الفصل الخامس عشر
العمل مع مشروعات البيانات Access Data Projects

نتعرف فى هذا الفصل على مفهوم مشروعات بيانات Access أو Access Data Projects والتى يمكنك من خلالها نقل تطبيقات Access من تطبيقات أحادية المستخدم إلى تطبيقات تعمل بنظام الخادم/العميل Client/Server. نتعرف فى هذا الفصل على مكونات مشروعات البيانات وقواعد بيانات SQL Server على أن نستكمل الحديث فى الفصل القادم إن شاء الله.

بانتهاء هذا الفصل ستتعرف على:

- التعرف على مشروعات البيانات (ADP)
- التعرف على خادم SQL ومشروعات البيانات
- إلقاء نظرة على مشروع بيانات
- العمل مع جداول SQL Server داخل مصمم المشروعات
- شرح مربع خصائص الجدول
- فهم عروض SQL Server

التعرف على مشروع البيانات (ADP)

بواسطة Access 2007 يمكنك إنشاء مشروعات بيانات Access أو Access Data Projects وتختصر هكذا ADP ويطلق عليها أيضاً Access Projects "مشروعات Access" أو Client/Server Applications "تطبيقات الخادم/العميل". فكل هذه المسميات مرادف لعملية واحدة تسمح لك فى حالة العمل داخل شبكة اتصالات بالاتصال من خلال Access بقاعدة بيانات SQL Server بأشكالها المختلفة سواءً كانت SQL Server 2005 SP2 أو SQL Server 2000 SP2 Express (SSX) أو غيرها من إصدارات SQL Server السابقة كمحرك البيانات MSDE 2000 الشهير والذى تم استخدامه بكثرة فى الإصدارات السابقة من Access.

MSDE اختصار لعبارة Microsoft Desktop Engine وهو المحرك الذى كان مستخدماً ابتداءً من Access 2000 مع الإصدارات السابقة قبل نظام Windows Vista. وهو لا يعمل تحت نظام التشغيل Windows Vista. إذا كانت لك خبرة سابقة باستخدام MSDE 2000 فإن SQL Server Express أو SSX يعتبر تحديثاً له ومشابهاً له فى الاستخدام.

خصائص مشروعات البيانات ADP

تحتوى مشروعات بيانات Access على العديد من السمات التى تميزها عن قواعد بيانات Access المعتادة والتى على رأسها ما يلى:

- كما فى تطبيقات Access المحولة (Upsized Access Applications) التى شرحناها فى الفصل السابق، تبنى مشروعات البيانات أساساً على جداول SQL Server، إلا أنها لا تستخدم ملفات accdb. فى تخزين النماذج والتقارير وبقية كائنات قاعدة البيانات وإنما يتم تخزينها جميعاً فى ملف واحد من النوع adp. (أو ade. فى حالة تشفير مشروع البيانات)، حيث e تعنى Encrypted.

- لا تحتوى مشروعات البيانات على الاستعلامات (تذكر أن تطبيقات Access المحولة تحتوى على استعلامات) حيث يقوم SQL Server بتخزين جملة SELECT فى ملفات تسمى Views بعد تحويلها إلى لغة الآلة بدلاً من استخدام استعلامات التحديد QueryDef المستخدمة فى استعلامات Access العادية.
- تستخدم مشروعات البيانات الإجراءات المخزنة Stored Procedures بدلاً من الاستعلامات الإجرائية الموجودة داخل محرك Jet، والإجراءات المخزنة مثل ملفات Views فى أنها مترجمة مسبقاً ولكن الفرق أنها غير مقيَّدة باستعلامات التحديد وإنما تشمل الاستعلامات الإجرائية أيضاً، وتظهر كفاءتها فى عمليات الإضافة والحذف والتحديث .
- تحتوى مشروعات البيانات على نوافذ مصمم المشروعات Project Designers بدلاً من نوافذ تصميم الجداول والاستعلامات داخل Access، وعلى الرغم من اختلاف تخطيط مصمم المشروعات عن نوافذ تصميم جداول واستعلامات Access نوعاً ما، إلا أن الوظيفة متشابهة إلى حدٍ كبير فى الحالتين. أما صفحات البيانات (DataSheets) المستخدمة مع الجداول والاستعلامات فهى واحدة سواءً بقواعد بيانات Access أو بمشاريع البيانات.
- تحتوى مشروعات البيانات على الدوال المعرفة من قِبَل المستخدم User-Defined Functions (UDFs) المستخدمة فى إرجاع البيانات من الجداول إلى العروض والإجراءات المخزَّنة.
- يستخدم SQL Server الخصائص الموسعة Extended Properties فى الحصول على ما يكافئ حقول البحث Lookup Fields وصفحات البيانات الفرعية Subdatasheets وبذلك لا تفقد هذه المزايا عند التحويل إلى نموذج الخادم/العميل. كما يمكنك أيضاً من خلال هذه الخصائص تعيين أقنعة الإدخال Input Masks والعناوين Captions وتنسيقات عرض البيانات. كما يحتوى SQL Server على الإجراءات المخزنة المستخدمة فى قراءة وإضافة وحذف

الخصائص الممتدة المخصصة من قاعدة البيانات.

- يحتوى SQL Server على سمات لغة XML التى لا تعتمد على مثيلاتها داخل Access 2007.
- يتم تصميم النماذج والتقارير داخل مشروعات البيانات بنفس طريقة تصميمها داخل قواعد بيانات Access المعتادة كما يمكنك إحضار جميع كائنات Access إلى مشروعات البيانات عدا الجداول والاستعلامات

متى يتم استخدام مشروعات البيانات ADP

فى الواقع أنه يمكنك تطوير تطبيقات قواعد البيانات الكبيرة باستخدام بيئة Access العادية دون الحاجة لاستخدام مشروعات البيانات .ولكن هناك العديد من الحالات التى يصبح معها استخدام مشروعات البيانات أمراً ملحاً والتى يمكن إيجازها كما يلى :

- عند الاتصال مع قواعد بيانات SQL Server حيث تحتوى Access 2007 على بيئة تصميم رسومية للتعامل مع قواعد بيانات SQL Server أو إنشاء بيانات جديدة.
- التطبيقات التى تنوى تطويرها لتعمل تحت SQL Server سواءً فى المستقبل القريب أو على المدى البعيد. حيث يسهل تحويلها باستخدام معالج التحويل Upsizing Wizard.
- المشروعات الجديدة التى تحتاج إلى مساحات تخزين كبيرة أو التى تحتاج إلى تشفير جداول معينة أو حقول معينة بقاعدة البيانات. وكذلك الحال فى حالة زيادة عدد مستخدمى قاعدة البيانات عن ٢٥ مستخدم.
- فى حالة التطبيقات التى تتطلب وضع قيود على تعديل بعض الجداول والاستعلامات لبعض المستخدمين أو مجموعات المستخدمين ، حيث لا يدعم Access 2007 هذا النوع من تأمين البيانات.

- المشروعات التى تستخدم النسخة المماثلة Replication من SQL Server إلى SQL Server آخر، أكثر من النسخة المماثلة من Access إلى SQL Server. حيث لا يدعم Access 2007 استخدام النسخ المماثلة Replication.

