التعليم عن بعد

محتوى

قواعد البيانات

لـ د. مصلح العضايلة

IbtiHAlinO
للحصول على الملزمة من الإنترنت : Ibtihalino.blogspot.com

المحاضرة التمهيدية

أهمية قواعد البيانات :

تعتبر قواعد البيانات ذات أهمية كبيرة لتقدم أي مجتمع خصوصا و نحن نعيش عصر التقنية و المعلوماتية و بالتالي فنحن بحاجة إلى آلية لتنظيم البيانات (والتي أصبحت في عصرنا كبيرة و متنوعة) و المحافظة عليها من التلف و ضمان أمنها.

<u>سهولة إدارتها</u>

سهولة في التخزين والإضافة والتعديل
وسرعة في البحث والاستعلام
التحليل الإحصائي
سهولة وسرعة استخراج التقارير

محتوى المقرر :

- مقدمة في أنظمة قواعد البيانات (Introduction To Database)
- مبادئ قواعد البيانات العلاقية (Fundamentals of Relational Database)
- تصميم نموذج الكيان و العلاقة الرابطة (Design of Entity-Relationship Diagram (ERD)
- تصميم قواعد البيانات العلاقية (Relational Database Design
- تطبيق قواعد البيانات و امن المعلومات
- ادارة قواعد البيانات
- نظام إدارة قواعد البيانات أكسس 2007

المراجع و المصادر التعليمية :

- د. خالد ناصر السيد (اصول تصميم قواعد البيانات و لغة SQL) مكتبة الرشد للنشر،الطبعة الثالثة ،٢٠٠٦.
- عبدالله بن عبدالعزيز الموسى : (مقدمة في الحاسوب والانترنت) ، مؤسسة شبكة البيانات، الرياض، الطبعة السادسة ، ٢٠١٠.

المراجع و المصادر المساعدة :

- المصري و نفاذا (مبادئ انظمة قواعد البيانات) جريج توبن للنشر ، نيويورك ، الطبعة الخامسة ، ٢٠٠٧
- خطوة خطوة (Step by Step Microsoft Office Access) الدار العربية للعلوم ، الطبعة الاولى ،٢٠٠٧

مقدمة في أنظمة قواعد البيانات

-أصبحت قواعد البيانات و تطبيقاتها عنصرا جوهرياً في تسيير أمور الحياة اليومية في المجتمع المعاصر ، حيث أن جميع الأنشطة التي يمارسها أفراد المجتمع من تسجيل مواليد و وفيات و نتائج دراسية و وثائق السفر و العمليات البنكية و غيرها الكثير يجب فيها التعامل مع احد قواعد البيانات .
-كافة الأنشطة السابقة تدخل في نطاق التطبيقات التقليدية لقاعدة البيانات .
-توجد حالياً تطبيقات متقدمة لقواعد البيانات مثل استخدام الذكاء الاصطناعي و التجارة الالكترونية

خواص قواعد البيانات :

-تمثل بعض مظاهر العالم الحقيقي. أي إنها تمثل حالة من حالات البيانات التي تصف موضوع حقيقي
-تمثل مجموعة من البيانات المتلاصقة منطقيا و تحتوي على معنى ضمني
-يتم تصميمها و تخزين البيانات فيها من أجل غرض معين

مفهوم قواعد البيانات :

-قواعد البيانات (Database): هي أسلوب محدد لتنظيم المعلومات يبسط كيفية إدخالها و تعديلها و استخراجها في صورة ملائمة و مفهومة للمستخدم لمجموعة مشتركة من البيانات المترابطة و المتجانسة منطقيا
-هي مجموعة من عناصرِ البيانات المنطقية المرتبطة مع بعضها البعض بعلاقة معينة ، و تتكون قاعدة البيانات من جداول (واحد أو أكثر). و يتكون الجدول أعمدة (حقول Fields) ومن صفوف (سجلات Record) .

نظم ملفات البيانات :

- استخدام الملفات في تخزين البيانات.
- استخدام المبرمجون ملفات البيانات في تخزين المعلومات لفترة طويلة.
- أدى استخدام الملفات إلى ظهور بعض المشاكل و العيوب .
- أدى إلى تطوير أسلوب التعامل مع الملفات و بذل الجهد و الوقت
- في نظام معالجة البيانات كان كل برنامج يصمم لأداء غرض معين وله الملفات الخاصة به ، دون وجود إطار عام يربط جميع البرامج أو يسمح بإضافة برامج جديدة بسهولة.

أنواع الملفات:

-**ملف تتابعي :** يتم تخزين سجلات البيانات بشكل تتابعي بنفس ترتيب وصولها للملف سجل بعد سجل. لاسترجاع البيانات تجري عملية قراءة السجلات من أول سجل إلى أخر سجل و بشكل تتابعي.

-**ملف عشوائي :** يتم تخزين سجلات البيانات بشكل عشوائي مع معرفة موقع أو عنوان كل سجل بيانات ، و تتم قراءة البيانات مباشرة عن طريق العنوان.

-**ملف مفهرس:** يستخدم فهرس أشبه بفهرس الكتاب من خلاله يتم الوصول إلى أي سجل بيانات ، يتم عمل الفهرس من خلال احد حقول البيانات.

مشاكل الملفات:

- **تكرار البيانات :** تكرار البيانات في أكثر من ملف مما يضيع حيز التخزين و الجهد و الوقت.
- **عدم تجانس أو توافق البيانات:** نفس المعلومة تكون مخزنه في أكثر من ملف عند تعديلها قد لا نعدلها في الملفات الأخرى.
- **عدم المرونة :** عملية التعديل و الحذف تتطلب جهد و وقت و كلفة عالية.
- الافتقار إلى المواصفات القياسية.
- معدل منخفض لإنتاج البرامج
- مشاركة محدودة جداً بين البرامج المختلفة و ملفات البيانات.
- **صعوبة الصيانة** أي تعديل لملف يلزم تعديل كافة البرامج الخاصة به .
- امن سرية المعلومات تكون على نطاق محدود.

نظم قواعد البيانات :

-نشأت قواعد البيانات و نظم قواعد البيانات من اجل إيجاد بديل لملفات البيانات و نظم معالجتها بحيث تحل كافة المشكلات و القيود و الصعوبات التي يواجهها المستخدمون في تعاملهم مع الملفات.
-**البيانات :** هي كافة البيانات المطلوب إدخالها أو الاستعلام عنها ، حيث كل بيان يمثل عنصر مستقل مثل (اسم المريض ، رقم الغرفة ، العنوان ، ....)
-**المعلومات** : هي البيانات التي تمت معالجتها و وضعها في صورة ملائمة و مفهومة للمستخدم.
-نظم قواعد البيانات هي أسلوب محدد لتنظيم البيانات يبسط كيفية إدخالها و تعديلها و استخراجها اما بنفس الشكل المدخل أو مجمعة في صورة إحصائية أو تقارير أو شاشات استعلام مع التحكم في كل عملية.
-تصميم قاعدة البيانات يشمل تحديد أنواع البيانات و التراكيب و القيود على كافة البيانات.
-بناء قاعدة البيانات هو عملية تخزين البيانات نفسها في وسط تخزين تتحكم به نظم قواعد البيانات.
-عند تصميم قاعدة بيانات يجب تحديد المستخدمين و التطبيقات الذين سيستخدمون قاعدة البيانات.

أمثلة نظم إدارة قواعد البيانات :

يمكن إنتاج و معالجة قاعدة البيانات باستخدام الحاسب الآلي بواسطة مجموعة من البرامج التطبيقية المصممة خصيصاً لهذا الغرض أو بواسطة نظم إدارة قواعد البيانات (DBMS) مثل :

- MS-Access
- Oracle
- Sybase
- Power Builder
- Informix

الفرق بين نظم قواعد البيانات و نظم الملفات التقليدية :

يوجد العديد من الخواص التي تفرق بين قواعد البيانات و نظم الملفات التقليدية وهي :

الوصف الذاتي للبيانات (Self-Description Nature):

تحتوى قواعد البيانات علي البيانات و وصف البيانات و ذلك عن طريق إنشاء فهرس البيانات و الذي يحتوي على ما يسمى (Meta-data)

الفصل بين البرامج و البيانات(Program/Data Insulation):

لا تحتوى البرامج على وصف البيانات بل يوجد فصل بينهما مما يتيح إمكانية تعديل شكل البيانات بدون الحاجة لتعديل البرامج

المشاركة في البيانات و التعامل مع العديد من المستخدمين (Data Sharing and Multi-user system):

تتيح قواعد البيانات المشاركة في استخدام البيانات و كذلك تعطي إمكانية تعامل العديد من المستخدمين مع نفس قواعد البيانات في نفس الوقت بدون مشاكل

قواعد البيانات و نظم إدارة قواعد البيانات :

- نظام إدارة قواعد البيانات : هو مجموعة من البرامج التي يمكن استخدامها في إنشاء و معالجة قاعدة بيانات.
- نظام إدارة قواعد البيانات هو نظام برامجي متعدد الأغراض يسهل تعريف و بناء و معالجة قواعد البيانات التطبيقية.
- يمكن أن تصمم قاعدة بيانات واحدة تستخدم مع العديد من البرامج و التطبيقات.

خواص أخرى لقواعد البيانات :

-يمكن أن تكون قاعدة البيانات في أي حجم فيمكن أن تحتوي على القليل من السجلات أو المئات منها ويمكن أن تحتوي على مئات الملايين من السجلات
-يمكن أن يتم إنشائها و التعامل معها يدويا أو باستخدام الحاسبات الآلية
-إذا تم استخدام الحاسب الآلي لإدارة قواعد البيانات فإن ذلك يتم عن طريق مجموعة من البرامج التي تصمم خصيصا لذلك أو عن طريق استخدام نظم إدارة قواعد البيانات ..

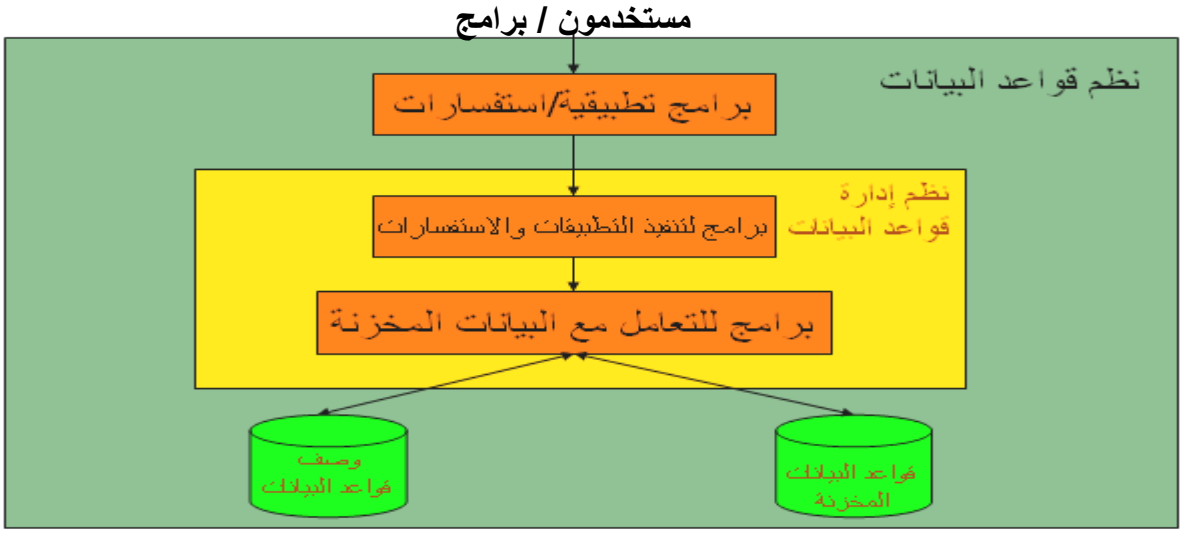
نظام إدارة قواعد البيانات (Database Management System DBMS) :

تسمى قواعد البيانات + نظم إدارة قواعد البيانات بنظم قواعد البيانات
(قواعد البيانات + نظم إدارة قواعد البيانات ← نظم قواعد البيانات)
(Database (DB) + DBMS → Database System (DBS))

مميزات استخدام قواعد البيانات :

-ندرة التكرار وإمكانية التحكم في تكرار البيانات
-امن و سرية البيانات عالية جداً
- فرض القيود على المستخدمين الذين ليس لهم صلاحيات معينة
- توفير بيئة تخزين مناسبة و صعوبة فقد البيانات
-السماح باستنباط معلومات من البيانات المتواجدة
-توفير واجهات متعددة لتعامل المستخدم مع البيانات
-تمثيل العلاقات المعقدة بين البيانات بسهولة
-تكامل البيانات بشكل عالي و متناسق.
-سهولة الصيانة حيث أي تعديل يتم بكل سهولة و من مكان واحد
-توفير طرق متعددة للحصول على النسخ الاحتياطية و كذلك معالجة البيانات في حالات الأعطال التي قد تحدث لقواعد البيانات
-تساعد على وضع معايير قياسية للتعامل مع البيانات
-تقليل زمن تطوير البرامج
-المرونة الشديدة في استخدام و تعديل البيانات
-توفير بيانات على درجة عالية من التحديث
-اقتصادية الاستخدام
-المرونة العالية في مشاركة البيانات و بكل سهولة

مخطط يوضح قواعد البيانات و نظم إدارتها



مستخدم قواعد البيانات :

مدير قواعد البيانات(DBA):

يقوم بإدارة قواعد البيانات و التحكم في صلاحيات العمل و مراقبة النظام و تحسين أداء قواعد البيانات .

مصمم قواعد البيانات (DB Designer):

يقوم بتصميم قواعد البيانات ليتم إنشائها و بنائها بطريقة ذات كفاءة عالية طبقا لمتطلبات المستخدم .

مستخدم قواعد البيانات (End User):

بعض المستخدمين يكون لديهم الخبرة الكافية لإعداد الاستفسارات المطلوبة بلغة الاستفسارات ، و بعض المستخدمين ليس لديهم الخبرة فيتم إنشاء برامج خاصة لهم يقومون بتشغيلها للحصول على المطلوب .

محلل النظم و مبرمج النظم (Analyst & Programmer) :

- يقوم بتحديد متطلبات المستخدم و تطوير هذه المواصفات المطلوبة لتحديد المطلوب من قواعد البيانات.
- بينما يقوم مبرمج النظم بتنفيذ المتطلبات لإنشاء التطبيقات المناسبة
- هندسة النظم هي عملية تحليل النظام بالإضافة لعملية إنشاء البرامج التطبيقية

(محلل النظم + مبرمج النظم ← مهندس النظم)
(Analyst + Programmer → Software Engineer)

متى لا نستخدم قواعد البيانات؟

- إذا كانت تكلفة الإعداد عالية بالنسبة لحجم المشروع
- إذا كانت قاعدة البيانات و التطبيقات بسيطة و سهلة
- إذا كان المشروع يحتاج لسرعة استجابة عالية جدا و بشكل ضروري
- إذا كان العمل لا يحتاج إلى بيئة ذات عدة مستخدمين

أشخاص يتعاملون مع قواعد البيانات بطريقة غير مباشرة :

هؤلاء لا يهتمون بقواعد البيانات ذاتها و لكنهم يقدمون لمستخدم قواعد البيانات البيئة اللازمة لهم وهم :

مصمموا و منفذوا نظم إدارة قواعد البيانات:

يقومون بتصميم و تنفيذ نظم إدارة قواعد البيانات نفسها.

مطوروا البرامج المساعدة :

يقومون بتطوير البرامج المساعدة مثل برامج تحليل النظم ، تصميم النظم ، إنشاء و تطوير التطبيقات ، إنشاء التقارير و واجهات التطبيق .

المشغلون و أفراد الصيانة :

يقومون بتشغيل النظم و إدارتها و صيانتها و كذلك صيانة البرامج و الأجهزة المستخدمة في إنشاء و تطوير قواعد البيانات.

هيكلية نظم إدارة قواعد البيانات
Architecture of DBMS

مكونات بيئة نظم قواعد البيانات :

- **المكونات المادية :** المكونات المادية من حواسيب و خوادم و أجهزة و معدات .
- **المكونات البرمجية :** نظم البرمجة الخاصة بقواعد البيانات
- **البيانات :** هي العنصر المركزي لقواعد البيانات
- **الإجراءات و العمليات :** هي التعليمات التي تحكم التصميم و استخدام قواعد البيانات بالشكل الأفضل .
- **المستخدمون :** الأشخاص الذين يتعاملون مع قواعد البيانات

مبادئ قواعد البيانات :

قاعدة البيانات هي أسلوب محدد لتنظيم المعلومات يبسط كيفية الإدخال و الإخراج في إطارات مختلفة مع التحكم في كل عملية.

- **أنواع قواعد البيانات :**

-قواعد البيانات الشبكية (Network Database)
-قواعد البيانات الهرمية (Hierarchical Database)
-قواعد البيانات العلاقية (Relational Database)

هيكلية نظم إدارة قواعد البيانات : (DBMS Architecture)

١. **النظام المركزي (Centralized system):**

وفيه تتواجد جميع وظائف قواعد البيانات و النظم التطبيقية و واجهات التعامل مع المستخدم وغيرها من البرامج في نظام واحد مركزي .

٢. **نظام الخادم - العميل (Client-Server):**

وفيه يحتوي العميل (يكون عادة عبارة عن حاسب شخصي) النظم التطبيقية و واجهات التعامل مع المستخدم بينما يقوم الخادم بوظائف قواعد البيانات (وفي بعض النظم الحديثة قد يقوم العميل ببعض وظائف قواعد البيانات)

نماذج البيانات (Data Models) :

- هو وصف للبيانات أو أنشطة أو أحداث في مكان ما لجعل البيانات منظمة و مفهومة.
- بناء البيانات هو تحديد نوع البيانات و العلاقات بين البيانات و القيود المفروضة عليها
- يمكن أن يحتوي نموذج البيانات على بعض العمليات الأساسية (مثل كيفية تعديل أو استرجاع البيانات)
- في نظم البيانات الشيئية يمكن أن يحتوي النموذج علي مجموعة من العمليات التي يعرفها المستخدم علي البيانات

حالات قواعد البيانات (Instances) :

البيانات المتواجدة داخل قواعد البيانات في لحظة معينة تسمي "حالة قواعد البيانات أو الوضع الحالي لقواعد البيانات (DB State or Current Set of Occurrence or Instance)

-يتم إنشاء الوضع الابتدائي لها عند إدخال البيانات لأول مرة ثم يتغير وضعها عند إجراء العمليات المختلفة على البيانات (إضافة – حذف – تعديل)
-تسمى حالة البيانات هذه "Extension"

هيكلية نظم قواعد البيانات (DB System) :

تحتوي نظم قواعد البيانات على ثلاث مستويات من المخططات و ذلك لدعم الخواص التي يجب أن تقدمها نظم إدارة قواعد البيانات :

١. مستوى البيانات الخارجي (The External or View Level)
٢. المستوى المفاهيمي (The Conceptual Level)
٣. المستوي الداخلي (Internal Level)

١. مستوى البيانات الخارجي The External or View Level:

- هو الجزء الذي يستهدف المستخدمين
- التخاطب و الاتصال و استرجاع البيانات
- يستخدم برامج تطبيقية و برامج رسومية أو مباشر
- مرحلة التحليل

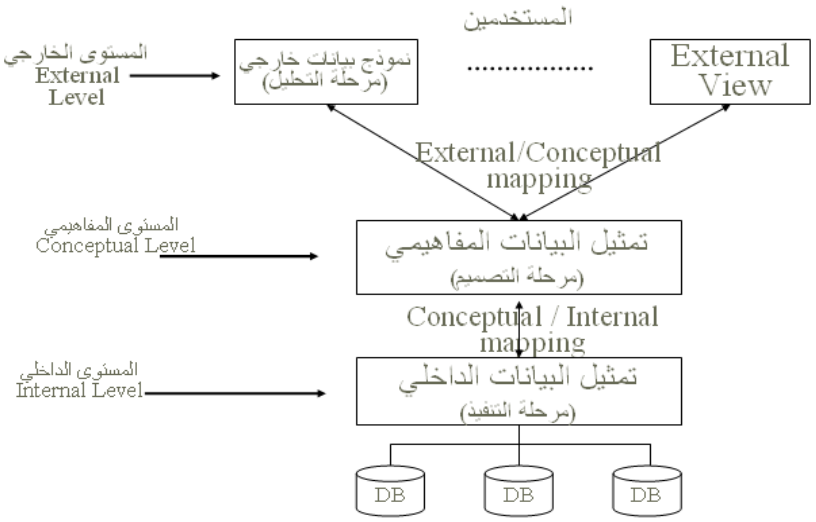
٢. المستوى المفاهيمي The Conceptual Level:

- يحتوي علي Conceptual Schema التي تصف بناء البيانات في قواعد البيانات – نموذج البيانات المنطقي
- تقوم بإخفاء التفاصيل الخاصة بالبناء الفعلي للبيانات
- تقوم بوصف الكيانات ، نوع البيانات ، العلاقات ، القيود و كذلك العمليات التي يعرفها المستخدم
- يمكن استخدام Conceptual data model أو Representation data model في بناء هذا المستوي و يطلق عليه مرحلة التصميم

٣. المستوى الداخلي (Internal Level):

- وهو يحتوي علي المخطط الداخلي و الذي يقوم بوصف التخزين الفعلي لقواعد البيانات و عملية إنشاء قاعدة البيانات.
- مرتبط بالأجهزة و البرامج
- هذا المخطط الداخلي يتم وصفه باستخدام نموذج (Physical Data Model) الذي يركز على تمثيل و إنشاء قواعد البيانات المصممة على جهاز الحاسب و يطلق عليه مرحلة التنفيذ.

شكل يوضح الثلاث مستويات لمخططات قواعد البيانات : (The Three-Schema Architecture)



ملاحظات علي الثلاث مستويات لمخططات نظم قواعد البيانات :

- تعتبر طريقة مناسبة و أداة سهلة للمستخدم ليفهم و يتخيل مستويات مخططات البيانات داخل نظم قواعد البيانات
- معظم نظم إدارة قواعد البيانات لا تفصل تماما بين المستويات الثلاث
- تقوم نظم إدارة قواعد البيانات بتحويل المخططات بين المستويات الثلاث و تحويل البيانات بين هذه المستويات (mapping)
- التحويل بين المخططات (mapping) يعتبر عملية مستهلكة للوقت ولذلك فإن بعض نظم إدارة قواعد البيانات لا تدعم المستوي الثالث (External Level)
- معظم نظم إدارة قواعد البيانات تحتوي علي المستوي الثالث (External level) داخل المستوي الثاني (Conceptual level)
- بعض نظم إدارة قواعد البيانات تحتوي علي التفاصيل الفعلية (Physical details) داخل المستوي الثاني (Conceptual level)

استقلالية البيانات (Data Independence) :

- هي المقدرة علي تغيير مخطط البيانات في مستوي معين بدون وجوب تغير المخطط في المستويات الأخرى
- عند تغيير المخطط في مستوى معين فإن الذي يتغير هو طرق التحويل (mapping) بين المستويات
- يوجد نوعان من استقلالية البيانات و هما :

  - الاستقلال المنطقي (Logical Data Independence)
  - الاستقلال الفعلي (Physical Data Independence)

<u>الاستقلال المنطقي (Logical Data Independence):</u>

- هي المقدرة علي تغيير مخطط البيانات في المستوي الثاني (Conceptual Level) بدون الحاجة إلى تغيير المخطط في المستوي الثالث (External Level) وكذلك بدون تغيير البرامج التطبيقية
- يكون التغيير في المستوي الثاني لكي تستوعب قواعد البيانات التغيرات التي قد تحدث في المخطط نتيجة زيادة أو حذف عناصر بيانات
- التطبيقات التي تتعامل مع العناصر التي تغيرت هي فقط التي يتم تعديلها أما باقي التطبيقات فلا تتغير

<u>الاستقلال الفعلي (Physical Data Independence):</u>

- هي المقدرة علي تغيير مخطط البيانات في المستوي الأول (Internal Level) بدون الحاجة إلى تغير المخطط في المستوي الثاني (Conceptual Level)
- يكون التغيير في المستوي الأول (Internal Level) بسبب التغيرات التي قد تحدث نتيجة استخدام أساليب جديدة في تنظيم الملفات من أجل تحسين أداء النظام
- التطبيقات التي تتعامل مع العناصر التي تغيرت هي فقط التي يتم تعديلها أما باقي التطبيقات فلا تتغير

لغات نظم إدارة قواعد البيانات :

<u>لغة وصف البيانات (Data Definition Language DDL):</u>

- تستخدم بواسطة مدير قواعد البيانات (DBA) وكذلك مصمم قواعد البيانات لتعريف بناء قواعد البيانات
- يوجد مترجم للغة التعريف (DDL Compiler) وذلك لترجمة هذه اللغة و إنتاج برامج يتم تنفيذها لتقوم بإنشاء مخططات البيانات و تخزينها داخل فهرس قواعد البيانات (DB Catalog).

لغة تعريف الأشكال(View Definition Language VDL):

تستخدم في بعض نظم إدارة قواعد البيانات التي تستخدم هيكل قواعد البيانات الثلاثي بطريقة حقيقية وذلك لتعريف مخطط البيانات في المستوي الخارجي (External Level) النماذج Forms الرسومية GUI التفاعل من خلال القوائم menu

لغة التعامل مع البيانات (Data Manipulation Language DML):

تستخدم لاسترجاع و إدخال و حذف و تعديل البيانات.

- **ملاحظات على لغات قواعد البيانات :**

- نظم إدارة قواعد البيانات الحالية تستخدم لغة واحدة شاملة تحتوي علي لغات DDL,VDL,DML
- لغة الاستفسار الهيكلية (SQL) هي لغة تستخدم مع نموذج البيانات العلائقي و تحتوي علي لغات DDL,VDL,DML وكذلك الجمل الخاصة بتعديل مخطط البيانات

بيئة نظم قواعد البيانات :

نظم إدارة قواعد البيانات هي نظم معقدة و تحتوي على العديد من الوحدات التي تدعم ما يحتاجه المستخدم من وظائف ومنها:

مترجم لغة تعريف البيانات (DDL Compiler):

لترجمة تعريف مخطط البيانات و التأكد من صحته ثم تخزين هذا التعريف داخل فهرس النظام

منفذ قواعد البيانات (Run-Time DB processor):

يقوم بالتعامل مع قواعد البيانات عند تشغيل أي أمر خاص بقواعد البيانات

مترجم لغة الاستفسارات (Query Compiler):

يتعامل مع الاستفسارات عن طريق فهم الأوامر و ترجمتها ثم إرسالها إلى منفذ قواعد البيانات لتنفيذها

خدمات تقدمها نظم إدارة قواعد البيانات :

تقوم بعض نظم إدارة قواعد البيانات بتقديم خدمات إضافية تساعد المستخدم في إدارة نظم قواعد البيانات مثل :

تحميل البيانات (Loading):

وهي عبارة عن عملية تحويل البيانات الموجودة سابقا في النظم القديمة الى شكل ملائم للتصميم الجديد بدون الحاجة الى إعادة إدخالها يدويا و الذي يكون غير ممكن عمليا في كثير من الحالات. و يوجد بعض الأدوات المساعدة و التي تقوم بتحويل البيانات من الشكل القديم التي كانت عليه الى الشكل الجديد و الملائم لقواعد البيانات المصممة حديثا

النسخ الاحتياطية (Backup):

عملية إنشاء نسخ احتياطية للبيانات الموجودة بهدف تأمين البيانات من الأعطال التي قد تؤدي لضياعها

تنظيم الملفات (File reorganization):

عملية إعادة تنظيم الملفات علي أسطوانات التخزين بهدف تحسين أداء النظام

مراقبة الأداء (Performance monitoring):

تستخدم لمراقبة وتسجيل أداء قواعد البيانات وبذلك تقدم لمدير قواعد البيانات (DBA) الإحصائيات اللازمة لتحليل أداء النظام و دراسة كيفية تحسينه (بعض النظم تقدم أيضا حلول لرفع الأداء)

أدوات تدعم عمل مستخدم قواعد البيانات :

CASE tools (أدوات مساعدة هندسة النظم):

- تستخدم في مراحل تصميم قواعد البيانات و يوجد العديد من الأدوات التي تقوم بتنفيذ الكثير من المراحل التي يمر بها تصميم النظام.

- **أدوات تطوير النظم :**

تستخدم عند تطوير نظم قواعد البيانات سواء أكانت لتصميم قواعد البيانات أو واجهات التعامل مع المستخدم أو تعديل و إنشاء الاستفسارات علي البيانات وكذلك أثناء إنشاء البرامج التطبيقية.

- **برامج الاتصال عبر الشبكات:**

و تستخدم لتقديم إمكانية التعامل مع قواعد البيانات عبر الشبكات

تصنيف قواعد البيانات

معيار التصنيف	التصنيف
نموذج البيانات	• شبكي (Network) • هرمي (Hierarchical) • علائقي (Relational) • شيئي علائقي (Object Relational)
عدد المستخدمين	• مستخدم واحد (Single User) • متعدد المستخدمين (Multi-users)
عدد أماكن التشغيل	• مركزي (Centralized) • الخادم/العميل (Client-Server) • موزع (Distributed)

قواعد البيانات العلاقية
Relational Databases

مبادئ قواعد البيانات العلاقية :

- نظراً لقوة ((Relational Database Management System (RDMS أصبحت هي النوع الوحيد المستخدم حالياً، لما تقدمه من قوه و كفاءة و أدوات مساعدة للمبرمجين
- تعتمد قاعدة البيانات العلاقية في تصميمها على المفاهيم الطبيعية الموجودة في بيانات نموذج العالم المصغر الذي تمثله قاعدة البيانات.
- أساس قواعد البيانات العلاقية هو العلاقات الرابطة بين البيانات و التي تعتبر الجزء الأهم و الذي يمثل اغلب التعاملات مع قاعدة البيانات.

- **مثال : نموذج قاعدة بيانات بسيطة (قاعدة بيانات لمستشفى)**

Patient (مريض)

رقم المريض	الاسم	الجنس	رقم الغرفة	الطبيب
10	ناصر	1	100	احمد
20	نهى	2	200	سمير
30	عبدالله	1	100	احمد

Medicine (دواء)

رقم الدواء	اسم الدواء	المصنع
A1	بنادول	مصر
B2	اسبرين	الاردن
C2	انسولين	السعودية

Treated By (يعالج بواسطة)

رقم المريض	رقم الدواء	الكمية
10	A1	1
30	C2	3
20	A1	4

- نموذج قاعدة البيانات (Database Model) هو نموذج يبين لنا صورة كاملة لنظام المعلومات و الوظائف و القيود الموجودة داخل قاعدة البيانات و يركز على التكامل بين البيانات.
- لتعريف قاعدة البيانات يجب تحديد تركيب السجلات التي يمكن تخزينها في كل ملف و تحديد الأنواع المختلفة لعناصر البيانات.
- كل سجل يحتوي على بيانات تمثل مثلا رقم المريض و اسم المريض و الطبيب و الجنس و رقم الغرفة (المثال السابق).

- يجب أن نحدد نوع البيانات لكل عنصر بيانات داخل السجل مثل:-

- اسم المريض سلسلة حروف
- رقم المريض يكون رقم صحيح

- يجب أن يكون هنالك رابط بين السجلات المختلفة حيث نجد معلومات عن مريض محدد مثلاً احمد في ملف "مريض" و معلومات عن الدواء و من يعالجه في ملفي "دواء" و "يعالج بواسطة".

- الكثير من العلاقات في قواعد البيانات تربط انواع مختلفة من السجلات مع بعضها البعض و يطلق عليها العلاقات الرابطة "Relationships" .

- كذلك تتضمن معالجة قواعد البيانات الاستعلام و التعديل مثل :-

- استخرج أسماء المرضى الذين يعالجهم د. عبدالله.
- عدّل اسم المريض احمد إلى محمد.

مقدار تشاركية العلاقات – درجة العلاقة

- **ارتباط الجداول والعلاقات بعضها ببعض.**
- **درجة العلاقة إما:**

- واحد - واحد
- واحد – متعدد
- متعدد – واحد
- متعدد – متعدد

مسافر - تذكرة
طالب – كتب مستعارة
مؤلفون - كتاب
طلاب - نشاطات

مقدار تشاركية العلاقات – درجة العلاقة

- واحد – واحد :

جدول المسافر

رقم السجل المدني	الاسم
1	احمد
2	منى
3	سعيد

جدول التذكرة

رقم التذكرة	من	الى	رقم المقعد	رقم السجل المدني
ت1	عمان	الدمام	20	1
ت2	عمان	الدمام	35	2
ت3	عمان	الدمام	15	3
ت4	عمان	الدمام	10	4

- واحد – متعدد :

جدول الطالب

رقم الطالب	اسم الطالب
1	احمد
2	منى
3	سعيد

جدول الكتب في المكتبة

رقم الكتاب	عنوان الكتاب	تاريخ الاعارة	رقم الطالب
ب1	قواعد البيانات		1
ب2	الرياضيات		2
ب3	الحاسوب		1
ب4	التربية		3

IbtiHAlinO

- متعدد – متعدد :

جدول النشاطات

رقم النشاط	اسم النشاط	الرسوم
ن1	السباحة	100
ن2	الشطرنج	20
ن3	التنس	50

جدول الطلب

رقم الطلب	اسم الطلب
1	احمد
2	منى
3	سعيد

جدول الاشتراك بالأنشطة

رقم النشاط	رقم الطلب
ن1	1
ن1	2
ن2	1
ن3	2

درجة الجدول

- عدد الاعمدة التي يحتويها الجدول

جدول النشاطات

رقم النشاط	اسم النشاط	الرسوم
ن1	السباحة	100
ن2	الشطرنج	20
ن3	التنس	50

درجة الجدول 3

جدول الطلب

رقم الطلب	اسم الطلب
1	احمد
2	منى
3	سعيد

درجة الجدول 2

مفاهيم أساسية في قواعد البيانات العلاقية :

- **البيانات "Data" :** هي أي حدوث للبيانات التي تصف أي كائن
- **البيانات الوصفية "Metadata" :** هي البيانات التي تصف البيانات المخزنة وصفاً دقيقاً و يطلق عليها Data about data
- **الكينونة "Entity" :** هي وحدة معلومات تمثل فئة أو مجموعة من الأشياء أو الكائنات أو الأنشطة، هذه الوحدة لها مواصفات (خصائص) تصفها و تخصصها و تعبر عن مجموعة الكائنات التي تنتمي إليها، هذه المجموعة هي أمثلة أو حالات أو نماذج أو كائنات تتبع هذا الكيان.
- و في اغلب الأحيان يكون اسم الكيان اسماً مفرداً.

- **أمثلة على الكيان من الأمثلة السابقة : مريض ، دواء ، يعالج بـ .**

- **العلاقة الرابطة "Relationships":**

- هي العلاقة التي تربط بين الكيانات و تمثل رابطة العالم المصغر الذي تمثله قاعدة البيانات.
- تعبر العلاقات الرابطة عن الروابط بين البيانات في الواقع و تمثل في اغلب الأحوال بفعل مضارع أو فعلاً مبني للمجهول

- **أمثلة على العلاقات الرابطة**
  - **الكيان طالب و الكيان مدرس و مقرر دراسي يوجد بينهم عدة علاقات رابطه منها :**
    - الطالب يدرس مقرر درسي
    - المدرس يُدرس المقرر الدراسي.
    - المدرس يُدرس الطالب المقرر الدراسي .
    - المدرس يرشد الطالب إلى المقرر المناسب.
    - الطالب يُرشد بواسطة المدرس .

الخاصية أو الحقل "Attribute" : هي صفه تصف كيان معين و قيمتها هي احد مكونات سجلات البيانات مثل رقم الطالب و اسم الطالب في الجدول (العلاقة) طالب.

عنصر البيانات "Data Item": هو اقل وحدة بيانات مثل قيمه مخصصة مثلا رقم الطالب ١٠٠٠ ، اسم الطالب احمد حيث احمد و الرقم ١٠٠٠ هي عناصر بيانات.

عنصر بيانات مجمع "Data aggregate" : هو عنصر بيانات يتكون من عناصر بيانات بسيطة اصغر مثال اسم الطالب (محمد احمد عبدالله) حيث أن الاسم هنا مجمع من ثلاث بيانات اصغر هي الاسم الأول و اسم الأب و العائلة.و يمكن تقسيمها إلى ثلاث حقول مختلفة تمثل جميعها الاسم الكامل للطالب.

- **سجل "Record" :**

هو تجميع لعناصر بيانات تمثل احد أمثلة أو حالات كيان محدد .مثلا :
  - كل طالب له (اسم و رقم و تخصص)
  - وبالتالي مثال لسجل طالب:

(احمد ، ١٠٠٠ ، حاسوب)
(عبدالله ، ٢٥٠٠ ، علوم)

- كل قيمة من قيم السجل تمثل عناصر بيانات لخاصية من خواص الكيان.

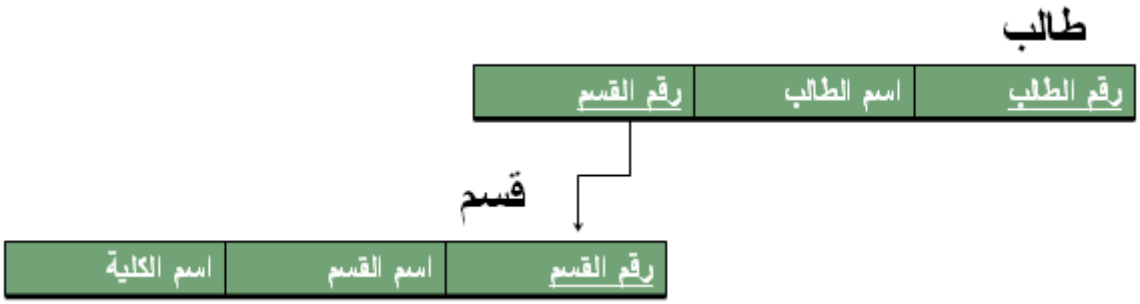
- **المفتاح "Key" :**

هو خاصية واحدة أو (عدة خصائص مجتمعه) من خصائص الكيان تستخدم لاختيار سجل أو أكثر من سجلات ذلك الكيان و يوجد منها ثلاث أنواع:-

المفتاح الرئيسى "Primary Key": المفتاح الأساسي لكيان هو احد خصائص هذا الكيان و قيمته تكون وحيدة في كل سجل و لا تتكرر (Unique) في أي سجل أخر من نفس الكيان ، و يجب كذلك أن تحتوي على قيمة و لا يجوز تركها فارغة مثل رقم الطالب في جدول طلاب ، حيث كل طالب يجب أن يكون له رقم مختلف عن زملائه ، و يوضع خط مستقيم أسفل الحقل للدلالة على انه مفتاح رئيسي.

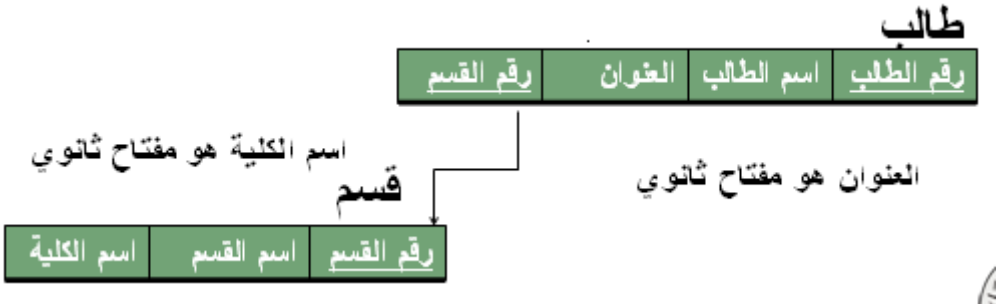
رقم الطالب	اسم الطالب	التخصص	الكلية

المفتاح الأجنبى "Foreign Key": هو عبارة عن خاصية عادية من ضمن خواص الكيان و موجودة كخاصية مفتاح أساسي في كيان آخر ، نميز هذا المفتاح بوضع خط متقطع أسفل اسم الخاصية.



- ليس بالضرورة أن يكون اسمي الحقلين متشابهين في الكيانين
- المفتاح الخارجي يستخدم للربط بين الكيانات لاستخراج بيانات ذات صلة بين كيانين مختلفين، مثل استخراج اسم القسم و الكلية التي يدرس فيها الطالب.

المفتاح الثانوي "Secondary Key": هو أي خاصية يمكن استخدامها لاختيار سجلات معينة من بين السجلات الموجودة في الكيان .



جدول المسافر

رقم السجل المدني	الاسم
1	احمد
2	منى
3	سعيد

جدول التذكرة

رقم التذكرة	من	الى	رقم المقعد	رقم السجل المدني
ت1	عمان	الدمام	20	1
ت2	عمان	الدمام	35	2
ت3	عمان	الدمام	15	3
ت4	عمان	الدمام	10	4

IbtiHAlinO

جدول الطالب

رقم الطالب	اسم الطالب
1	احمد
2	منى
3	سعيد

جدول النشاطات

رقم النشاط	اسم النشاط	الرسوم
ن1	السباحة	100
ن2	الشطرنج	20
ن3	التنس	50

جدول الاشتراك بالأنشطة

رقم النشاط	رقم الطالب
ن1	1
ن1	2
ن2	1
ن3	2

مخططات قواعد البيانات (Schemas) :

وصف قواعد البيانات يسمى "مخطط قواعد البيانات" (Schema)

-يستخدم المخطط عند تصميم قواعد البيانات
-هذا المخطط لا يتوقع تغيره بشكل تكراري
-يتم عادة تمثيل هذا المخطط باستخدام شكل أو رسم هندسي
-يوضح هذا المخطط بعض الأشياء مثل أسماء السجلات و أسماء الحقول وقد لا تظهر فيه نوع البيانات المستخدمة أو العلاقات بين البيانات
-يسمى هذا المخطط "Intension"
-هذا المخطط يتم تخزين وصفه داخل قواعد البيانات وهذا ما يعرف باسم "meta-data"

مخطط لبيانات جامعة (Schema)

- الطالب

الاسم	رقم الطالب	الفصل	التخصص

- المقرر

رقم المقرر	اسم المقرر	عدد الساعات	القسم

- المتطلب

رقم المقرر	رقم المتطلب السابق

- الشعبة

رقم الشعبة	رقم المقرر	الفصل	السنة	المحاضر

- كشف-الدرجات

رقم الطالب	رقم الشعبة	الدرجة

قواعد البيانات العلاقية
Relational Databases

مقدار تشاركية العلاقات – درجة العلاقة

متعدد- متعدد:

جدول الطالب

رقم الطالب	اسم الطالب
١	احمد
٢	هناء
٣	سعيد

جدول النشاطات

رقم النشاط	اسم النشاط	الرسوم
ن١	السباحة	١٠٠
ن٢	الشطرنج	٢٠
ن٣	التنس	٥٠

جدول الاشتراك بالأنشطة

رقم النشاط	رقم الطالب
ن١	١
ن١	٢
ن٢	١
ن٣	٢

مفاهيم أساسية في قواعد البيانات العلاقية :

-**البيانات "Data" :** هي اي حدوث للبيانات التي تصف اي كائن

-**البيانات الوصفيه "Metadata" :** هي البيانات التي تصف البيانات المخزنه وصفاً دقيقاً و يطلق عليها Data about data

-**الكينونه "Entity" :** هي وحدة معلومات تمثل فئة او مجموعة من الاشياء او الكائنات او الانشطة، هذه الوحدة لها مواصفات (خصائص) تصفها و تخصصها و تعبر عن مجموعة الكائنات التي تنتمي اليها ، هذه المجموعه هي امثلة او حالات او نماذج او كائنات تتبع هذا الكيان.
- و في اغلب الاحيان يكون اسم الكيان اسماً مفرداً.
-امثلة على الكيان من الامثلة السابقة : مريض ، دواء ، يعالج بـ .

-**العلاقة الرابطة"Relationships":** هي العلاقة التي تربط بين الكيانات و تمثل رابطة العالم المصغر الذي تمثله قاعدة البيانات.

-تعبر العلاقات الرابطة عن الروابط بين البيانات في الواقع و تمثل في اغلب الاحوال بفعل مضارع او فعلاً مبني للمجهول

امثلة على العلاقات الرابطة :

الكيان طالب و الكيان مدرس و مقرر دراسي يوجد بينهم عدة علاقات رابطه منها :
- الطالب يدرس مقرر درسي
- المدرس يُدرس المقرر الدراسي.
- المدرس يُدرس الطالب المقرر الدراسي .
- المدرس يرشد الطالب الى المقرر المناسب.
- الطالب يُرشد بواسطة المدرس .

الخاصية او الحقل "Attribute" : هي صفه تصف كيان معين و قيمتها هي احد مكونات سجلات البيانات مثل رقم الطالب و اسم الطالب في الجدول (العلاقة) طالب

عنصر البيانات "Data Item": هو اقل وحدة بيانات مثل قيمه مخصصة مثلا رقم الطالب ١٠٠٠ ، اسم الطالب احمد حيث احمد و الرقم ١٠٠٠ هي عناصر بيانات.

عنصر بيانات مجمع "Data aggregate" : هو عنصر بيانات يتكون من عناصر بيانات بسيطة اصغر مثال اسم الطالب (محمد احمد عبدالله) حيث ان الاسم هنا مجمع من ثلاث بيانات اصغر هي الاسم الاول و اسم الاب و العائله.و يمكن تقسيمها الى ثلاث حقول مختلفة تمثل جميها الاسم الكامل للطالب.

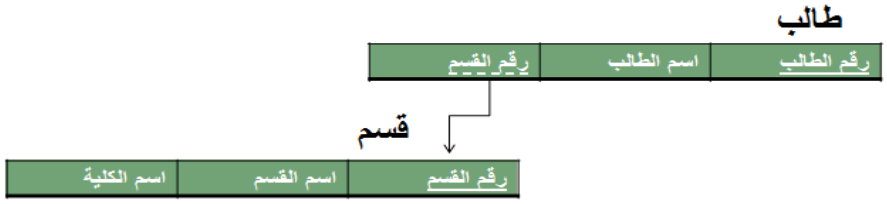
سجل "Record" : هو تجميع لعناصر بيانات تمثل احد امثلة او حالات كيان محدد .مثلا :
كل طالب له (اسم و رقم و تخصص)

و بالتالي مثال لسجل طالب :
(احمد ، ١٠٠٠ ، حاسوب)
(عبدالله ، ٢٥٠٠ ، علوم)

- كل قيمة من قيم السجل تمثل عناصر بيانات لخاصية من خواص الكيان.

• رقم الطالب	• اسم الطالب	• التخصص	• الكلية

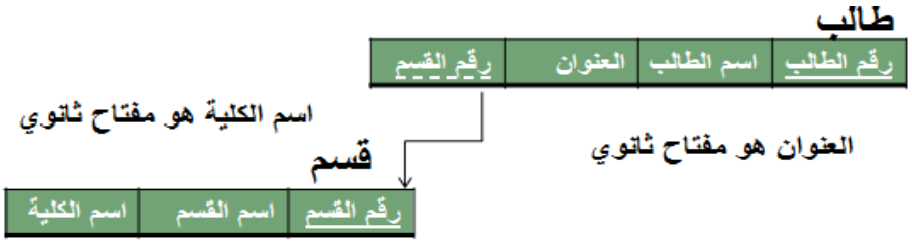
- **المفتاح الاجنبي "Foreign Key":** هو عباره عن خاصية عادية من ضمن خواص الكيان و موجودة كخاصية مفتاح اساسي في كيان آخر ، نميز هذا المفتاح بوضع خط متقطع اسفل اسم الخاصية.



-ليس بالضرورة ان يكون اسمي الحقلين متشابهين في الكيانين

-المفتاح الخارجي يستخدم للربط بين الكيانات لاستخراج بيانات ذات صلة بين كيانين مختلفين، مثل استخراج اسم القسم و الكلية التي يدرس فيها الطالب.

المفتاح الثانوي "Secondary Key": هو اي خاصية يمكن استخدامها لاختيار سجلات معينة من بين السجلات الموجودة في الكيان .



مبادئ قواعد البيانات العلاقية :

- مثال : نموذج قاعدة بيانات بسيطة (قاعدة بيانات لمستشفى)

Patient (مريض)

رقم المريض	الاسم	الجنس	رقم الغرفة	الطبيب
١٠	ناصر	١	١٠٠	احمد
٢٠	نهى	٢	٢٠٠	سمير
٣٠	عبدالله	١	١٠٠	احمد

Medicine (دواء)

رقم الدواء	اسم الدواء	المصنع
A1	بنادول	مصر
B2	اسبرين	الاردن
C2	انسولين	السعودية

Treated_By (يعالج بواسطة)

رقم المريض	رقم الدواء	الكمية
10	A1	١
30	C2	3
20	A1	4

نماذج البيانات (Data Models) :

- هو وصف للبيانات عن شاء او انشطة او احداث في مكان ما لجعل البيانات منظمة و مفهومة.
- بناء البيانات هو تحديد نوع البيانات و العلاقات بين البيانات و القيود المفروضة عليها
- يمكن أن يحتوي نموذج البيانات علي بعض العمليات الأساسية (مثل كيفية تعديل أو استرجاع البيانات)
- في نظم البيانات الشيئية يمكن أن يحتوي النموذج علي مجموعة من العمليات التي يعرفها المستخدم علي البيانات

مخططات قواعد البيانات (Schemas) :

وصف قواعد البيانات يسمي " مخطط قواعد البيانات " (Schema) :

- يستخدم المخطط عند تصميم قواعد البيانات
- هذا المخطط لا يتوقع تغيره بشكل تكرارى
- يتم عادة تمثيل هذا المخطط باستخدام شكل أو رسم هندسي
- يوضح هذا المخطط بعض الأشياء مثل أسماء السجلات و أسماء الحقول وقد لا تظهر فيه نوع البيانات المستخدمة أو العلاقات بين البيانات
- يسمى هذا المخطط "Intension"
- هذا المخطط يتم تخزين وصفه داخل قواعد البيانات وهذا ما يعرف باسم "meta-data"

مخطط لبيانات جامعة (Schema) :

- الطالب

الإسم	رقم الطالب	الفصل	التخصص

- المقرر

رقم المقرر	اسم المقرر	عدد الساعات	القسم

- المتطلب

رقم المقرر	رقم المتطلب السابق

- الشعبة

رقم الشعبة	رقم المقرر	الفصل	السنة	المحاضر

- كشف-الدرجات

رقم الطالب	رقم الشعبة	الدرجة

حالات قواعد البيانات (Instances) :

البيانات المتواجدة داخل قواعد البيانات في لحظة معينة تسمي "حالة قواعد البيانات أو الوضع الحالي لقواعد البيانات (DB State or Current Set of Occurrence or Instance)

- يتم إنشاء الوضع الابتدائى لها عند إدخال البيانات لأول مرة ثم يتغير وضعها عند إجراء العمليات المختلفة على البيانات (إضافة – حذف – تعديل)
- تسمى حالة البيانات هذه "Extension"

الروابط في قواعد البيانات العلاقية :

-قواعد البيانات العلاقية تركز بشكل اساسي على الروابط بين عناصر البيانات او بين الكيانات او سجلات البيانات.
-اهم اسباب نجاح قواعد البيانات العلاقية هي تمثيلها للروابط المختلفة التي توفر امكانيات استعلام سهلة و قوية.

