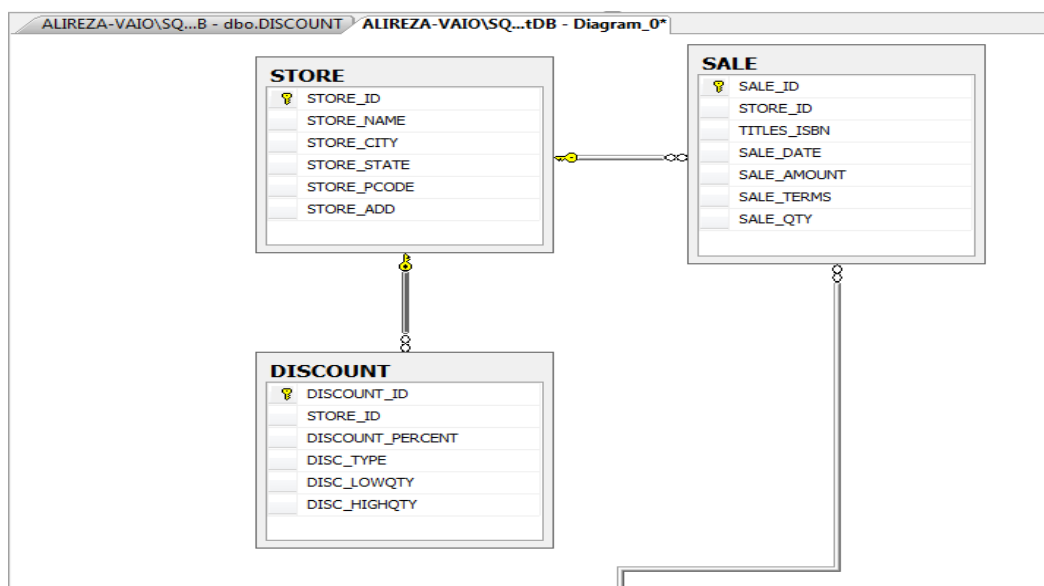
برای تعریف کلید خارجی در یک جدول بر روی یکی از فیلدهای آن جدول از نمای Design کلیک راست نموده و گزینه Relationships را انتخاب می‌نماییم.



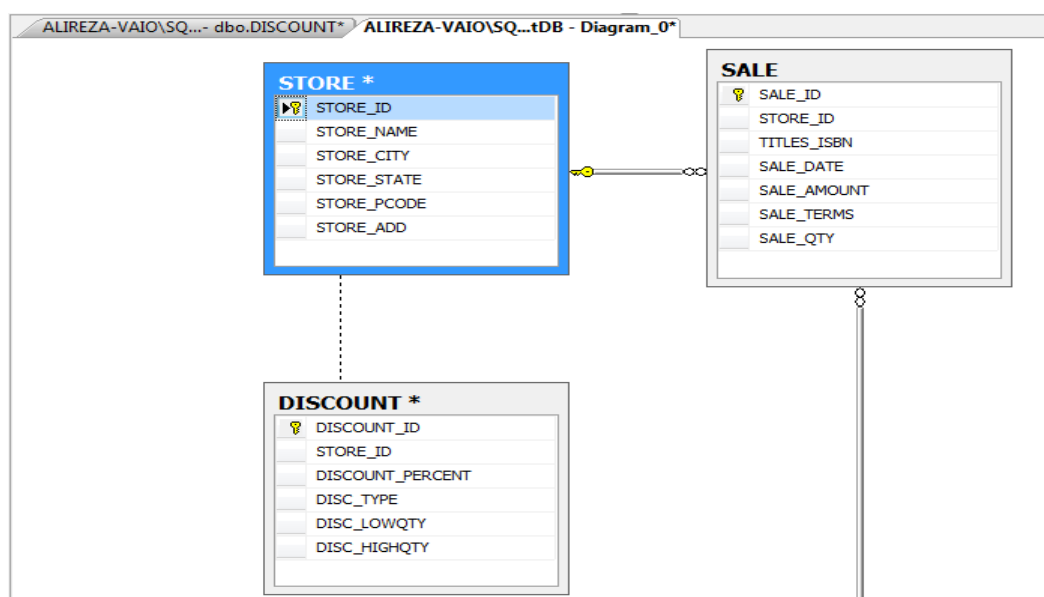
سپس در پنجره Foreign Key Relationships بر روی دکمه Add کلیک می‌نماییم. سپس یک نام مناسب برای رابطه کلید خارجی مان وارد می‌نماییم و در قسمت Tables and Columns Specification جداول و ستونهای کلید اصلی و کلید خارجی متناظرشان را انتخاب می‌نماییم.