ولكى يتمكن مستخدموك من استخدام مشروع البيانات، يجب تثبيت Access 2000 أو أيٍ من إصدارات Access التالية إلا إذا قمت باستخدام نسخة تشغيل Access 2007 (Access 2007 Runtime Version) لإنشاء نسخة قابلة للتوزيع من مشروع البيانات. ويقوم Access 2007 افتراضياً بحفظ مشروع البيانات بتنسيق Access 2002/2003 إلا أنك تستطيع حفظ المشروع بتنسيق Access 2000 إن أردت، حيث لا تدعم مشروعات البيانات تنسيق Access 2007. كما يجب أيضاً تثبيت SQL Server Express (SSX) لدى المستخدمين فى حالة استخدام مشروع البيانات لقاعدة بيانات محلية.

التعرف على خادم SQL (SQL Server) ومشروعات البيانات (ADP)

الاتجاه الحديث عند تطوير نظم إدارة قواعد البيانات هو استخدام قواعد بيانات الخادم/العميل الخلفية Client/Server back end حينما تكون المرونة والتأمين مطلوبة بدرجة كبيرة بقاعدة البيانات.

إصدارات SQL Server

يأتى SQL Server 2005 فى سبعة إصدارات أو طبعات مختلفة وهى الإصدار Compact والإصدار Express والإصدار Evaluation والإصدار Developer والإصدار Workgroup والإصدار Standard والإصدار Enterprise، حيث يتم عادةً تشغيل الإصدارين الأخيرين Standard وEnterprise بنظام التشغيل Windows 2003 Server أو أيٍ من إصداراته التالية، بينما يعتبر الإصدار Compact (يختصر هكذا SSCE) قاعدة بيانات علائقية التى ربما تحل يوماً ما محل

جداول Access داخل مشروعات البيانات. أما الإصدار Express أو SSX فلا يوجد أى قيود على استخدامه وهو ما يعنى إمكانية توزيعه مع أى تطبيق Access تقوم بإنشائه بسهولة تامة. وتستخدم جميع هذه الإصدارات كود T-SQL الذى يتوافق فى جزء كبير مع كود ANSI-92 SQL.

مزايا وعيوب مشروعات البيانات

أهم مزايا الانتقال من قواعد بيانات الخادم/العميل التى تعتمد على فصل البيانات عن واجهة المستخدم إلى مشروعات البيانات، هى استخدام تقنية OLE DB وتقنية ADO الجديدة بدلاً من مثيلاتها السابقة ODBC وDAO، وذلك لأن ADO متوافقة بشكل كبير مع لغات البرمجة مثل VBScript وJavaScript المستخدمة فى تطبيقات الويب وهذا على عكس DAO، كما أن جميع كائنات التطبيق تشترك فى اتصال OLE DB واحد فقط لخادم SQL مما يعمل على استخدام موارد الخادم بشكلٍ فائق. وجدير بالذكر أن SQL Server لا يدعم أنواع البيانات الجديدة داخل Access 2007، مثل المرفقات Attachments وحقول المذكرة القابلة للإضافة فقط، وحقول البحث متعددة القيمة Multivalued lookup fields (MVLF) أو نوع البيانات التقليدى OLE Object.

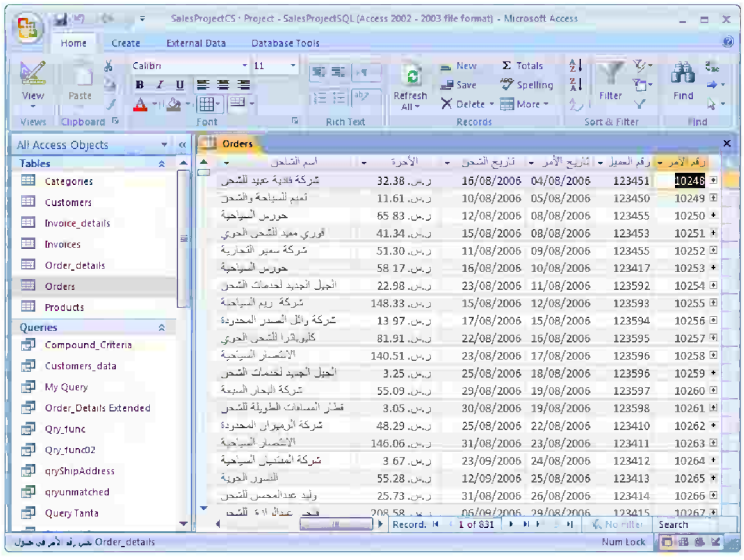
وحقيقةً، فلن يظهر الفارق بين كلٍ من OLE DB وADO من جهة وODBC وDAO من جهةٍ أخرى إلا مع الجداول التى يتعدى عدد سجلاتها ١٠٠ ألف سجل (صف)، إلا أن الاتصال المباشر للمشروعات بخادم SQL يمنحك ميزة الاستفادة من العروض والإجراءات المخزَّنة المعرفة مسبقاً داخل SQL Server والتى تعمل على تحسين استجابة الخادم وخاصةً فى حالة احتواء قاعدة البيانات على جداول بها عدد كبير من السجلات.

ولكن على عكس تطبيقات Access العادية ذات التنسيق accdb. والتى تستطيع الاتصال بأى قاعدة بيانات علائقية RDBMS، تستطيع مشروعات البيانات الاتصال بخادم SQL فقط.

إلقاء نظرة خاطفة على مشروع بياناتك

يحتوى القرص المدمج المرفق على مشروع البيانات SalesProjectCS.adp الذى يتطلب الاتصال بقاعدة البيانات SalesProjectSQL على خادم SQL Server، وهذا بالطبع يتطلب تثبيت SQL Server Express (SSX) على حاسبك وربط المشروع بملف قاعدة البيانات SalesProjectSQL.mdf وملف الولوج المصاحب SalesProjectSQL.ldf.

انتقل إلى المجلد الخاص بهذا الفصل على القرص المدمج المصاحب للكتاب ثم انقر ملف المشروع SalesProjectCS نقراً مزدوجاً لفتحه داخل نافذة Access. تظهر مجموعة الجداول Tables بلوحة التنقل بالجانب الأيسر من النافذة، انقر الجدول Orders نقراً مزدوجاً لفتحه داخل النافذة (انظر شكل ١-١٥).



شكل ١-١٥ المشروع داخل نافذة Access.

وبالنظر إلى الشكل السابق، نلاحظ اختلاف لوحة التنقل داخل مشروعات البيانات قليلاً عن لوحة التنقل الموجودة داخل تطبيقات Access المعتادة التى تحتوى على روابط للجداول، حيث تكمن أهم أوجه الاختلاف فيما يلى:

- تظهر الجداول المخزنة بقواعد بيانات SQL Server داخل المجموعة Tables بلوحة التنقل بالرمز كما لو كانت موجودة داخل قاعدة بيانات محلية، أى بدون أسهم الروابط التى تدل على أن الجدول موجود بقاعدة بيانات أخرى غير الحالية. فإذا قمت بفتح أحد الجداول، كما فى الجدول Orders بالشكل السابق، تحصل على نفس النتيجة التى نحصل عليها داخل تطبيقات Access المعتادة.
- تحتوى المجموعة Queries داخل لوحة التنقل على العروض Views (مثل العرض Customers_data) والإجراءات المخزَّنة Stored Procedures بالإضافة إلى الدوال Functions (مثل الدالة Orders_cust).
- تستخدم العروض داخل المجموعة Queries الرمز وهو نفس رمز استعلامات التحديد Select Queries داخل Access وذلك للتشابه الكبير بين العروض واستعلامات التحديد التى ينتج عنها كائن من النوع QueryDef.
- تستخدم الدوال (Functions) داخل المجموعة Queries الرمز ، وتقوم بإرجاع قيم من الجداول تماماً كما فى حالة العروض إلا أنها تدعم استخدام المعاملات Parameters.

