

تكون قواعد المعطيات فرعاً أساسياً من فروع المعلوماتية، وتعتبر الأدوات التي تقدمها هامة لتطوير أنواع عديدة من التطبيقات المعلوماتية وخاصة في التطبيقات الإدارية والمالية والنظم المساعدة في اتخاذ القرار.

● قاعدة المعطيات (Database -DB):

مجموعة متكاملة من المعطيات المتعلقة ببعضها البعض (البيانات سواء أكانت data أو metadata) .

وبالطبع هذه البيانات سوف تنظم في مجموعة من الملفات وينظم التعامل مع هذه الملفات ببرنامج نسميه نظام إدارة قواعد المعطيات DBMS

● قواعد المعطيات العلائقية (Relational Databases):

هي أحد أنواع قواعد المعطيات التي تعتمد على النموذج العلائقي (وهناك نماذج أخرى مثل النموذج الشبكي، النموذج الهرمي،...)، وعلائقي يعني أن البيانات منظمة في جداول بأعمدة وصفوف وبالتالي ستكون قاعدة البيانات مكونة من مجموعة من الجداول، وتعتمد على مفاهيم الجبر العلائقي (Relational Algebra) والذي يتمثل ببساطة بعمليات التقاطع، الاجتماع، الفرق، المجموعات،... الخ.

● نظم إدارة قواعد المعطيات (Database Management Systems -DBMS):

من الأخطاء الشائعة، إطلاق اسم قواعد المعطيات على أنظمة إدارة قواعد المعطيات العلائقية. فأنظمة Oracle، MS SQL server، MS Access هي أنظمة إدارة قواعد المعطيات، والهدف منها توفير بيئة ملائمة تمكننا من تخزين المعلومات ضمن قاعدة المعطيات واسترجاعها والعديد من المهام، ومن أهم مهامها:

١. تعريف بني تخزين المعلومات.

٢. إيجاد التقنيات الملائمة للتعامل مع المعلومات المخزنة.

٣. تقديم نظم أمن وحماية للمعلومات المخزنة من الوصول غير المشروع.

٤. تجنب التضارب في المعلومات نتيجة تشارك عدة مستثمرين في الوصول إلى المعلومات.

يتم التواصل مع أنظمة قواعد المعطيات باستخدام لغة الاستعلامات البنوية SQL (Structure Query Language) .

● توصيف قاعدة المعطيات:

يوجد ثلاث مستويات لتوصيف قاعدة المعطيات، ويقوم نظام إدارة قواعد المعطيات بالترابط بين هذه المستويات.

١. مستوى المخطط المفاهيمي: يظهر عناصر المعلومات التي ستقوم قاعدة المعطيات بإدارتها ويعبر عنه بعدة طرق منها.

Entity Relationship Diagrams – ERD

٢. المستوى المنطقي: يتم فيه توصيف قاعدة المعطيات بلغة قياسية تمكن من التعامل مع المعطيات مثل لغة

Data Definition Language - DDL

٣. المستوى الفيزيائي (الداخلي): توصف الطريقة الفعلية لتخزين المعطيات.

● لغات المعطيات:

يوفر كل نظام إدارة معطيات على الأقل لغة واحدة تتيح لمستخدميه تعريف بنية قاعدة المعطيات، وصلاحيات الوصول إلى المعطيات، وغيرها، وتعتبر لغة الاستفسارات البنوية SQL لغة قياسية للتعامل مع قواعد المعطيات العلائقية، وهي لغة غير إجرائية، وإنما لغة تصريحية، وتقسم إلى:

١. لغة استعلام المعطيات (Data Query Language -DQL)

تستخدم للاستفسار عن البيانات واستعادتها من قاعدة المعطيات.

٢. لغة تعريف المعطيات (Data Definition Language – DDL)

تستخدم لتعريف كائنات قواعد المعطيات

٣. لغة التعامل مع المعطيات (Data Manipulation Language –DML)

تستخدم للوصول و التعامل مع المعطيات مثل تعديلها أو إضافتها أو حذفها.

٤. لغة التحكم بالمعطيات (Data Control Language – DCL)

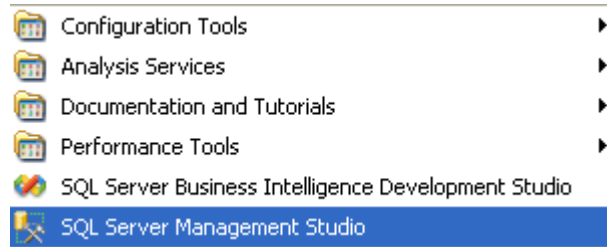
تستخدم من أجل منح أو سحب الصلاحيات.

ملاحظة:

قاموس المعطيات (Data Dictionary) يعرف بأنه ملف يحوي meta data (المعطيات السامية) وهي معلومات عن المعطيات تستدعى قبل قراءة أو تعديل المعطيات في نظم قواعد المعطيات.

• SQL Server :

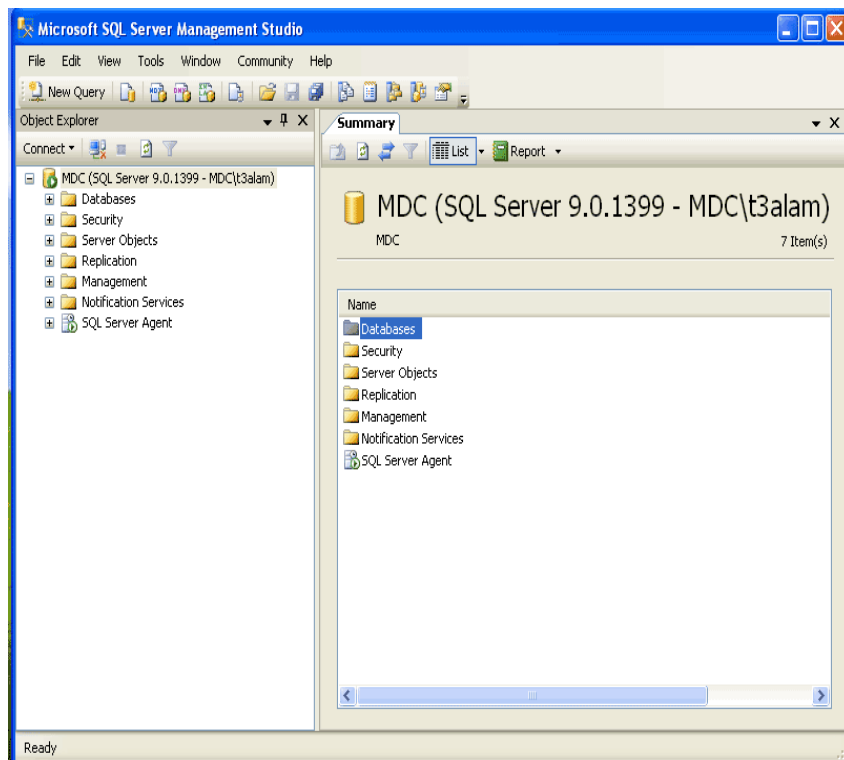
بعد تنزيل SQL Server على جهازك اذهب إلى start ثم programs ثم Microsoft SQL Server ثم SQL Server Management Studio كما في الشكل.