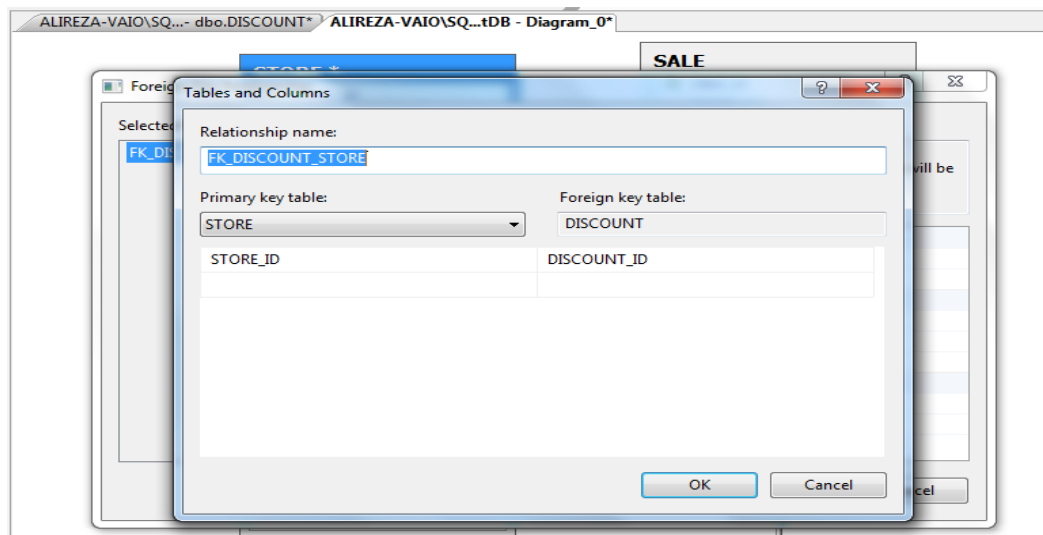
روش دیگر تعریف کلید خارجی استفاده از نمودار پایگاه داده است. به این منظور از فولدر Database Diagrams، نمودار پایگاه داده مورد نظرمان را انتخاب نموده و بر روی آن دابل کلیک می‌نماییم. یک نمودار پایگاه داده نوعی در شکل زیر نمایش داده شده است.



برای ایجاد کلید خارجی با استفاده از نمودار پایگاه داده ابتدا بر روی فیلد کلید اصلی جدول مبدا کلیک نموده و دکمه ماوس را نگاه می‌داریم و با حرکت دادن اشاره‌گر ماوس بر روی جدول مقصد کلیک ماوس را بر روی جدول مقصد رها می‌نماییم.



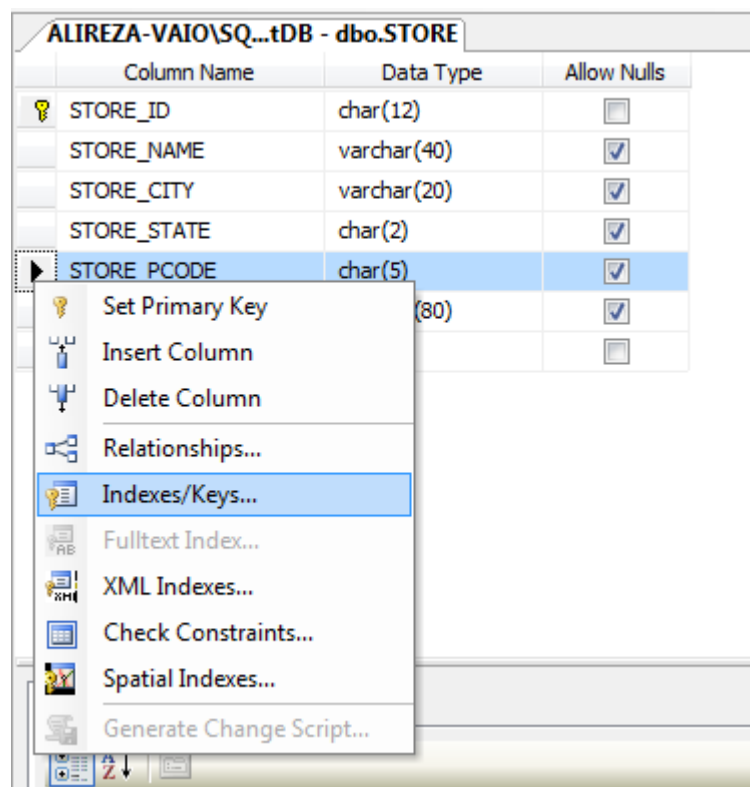
با رها کردن کلیک ماوس بر روی جدول مقصد پنجره Tables and Columns مطابق شکل زیر ظاهر می شود.