تتطلب العروض والدوال تعريف الحقول المختلفة صراحةً بدلاً من استخدام الرمز * للإشارة إلى جميع الحقول وذلك للاستفادة بالخصائص الممتدة الموجودة داخل SQL Server. فعلى سبيل المثال، إذا احتوى الاستعلام على الصيغة SELECT * FROM tableName، فلن تظهر صفحات البيانات الفرعية ولا حقول البحث بنتيجة العرض أو الدالة. كما لا تدعم مجموعات السجلات الناتجة من الإجراءات المخزَّنة هذه الخصائص من الأساس.

- تستخدم الإجراءات المخزنة (Stored Procedures) داخل المجموعة Queries الرمز ، وتقوم بتنفيذ الاستعلامات الإجرائية Action Queries

والاستعلامات التى تحتوى على المعاملات، وتتميز بكفاءة التنفيذ العالية مقارنةً بعبارات SQL المركبة المماثلة.

- تقوم مخططات قاعدة البيانات Database Diagrams بنفس وظيفة العلاقات Relationships داخل Access وتحتوى على نفس الرمز إلا أنها تختلف قليلاً فى مظهرها عن مثيلاتها داخل Access.

أما بقية الكائنات كالنماذج والتقارير على سبيل المثال فلا يوجد اختلاف بينها وبين مثيلاتها داخل Access، ما عدا وجود بعض الاختلافات البسيطة فى خصائص النماذج والتقارير، فعلى سبيل المثال، يمكنك تخصيص الخاصية Record Source لنموذج أو تقرير بأحد العروض أو إحدى الدوال أو الإجراءات المخزَّنة.

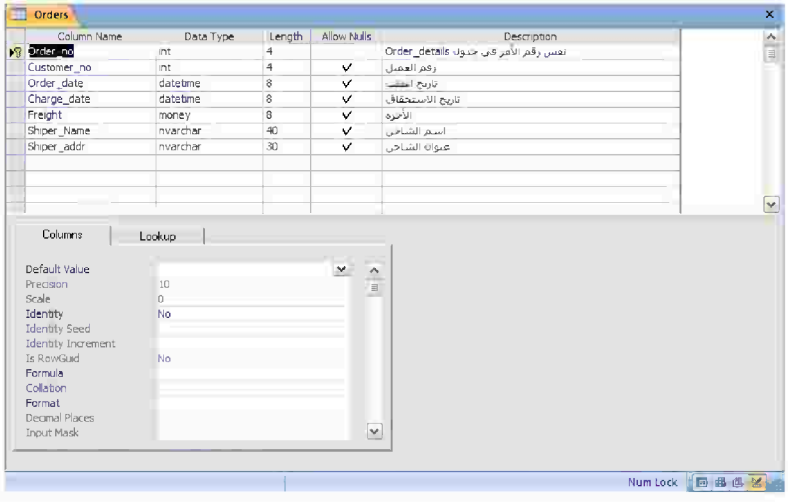
العمل مع جداول SQL Server داخل مصمم المشروعات

تحتوى مشروعات البيانات على نافذة تصميم تسمى مصمم المشروعات Project Designer، حيث يمكنك من خلال هذه النافذة تغيير تركيب الجداول والعلاقات والعروض والدوال والإجراءات المخزنة. ويقوم مصمم المشروعات بتنفيذ عبارات SQL التى تعكس التغييرات التى قمت بها من خلاله كلما قمت بإجراء عملية الحفظ. وتعتبر القدرة على تعديل تصميم جداول SQL Server من السمات الهامة بمشروعات البيانات، حيث لا يمكنك أداء ذلك مع الجداول المرتبطة بقواعد البيانات العادية. فعلى الرغم من عدم وجود الجدول بتطبيق Access، إلا أنك تستطيع تعديل تصميمه.

وتظهر الجداول داخل مشروع البيانات بطريقة مشابهة لظهورها داخل تطبيقات Access المعتادة (انظر شكل ٢-١٥)، حيث تظهر قيم التاريخ بالتنسيق المختصر بينما تظهر الحقول النقدية بالعملة المعرفة بالإعدادات الإقليمية بنظام التشغيل.

شكل ٢-١٥ الجدول Orders بمشروع البيانات.

وكما رأينا منذ قليل، فعند فتح جداول SQL Server داخل مشروعات البيانات، فإنها تظهر فى واجهة عرض صفحة بيانات Access المعتادة (راجع شكل ٢-١٥). أما طريقة عرض تصميم الجدول داخل مصمم المشروعات، فتختلف نوعاً ما (انظر شكل ٣-١٥).



شكل ٣-١٥ جدول مشروع البيانات فى طريقة عرض التصميم داخل مصمم المشروعات.

يلاحظ من شكل ٣-١٥ التشابه الكبير بين بيئة تصميم جداول SQL Server وجداول Access المعتادة، حيث يوجد قسمان، قسم لأسماء الحقول ونوع بياناتها والآخر لخصائصها. ففى القسم العلوى من الشاشة يتم تعريف الحقل من خلال أربعة أعمدة أساسية هى "اسم العمود" Column Name ونوع بياناته Data Type وطوله Length وهل يشتمل على بيانات أم لا بالإضافة إلى عمود خامس للملاحظات. أما القسم السفلى فيحتوى على تبويبين الأول بعنوان Columns "أعمدة" والثانى بعنوان Lockup "بحث"، حيث يحتوى التبويب الأول على خصائص الحقل المختار بالقسم العلوى، ويلاحظ تنشيط بعض الخصائص وتعطيل الخصائص الأخرى تبعاً لنوع بيانات الحقل.

يوضح الجدول التالى الأعمدة الموجودة بالجزء العلوى وأهم الخصائص الموجودة بالتبويب Columns بالجزء السفلى مع مقارنتها بالخصائص المماثلة داخل Access.

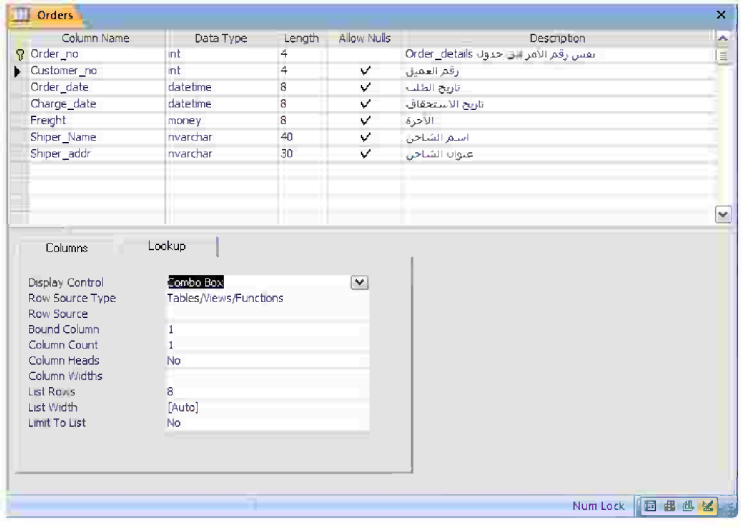
جدول أعمدة حقول الجدول وخصائص الحقول

الخاصية	الاستخدام
Column Name	تستخدم فى تعيين اسم الحقل وتقوم مقام عمود "اسم الحقل" Field Name الموجود فى Access، ويمكنك استخدام المسافات إلا أنها غير مستحبة.
Datatype	يعبر عن "نوع البيانات" Data Type و"حجم الحقل" Field Size الموجودين فى Access ولكن باستخدام مصطلحات SQL Server.
Length	نفس "حجم الحقل" Field Size الموجود فى الحقول النصية، ما عدا الأعمدة التى من النوع "حرف" Char فلها حجم ثابت.
Allow Nulls	السماح بعدم إدخال قيمة وهى عكس خاصية Required الموجودة فى Access. وللسماح لحقل