رموز الرسم البياني في قواعد البيانات العلاقية :

المفهوم	الرمز
الكيان طالب	طالب
العلاقة الرابطة يدرس	يدرس
الخاصية اسم الطالب	اسم الطالب
المفتاح الاساسي رقم الطالب (خاصية)	رقم الطالب

نسبة المشاركة في العلاقة الرابطة :

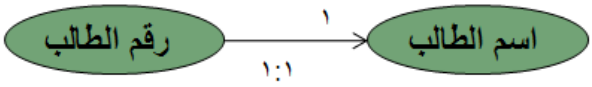
- اي رابطة بين عناصر البيانات هي بالاساس تربط عنصر بيانات معين اما بعنصر بيانات معين او بعدة عناصر بيانات.
- العلاقة الرابطة بين الكيانات (او السجلات) هي بالاساس تربط كيان بكيانات اخرى.
- (Cardinality) هو مفهوم يتحكم في الروابط و يعبر عن نسبة المشاركة العلاقة او الرابطة بين عنصر و اخر او كيان و اخر.
- تعني نسبة المشاركة عدد العناصر او السجلات المشاركة في العلاقة الرابطة.

تحدد الكارديناليتي مفهومين مرافقين للعلاقة الرابطة :-

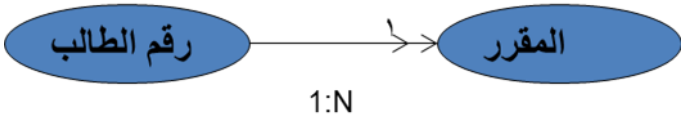
اختياري : اي انه يمكن ان تكون المشاركة صفر او اكثر.
اجباري: اي انه لابد ان تكون هنالك المشاركة بعنصر واحد على الاقل او اكثر.

انواع الروابط بين عناصر البيانات :

رابطة واحدة One Association : رابطة بين عنصرين تعني ان كل عنصر بيانات من خاصية ما يقابلها عنصر بيانات واحد من العنصر الثاني (كل رقم طالب يقابله اسم طالب واحد)



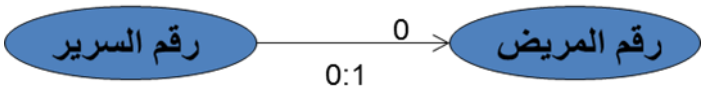
رابطة متعددة Many Association : رابطة بين عنصرين تعني ان كل عنصر بيانات من خاصية ما يقابلها عناصر بيانات متعددة من العنصر الثاني (كل رقم طالب يقابله اكثر من مقرر مادة)



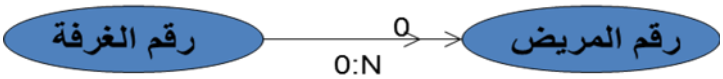
رابطة كاردينالتي (Cardinal Association) :

نوع الرابطة هنا يتداخل مع الرابطة الواحدة و الرابطة المتعددة

مع الرابطة الواحدة تحدد نسبة 0:1 اي من صفر الى واحد مثل الرابط بين رقم السرير و رقم المريض



مع الرابطة المتعددة تحدد نسبة 0:N اي من صفر الى واحد مثل الرابط بين رقم السرير و رقم المريض



طبعا ممكن ان تكون النسبة ١ بدل صفر في جميع الامثلة اعلاه

مبادئ قواعد البيانات العلاقية
Fundamentals of Relational Databases

تصميم نموذج الكيان و العلاقة الرابطة
Design of Entity-Relationship Diagram (ERD)

انواع العلاقات الرابطة بين الكيانات (السجلات) :

العلاقات الرابطة بين الكيانات هي اهم ما يميز قاعدة البيانات العلاقية ، حيث تتوقف قاعدة البيانات التي نصممها و ننفذها بشكل كبير على انواع العلاقات الرابطة

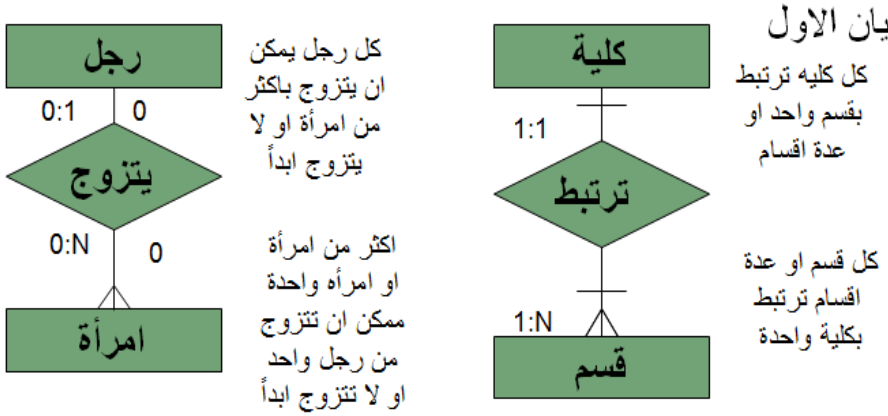
تقسم الى ثلاث انواع :-

علاقة سجل واحد مع سجل واحد
علاقة سجل واحد مع عدة سجلات
علاقة عدة سجلات مع عدة سجلات

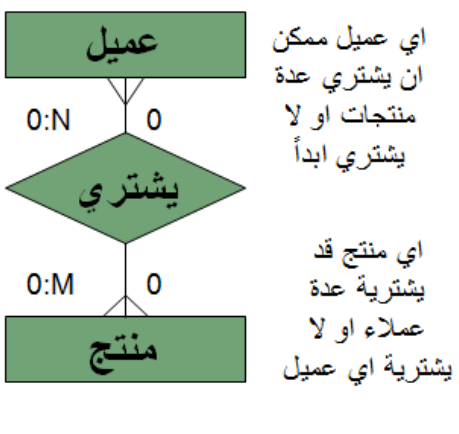
● **علاقة سجل واحد مع سجل واحد(One to one)**:

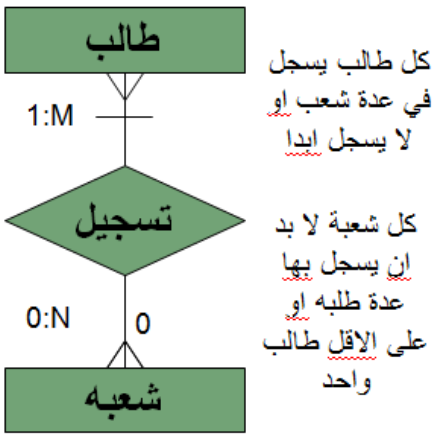
تعني هذه العلاقة ان اي سجل يرتبط مع سجل واحد على الاكثر من الكيان الثاني

● **علاقة سجل واحد مع عدة سجلات(one to many)**: تعني هذه العلاقة ان اي سجل من الكيان الاول يمكن ان يرتبط مع عدة سجلات من الكيان الثاني ، و يرتبط اي سجل من الكيان الثاني مع سجل واحد على الاكثر من الكيان الاول



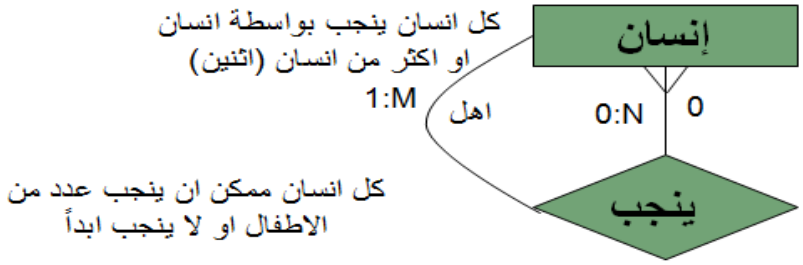
● **علاقة عدة سجلات مع عدة سجلات (Many to many)**: اي سجل من الكيان الاول يرتبط مع عدة سجلات من الكيان الثاني و كذلك يرتبط اي سجل من الكيان الثاني مع عدة سجلات من الكيان الاول.



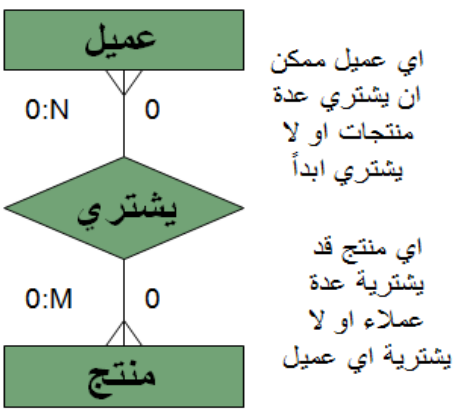


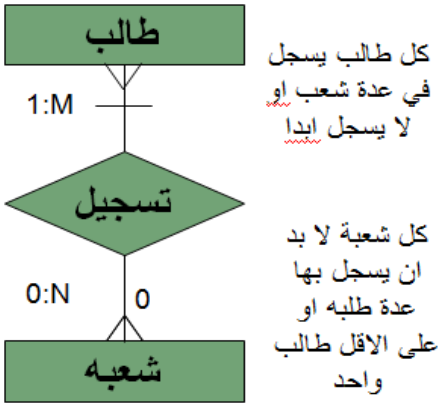
درجة العلاقة الرابطة بين السجلات (الكيانات)
degree of association between Entities

تحدد درجة العلاقة الرابطة عدد الكيانات المشاركة في العلاقة الرابطة او بمعنى اخر عدد الكيانات التي تربط بينهما العلاقة.
علاقة احادية (Unary Relationship): تكون درجة العلاقة الرابطة احادية اذا كانت العلاقة الرابطة بين الكيان و نفسه ، اي تربط سجلات من نفس الكيان

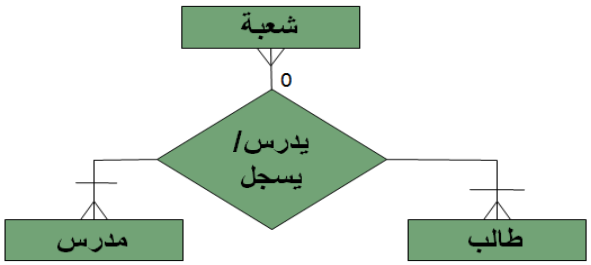


● **علاقة ثنائية (Binary Relationship)**:تكون درجة العلاقة الرابطة ثنائية اذا كانت العلاقة الرابطة بين كيانين مختلفين.

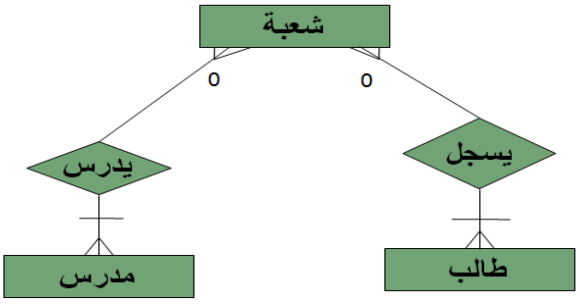




علاقة ثلاثية (Ternary Relationship): تكون درجة العلاقة الرابطة ثلاثية اذا كانت العلاقة الرابطة بين ثلاث كيانات مختلفة. في اغلب قواعد البيانات تحول الرابطة الثلاثية الى ثنائية.



- علاقة ثلاثية (Ternary Relationship) و التحويل الى علاقة ثنائية



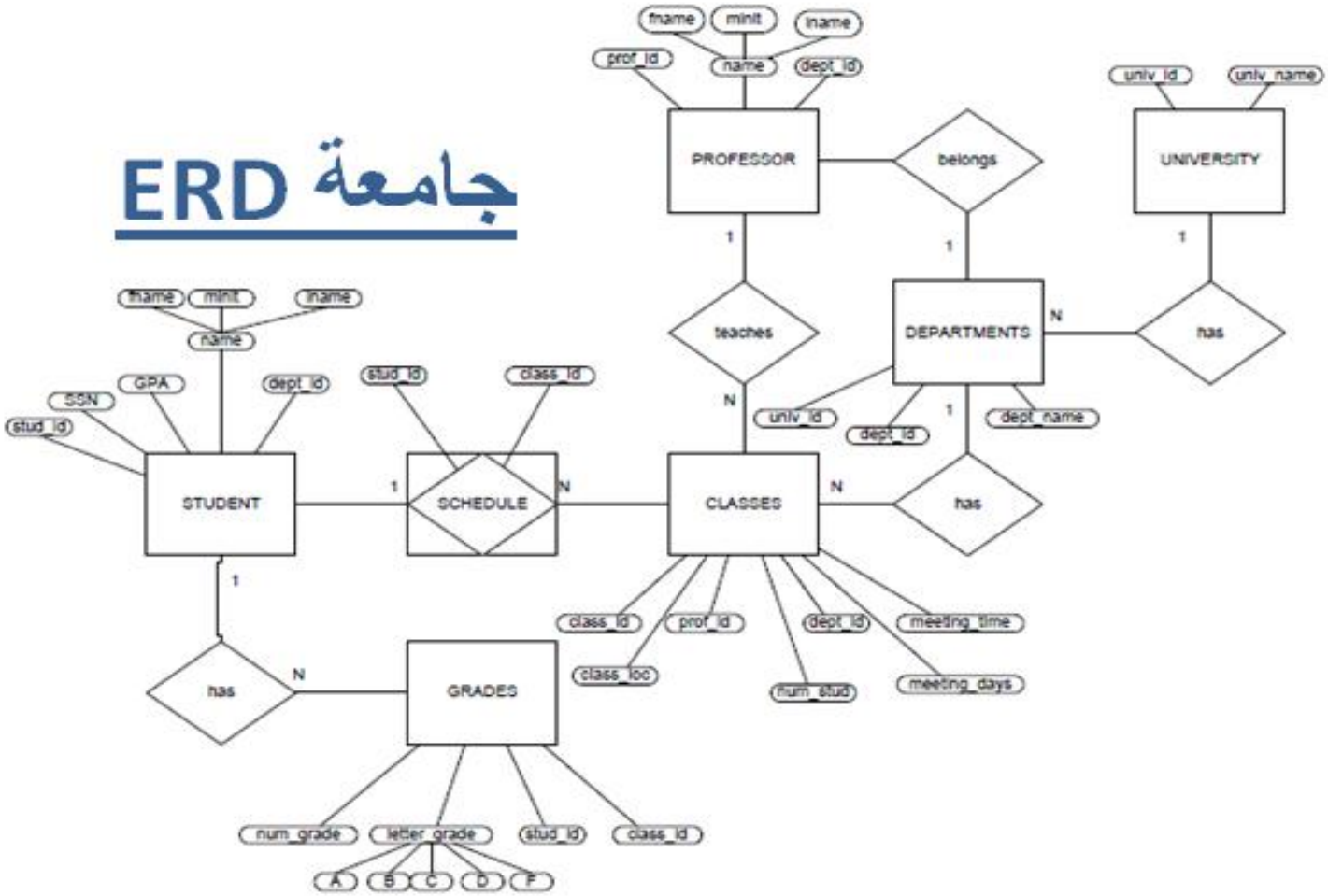
النموذج العلاقي (Relational Model) :

- لبناء قواعد بيانات يفضل مبرمجو و مصممو قواعد البيانات استخدام نموذج البيانات العلاقي . هذا النموذج مبني على العلاقات بين الكيانات و البيانات.
- اغلب البرامج و النظم المستخدمة في مجال قواعد البيانات صممت لبناء هذا النوع.
- مميزات النموذج العلاقي : يتميز النموذج العلاقي عن غيره من نظم قواعد البيانات للاسباب التالية.
- له الادوات و الخصائص التي تميزه عن غيره و الخاصه به.
- يمثل منطقياً كافة الكيانات و العلاقات و خصائصها.
- يعد تطويراً و امتداداً لشكل ملف البيانات التقليدي.
- يستخدم المفاتيح الاساسية و الأجنبية للربط بين الكيانات.
- اعتماداً عليه بنيت خصائص قواعد البيانات التي تجعلها لا تتأثر بمشاكل الصيانه.
- يمكن تطبيق كافة العمليات الحسابية و المنطقية على مكوناته.

نموذج الكيان و العلاقة الرابطة (Entity Relationship Model) :

- احد اشهر و اهم طرق تمثيل و تصميم قواعد البيانات هو نموذج الكيان و العلاقة الرابطة (Entity-Relationship Data Model (ERD)).

- نموذج (ERD) هو النموذج الذي يتم استخدامه لانشاء قواعد البيانات على الحاسب الآلي و له قواعد و اشكال محددة تصف الكيانات الموجودة في تطبيق معين و العلاقات الرابطة بين تلك الكيانات و خصائصها و كذلك القيود المفروضة على كل منها.
- يمثل تصميم قاعدة البيانات.
- جميع الاشكال التي تم دراستها هي نماذج مبسطة من نماذج (ERD)



أسلوب تصميم نموذج الكيان و العلاقة الرابطة :

يتم تصميم قواعد البيانات باستخدام إحدى طريقتين :

١- استخدام الرسم البياني للكيان و العلاقة الرابطة (Entity Relationship diagram)
٢- تطبيع قواعد البيانات (Database Normalization)

- يتكون تصميم ERD من مجموعة من الكيانات (Entity) تربط بعضها ببعض علاقات رابطة (Relationship).
- السجلات التي تتبع الكيانات عبارة عن بيانات شبه ثابتة ،ونادرا ما تحتاج الى التعديل (Static Data)
- يتم تحديد خصائص كل كيان
- الخاصية التي تميز كل سجل يتبع الكيان ولا تتكرر هي خاصية المفتاح الرئيسي(Primary Key)
- السجلات التي تصف العلاقات الرابطة فهي عبارة عن بيانات تتجدد و تتغير و تضاف و تحذف بشكل متواصل (Dynamic Data).
- يجب تحديد لكل علاقة رابطة الخصائص التي تساعد على وصف العلاقة الرابطة بين كل كيانين
- يجب تحديد نوع العلاقة :

- واحد إلى واحد(One-to-One) - واحد إلى كثير(One-to-Many) - كثير إلى كثير(Many-to-Many)

- يجب تحديد نسبة المشاركة (٠أو١)

تطبيق قاعدة بيانات الكلية المصغر :

<u>في قاعدة بيانات الكلية نهتم بـ :</u>

-تخزين بيانات الطلبة (الرقم الجامعي، الاسم، العنوان، التخصص)
-بيانات المقررات التي يسجلها الطالب (رمزالمقرر، اسم المقرر ، عدد الساعات المعتمدة،العام الدراسي، الفصل الدراسي، العلامة)
-بيانات أعضاء هيئة التدريس(الرقم،الاسم،الهاتف، القسم ، المقررات)

التصميم :

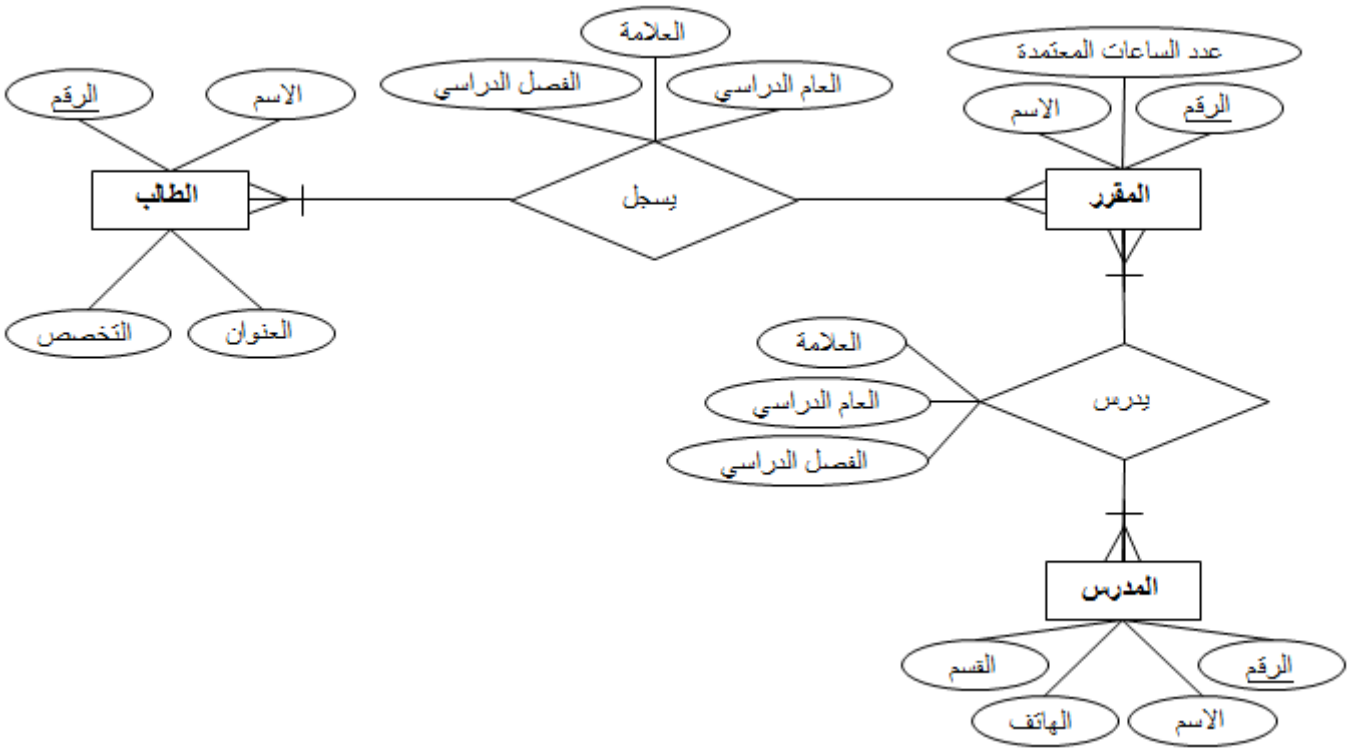
١- الكيانات : هي وحدة معلومات لها خصائص تصفها تخصها و انها تكون أسماء. وقد تم تحديد الكيانات التالية :

- الطالب ، و خصائصه هي: (رقم الطالب ، الاسم ، العنوان ، التخصص)
- المدرس ، و خصائصه هي: (رقم المدرس ، الاسم ، الهاتف ، القسم)
- المقرر ، و خصائصه هي: (رمز المقرر، اسم المقرر ، عدد الساعات المعتمدة)
- ملاحظة (١) : المعلومات المذكورة هي معلومات شبه ثابتة (Static)
- ملاحظة (٢) : لم يتم ذكر معلومات عن المواد التي يدرسها الطالب في جدول الطالب، ولم يتم ذكر معلومات عن المواد التي يدرسها المدرس في جدول المدرس ، ولم يتم ذكر معلومات الفصل الدراسي في جدول المقرر ، فهي كلها معلومات متغيرة (Dynamic)لا تذكر في الكيانات.
- ملاحظة (٣) : تم تحديد الصفة المميزة لكل كيان بوضع خط تحتها.

٢- العلاقات الرابطة : هي عبارة عن فعل يمثل العلاقة بين كيان ونفسه أو كيانين أو ثلاثة كيانات معا. وقد تم تحديد العلاقات التالية :

- يُسجِّل :
  - هي علاقة تربط الطالب بالمقررات التي يسجلها للدراسة
  - لها الخصائص (الفصل الدراسي ، العام الدراسي ، العلامة)
  - كل طالب يمكنه تسجيل عدة مقررات ، وكل مقرر يمكن ان يسجله اكثر من طالب ، أي ان نوع العلاقة كثير إلى كثير(M:N(Many –to – Many
- يُدرِّس :
  - هي علاقة تربط عضو هيئة التدريس بالمقرر الدراسي
  - لها الخصائص(الفصل الدراسي ، العام الدراسي)
  - كل مدرس يمكنه تدريس عدة مقررات،وكل مقرر يمكن ان يدرسه اكثر من مدرس ، أي ان نوع العلاقة كثير إلى كثير(M:N(Many –to – Many

٣- نموذج الكيان العلاقة :



تطبيق قاعدة بيانات المستشفى المصغر :

١)الكيانات :

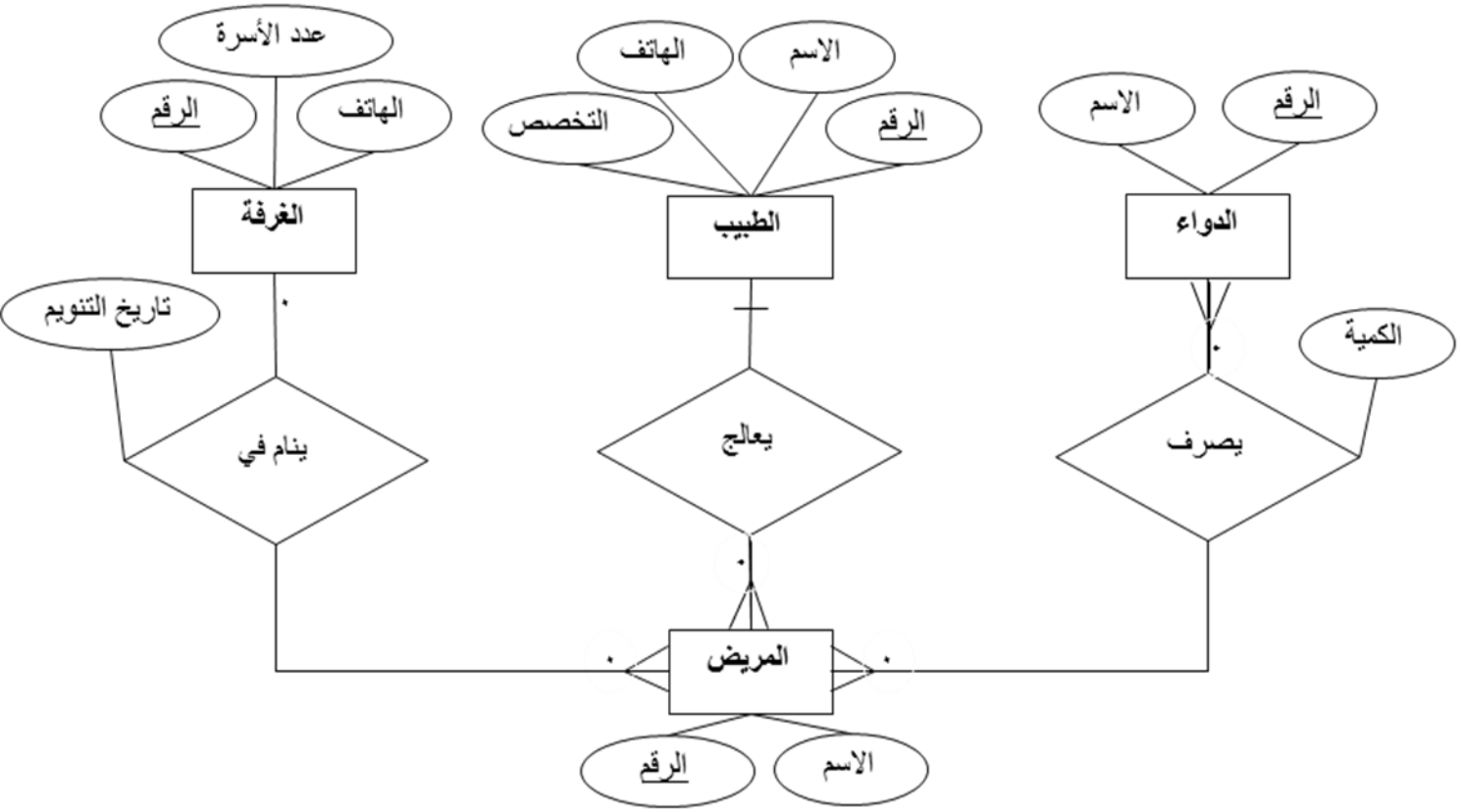
- المريض (الرقم، الاسم)
- الدواء (الرقم ، الاسم)
- الطبيب (الرقم ، الاسم ، الهاتف ، التخصص)
- الغرفة (الرقم ، الهاتف ، عدد الأسرة)

٢) العلاقات الرابطة :

- يعالج :
  - علاقة رابطة بين الطبيب و المريض
  - المريض يتبع دكتور واحد ، اما الدكتور فيتبعه عدة مرضى و يمكن الا يتبعه أي مريض و بالتالي تكون الكارديناليتي (١:١) من جهة الطبيب و تكون الكارديناليتي (0:N) من جهة المريض ، وعليه يكون نوع العلاقة واحد إلى كثير(1:N)
- ينام في :
  - علاقة رابطة بين المريض والغرفة
  - كل غرفة يقيم فيها اكثر من مريض، أو لا يقيم فيها احد، لذلك فالكارديناليتي من جهة المريض هي (0:N)
  - كل مريض يمكن ان ينام في غرفة واحدة، او ليتم تنويمه في حالة العيادة الخارجية، لذلك فالكارديناليتي من جهة الغرفة هي (0:1)
  - نوع العلاقة واحد إلى كثير (1:N)
  - لها علاقة (تاريخ التنويم)
- يصرف :
  - علاقة رابطة بين المريض والدواء
  - المريض يمكن ان يصرف اكثر من دواء او لا يصرف دواءا ، وبالتالي الكارديناليتي تكون (0:N)من ناحية الدواء
  - كل دواء يمكن صرفه من اكثر من مريض، ويمكن ان لايصرفه أي مريض ، وبالتالي الكارديناليتي تكون (0:N)من ناحية المريض
  - و عليه يكون نوع العلاقة كثير إلى كثير(N:M)

هذه العلاقة لها الخاصية (الكمية)

٣) نموذج الكيان العلاقة :



تطبيق الإستعارة الإلكترونية :

١) الكيانات :
- المستعير (الرقم ، الاسم ، العمل ، جهة العمل ، تاريخ الميلاد)
- الكتاب (الرقم ، عنوان الكتاب ، عدد الصفحات ، دار النشر ، سنة النشر)
- المؤلف (اسم المؤلف ، العنوان الالكتروني)

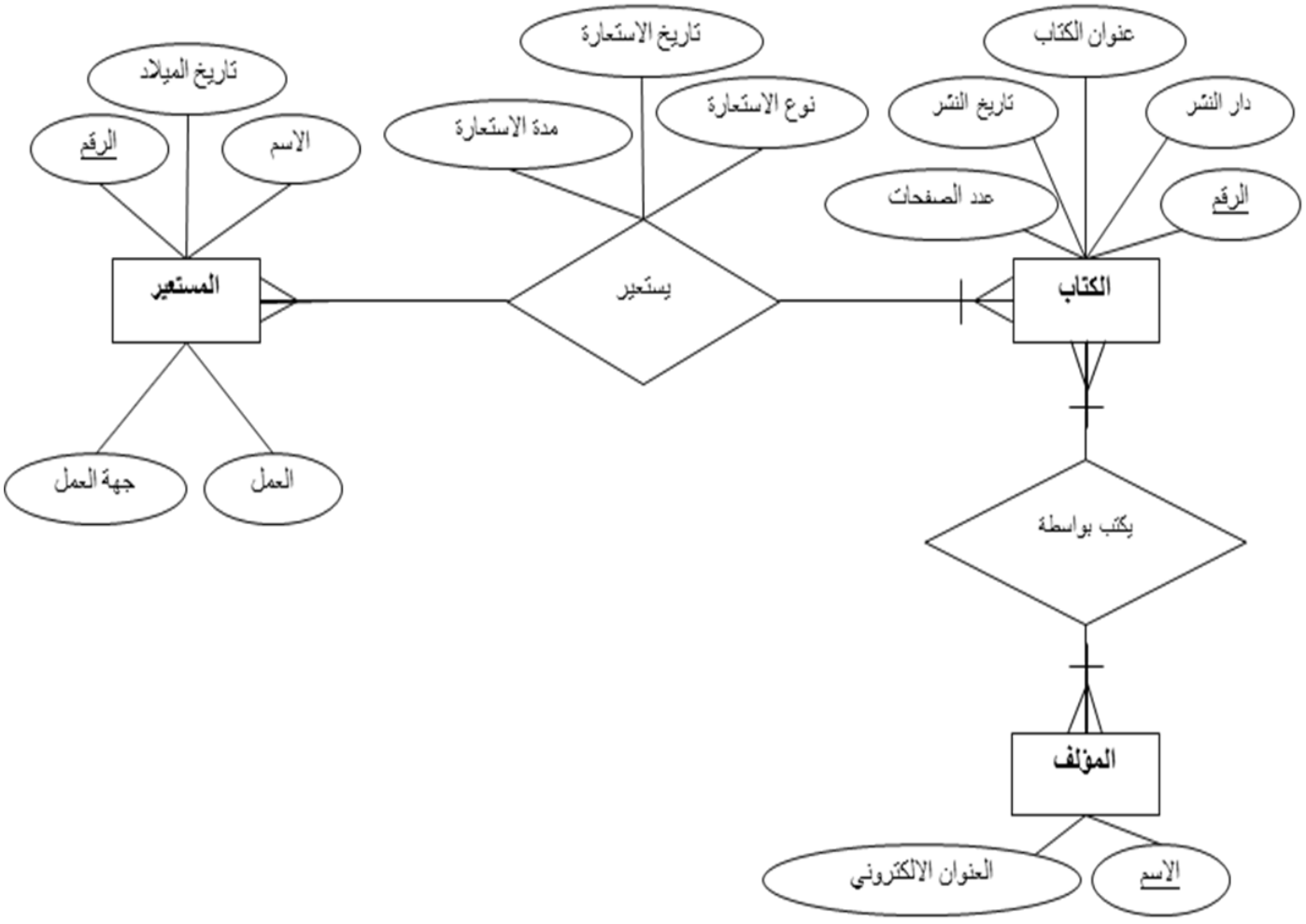
٢) العلاقات الرابطة :
- يستعير:

□ علاقة رابطة بين المستعير و الكتاب
□ لها الخصائص (نوع الاستعارة ، مدة الاستعارة ، تاريخ الاستعارة)
□ يمكن لأي مستعير أن يستعير عدة كتب ، وعلى الاقل كتاب واحد ، لذلك الكارديناليتي (1:N)من ناحية الكتاب
□ يمكن ان يستعير الكتاب عدة مستعيرين ، أولا احد ، لذلك تكون الكاردينالتي (0:N)من ناحية المستعير.
□ هذه العلاقة من النوع كثير إلى كثير (N:M)

- كُتِبَ بواسطة :

□ علاقة رابطة بين المؤلف و الكتاب
□ يمكن لأي مؤلف أن يكتب عدة كتب ، وعلى الاقل كتاب واحد ، لذلك الكارديناليتي (1:N)من ناحية الكتاب
□ يمكن ان يشترك في تاليف اي كتاب عدة مؤلفين ، أو على الاقل مؤلف واحد ، لذلك تكون الكارديناليتي (1:N) من ناحية المؤلف.
□ هذه العلاقة من النوع كثير إلى كثير (N:M)

٣) نموذج الكيان العلاقة :



تصميم نموذج الكيان والعلاقة الرابطة - تابع
Design of Entity-Relationship Diagram (ERD)

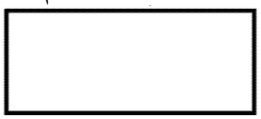
ما هو مخطط الكيان العلاقة ؟

هو نموذج عالي المستوي يقوم بعرض بناء البيانات ، و يتم استخدام هذا النموذج أثناء مرحلة التصميم المفاهيمي للنموذج الأولى ، و ينتج عن ذلك النموذج الأولي ، لقاعدة البيانات و الذي عن طريقه نقوم بتصميم مخطط قاعدة البيانات ، ويتم تمثيل بناء البيانات و القيود المطلوبة عليها باستخدام أشكال رسومية سهلة ومحددة.

مكونات مخطط الكيان العلاقة :

الكيان أو الكينونة(Entity) : هو الوحدة الأساسية التي يتم تمثيلها بنموذج الكينونة / العلاقة و يشير هذا الكيان إلي "شيء" حقيقي في الحياة سواء كان لهو وجود فعلي مثل (طالب – موظف – سيارة - ... الخ) أو وجود منطقي مثل (شركة – وظيفة – مقرر -... الخ) .

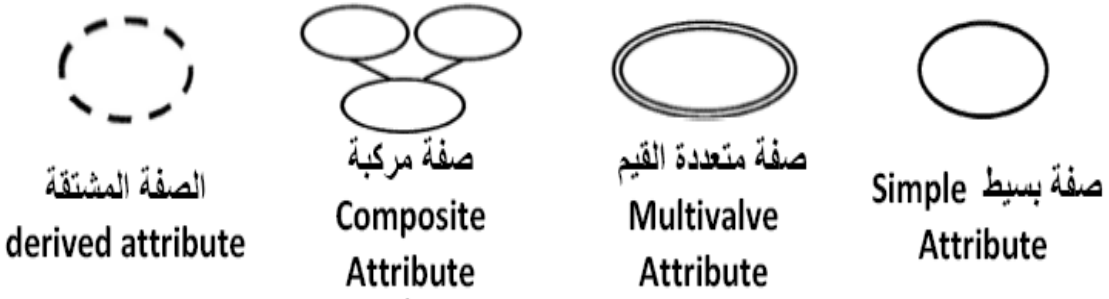
- و يتم تمثيل الكيان باستخدام شكل مستطيل يكتب داخلها اسم الكيان أو الكينونة



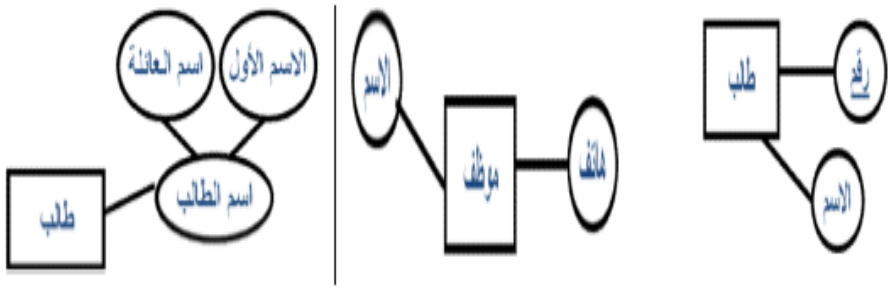
الصفة (Attribute):هي صفة معينة تصف الكيان و تكون تابعة لهو في المخطط ، مثل اسم الموظف ، عمر الطالب ، مرتب موظف ، درجة طالب ، عدد الساعات الدراسية لمقرر ، لاحظ أن كل صفة تتبع كيانا معينا.

- الصفة يمكن أن تكون صفة بسيطة – أو صفة مركبة
- الصفة يمكن أن تكون أحادية القيمة – أو صفة متعددة القيم
- يوضع خط تحت اسم الصفة أو الصفات التي تمثل مفتاح الكيان

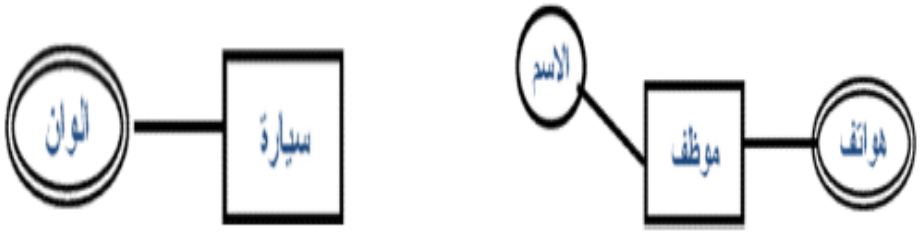
- **الصفة المشتقة :** وهي صفة بسيطة يمكن الحصول عليها من صفة بسيطة أخرى ، مثل الصفة عمر الطالب التي يمكن الحصول عليها من تأريخ الميلاد
- يتم تمثيل الصفة باستخدام شكل بيضاوي و الصفة متعددة القيم يتم تمثيلها بالشكل البيضاوي المزدوج



شكل تمثيل الصفات المختلفة للكيان

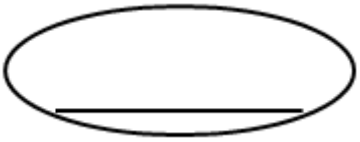


الصفة البسيطة مثل اسم طالب، هاتف موظف. الصفة المركبة مثل اسم الطالب(الاسم الأول – اسم العائلة)

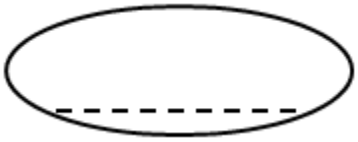


الصفة متعددة القيم مثل ألوان و هواتف، فالكيان موظف (قد يكون له هاتف أو اثنين أو أكثر)
والكيان سيارة (قد تكون من لون واحد أو أي عدد من الألوان)
شكل مثال توضيحي

صفة المفتاح الرئيسي (Primary Key Attribute) : هي تلك الصفة المميزة للكيان ، و يتم تمثيلها بشكل بيضاوي مع خط تحت إسم الصفة ، وقد يكون للكيان أكثر من صفة لتمثل معا المفتاح الرئيسي

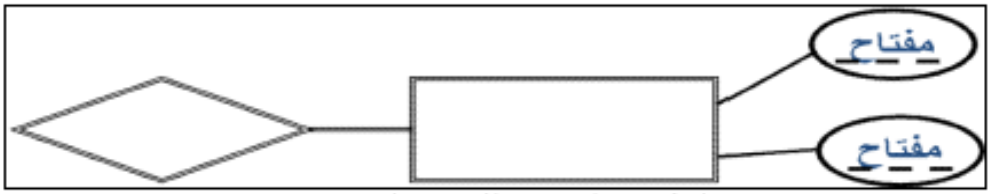


صفة المفتاح الجزئي (Partial Key Attribute) : هي تلك الصفة التي لم ترقى لتكون مميزة للكيان ،ولكنها صفة قد تساعد في تكوين صفة مميزة إذا تم ضمها إلى صفة مميزة من كيان آخر، ويتم تمثيلها بشكل بيضاوي مع خط متقطع تحت إسم الصفة



الكيان الضعيف(Weak Entity) : هو ذلك الكيان الذي ليس لديه مفتاح رئيسي يميز بياناته عن بعضها البعض ، وعادة ما يقترن الكيان الضعيف بكيان قوي عن طريق علاقة تعريف تقويه ، و يتم تمثيل الكيان الضعيف بمستطيل مزدوج

- يرتبط المفتاح الجزئي بالكيان الضعيف ، ليساعد فيما بعد بتكوين مفتاح رئيسي للكيان الضعيف.

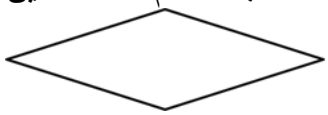


المفتاح الجزئي للكيان الضعيف

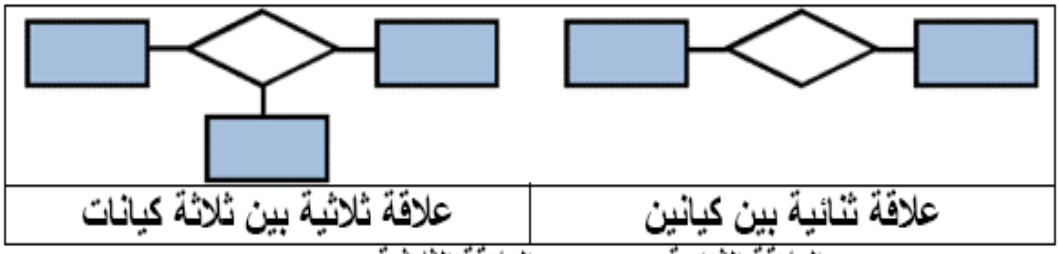
- يرتبط المفتاح الجزئي بالكيان الضعيف، ليساعد فيما بعد بتكوين مفتاح رئيسي للكيان الضعيف.

العلاقات (Relation): العلاقة (R) بين مجموعة من الكيانات (E1,E2, ...En) هي مجموعة تمثل الارتباطات بين هذه الكيانات ، كل وحدة في العلاقة (R) هي عبارة عن اتحاد أو ارتباط بين الكيانات المرتبطة بهذه العلاقة ، بحيث أن هذه الوحدة تمثل بصف واحد من كل كيان مشارك في العلاقة.

- في نموذج الكيان / العلاقة ، يجب أن يتم تمثيل المرجعية من كيان إلى كيان آخر ، باستخدام "علاقة" وليس كصفة في الكيان
- يتم تمثيل العلاقة في نموذج الكينونة/العلاقة باستخدام شكل المعين



- درجة العلاقة:لكل علاقة درجة، وتتحدد هذه الدرجة بعدد الكيانات المرتبطة بهذه العلاقة (ثنائية – ثلاثية - ....)



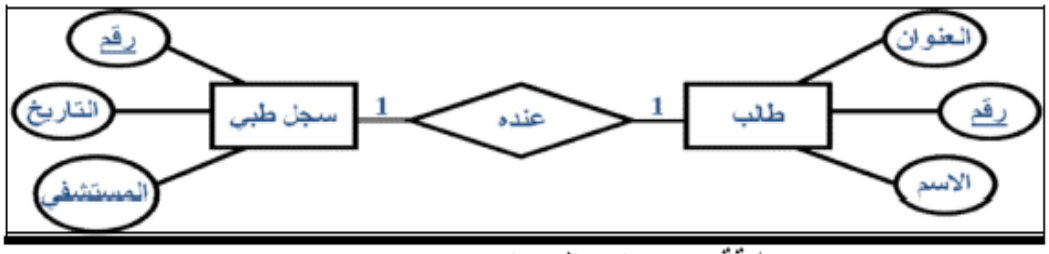
العلاقة الثنائية Binary والعلاقة الثلاثية Ternary

نوع العلاقة (Cardinality Ratio): المصطلح يعني نسبة الارتباط بين وحدات الكيان ، التي ترتبط بنفس العلاقة ، وفي العلاقة الثنائية بين كيانين ، نوع العلاقة هو عدد الوحدات (Instances) في العلاقة التي يمكن أن يشترك فيها الكيان وهي ثلاثة أنواع :

١. **علاقة واحد - الى - واحد (one-to-one):** وفيها ترتبط وحدة واحدة من الكيان الأول بوحدة واحدة من الكيان الآخر على الأكثر، و يرمز لها بالرمز1:1

٢. **علاقة واحد - الى- كثير(one-to-many) :** وفيها يمكن أن ترتبط وحدة واحدة من أحد الكيانات بأكثر من وحدة في الكيان الآخر، و العكس غير صحيح ، و يرمز لها بالرمز 1:N

٣. **علاقة كثير - الى- كثير(many-to-many) :** و فيها يمكن أن ترتبط أكثر من وحدة من الكيان الأول بأكثر من وحدة في الكيان الآخر، و العكس، أي يمكن لأي وحدة في الكيان الآخر أن ترتبط بأي وحدة في الكيان الأول ، و يرمز لها بالرمز M:N

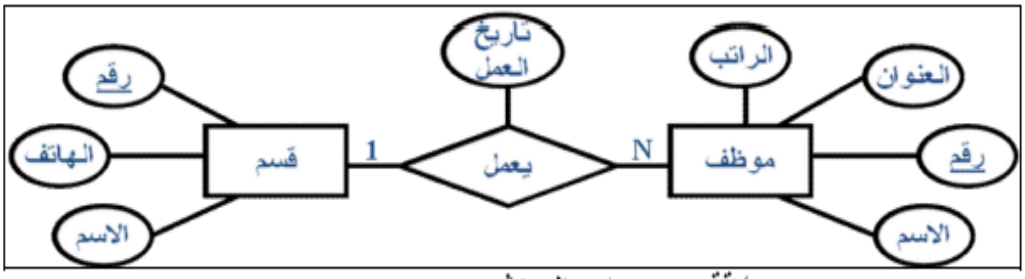


علاقة ١:١ واحد-الى-واحد (one-to-one)

لاحظ أنه لكل طالب سجل طبي واحد (نوع العلاقة ١)، والسجل يكون لطالب واحد (نوع العلاقة ١).

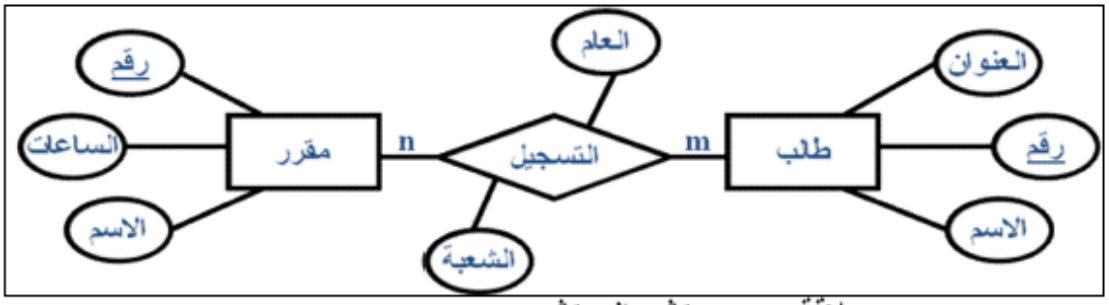
علاقة 1:N واحد-الى-كثير (one-to-many)

لا حظ أنه كل ابن يتبع لموظف واحد، لأنه لكل إبن أب واحد!، ولكن الموظف قد يكون له عدة أبناء.



علاقة 1:N واحد-الى-كثير (one-to-many)

لاحظ أنه لكل موظف قسم واحد، فالموظف لا يمكن أن ينتمي لأكثر من قسم إداري واحد، ولكن القسم قد يكون فيه عدة موظفين.



علاقة M:N كثير-الى-كثير (many-to-many)

لاحظ أن الطالب قد يكون له عدة مقررات، وكذلك المقرر يمكن أن يسجله عدة طلبة.

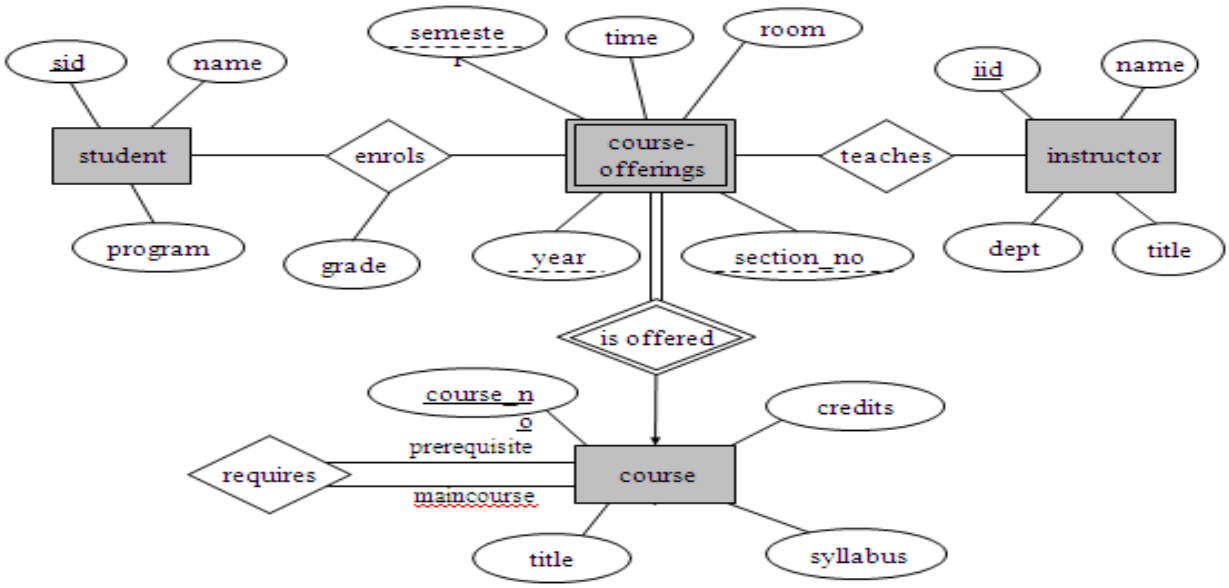
أنواع القيود على العلاقات Relationship Constraints:

كما أوضحنا سابقا فالعلاقات تختلف فيما بينها ، وكذلك شروط العلاقة تختلف حسب طبيعة العلاقة ، وحسب الكيانات المرتبطة، وقيد العلاقة هو القيد الذي يعتمد على طبيعة اشتراك كيانين في علاقة ما ، هل هو اشتراك كلي أم جزئي؟، ويحدد نوع الاشتراك ما إذا كان وجود الكيان يعتمد على كونه مرتبط بكيان آخر عن طريق العلاقة.