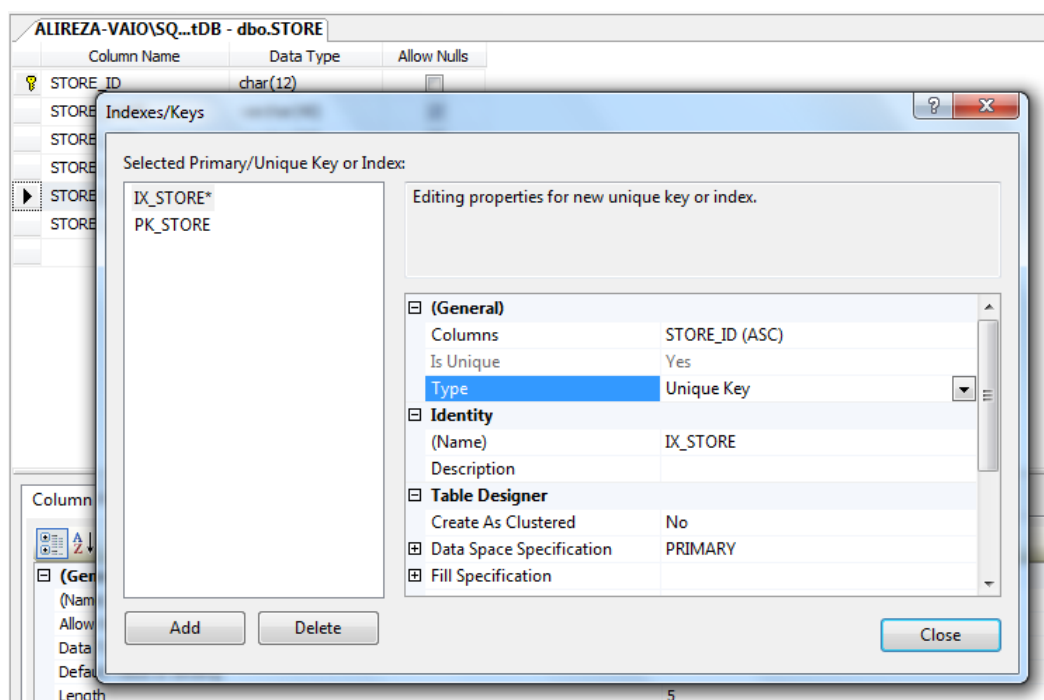
ابتدا در بخش Relationship name نامی مناسب برای رابطه کلید خارجی وارد می نماییم. سپس ستون کلید اصلی در جدول مبدا و ستون متناظر آن در جدول مقصد را انتخاب می نماییم و بر روی دکمه OK کلیک می نماییم. مجدداً بر روی دکمه OK کلیک می نماییم.

▪ Unique Key :

کلید یکتا کلیدی منحصر بفرد است که ممکن است مقدار Null نیز به خود بگیرد. از کلید یکتا زمانی استفاده می شود که ستونی در جدول داشته باشیم که کلید اصلی نباشد ولی بخواهیم مقادیر آن ستون منحصر بفرد باشد. همانطور که می دانید نمی توان در یک جدول بیش از یک کلید اصلی داشت به همین دلیل برای حصول اطمینان از منحصر بفرد بودن مقادیر ستونهای غیر کلید از کلید یکتا استفاده می شود. برای ایجاد یک کلید یکتا در جدول مورد نظر بر روی ستون مورد نظر کلیک راست نموده و گزینه Indexes/Keys را انتخاب می نماییم.