الخاصية	الاستخدام
	بطول صفرى، يتم تنشيط المربع الموجود أسفل العمود Allow Nulls والمقابل لهذا الحقل .
Description	نفس خاصية "الوصف" Description الموجودة داخل Access.
Default Value	نفس خاصية "القيمة الافتراضية" Default Value داخل Access.
Precision	تستخدم مع الحقول الرقمية والحقول العشرية، ويتم من خلالها تحديد عدد الأرقام الموجودة بالعمود (الحقل).
Scale	تستخدم مع الحقول الرقمية والحقول العشرية، ويتم من خلالها تحديد عدد الأرقام الموجودة إلى اليمين من العلامة العشرية.
Identity	تتشابه هذه الخاصية مع نوع البيانات AutoNumber داخل Access. اختر Yes لزيادة قيمة الحقل بمقدار معين مع كل سجل يتم إضافته إلى الجدول.
Identity Seed	يتم من خلالها تحديد القيمة الابتدائية للحقل فى حالة تنشيط الخاصية Identity.
Identity Increment	يتم من خلالها تعيين مقدار الزيادة فى قيمة الحقل بين كل سجل والسجل التالى فى حالة تنشيط الخاصية Identity.
Is RowGuid	عند اختيار Yes بهذه الخاصية فهذا يعنى احتواء الصف على معرف فريد عام يسمى GUID يتم استخدامه فى عمليات استخراج النسخ المطابقة Replication

الخاصية	الاستخدام
Formula	تحتوى على تعبير مستخدم فى إنشاء قيمة لحقل محسوب
Collation	تعيين ترتيب عملية الفرز فى حالة الأعمدة الحرفية
Format	يمكنك من خلال هذه الخاصية اختيار تنسيق معرف مسبقاً تماماً كما فى الخاصية المصاحبة Format داخل Access
Decimal Places	تعيين عدد الأرقام العشرية تماماً كما فى الخاصية Decimal Places داخل Access
Input Mask	يمكنك من خلال هذه الخاصية كتابة سلسلة من الحروف لاستخدامها كقناع إدخال لبيانات الحقل مثل >LLLLL للحصول على حروف كبيرة Capital، كما يمكنك استخدام معالج قناع الإدخال فى تعيين سلسلة البيانات.

أما التبويب Lookup الموجود بالجزء السفلى من مصمم المشروعات بجوار التبويب Columns فيمكنك من خلاله تعيين قيم الخصائص الإضافية التى تتشابه تماماً مع مثيلاتها داخل Access، حيث يمكنك تخصيص الخاصية Row Source بعبارة T-SQL أو جدول SQL Server أو عرض أو دالة أو حتى قيمة أو قائمة من الحقول (انظر شكل ١٥-٤).

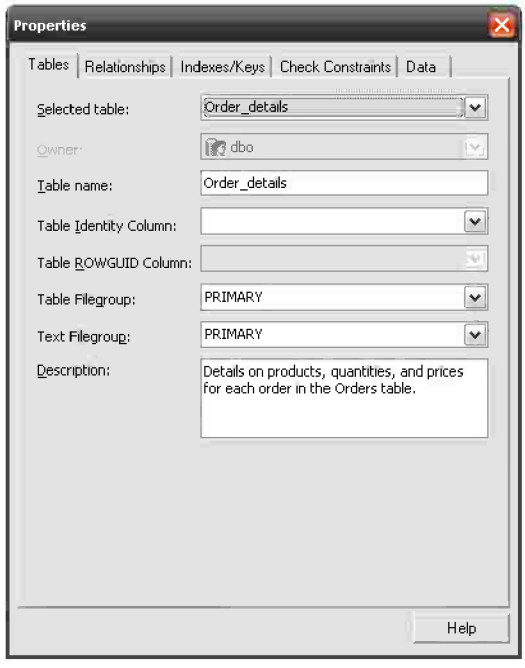


شكل ٤-١٥ تتطابق محتويات التبويب Lookup تماماً مع مثيلاتها داخل Access .

شرح مربع خصائص الجدول

يختلف مربع خصائص الجدول Properties داخل SQL Server عن نظيره داخل Access. من نافذة عرض تصميم الجدول (نافذة مصمم المشروعات) انقر التبويب Design ثم انقر الزر Property Sheet بالجزء Show/Hide وحينئذٍ يظهر مربع خصائص الجدول Properties والذى يحتوى بدوره على خمسة تبويبات هى Tables وRelationships وIndexes/keys وCheck Constraints وData (انظر شكل ٥-١٥).

وفيما يلى نوضح العناصر الأساسية بكل تبويب من هذه التبويبات.



شكل ٥-١٥ التبويب Tables بمربع خصائص الجدول Order Details.

استخدام التبويب Tables

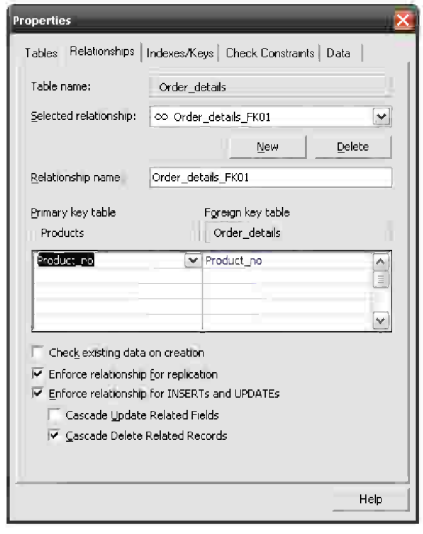
يحتوى التبويب Tables على الخصائص التالية (راجع شكل ٥-١٥):

- *الجدول المختار* **Selected Table** وهو عبارة عن قائمة منسدلة ، إلا أنك لا تستطيع اختيار جدول غير الجدول المفتوح حالياً.
- *اسم الجدول* **Table Name** ويحتوى على اسم الجدول والذى يمكن تغييره، ولكن مع مراعاة الشروط التى ذكرناها سابقاً، حيث لا يتيح **ANSI SQL** استخدام المسافات أو علامات الترقيم (عدا الشرطة التحتية) أثناء تسمية الجداول. فإذا أردت أن يتضمن الاسم الحروف الغير مسموح بها، ضع الاسم بين علامتى " " (راجع شكل ٥-١٥).
- *اسم عمود الهوية* **Table Identity Column** ويمكنك من خلاله تخصيص قيمة الخاصية **Identity** إلى أحد الحقول من خلال القائمة المنسدلة.