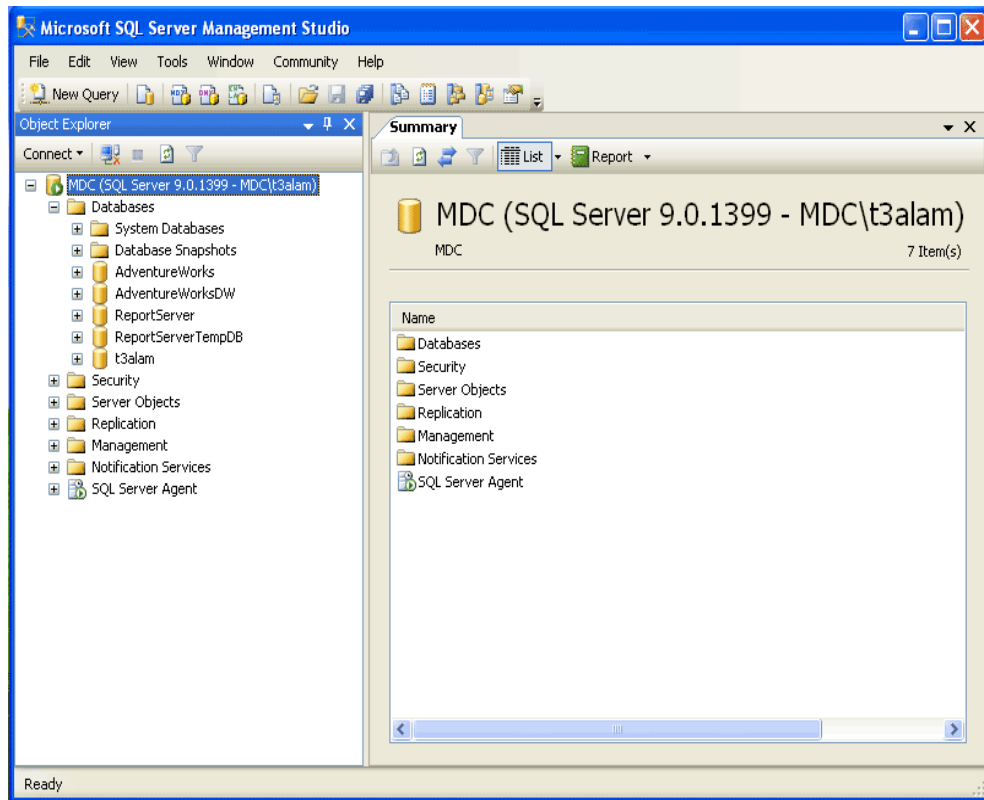
بعد ذلك قم بالاتصال بالserver، وإن كنت على نفس الجهاز اضغط connect .

ملاحظة: window authentication تعني العمل على الجهاز الحالي

SQL server authentication فيجب تحديد login و password وذلك لوجود أكثر من مستخدم .



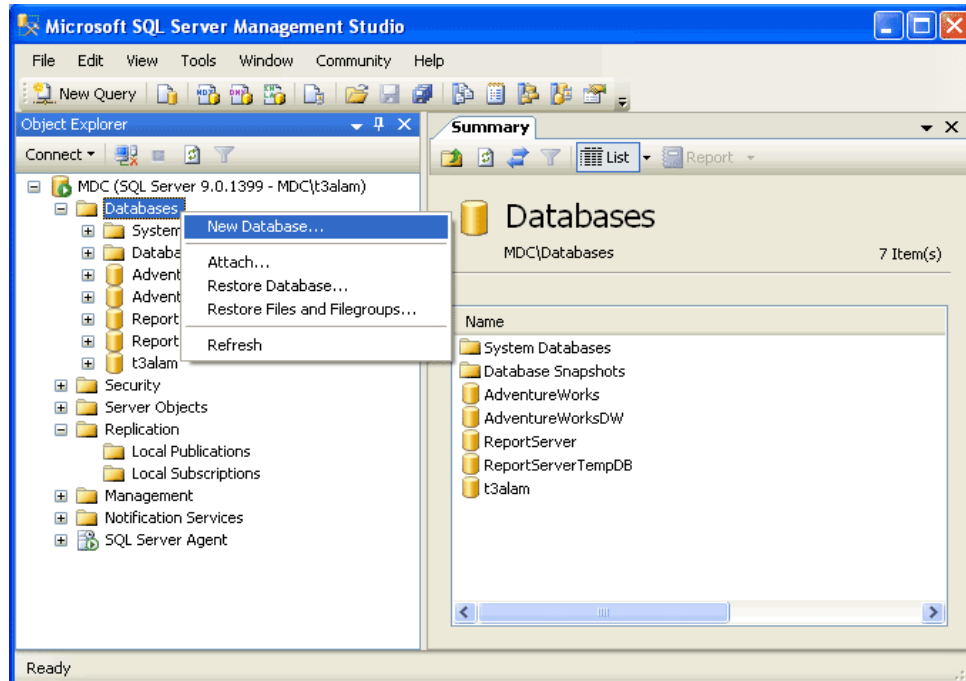
وبالنقر على علامة الزائد بجوار Databases سترى مجلد باسم System Databases يحتوي على قواعد بيانات النظام والتي هي:



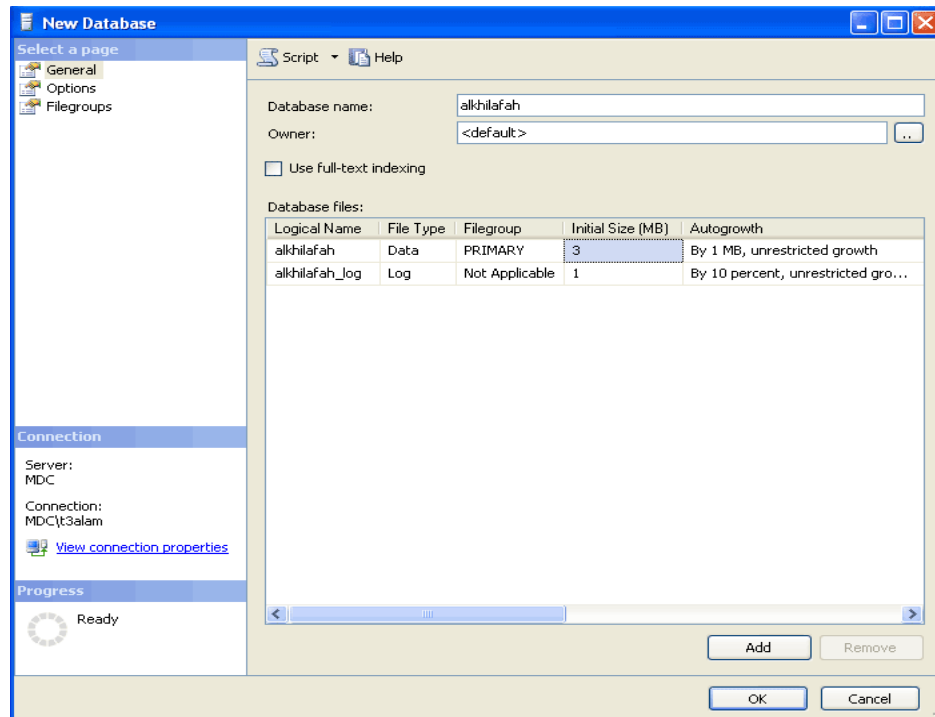
١. قاعدة البيانات **master**: تعتبر أهم قاعدة بيانات تحتوى على كل المعلومات الخاصة بالمستخدمين وكلمات سرهم وصلاحياتهم في التعامل مع قواعد البيانات ، أماكن وجود ملفات الـ **Data** ، إعدادات قواعد البيانات ، رسائل الأخطاء .. .
٢. قاعدة البيانات **tempdb** : قاعدة بيانات تنشأ عند تشغيل الـ **server** وتوضع فيها البيانات المؤقتة التي يحتاج الـ **server** لتخزينها مؤقتاً أثناء عمل الإجراءات المخزنة مثلاً .. وعند إغلاق الـ **Sql server** وإعادة فتحه تضيع هذه البيانات وتصبح قاعدة البيانات **tempdb** فارغة وجاهزة لاستقبال بيانات مؤقتة جديدة.
٣. قاعدة البيانات **model** : هي عبارة عن موديل يمكنك التعديل فيه حيث أن كل قاعدة بيانات جديدة تنشئها تأخذ كل خصائصها المبدئية من هذه القاعدة ، إذ يمكنك تعديلها لتناسب احتياجاتك في قواعد البيانات الجديدة من حيث الحجم مثلاً أو إنشاء جدول افتراضي بها ليتم إنشائه تلقائياً داخل أي قاعدة بيانات جديدة تقوم بإنشائها.
٤. قاعدة البيانات **msdb** : وهي المسؤولة عن كل الأوامر والوظائف التي سيقوم بها الإنسان الآلي المساعد في الـ **SQL Server** وهو **SQL Server Agent** أي تنفيذ كل الوظائف التلقائية التي تريد تنفيذها بالتعاون مع الـ **Windows** على قاعدة البيانات أو على أي شيء متعلق بالـ **Windows** بناء على حدث معين يحدث في قاعدة البيانات أو وقت معين .