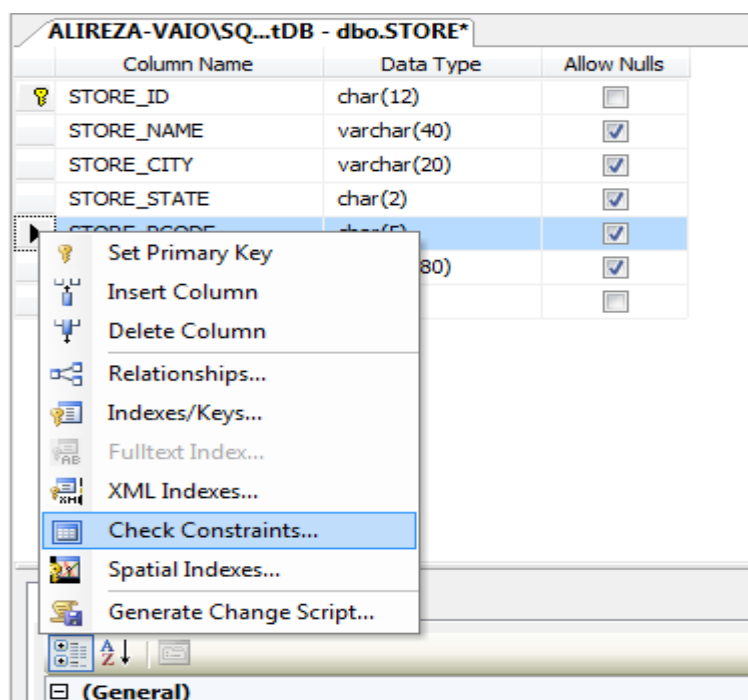
سپس در پنجره **Indexes/Keys** بر روی دکمه **Add** کلیک می‌نماییم. توجه نمایید که در پنجره **Indexes/Keys** تمامی کلیدهای تعریف شده برای آن جدول نمایش داده می‌شوند. بنابراین حتما ابتدا بر روی دکمه **Add** کلیک نمایید تا یک کلید جدید ایجاد شود و از تغییر هر یک از کلیدهای موجود خودداری نمایید.



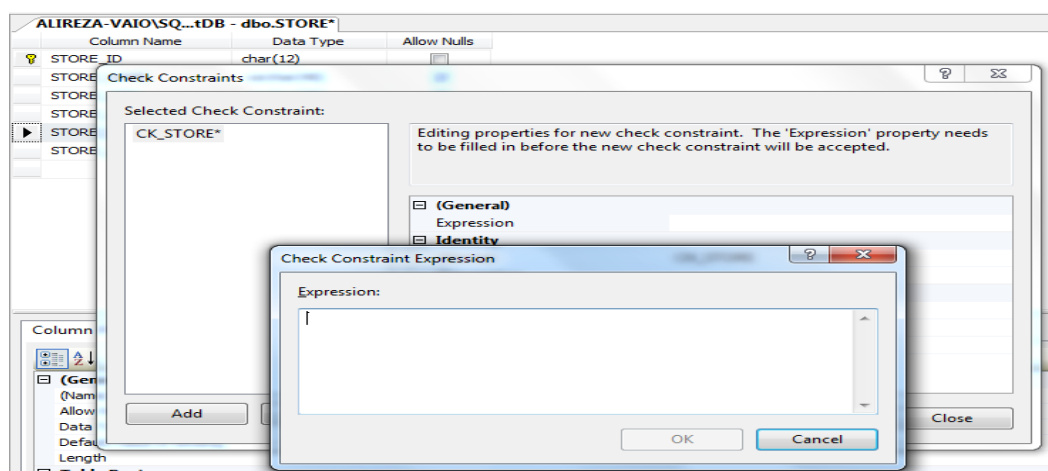
از بخش Columns نام ستون مورد نظر را انتخاب نمایید. در بخش Type گزینه Unique Key را انتخاب نمایید. در صورت تمایل در بخش (Name) نام پیش فرض را تغییر داده و نام دلخواه خود را وارد نمایید. در پایان بر روی دکمه OK کلیک نمایید.