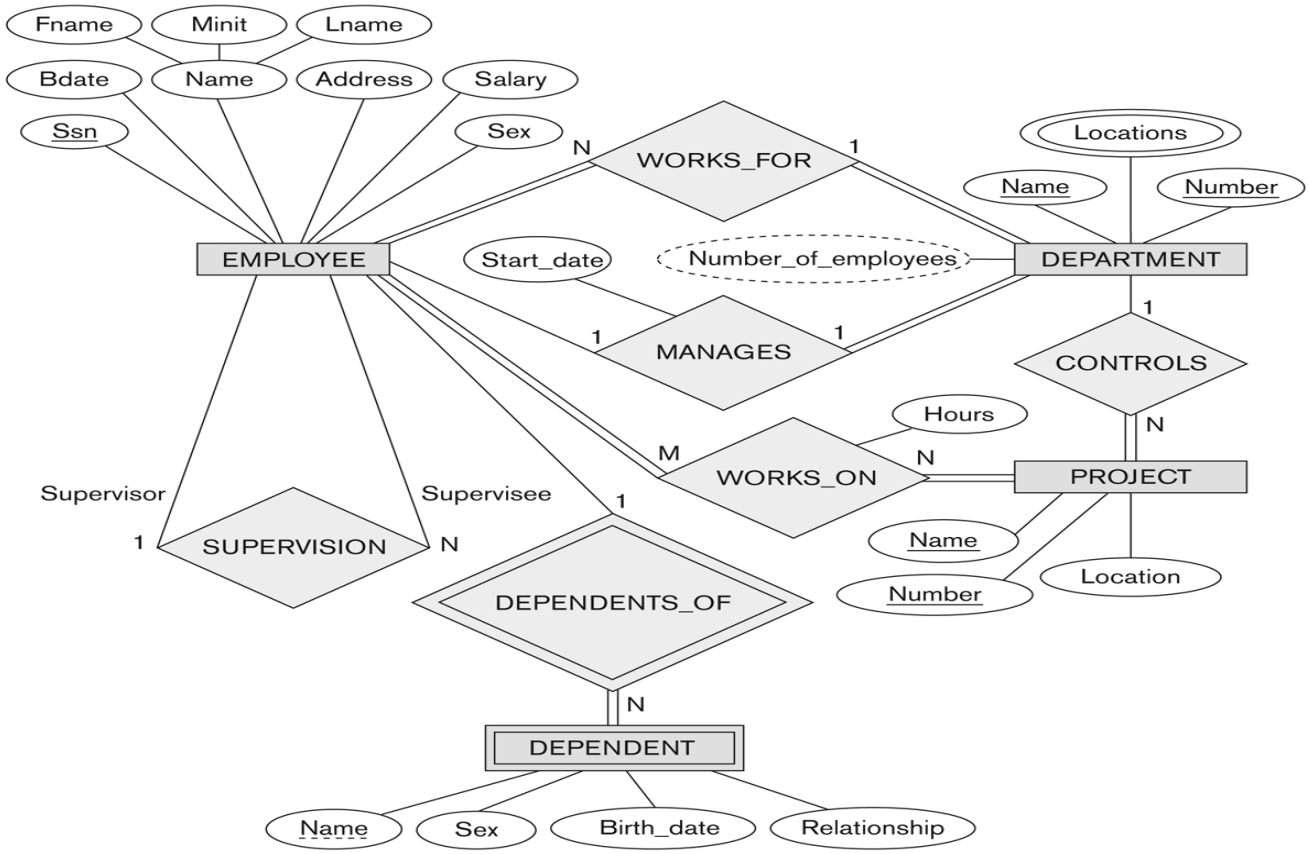
١.الاشتراك الكلي (Total participation) : نقول أن العلاقة علاقة الاشتراك الكلي ، إذا كان كل وحدة في الكيان الأول يجب أن ترتبط بوحدة من الكيان الآخر ضمن العلاقة ، يسمي هذا القيد بقيد "ارتباط الوجود" ، أي أن وجود وحدة من كيان ما يستلزم ارتباطها بوحدة من كيان آخر، ومثال ذلك كيان (سجل الاكاديمي) لطالب في نظام معلومات الجامعة ، إذ لا بد أن يرتبط أي قيد في السجل الطبي بطالب ما في كيان الطلاب ، و إلا فلا معنى للعلاقة هذه.

- ويتم تمثيل قيد الاشتراك الكلي ، برسم خط مزدوج ، يربط الكيانات المرتبطة بهذه العلاقة من جهة الكيان المعتمد على الاشتراك الكلي

٢.الاشتراك الجزئي (Partial participation) : نقول أن العلاقة علاقة اشتراك جزئي ، إذا كانت بعض الوحدات في الكيان المشترك بالعلاقة ترتبط ببعض الوحدات في الكيان الآخر ضمن العلاقة ، ويتم تمثيل قيد الاشتراك الجزئي برسم خط مفرد يربط الكيانات المرتبطة.



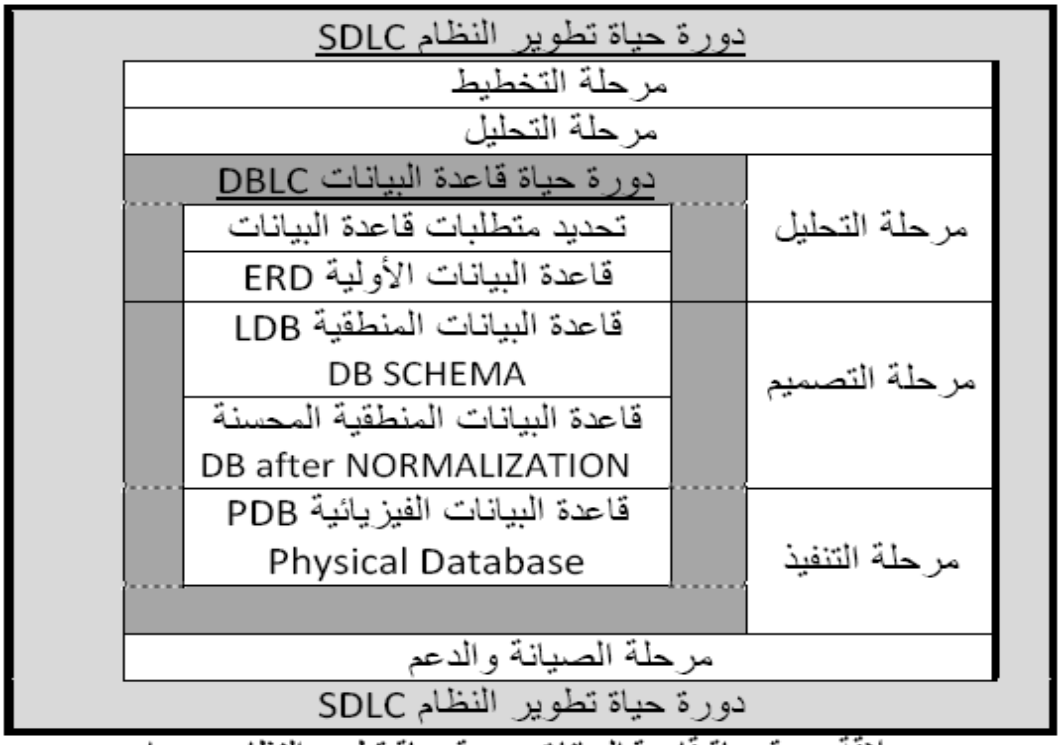
E-R diagram for a university.



تصميم قواعد البيانات العلاقية
Design of Relational Database

دورة حياة قاعدة البيانات Database Life Cycle (DBLC)

- إن عملية تطوير قاعدة البيانات تمر بمجموعة من المراحل ، هذه المراحل المتتالية تسمى بدورة حياة قاعدة البيانات.
- هذه المراحل أو دورة الحياة تمر بصورة متزامنة ضمن مراحل دورة حياة نظام المعلومات ، كما يوضح الشكل التالي:



علاقة دورة حياة قاعدة البيانات بدورة حياة تطوير النظام عموما

تتكون دورة حياة قاعدة البيانات من المراحل التالية :

١. تحديد المواصفات و المتطلبات الخاصة بقاعدة البيانات ، وهي مرحلة جزئية ضمن جمع مواصفات و متطلبات نظام المعلومات في مرحلة التحليل.
٢. إعداد قاعدة البيانات الأولية ، و فيها يتم تصميم نموذج اولي للبيانات بواسطة مخططات الكيان العلاقة(E-RD).
٣. تصميم قاعدة البيانات المنطقية ، تحويل قاعدة البيانات الأولية أو مخطط الكيان/العلاقة إلى مخطط الاسكيما ، و ذلك بإتباع قواعد التحويل

٤. تحسين قاعدة البيانات المنطقية ، وذلك بتطبيق قواعد تطبيع البيانات Normalization التي تهدف إلى تقليل تكرارية البيانات من أجل رفع كفاءة قاعدة البيانات ما أمكن .
٥. تنفيذ قاعدة البيانات الفيزيائية physical database: وفي هذه المرحلة يتم كتابة أكواد إنشاء قاعدة البيانات بلغةSQL ، و يحدد فيها بنية الجداول و نوع بيانات الحقول والمفاتيح الاساسية و الاجنبية و باقي شروط تصميم قاعدة البيانات ثم تنفيذ ذلك ضمن مدير قاعدة بيانات DBMS مناسب ، مثل oracle, access, (sqlserver, mysql …. etc)

مخطط قواعد البيانات Database Schema :

هو مخطط يصف قاعدة البيانات بشكل رسومي تمهيدا لبنائه على شكل جداول في نظام إدراة قواعد بيانات DBMS
هو مخطط ينتج عن عملية إخضاع مخطط الكيان العلاقة لخوارزمية التحويل Mapping Algorithm

لوصف مخطط قواعد البيانات ، نستخدم المصطلحات التالية :

- **العلاقة (جدول السكيما)relation :** أو يمكن أن نطلق عليها إسم الجداول ، وهي مكونات مخطط قاعدة البيانات الناتجة من إجراء عمليات تحويل مخطط كينونة - علاقة.

- **الحقل field :** هو العمود column الذي يشكل جزء من مكونات الجدول ، و يتكون من مجموعة من الأعمدة أو الحقول التي تتميز بتجانس بيانات كل حقل على أنه يمكن أن يكون نوع بيانات كل حقل مختلفا عن بيانات النوع الآخر

- **السجل Record :** هو الصفrow الذي يمثل وحدة instance من وحدات الكيان ، بعد تحويله إلى جدول ، و يتكون الصف من الخلايا الناتجة عن تقاطعه مع الأعمدة المكونة للجدول.

  - المفتاح الرئيسي Primary Key(PK): هو حقل في جدول يتميز بأن قيمه وحيدة في جميع صفوف الجدول ، وتكون قيمته مميزة لكل صف عن أي صف آخر.

  - المفتاح الأجنبي foreign key(FK) : هو حقل موجود في جدول وهو لا يمثل واحدة من صفاته ، و لكنه يعتبر مفتاح أجنبيا لأنه يمثل جدولا آخر ، ويجب أن يكون هو نفسه المفتاح الرئيسي في ذلك الجدول، أو على الأقل تكون قيمته وحيده unique value و يقوم المفتاح الأجنبي بتمثيل العلاقة relationship و الربط بين جدولين.

التحويل من مخطط الكيان العلاقة إلى مخطط قواعد بيانات : Mapping ERD to DB schema

تتم عملية تحويل مخطط ERD ، بتطبيق مجموعة من الخطوات البسيطة ، تسمى خوارزمية التحويل Mapping Algorithm ، وتتكون هذه الخطوات من جميع الحالات البسيطة المحتملة التي قد تكون موجودة في النموذج الأولى و يتم تطبيق هذه الخوارزمية كاملة مع تجاوز الحالات التي لم تظهر في النموذج الأولي

خوارزمية التحويل (Mapping Algorithm):

١. تحويل الكيانات العادية (القوية)
٢. تحويل الكيانات الضعيفة
٣. تحويل العلاقات الثنائية من النوع 1:1
٤. تحويل العلاقات الثنائية من النوع 1:N
٥. تحويل العلاقات الثنائية من النوع N:M
٦. تحويل الصفات متعددة القيم
٧. تحويل العلاقات فوق الثنائية

١. تحويل أنواع الكيانات العادية : يتم هنا تحويل جميع الكيانات العادية ، أي الكيانات غير الضعيفة ، بإنشاء جدول يتكون من الحقول التي تقابل صفات ذلك الكيان. ويتم تحديد أحد مفاتيح الكيان ، و تسميته بالمفتاح الرئيسي primary key(PK) و إذا كانت الصفة التي تمثل المفتاح من النوع المركب فإن المفتاح الرئيسي سيكون مجموعة الحقول التي تنشأ من الصفة المركبة.

٢. تحويل الكيانات الضعيفة : يتم تحويل كل واحدة من الكيانات الضعيفة ، بإنشاء جدول يتكون من الحقول التي تقابل صفات ذلك الكيان ، كما يجب إضافة المفتاح الرئيسي للكيان القوي الذي يتبعه ذلك الكيان الضعيف ، و يكون المفتاح الرئيسي PK للجدول الجديد ، عبارة عن مفتاح مركب مكون من المفتاح الأجنبي FK بالإضافة إلى المفتاح الجزئي (Partial Key) الخاص به.

٣. تحويل العلاقات الثنائية من النوع (1:1): إذا كانت العلاقة بين الكيانين علاقة واحد- إلى - واحد فإن عملية التحويل تتم وفق عدة خيارات أشهرها ، خيار يسمى بطريقة المفتاح الأجنبي ، وفيه يتم إضافة المفتاح الرئيسي لأحد الجدولين إلى الجدول الآخر كمفتاح أجنبي و يفضل أن يكون الجدول الذي يحتوي على المفتاح الأجنبي ، هو الجدول الذي يكون نوع قيد اشتراكه في العلاقة ، من نوع (الاشتراك الكلي)

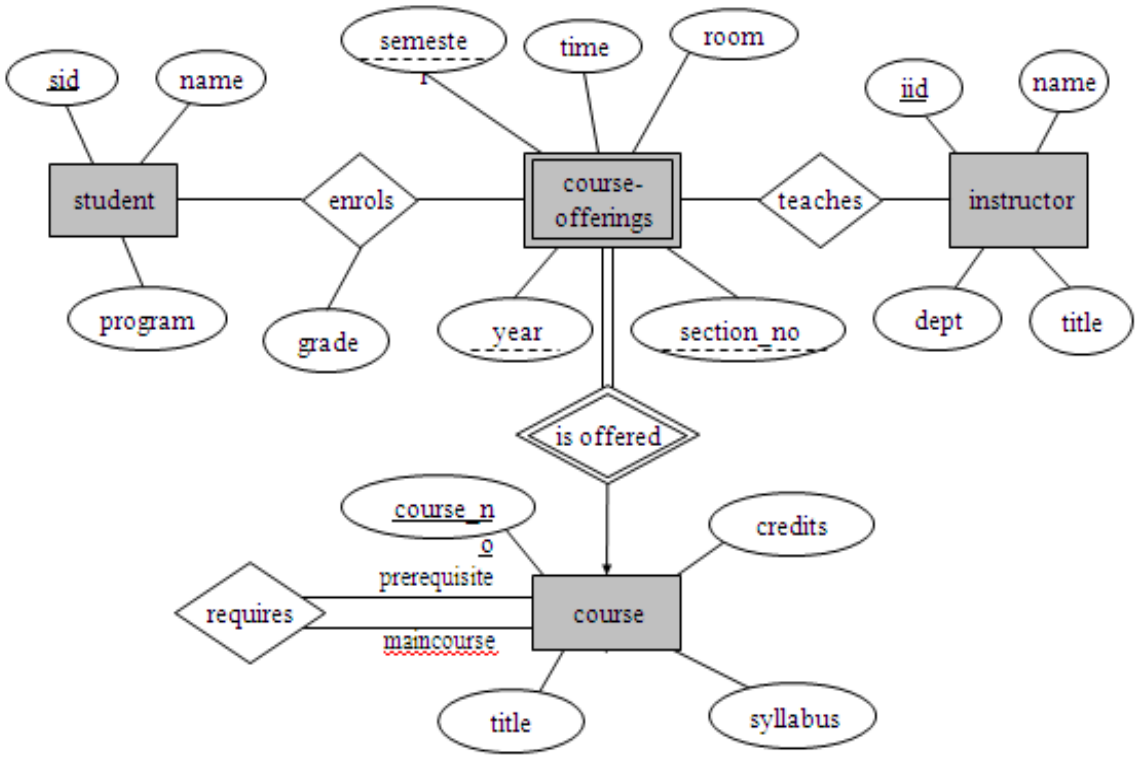
٤. تحويل العلاقات الثنائية من النوع (1:N): يتم هنا إنشاء جدولين لتمثيل الكيانين المرتبطين ، على أن يتم تطبيق طريقة المفتاح الأجنبي السابقة ، وذلك بإضافة المفتاح الرئيسي للجدول من جهة العلاقة (N) إلى الجدول الآخر المرتبط بالعلاقة (١) ، بغض النظر عن نوع قيد الاشتراك.

٥. تحويل العلاقات الثنائية من النوع (N:M): في هذا النوع من العلاقات ، يتم استحداث جدول جديد ، فيكون الناتج من هذه العلاقة ثلاثة جداول ، جدولين لتمثيل الكيانين المرتبطين بالعلاقة و يضم الجدول الثالث حقلين كمفتاحين أجنبيين يمثلان المفتاحين الرئيسيين في الجدولين ، و يمكن إضافة أي حقل آخر يكون له مغزى ، كأن تكون العلاقة لها صفة بذاتها ، فتتحول الصفة إلى حقل في الجدول الجديد.

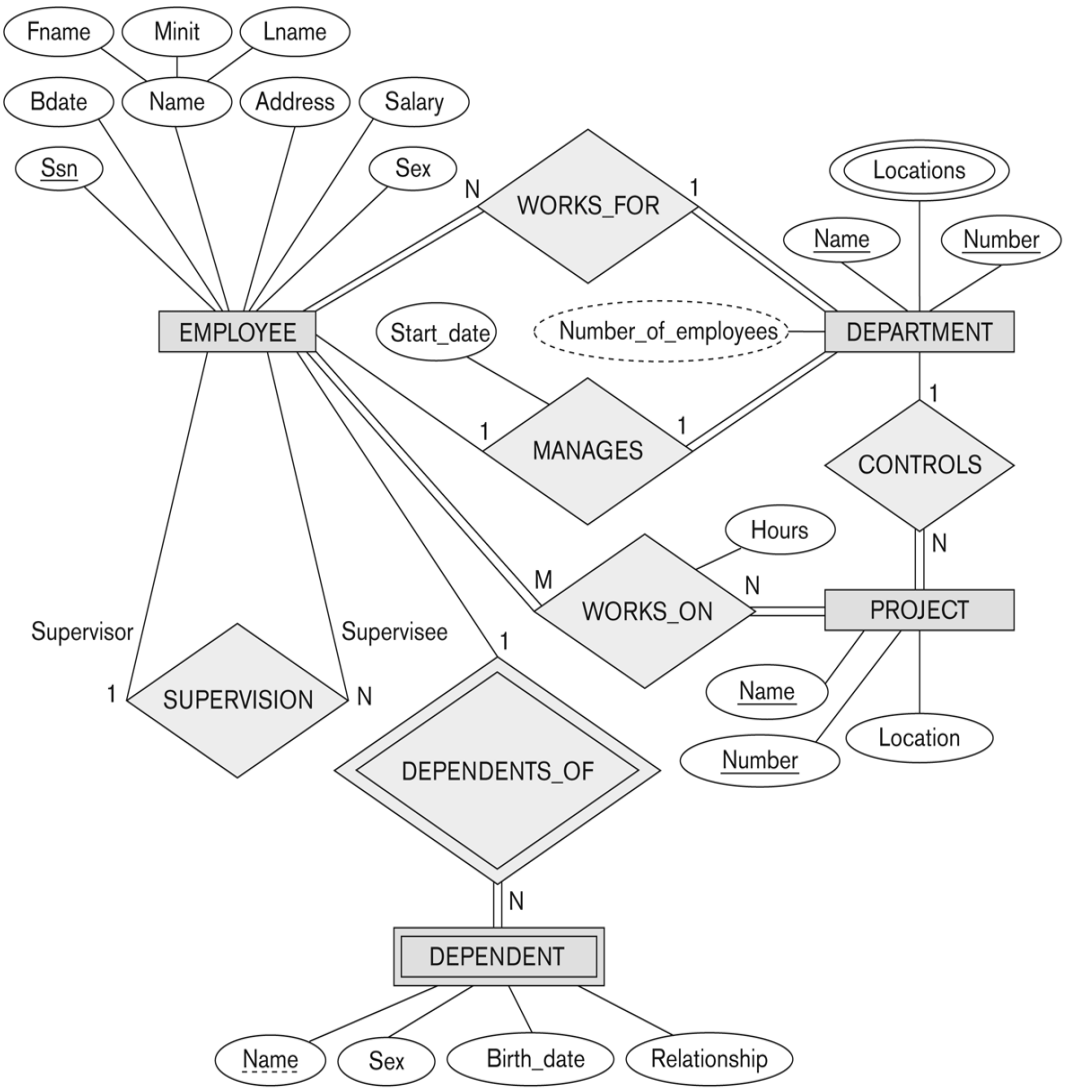
٦. تحويل الصفات متعددة القيم : يتم في هذه الحالة ، عادة ، إنشاء جدول جديد يضم الصفة المتعددة القيم كحقل ، و يضاف إلى الجدول مفتاح أجنبي FK يكون ممثلا للمفتاح الرئيسي في الجدول الناتج من الكيان الذي يحتوي على الصفة متعددة القيم . أما الصفات المركبة فتتحول إلى صفات بسيط ، فحقول عادية كما أوضحنا أعلاه ، و الصفات ذات القيم المشتقة تلغى من الجدول ، لأنها صفات قابلها للاشتقاق من صفات أخرى ، فلا داعي لوجودها

٧. تحويل العلاقات غير الثنائية ، كالعلاقة الثلاثية وما فوقها : في حالات نادرة تظهر لدينا علاقات معقدة ، كالعلاقة الثلاثية (بين ثلاثة كيانات) و الرباعية وما فوقها ، و تعالج هذه الحالة بطريقة معالجة الحالة الخامسة (حالة تحويل العلاقات الثنائية من النوع (N:M) حيث يتم إنشاء جدول جديد ، و إضافة المفاتيح الرئيسية للجداول المشتركة ، حسب عددها إلى الجدول الجديد كمفاتيح أجنبية مكونة بمجموعها ، مفتاحا مركبا يمثل المفتاح الرئيسي للجدول.

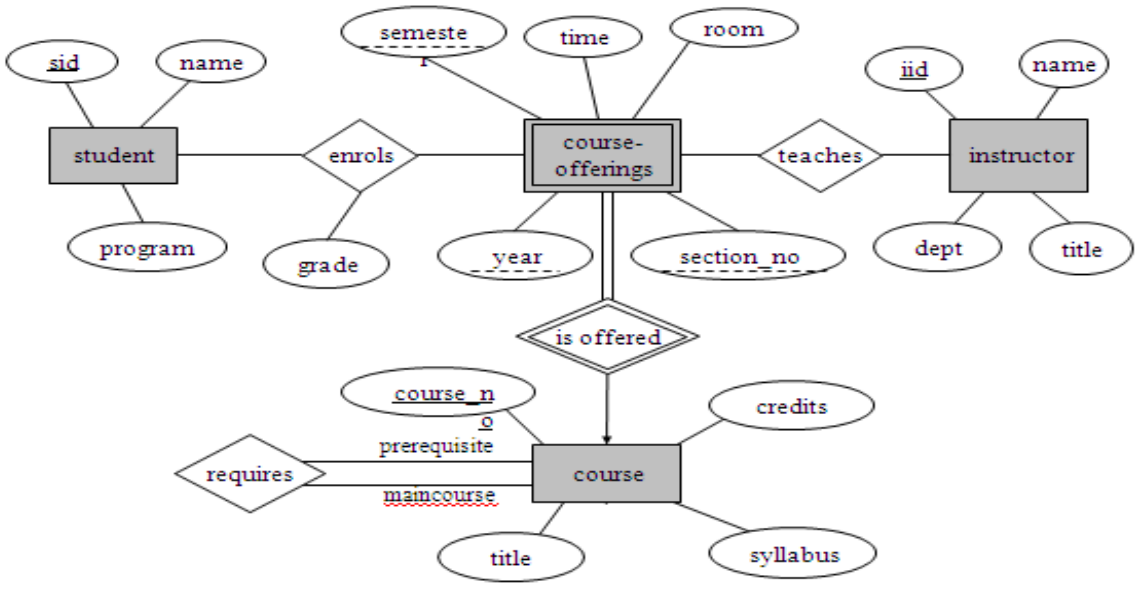
تمرين: حول كل من نماذج الكيان العلاقة التاليين إلى ما يقابلهما من مخطط قواعد البيانات Database Schema



E-R diagram for a university.



تحويل مخطط الكيان العلاقة لجامعة إلى ما يقابله من مخطط قواعد البيانات (جداول)



E-R diagram for a university.

١. نبدأ بتحويل الكيان العادي :

- يتم تمثيل الكيان العادي (القوي) بشكل المستطيل أحادي الإطار ، و يحتوي مخطط الكيان العلائقي السابق على ثلاث كيانات هي :

-الطالب (Student)
-المحاضر (Instructor)
-المقرر (Course)

- و تتم عملية تحويل الكيانات القوية بتمثيل كل منها بجدول يحمل إسم الكيان ، و يحتوي حقولا تمثل الصفات (تمثل الصفات بالشكل البيضاوي) المرتبطة (الارتباط يمثل بخط مستقيم) بالكيان.
- في حالة الصفة المركبة (شكل بيضاوي مرتبط باشكال بيضاوية جزئية) يتم أخذ الأجزاء المكونة للصفة المركبة.
- يتم تجاهل الصفة المشتقة (تمثل بشكل بيضاوي متقطع الاطار) بسبب القدرة على إشتقاقها بجملة إستعلام.
- أما الصفة متعددة القيمة (تمثل بشكل بيضاوي مزدوج الإطار) فيتم إنشاؤها في جدول مستقل يحمل إسم الكيان و الصفة متعددة القيمة ، و يحتوي حقولا تمثل الصفة متعددة القيمة وصفة المفتاح الرئيسي للكيان.

- نلاحظ في هذا المثال أن كل الصفات من النوع البسيط
- و يكون المفتاح الرئيسي (Primary Key) للجدول هو مجموعة صفات المفتاح الرئيسي المرتبطة بالكيان
- و ينتج عن عملية التحويل الجداول التالية :

STUDENT	*sid*	*name*	*program*

INSTRUCTOR	*iid*	*name*	*dept*	*title*

COURSE	***courseno***	***title***	***syllabus***	***Credits***

٢. تحويل الكيان الضعيف :

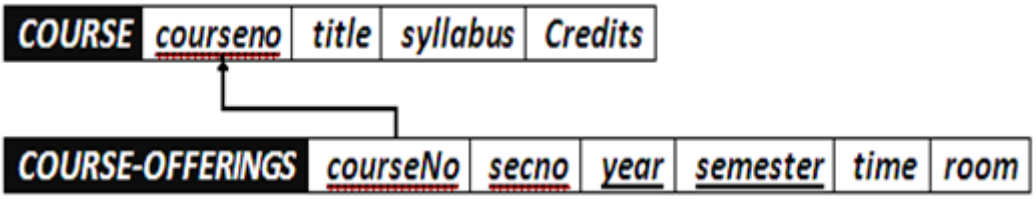
- يتم تمثيل الكيان الضعيف بشكل مستطيل مزدوج الإطار ، و سبب ضعف الكيان ، ينتج من عدم وجود صفة مفتاح رئيسي له ، ولكن يحتوي على صفة مفتاح جزئي ، ولدينا في هذا المثال كيان ضعيف واحد هو:
- كيان الشعب الفصلية المقترحة (Course-Offerings)
- و يحتوي هذا الكيان على صفات المفتاح الجزئي التالية :

أ - السنة (Year)
ب - الفصل (Semester)
ج - رقم الشعبة (Section-no)

- و يجب أن يرتبط الكيان الضعيف بكيان قوي بواسطة علاقة تعريف (تمثل علاقة التعريف بشكل معين مزدوج الإطار) ، وذلك في سبيل تقوية الكيان الضعيف.
- و تتم عملية تحويل الكيان الضعيف بتحويله إلى جدول يحمل إسم الكيان الضعيف ، و يحتوي حقولا من الصفات المرتبطة به ، بالاضافة إلى حقل المفتاح الرئيسي من جدول الكيان القوي المرتبط معه بعلاقة تعريف ، وفي حال وجود أي صفة على علاقة التعريف ، يتم تمثيلها بحقل في الجدول . و يكون المفتاح الرئيسي للجدول هو مفتاح الكيان القوي بالإضافة الى المفاتيح الجزئية في الكيان الضعيف.
- و ينتج عن عملية التحويل الجدول التالي :

COURSE-OFFERINGS	***courseno***	***secno***	***year***	***semester***	*time*	*room*

- و بذلك يرتبط جدول الشعب الفصلية المقترحة (Course-Offerings) بجدول المقرر (Course) بوجود المفتاح الأجنبي (Foreign Key) رقم المقرر (courseno)



٣. تحويل العلاقات :

- يتم تمثيل العلاقة بشكل معين أحادي الإطار
- في هذه الحالة يتم التعامل مع العلاقة حسب نوعها كل على حده، حيث تصنف العلاقات إلى الأنواع التالية :

أ - علاقة واحد إلى واحد (One-to-One Relationship)
ب -علاقة واحد إلى كثير (One-to-Many)
ج -علاقة كثير إلى كثير(Many-to-Many)

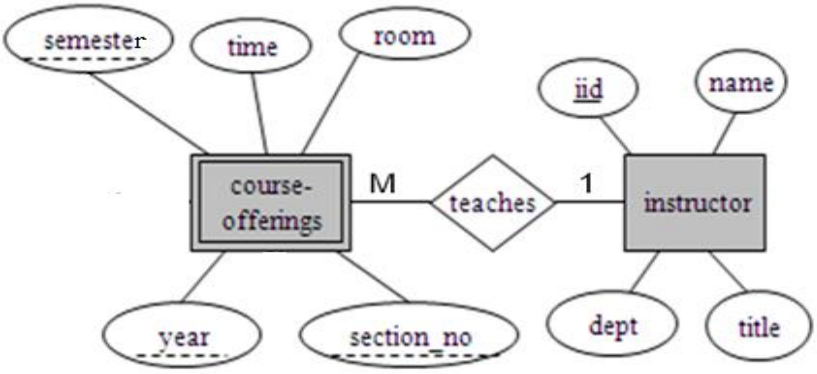
- يتم تجاهل علاقة التعريف (شكل معين مزدوج الإطار)، لانه تم بناؤها مسبقا عند تحويل الكيان الضعيف

٣.أ) تحويل علاقة واحد إلى واحد:

- إذا كانت العلاقة بين الكيانين علاقة واحد- إلى - واحد فإن عملية التحويل تتم وفق عدة خيارات أشهرها ، خيار يسمى بطريقة المفتاح الأجنبي ، وفيه يتم إضافة المفتاح الرئيسي لأحد الجدولين إلى الجدول الآخر كمفتاح أجنبي و يفضل أن يكون الجدول الذي يحتوي على المفتاح الأجنبي ، هو الجدول الذي يكون نوع قيد اشتراكه في العلاقة ، من نوع (الاشتراك الكلي) .وفي حال وجود صفة على العلاقة يتم إضافتها كحقل إضافي مع المفتاح الأجنبي.
- في هذا المثال لا توجد علاقة من النوع واحد - إلى- واحد

٣.ب) تحويل علاقة واحد إلى كثير:

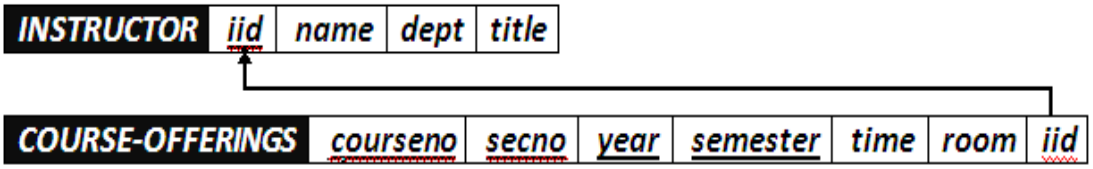
- في هذه الحالة يتم أخذ نسخة من المفتاح الرئيسي من الجدول ذو طرف العلاقة واحد ويتم إضافته كحقل مفتاح أجنبي في جدول طرف العلاقة كثير. وفي حال وجود صفة على العلاقة يتم إضافتها كحقل إضافي مع المفتاح الأجنبي.
- في هذا المثال لدينا علاقة يُدرس (teaches) بين كيان المحاضر(Instructor) وكيان الشعب الفصلية المقترحة(Course-Offerings)،
- في هذا المثال لدينا علاقة يُدرس (teaches) بين كيان المحاضر (Instructor) وكيان الشعب الفصلية المقترحة (Course-Offerings).



- وعليه يتم أخذ المفتاح الرئيسي (iid) من جدول Instructor كونه طرف العلاقة واحد ، و يضاف كمفتاح أجنبي في جدول course-offerings، و بالتالي يتم تعديل جدول Course-Offerings ليصبح بالشكل التالي:

COURSE-OFFERINGS	courseno	secno	year	semester	time	room	iid

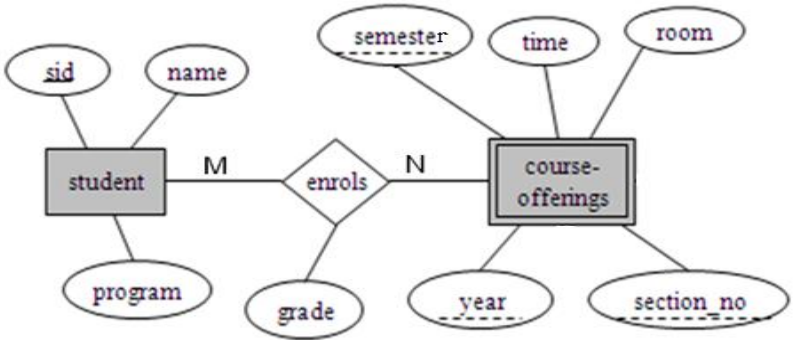
- وبذلك يرتبط جدول الشعب الفصلية المقترحة (Course-Offerings) بجدول المحاضر (Instructor) بوجود المفتاح الأجنبي رقم المحاضر (iid)



٣.ج) تحويل علاقة كثير إلى كثير:

- في هذا النوع من العلاقات يتم إنشاء جدول جديد يحمل إسم العلاقة ، وتكون حقوله هي حقول المفتاح الرئيسي من كلا الجدولين المشاركين في العلاقة مكونة المفتاح الرئيسي للجدول الجديد ، وفي حال وجود صفة على العلاقة يتم إضافتها كحقل إضافي في الجدول

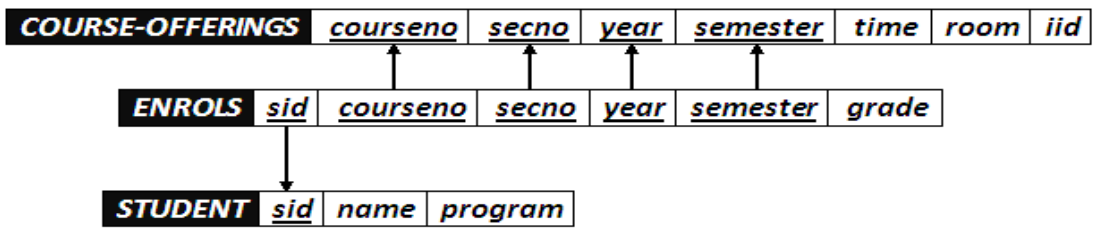
- في هذا المثال ، يوجد لدينا علاقتين من نوع كثير إلى كثير، هما :
  - علاقة يُسَجِّل (enrols)



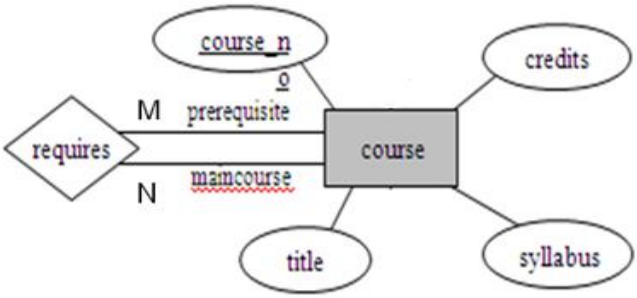
- ينتج عن هذه العلاقة الجدول التالي:

ENROLS	sid	courseno	secno	year	semester	grade

- و بذلك يرتبط جدول الشعب الفصلية المقترحة (Course-Offerings) بجدول الطالب (Student) و جدول يُسجل (Enrolls) بوجود المفتاح الأجنبي رقم الطالب (sid) من جدول الطالب و المفتاح الأجنبي (courseno, secno, semester, year) من جدول الشعب الفصلية المقترحة (CourseOfferings)



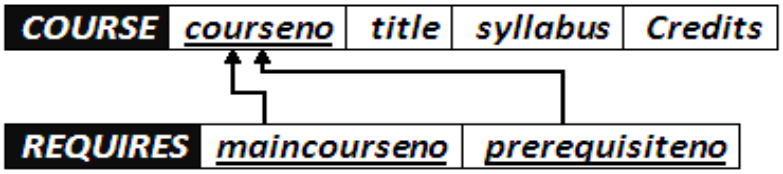
٢)علاقة يتطلب (Requires) :



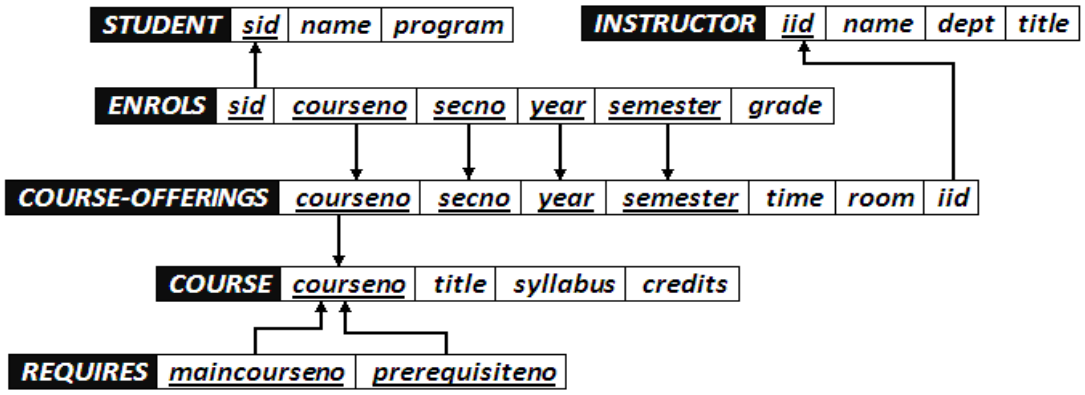
- هذه العلاقة تسمى علاقة تغذية راجعة (Recursive Relationship) أو علاقة كيان مع نفسه ، و يمكن تصنيفها كعلاقة أحادية ذات تغذية راجعة ، وفي هذا النوع من العلاقات يتم وضع التسمية المقترحة لحقول الجدول الناتج كون الحقول ناتجة عن حقل واحد هو حقل المفتاح الرئيسي ، وبالتالي يجدر بنا إعادة التسمية لتفادي الخطأ في تشابه الإسم .
- و ينتج عن هذه العلاقة الجدول التالي :

REQUIRES	maincourseno	prerequisiteno

- و بذلك يرتبط جدول يتطلب (Requires) مع جدول المقرر (course) :

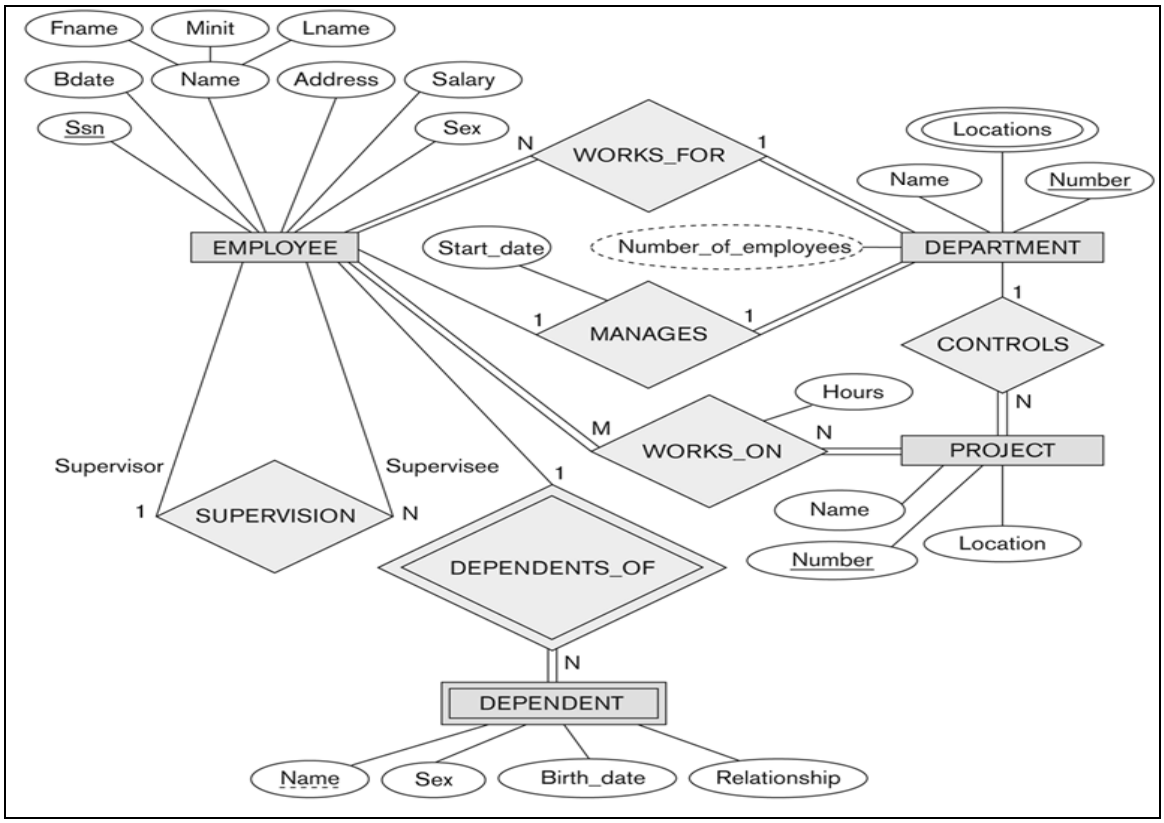


- و بذلك نكون قد أنشأنا مخطط قواعد البيانات الناتج عن التحويل للمثال رقم (١) ، وهو على الشكل التالي :



من هنا تشترك المحاضرة السابعة الجزء الثاني مع السابعة بنفس الشرائح ☺

- نلاحظ الارتباط الوثيق بين جداول قاعدة البيانات ، فلا يجوز أن يكون هناك جدول دون علاقة بباقي الجداول ، أو بمعزل عن الجداول الأخرى في قاعدة البيانات



١- بتحويل الكيان العادي :

- يحتوي مخطط الكيان العلائقي السابق على ثلاث كيانات عادية هي :

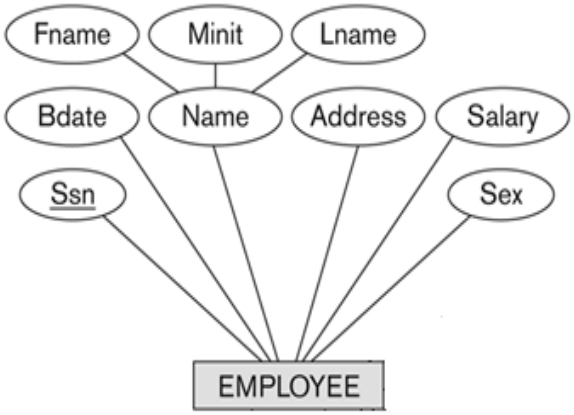
أ - الموظف (Employee)

ب -القسم(Department)

ج -المشروع(Project)

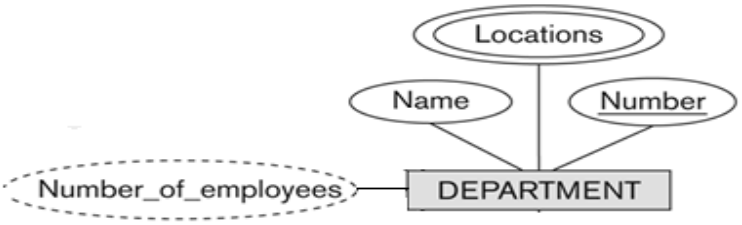
1.أ) كيان الموظف (Employee) :

- نلاحظ في هذا الكيان وجود صفة مركبة هي صفة الإسم (Name) و التي تتكون من الصفات الجزئية الاسم الأول (Fname) ، و حرف الاسم الأوسط (Minit) ، و الاسم الأخير (Fname). وكما اوضحنا سابقا ، فإن الصفة المركبة في عملية التحويل تدرج بصفاتها الجزئية فقط
- و ينتج عن عملية التحويل الجدول التالي :

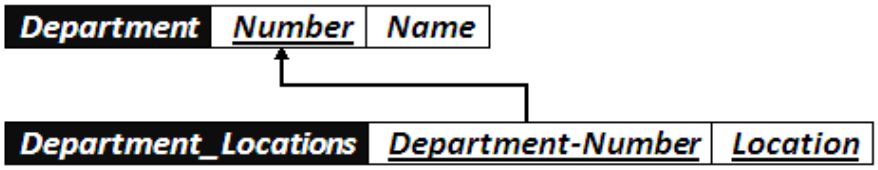


EMPLOYEE	*Ssn*	*Fname*	*Minit*	*Lname*	*Bdate*	*Address*	*Salary*	*Sex*

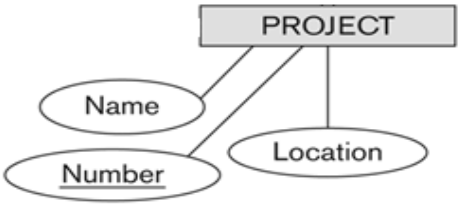
١.ب) كيان القسم (Department) :



- نلاحظ في هذا الكيان وجود صفة مشتقة هي عدد الموظفين (Number of employees) ، وذكرنا سابقا أننا في عملية التحويل نتجاه هذه الصفة ، ليتم بناؤها لاحقا بجملة إستعلام
- كما نلاحظ وجود صفة متعددة القيمة وهي المواقع (locations) وذكرنا سابقا أن الصفة متعددة القيمة يتم تحويلها إلى جدول مستقل مع المفتاح الرئيسي للكيان ، ويكون إسم الجدول مكون من إسم الكيان مضافا إليه إسم الصفة متعددة القيمة.
- و ينتج عن عملية التحويل الجدولين التاليين :



١.ج) كيان المشروع :

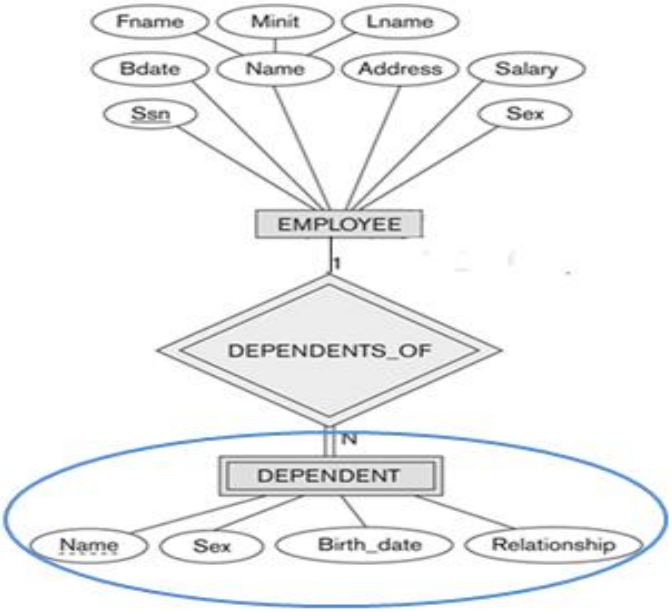


- هذا الكيان يحتوي على صفات بسيطة فقط ، وبالتالي ينتج عن عملية التحويل الجدول التالي :

Project	Number	Name	Location

٢. تحويل الكيان الضعيف :

- في هذا المثال كيان ضعيف واحد هو:
- كيان المعتمد على (Dependent)، حيث يصف هذا الكيان أفراد عائلة الموظف المعتمدين عليه في إعالتهم.



- و يحتوي هذا الكيان على صفة المفتاح الجزئي الإسم (Name) ، التالية :

- و يرتبط هذا الكيان الضعيف بكيان قوي هو كيان الموظف (Employee) بواسطة علاقة التعريف يعتمد على (Dependent_of).
- و تتم عملية تحويل الكيان الضعيف بتحويله إلى جدول يحمل إسم الكيان الضعيف (Dependent) ، ويحتوي حقولا من الصفات المرتبطة به ، بالاضافة إلى حقل المفتاح الرئيسي من جدول الكيان القوي (Employee) المرتبط معه بعلاقة التعريف (Dependent-of). و يكون المفتاح الرئيسي للجدول هو مفتاحا للكيان القوي (Empoyee-Ssn) بالإضافة إلى المفتاح الجزئي في الكيان الضعيف (Dependent-Name).