- *عمود الدليل* **Table ROWGUID Column** ومن خلاله يمكنك اختيار الحقل الذى يحتوى على قيمة **GUID** المستخدمة فى تمييز السجلات المختلفة.
- *مجموعة ملف الجدول ومجموعة ملف النصوص* **Table File Group And Text File Group** وكلاهما يحتوى عادةً على المجموعة **PRIMARY**، حيث يتيح **SQL Server** لمسئولى قاعدة البيانات **DBAs** إنشاء عدة ملفات نظام تشغيل لنفس الجدول، كما تتيح لهم أيضاً إنشاء حقول نصيَّة تحاكى حقول المذكرة الموجودة فى **Access** لاستخدامها مع مجموعة ملف الجدول أو ملف النصوص الخاص بهم.
- *الوصف* **Description** ويمكنك من خلال هذه الخاصية إضافة وصف مبسط عن الجدول، وهى نفس الخاصية الموجودة داخل **Access**.

استخدام التبويب Relationships

يحتوى التبويب **Relationships** على مجموعة الخصائص التى تحدد علاقة الجدول بالجداول الأخرى داخل قاعدة البيانات، وهذه الخصائص هى (انظر شكل ٦-١٥):



شكل ٦-١٥ محتويات التبويب **Relationships** بمربع خصائص الجدول **Order Details**.

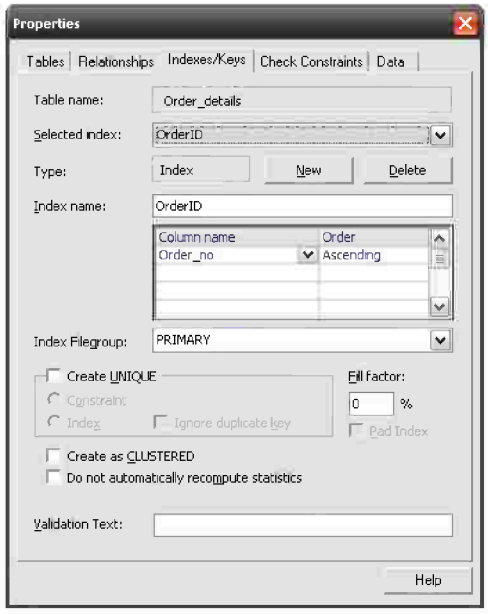
- *العلاقة المختارة* **Selected relationship** وهى عبارة عن قائمة منسدلة يتم من

خلالها اختيار نوع العلاقة بين الجدول وأحد الجداول الأخرى الموجودة بقاعدة.

- *اسم العلاقة* **Relationship name** ويحتوى على اسم مميز للعلاقة، ويمكنك تعديل هذا الاسم بشرط وضعه بين قوسين [ ].
- *العمود* **Primary key table** ويحتوى على اسم الجدول الأساسى بالعلاقة بالإضافة إلى الحقل الأساسى المستخدم فى إنشاء العلاقة.
- *العمود* **Foreign key table** ويحتوى على اسم الجدول الثانوى بالعلاقة بالإضافة إلى الحقل الأجنبى المستخدم فى إنشاء العلاقة.
- *نشط مربع الاختيار* **Check Existing Data on Creation** إذا أردت اختبار عملية تكامل البيانات حينما تقوم بإضافة علاقة جديدة، وهو الوضع الطبيعى داخل Access، لذا لا يحتوى Access على هذه الخاصية.
- *نشط مربع الاختيار* **Enforce Relationship for Replication** إذا أردت تطبيق تكامل البيانات على النسخ المطابقة من الجدول.
- *نشط مربع الاختيار* **Enforce Relationship for INSERTs and UPDATEs** إذا أردت تطبيق تكامل البيانات عند إجراء عمليات الحذف والتحديث تماماً كما فى الخاصية Enforce Referential Integrity داخل Access. وفى حالة تنشيط هذا المربع، يمكنك تنشيط أو تعطيل أيٍ من مربعى الاختيار Cascade Update Related Fields و Cascade Delete Related Records أو كلاهما، حيث يعمل المربعان بنفس طريقة عملهما داخل Access.

استخدام التبويب Indexes/Keys

يحتوى التبويب **Relationships** على مجموعة خصائص الفهارس الموجودة بالجدول، وهذه الخصائص هى (انظر شكل ٧-١٥):



شكل ٧-١٥ محتويات التبويب **Indexes/Keys** بمربع خصائص الجدول **Order Details**.

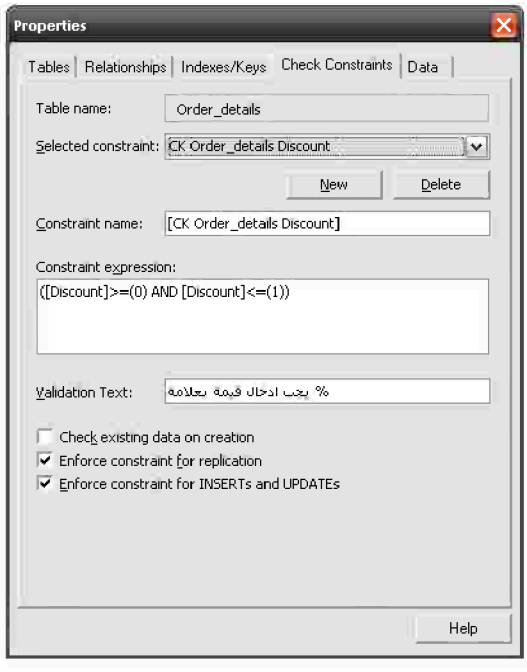
- تحتوى القائمة **Selected Index** على الفهارس المعرفة داخل الجدول، اختر منها الفهرس المناسب ولاحظ تغيير العنوان الموجود بمربع النص **Type** تبعاً لنوع الفهرس المختار سواءً كان مفتاحاً أساسياً **Primary Key** أم فهرس **Index** أم قيد فريد **Unique Constraint**.
- يحتوى مربع النص **Index name** على الاسم المقترح للفهرس المختار من قِبَل **SQL Server**. قم إن أحببت بتغيير هذا الاسم.
- يحتوى العمود **Column name** على اسم الحقل المستخدم فى عملية الفهرسة بينما يحتوى العمود **Order** على طريقة الفرز، وهى نفس العناصر الموجودة بالمربع الحوارى **Indexes** داخل **Access**.

- يتم تنشيط القائمة **Index Filegroup** فقط فى حالة إضافة فهرس جديد ومن خلالها يتم اختيار مجموعة الملفات التى ينتمى إليها الفهرس الجديد.
- يمكنك من خلال مربع المجموعة **Create UNIQUE** تعين خصائص الفهرس عند إنشاء فهرس جديد. نشط مربع الاختيار **Create UNIQUE** إذا أردت عدم تكرار نفس القيمة بحقل الفهرس المحدد، حيث يكافأ هذا المربع القيمة **No Duplicates** داخل **Access**. نشط زر الاختيار **constraint** أو **Index** لإنشاء قيد أو فهرس على الترتيب ثم نشط مربع الاختيار **Ignore duplicate key** إذا أردت تجاهل السجلات التى تحتوى على قيم مكررة بالحقل المختار عند عمليات الإضافة.
- نشط مربع الاختيار **Create as CLUSTERED** إذا أردت أن يقوم **SQL Server** بترتيب سجلات الجدول تبعاً لقيمة المفتاح الأساسى مما يعمل على زيادة كفاءة عمليات الإضافة والحذف التى تتم داخل الجدول. ويفضل عادةً تنشيط هذا المربع مع حقل المفتاح الأساسى داخل كل جدول من جداول قاعدة البيانات لتحسين كفاءتها.
- فى حالة عدم إضافة السجلات إلى الجدول بنفس ترتيب المفتاح الأساسى، فإن إضافة مساحة فارغة (تكون عادةً من ١٠% إلى ٢٠%) إلى صفحة الفهرس تعمل على تحسين كفاءة عمليات الإضافة، حيث يمكنك تعيين هذه النسبة من خلال مربع النص **Fill Factor**. يمكنك أيضاً تنشيط مربع الاختيار **Pad Index** إذا أردت الاحتفاظ بسجلين فارغين داخل الجداول العنقودية **Clustered tables**.
- نشط مربع الاختيار **Do not automatically recomputed statistics** إذا أردت زيادة سرعة إنشاء الفهارس داخل الجداول الكبيرة، إلا أنه هذا قد يكون على حساب كفاءة الاستعلامات المبنية على هذه الجداول.
- يمكنك من خلال مربع النص **Validation Text** تعيين نص الخطأ الذى يظهر عند محاولة إضافة أو تحديث قيمة تتعارض مع القيد **UNIQUE**.أما **Access**