▪ Check Constraints :

Constraint ها قوانینی هستند که بر روی مقادیر یک فیلد اعمال می‌شوند. برای اعمال یک قانون بر روی مقادیر یک فیلد ابتدا لازم است که تمامی مقادیر حال حاضر آن فیلد نیز از این قانون جدید پیروی نمایند به این معنی که اگر ردیف (یا ردیفهایی) وجود داشته باشند که قانون را نقض نمایند امکان به اجرا درآوردن قانون جدید تا زمان اصلاح مقادیر متناقض وجود نخواهد داشت. به منظور تعریف یک قانون جدید، در جدول موردنظر بر روی فیلد موردنظر کلیک راست نموده و گزینه Check Constraints را انتخاب می‌نماییم.



پنجره Check Constraints مطابق شکل زیر ظاهر می‌شود. برای افزودن یک قانون جدید بر روی دکمه Add کلیک می‌نماییم. با کلیک در بخش Expression پنجره Check Constrains. پنجره Check Constraint Expression ظاهر می‌شود. در این پنجره می‌توانید هر تعداد عبارات کنترلی که لازم دارید را وارد نمایید. سپس بر روی دکمه OK کلیک نمایید. در صورتی که عبارات وارد شده از نظر نحوی خطایی داشته باشند پیغامی مبنی بر وجود خطا دریافت خواهید نمود. در صورت عدم وجود خطا به پنجره Check Constraints بازخواهید گشت. بر روی دکمه OK کلیک نمایید تا از صفحه Check Constraints خارج شوید.



جلسه پنجم : آشنایی با View ها و ابزار Query Designer

۵-۱ هدف:

در این جلسه از آزمایشگاه پایگاه داده‌ها به بررسی View ها در پایگاه داده Microsoft SQL Server می‌پردازیم.

۵-۲ مقدمه:

همانطور که می‌دانید هر پایگاه داده از مجموعه‌ای از اشیاء تشکیل شده است، که این اشیاء برای ذخیره نمودن داده‌ها و یا ارجاع به داده‌ها مورد استفاده قرار می‌گیرند. مهمترین اشیاء در پایگاه داده عبارتند از Table ها، Constraint ها، View ها، Stored Procedure ها، Index ها و Trigger ها. بعنوان مثال Table ها بعنوان مهمترین شیء برای ذخیره سازی داده‌ها در پایگاه داده مورد استفاده قرار می‌گیرند. View ها نیز اشیایی هستند که برای ارجاع به داده‌ها مورد استفاده قرار می‌گیرند.

۵-۳ شرح:

در مورد تعریف View می‌توان گفت که یک Select در سطح پایگاه داده است. اجازه دهید قبل از ورود به مبحث View ها ابتدا کمی درباره مزایای استفاده از آنها صحبت کنیم. View ها اشیایی از پایگاه داده هستند که برای ارجاع به داده‌ها مورد استفاده قرار می‌گیرند. تفاوت بین View و یک Select عادی این است که در View کاربر امکان تغییر داده‌ها را ندارد و بدین سبب امنیت داده‌ها تامین می‌شود و از پیچیدگی‌های سیستمی کاسته می‌شود.

حال که کمی با مزایای View ها آشنا شدیم به شرح چگونگی ایجاد آنها می‌پردازیم. یکی از روشها با استفاده از نوشتن یک پرس و جو است. بدین منظور در پنجره نرم افزار Microsoft SQL Server Management Studio بر روی دکمه New Query کلیک می‌نماییم و یا از صفحه کلید از کلیدهای میانبر Ctrl+N استفاده می‌نماییم. پس از باز شدن سربری جدید با پیشوند SQLQuery[x].sql اقدام به تایپ عبارات زیر می‌نماییم.

```
Create View MyView as
Select empno, ename, job, sal
From emp
Where sal between 1500 and 5000
```