- و ينتج عن عملية التحويل الجدول التالي :

DEPENDENT	Employee-Ssn	Dependent-Name	Relationship	Sex	Birth_date

و بذلك يرتبط جدول (Dependent) بجدول (Employee) بوجود المفتاح الأجنبي رقم الموظف (Employee-Ssn)

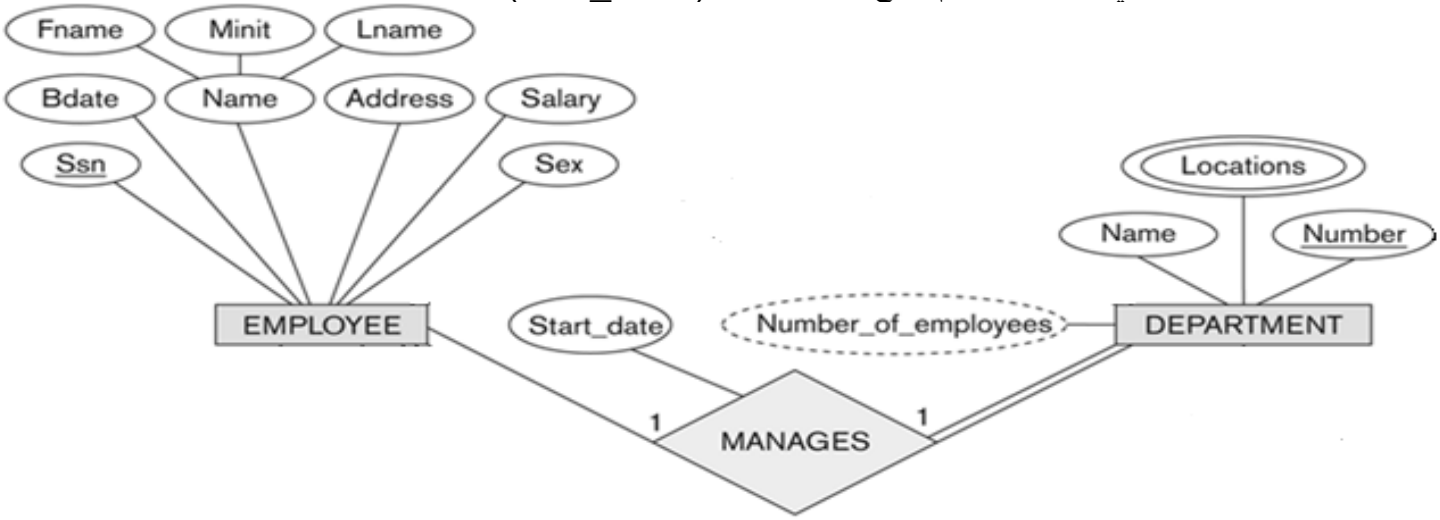
EMPLOYEE	Ssn	Fname	Minit	Lname	Bdate	Address	Salary	Sex

DEPENDENT	Employee-Ssn	Dependent-Name	Relationship	Sex	Birth_date

تحويل العلاقات :

٣.أ) تحويل علاقة واحد إلى واحد :

في هذا المثال توجد علاقة من النوع واحد- إلى- واحد ، وهي علاقة (Manages) بين كيان الموظف ، و كيان القسم ، و العلاقة ذات إشتراك كلي من جهة القسم ، مع وجود الصفة (start_date) على العلاقة.

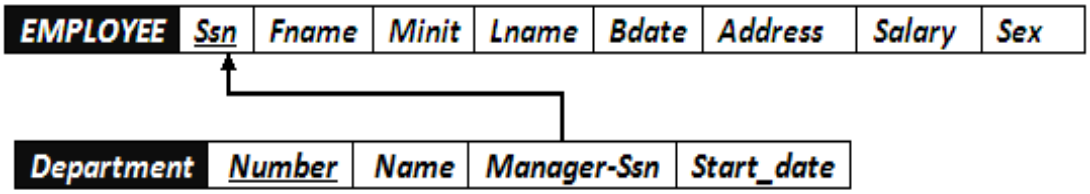


- وبناءا علي النقطة السابقة ، يتم أخذ نسخة من المفتاح الرئيسي لجدول الموظف (Employee_Ssn) ، مضافا إليها الصفة على العلاقة (start_date)، و وضعهما كتعديل على جدول القسم (department) ، ويفضل إعادة تسمية المفتاح الرئيسي ليدل على العلاقة وهي علاقة ادارة ، فهو مدير (Manager) ، وبالتالي بدلا من التسمية Employee_Ssn تصبح التسمية Manager_Ssn

- وعليه يتم التعديل على جدول القسم (Department) ليصبح بالشكل التالي :

Department	Number	Name	Manager-Ssn	Start_date

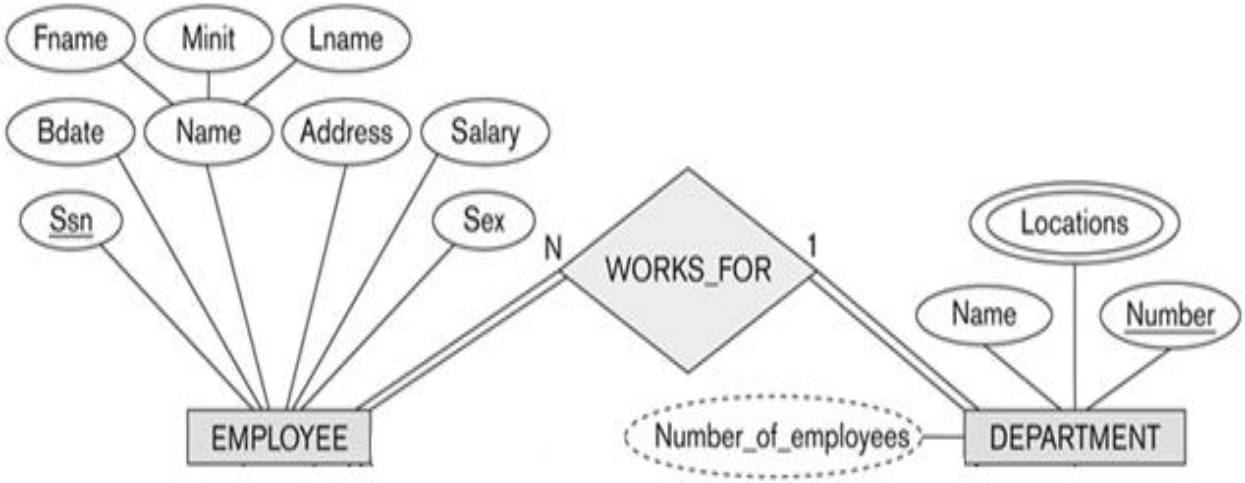
- وبذلك يرتبط جدول (Department) بجدول (Employee) بوجود المفتاح الأجنبي رقم الموظف (Employee-Ssn).



٣.ب) تحويل علاقة واحد إلى كثير:

- في هذا المثال لدينا ثلاث علاقات من نوع واحد إلى كثير وهي:

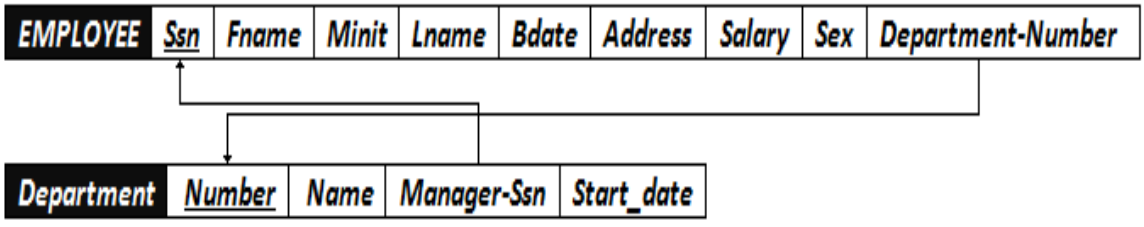
١.علاقة موظف يعمل في قسم (Works_for)



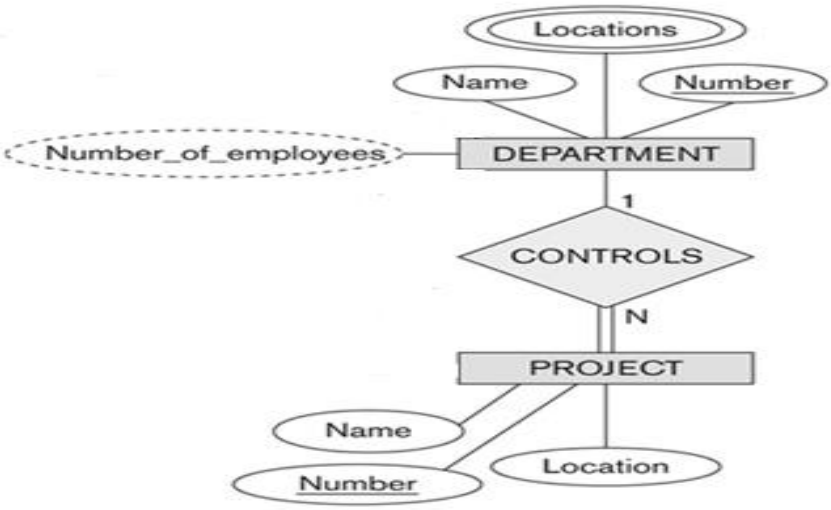
- نتيجة لهذه العلاقة يتم التعديل على جدول الموظف(Employee) ليحتوي على رقم القسم (Department_Number) الذي يعمل فيه ، ليصبح جدول الموظف (Employee) بالشكل التالي:

EMPLOYEE	Ssn	Fname	Minit	Lname	Bdate	Address	Salary	Sex	Department-Number

- و عليه تتوثق العلاقة بين جدول الموظف و القسم ، بإضافة مفتاح أجنبي جديد ، وهو رقم القسم في جدول الموظف ، كما يظهر في الشكل التالي :



٢. علاقة قسم يتحكم بمشروع (Controls)



- نتيجة لهذه العلاقة يتم التعديل على جدول المشروع (Project) ليحتوي على رقم القسم (Department_Number) الذي يتحكم في المشروع ، ليصبح جدول المشروع (Project) بالشكل التالي :

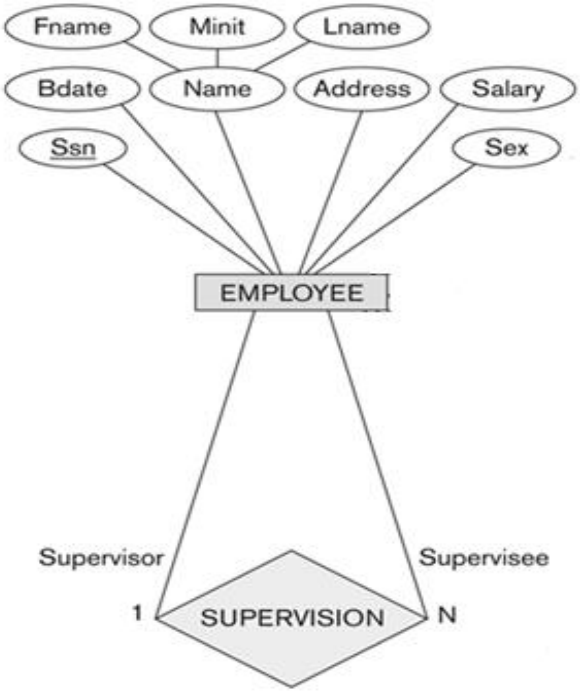
Project	*Number*	*Name*	*Location*	*Department_Number*

- وعليه يرتبط جدول القسم بجدول المشروع بواسطة المفتاح الأجنبي رقم القسم (Department-Number) الذي تم إضافته إلى جدول المشروع ، كما يظهر في الشكل التالي:

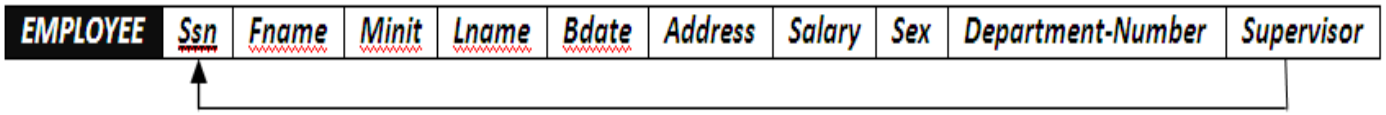
Department	***Number***	***Name***	***Manager-Ssn***	***Start_date***

Project	***Number***	***Name***	***Location***	***Department_Number***

٣.علاقة إشراف موظف على موظف (Supervision):

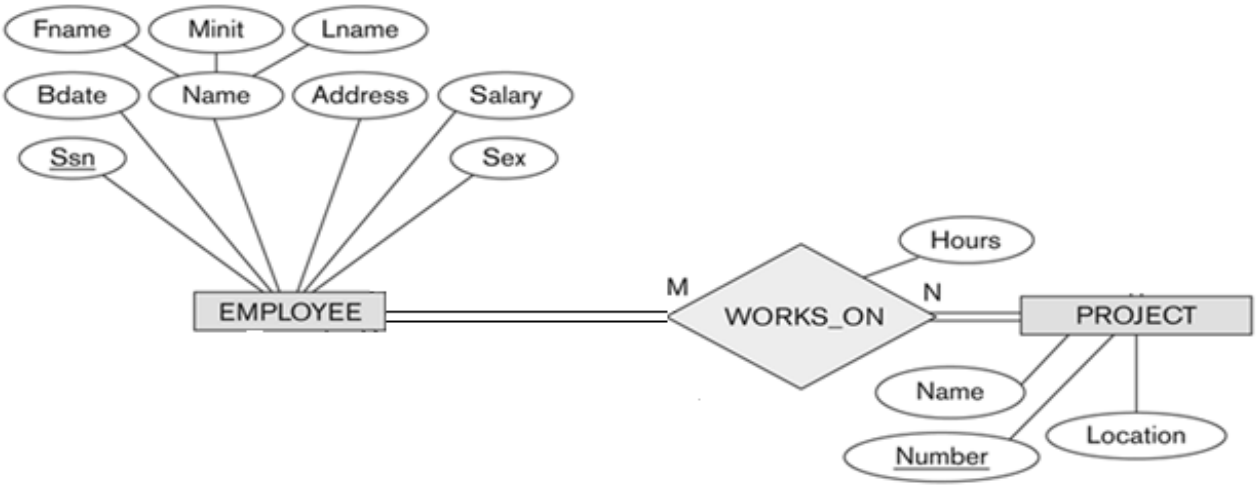


- هذه العلاقة هي علاقة أحادية ذات تغذية راجعة ، أي علاقة كيان على نفسه.
- نتيجة هذه العلاقة أن يتم إضافة حقل جديد هو حقل المشرف (Supervisor)، وهو حقل يعبر عن رقم الموظف ، مع إعطائه خصوصية كونه مشرفا على غيره من الموظفين الحقل يتم إضافته إلى جدول الموظف (Employee) ليصبح بالشكل التالي :



٣.ج) تحويل علاقة كثير إلى كثير:

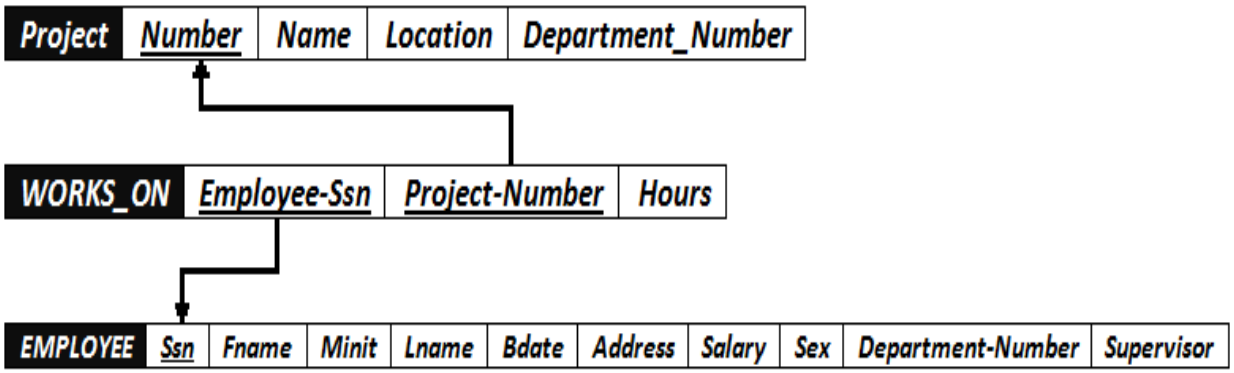
- في هذا المثال ، يوجد لدينا علاقة واحدة من نوع كثير إلى كثير، هي علاقة موظف يعمل على مشروع (Works_on)



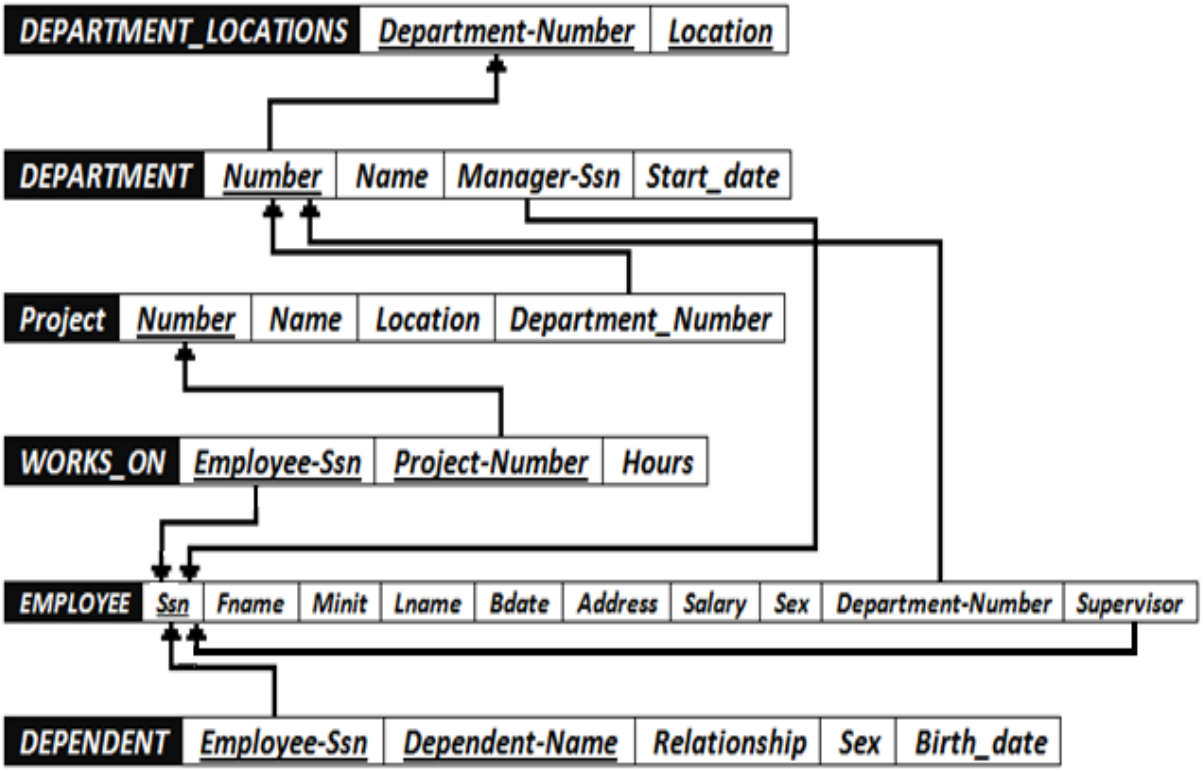
- وفي هذه الحالة يتم إنشاء جدول جديد تحت إسم العلاقة (Works-on)، ونقوم بأخذ المفتاح الرئيس من كل الجدولين ، الموظف (Employee) والمشروع (Project) ، بالإضافة إلى الصفة عدد الساعات (Hours) المرتبطة بالعلاقة ، و يصبح لدينا الجدول التالي :

WORKS_ON	Employee-Ssn	Project-Number	Hours

و بذلك يرتبط جدول الموظف (Employee) و جدول المشروع (Project) عن طرق الجدول الجديد (Works_on) عن طريق المفاتيح الأجنبية رقم الموظف (Employee-Ssn) من جدول الموظف Employee ، و رقم المشروع (Project-Number) من جدول المشروع (Project)



- وبذلك نكون قد أنشأنا مخطط قواعد البيانات الناتج عن التحويل للمثال رقم (2) ، وهو على الشكل التالي:



تصميم قواعد البيانات العلاقية
Design of Relational Database

تحسين قواعد البيانات :

- يمكن العمل على تحسين قواعد البيانات باستخدام ما يعرف بتطبيع قواعد البيانات Database Normalization، و الذي يستخدم لإزالة عيوب البيانات المخزنة و الوصول إلى مخطط قواعد بيانات متين ، و يتم تطبيق التطبيع على أربعة مراحل متتالية هي :

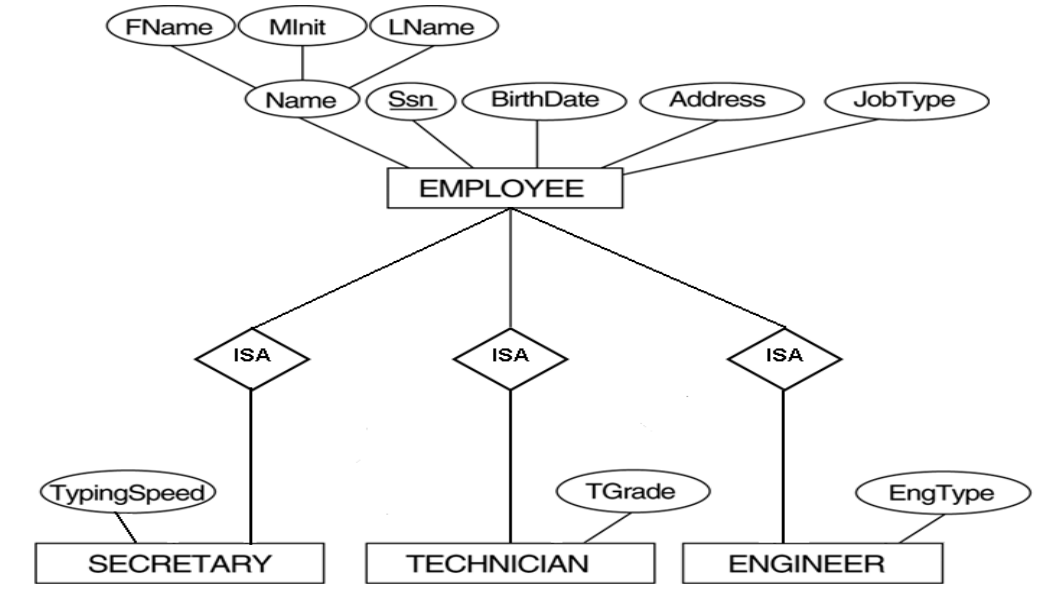
١) شكل التطبيع الأول (First Normalization Form)
٢) شكل التطبيع الثاني (Second Normalization Form)
٣) شكل التطبيع الثالث (Third Normalization Form)
٤) شكل التطبيع الرابع (Fourth Normalization Form)

- و نكتفي هنا بذكرها دون الخوض في تفاصيلها

مفهوم العلاقة الرابطة ISA :

- هي علاقة بين كيانين ،أحدهما طبقة أعلى (أصل أو أب)، و الأخرى طبقة أسفل (فرع أو إبن) متفرعة من الطبقة الأصل.
- عملية تحويل العلاقة ISA تختلف ، فهي تربط كيان الفرع بكيان الأصل بإستخدام المفتاح الرئيسي في كيان الأصل ، مكونة بذلك جدولا جديدا لكل كيان فرع مكون من حقل المفتاح الرئيسي من كيان الأصل مضاف إليه خواص كيان الفرع.

- مثال : ينتج عن تحويل علاقة ISA في مخطط الكيان العلاقة المقابل الجداول التالية:



EMPLOYEE

SSN	FName	MInit	LName	BirthDate	Address	JobType

SECRETARY

SSN	TypingSpeed

TECHNICIAN

SSN	TGrade

ENGINEER

SSN	EngType

تصميم قواعد البيانات من مستندات :

إن عملية تمثيل البيانات من نماذج تم جمعها من موقع الدراسة او الحالة المراد بناء قواعد البيانات إليها ، قد تختلف ، فمثلا لو أخذنا بعين الإعتبار الشكل التالي :

- وجود الشعار
- قيم محسوبة
- قيم مستنتجة
- سجلات من اكثر من كيان...الخ

Sultanate of Oman
CR No:1/602225
P.O. BOX: 430
P.C: 314 Al-Maldah
Finance code:10979601
Email: alfakhama@gmail.com

AL FAKHAMA

alfakhama@gmail.com

طرق الدفع : شيك

التاريخ : ٢٠١٠-١٠-٢٦

الطلبات	الكميه	بيسه	ريال	المجموع بيسه	المجموع ريال
مبيض	٧	٢٠٠	٤	٤٠٠	٢٩
مطهر حمامات	٥	٨٠٠	٢	-	١٤
ديتول	٤	-	٣	-	١٢
صابون ارضية	٣	٤٠٠	-	٢٠٠	١
معطر جو طيب	٧	٥٠٠	١	٥٠٠	١٠
إجمالي المبلغ				١٠٠	٦٧

البيانات الغير ممثلة في قواعد البيانات :

البيانات التي لا يتم تسجيلها في قاعدة البيانات :

- الشعارات او اسم الجهة صاحبة المستند.
- الرقم المسلسل او رقم كل صفحة مطبوعة أو تاريخ الطباعة
- البيانات التي يمكن إشتقاقها أو حسابها من بيانات أخرى
- الملاحظات و التوقيعات و التعليقات

مشاكل البيانات :

ذكرنا فيما سبق مشاكل ملفات البيانات ، و التي هي على علاقة وطيدة بمشاكل البيانات ، ونذكر منها :

١. تكرار البيانات
٢. مشاكل إدخال البيانات
٣. مشاكل حذف البيانات
٤. مشاكل التعديل و الحذف للبيانات

تكرار البيانات :

لنأخذ الجدول التالي :

ST#	NAME	ADDRESS	CRS#	TITLE	HOURS	INST#	INAME	DEPT	GRADE
5	FAHAD	QASSIM	CS101	Intro	3	7	Ali	CS	A
5	FAHAD	QASSIM	CS102	C	4	12	Jameel	MATH	B
5	FAHAD	QASSIM	CS103	C++	4	2	Rashid	CS	C+
5	FAHAD	QASSIM	CS325	DB1	3	2	Khalid	CS	B+
5	FAHAD	QASSIM	CS426	DB2	3	3	Tareq	CS	B

- نلاحظ أن :
  - بيانات الطالب تتكرر تخزينها مع تسجيله كل مقرر
  - تتكرر بيانات كل مقرر مع كل طالب يسجل ذلك المقرر
  - تتكرر بيانات كل مدرس مع كل مقرر يسجله طالب ما

- ينتج عن هذا التكرار مشاكل كثيرة مثل :
  - استهلاك حيز التخزين.
  - زيادة وقت إدخال البيانات
  - تضييع وقت القائمين على عملية الإدخال
  - ئؤثر على سرعة معالجة البيانات و استهلاك الاجهزة

مشاكل إدخال البيانات :

تتعدد مشاكل إدخال البيانات ، فنظرة إلى الجدول ذاته يظهر لنا المشاكل التالية:
- لا نستطيع إدخال بيانات أي مقرر لم يسجله طالب واحد على الأقل.
- لا يمكننا ادخال بيانات مدرس لم يدرس مقرر درسه طالب واحد على الاقل
- عند ادخال بيانات طالب جديد ، نضطر أن نترك معلومات المقرر و المدرس فارغا
- بسبب ادخال بيانات معينة اكثر من مرة، فإنه يزيد امكانية حدوث إدخال خطأ للبيانات مما يسبب تضارب في البيانات بغض النظر كان الخطأ مقصودا أو لا

مشاكل الحذف للبيانات :

عند حذف بيانات ما ، فان ذلك قد يؤثر على وجود بيانات أخرى في الجدول ذاته :
- عند حذف بيانات طالب وحيد في مقرر ما،يتم حذف بيانات المقرر نهائيا ونفقد بياناته
- عند حذف بيانات مدرس يدرس مقرر يحتوي على طالب وحيد، نفقد معلومات الطالب
- عند حذف سجل طالب أو مقرر أو مدرس ، فإنه يجب علينا فعل ذلك في سجلا أخرى متعلقة بنفس المحذوف الأمر الذي قد ننساه أو لا نستطيع حذفه

مشاكل التعديل للبيانات :

التعديل في بيانات سجل ما قد يخلق تضاربا مع معلومات سجل آخر، من الجدول ذاته :
- عند تعديل بيانات مقرر أو طالب أو مدرس، يجب أن نجري نفس التعديلات في كافة مواضع تخزين تلك البيانات
- ربما يحدث خطأ في تعديل البيانات في موضع تخزين معين دون مواضع أخرى سهوا أو عمدا. ينتج عن ذلك عد توافقية البيانات موضوع التعديل

أسباب فقد البيانات :

فقد البيانات أو ضياعها يحدث نتيجة أسباب كثيرة ، نذكر منها :

- خطأ بشري في إدخال بيانات غير سليمة
- عدم إكتمال تنفيذ بعض العمليات التي تجرى على البيانات
- تعطل نظم البرامج
- تعطل الأجهزة
- تعطل خطوط نقل البيانات (الشبكات)
- فيروسات الحاسب
- كوارث طبيعية

إمكانية إستعادة البيانات :

- يوفر نظام إدارة قواعد البيانات (DBMS) عدة تقنيات لمواجهة مشاكل فقد البيانات ، و استعادتها الى الحالة السابقة للفقد أو الخطأ مباشرة.
- من الامكانيات المتاحة للاستعادة نذكر :

- **النسخ الاحتياطي (Backup):** يوفر DBMS إجراءا آليا لعمل نسخة احتياطية لكامل قاعدة البيانات.

- **مفكرة النظام (System Log):** وهي آلية يستخدمها DBMS لتسجيل كافة التعاملات مع قاعدة البيانات

- **نقط الإختبار(Check Point):** وهو سجل ينشئه DBMS ليسجل فيه عملية فحص للنظام ، و اعتبار عملية الفحص الناجحة نقطة إسترجاع ممكنة

- **برنامج إدارة الإستعادة(Recovery Manager):** هو برنامج يقوم بإرجاع قاعدة البيانات إلى الحالة السليمة عند حدوث الأعطال ، ثم يعيد تنفيذ تعاملات البرامج و المستخدمين من المفكرة (مفكرة النظام)

طرق استعادة البيانات :

اعتمادا على نوع فقد البيانات ، و إمكانيات الإستعادة المتوفرة ، يمكن إستخدام أحد طرق الإستعادة التالية :

- **الإستعادة العكسية (Backward Recovery):** تستخدم لعمل تراجع عن الفعل أي Undo و العودة للحالة السابقة.

- **الإستعادة الأمامية (Forward Recovery):** تستخدم للعودة إلى نقطة مرجعية صالحة للإستخدام ، و بدء الإجراءات منها للوصول للوضع السليم أي Redo

- **إعادة التحميل وإعادة التشغيل (Restore & Rerun):** تستخدم للتعاملات السابقة للعطل بعد آخر نسخة إحتياطية . حيث يجري تحميل النسخة الاحتياطية، ثم إعادة تشغيل التعاملات التي تمت بعد عملية النسخ إلى وقت حدوث العطل.

- **سلامة وتكامل التعامل (Transaction Integrity):** حركة العمل(Transaction) هي مجموعة من العمليات التي إما أن تتم معا أولا تتم إطلاقا ، لذلك عند حدوث العمليات إذا كان تأثيرها يؤدي إلى ضياع أو تضارب في البيانات ، فإنها لا تتم Rollback، و إلا فانها تتم Commit.

أنواع فقد البيانات :

- فقد البيانات أنواع تتراوح ما بين إدخال قيم غير صحيحة لبيانات معينة إلى الفقد الكامل لبيانات قاعدة البيانات.
- بناءا على نوع فقد البيانات ،يتم تحديد طريقة الإستعادة المناسبة

من أنواع فقد البيانات :

تسجيل بيانات غير صحيحة :

- تصحيح الخطأ يدويا إذا كان الخطأ بسيطا
- اذا كانت الاخطاء كثيرة ، يمكن إستخدام الاستعادة العكسية أو البدء من آخر نقطة فحص

التعاملات المجهضة (الغير مكتملة) :

- نستخدم تقنية إلغاء / التراجع عن كافة نتائج التعاملات غير المكتملة

فناء قاعدة البيانات (Database Destruction):

- الاستعادة باعادة التحميل من النسخة الاحتياطية ثم تنفيذ كافة التعاملات بالاستعادة الامامية

تعطل النظام مع سلامة قاعدة البيانات (System Failure):

- نستخدم تقنية إلغاء / التراجع عن آخر تعاملات او البدء من آخر نقطة فحص

أمن قواعد البيانات :

- مع تقدم التكنولوجيا أصبحت الامور أكثر يسرا على المستخدم، كما اصبحت الامور أكثر خطورة بسبب الاختراقات الممكنة عن طريق شبكات الحاسوب التي تسبب خسائر طائلة في المال و المعلومات.
- يعرف أمن قاعدة البيانات على أنه حماية قاعدة البيانات من الاستخدام الخطأ أو الاضرار المتعمد للبيانات
- على من تقع مسئولية أمن قواعد البيانات ؟

- تقع المسؤولية عل مدير قاعدة البيانات DBA ، بسبب الصلاحيات الممنوحة له في استخدام الوسائل و السياسات اللازمة لحماية قاعدة البيانات

- ما هي الوسائل المستخدمة في حماية قواعد البيانات؟

- استخدام الجداول الافتراضية بدلا من الجداول الأصلية ، الامر الذي يقيد حرية التعامل مع البيانات الأصلية دون تعطيل عمليات الاستعلام
- استخدام قواعد الترخيص بالصلاحيات من قبل DBA بشكل كفؤ ، بحيث يحكم من يصل المعلومات بضوابط أمنية
- استخدام برامج تحجيم المستخدمين ، لتقييدهم وسد الطرق عليهم في الوصول لقاعدة البيانات بطريقة غير مشروعة
- استخدام برامج التشفير أو الترميز في هذه الحالة حتى لو تم الوصول للبيانات فلن يتم فهماها بسبب تشفيرها

برنامج إدارة قواعد البيانات مايكروسوفت أكسس ٢٠٠٧
Microsoft Access 2007

مقدمة :

- يعتبر برنامج Microsoft Access واحد من أشهر قواعد البيانات و التي تستخدم في ترتيب قواعد البيانات و استخراج النتائج منها و عمل الاستفسارات اللازمة.

- وهو عبارة عن برنامج رسومي يعمل تحت بيئة Windows الرسومية و يحتوي هذا البرنامج على مجموعة متنوعة من الكائنات التي يمكن استخدامها لعرض المعلومات و إدارتها مثل الجداول و النماذج و التقارير و الاستعلامات.

مميزات مايكروسوفت أكسس :

١. جمع جميع كائنات القاعدة في ملف واحد يأخذ الامتداد accdb وهذا ولاشك أسهل في التعامل مع القاعدة و إن كان قد يمثل خطورة على القاعدة من جهة أن تلف هذا الملف يتلف معه كل كائنات القاعدة .
٢. استيراد و تصدير أنواع مختلفة من البيانات إلى برامج مجموعة الأوفس أو إلى قواعد و برامج أخرى .
٣. تعدد درجات الأمان في القاعدة و تعدد المستخدمين .
٤. إمكانية وضع القاعدة على شبكة اتصالات داخلية و تشغيلها من عدة مستخدمين في آن واحد .
٥. وجود خصائص و طرق تمكن المستخدم من التحكم الكامل في القاعدة و بياناتها ومنع تغيير تصميمها .

مايكروسوفت أكسس و قواعد البيانات العلائقية :

-يطلق على قواعد بيانات ميكروسوفت أكسس اسم قواعد البيانات العلائقية و يقصد بها قواعد البيانات التي تكون الجداول فيها مترابطة بينها بعلاقات في حقل واحد أو أكثر .
-الهدف الأساسي من ربط الجداول هو منع تكرار البيانات و الحد من مساحات التخزين الضائعة و الرفع من كفاءة قاعدة البيانات .

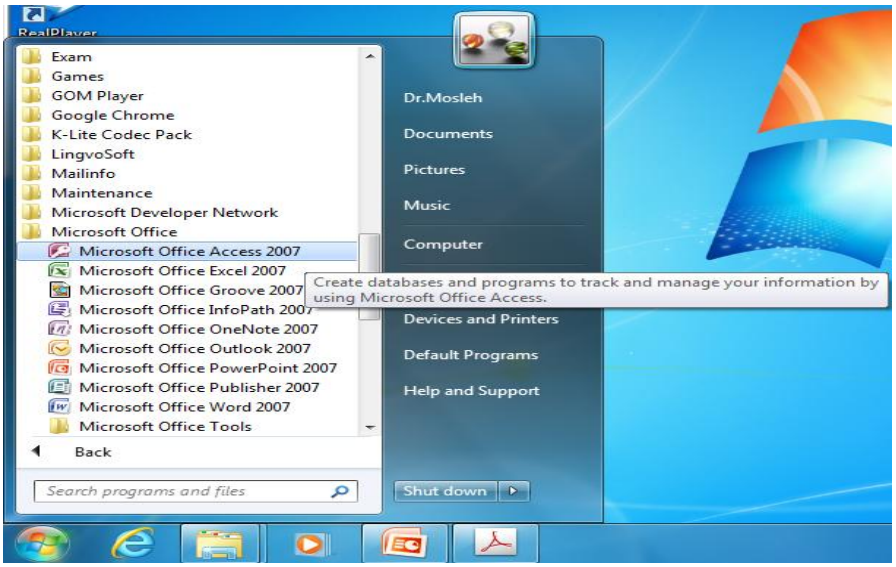
الكائنات المستخدمة في مايكروسوفت أكسس ٢٠٠٧ :

وضعت ميكروسوفت في هذا البرنامج كائنات تساعد المستخدم لإدخال البيانات و استخراجها من القاعدة و طباعتها ، منها :

١. **الجداول :** وهي مكان تخزين البيانات في القاعدة ، و تتكون الجداول من حقول (أعمدة) و سجلات (صفوف) .
٢. **استعلامات :** وهي كما يتضح من اسمها استعلام عن بيانات معينة في القاعدة تنطبق عليها معايير محددة أو كائنات لتنفيذ عمليات على البيانات في الجداول كحذف سجلات أو تحديثها أو إنشاء الجداول أو إلحاق سجلات بها
٣. **النماذج :** وهي مكان تسجيل البيانات التي ترغب في حفظها في الجدول ، و تحريرها و عرضها على شاشة المستخدم
٤. **التقارير :** وهي كائنات عرض و طباعة البيانات بأشكال و طرق و تنسيقات منوعة .

فتح برنامج أكسس ٢٠٠٧ :

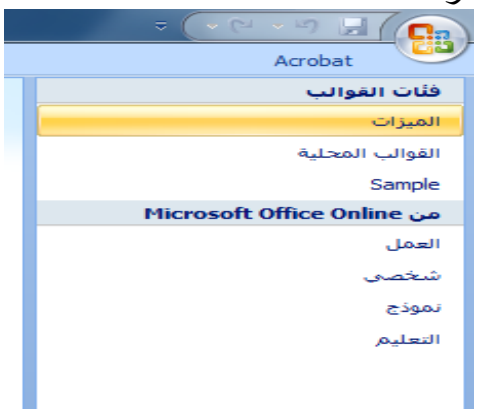
يتم فتح برنامج أكسس عن طريق الخطوات :
١. Start
٢. All Programs
٣. Microsoft Office
٤. Microsoft Office Access 2007



تظهر لنا الشاشة التالية:

إنشاء ملف قاعدة بيانات :

يمكنك إنشاء قاعدة بيانات من قوالب جاهزة



للإستخدام ، وهي نوعان :

١) محلية و Sample :

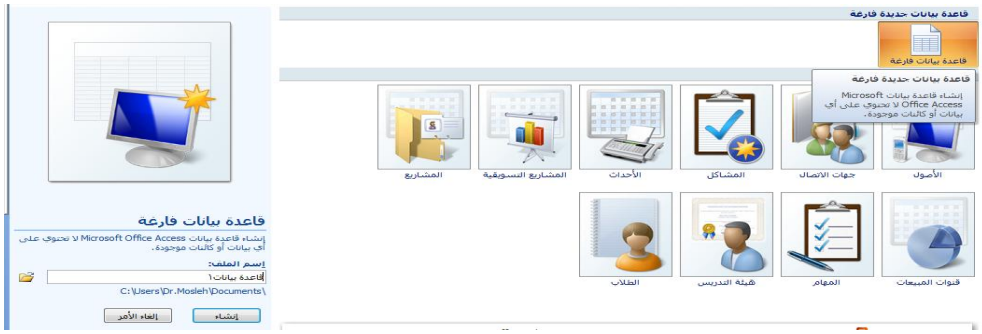
هي قوالب نموذجية و جاهزة للاستخدام ، موجودة على الحاسب الذي تعمل عليه وليس بالضرورة وجود رابط إنترنت لتحميلها

٢) Microsoft office Online :

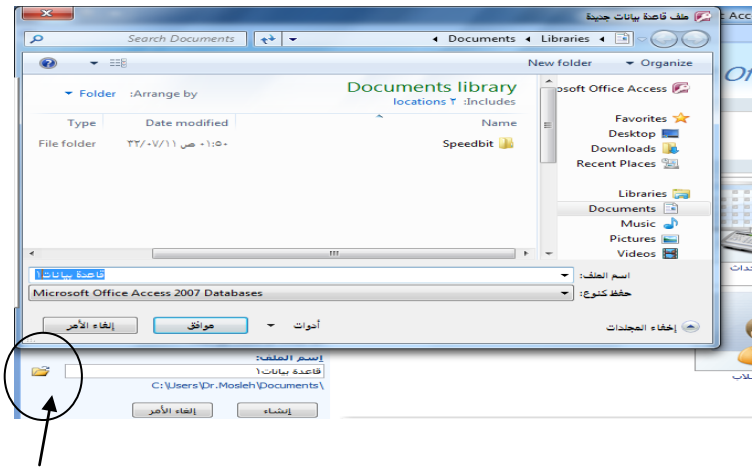
هي قوالب نموذجية و جاهزة للاستخدام ، متوفرة على موقع شركة مايكروسوفت و يجب وجود رابط إنترنت لتحميلها

- يمكنك إنشاء قاعدة بيانات جديدة عن طريق اختيار قاعدة بيانات فارغة

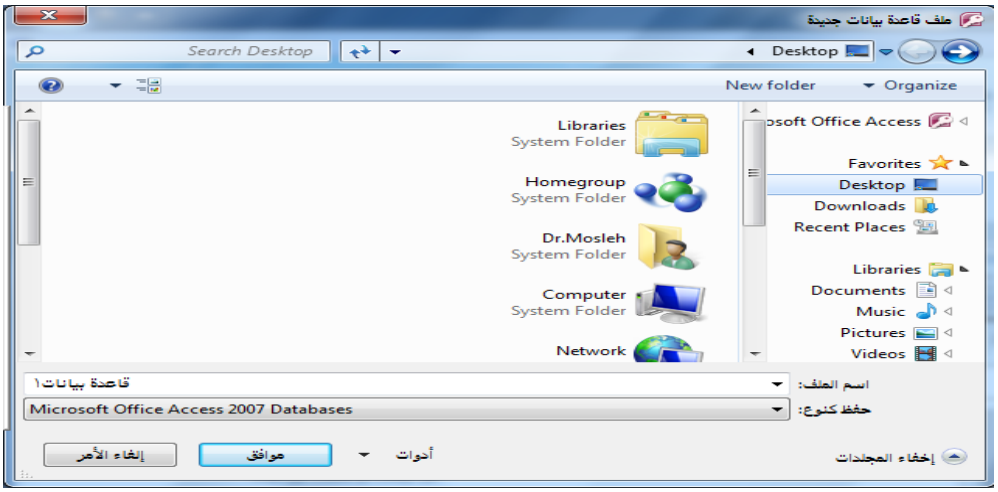
- عند إختيار قاعدة بيانات فارغة ، يطلب منك تحديد إسم و موقع ملف قاعدة البيانات



- لتغيير موقع الملف نضغط عل الرابط كما في الصورة ، ليظهر لنا صندوق حوار تخزين الملف

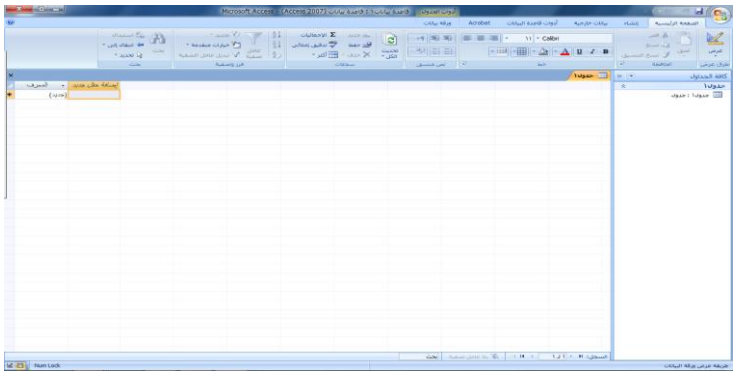


- تم اختيار موقع الملف ليكون على سطح المكتب ، و تم الإحتفاظ بإسم الملف المعياري كما هو "قاعدة بيانات ١ accdb " ،علما بأنه يمكننا تغيير إسم الملف كا نريد، ثم نضغط موافق



- نلاحظ تغير موقع الملف ، ثم نضغط زر إنشاء ، لإتمام عملية الإنشاء

- أصبح لدينا الآن ملف قاعدة بيانات فارغ جاهز للاستخدام

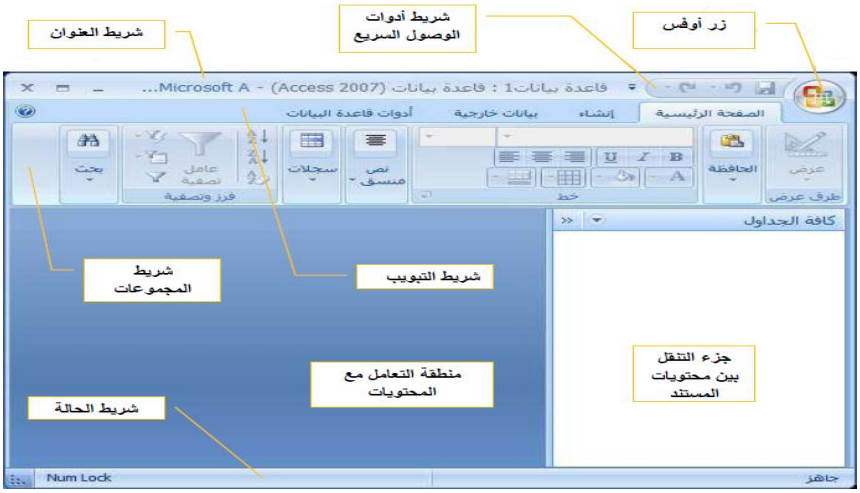


ملاحظة : خلافا لبرامج مايكروسوفت الأخرى ، يتم طلب تخزين ملف قاعدة البيانات أكسس ، و السبب أن هذا الملف يحتوي في داخله العديد من الكائنات الأخرى التي يجب أن تخزن أولا بأول ، مثل الجداول و الإستعلامات وعليه يجب أن نهيئ الملف لتخزين هذه الكائنات ، وذلك بحجز مكان لها على ذاكرة الجهاز الثانوية (القرص الصلب مثلا) و عملية الحجز تتم بتخزين الملف الذي يحتوي هذه الكائنات أي ملف قاعدة البيانات.

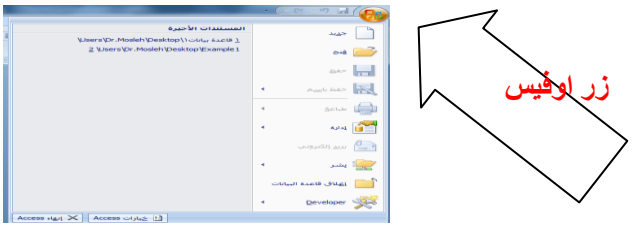
فتح ملف قاعدة بيانات قديم :

- يمكنك فتح ملف أكسس، إما بالضغط عل فتح من زر أوفيس ، أو بالضغط على الملف المطلوب إذا كان ظاهرا في المستندات الأخيرة

التعرف على بيئة أكسس ٢٠٠٧ :

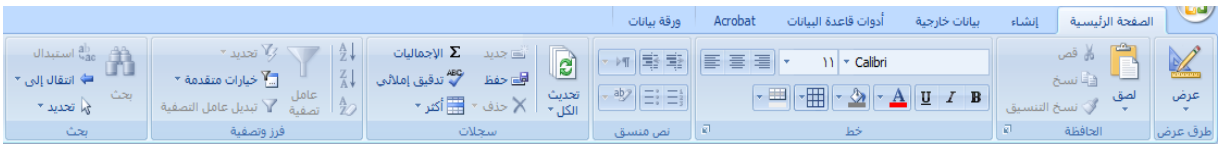


- **زر أوفيس :** يستخدم لفتح قائمة ملف المستخدمة في الإصدارات السابقة ، و التي تحتوي على أوامر الملفات من فتح و إنشاء و حفظ وطباعة ...إلخ



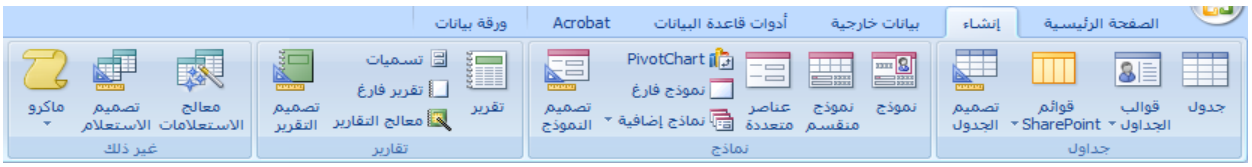
- تتميز بيئة أوفيس ٢٠٠٧ بإستخدامها لشريط التبويب بدلا من القوائم ، وهو شريط يحتوي على تبويبات متخصصة ، وفي كل تبويب مجموعات خاصة بهذا التبويب ، ولدينا التبويبات الرئيسية التالية:
  - تبويب الصفحة الرئيسية
  - تبويب إنشاء
  - تبويب بيانات خارجية
  - تبويب أدوات قواعد البيانات

- وهناك تبويبات ثانوية تظهر عند الحاجة،أو تظهر عند إستخدام كائن معين



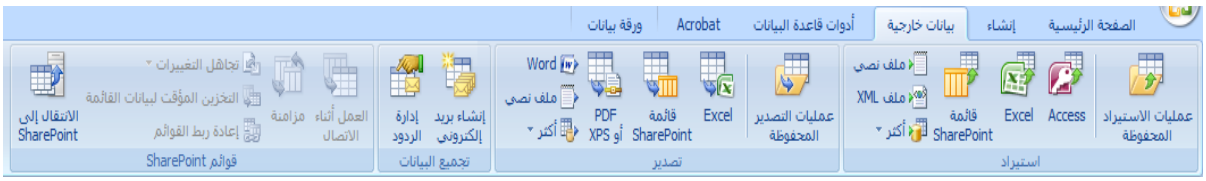
- **تبويب الصفحة الرئيسية : يستخدم في إنجاز مهام مثل :**

- تحديد طريقة عرض أخرى.
- نسخ و لصق من الحافظة.
- تعيين خصائص الخط الحالية.
- تعيين محاذاة الخط الحالية.
- تطبيق تنسيق النص المنسق على حقل "مذكرة".
- العمل مع السجلات (تتضمن الأوامر "تحديث" و"جديد" و"حفظ" و"حذف" و"الإجماليات" و"تدقيق إملائي" و"أكثر").
- فرز السجلات و تصفيتها.
- البحث عن السجلات.