فيحتوى على رسالة معرفة مسبقاً تظهر عند اختيار القيمة **No Duplicates**.

استخدام التبويب Check Constraints

يحتوى التبويب **Check Constraints** على مجموعة خصائص القيود المعرفة بالجدول، حيث تكافئ قيود الاختيار **CHECK Constraints** داخل **SQL Server** قواعد التحقق من الصحة داخل **Access**، وهذه الخصائص هى (انظر شكل ٨-١٥):



شكل ٨-١٥ محتويات التبويب **Check Constraints** بمربع خصائص الجدول **Order Details**.

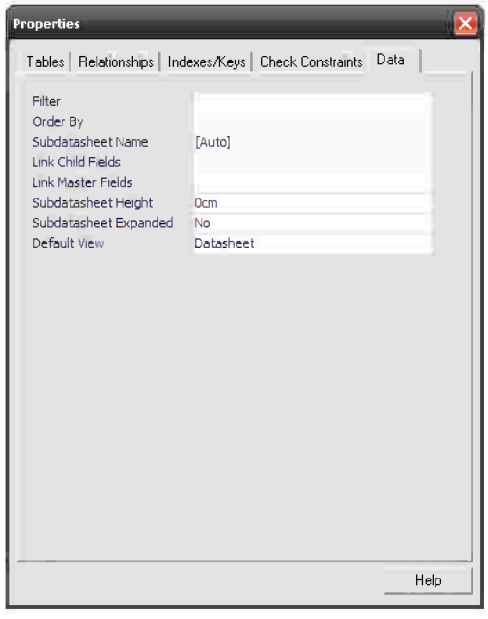
- اختر القيد الذى ترغب فى التحكم فى خصائصه من خلال القائمة **Selected constrain** والتى تحتوى على جميع القيود المعرفة داخل الجدول. فعلى سبيل المثال، يحتوى الجدول **Order Details** على ثلاثة قيود هى **CK_Discount** و**CK_Quantity** و**CK_UnitPrice**. ويمكنك إنشاء قيد جديد بنقر زر **New** أو حذف أحد القيود بنقر زر **Delete**.
- يحتوى مربع النص **Constraint name** على اسم القيد المختار وهو الاسم الذى

يقوم SQL Server بتعيينه، قم بتغييره إن أحببت.

- يحتوى مربع النص Constraint Expression على تعبير القيد (التحقق من الصحة) والذى تكون قيمته دائماً إما TRUE أو FALSE. لإنشاء قيد جديد، انقر زر New ثم قم بإدخال تعبير القيد إلى مربع النص Constrain Expression واسم القيد إلى مربع النص Constraint name.
- يحتوى مربع النص Validation Text على نص رسالة الخطأ الذى يظهر للمستخدم عند محاولة إضافة أو تحديث قيمة تتعارض مع القيد المحدد. وتكافأ هذه الخاصية تماماً الخاصية Validation Text داخل Access.
- نشط مربع الاختيار Check existing data on creation إذا أردت اختبار البيانات الموجودة بالفعل داخل الجدول للتأكد من تحقيقها للقيد المحدد بمجرد إنشائه.
- نشط مربع الاختيار Enforce constraint for replication إذا أردت تطبيق القيد على بيانات النسخة المطابقة Replicated data.
- نشط مربع الاختيار Enforce constraint for INSERTs and UPDATEs إذا أردت تطبيق القيد عند إجراء عمليات الإضافة والتحديث على الجدول.

استخدام التبويب Data

يحتوى التبويب Data على عدد من الخصائص الإضافية الموجودة داخل SQL Server والتى تدعم بعض سمات Access داخل مشروعات البيانات ADP مثل صفحات البيانات الفرعية Subdatasheets على سبيل المثال والتى يمكنك من خلالها عرض بيانات جدول مرتبط من خلال صفحة بيانات الجدول الأساسى، وهذه الخصائص هى (انظر شكل ٩-١٥):



شكل ٩-١٥ محتويات التبويب **Data** بمربع خصائص الجدول **Order Details**.

- يمكنك تعيين عامل تصفية محدد لبيانات الجدول من خلال مربع النص **Filter**.
- يمكنك تعيين الحقل المستخدم فى عملية الفرز من خلال مربع النص **Order By**.
- اختر الجدول (أو العرض أو الدالة) المستخدم كصفحة بيانات فرعية من القائمة المنسدلة **Subdatasheet Name**، أو اختر **[Auto]** إذا أردت نفس صفحة البيانات الفرعية الموجودة من قبل داخل **Access** أو **[None]** إذا أردت عدم إظهار صفحات فرعية من الأساس.
- قم بإدخال اسم الحقل أو الحقول المستخدمة فى عملية الربط بصفحة البيانات الفرعية (الجدول الفرعى أو العرض أو الدالة) إلى مربع النص **Link Child Fields**.
- قم بإدخال اسم الحقل أو الحقول المستخدمة فى عملية الربط بالجدول الحالى إلى مربع النص **Link Master Fields**.
- قم بإدخال ارتفاع صفحة البيانات الفرعية داخل الجدول إلى مربع النص

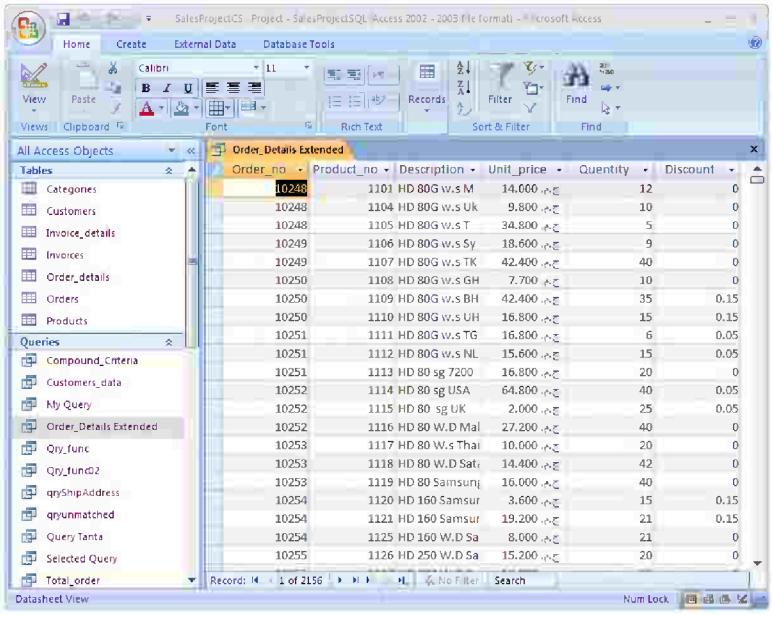
Subdatasheet Height.