● إنشاء قاعدة معطيات:

بالنقرة بواسطة الزر اليميني للماوس على **Databases** واختيار **New Database** كما في الشكل التالي.



نتيجة ذلك يظهر مربع حوار جديد لإنشاء قاعدة بيانات جديدة كما في الشكل التالي.



نستطيع من خلاله تحديد اسم قاعدة المعطيات ومالك هذه القاعدة (وهو المستخدم الذي لديه كل صلاحيات التعامل مع قاعدة المعطيات) ولو تركها <default> سيكون مستخدم الـ **Sql server** الأساسي هو مالك هذه القاعدة .

- بالتأشير على المربع **use full-text indexing** وهو المسؤول عن تنشيط عملية البحث داخل قاعدة المعطيات عن الكلمات أو الجمل داخل النصوص.

- ونجد أن هناك ملفان يتم انشاؤهما من قبل السيرفر لقاعدة المعطيات

الأول (ملف **Data**) يملك الامتداد **MDF** وهو المسؤول عن تخزين المعطيات.

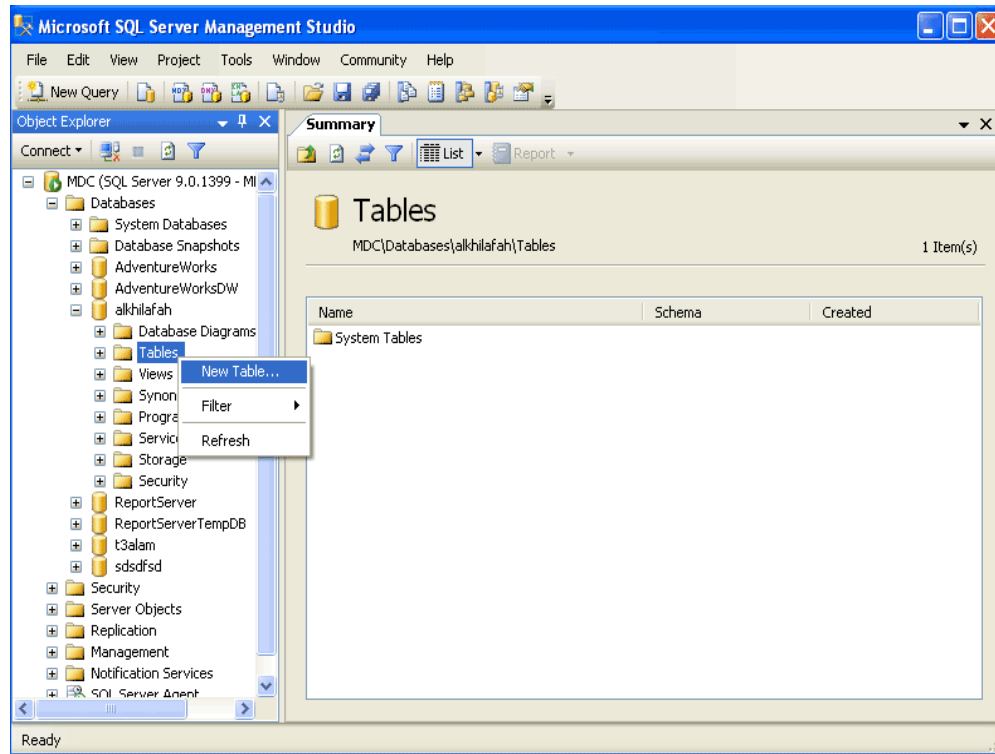
الثاني (ملف **Log**) يملك الامتداد **LDF** وهو المسؤول عن تخزين أي تعديلات على قاعدة المعطيات ، ويعتبر المصدر الأساسي لإرجاع البيانات المفقودة من ملف المعطيات.

ويمكنك إضافة ملفات أخرى عن طريق الزر **add** .

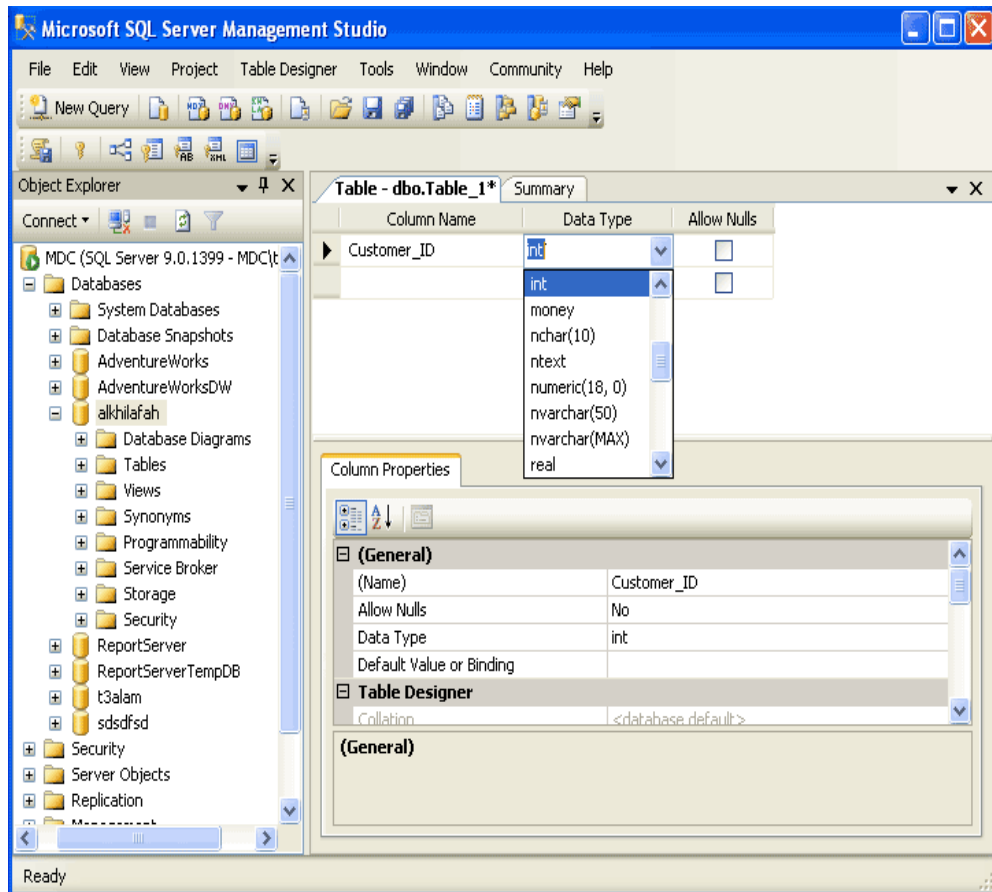
ويمكنك من خلال نافذة إنشاء قاعدة المعطيات تحديد الحجم الابتدائي للملف ومقدار الزيادة في حجمه كلما زادت المعطيات المخزنة فيه.