پرس و جوی بالا یک View جدید بنام MyView را ایجاد نموده و نتایج حاصل از پرس و جو را در آن ذخیره می‌نماید.

```

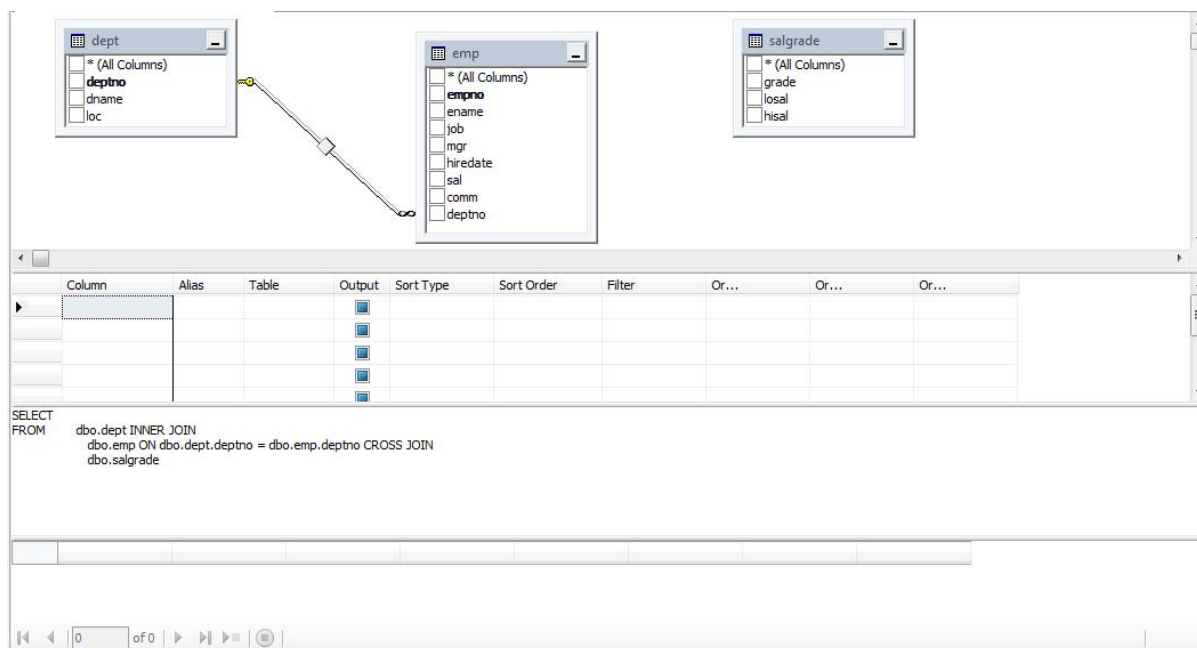
1 Create View MyView as
2 Select empno, ename, job, sal
3 From emp
4 Where sal between 1500 and 5000
    
```

	empno	ename	job	sal
1	7499	ALLEN	SALESMAN	1600
2	7566	JONES	MANAGER	2975
3	7698	BLAKE	MANAGER	2850
4	7782	CLARK	MANAGER	2450
5	7788	SCOTT	ANALYST	3000
6	7839	KING	PRESIDENT	5000
7	7844	TURNER	SALESMAN	1500
8	7902	FORD	ANALYST	3000

روش دوم ایجاد View ها استفاده از واسط گرافیکی ای است که در نرم افزار Microsoft SQL Server موجود است. بدین منظور بر روی فولدر Views در زیر فولدرهای پایگاه داده مورد نظر کلیک راست نموده و بر روی گزینه New View... کلیک می نماییم.

پنجره Add Table ظاهر می‌شود. در این پنجره نام تمامی جداول موجود در پایگاه داده نشان داده می‌شود. بر روی هر یک از جداول موردنظر کلیک نموده و دکمه Add را کلیک می‌نماییم. در انتها بر روی دکمه Close کلیک می‌نماییم.

پس از بستن پنجره Add Table می‌بایست کار طراحی پرس و جو را در ابزار گرافیکی Query Designer که شکل آن در زیر نشان داده شده است ادامه دهید.