- من القائمة المنسدلة Subdatasheet Expanded اختر Yes إذا أردت إظهار صفحة البيانات الفرعية باستمرار أو No إذا أردت إظهارها فقط عند توسيع الزر ⊞ داخل الجدول.
- من القائمة المنسدلة Default View، ابق على الخيار Datasheet لعرض صفحة البيانات الفرعية بالطريقة المعتادة (طريقة عرض صفحة البيانات) أو اختر PivotTable أو PivotChart لعرضها فى صورة جدول محورى أو مخطط محورى على الترتيب.

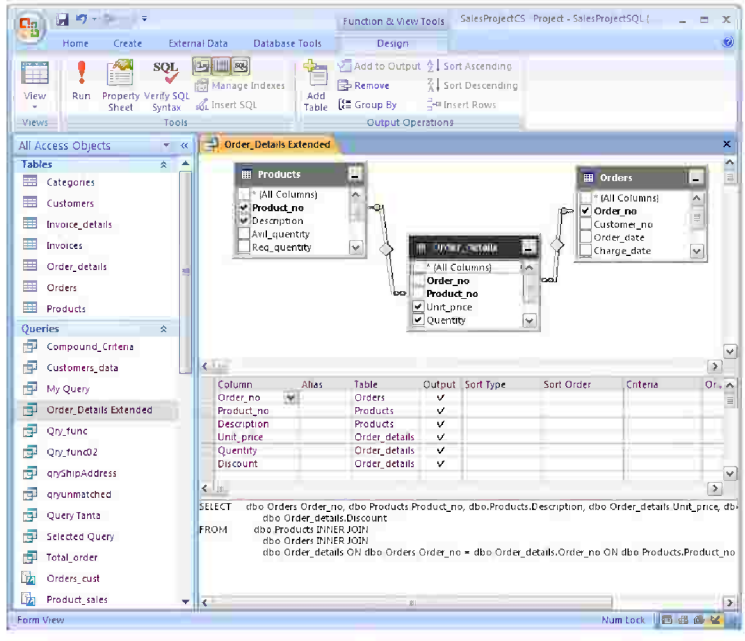
فهم عروض خادم SQL (SQL Server Views)

تشبه العروض Views الجداول فى طريقة عرض صفحة البيانات. يوضح شكل ١٠-١٥ العرض Order_Details Extended فى طريقة عرض صفحة البيانات بينما يوضح شكل ١١-١٥ نفس العرض ولكن فى طريقة عرض التصميم والذى يشبه بدوره نافذة تصميم الاستعلام داخل Access ومنه نستخلص الملاحظات التالية:

- تتكون نافذة عرض التصميم من ثلاثة أقسام يمكن ظهور واحد منها على الأقل ويتم التحكم فيها عن طريق الأزرار الموجودة فى شريط التصميم الموجود أعلى النافذة .
- القسم الأعلى هو قسم "المخطط" Diagram والذى يحتوى على قوائم الجداول المستخدمة فى تكوين العرض .ويتم استعمال رمز المفتاح 🔑 للتعبير عن حقل "مفتاح أساسى" Primary Key ورمز مالانهاية ∞ للتعبير عن حقل "مفتاح أجنبى" Foreign Key .كما تظهر حقول المفاتيح الأساسية بخط سميك .



شكل ١٠-١٥ العرض Order_Details Extended فى طريقة عرض صفحة البيانات.



شكل ١١-١٥ العرض Order_Details Extended فى طريقة عرض التصميم.

- القسم الأوسط هو قسم "الشبكة" Grid وهو مشابه إلى حدٍ ما لقسم QBE الموجود فى استعلامات Access، حيث يحتوى على الحقول المستخدمة فى تكوين العرض.
- القسم السفلى هو قسم SQL والذى يحتوى على جملة Transact-SQL المستخدمة فى إنشاء العرض .
- أضاف شريط عرض التصميم الموجود بالجزء العلوى من الشاشة خمسة أزرار جديدة يمكن بيانها كما يلى:

١. المخطط Digram : ويستخدم لإظهار وإخفاء قسم المخطط الموجود بأعلى النافذة.

٢. الشبكة Grid : ويستخدم لإظهار وإخفاء قسم الشبكة الموجود فى وسط النافذة .

٣. SQL : ويستخدم لإظهار وإخفاء مربع النص الموجود أسفل النافذة والذى يحتوى على جملة Transact-SQL المستخدمة فى إنشاء العرض.

٤. التحقق من بناء عبارة SQL : ويستخدم لتشغيل مختبر القواعد الذى يقوم باختبار صحة جملة SQL دون أن يقوم بتشغيل العرض نفسه .

٥. تجميع حسب Group By : ويستخدم لإضافة عمود Group By لتجميع السجلات حسب حقول معينة. وهى مثل Group By الموجودة فى Access.

- بالنظر إلى جملة SQL المستخدمة لبناء العرض الذى بين أيدينا تجدها كما يلى:

```
SELECT     dbo.Orders.Order_no,
           dbo.Products.Product_no,
           dbo.Products.Description,
           dbo.Order_details.Unit_price,
           dbo.Order_details.Quentity,
           dbo.Order_details.Discount
FROM       dbo.Products INNER JOIN
           dbo.Orders INNER JOIN
```

```
                dbo.Order_details ON
dbo.Orders.Order_no = dbo.Order_details.Order_no
ON dbo.Products.Product_no = dbo.Order_details.Product_no
```

تلاحظ تشابه الكود إلى حدٍ كبير مع مثيله داخل Access SQL.

يتم داخل العروض استخدام الدالة Convert() فى تحويل نوع بيانات الحقل المحسوب من عددى إلى عملة Money.

لاحظ أيضاً عدم إمكانية تطبيق عمليات الفرز باستخدام Order By المستخدمة مع SQL العادية ، ولكن من حسن الطالع أنه يمكنك تطبيق عوامل الفرز والتصفية فى طريقة عرض صفحة البيانات عن طريق شريط التصميم أو قائمة Records كما فى Access تماماً.

عند استخدام ORDER BY داخل العروض والجداول الناتجة من الدوال السطرية، يجب استخدام المعامل TOP داخل العبارة SELECT. وكنا فى عروض SQL Server 2000 عادةً ما نستخدم التركيب SELECT TOP 100 PERCENT. أما فى Access 2007 فيجب تعيين الخاصية Order By المصاحبة للعرض وتخصيصها بالحقل المستخدم فى الفرز أو الحقول المستخدمة فى الفرز مع فصل هذه الحقول بعلامة الفاصلة (,) أو استخدام التركيب TOP 2147483647 بدلاً من TOP 100 PERCENT.