• إنشاء جداول قاعدة المعطيات:

فأسفل مجلد قاعدة المعطيات يوجد مجلد يسمى **Tables** وهم المجلد الذي يحوى جداول قاعدة المعطيات وعندها قام بالنقرة بواسطة الزر اليميني للماوس واختار من القائمة **New Table**.



فتظهر نافذة تصميم الجداول والتي نستطيع من خلالها تحديد أسماء حقول الجدول ونوعها وخواصها.



• أنواع المعطيات:

الأعداد:

name	Size	domain	Description
Int	4 B	-2³¹ to 2³¹-1	
bigint	8 B	-2⁶³ to 2⁶³-1	
smallint	2 B	to 2¹⁵-1 -2¹⁵	
tinyint	1 B	0-255	صغير جداً
Decimal أو numeric			الأرقام العشرية decimal(8,2) - decimal(8,2) ٨ عدد الخانات الكلي ومجاله من ١ إلى ٣٨ . والقيمة الافتراضية ١٨ ٢ عدد الخانات بعد الفاصلة ومجاله من الصفر إلى عدد الخانات الكلي و القيمة الافتراضية ٠

money	8 B		خانات عشرية ٤ (بعد الفاصلة)
smallmoney	4 B		خانات عشرية ٤ (بعد الفاصلة)
Bit	1 b		للقيم المنطقية True=1 False=0

الأعداد التقريبية:

name	Size	Domain
Float	8 B	- 1.79E+308 to -2.23E-308
real	4 B	-3.40E + 38 to -1.18E - 38

الوقت و التاريخ:

name	Domain
smalldatetime	January 1, 1900, through June 6, 2079
datetime	January 1, 1753, through December 31, 9999

السلاسل الحرفية:

name	Size	domain	Description
Char		1-8000 char	
varchar		1-8000 char	سلسلة محرفيه تحجز الحجم الخاص بالتخزين بشكل ديناميكي
Text	2 GB		

السلاسل الحرفية من النوع Unicode

name	Size	Domain
nchar		1-4000 char
nvarchar		1-4000 char
ntext	1 GB	

السلاسل الثنائية (Binary Strings):

name	domain	Description
binary	٨٠٠٠ بايت	

Varbinary		
image		Binary (img,video,Sound)

أنواع أخرى:

name	Description
xml	خاص بلغة التأشير الموسعة
Timestamp	مؤشر زمني لآخر التغيرات على السجل وبالتالي يزداد العداد عند كل عملية تعديل ويخزن كنوع binary
Uniqueidentifier	يخزن كـ binary ويقبل رموز الست عشري وهي قيمة فريدة

● ملاحظات:

١. كيفية إنشاء حقل ذو قيمة تزايدية Auto increment:

من خواص الحقل نختار الخاصية Identity specification ثم نختار Is Identity=yes ، ويمكن تحديد قيمة التزايد من خلال الخاصية Identity Increment

٢. لتحديد قيمة افتراضية لحقل :

نحدد القيمة الافتراضية للخاصية Default value or binding

٣. لنقل قاعدة البيانات يجب أخذ الخيار Tasks ومن ثم Detach ثم الانتقال إلى مكان الملفات (data)mdf و (log) ldf

٤. للاتصال مع قاعدة ما يجب أخذ الخيار Attach ، وفي حالة عدم مشاهدة المخطط نأخذ خصائص القاعدة ومن ثم نختار File Group ونحدد الـ Owner

● أنواع المفاتيح (Keys):

١. المفتاح الرئيسي (Primary Key):

a. Unique لا يسمح بقيم تكرارية أي يملك قيم فريدة لكل سجل.

b. لا يسمح بقيمة Null

c. يفيد في تحديد العلاقات بين الجداول.

d. لا يمكن أن نعرف إلا مفتاح أساسي واحد لكل جدول.

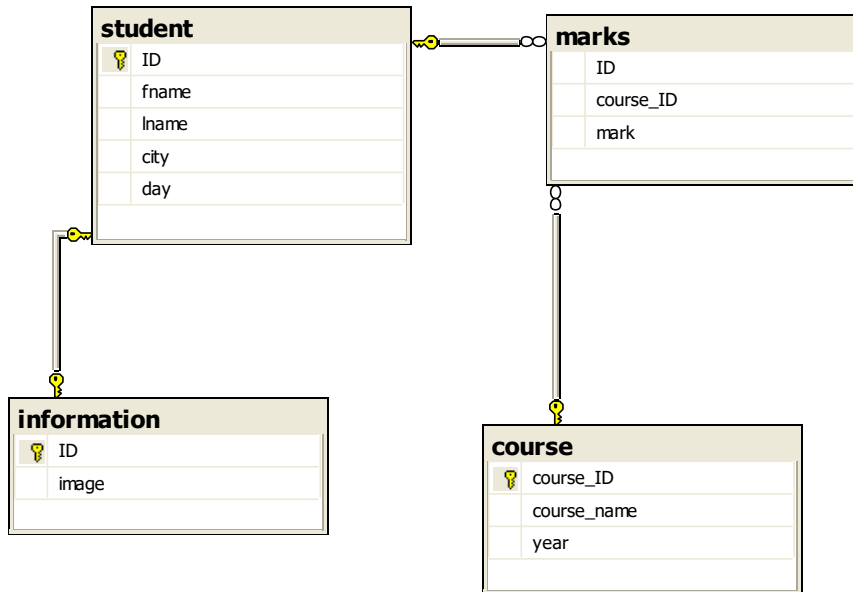
٢. المفتاح الثانوي (Foreign Key):

a. يسمح بالقيم التكرارية

b. يستخدم لإظهار العلاقات بين الجداول ويضمن سلامة وتكامل البيانات، وهو يمثل الحقل في جدول الابن والذي يربط مع حقل الجدول الأب وهو حقل ذو مفتاح رئيسي، وبالتالي أي سجل يتم إضافته في الجدول الابن يجب أن يقابل سجل في الجدول الأب أو أن يحوي قيمة **null** في العمود الموافق.

c. يمكن ان يحوي الجدول ٢٥٣-٠ مفتاح ثانوي

d. السلوك الافتراضي للمفتاح الثانوي أن يمنع حذف أي سجل من السجلات الأب عند وجود سجلات مرتبطة بهذا السجل في الجدول الابن؟ لكن احيانا نحتاج إلى هذا الحذف الاتوماتيكي، ويتم عن طريق **cascade on delete**

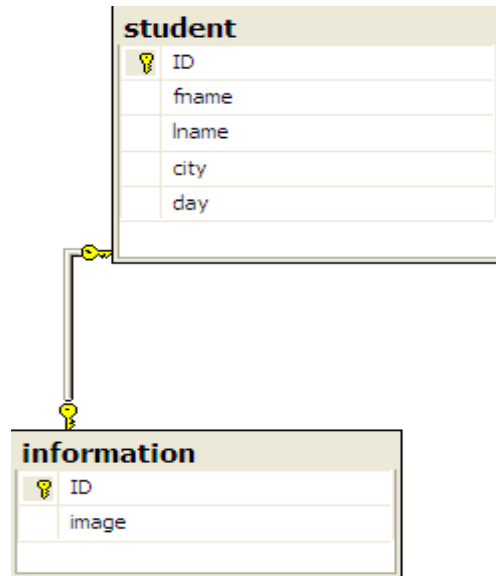


• العلاقات (relationships)

○ أنواع العلاقات:

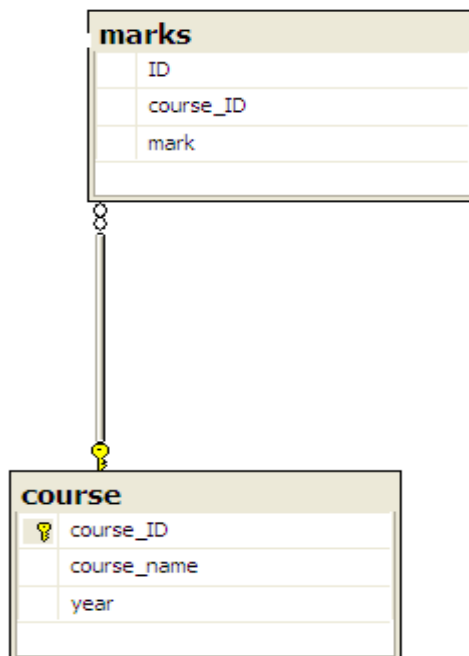
١. One to one :

مثال : جدول الطلاب وجدول يحوي صور عن الطلاب.