- **تبويب إنشاء :** وهو التبويب الأكثر أهمية و يستخدم في إنجاز مهام مثل :

- إنشاء جدول فارغ جديد.
- إنشاء جدول جديد باستخدام قالب الجدول.
- إنشاء جدول فارغ جديد في طريقة عرض "التصميم".
- إنشاء نموذج جديد يستند إلى جدول أو استعلام نشط.
- إنشاء pivot table أو مخطط جديد.
- إنشاء تقرير جديد يستند إلى جدول أو استعلام نشط.
- إنشاء استعلام أو ماكرو أو وحدة نمطية أو فئة وحدة نمطية جديدة.



- **تبويب بيانات خارجية: يستخدم في إنجاز مهام مثل :**

- استيراد بيانات خارجية أو الارتباط بها.
- تصدير بيانات.
- تجميع البيانات و تحديثها باستخدام البريد الإلكتروني.
- العمل مع قوائم SharePoint غير المتصلة.
- إنشاء عمليات الاستيراد و التصدير المحفوظة.
- نقل كافة أجزاء قاعدة البيانات أو جزء منها إلى موقع SharePoint جديد أو موجود.

IbtiHAlinO

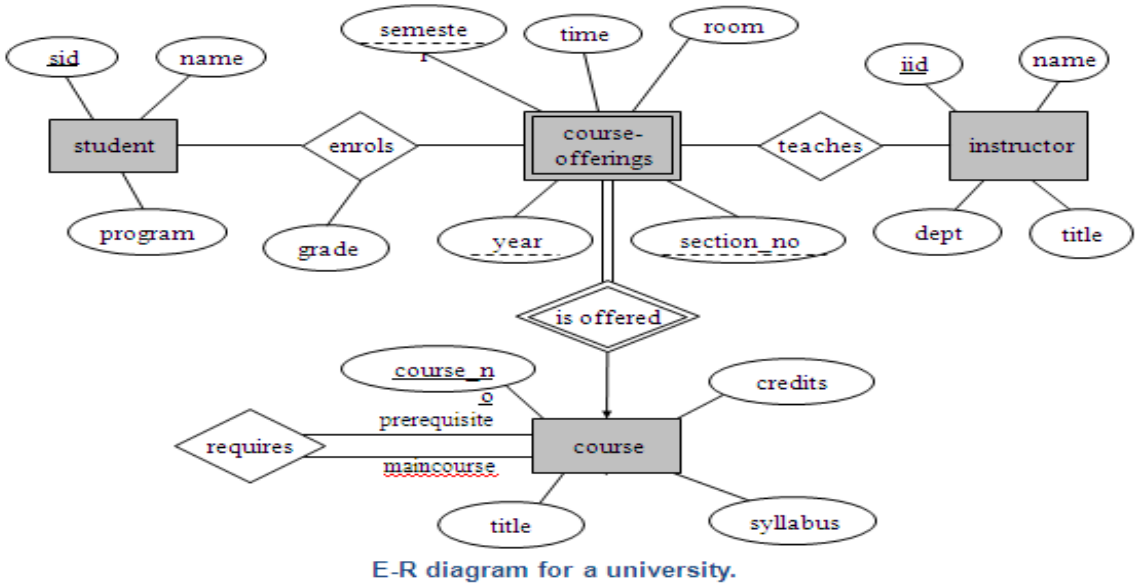
تبويب أدوات قاعدة البيانات:يستخدم في إنجاز مهام مثل

- تشغيل محرر Visual Basic أو ماكرو.
- إنشاء علاقات جدول وعرضها.
- إظهار/إخفاء تبعيات الكائنات أو ورقة الخصائص.
- تشغيل توثيق قاعدة البيانات أو تحليل الأداء.
- نقل البيانات إلى Microsoft SQL Serverأو قاعدة بيانات) Accessالجداول فقط).
- تشغيل "إدارة الجداول المرتبطة" إدارة وظائف Accessالإضافية.
- إنشاء وحدة نمطية Visual Basic for Applications (VBA.)

جزء التنقل : وهو الجزء الذي يحتوي كل الكائنات التي تحتويها قاعدة البيانات ، وبشكل رئيسي تحتوي على :

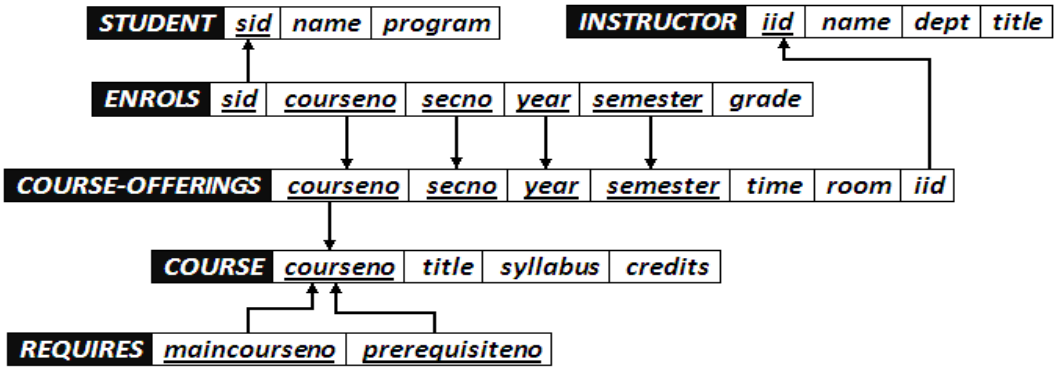
١. الجداول
٢. الاستعلامات
٣. النماذج
٤. التقارير

- تذكير بمثال الجامعة : أخذنا في محاضرة سابقة مخطط الكيان العلاقة التالي:



E-R diagram for a university.

- ونتج عن تحويل مخطط الكيان العلاقة ، مخطط قواعد البيانات التالي:



- **سنقوم في هذه المحاضرة، وما يليها باستخدام هذا المثال للشرح و التطبيق**

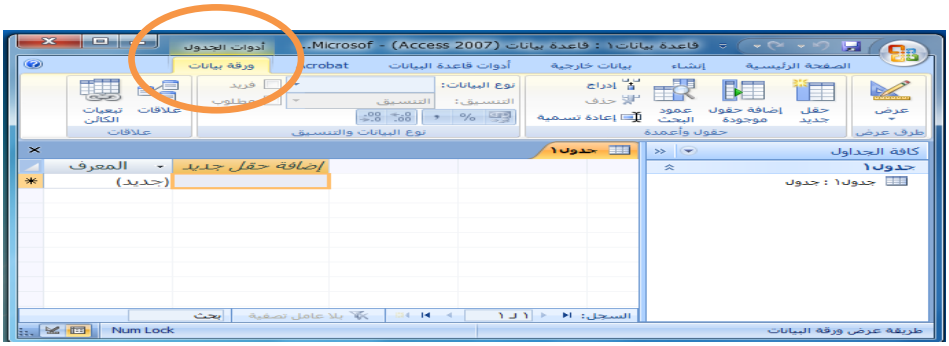
ما هو الجدول؟

- يعتبر الجدول هو الكائن الأساسي في بناء قاعدة البيانات ، حيث أنه المخزن الحقيقي للبيانات.
- يحتوي الجدول على بيانات حول موضوع معين مثل الطلاب المحاضرون المقررات و يتكون من صفوف تسمى سجلات ، ومن أعمدة تسمى حقول.
- يحتوي السجل في الجدول على معلومات متعددة حول عنصر معين ، كان يكون سجل معلومات عن الطالب أو سجل معلومات عن المقرر.
- أما الحقل فهو عبارة عن معلومة من نوع معين يمكن سردها لأكثر من عنصر ، بحيث تكون معلومات الحقل متجانسة في النوع.
- و بالتالي يتكون السجل الواحد من عدد من الحقول التي تصف معلومات لشخص أو شيء ما.
- يمكن أن تحتوي قاعدة البيانات على جدول فأكثر (ملف واحد يحتوي عدة جداول)

طرق إنشاء الجدول :

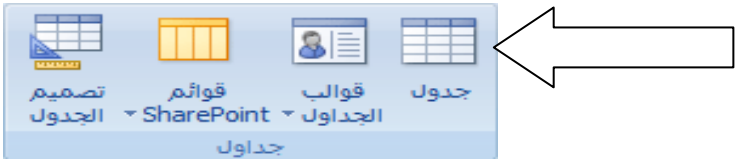
يمكن إنشاء الجدول بأكثر من طريقة:

١. عند إنشاء ملف قاعدة بيانات جديد، يتم إنشاء جدول فارغ من قبل برنامج الأكسس



يتم إدراج جدول جديد في قاعدة البيانات و يتم فتحه في طريقة عرض "ورقة البيانات".

٢. عند إنشاء فتح ملف قاعدة بيانات مخزن سابقا، في علامة التبويب إنشاء، في المجموعة جداول، انقر فوق جدول



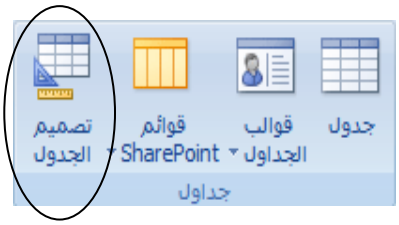
يتم إدراج جدول جديد في قاعدة البيانات ويتم فتحه في طريقة عرض "ورقة البيانات".

٣. إنشاء جدول استناداً إلى قالب جدول، لإنشاء جداول "جهات الاتصال" أو "المهام" أو "المشاكل" أو "الأحداث" أو "الأصول" ربما ترغب بالبدء بقوالب الجداول التي تأتي مع Office Access 2007.وذلك في علامة التبويب إنشاء، في المجموعة جداول، انقر فوق قوالب الجدول ثم حدد واحدًا من القوالب المتوفرة من القائمة(قم بتجربة القوالب الموجودة)

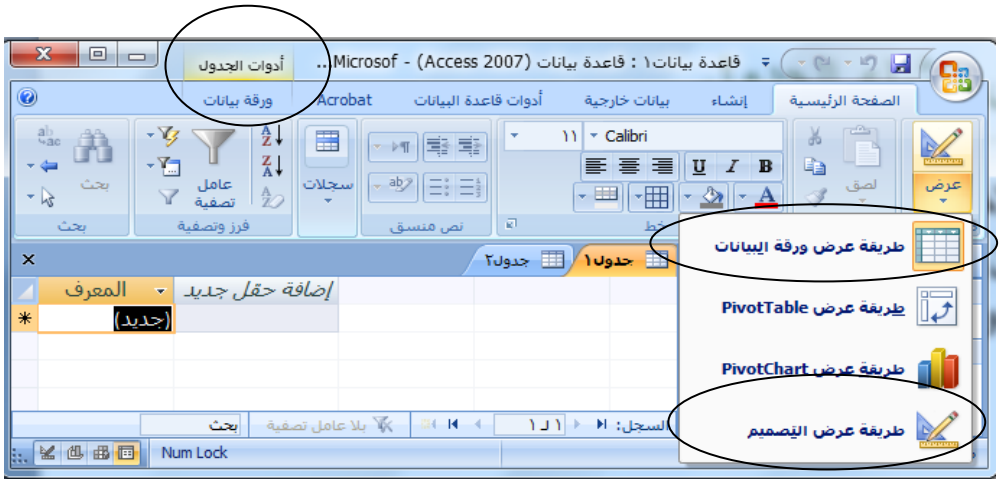
٤. إنشاء جدول جديد بواسطة الاستيراد أو الارتباط ببيانات خارجية . وذلك ضمن علامة التبويب بيانات خارجية في المجموعة استيراد ، انقر فوق أحد مصادر البيانات المتاحة.

اتبع الإرشادات الموجودة في مربعات الحوار.
ينشئ Access الجدول الجديد و يعرضه في "جزء التنقل".

٥. إنشاء جدول جديد بواسطة تصميم الجداول ، وهي الطريقة الأكثر إستخداما ، لأنها تعطيك الفرصة لبناء هيكل الجدول ومن ثم إدخال البيانات ، و ذلك عن طريق التبويب إنشاء مجموع جداول.



- **ملاحظة١:** عند إنشاء جدول بالطريقة الأولى و الثانية يمكنك إدخال البيانات مباشرة ، و يقوم برنامج الأكسس ببناء الهيكل للجدول.
- **ملاحظة٢:** عند بناء الجدول بالطريقة الخامسة ، يقوم المستخدم ببناء هيكل الجدول ومن ثم الإنتقال لإدخال البيانات ، وهو الأفضل
- **ملاحظة ٣:** يمكن التنقل بين الطريقتين بكل سهولة و يسر عن طريق تبويب ورقة بيانات الذي يظهر في حالة إنشاء جدول ، حيث يمكن التبديل بين التصميم و ورقة البيانات



- التغيير بعد التحويل إلى وضع التصميم

أنواع البيانات التي يمكن أن يبني منها الجدول :

يجب التعامل مع نوع بيانات الحقل على أنها مجموعة كفاءات يتم تطبيقها على كافة القيم المتضمنة في الحقل و تساعد في تحديد نوع بيانات تلك القيم . فعلى سبيل المثال، قد تتضمن القيم المخزنة في حقل "نصي" أحرفًا و أرقامًا و مجموعة محدودة من علامات الترقيم فقط . علاوة على ذلك ، قد يبلغ الحد الأقصى لعدد الأحرف التي يحتوي عليها الحقل "النصي" ٢٥٥ حرفًا.

هناك عشرة أنواع مختلفة من البيانات في Access :

١. مرفق عبارة عن الملفات ، مثل الصور الرقمية. و يمكن إرفاق ملفات متعددة لكل سجل . ولا يتوفر هذا النوع من البيانات في الإصدارات السابقة من .Access
٢. ترقيم تلقائي عبارة عن الأرقام التي يتم إنشاؤها تلقائيًا لكل سجل.
٣. عملة عبارة عن القيم المالية.
٤. التاريخ / الوقت عبارة عن التواريخ و الأرقام .
٥. ارتباط تشعبي عبارة عن الارتباطات التشعبية ، مثل عناوين البريد الإلكتروني.
٦. مذكرة عبارة عن مجموعات نصية طويلة و نصوص تستخدم تنسيق نصي و سيكون الاستخدام الفعلي لحقل "المذكرة" وصفًا مفصلاً للمنتج.
٧. رقم عبارة عن القيم الرقمية ، مثل المسافات. لاحظ وجود نوع منفصل لبيانات كل عملة.
٨. كائن OLE(كائنات OLE) كائن :OLE كائن يدعم بروتوكول OLE لارتباطه و تضمينه . يمكن أن يرتبط كائن OLE لملقم) OLE على سبيل المثال ، صورة Windows أو جدول بيانات Microsoft Excel أو تضمينه في حقل أو نموذج أو تقرير) ، مثل مستندات .Word
٩. نص عبارة عن قيم أبجدية رقمية صغيرة ، مثل الاسم الأخير أو عنوان الشارع.
١٠.موافق / غير موافق قيم منطقية

<u>تفصيل بعض أنواع البيانات التي يمكن أن يبني منها الجدول :</u>

١. مرفق :

-الغرض يمكنك استخدام حقل المرفق لإرفاق العديد من الملفات بداية من الصور و حتى السجلات.
-من المفترض أنه لديك قاعدة بيانات لجهات الاتصال الخاصة بالوظائف. يمكنك استخدام حقل المرفق لإرفاق صورة لكل جهة اتصال ، كما يمكنك إرفاق سيرة ذاتية أو أكثر لجهة اتصال بنفس الحقل في هذا السجل.
-بالنسبة لبعض أنواع الملفات ، يقوم Accessبضغط كل مرفق بمجرد إضافته.
-يمكنك إرفاق العديد من أنواع الملفات المختلفة إلى السجل ، لكن قد يتم حظر بعض أنواع الملفات التي قد تشكل مخاطر أمنية و كقاعدة عامة ، يمكنك إرفاق أي ملف تم إنشاؤه باستخدام أحد برامج نظام Microsoft Office 2007 كما يمكنك إرفاق ملفات السجلات log والملفات النصية text أو txt و الملفات المضغوطة zip

٢. رقم :

<u>يتم تحديد حجم الحقل لنوع البيانات رقم عن طريق تخصيص حجم الحقل إلى أحد الخيارات التالية :</u>
- بايت - يُستخدم للأعداد الصحيحة التي تتراوح من ٠ إلى ٢٥٥. حيث إن التخزين المطلوب هو ١ بايت.
- عدد صحيح - يُستخدم للأعداد الصحيحة التي تتراوح من ٠ إلى ٢٥٥ حيث إن التخزين المطلوب هو ٢ بايت.
- عدد صحيح طويل - يُستخدم للأعداد الصحيحة التي تتراوح من - ٢،١٤٧،٤٨٣،٦٤٨ إلى ٢،١٤٧،٤٨٣،٦٤٧. حيث إن التخزين المطلوب هو ٤ بايت.
- يُستخدم العدد المفرد لقيم الفاصلة العائمة الرقمية التي تتراوح من -٣,٤ * ١٠^٣٨ إلى ٣,٤ * ١٠^٣٨ و إلى ما يصل إلى ٧ أرقام رئيسية. حيث إن التخزين المطلوب هو ٤ بايت.
- يُستخدم العدد المزدوج لقيم الفاصلة العائمة الرقمية التي تتراوح من -١,٧٩٧ * ١٠^٣٠٨إلى +١,٧٩٧ * ١٠^٣٠٨ و إلى ما يصل إلى ١٥ رقمًا صحيحًا . حيث إن التخزين المطلوب هو ٨ بايت.
- يُستخدم معرف النسخ المتماثلة لتخزين المعرف الفريد العمومي المطلوب للنسخ المتماثلة. حيث إن التخزين المطلوب هو ١٦ بايت. لاحظ أنه لا يتم اعتماد النسخ المتماثلة باستخدام تنسيق الملف . .accdb
- يُستخدم الرقم العشري للقيم الرقمية التي تتراوح من - ٩,٩٩٩ * ١٠^٢٧ إلى +٩,٩٩٩ * ١٠^٢٧، حيث إن التخزين المطلوب هو ١٢ بايت.

-<u>ملاحظة :</u> للحصول على أفضل أداء ، يجب دومًا تحديد أقل حجم حقل مناسب

٣- الوقت/التاريخ :

يمكن اختيار تنسيق معين خاص بالتاريخ من ضمن التنسيقات التالية :

التاريخ العام لن يتم عرض الوقت بشكل افتراضي إذا كانت القيمة تاريخًا فقط، ؛ كما أنه لن يتم عرض التاريخ إذا كانت القيمة وقتًا فقط . يجمع هذا الإعداد بين الإعدادين " التاريخ القصير " و " الوقت الطويل ".

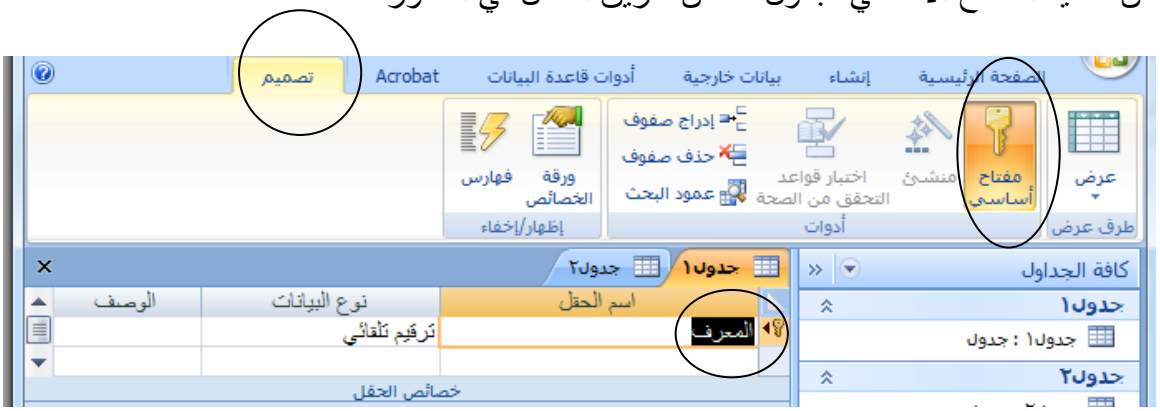
أمثلة :

- ٠٧/٣/٤
- ٠٥:٣٤:٠٠ م
- ٠٧/٣/٤ ٠٥:٣٤:٠٠ م

-يعد التاريخ الطويل هو نفس إعداد " التاريخ الطويل " الموجود في الإعدادات الإقليمية في Windows مثال: السبت ، ٣ ابريل ، ٢٠٠٧.
-يعرض التاريخ المتوسط التاريخ كـ dd-mmm-yyyy مثال: ٣-ابريل-٢٠٠٧.
-يعتبر التاريخ القصير هو نفس إعداد " التاريخ القصير " الموجود في الإعدادات الإقليمية في Windows مثال: ٠٧/٣/٤
-تحذير يفترض بإعداد "التاريخ القصير" أن تكون التواريخ فيما بين ٠٠/١/١ و ٢٩/٣١/١٢ من تواريخ القرن الحادي و العشرين (أي أن السنين يُفترض أن تكون من ٢٠٠٠ إلى ٢٠٢٩). و يفترض أن تكون التواريخ فيما بين ٣٠/١/١ و٩٩/٣١/١٢من تواريخ القرن العشرين (أي أن السنين يُفترض أن تكون من ١٩٣٠ إلى ١٩٩٩).
-يعد التاريخ الطويل هو نفس الإعداد الموجود في علامة التبويب وقت من الإعدادات الإقليمية في Windows مثال: ٥:٣٤:٢٣ م.
-يعرض الوقت المتوسط الوقت بالساعات و الدقائق مفصولة بحرف فاصل زمني. مثال : ٥:٣٤ م.
-يعرض الوقت القصير الوقت بالساعات و الدقائق مفصولة بفاصل زمني باستخدام تنسيق ٢٤ ساعة. مثال : ١٧:٣٤

المفتاح الأساسي :

- المفتاح الأساسي هو ذلك الحقل (أو مجموعة الحقول) الذي يمكن عن طريقه تمييز سجلات الجدول الواحد عن بعضها البعض .
- يمكن تحديد المفتاح الأساسي لجدول ما عن طريق الشكل في الصورة



- يعتبر المفتاح الأساسي (Primary Key) جزءا رئيسا في ربط جداول قاعدة البيانات بعضها ببعض.

بناء جداول الجامعة :

في مثال الجامعة لدينا عدد ٦ جداول هي :
- جدول الطالب (Student) - جدول المدرس (Instructor) - جدول الطالب الفصلي (Enrols)
- جدول الشعب الفصلي (Course-Offerings) - جدول المقررات (Course) - جدول المتطلب السابق (Requires)

جدول الطالب (Student):

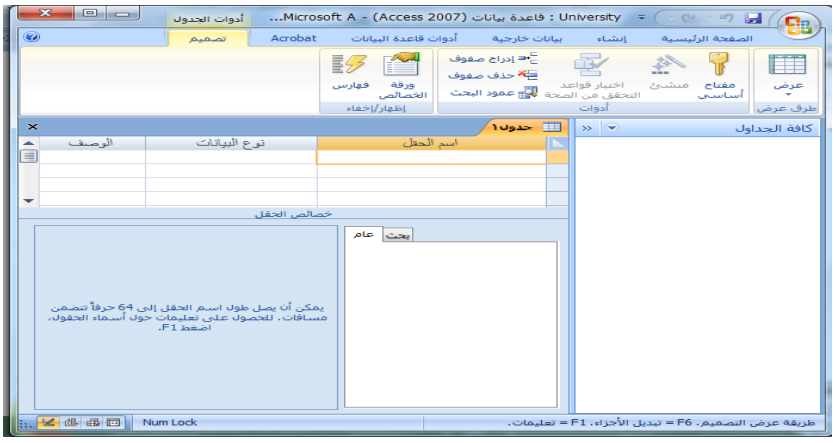
يتكون من الحقول التالية:

- رقم الطالب (رقم ، رقم صحيح طويل) ، مفتاح أساسي
- إسم الطالب (نص، ١٥ حرف)
- التخصص (نص، ٢٥ حرف)

خطوات إنشاء جدول الطالب :

- نقوم بإنشاء ملف جديد و نخزنه تحت إسم University على سطح المكتب.

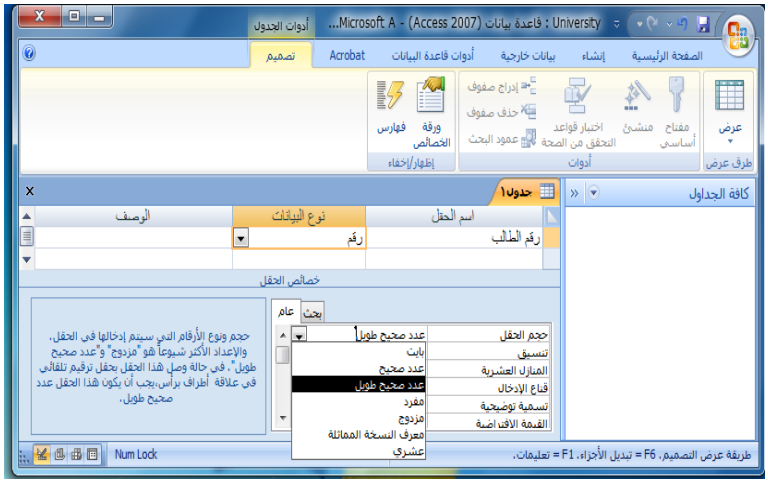
- نقوم بإنشاء جدول بإستخدام طريقة تصميم الجداول



- ندخل إسم الحقل الأول في خانة إسم الحقل

- نقوم بإدخال نوع بيانات الحقل في خانة نوع البيانات من ضمن الخيارات المتوفرة

- نقوم بإدخال حجم الحقل من بين الخيارات المعطاه

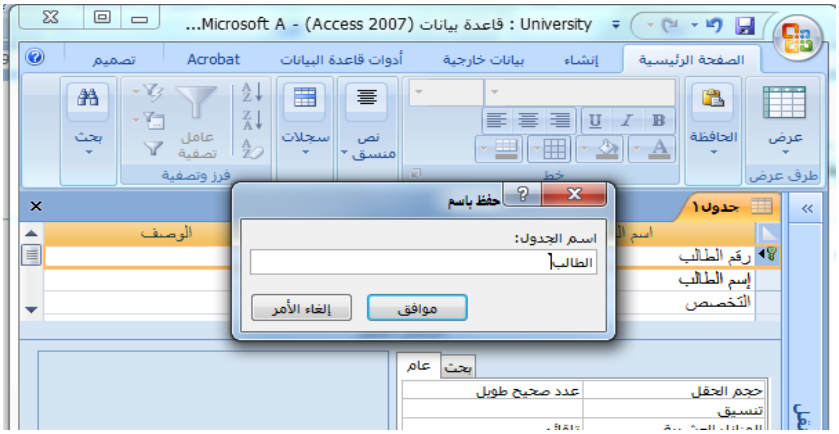


- نكرر نفس العملية لحقل إسم الطالب

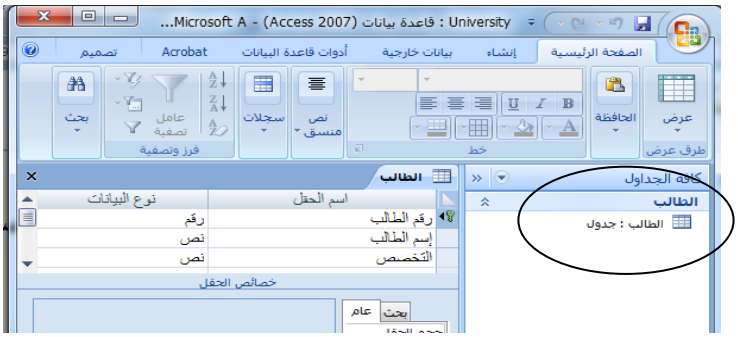
- نكرر نفس العملية لحقل التخصص

- نقوم بتحديد المفتاح الأساسي عن طريق إختيار الحقل ، ومن ثم الضغط على أيقونة مفتاح أساسي

- نقوم بتخزين الجدول إما بالضغط على شكل القرص المرن في شريط الوصول السريع أو من قائمة زر أوفيس نختار حفظ أو حفظ بإسم ، نكتب إسم الجدول و نضغط موافق



- بعد الحفظ يظهر الجدول في جزء التنقل

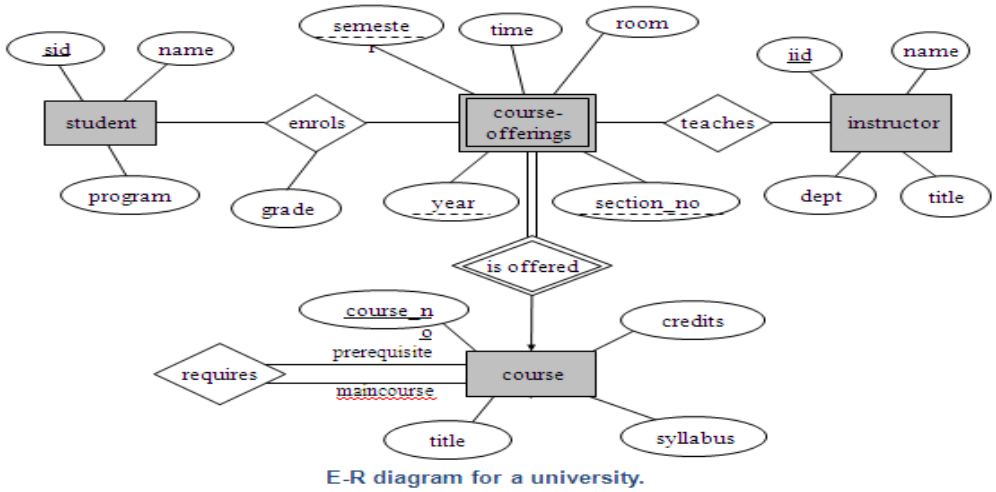


- ننتقل إلى نمط ورقة بيانات لإدخال بيانات الجدول ، ومن ثم ندخل البيانات سجلا بعد الآخر

- نقوم بإنشاء باقي الجداول بذات الطريقة (راجعوا جميع الرسومات أعلاه لأنه كررها نفسها)

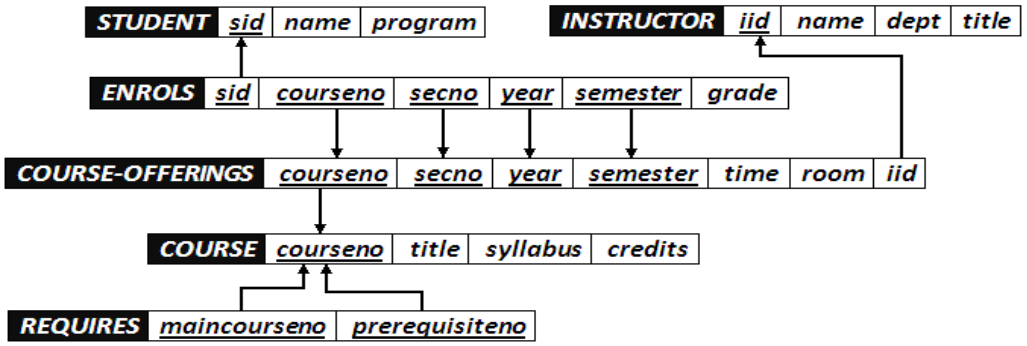
تذكير بمثال الجامعة :

أخذنا في محاضرة سابقة مخطط الكيان العلاقة التالي :



E-R diagram for a university.

و نتج عن تحويل مخطط الكيان العلاقة ، مخطط قواعد البيانات التالي :



حيث قمنا في المحاضرة السابقة ببناء الجداول المذكورة في هذا المخطط ، و نقوم في هذه المحاضرة ببناء العلاقات اللازمة

- ملاحظة (١) : للمضي قدما في هذه المحاضرة ، عليك مراجعة ما تم شرحه مسبقا عن العلاقات و أنواعها.
- ملاحظة (٢) : عليك مقارنة نتائج العلاقات في أكسس مع مخطط قواعد البيانات للجامعة لمعرفة مدى التطابق.

لماذا ننشئ علاقات الجداول ؟

هناك العديد من الأسباب التي تجعل من الضروري إنشاء علاقات جداول قبل إنشاء كائنات قاعدة بيانات أخرى ، مثل النماذج و الاستعلامات و التقارير :

- تؤثر علاقات الجداول على تصميمات الاستعلام ، للعمل مع سجلات من أكثر من جدول واحد ، يجب عليك غالبًا إنشاء استعلام يربط هذه الجداول . يعمل الاستعلام من خلال مطابقة القيم في حقل المفتاح الأساسي للجدول الأول بحقل مفتاح خارجي في الجدول الثاني.
- تؤثر علاقات الجداول على تصميمات النموذج و التقرير ، حيث يقدم أكسس اختيارات تستند إلى هذه العلاقات.
- علاقات الجداول هي الأساس حيث يمكنك تحسين التكامل المرجعي ليساعدك على منع السجلات الوحيدة في قاعدة البيانات (السجل الوحيد هو سجل يرجع إلى سجل آخر غير موجود)

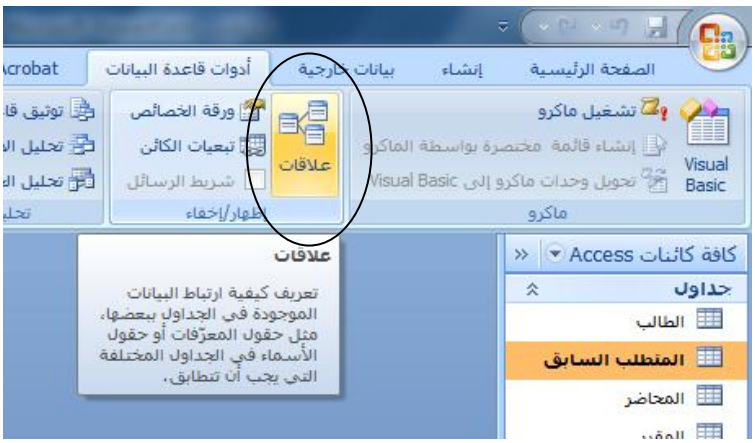
فهم التكامل المرجعي :

- عندما تصمم قاعدة بيانات ، فإنك تقسم المعلومات في جداول حسب الموضوع للحد من تكرار البيانات.
- الهدف من التكامل المرجعي هو منع السجلات الوحيدة و الحفاظ على المراجع متزامنة بحيث لا يحدث عدم توافقية في البيانات بين الجداول المختلفة.
- يمكنك فرض التكامل المرجعي عن طريق تمكينه لعلاقة جدول .بحيث يرفض أكسس أية عملية تمنع التكامل المرجعي لعلاقة الجدول.
- في حال احتجت لتغيير قيمة المفتاح الأساسي ، بناءا عليه يجب عليك تحديث قيمته في الجداول المرتبطة معه ، و يمكن ذلك عن طريق إختيار " تتالي تحديث الحقول المرتبطة " ، ليقوم أكسس بعملية التحديث بالنيابة عنك.
- في حال احتجت لحذف قيمة المفتاح الأساسي ، بناءا عليه يجب عليك حذف قيمته في الجداول المرتبطة معه ، و يمكن ذلك عن طريق إختيار " تتالي حذف السجلات المرتبطة "، ليقوم أكسس بعملية الحذف بالنيابة عنك.

عرض علاقات الجداول :

لعرض علاقات الجداول :

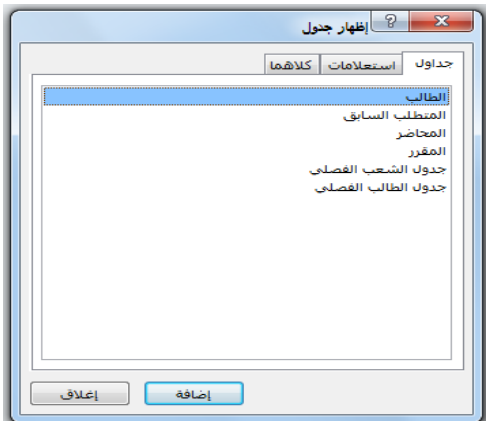
- انقر فوق علاقات ضمن علامة التبويب أدوات قاعدة البيانات. يفتح الإطار " علاقات " و يعرض أي علاقات موجودة . إذا لم يتم تعريف أي علاقات جداول بعد ، و كنت تفتح الإطار علاقات للمرة الأولى ، يطالبك أكسس بإضافة جدول أو استعلام إلى الإطار.



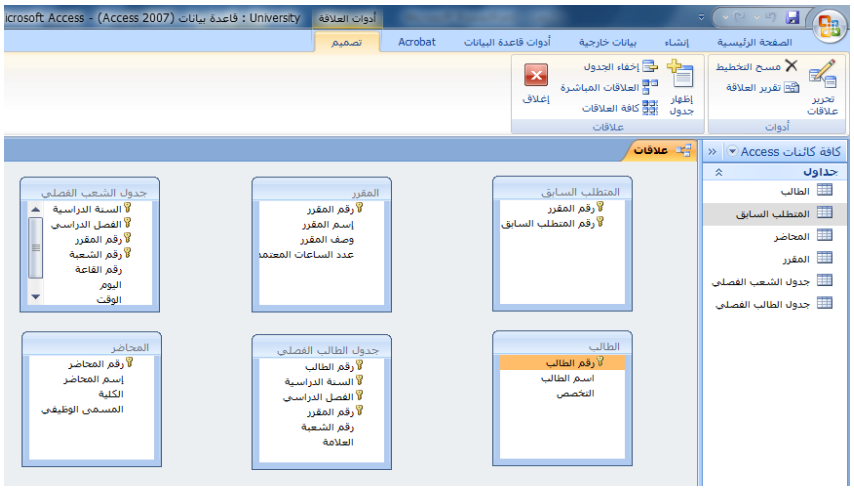
إنشاء علاقة جدول :

١. أفتح لوحة العلاقات
٢. إضغط بزر الفأرة الأيمن لإظهار الجداول

٣. أختر الجداول / الإستعلامات المطلوبة .



٤. قم بإنشاء العلاقة :

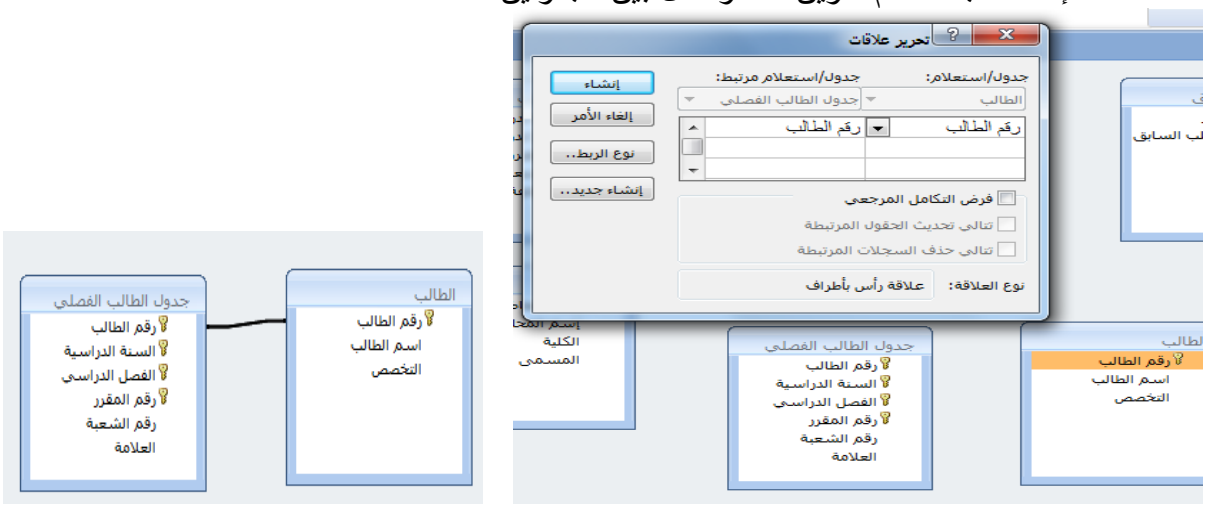


هناك أكثر من طريقة لإنشاء علاقة جدول بآخر، سنذكر منها أبسط طريقتين :

١- بإستخدام سحب و إفلات الفأرة ٢- بإستخدام قائمة تحديد علاقة

١. باستخدام سحب و إفلات الفأرة :

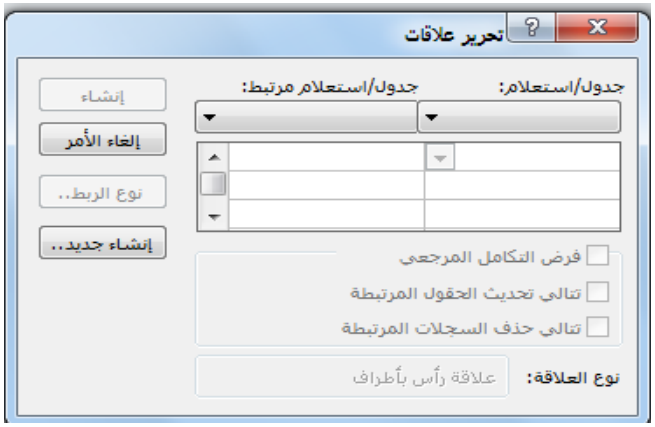
- اضغط على حقل المفتاح الأساسي في الجدول الأول ، بزر الفأرة الأيسر.
- حرك الفأرة مع إستمرار الضغط على زر الفأرة الأيسر باتجاه الجدول الثاني ، و بالتحديد المفتاح الخارجي المطلوب
- أترك زر الفأرة ، لتظهر لك شاشة تحرير العلاقات
- اضغط إنشاء لتجد أنه تم تكوين خط واصل بين الجدولين



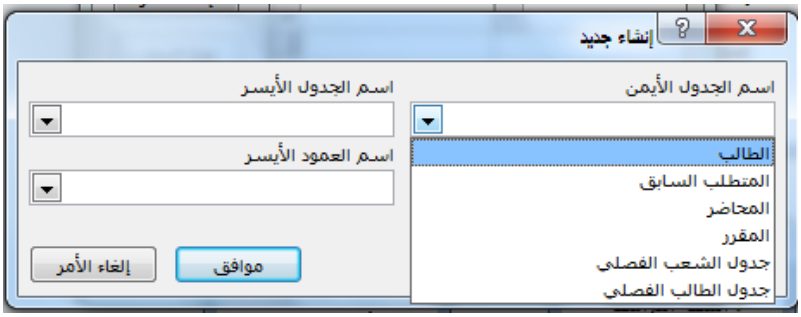
٢. باستخدام شاشة تحرير العلاقات

أ - اضغط على أيقونة تحرير علاقات

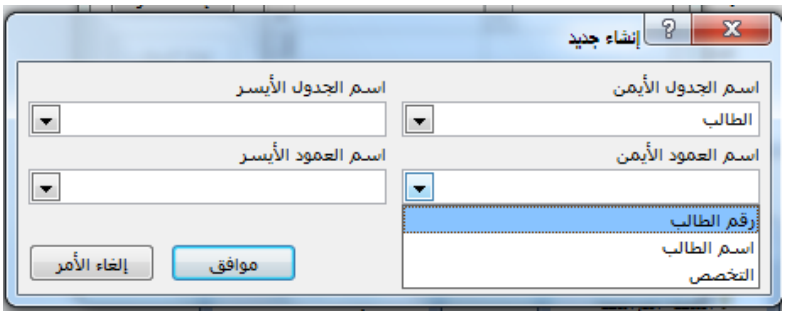
ب- تظهر لك شاشة تحرير علاقات ، إضغط إنشاء جديد



ج- في شاشة إنشاء جديد ، إختر إسم الجدول الأيمن

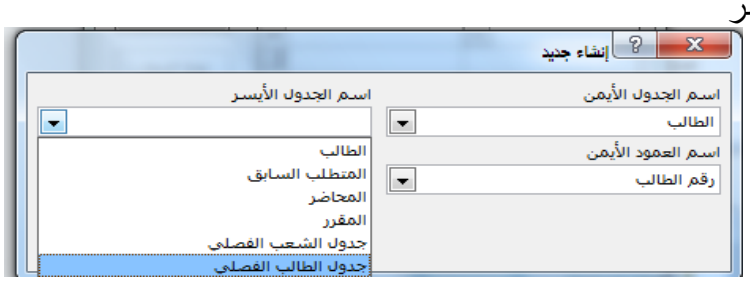


د- أختر أسم العمود الأيمن

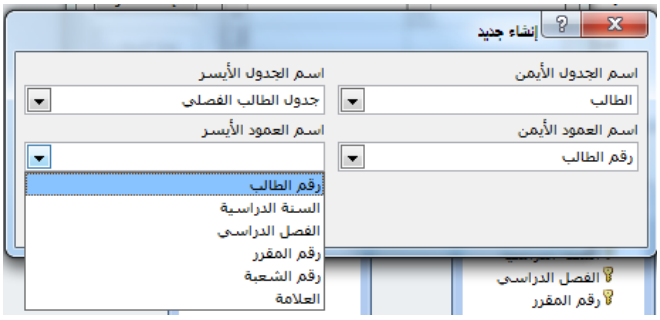


IbtiHAlinO

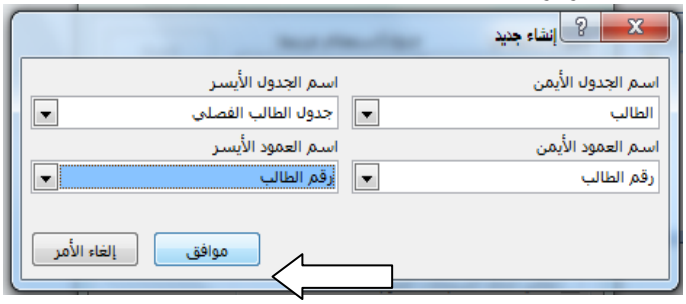
هـ - أختر أسم الجدول الأيسر



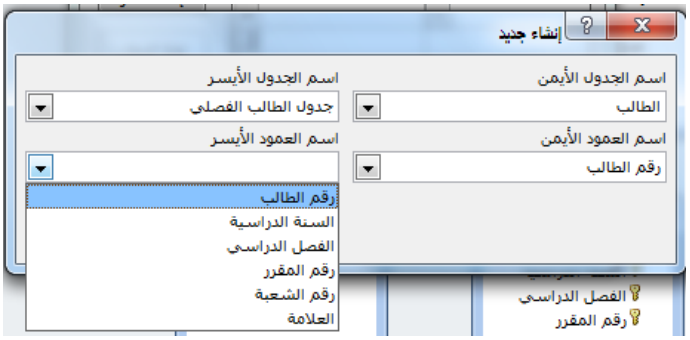
و - أختر إسم العمود الأيسر



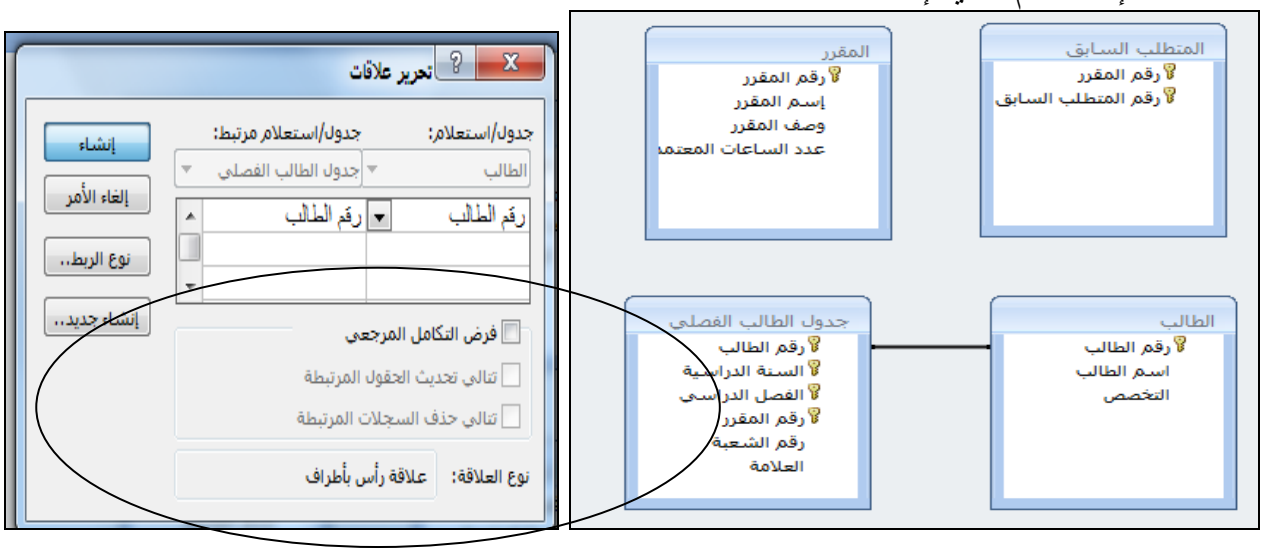
ز - أضغط موافق ، لتعود إلى شاشة تحرير علاقات