ابزار Query Designer از چهار بخش تشکیل شده است :

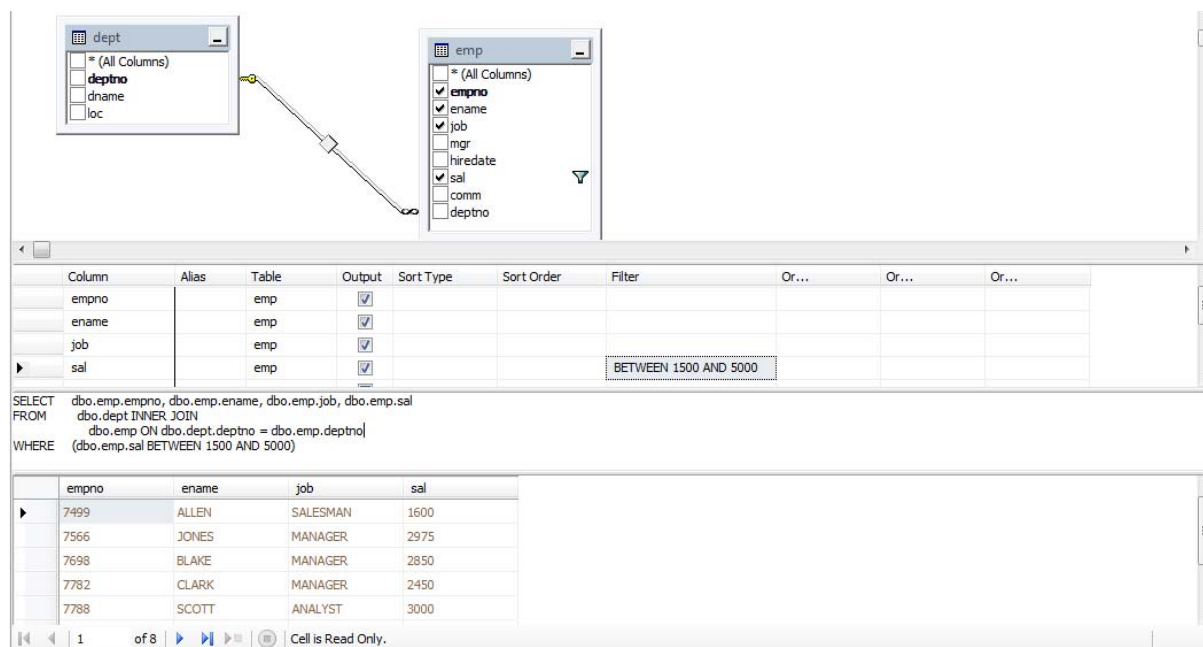
بخش اول Diagram Pane است که جداول و ارتباط آنها با یکدیگر را نشان می‌دهد.

بخش دوم Criteria Pane است که معیارهای پرس و جوی موردنظرمان را در آن وارد می‌نماییم (معیارهایی از قبیل شرایطی که برای نتایج پرس و جو در نظر داریم)

بخش سوم SQL Pane است که کد SQL متناظر با شرایط تعیین شده در Criteria Pane و ارتباط جداول در Diagram Pane را نشان می‌دهد.

بخش چهارم Results Pane است که نتایج حاصل پرس و جو پس از اجرا در این بخش نمایش داده می‌شوند.

حال که با بخشهای مختلف ابزار گرافیکی Query Designer آشنا شدیم می‌خواهیم پرس و جوی در ابتدای این جلسه از آزمایشگاه پایگاه‌داده را که با نوشتن عبارات SQL اجرا نمودیم با استفاده از ابزار Query Designer اجرا نماییم.



مطابق با شکل بالا در بخش Diagram Pane جدول emp را به Query Designer اضافه می‌نماییم. در مورد این پرس و جو نیازی به افزودن جدول dept نداشتیم ولی به جهت نمایش رابطه آن با جدول emp آن را شامل نمودیم. در ضمن فیلدهای sal, job, ename, empno را از جدول emp انتخاب می‌نماییم.

در بخش Criteria Pane در ستون Filter در روبروی فیلد job عبارت Between 1500 and 5000 را تایپ می‌نماییم.

با تایپ عبارت Between 1500 and 5000 در فیلد Filter از ردیف job مشاهده می‌نمایید که این عبارت در قالب یک عبارت شرطی Where برای فیلد job در SQL Pane افزوده می‌شود.

پس از مشخص نمودن معیارهای پرس و جو در بخش Criteria Pane از منوی Query Designer گزینه Execute SQL را انتخاب می‌نماییم و یا از صفحه کلید، کلیدهای میانبر Ctrl + R را فشار می‌دهیم.

نتیجه پرس و جو در بخش Results Pane ظاهر می‌شود.