٢. One to many :

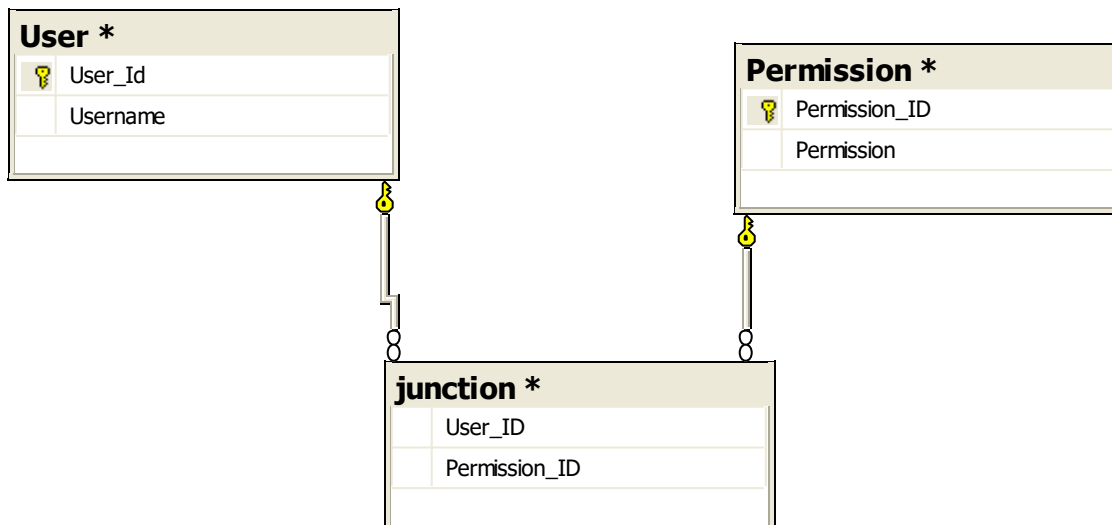
مثال : جدول الطلاب و جدول العلامات



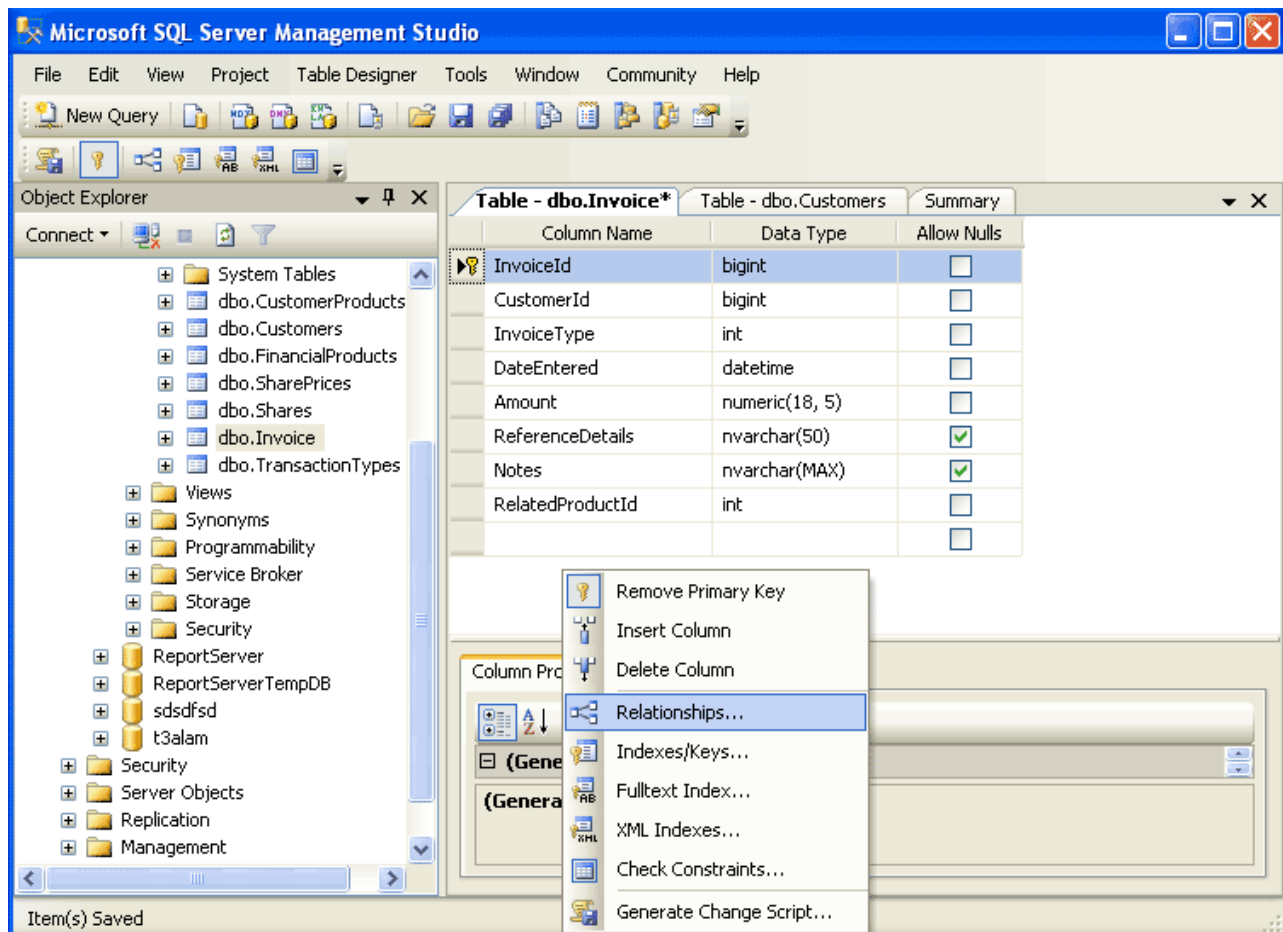
٣. Many to many :