تعديل تصميم العرض

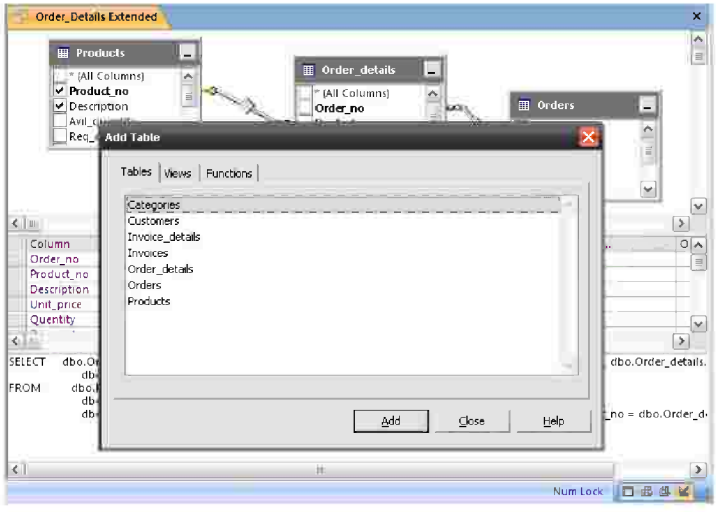
لمعرفة كيفية تعديل تصميم العرض، دعنا نقوم بإضافة المزيد من الجداول والشروط للعرض الذى بين أيدينا. تابع معنا الخطوات الآتية:

١. من لوحة التنقل الموجودة بالجانب الأيسر من إطار مشروع البيانات SalesProjectCs تأكد من ظهور التبويب Queries وإلا قم بإظهاره.

٢. اختر عرض Order_Details Extended ثم انقر زر Design، يظهر العرض فى طريقة التصميم (راجع شكل ١١-١٥).

٣. من شريط الأدوات الموجود بالجزء العلوى من شاشة التصميم، انقر الزر من

التبويب Design، وحينئذٍ يظهر المربع الحوارى Add Table (انظر شكل ١٢-١٥) .



شكل ١٢-١٥ يمكنك من خلال هذا المربع الحوارى إضافة الجداول والعروض والدوال إلى الجدول.

٤. حرك جدولى Products وOrder Details إلى اليسار ثم انقر جدول Customers نقراً مزدوجاً لإضافته إلى نافذة العرض.

٥. انقر مربع الاختيار الموجود على يسار كلٍ من Customer_no من جدول Orders وCustomer_name من جدول Customers وذلك لإضافتهما إلى العرض.

٦. انقر بزر الفأرة الأيمن خط الصلة الموجود بين جدولى Order Details وProducts، تظهر قائمة مختصرة، اختر منها Properties، يظهر المربع الحوارى Properties والذى يحتوى على خصائص الصلة (انظر شكل ١٣-١٥).

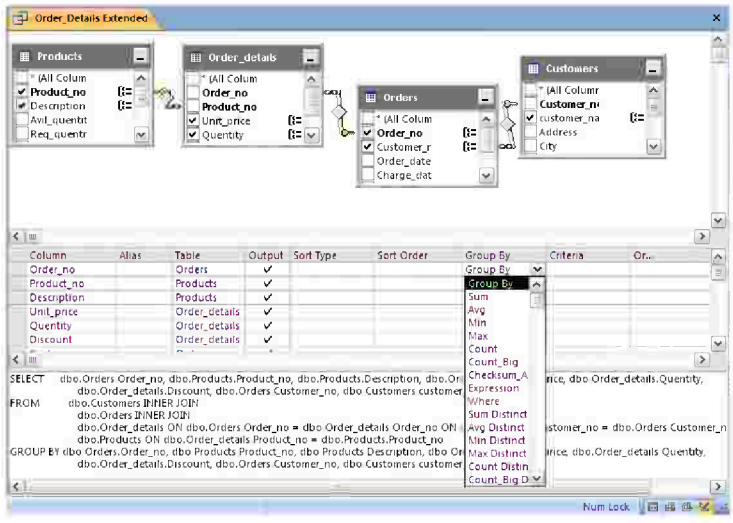
٧. اختر نوع الصلة من القائمة المنسدلة التى تحتوى على المعاملات.إذا نشطت مربع الاختيار All rows from Order_Details فقط سيتم إنشاء صلة خارجية يمنى Right Outer Join ويتم إنشاء مربع فى الجهة اليمنى لمعين الصلة هكذا

. وإذا نشطت مربعى الاختيار معاً، فسيتم إنشاء صلة خارجية كاملة **Full Outer Join** ويصبح رمز الصلة على شكل مربع هكذا .

شكل ١٣-١٥ التحكم فى خصائص الصلة.

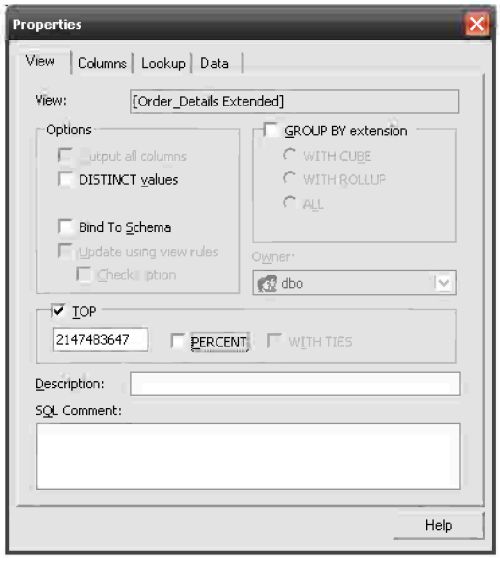
٨. قم بتعطيل مربعى الاختيار الموجودين بالمربع الحوارى **Properties** إذا كنت قد قمت بتنشيط أيٍ منهما أو كلاهما ثم قم بإغلاق المربع الحوارى.

٩. انقر زر Group By من التبويب **Design** بشريط التصميم، يتم إضافة عمود **Group By** إلى العرض كما يتم إضافة الرمز ∑ بجانب الحقول المختارة بالجزء العلوى من العرض، حيث يمكنك من خلال العمود **Group By** (تماماً كما فى **Totals** داخل **Access**) اختيار **Group By** أو **Avg** أو **Max** أو **Min** وغيرها من الخيارات الأخرى التى لا يوجد بعضها داخل **Access** (انظر شكل ١٤-١٥).



شكل ١٤-١٥ إضافة العمود Group By إلى العرض.

١٠. انقر زر الخصائص لإظهار خصائص العرض، وحينئذٍ يظهر المربع الحوارى Properties (انظر شكل ١٥-١٥) .

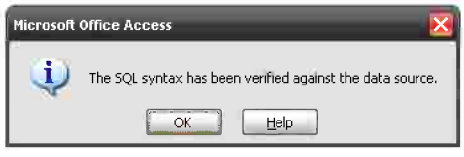


شكل ١٥-١٥ التحكم فى خصائص العرض.

١١. من التبويب View، نشط مربع الاختيار Group By Extension ثم نشط زر

الاختيار **With Cube**. ومن مربع المجموعة **Options**، نشط مربع الاختيار **Distinct values** إذا أردت عدم تكرار البيانات المتشابهة. قم بإغلاق المربع الحوارى **Properties**.

١٢. انقر الزر **Verify SQL Syntax** للتأكد من صحة التعديلات التى قمت بها فى الخطوات السابقة .تظهر رسالة تخبرك بأن جملة **SQL** صحيحة ولا تحتوى على أخطاء (انظر شكل ١٦-١٥) .



شكل ٦-١٥ يخبرك مربع الرسالة بصحة عبارة **SQL**.

١٣. انقر زر **Ok** ثم قم بإغلاق نافذة تصميم العرض ولا تحفظ التغييرات التى قمت بها وذلك بنقر زر **No** عندما تظهر رسالة الحفظ.