ح -اختر إسم العمود الأيسر



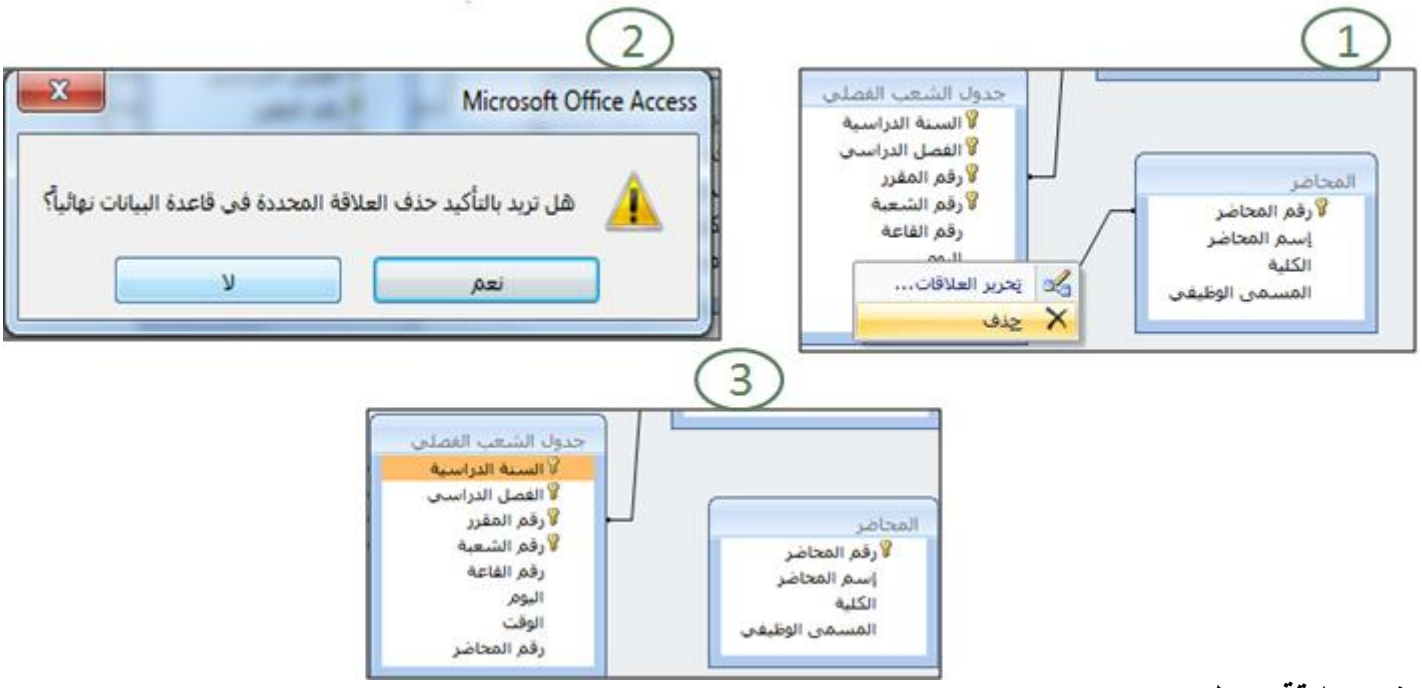
ط -أضغط إنشاء لتتم عملية إنشاء العلاقة



ملاحظة : هناك حالات تكون فيها العلاقة بين الجدولين على أكثر من حقل ، مثل علاقة جدول الشعب الفصلي بجدول الطالب الفصلي ، عندها يتم إختيار باقي الحقول في شاشة تحرير العلاقات بعد الإنتهاء من شاشة إنشاء جديد

حذف علاقة جدول :

- لحذف علاقة بين جدولين ، إضغط بزر الفأرة الأيمن على الخط الممثل للعلاقة بين الجدولين ، و إختر حذف

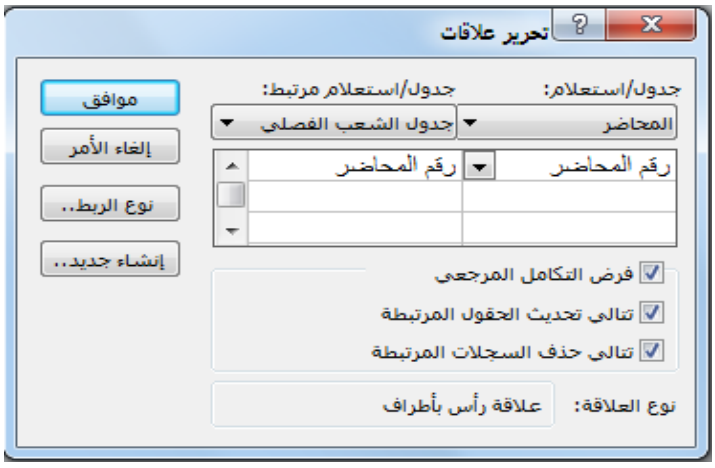


تغيير علاقة جدول :

- لإجراء تغيير على علاقة ما بين جدولين ، إضغط بزر الفأرة الأيمن على الخط الممثل للعلاقة بين الجدولين ، و إختر تحرير علاقات ، ومن شاشة تحرير علاقات نجري التغيير ، ثم نضغط موافق

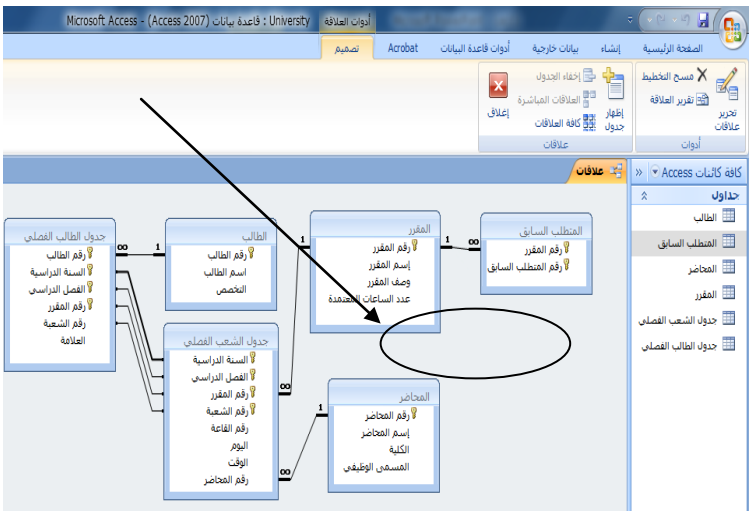
فرض التكامل المرجعي :

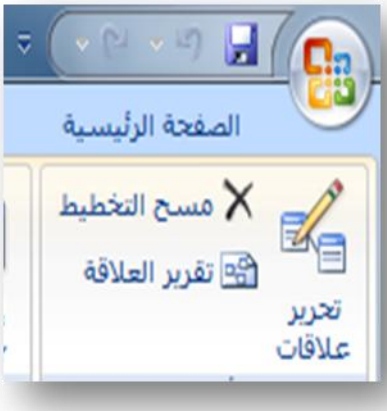
لفرض التكامل المرجعي على علاقة ما بين جدولين ، نختار شاشة تحرير العلاقات و نُفَعِّل خيار " فرض التكامل المرجعي "، وفي حالة رغبتنا بإعطاء المستخدم الفرصة للتعديل على قيمة الحقل الأساسي أو حذف قيمة حقل أساسي نفعل باقي الخيارات



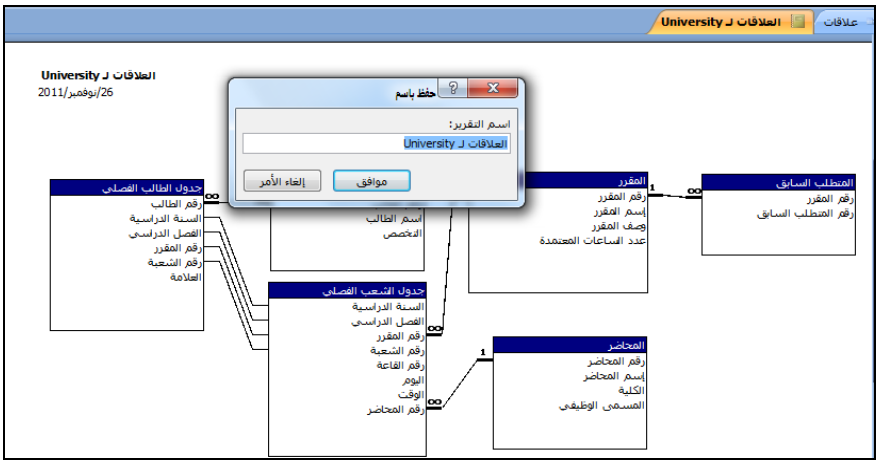
إنشاء تقرير العلاقات :

- لإنشاء تقرير بالعلاقات المنشأة ، إختر أيقونة تقرير العلاقة كما هو مبين في الشكل :

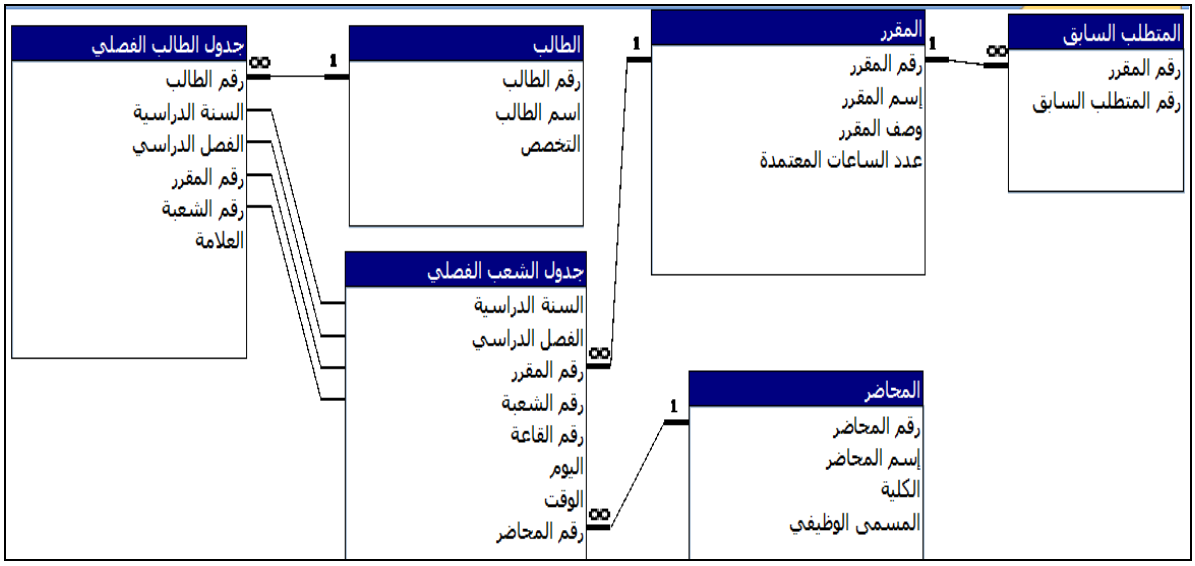


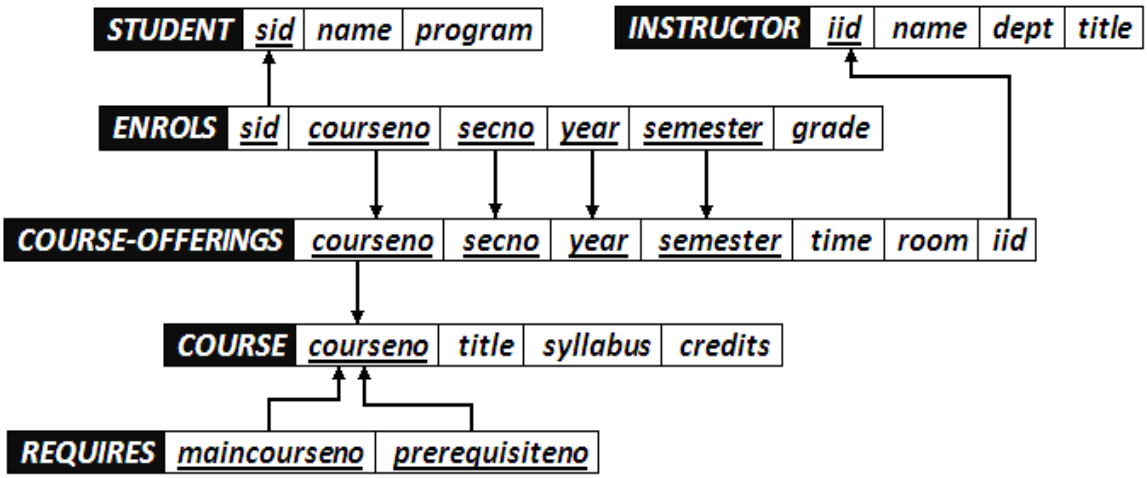


- يظهر لنا تقرير العلاقات أدناه ، و يمكن تسميته العلاقات لـ University



مقارنة مخطط قواعد البيانات مع تقرير العلاقة





إنشاء النماذج و التقارير:

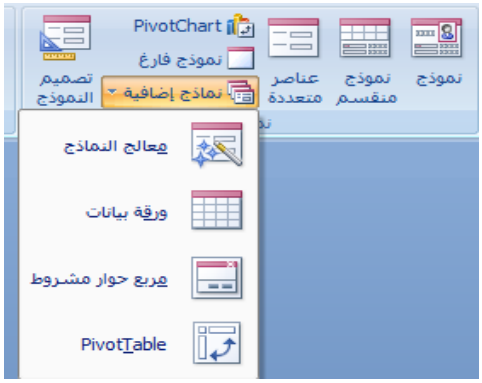
ما هو النموذج ؟

- النموذج عبارة عن كائن يمكنك من إنشاء لوحات رسومية تعرض على شاشة المستخدم ، ليتمكن من خلالها إدخال البيانات أو إسترجاعها أو تعديلها أو حذفها .
- كما يمكن عن طريق النماذج تفعيل بعض الإجراءات البرمجية .

طرق إنشاء النماذج :

- **هناك أكثر من طريقة لإنشاء النموذج ، وكلها متوفرة في تبويب إنشاء ، ضمن مجموعة نماذج**

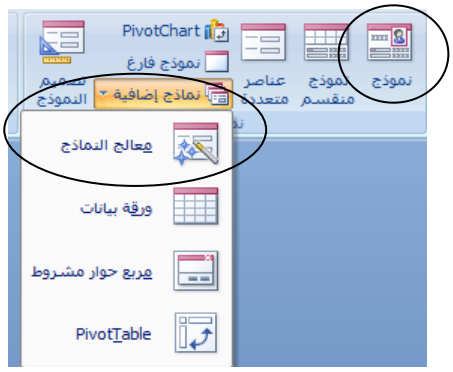
١. **نموذج :** لإنشاء نموذج يسمح لك بإدخال كل سجل على حدة



٢. **نموذج منقسم :** لإنشاء نموذج منقسم يقوم بإظهار ورقة بيانات في المقطع العلوي و نموذج في المقطع السفلي لإدخال معلومات حول السجل المحدد في ورقة البيانات
٣. **عناصر متعددة :** لإنشاء نموذج يقوم بعرض سجلات متعددة في ورقة بيانات مع إظهار سجل واحد في كل صف
٤. **Pivot Chart :** لعرض البيانات على شكل رسم بياني
٥. **نموذج فارغ :** لعرض نموذج فارغ يتم إضافة الحقول المطلوبة فيه عن طريق قائمة بالحقول المتوفرة
٦. **تصميم النماذج :** إنشاء نموذج جديد فارغ بطريقة عرض التصميم ، حيث يمكنك إجراء تغييرات تصميم متقدمة على النماذج ، مثل إضافة عناصر تحكم مخصصة ، و إضافة تعليمات برمجية
٧. **معالج النماذج :** لإنشاء نموذج خطوة بخطوة بمساعدة أكسس
٨. **ورقة بيانات :** لإنشاء نموذج على شكل ورقة بيانات
٩. **مربع حوار مشروط :** لإنشاء نموذج حوار يحتوي على زر إلغاء الأمر و زر موافق
١٠. **Pivot Table :** لإنشاء نموذج يعرض البيانات بشكل جدول مفصلي .

طرق إنشاء النماذج :

سنكتفي في هذا السياق بشرح طريقتين فقط في إنشاء النماذج ، و يترك للطالب التعرف على الطرق الأخرى :



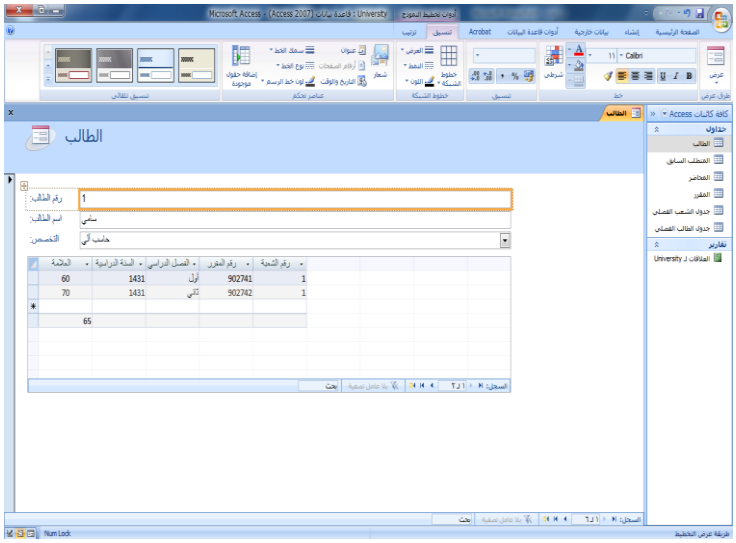
- نموذج - معالج النماذج

إنشاء نموذج بالضغط على أيقونة "نموذج" :

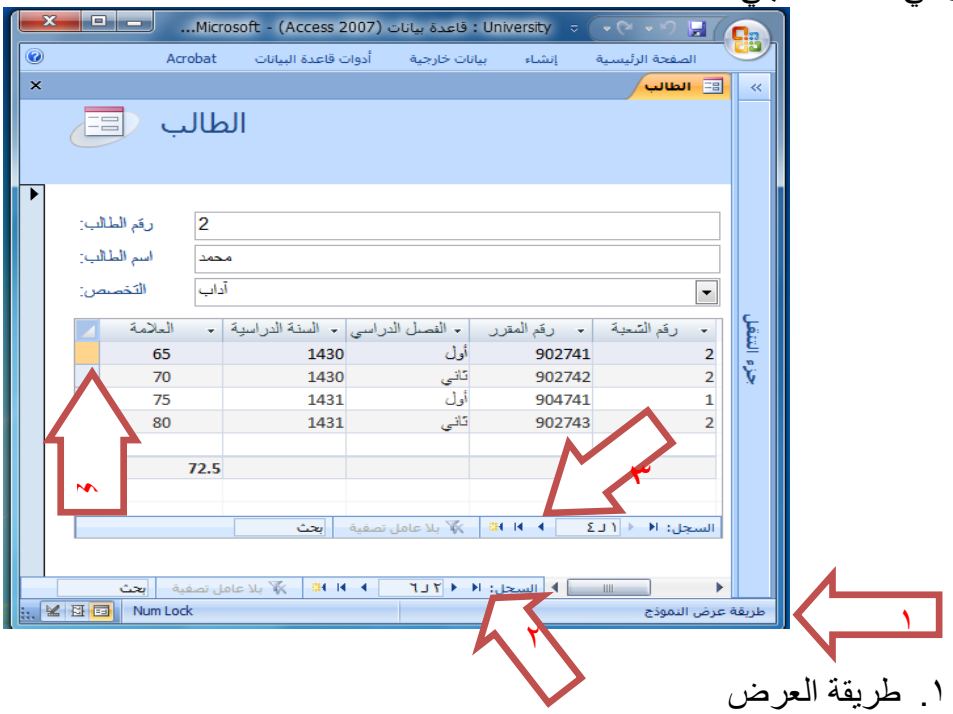
تعتبر هذه الطريقة الأسهل في إنشاء النماذج حيث أنها تتكون من خطوتين إثنتين فقط :

١. إختيار الجدول أو الإستعلام المراد عمل النموذج له من جزء التنقل
٢.إختيار أيقونة "نموذج من مجموعة نماذج ضمن تبويب إنشاء

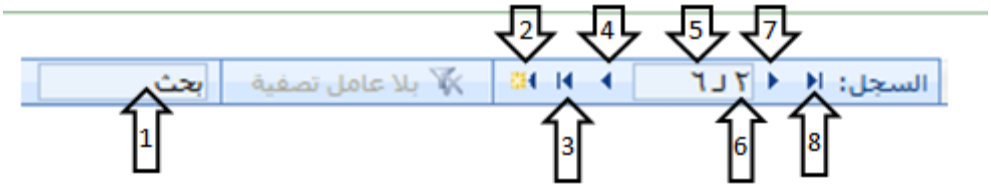
- بناءا على الخطوتين السابقتين يظهر لدينا النموذج كما في الشكل



- نرى في الشكل ما يلي :



١. طريقة العرض
٢. شريط تنقل النموذج الرئيسي
٣. شريط تنقل النموذج الفرعي
٤. السجل الفعال في النموذج الفرعي



- نرى في الشكل شريط تنقل السجلات ، وفيه التفصيلات التالية :

١. منطقة البحث حسب رقم السجل
٢. لإنشاء سجل جديد
٣. للإنتقال إلى السجل الأخير
٤. للإنتقال إلى السجل التالي
٥. عدد السجلات الكلي
٦. رقم السجل الحالي (السجل الفعال)
٧. للإنتقال للسجل السابق
٨. للإنتقال للسجل الأول

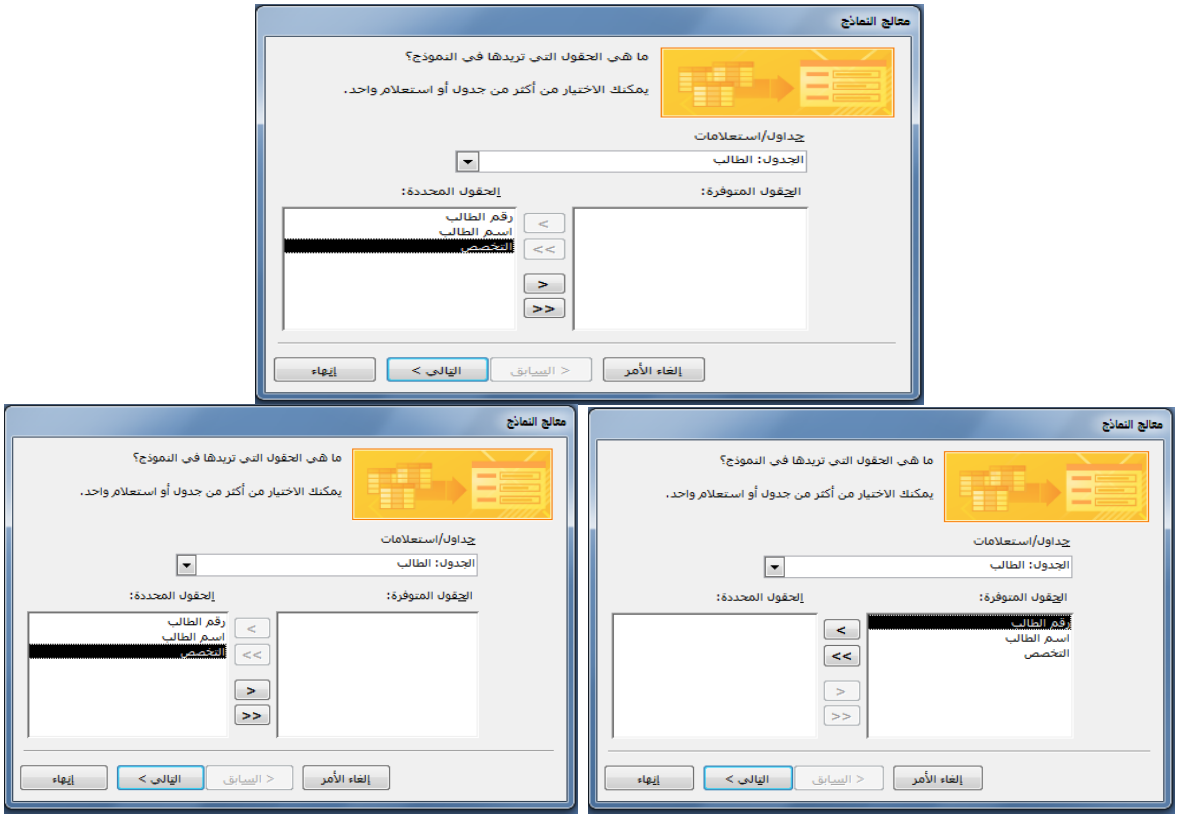
إنشاء نموذج بإستخدام طريقة معالج النماذج :

- في هذه الطريقة يتم إنشاء النموذج خطوة بخطوة و بمساعدة برنامج أكسس . قم بإختيار التبويب إنشاء - المجموعة نماذج - نماذج إضافية - معالج النماذج

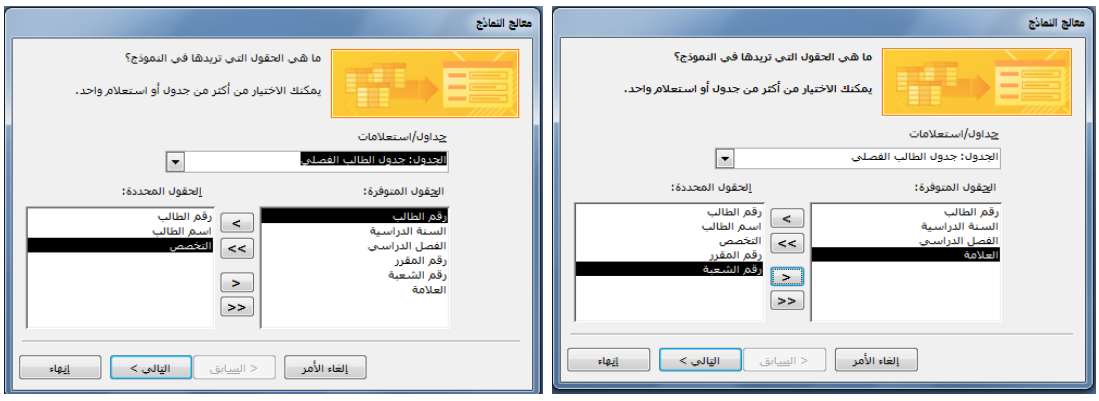
IbtiHAlinO

- **تظهر لدينا شاشة معالج النماذج بالخطوات كالتالي :**

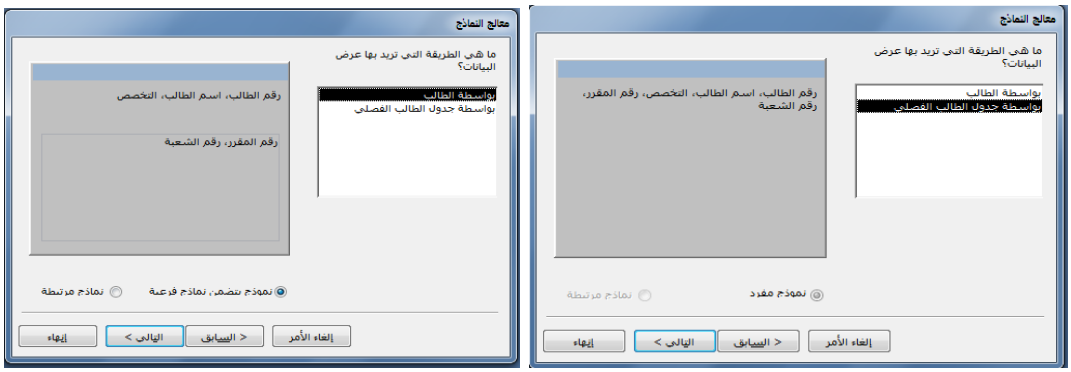
١. تحديد مصدر البيانات(جداول أو إستعلامات) و الحقول من ذلك المصدر.



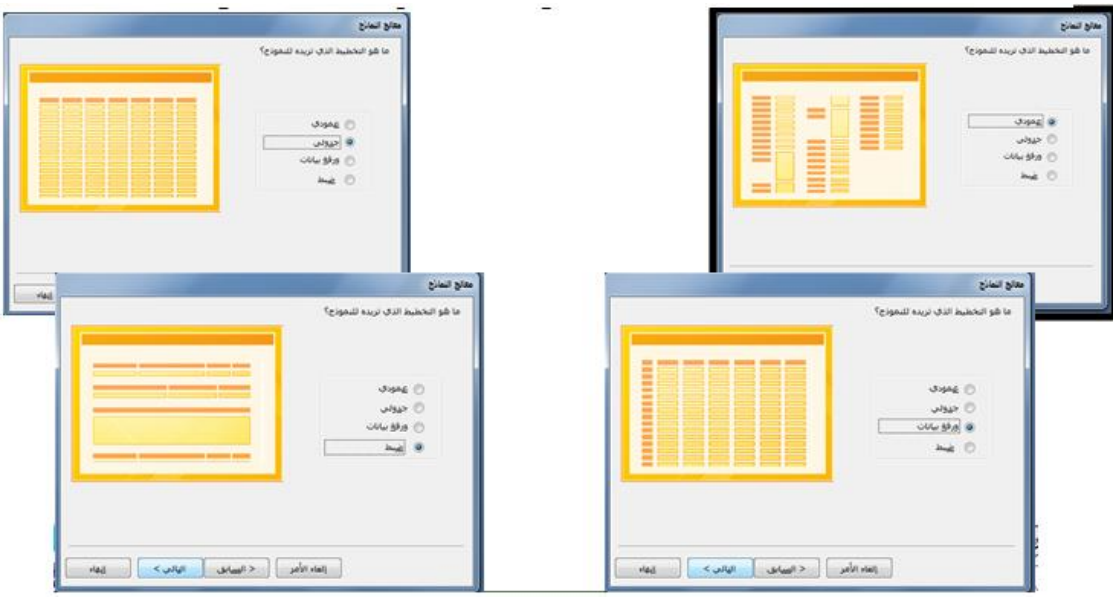
- ملاحظة يمكن الإختيار من أكثر من مصدر بشرط وجود علاقة ربط بين تلك المصادر.
- يمكن إختيار أو التراجع عن إختيار بعض أو كل الحقول في جهة المصدر



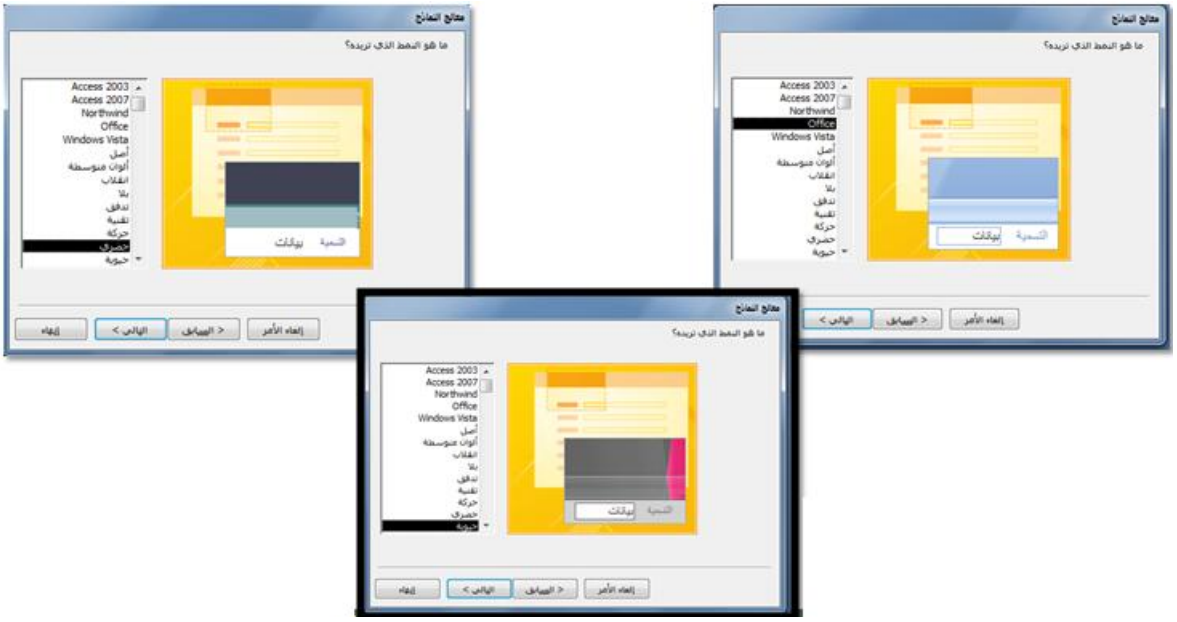
٢. بسبب إختيار بياناتنا من أكثر من مصدر ، فإنه يطلب منا هذه الخطوة الإضافية التي تتعلق بعرض البيانات من وجهة نظر المصدر الأول أو الثاني أو الثالث...



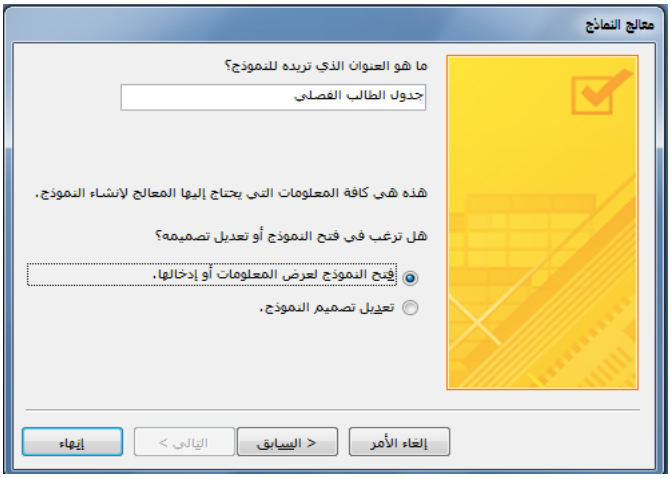
٣. إختيار التخطيط ، وهي طريقة توزيع البيانات في داخل النموذج



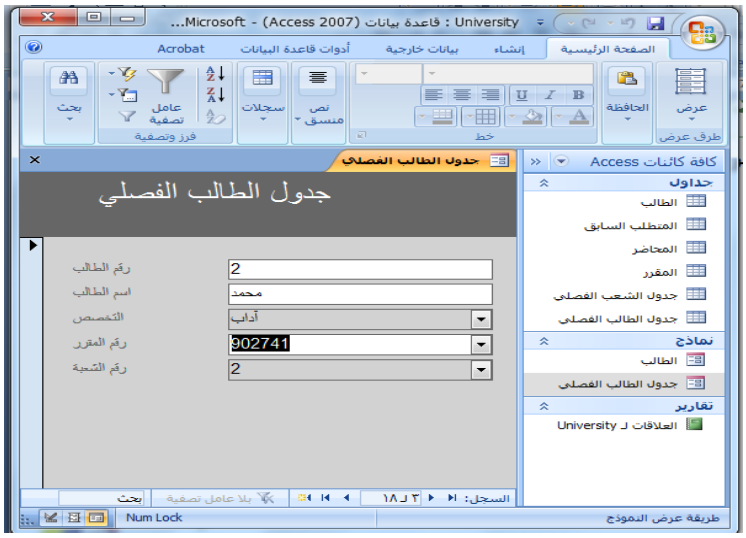
٤. إختيار النمط ، و يقصد بالنمط شكل الخط ، و لونه ولون الخلفية ..إلخ



٥. تحديد عنوان النموذج



- و نهاية يظهر لدينا النموذج بالشكل التالي :



طرق عرض النموذج :

- بعد أن يتم إنشاء النموذج ، فإنه يمكن عرضه بأكثر من طريقة :

طريقة عرض النموذج :

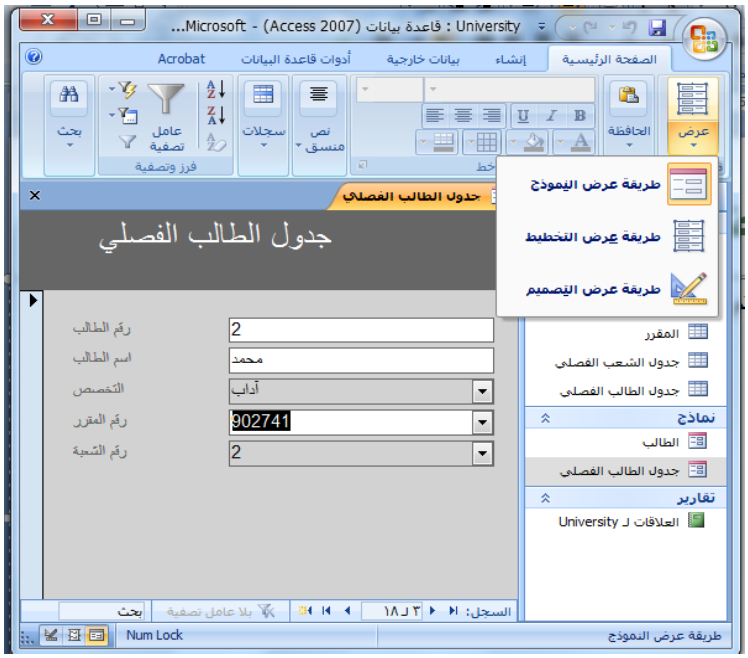
لا يمكن التغيير فيها، وهي الشاشة التنفيذية للنموذج

طريقة عرض التخطيط :

وفيها يمكن للمستخدم أن يغير في تخطيط النموذج ، إضافة أو حذف حقل من قائمة.

طريقة عرض التصميم :

وفيها يمكن للمستخدم إجراء تغييرات كبيرة على النموذج من حيث التصميم بأكمله

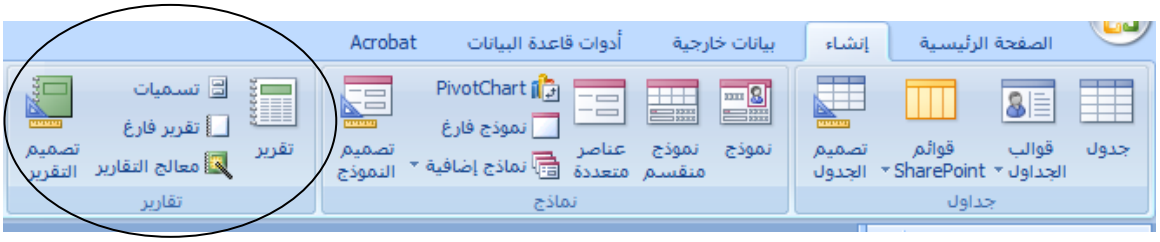


ما هو التقرير؟

- التقرير عبارة عن كائن يساعد في إنشاء ورقة للطباعة على الطابعة ، ويمكن أخذ محتوياته من جدول أو إستعلام ، واحد أو أكثر.
- كثيرا ما يتم الحكم على برنامج إدارة قاعدة البيانات من خلال التقارير التي يمكن طباعتها من خلاله ، كون النسخة الورقية أكثر تداولا من النماذج ، وهذا لا يقلل من أهمية النماذج.

طرق إنشاء التقارير :

- يمكن إنشاء التقرير في أكسس ٢٠٠٧ بإحدى الطرق الآتية :



١. **تقرير :** إنشاء تقرير لجدول أو إستعلام مختار من جزء التنقل
٢. **تسميات :** لتجهيز تقرير على شكل تسميات labels ليتم طباعتها و إلصاقها
٣. **تقرير فارغ :** إنشاء تقرير فارغ ، ليتم إضافة الحقول المطلوب من قائمة حقول موجودة
٤. **معالج التقارير :** لإنشاء التقرير خطوة بخطوة بمساعدة أكسس
٥. **تصميم التقارير :** إنشاء التقرير بشكل أكثر تحديدا و تخصصا ، بما يسمح بإضافة أدوات تحكم و جمل برمجية خاصة

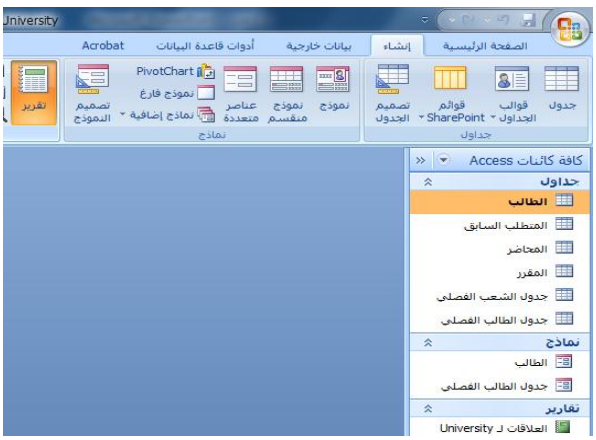
- **سنكتفي في هذا السياق بشرح طريقتين فقط في إنشاء التقارير ، و يترك للطالب التعرف على الطرق الأخرى:**

- تقرير - معالج التقارير

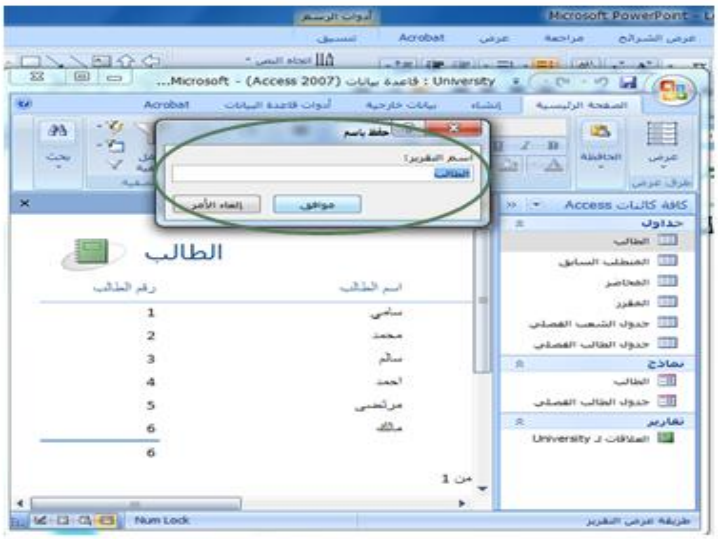
إنشاء تقرير بإستخدام الضغط على أيقونة "تقرير" :

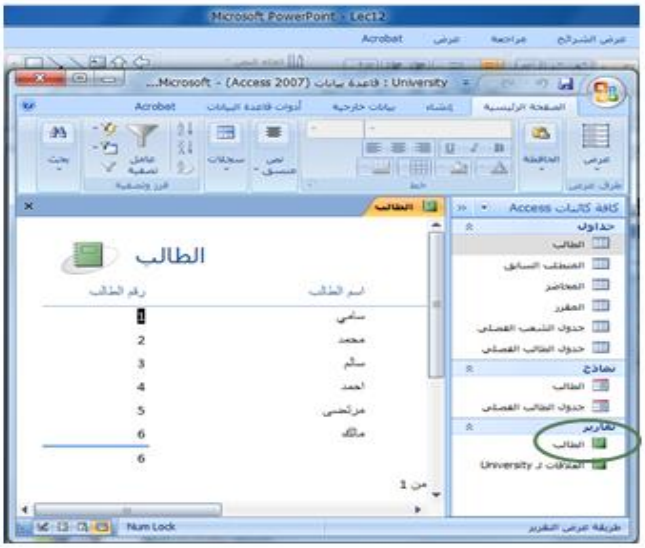
- **تعتبر هذه الطريقة الأسهل في إنشاء التقارير ، حيث تتلخص في خطوتين إثنتين :**

١. إختيار الجدول أو الإستعلام المطلوب تقريره
٢. الضغط على أيقونة "تقرير"



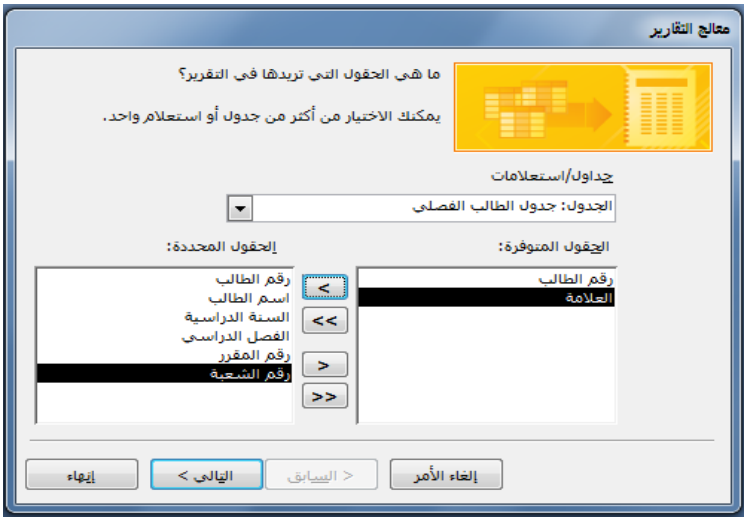
- فيظهر لدينا التقرير بالشكل التالي :



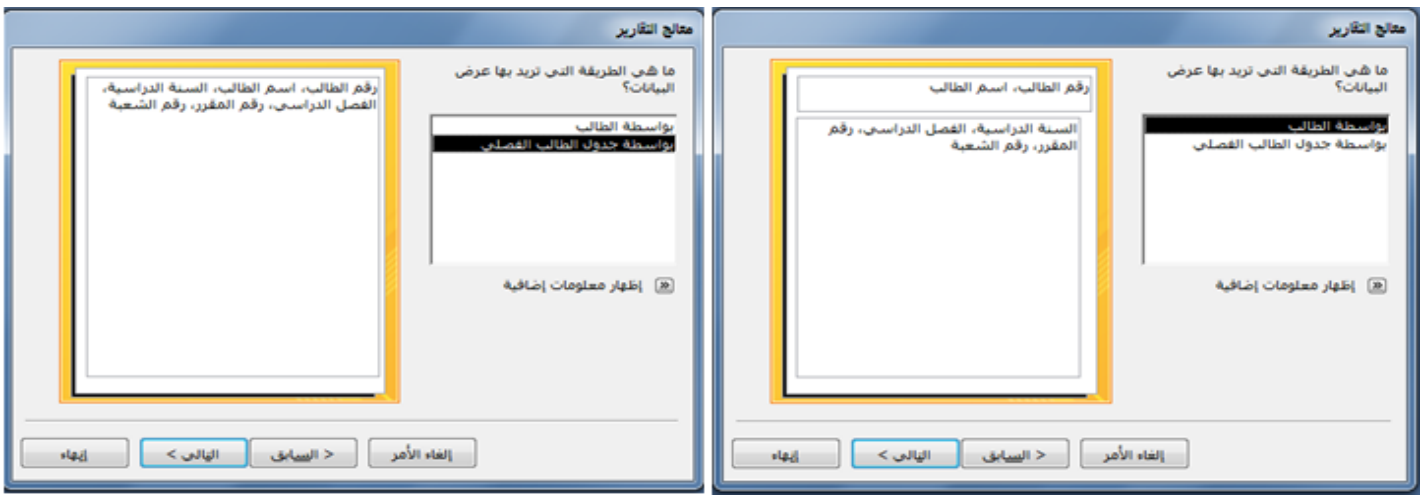


تمر عملية إنشاء التقرير بطريقة معالج التقارير بالخطوات التالية:

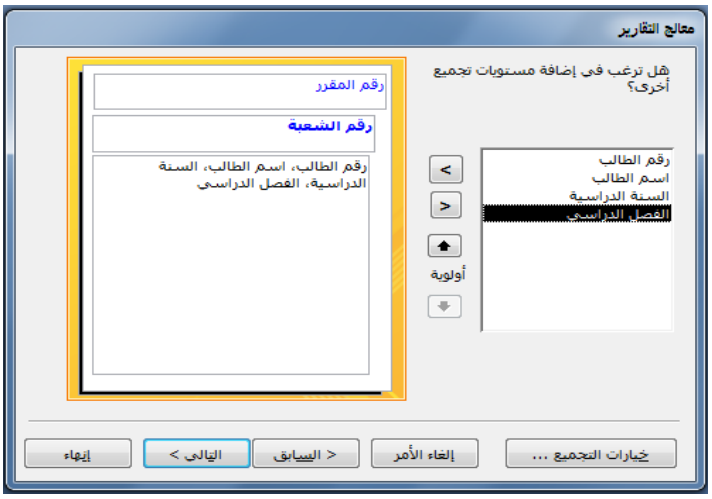
١. تحديد مصدر بيانات التقرير و الحقول المطلوبة من ذلك المصدر و ينطبق عليه ما ينطبق على النموذج بهذا الخصوص



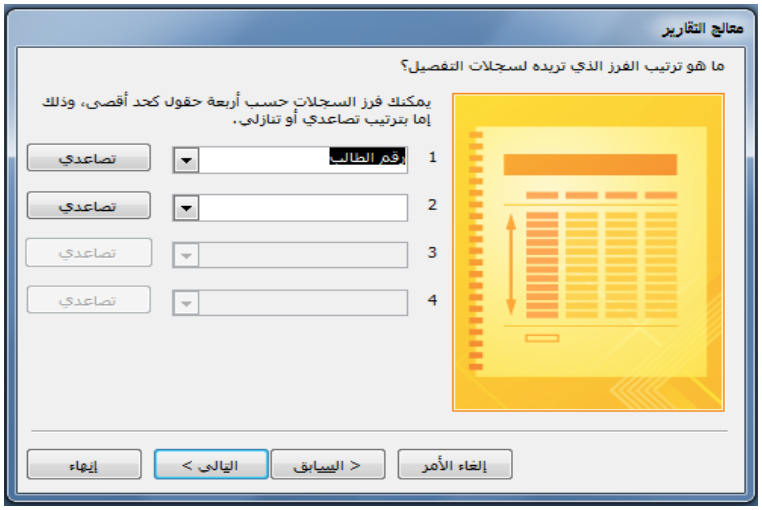
٢. إختيار وجهة النظر في عرض البيانات ، كون البيانات من أكثر من مصدر



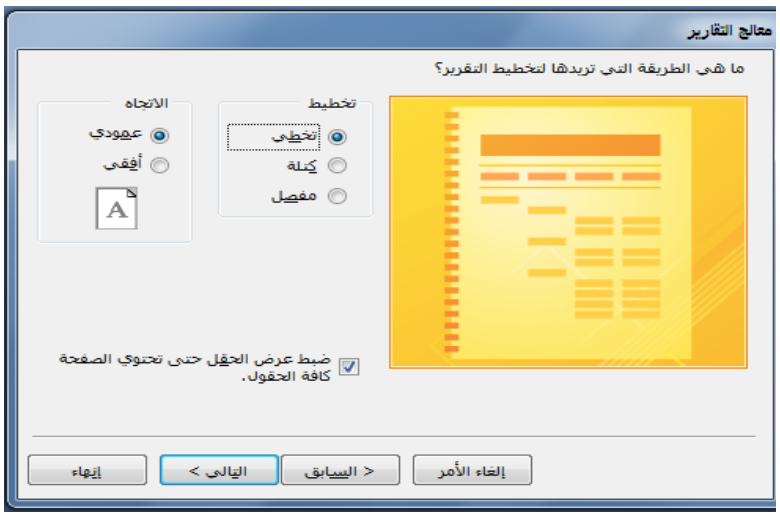
٣. إختيار حقل أو أكثر ليتم تصنيف البيانات إلى مجموعات حسب الحقول المختارة



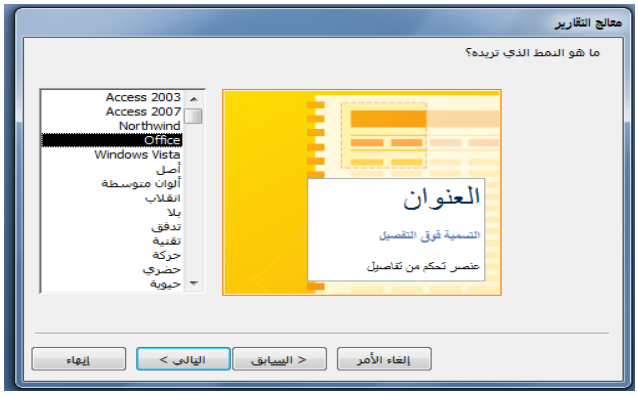
٤. ترتيب أو فرز بيانات تقرير حسب حقل معين