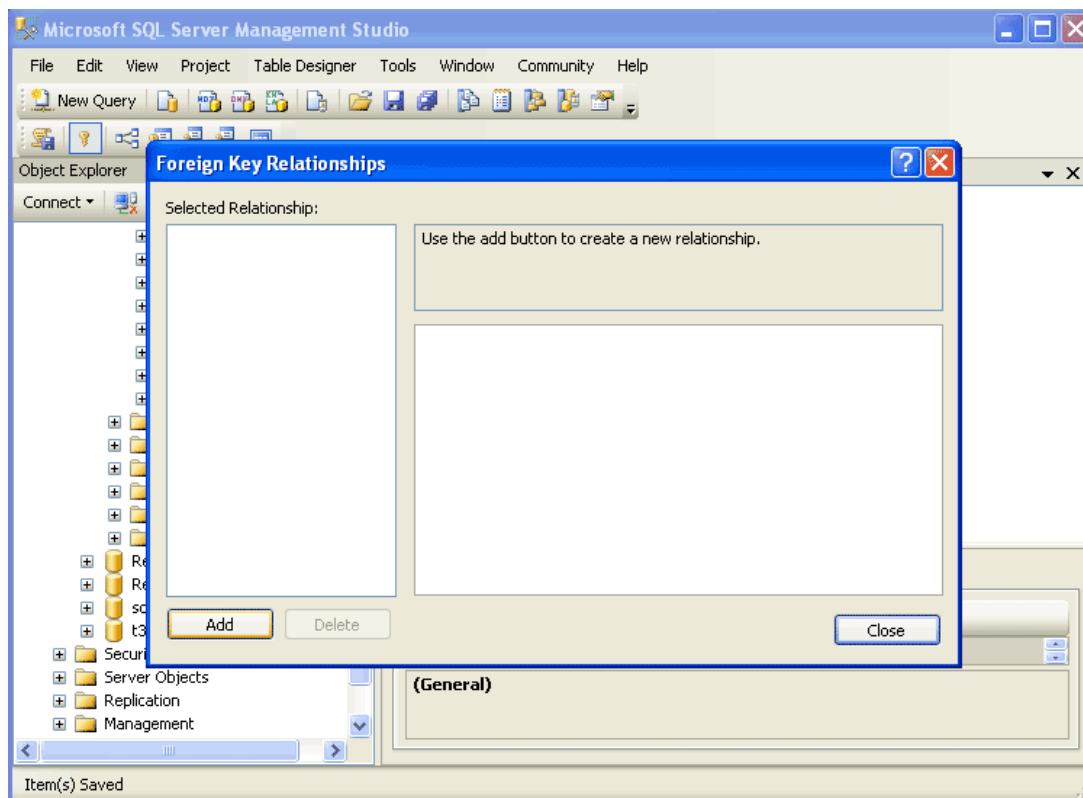
تنشأ العلاقة بين ثلاث جداول أي يوجد جدول وسيط (جدول الملتقى) يخزن صفة متعلقة بالعلاقة وليس بالجدولين.

مثال : جدول المستخدمين و جدول السماحيات و الجدول الوسيط يحوي رقم المستخدم ورقم السماحية .

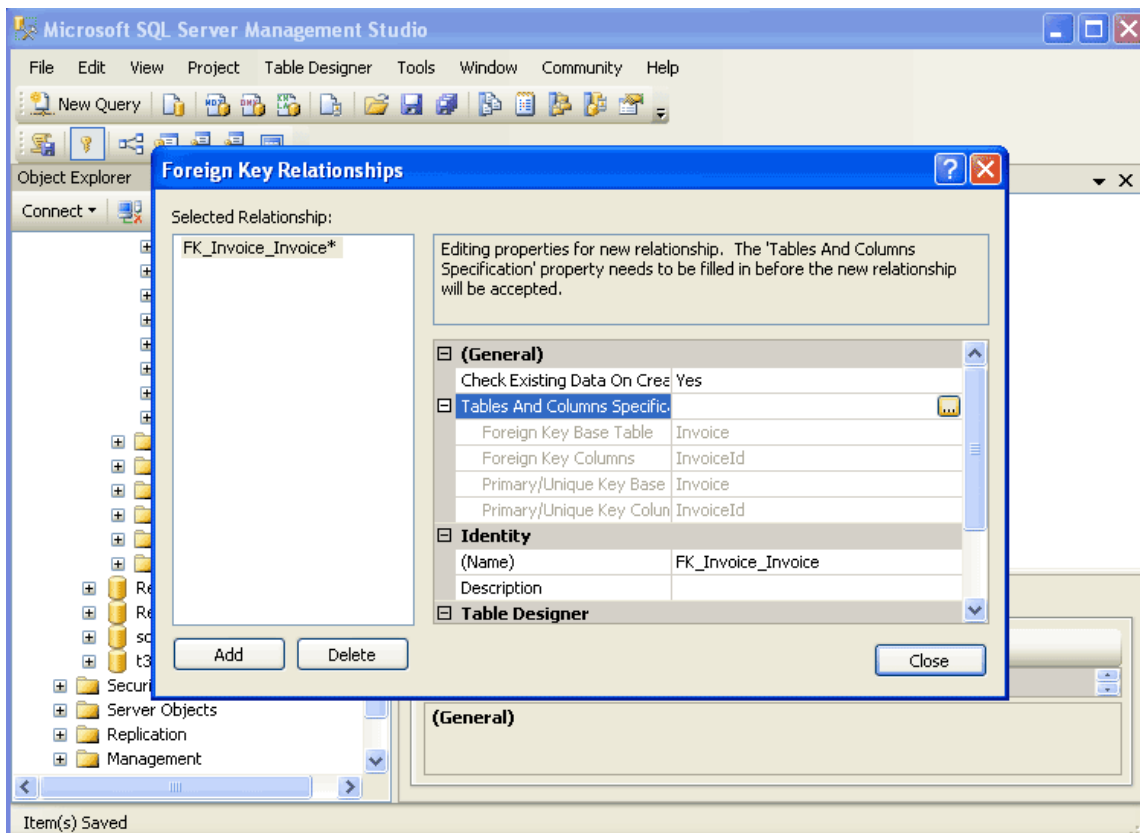


○ إنشاء العلاقات:

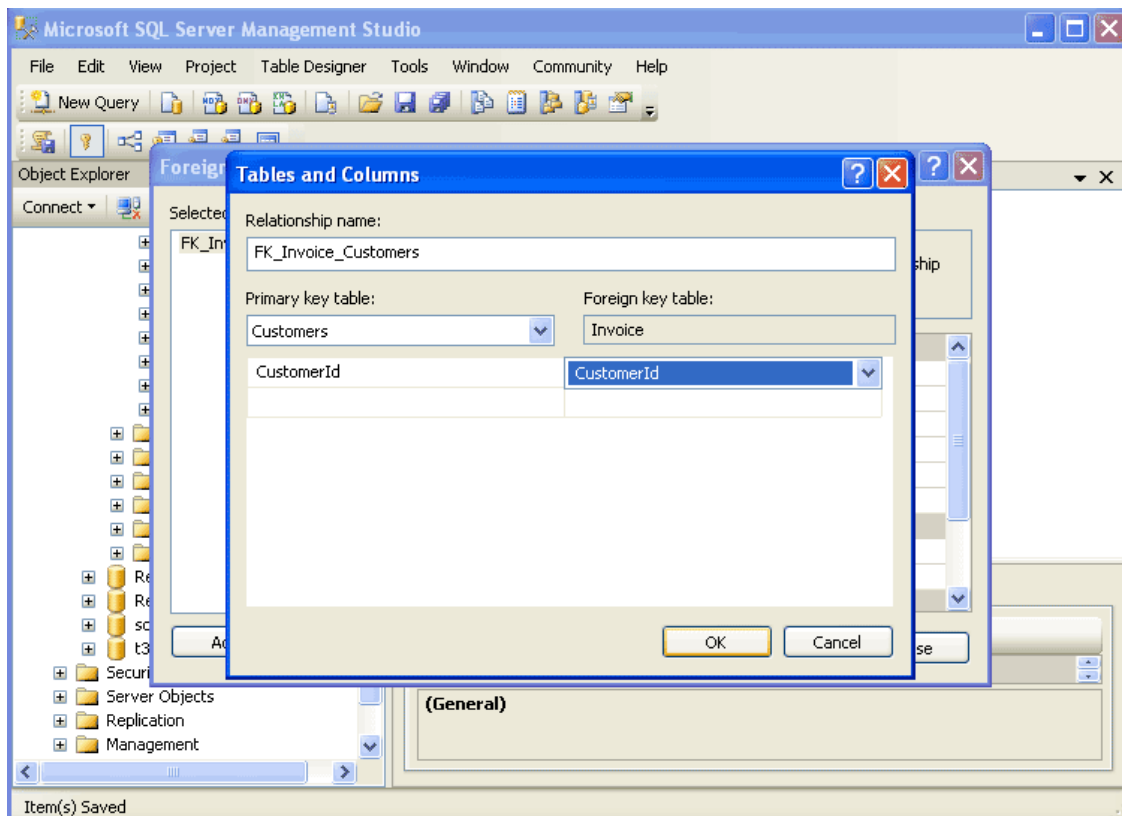
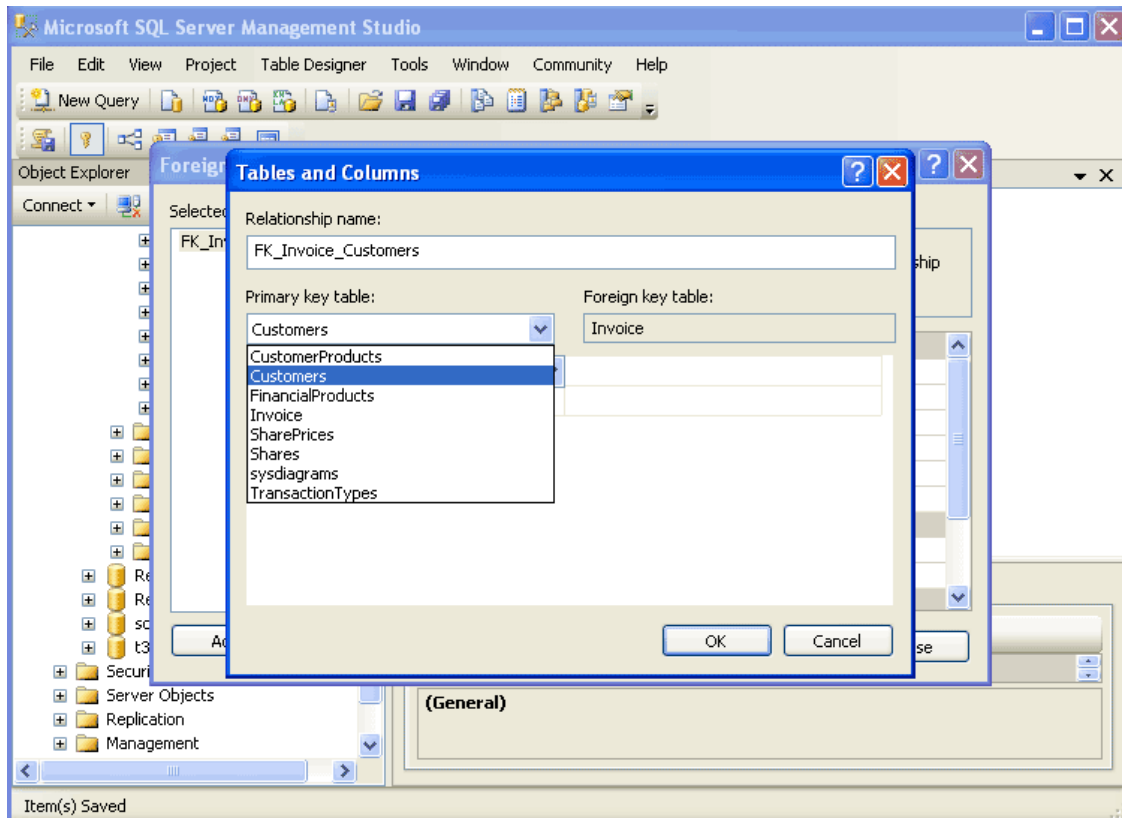