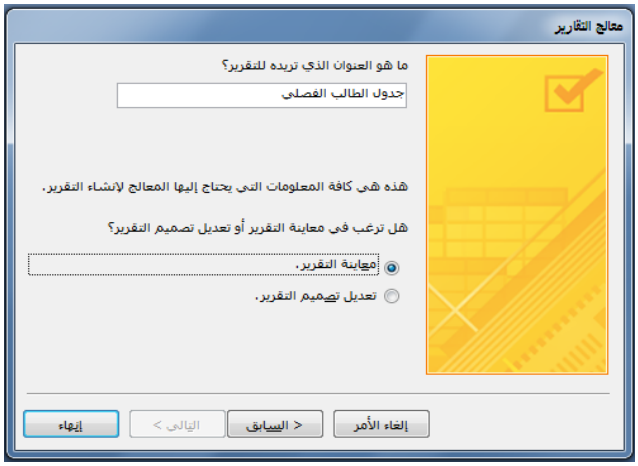
٥. إختيار التخطيط المناسب ، و إتجاه الصفحة للطباعة



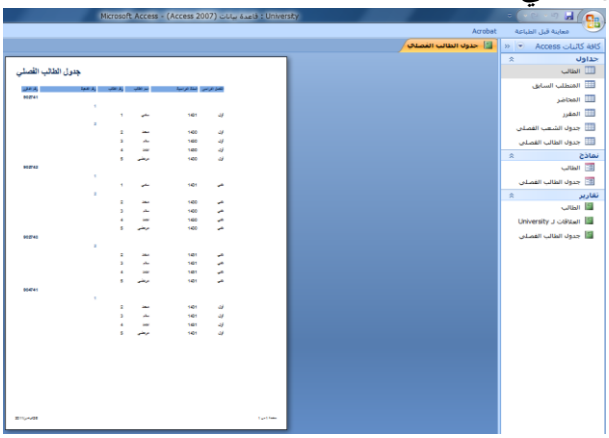
٦. إختيار النمط المناسب



٧. إختيار عنوان التقرير



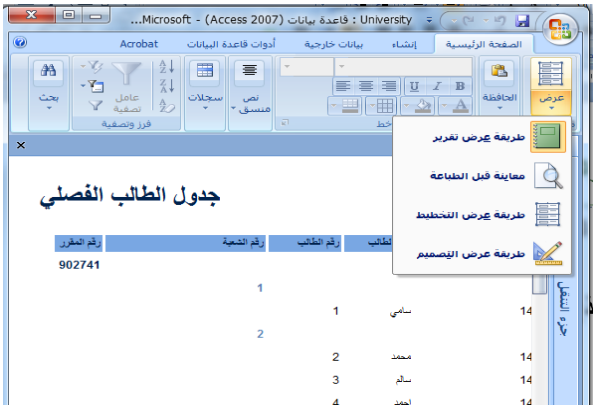
- **يظهر لدينا التقرير بالشكل التالي :**



طرق عرض التقرير :

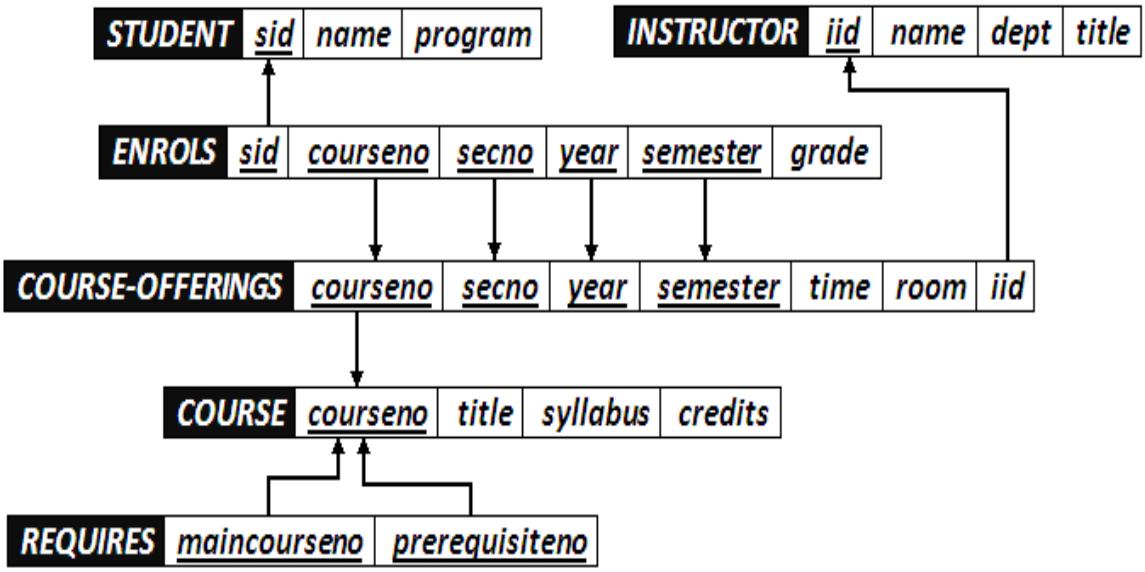
- **يمكن عرض التقرير بإحدى الطرق التالية :**

- طريقة عرض تقرير - معاينة قبل الطباعة - طريقة عرض التخطيط - طريقة عرض تصميم التقرير



إنشاء الاستعلام

تذكير بمثال الجامعة :



جداول مثال الجامعة :

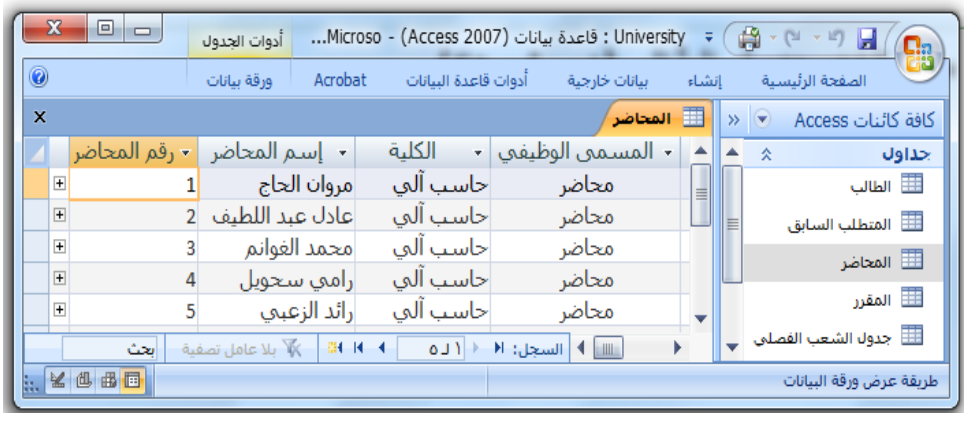
في مثال الجامعة لدينا عدد ٦ جداول هي :

- جدول الطالب(Student)
- جدول المحاضر (Instructor)
- جدول الطالب الفصلي(Enrols)
- جدول الشعب الفصلي(Course-Offerings)
- جدول المقرر(Course)
- جدول المتطلب السابق(Requires)

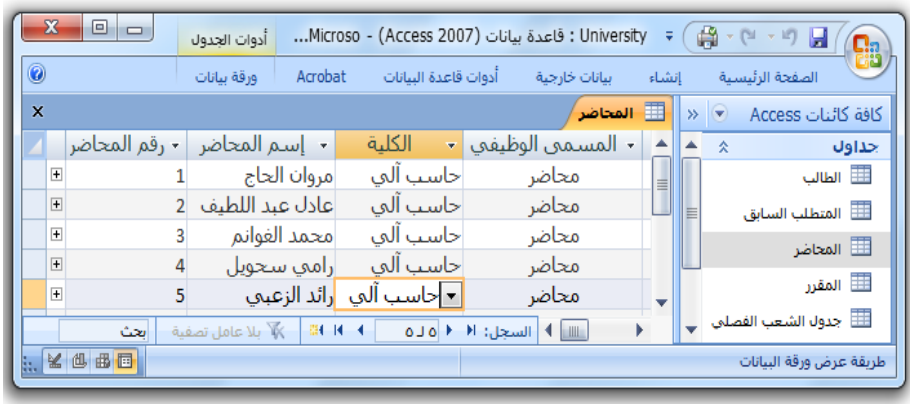
بيانات جداول الجامعة :

- من الضروري أن تكون البيانات في جداول قاعدة البيانات متكاملة مع بعضها البعض.
- و لتوضيح درسنا من المهم أن نعرض بيانات الجداول لمقارنة نتائج الاستعلام مع بيانات الجداول

- بيانات جدول الطالب (Student)

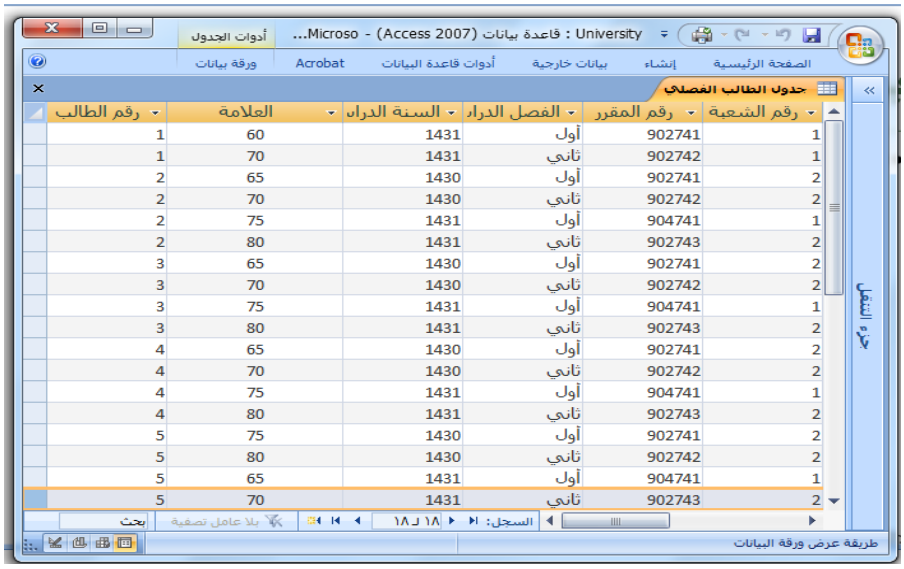


- بيانات جدول المحاضر(Instructor)



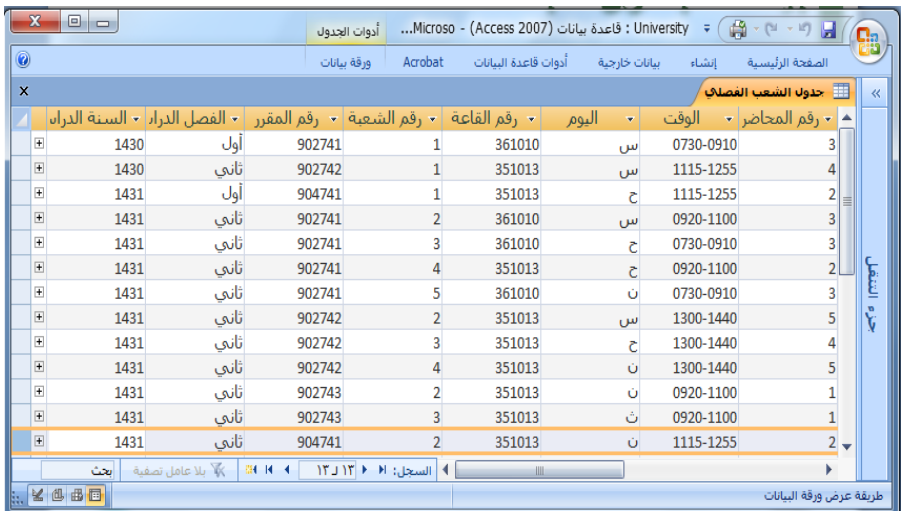
رقم المحاضر	إسم المحاضر	الكلية	المسمى الوظيفي
1	مروان الحاج	حاسب آلي	محاضر
2	عادل عبد اللطيف	حاسب آلي	محاضر
3	محمد الغوانم	حاسب آلي	محاضر
4	رامي سحويل	حاسب آلي	محاضر
5	رائد الزعبي	حاسب آلي	محاضر

- بيانات جدول الطالب الفصلي(Enrols)



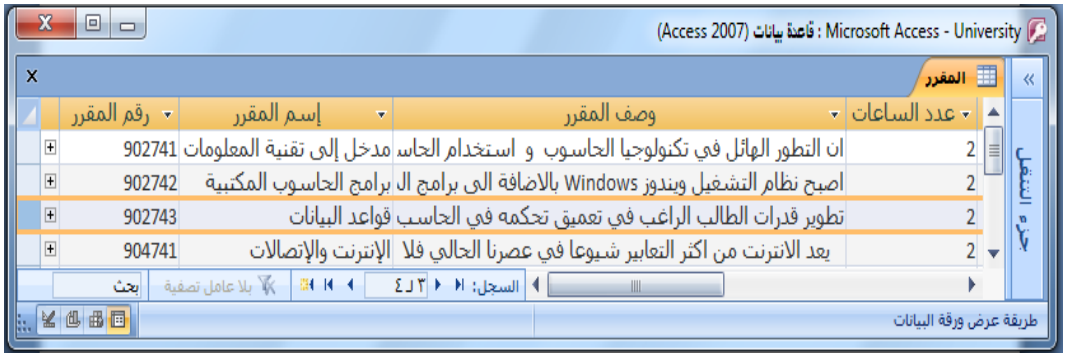
رقم الطالب	العلامة	السنة الدراسية	الفصل الدراسي	رقم المقرر	رقم الشعبة
1	60	1431	أول	902741	1
1	70	1431	ثاني	902742	1
2	65	1430	أول	902741	2
2	70	1430	ثاني	902742	2
2	75	1431	أول	904741	1
2	80	1431	ثاني	902743	2
3	65	1430	أول	902741	2
3	70	1430	ثاني	902742	2
3	75	1431	أول	904741	1
3	80	1431	ثاني	902743	2
4	65	1430	أول	902741	2
4	70	1430	ثاني	902742	2
4	75	1431	أول	904741	1
4	80	1431	ثاني	902743	2
5	75	1430	أول	902741	2
5	80	1430	ثاني	902742	2
5	65	1431	أول	904741	1
5	70	1431	ثاني	902743	2

- بيانات جدول الشعب الفصلي(Course-Offerings)

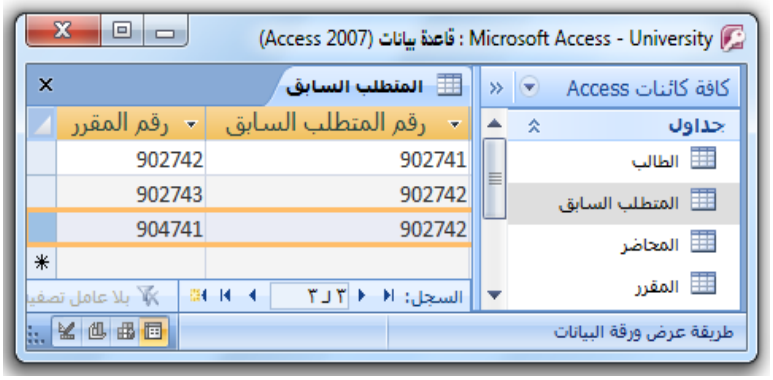


رقم المحاضر	الوقت	اليوم	رقم القاعة	رقم الشعبة	رقم المقرر	الفصل الدراسي	السنة الدراسية
3	0730-0910	س	361010	1	902741	أول	1430
4	1115-1255	س	351013	1	902742	ثاني	1430
2	1115-1255	ح	351013	1	904741	أول	1431
3	0920-1100	س	361010	2	902741	ثاني	1431
3	0730-0910	ح	361010	3	902741	ثاني	1431
2	0920-1100	ح	351013	4	902741	ثاني	1431
3	0730-0910	ن	361010	5	902741	ثاني	1431
5	1300-1440	س	351013	2	902742	ثاني	1431
4	1300-1440	ح	351013	3	902742	ثاني	1431
5	1300-1440	ن	351013	4	902742	ثاني	1431
1	0920-1100	ن	351013	2	902743	ثاني	1431
1	0920-1100	ث	351013	3	902743	ثاني	1431
2	1115-1255	ن	351013	2	904741	ثاني	1431

- بيانات جدول المقرر(Course)



- بيانات جدول المتطلب السابق(Requires)



ما هو الاستعلام؟

- الاستعلام هو طريقة لاستخلاص البيانات من جداول قاعدة البيانات أو من إستعلامات أخرى
- يستخدم الاستعلام للاجابة عن أسئلة بسيطة أو لإجراء عمليات حسابية أو تجميع بيانات من جداول مختلفة، أو إضافة بيانات جدول أو تغييرها أو حذفها.
- تسمى الاستعلامات المستخدمة في استرداد البيانات أو تطبيق عمليات حسابية بإستعلامات التحديد.
- تسمى إستعلامات إضافة البيانات أو تغييرها أو حذفها بإستعلامات الإجراء
- يمكن إستخدام الإستعلام كمصدر للبيانات في بناء النماذج أو بناء التقارير

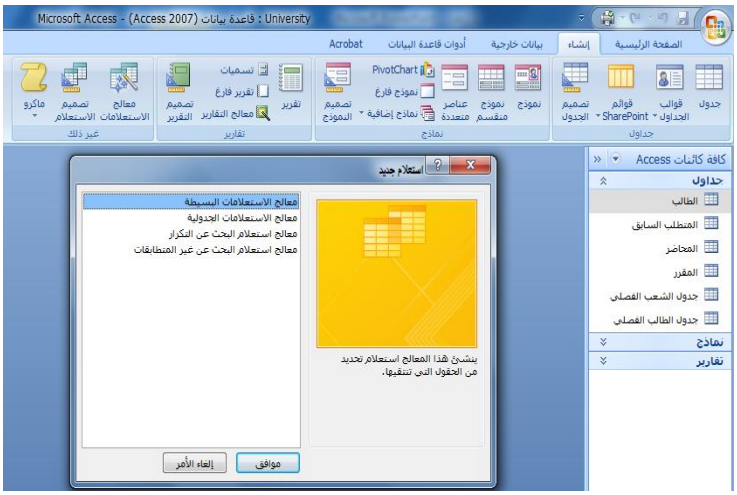
طرق إنشاء الاستعلام :

- يمكن إنشاء الإستعلام من التبويب إنشاء- مجموعة غيرذلك بإحدى طريقتين :

١. معالج الاستعلامات
٢. تصميم الاستعلام

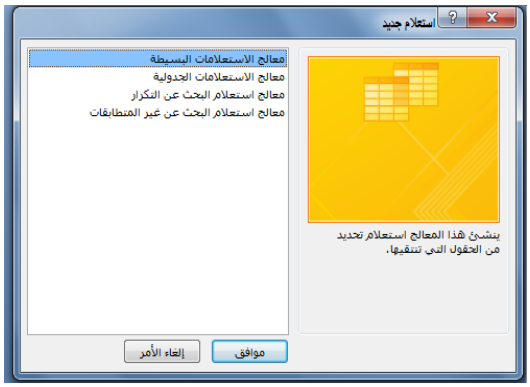
إنشاء الاستعلام باستخدام "معالج الاستعلامات"

من التبويب إنشاء – المجموعة غيرذلك – نختار معالج الاستعلامات



- نلاحظ ظهور شاشة "استعلام جديد" تحتوي على الخيارات:

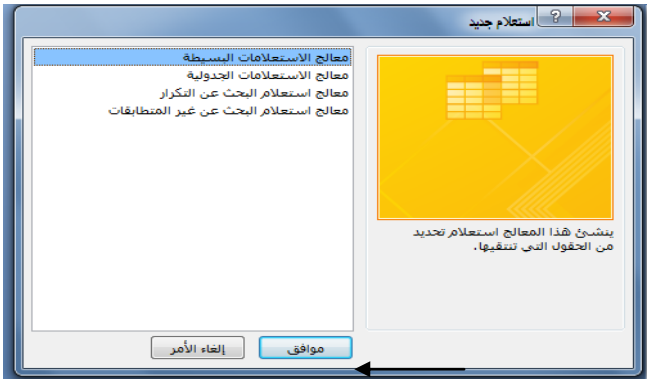
  - معالج الاستعلامات البسيطة
  - معالج الاستعلامات الجدولية
  - معالج استعلام البحث عن التكرار
  - معالج استعلام البحث عن غير المتطابقات



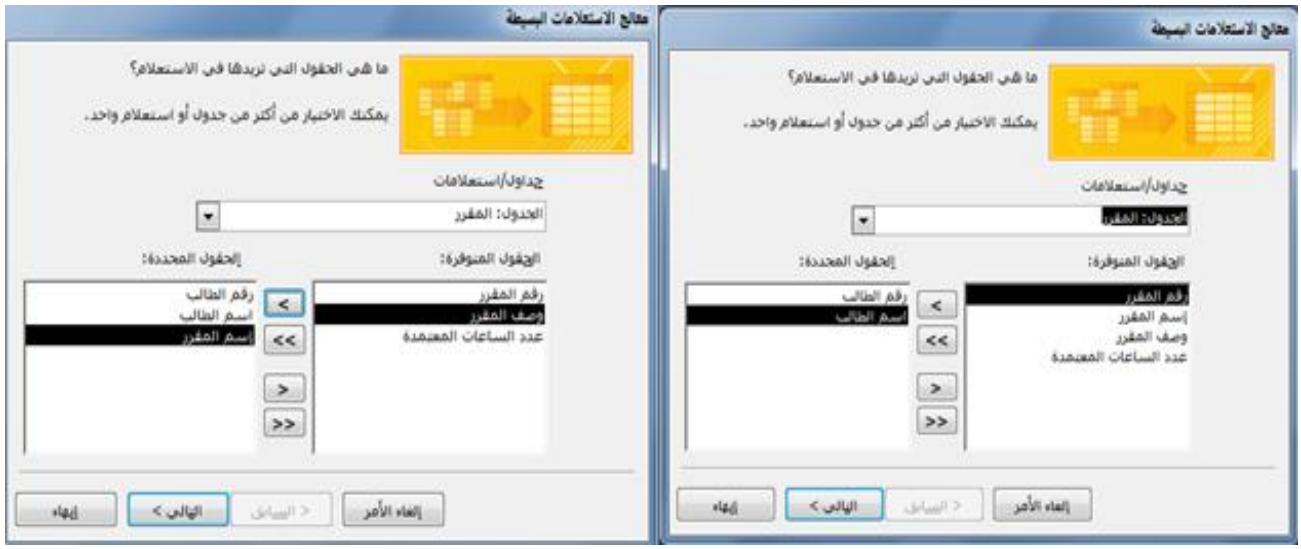
- **معالج الاستعلامات البسيطة : ينشئ هذا** المعالج استعلام تحديد من الحقول التي تنتقيها.

- **مثال :**

نرغب بانشاء استعلام يقوم بعرض المعلومات (رقم الطالب ، اسم الطالب، اسم المقرر، العلامة)

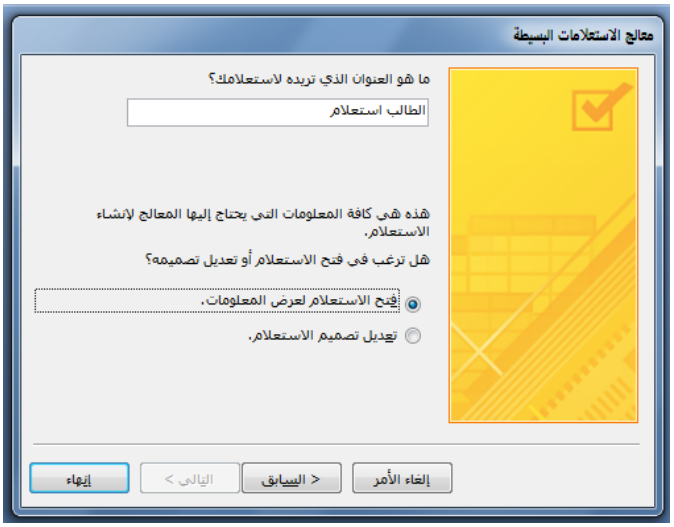


- معالج الاستعلامات البسيطة :



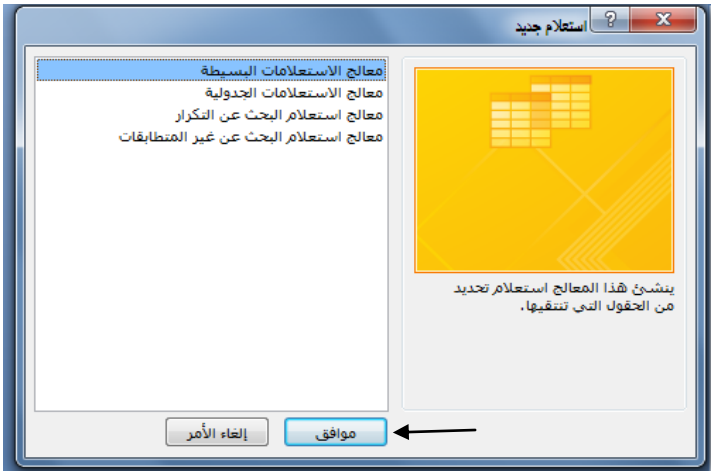
- بإمكانك الاختيار بين استعلام تفصيل أو استعلام ملخص ، لنختار تفصيل

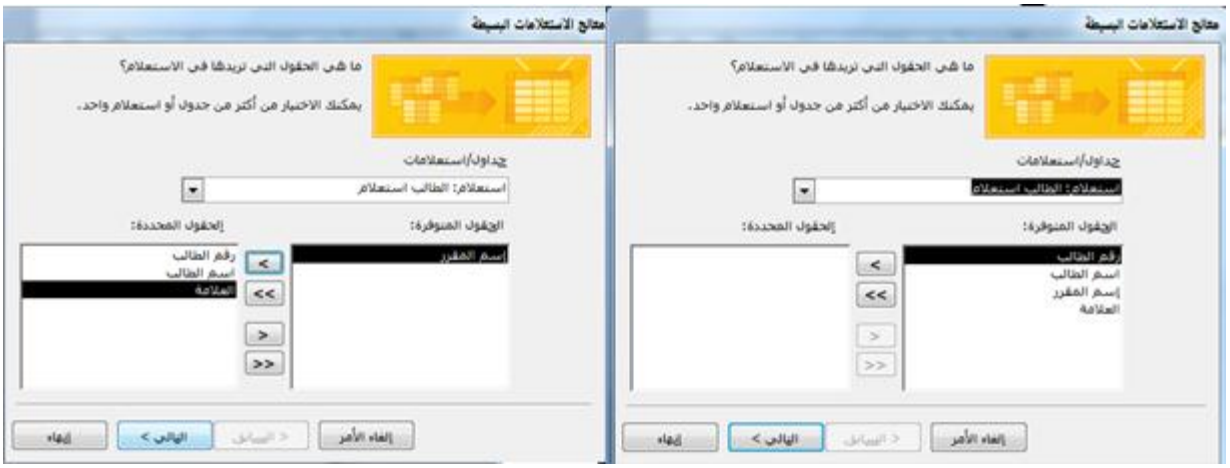
بإمكانك الاختيار بين فتح الاستعلام لعرض المعلومات ، أو تعديل تصميم الاستعلام ، فيما يختص بتعديل تصميم الاستعلام ، فسوف نشرحه في جزء منفصل ، لذلك سنختار فتح الاستعلام لعرض المعلومات



تظهر لدينا نتائج الاستعلام

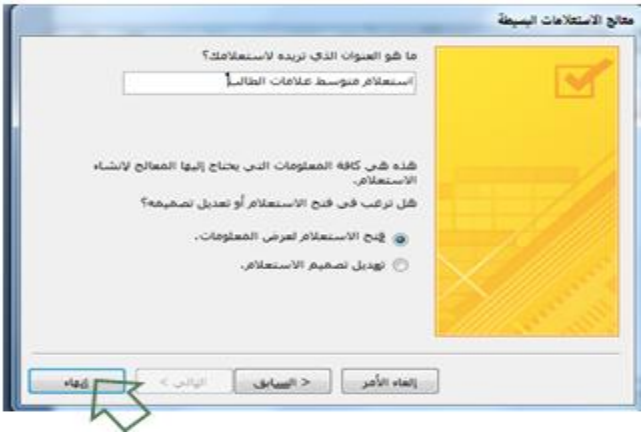
نرغب بانشاء استعلام يقوم بعرض المعلومات (رقم الطالب ، اسم الطالب، متوسط علامات الطالب)


استعلام جديد
معالج الاستعلامات البسيطة
معالج الاستعلامات الجدولية
معالج استعلام البحث عن التكرار
معالج استعلام البحث عن غير المتطابقات
ينشئ هذا المعالج استعلام تحديد من الحقول التي تنتقيها.
موافق
إلغاء الأمر


معالج الاستعلامات البسيطة
ما هي الحقول التي تريدها في الاستعلام؟
يمكنك الاختيار من أكثر من جدول أو استعلام واحد.
جداول/استعلامات
الحقول المتوفرة:
الحقول المحددة:
رقم الطالب
اسم الطالب
اسم المقرر
العلامة
التالي >
إلغاء الأمر
إنهاء

معالج الاستعلامات البسيطة
هل تريد استعلام تفصيل أم استعلام ملخص؟
تفصيل (عرض كل حقل لكل سجل)
ملخص
خيارات التلخيص ...
1
3
التالي >
إلغاء الأمر
إنهاء

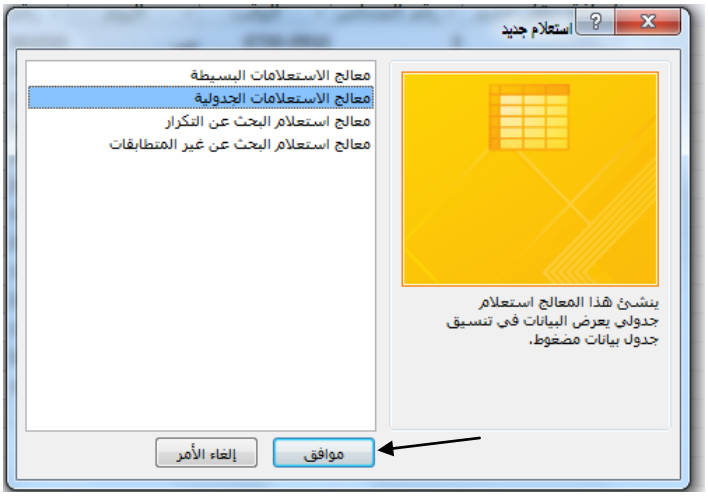
خيارات التلخيص
ما هي قيم الملخص التي تريد حسابها؟
موافق
إلغاء الأمر
2
الحقل
مجموع
متوسط
أدنى
أقصى
العلامة
عدد السجلات في جدول الطالب الفصلي

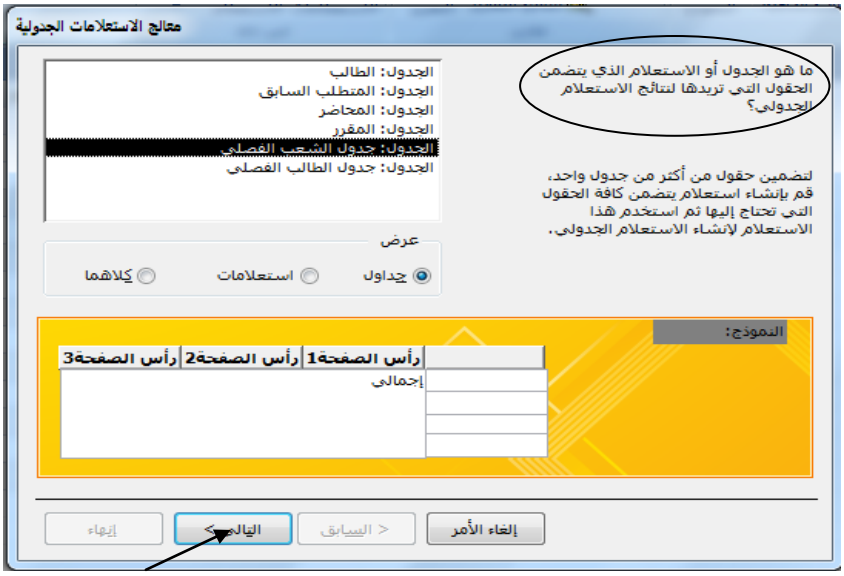

معالج الاستعلامات البسيطة
ما هو العنوان الذي تريده لاستعلامك؟
استعلام متوسط علامات الطالب
هذه هي كافة المعلومات التي يحتاج إليها المعالج لإنشاء الاستعلام.
هل تريد فتح الاستعلام أو تعديل تصميمه؟
فتح الاستعلام لعرض المعلومات.
تعديل تصميم الاستعلام.
إلغاء الأمر
< السابق
إنهاء

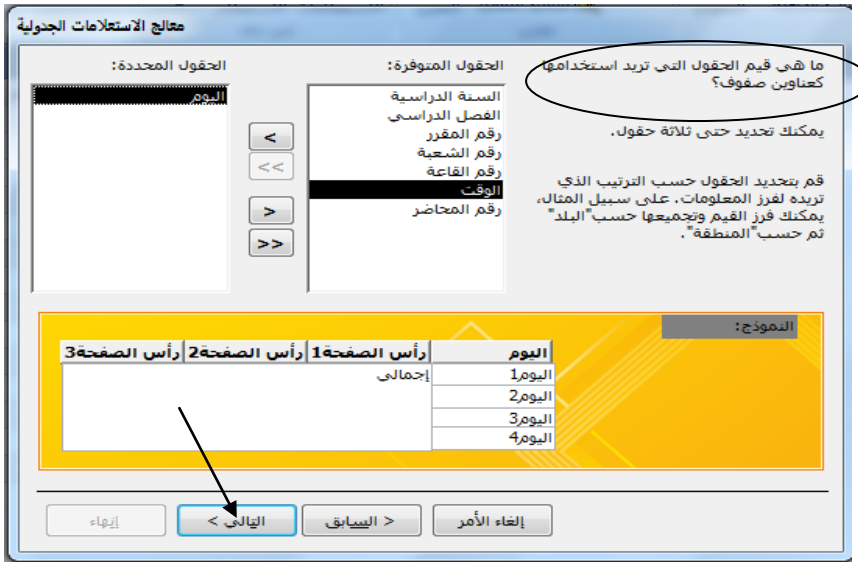
استعلام متوسط علامات الطالب
رقم الطالب
اسم الطالب
من العلامة Avg
1 سامي 65
2 محمد 72.3333333333333
3 سالم 72.3333333333333
4 أحمد 72.3333333333333
5 مرتضى 73
السجل: 1 من 5
جزء التنقل
Acrobat
أدوات قاعدة
بيانات خارجي
إنشاء
الصفحة الر

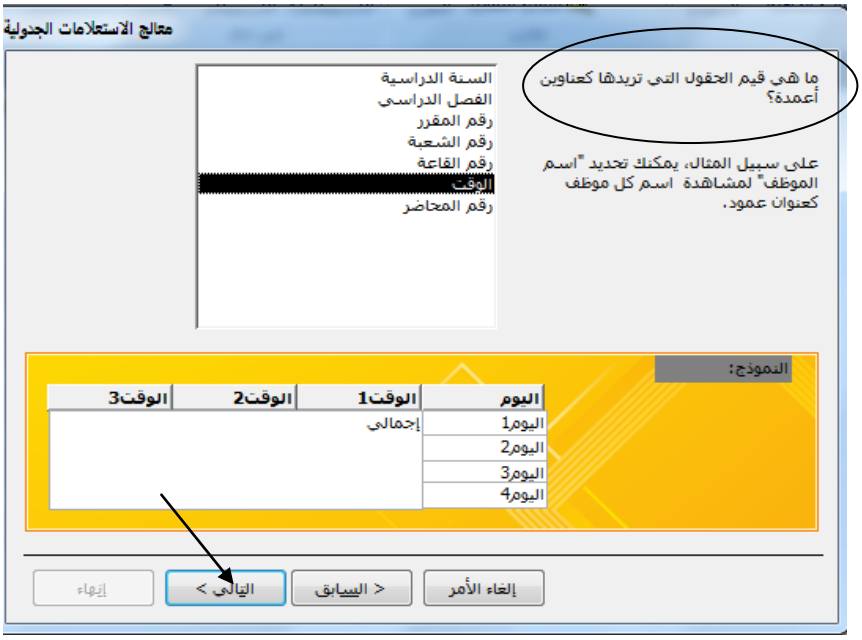
- **معالج الاستعلامات الجدولية :**

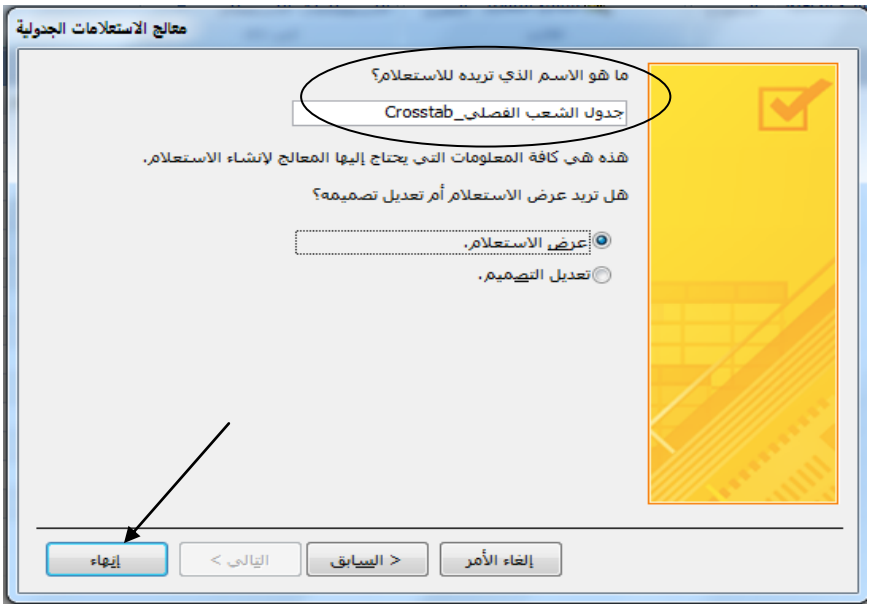
ينشئ هذا المعالج استعلام جدولي يعرض البيانات في تنسيق جدول بيانات مضغوط و يطلق عليه جدول مفصلي أيضا
مثال : أنشئ إستعلام جدولي يظهر عدد الشعب في كل يوم وكل وقت








معالج الاستعلامات الجدولية
ما هي قيم الحقول التي تريدها كعناوين أعمدة؟
السنة الدراسية
الفصل الدراسي
رقم المقرر
رقم الشعبة
رقم القاعة
اليوم
رقم المحاضر
على سبيل المثال، يمكنك تحديد "اسم الموظف" لمشاهدة اسم كل موظف كعنوان عمود.
النموذج:
اليوم
الوقت1
الوقت2
الوقت3
اليوم1
اليوم2
اليوم3
اليوم4
إجمالي
إلغاء الأمر
< السابق
التالي >
إنهاء

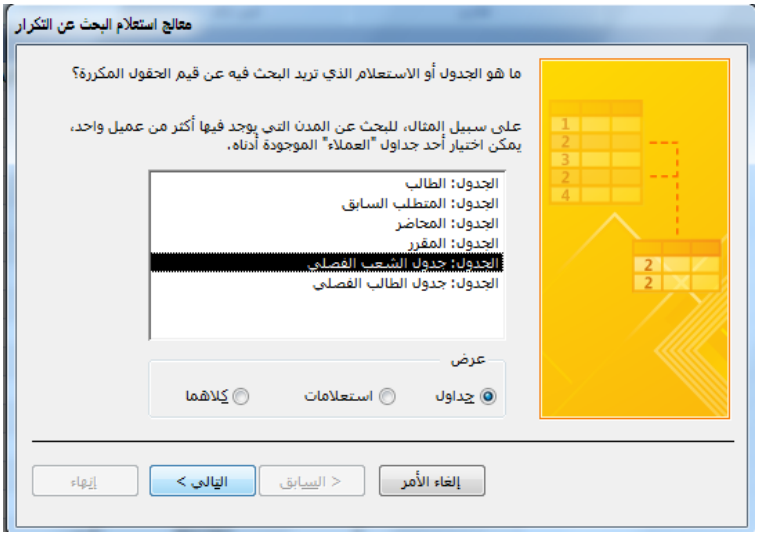

معالج الاستعلامات الجدولية
ما هو الاسم الذي تريده للاستعلام؟
جدول الشعب الفصلي_Crosstab
هذه هي كافة المعلومات التي يحتاج إليها المعالج لإنشاء الاستعلام.
هل تريد عرض الاستعلام أم تعديل تصميمه؟
عرض الاستعلام.
تعديل التصميم.
إلغاء الأمر
< السابق
التالي >
إنهاء

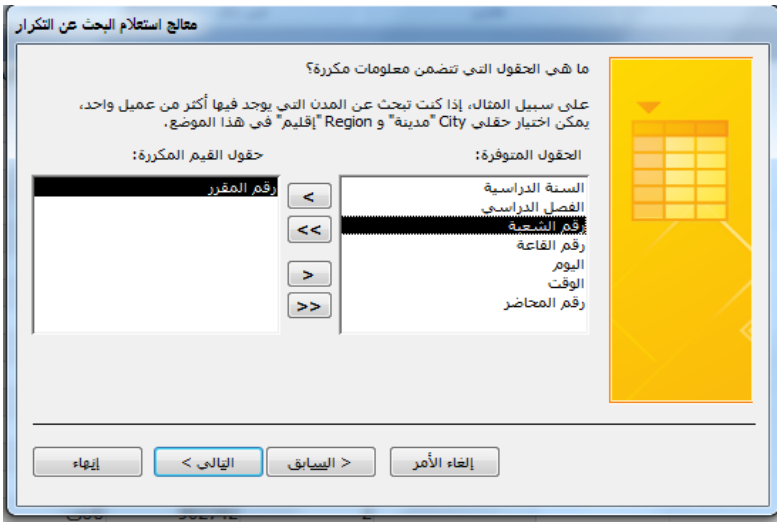
Microsoft Access - University : قاعدة بيانات (Access 2007)
جدول الشعب الفصلي_Crosstab
اليوم
0730-0910
0920-1100
1115-1255
1300-1440
إجمالي رقم المقرر
ث
ح
س
ن
1
4
4
4
السجل:
بحث
بلا عامل تصفية
طريقة عرض ورقة البيانات
جزء التنقل

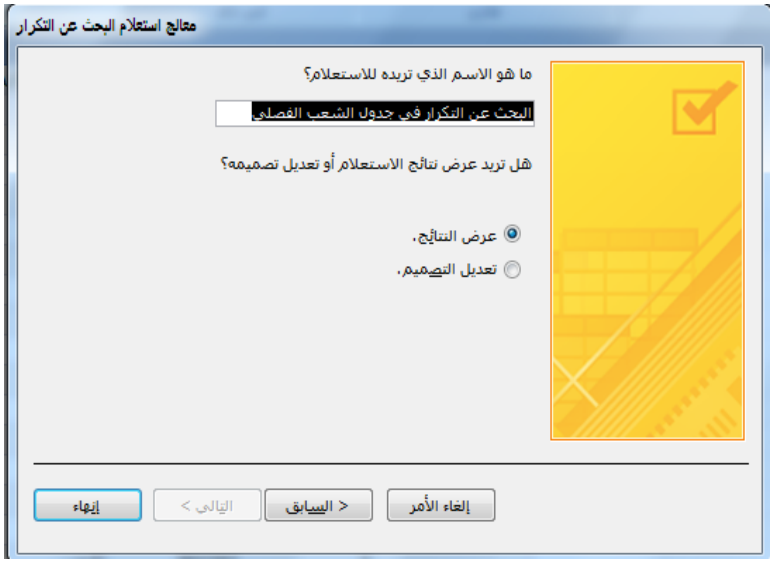
إنشاء الاستعلام

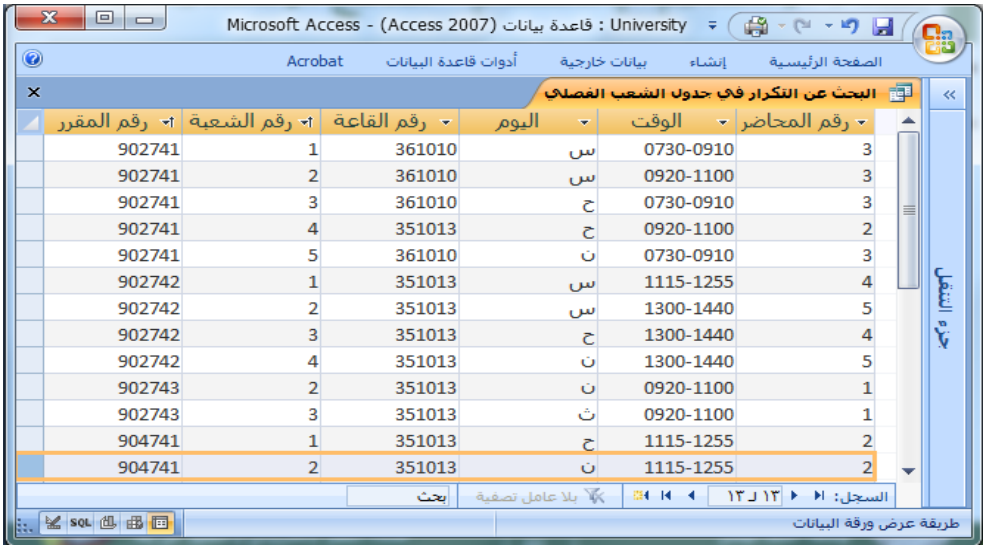
إنشاء الاستعلام باستخدام "معالج الاستعلامات"

- معالج استعلام البحث عن التكرار
- **مثال :** أنشئ إستعلام تكرار ، موضحا فيه المقررات متعددة الشعب في جدول الشعب الفصلي



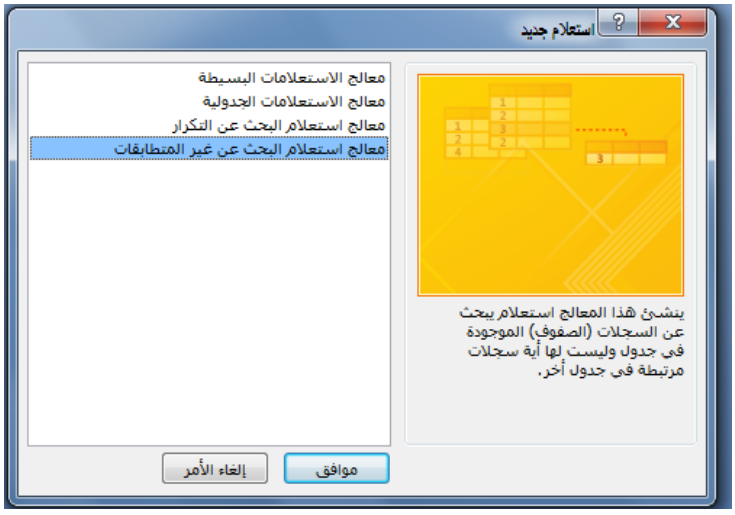


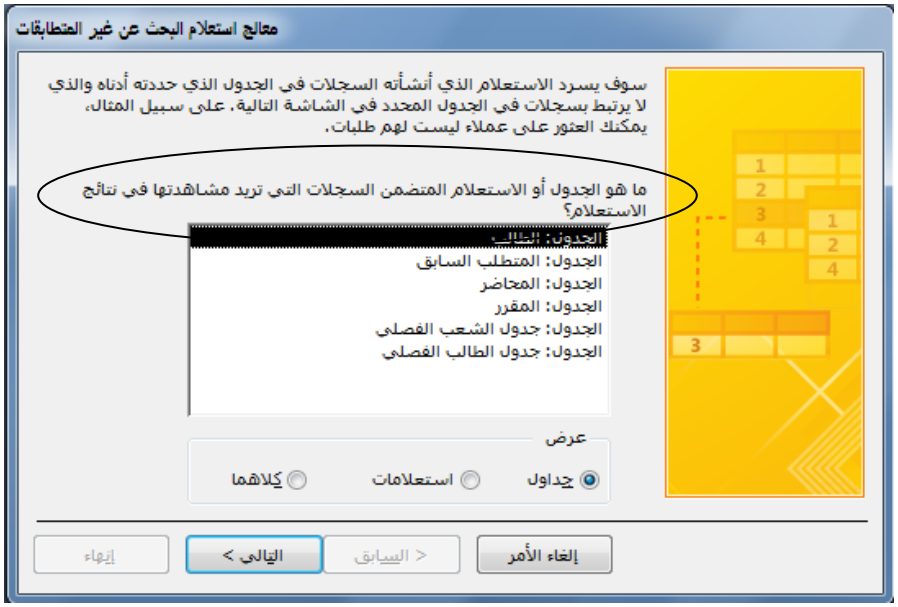


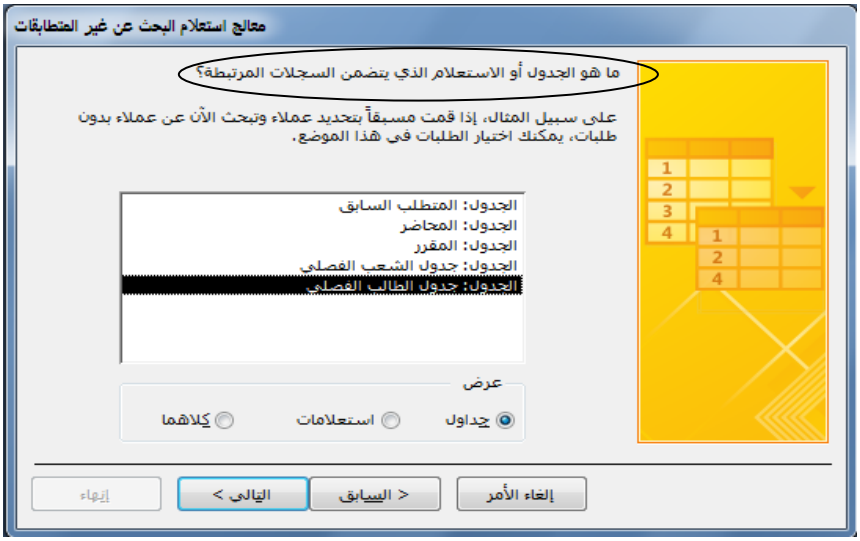


- **معالج استعلام البحث عن غير المتطابقات :**

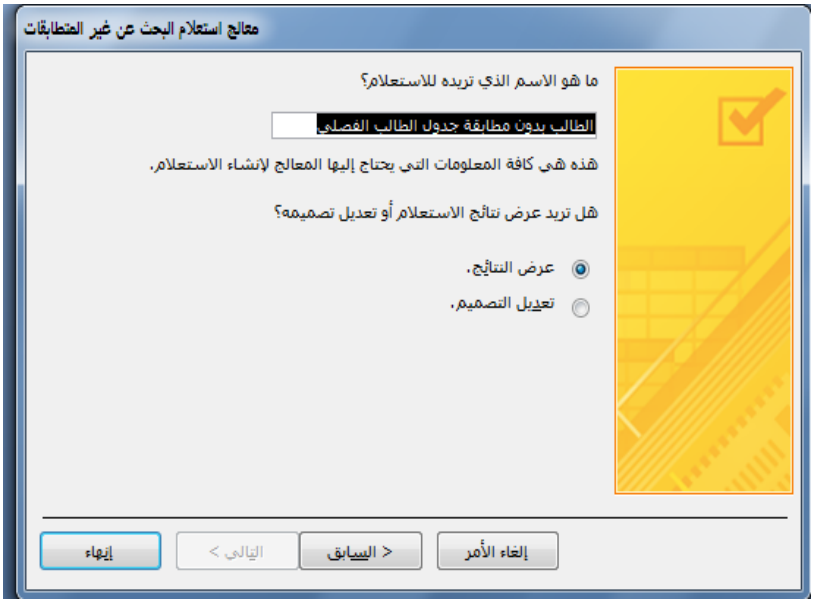
ينشئ هذا المعالج استعلام للبحث عن السجلات الموجودة في جدول ، وليس لها سجلات مرتبطة في جدول آخر
مثال: أنشئ استعلام بحث عن غير متطابقات ، لايجاد الطلاب الذين ليس لهم مقررات مسجلة في جدول الطالب الفصلي

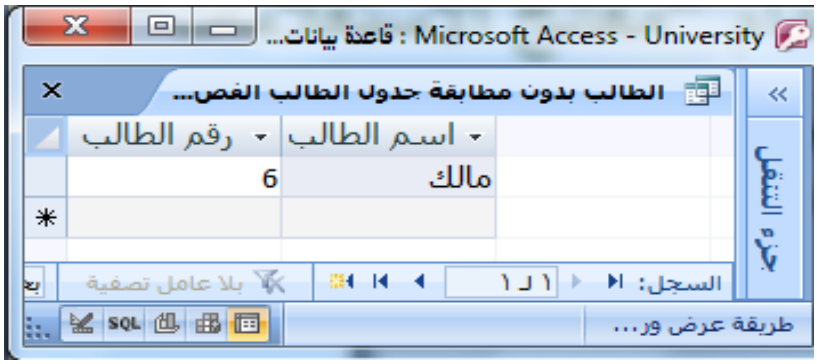



معالج استعلام البحث عن غير المتطابقات
سوف يسرد الاستعلام الذي أنشأته السجلات في الجدول الذي حددته أدناه والذي لا يرتبط بسجلات في الجدول المحدد في الشاشة التالية. على سبيل المثال، يمكنك العثور على عملاء ليست لهم طلبات.
ما هو الجدول أو الاستعلام المتضمن السجلات التي تريد مشاهدتها في نتائج الاستعلام؟
الجدول: الطلاب
الجدول: المتطلب السابق
الجدول: المحاضر
الجدول: المقرر
الجدول: جدول الشعب الفصلي
الجدول: جدول الطالب الفصلي
عرض
جداول
استعلامات
كلاهما
إلغاء الأمر
< السابق
التالي >
إنهاء


معالج استعلام البحث عن غير المتطابقات
ما هو الجدول أو الاستعلام الذي يتضمن السجلات المرتبطة؟
على سبيل المثال، إذا قمت مسبقاً بتحديد عملاء وتبحث الآن عن عملاء بدون طلبات، يمكنك اختيار الطلبات في هذا الموضع.
الجدول: المتطلب السابق
الجدول: المحاضر
الجدول: المقرر
الجدول: جدول الشعب الفصلي
الجدول: جدول الطالب الفصلي
عرض
جداول
استعلامات
كلاهما
إلغاء الأمر
< السابق
التالي >
إنهاء

معالج استعلام البحث عن غير المتطابقات
ما هي المعلومة الموجودة في كلا الجدولين؟
على سبيل المثال، قد يتضمن كل من جدول "العملاء" وجدول "الطلبات" حقل "معرف العميل". وقد تكون أسماء الحقول المتطابقة مختلفة.
حدد الحقل المتطابق في كل جدول، ثم انقر فوق الزر <=>.
الحقول في 'الطالب' :
رقم الطالب
اسم الطالب
التخصص
<=>
الحقول في 'جدول الطالب الفصلي' :
رقم الطالب
السنة الدراسية
الفصل الدراسي
رقم المقرر
رقم الشعبة
العلامة
الحقول المتطابقة:
رقم الطالب <=> رقم الطالب
إلغاء الأمر
< السابق
التالي >
إنهاء





إنشاء الاستعلام باستخدام "تصميم الاستعلام"

- يمكن إنشاء الإستعلام من التبويب إنشاء- مجموعة غيرذلك بإحدى طريقتين:

٢. تصميم الاستعلام

إن عملية إنشاء استعلام بطريقة تصميم الاستعلام تتسم بالسهولة و اليسر ، ولكن يواجه بعض الطلاب مشكلة في صياغة الشرط في الإستعلام .

- **يمكن تلخيص خطوات تصميم الاستعلام كما يلي :**

١. إختيار مصدر المعلومات
٢. ربط مصادر المعلومات في حالة عدم وجود علاقة ربط لتفادي تكرار المعلومات
٣. إختيار الحقول المراد عرضها من ذلك المصدر
٤. إختيار الحقول المراد وضع شرط عليها إن لم تكن قد أختيرت في الخطوة السابقة (٢)
٥. وضع الشرط المطلوب أو مجموعة الشروط المطلوبة

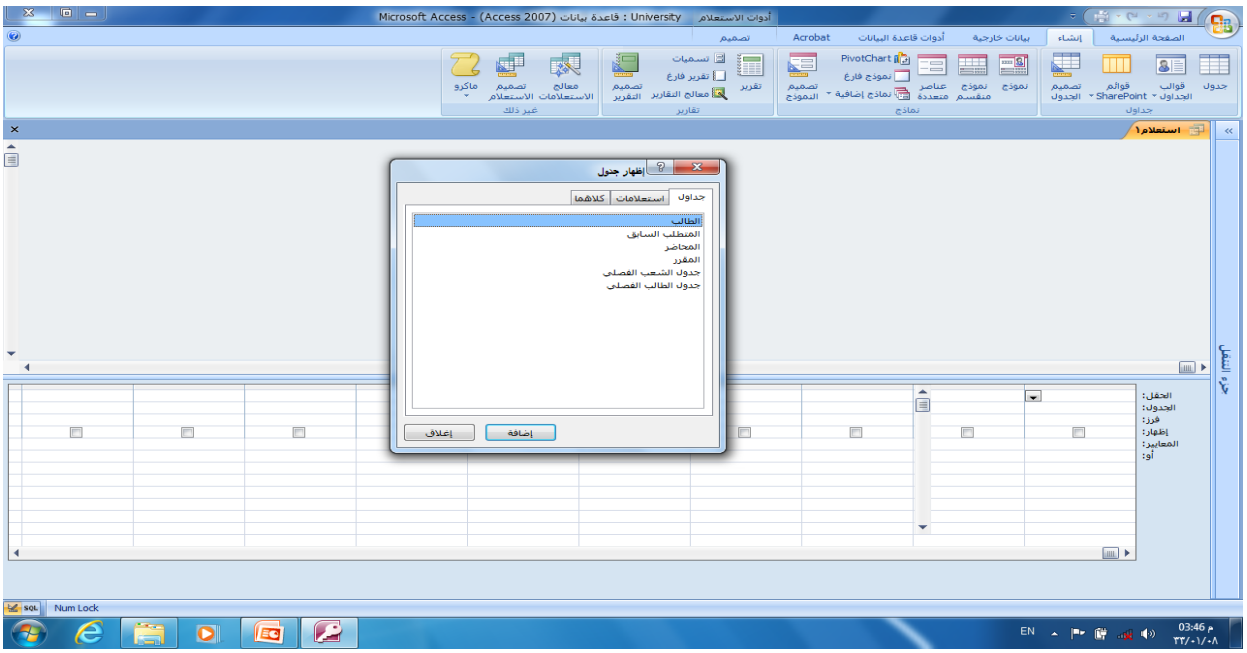
مثال : أنشئ إستعلام يقوم بعرض أسماء الطلاب الناجحين في مادة مدخل إلى تقنية المعلومات ، إذا علمت ان علامة النجاح هي ٦٠ فما فوق

- **التحليل للسؤال :**

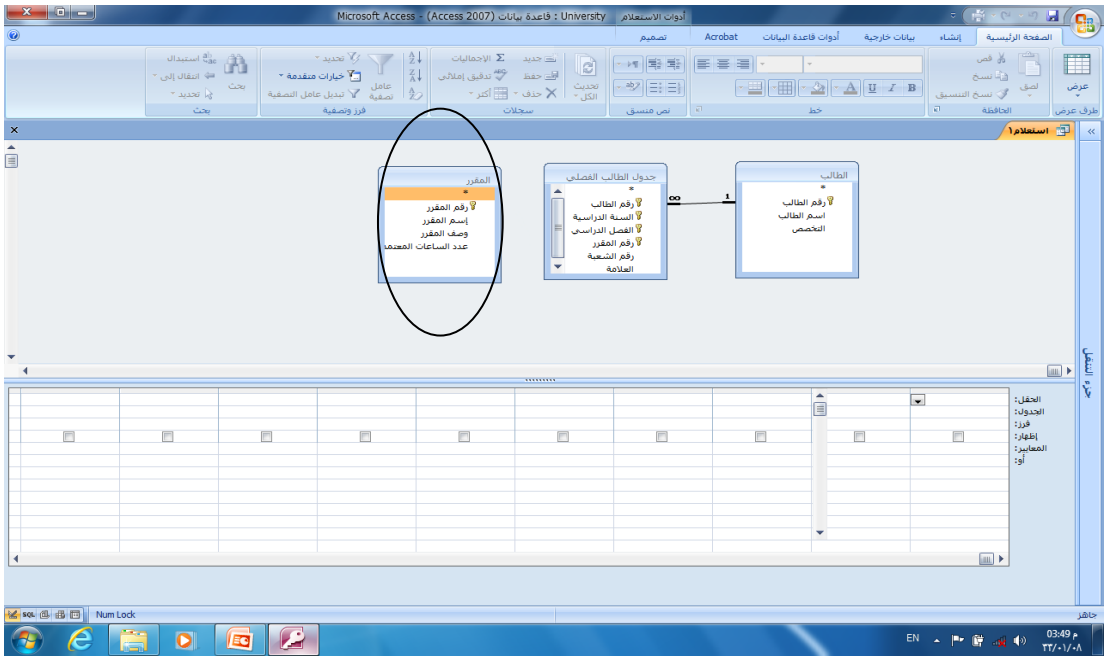
١. الحقول المطلوب عرضها : اسم الطالب.
٢. الحقول المطلوب شرطها : حقل العلامة و حقل إسم المقرر
٣. الشرط:
   - الشرط الأول : حقل العلامة أكبر(>) من ٦٠
   - الشرط الثاني : حقل اسم المقرر يساوي(=) القيمة "مدخل إلى تقنية المعلومات"
   - الشرط الأول والشرط الثاني يتحققان معا (و) (AND)

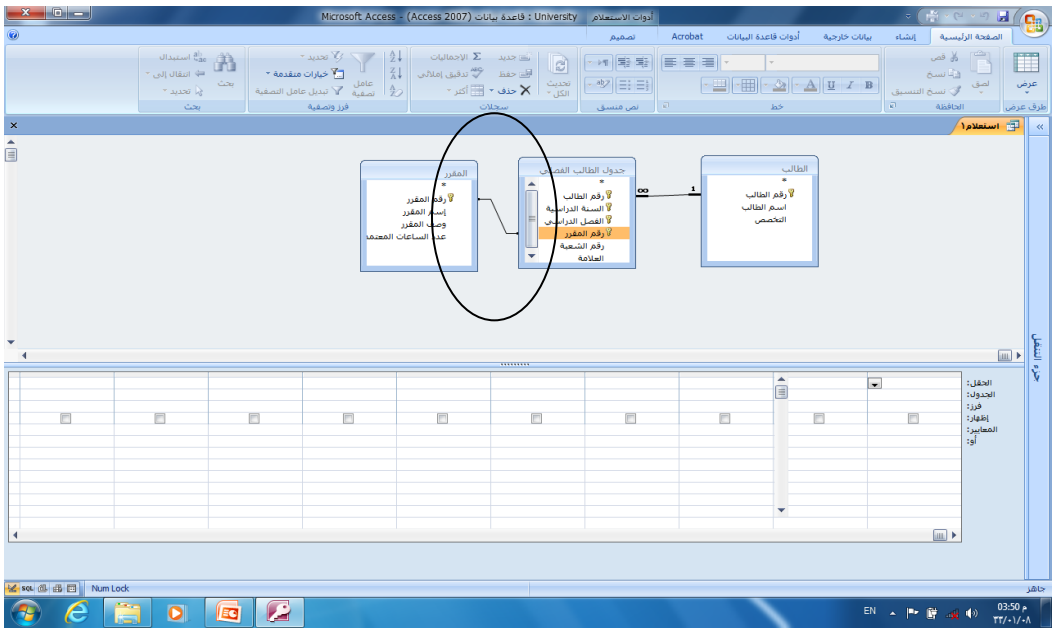
مثال : أنشئ إستعلام يقوم بعرض أسماء الطلاب الناجحين في مادة مدخل إلى تقنية المعلومات ، إذا علمت ان علامة النجاح هي ٦٠ فما فوق

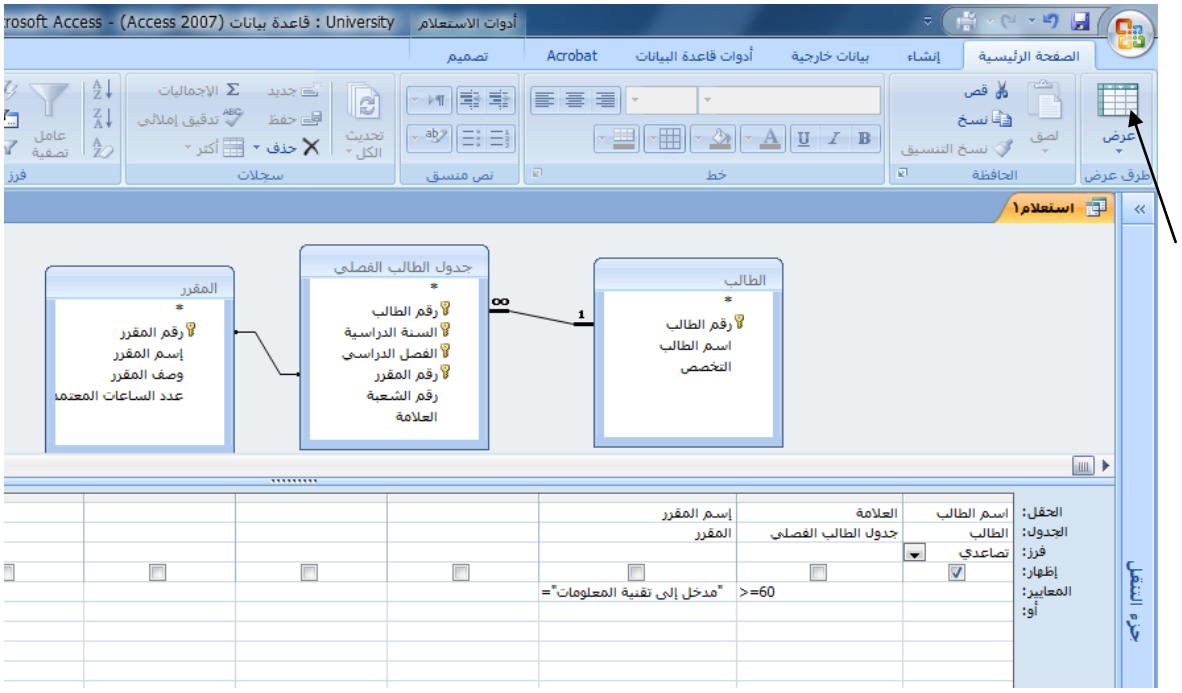
الحل

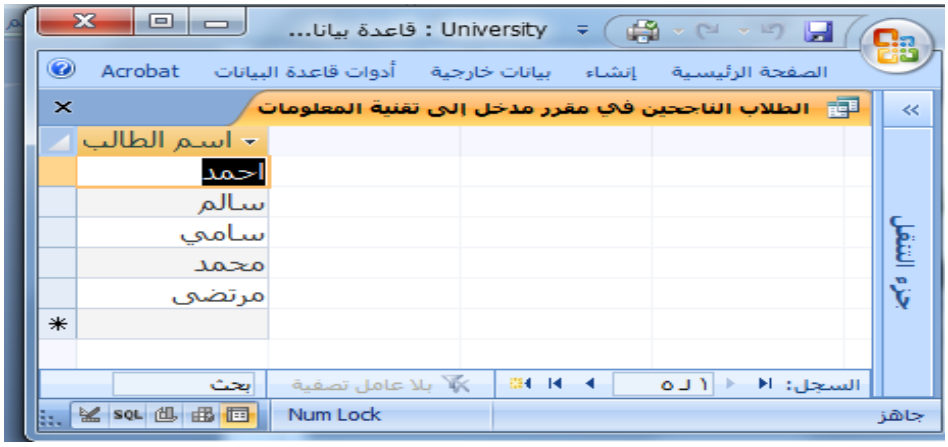


IbtiHAlinO

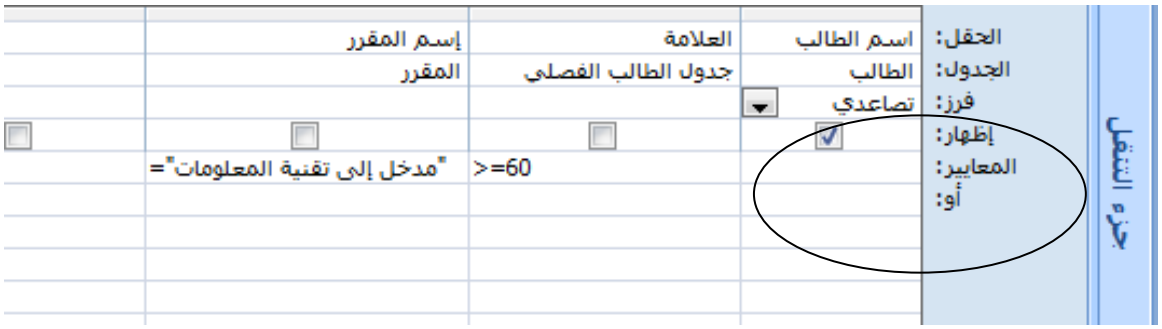

Microsoft Access - (Access 2007) : University
أدوات الاستعلام
الصفحة الرئيسية
إنشاء
بيانات خارجية
أدوات قاعدة البيانات
Acrobat
تصميم
استعلام1
المقرر
رقم المقرر
إسم المقرر
وصف المقرر
عدد الساعات المعتمد
جدول الطالب الفصلي
رقم الطالب
السنة الدراسية
الفصل الدراسي
رقم المقرر
رقم الشعبة
العلامة
الطالب
رقم الطالب
اسم الطالب
التخصص
∞
1
الحقل:
الجدول:
فرز:
إظهار:
المعايير:
أو:
جزء التنقل
جاهز
Num Lock
EN
03:49 م
٢٣/٠١/٠٨


Microsoft Access - (Access 2007) : University
أدوات الاستعلام
الصفحة الرئيسية
إنشاء
بيانات خارجية
أدوات قاعدة البيانات
Acrobat
تصميم
استعلام1
المقرر
رقم المقرر
إسم المقرر
وصف المقرر
عدد الساعات المعتمد
جدول الطالب الفصلي
رقم الطالب
السنة الدراسية
الفصل الدراسي
رقم المقرر
رقم الشعبة
العلامة
الطالب
رقم الطالب
اسم الطالب
التخصص
∞
1
الحقل:
الجدول:
فرز:
إظهار:
المعايير:
أو:
جزء التنقل
جاهز
Num Lock
EN
03:50 م
٢٣/٠١/٠٨


أدوات الاستعلام
الصفحة الرئيسية
إنشاء
بيانات خارجية
أدوات قاعدة البيانات
Acrobat
تصميم
استعلام1
المقرر
رقم المقرر
إسم المقرر
وصف المقرر
عدد الساعات المعتمد
جدول الطالب الفصلي
رقم الطالب
السنة الدراسية
الفصل الدراسي
رقم المقرر
رقم الشعبة
العلامة
الطالب
رقم الطالب
اسم الطالب
التخصص
∞
1
الحقل:
الجدول:
فرز:
إظهار:
المعايير:
أو:
اسم الطالب
الطالب
تصاعدي
العلامة
جدول الطالب الفصلي
>=60
إسم المقرر
المقرر
"مدخل إلى تقنية المعلومات"=
جزء التنقل



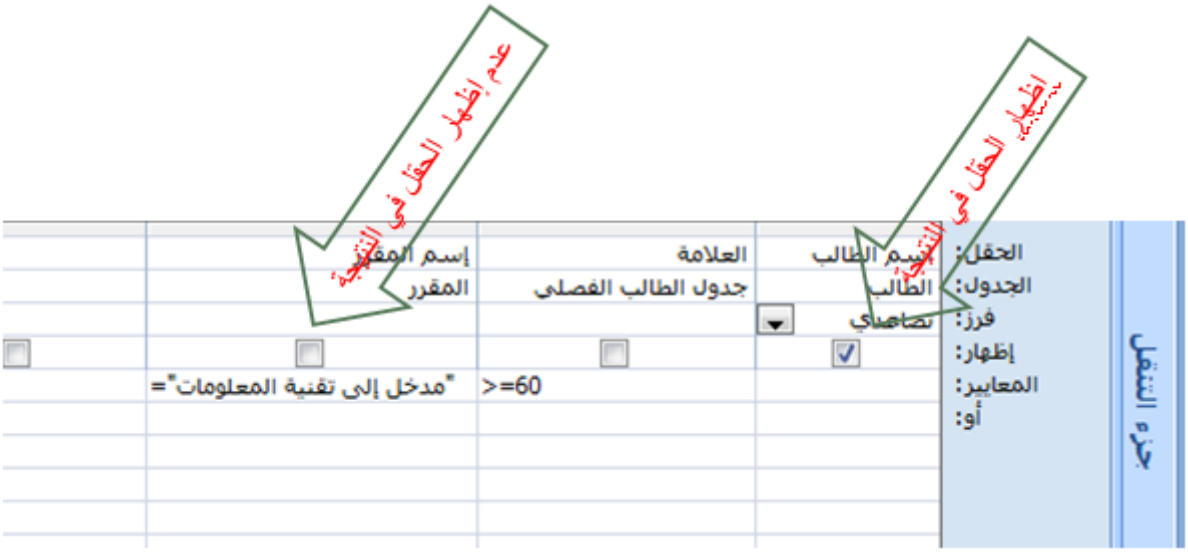
اذا يجدر التنويه هنا إلى أن التركيز على الشرط، وإذا كان هناك أكثر من شرط فيجب التركيز على العلاقة بينهما "و" AND أم "أو" OR



- **العلاقات المنطقية :** أكبر >، أصغر <، يساوي =، لا يساوي <>، اكبر او يساوي=>، أصغر أو يساوي =<
- **العمليات المنطقية :** NOT،OR،AND
- **العمليات على الفترة :** BETWEEN
- **العمليات على السلاسل الرمزية :** LIKE ، بالاضافة لاستخدام رمز التصفية و البحث '*'، '$'

س: لو طلب منك اسماء الطلاب الذين تاريخ ميلادهم في العام ١٩٨٥، ما هو الشرط على التاريخ
جواب ١: LIKE "1985*"
جواب٢: >= #1/1/1985# AND <= #31/12/1985#
جواب٣: BETWEEN (#1/1/1985# AND #31/12/1985#)
جواب ٤: LIKE "*/*/1985"
س: لو طلب منك اسماء الطلاب الذين تبدأ أسماءهم بحرف "م"
جواب : LIKE "م*"
س: لو طلب منك اسماء الطلاب الذين ثاني حرف من إسمهم "م"
جواب: LIKE "*م$"
س: لو طلب منك اسماء الطلاب الذين ثاني حرف من إسمهم "م" ، و رابع حرف "د"
جواب: LIKE "*د$م$"
س: لو طلب منك اسماء الطلاب الناجحين في مقرر "مدخل" و يسكنون "عمان"
جواب : هنا شرطين مربوطين بعلاقة "و" ، اذا نكتب الشرطين في نفس سطر المعايير (criteria)، كل حسب حقله

- شرط النجاح(>=60)على حقل العلامة
- شرط العنوان ("عمان" LIKE)على حقل العنوان



مثال تطبيقي

- إستخدم برنامج إدارة قواعد البيانات :مايكروسوفت أكسس ٢٠٠٧، لبناء قاعدة البيانات الخاصة بطلاب جامعة، وقم بتخزين قاعدة البيانات تحت إسمك ورقمك الأكاديمي على سطح المكتب.
- تحتوي قاعدة البيانات على ما يلي

1) **جدول "الطالب"، الذي يتكون من الحقول التالية:**
   - رقم الطالب (رقم ، رقم صحيح طويل) ، مفتاح أساسي
   - إسم الطالب (نص، ٢٥ حرف)
   - إسم القسم (نص، ٣٠ حرف)، إستخدم معالج البحث لتخزين القيم (علوم حاسب،نظم معلومات)
   - تاريخ الميلاد(تاريخ/وقت، تاريخ قصير)
   - المدينة (نص، ٣٠ حرف)، إستخدم معالج البحث لتخزين القيم (الخالدية ، الهفوف ، الدمام)

رقم الطالب	إسم الطالب	إسم القسم	تاريخ الميلاد	المدينة
١٠	أحمد	علوم حاسب	١٩٩٠/١٧/٢	الخالدية
٢٠	أحمد	علوم حاسب	١٩٩٠/١٥/٣	الهفوف
٣٠	سالم	نظم معلومات	١٩٩١/٢١/٣	الدمام
٤٠	محمد	علوم حاسب	١٩٩١/٢٢/٢	الهفوف
٥٠	فهد	نظم معلومات	١٩٨٩/١٣/٥	الدمام
٦٠	خالد	علوم حاسب	١٩٨٩/١٤/٩	الخالدية

2) **جدول "المقرر"، الذي يتكون من الحقول التالية:**
   - رقم المقرر(رقم ، رقم صحيح طويل) ، مفتاح أساسي
   - إسم المقرر(نص، ٤٠ حرف)
   - رقم المتطلب السابق(رقم ، رقم طويل)

رقم المقرر	إسم المقرر	رقم المتطلب السابق
٥١١	لغة برمجة سي	
٥٢١	المفاهيم الشيئية ١	٥١١
٥٢٢	المفاهيم الشيئية ٢	٥٢١
٦١١	مبادىء قواعد البيانات	٥١١
٦٢١	قواعد بيانات متقدمة	٦١١

3) **جدول "العلامة"، الذي يتكون من الحقول التالية:**
   - رقم الطالب (رقم ،رقم صحيح طويل)، مفتاح أساسي
   - رقم المقرر (رقم ،رقم صحيح طويل)، مفتاح أساسي
   - العلامة (رقم ، بايت)

رقم الطالب	رقم المقرر	العلامة
١٠	٥١١	٧٠
٦٠	٥١١	٥٠
٢٠	٥١١	٦٠
٢٠	٥٢١	٨٠
٢٠	٦١١	٩٠
٣٠	٥٢٢	٨٧
٥٠	٥١١	٨٨
٣٠	٦٢١	٦٠
٤٠	٥١١	٥٥

٤) قم ببناء العلاقات التالية بين الجداول:
- جدول الطالب مع جدول العلامة بواسطة الحقل المشترك رقم الطالب.
- جدول المقرر مع جدول العلامة بواسطة الحقل المشترك رقم المقرر.

5) قم بتصميم تقرير يعرض جميع حقول جدول الطالب

6) قم بتصميم نموذج يعرض جميع حقول جدول المقرر

7) قم بتصميم استعلام يعرض الحقول (رقم الطالب، إسم الطالب، إسم القسم) بشرط أن يكون تاريخ ميلاد الطالب في شهر ٢ من العام ١٩٩٠.

8) إذا علمت أن مواد قسم نظم المعلومات يبدأ رقمها من اليسار بالرقم ٦، قم بتصميم تقرير يعرض الحقول(رقم المقرر، إسم المقرر) بشرط أن تكون هذه المقررات تابعة لقسم نظم المعلومات.

9) إذا علمت أن علامة النجاح في مقرر ما هي ٦٠ فما فوق، قم بتصميم نموذج يعرض الحقول(رقم الطالب،إسم الطالب) للطلبة الراسبين في مقرر رقم ٥١١.

مايكروسوفت أكسس ٢٠٠٧

- النصيحة التي نود توجيهها للطالب فيما يتعلق بجزئية الأكسس في هذا المقرر :
- بالرغم من أن الاختبار النهائي نظري ، إلا أن التطبيق العملي له الأثر الكبير في تعميق فهم الطالب ، و ترسيخ المعلومة و عليه فإن التطبيق العملي يساعد الطالب على الإجابة على أسئلة الإختبار و إن كانت نظرية
- أسئلة إختبار الأكسس قد تتضمن شاشات من التطبيق ، ومن ثم السؤال عنها
- نسأل الله لكم التوفيق في هذا المقرر و كل مقررات الإنتساب

تصميم قواعد البيانات العلاقية
مراجعة عامة

دورة حياة قاعدة البيانات Database Life Cycle (DBLC)

إن عملية تطوير قاعدة البيانات تمر بمجموعة من المراحل،هذه المراحل المتتالية تسمى بدورة حياة قاعدة البيانات.
هذه المراحل أو دورة الحياة تمر بصورة متزامنة ضمن مراحل دورة حياة نظام المعلومات، كما يوضح الشكل التالي

تتكون دورة حياة قاعدة البيانات من المراحل التالية:

١. تحديد المواصفات و المتطلبات الخاصة بقاعدة البيانات ، وهي مرحلة جزئية ضمن جمع مواصفات و متطلبات نظام المعلومات في مرحلة التحليل.
٢. إعداد قاعدة البيانات الأولية ، و فيها يتم تصميم نموذج اولي للبيانات بواسطة مخططات الكيان العلاقة (E-RD)
٣. تصميم قاعدة البيانات المنطقية ، تحويل قاعدة البيانات الأولية أو مخطط الكيان/العلاقة إلى مخطط الاسكيما ، وذلك بإتباع قواعد التحويل

مخطط قواعد البيانات Database Schema :

- هو مخطط يصف قاعدة البيانات بشكل رسومي تمهيدا لبنائه على شكل جداول في نظام إدراة قواعد بيانات DBMS
- هو مخطط ينتج عن عملية إخضاع مخطط الكيان العلاقة لخوارزمية التحويل Mapping Algorithm

التحويل من مخطط الكيان العلاقة إلى مخطط قواعد بيانات Mapping ERD to DB schema :

تتم عملية تحويل مخطط ERD ، بتطبيق مجموعة من الخطوات البسيطة ، تسمى خوارزمية التحويل Mapping Algorithm وتتكون هذه الخطوات من جميع الحالات البسيطة المحتملة التي قد تكون موجودة في النموذج الأولى و يتم تطبيق هذه الخوارزمية كاملة مع تجاوز الحالات التي لم تظهر في النموذج الأولي

- **خوارزمية التحويل (Mapping Algorithm):**

١. تحويل الكيانات العادية (القوية)
٢. تحويل الكيانات الضعيفة
٣. تحويل العلاقات الثنائية من النوع 1:1
٤. تحويل العلاقات الثنائية من النوع 1:N
٥. تحويل العلاقات الثنائية من النوع N:M
٦. تحويل الصفات متعددة القيم
٧. تحويل العلاقات فوق الثنائية

١. تحويل أنواع الكيانات العادية : يتم هنا تحويل جميع الكيانات العادية ، أي الكيانات غير الضعيفة ، بإنشاء جدول يتكون من الحقول التي تقابل صفات ذلك الكيان. و يتم تحديد أحد مفاتيح الكيان ، و تسميته بالمفتاح الرئيسي primary key(PK) و إذا كانت الصفة التي تمثل المفتاح من النوع المركب فإن المفتاح الرئيسي سيكون مجموعة الحقول التي تنشأ من الصفة المركبة.

٢. تحويل الكيانات الضعيفة : يتم تحويل كل واحدة من الكيانات الضعيفة ، بإنشاء جدول يتكون من الحقول التي تقابل صفات ذلك الكيان ، كما يجب إضافة المفتاح الرئيسي للكيان القوي الذي يتبعه ذلك الكيان الضعيف ، و يكون المفتاح الرئيسي PK للجدول الجديد ، عبارة عن مفتاح مركب مكون من المفتاح الأجنبي FK بالإضافة إلى المفتاح الجزئي (Partial Key) الخاص به.

٣. تحويل العلاقات الثنائية من النوع (1:1): إذا كانت العلاقة بين الكيانين علاقة واحد- إلى- واحد فإن عملية التحويل تتم وفق عدة خيارات أشهرها ، خيار يسمى بطريقة المفتاح الأجنبي ، و فيه يتم إضافة المفتاح الرئيسي لأحد الجدولين إلى الجدول الآخر كمفتاح أجنبي و يفضل أن يكون الجدول الذي يحتوي على المفتاح الأجنبي ، هو الجدول الذي يكون نوع قيد اشتراكه في العلاقة من نوع (الاشتراك الكلي)

٤. تحويل العلاقات الثنائية من النوع (1:N): يتم هنا إنشاء جدولين لتمثيل الكيانين المرتبطين على أن يتم تطبيق طريقة المفتاح الأجنبي السابقة ، وذلك بإضافة المفتاح الرئيسي للجدول من جهة العلاقة (N) إلى الجدول الآخر المرتبط بالعلاقة (١) ، بغض النظر عن نوع قيد الاشتراك.

٥. تحويل العلاقات الثنائية من النوع (N:M): في هذا النوع من العلاقات ، يتم استحداث جدول جديد ، فيكون الناتج من هذه العلاقة ثلاثة جداول ، جدولين لتمثيل الكيانين المرتبطين بالعلاقة و يضم الجدول الثالث حقلين كمفتاحين أجنبيين يمثلان المفتاحين الرئيسيين في الجدولين ، و يمكن إضافة أي حقل آخر يكون له مغزى ، كأن تكون العلاقة لها صفة بذاتها ، فتتحول الصفة إلى حقل في الجدول الجديد.

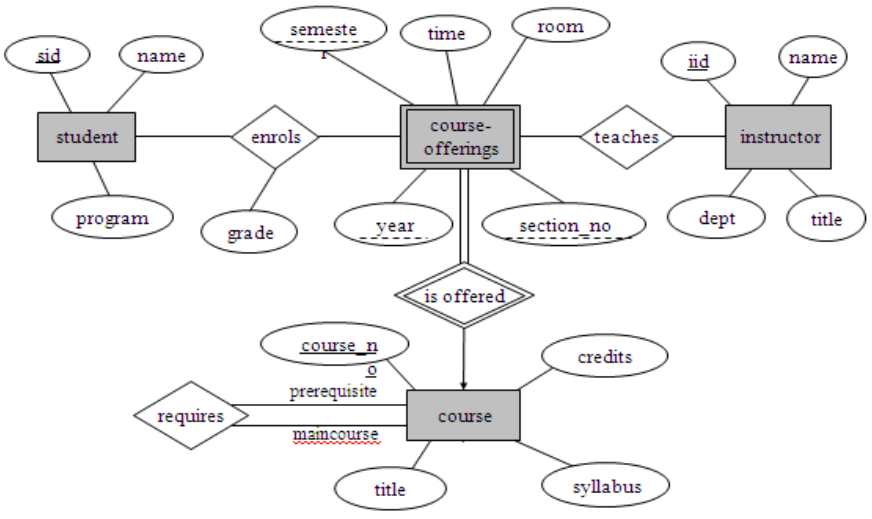
٦. تحويل الصفات متعددة القيم : يتم في هذه الحالة ، عادة ، إنشاء جدول جديد يضم الصفة المتعددة القيم كحقل ، و يضاف إلى الجدول مفتاح أجنبي FK يكون ممثلا للمفتاح الرئيسي في الجدول الناتج من الكيان الذي يحتوي على الصفة متعددة القيم . أما الصفات المركبة فتتحول إلى صفات بسيط ، فحقول عادية كما أوضحنا أعلاه و الصفات ذات القيم المشتقة تلغى من الجدول ، لأنها صفات قابلها للاشتقاق من صفات أخرى ، فلا داعي لوجودها

٧. تحويل العلاقات غير الثنائية ، كالعلاقة الثلاثية وما فوقها : في حالات نادرة تظهر لدينا علاقات معقدة ، كالعلاقة الثلاثية (بين ثلاثة كيانات) و الرباعية وما فوقها و تعالج هذه الحالة بطريقة معالجة الحالة الخامسة (حالة تحويل العلاقات الثنائية من النوع (N:M) حيث يتم إنشاء جدول جديد و إضافة المفاتيح الرئيسية للجداول المشتركة ، حسب عددها إلى الجدول الجديد كمفاتيح أجنبية مكونة بمجموعها ، مفتاحا مركبا يمثل المفتاح الرئيسي للجدول.

تمرين:

حول كل من نماذج الكيان العلاقة التاليين إلى ما يقابلهما من مخطط قواعد البيانات Database Schema

تحويل مخطط الكيان العلاقة لجامعة إلى ما يقابله من مخطط قواعد البيانات (جداول)



E-R diagram for a university.

١. نبدأ بتحويل الكيان العادي :

- يتم تمثيل الكيان العادي (القوي) بشكل المستطيل أحادي الإطار و يحتوي مخطط الكيان العلائقي السابق على ثلاث كيانات هي :

- الطالب (Student) - المحاضر (Instructor) - المقرر (Course)

- وتتم عملية تحويل الكيانات القوية بتمثيل كل منها بجدول يحمل إسم الكيان ، وي حتوي حقولا تمثل الصفات (تمثل الصفات بالشكل البيضاوي) المرتبطة (الارتباط يمثل بخط مستقيم) بالكيان.
- في حالة الصفة المركبة (شكل بيضاوي مرتبط باشكال بيضاوية جزئية) يتم أخذ الأجزاء المكونة للصفة المركبة.
- يتم تجاهل الصفة المشتقة (تمثل بشكل بيضاوي متقطع الاطار) بسبب القدرة على إشتقاقها بجملة إستعلام.
- أما الصفة متعددة القيمة (تمثل بشكل بيضاوي مزدوج الإطار) فيتم إنشاؤها في جدول مستقل يحمل إسم الكيان و الصفة متعددة القيمة و يحتوي حقولا تمثل الصفة متعددة القيمة وصفة المفتاح الرئيسي للكيان.

- نلاحظ في هذا المثال أن كل الصفات من النوع البسيط
- و يكون المفتاح الرئيسي (Primary Key) للجدول هو مجموعة صفات المفتاح الرئيسي المرتبطة بالكيان

و ينتج عن عملية التحويل الجداول التالية :

STUDENT	sid	name	program

INSTRUCTOR	iid	name	dept	title

COURSE	courseno	title	syllabus	Credits

٢. تحويل الكيان الضعيف:

- يتم تمثيل الكيان الضعيف بشكل مستطيل مزدوج الإطار ، و سبب ضعف الكيان ، ينتج من عدم وجود صفة مفتاح رئيسي له ، ولكن يحتوي على صفة مفتاح جزئي ، ولدينا في هذا المثال كيان ضعيف واحد هو : كيان الشعب الفصلية المقترحة (Course-Offerings)

و يحتوي هذا الكيان على صفات المفتاح الجزئي التالية:

أ - السنة(Year)
ب - الفصل(Semester)
ج - رقم الشعبة(Section-no)

- و يجب أن يرتبط الكيان الضعيف بكيان قوي بواسطة علاقة تعريف (تمثل علاقة التعريف بشكل معين مزدوج الإطار)، وذلك في سبيل تقوية الكيان الضعيف.
- و تتم عملية تحويل الكيان الضعيف بتحويله إلى جدول يحمل إسم الكيان الضعيف و يحتوي حقولا من الصفات المرتبطة به ، بالاضافة إلى حقل المفتاح الرئيسي من جدول الكيان القوي المرتبط معه بعلاقة تعريف ، وفي حال وجود أي صفة على علاقة التعريف ، يتم تمثيلها بحقل في الجدول. و يكون المفتاح الرئيسي للجدول هو مفتاح الكيان القوي بالإضافة الى المفاتيح الجزئية في الكيان الضعيف.

و ينتج عن عملية التحويل الجدول التالي :

COURSE-OFFERINGS	courseno	secno	year	semester	time	room

و بذلك يرتبط جدول الشعب الفصلية المقترحة (Course-Offerings) بجدول المقرر (Course) بوجود المفتاح الأجنبي (Foreign Key) رقم المقرر (courseno)

COURSE	courseno	title	syllabus	Credits

COURSE-OFFERINGS	courseNo	secno	year	semester	time	room

٣.تحويل العلاقات :

- يتم تمثيل العلاقة بشكل معين أحادي الإطار
- في هذه الحالة يتم التعامل مع العلاقة حسب نوعها كل على حده ، حيث تصنف العلاقات إلى الأنواع التالية:

أ - علاقة واحد إلى واحد (One-to-One Relationship)
ب -علاقة واحد إلى كثير (One-to-Many)
ج -علاقة كثير إلى كثير(Many-to-Many)

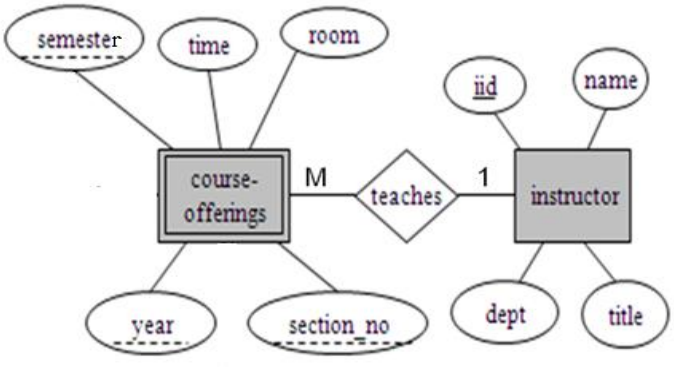
- يتم تجاهل علاقة التعريف (شكل معين مزدوج الإطار)، لانه تم بناؤها مسبقا عند تحويل الكيان الضعيف

٣.أ) تحويل علاقة واحد إلى واحد:

- إذا كانت العلاقة بين الكيانين علاقة واحد – إلى - واحد فإن عملية التحويل تتم وفق عدة خيارات أشهرها ، خيار يسمى بطريقة المفتاح الأجنبي ، و فيه يتم إضافة المفتاح الرئيسي لأحد الجدولين إلى الجدول الآخر كمفتاح أجنبي و يفضل أن يكون الجدول الذي يحتوي على المفتاح الأجنبي ، هو الجدول الذي يكون نوع قيد اشتراكه في العلاقة من نوع (الاشتراك الكلي).وفي حال وجود صفة على العلاقة يتم إضافتها كحقل إضافي مع المفتاح الأجنبي.
- في هذا المثال لا توجد علاقة من النوع واحد- إلى- واحد

٣.ب) تحويل علاقة واحد إلى كثير:

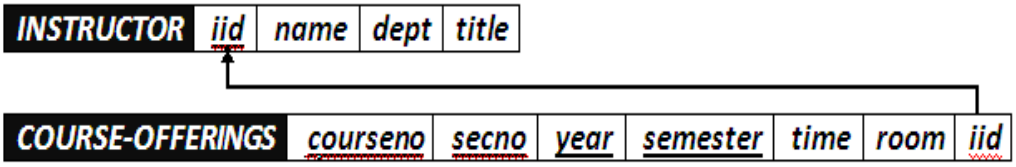
في هذه الحالة يتم أخذ نسخة من المفتاح الرئيسي من الجدول ذو طرف العلاقة واحد ويتم إضافته كحقل مفتاح أجنبي في جدول طرف العلاقة كثير. وفي حال وجود صفة على العلاقة يتم إضافتها كحقل إضافي مع المفتاح الأجنبي.



- في المثال أعلاه لدينا علاقة يُدرس (teaches) بين كيان المحاضر (Instructor) و كيان الشعب الفصلية المقترحة (Course-Offerings).
- وعليه يتم أخذ المفتاح الرئيسي (iid) من جدول Instructor كونه طرف العلاقة واحد ، و يضاف كمفتاح أجنبي في جدول course-offerings ، و بالتالي يتم تعديل جدول Course-Offerings ليصبح بالشكل التالي :

COURSE-OFFERINGS	courseno	secno	year	semester	time	room	iid

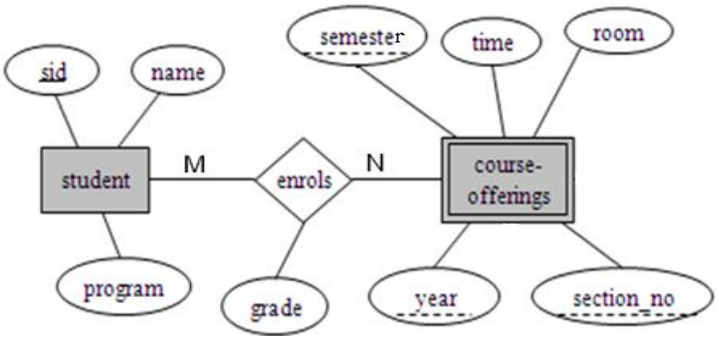
و بذلك يرتبط جدول الشعب الفصلية المقترحة (Course-Offerings) بجدول المحاضر (Instructor) بوجود المفتاح الأجنبي رقم المحاضر (iid)



٣.ج) تحويل علاقة كثير إلى كثير:

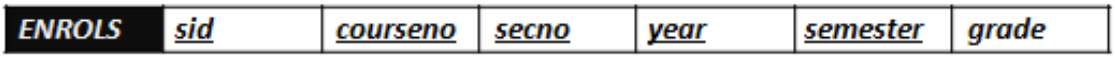
- في هذا النوع من العلاقات يتم إنشاء جدول جديد يحمل إسم العلاقة ، وتكون حقوله هي حقول المفتاح الرئيسي من كلا الجدولين المشاركين في العلاقة مكونة المفتاح الرئيسي للجدول الجديد ، وفي حال وجود صفة على العلاقة يتم إضافتها كحقل إضافي في الجدول

في هذا المثال ، يوجد لدينا علاقتين من نوع كثير إلى كثير، هما :

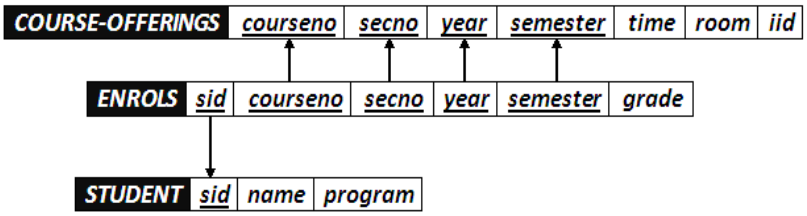


١. علاقة يُسَجِّل (enrols) :

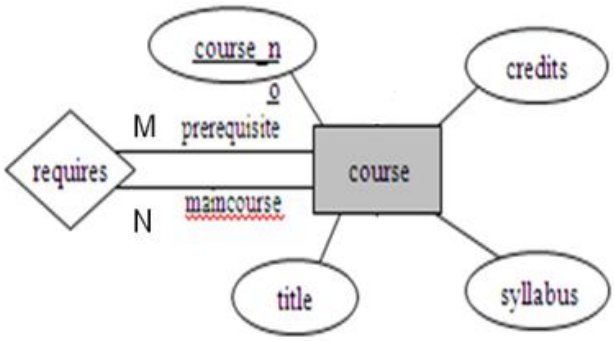
ينتج عن هذه العلاقة الجدول التالي:



و بذلك يرتبط جدول الشعب الفصلية المقترحة (Course-Offerings) بجدول الطالب (Student) و جدول يُسجل (Enrolls) بوجود المفتاح الأجنبي رقم الطالب (sid) من جدول الطالب و المفتاح الأجنبي (courseno, secno, semester, year) من جدول الشعب الفصلية المقترحة (Course-Offerings)



٢.علاقة يتطلب (Requires) :



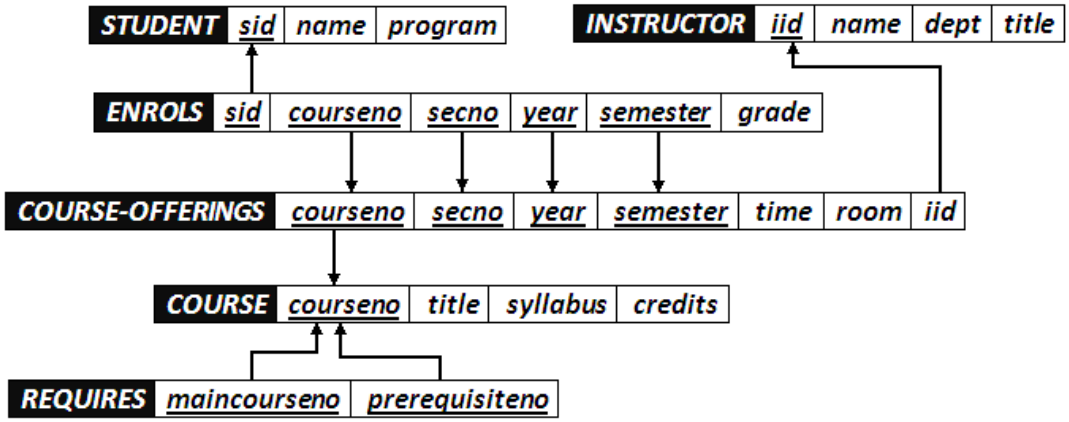
هذه العلاقة تسمى علاقة تغذية راجعة (Recursive Relationship) أو علاقة كيان مع نفسه ، و يمكن تصنيفها كعلاقة أحادية ذات تغذية راجعة ، وفي هذا النوع من العلاقات يتم وضع التسمية المقترحة لحقول الجدول الناتج كون الحقول ناتجة عن حقل واحد هو حقل المفتاح الرئيسي ، و بالتالي يجدر بنا إعادة التسمية لتفادي الخطأ في تشابه الإسم .

وينتج عن هذه العلاقة الجدول التالي :

REQUIRES	maincourseno	prerequisiteno

و بذلك يرتبط جدول يتطلب(Requires) مع جدول المقرر(course)

و بذلك نكون قد أنشأنا مخطط قواعد البيانات الناتج عن التحويل للمثال رقم (١)، وهو على الشكل التالي :



نلاحظ الارتباط الوثيق بين جداول قاعدة البيانات ، فلا يجوز أن يكون هناك جدول دون علاقة بباقي الجداول أو بمعزل عن الجداول الأخرى في قاعدة البيانات .

دعواااتكم .. ابتهالينو