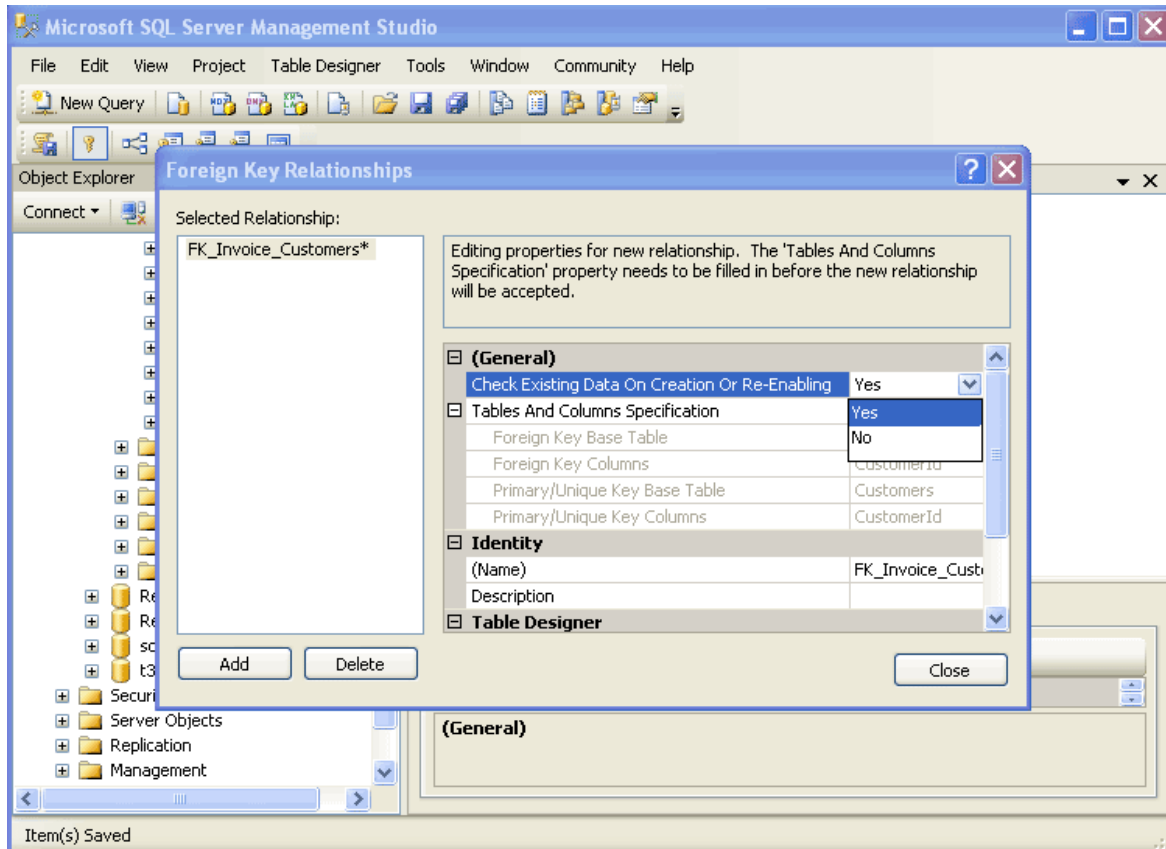


نقوم بضغط المفتاح add لإضافة علاقة جديدة

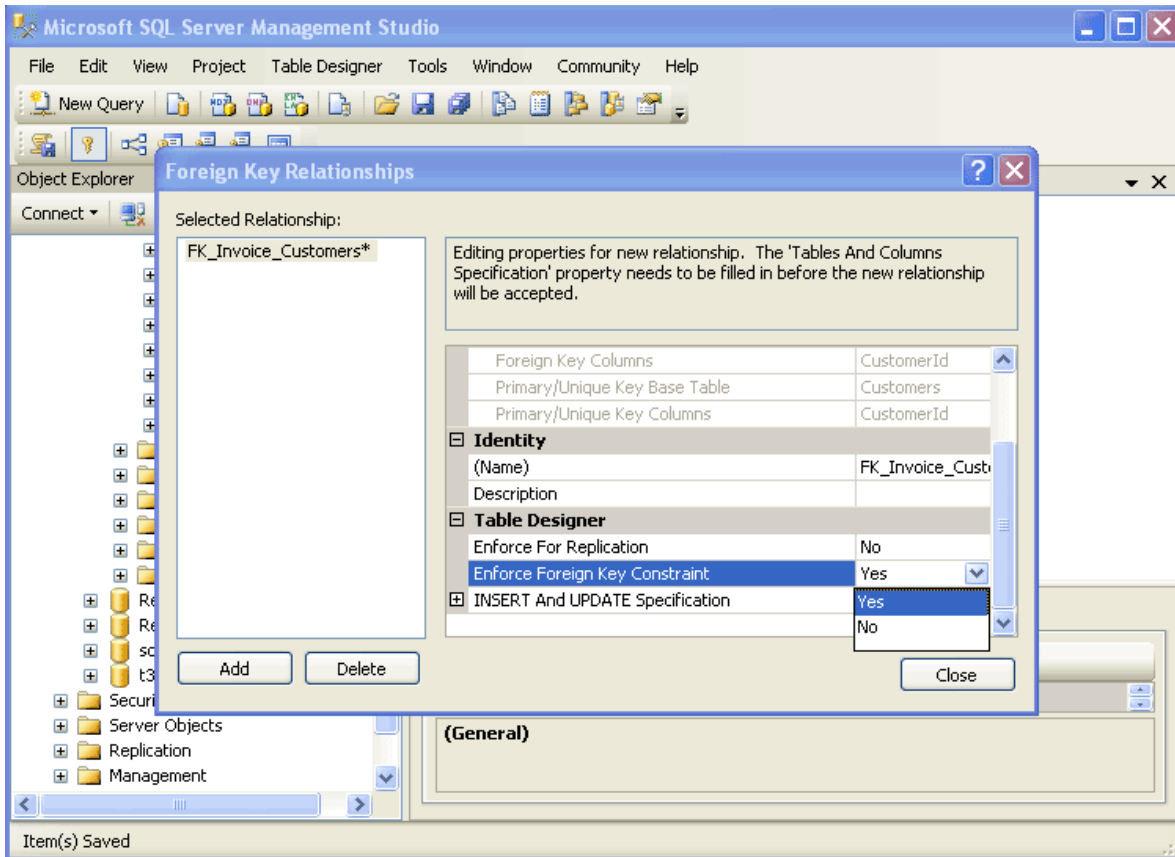


نقوم بفتح علامة الزائد بجوار tables and columns specification ثم نقوم بضغط الزر المنقط بجوارها ...

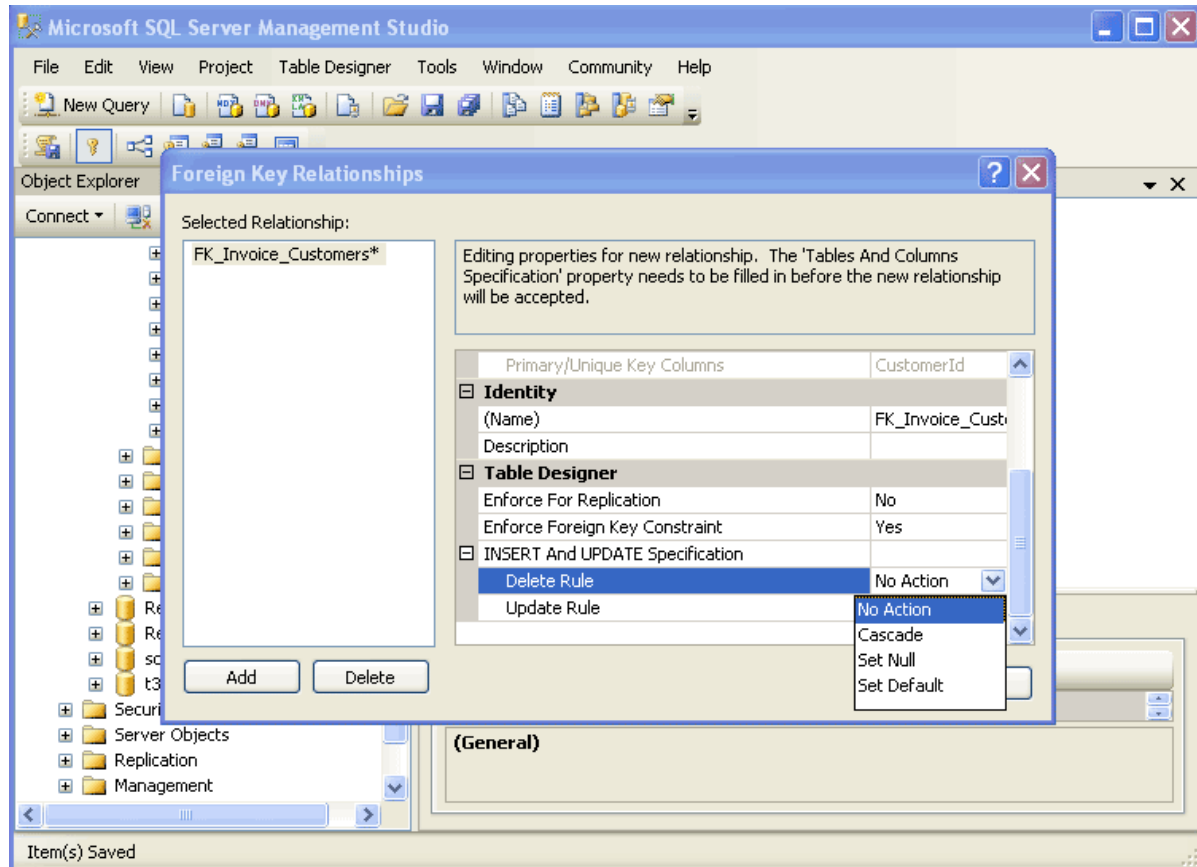




- في المجموعة General نجد أول خيار هو Check Existing Data On Creation Or Re-Enabling وهو المسؤول عن التأكد من أن بيانات الصف الذي سيستخدم في الربط متطابقة في الجدولين وذلك إن كنا نقوم بالربط وفي الصف بيانات وليس فارغا .. وعند اختيار Yes .. لابد أن تكون البيانات متطابقة وإلا فلن تتم عملية الربط بنجاح .
- أما الخيار Enforce Foreign Key Constraints فهو المسؤول عن جعل أي تغيير يحدث في أي صف من الصفين يتم مثله في الصف الآخر بالضبط كي يكون الصفين دائما متطابقين



- ومن الخيار الذي أسفل منه Insert And Update Specification نحدد طبيعة هذا التغيير بالنسبة إلى الرقم المسلسل وكل البيانات المتعلقة به.



- وهناك أربع خيارات في حالة الحذف أو التعديل .. هي بالترتيب

١- No Action ... أي لا شيء يحدث .

٢- Cascade أي شيء هنا يحدث مثله هناك .. أي لو حذفنا مستخدم هنا ستحذف كل الفواتير المتعلقة به " وهذا خيار خطير ولكن نحتاجه أحيانا "

٣- Set Null ... أي أنه عند حذف أو تعديل الرقم المسلسل مثلا في جدول الزبائن سيتم استبداله بالقيمة Null في جدول الفواتير وذلك للحفاظ على بيانات الفواتير وللدلالة على أن الزبون الخاص بها تم حذفه أو تعديله " لازم الصف يكون يقبل القيمة Null "

٤- Set Default ... نفس عمل الخيار Null .. ولكن هنا يحط القيمة الافتراضية إلي أنت كنت عاملها للصف عند إنشاء الجدول . " لازم تكون واضع قيمة افتراضية على الصف "

- أما عن الخيار **Enforce For Replication** وهي عبارة عن صنع صورة طبق الأصل من قاعدة البيانات في مكان آخر .. وعند تغيير البيانات في أحدهم يتم نفس التغيير في قاعدة البيانات الأخرى

• **ملاحظة:**

لتغيير خواص العلاقة بين جدولين ، نختار بالزر اليميني على الجدول Relationship .