قواعد معطيات

المعهد التقاني للحاسوب بحماة
قسم الشبكات

المحاضرة رقم 1

مقدمة في قواعد البيانات

Eng. Hana Sabbagh

في هذ المحاضرة

- مقدمة
- مصطلحات هامة
- مكونات قواعد البيانات
- مكونات نظام قواعد البيانات
- أنواع تطبيقات قواعد البيانات
- مثال حول قاعدة البيانات
- مستخدمي قواعد البيانات
- خصائص قواعد البيانات
- مزايا نظام إدارة قواعد البيانات
- التطور التاريخي لقواعد البيانات
- استخدام انظمة إدارة قواعد البيانات

مقدمة

- أصبح مصطلح قواعد البيانات من أهم مفردات المجتمعات المتقدمة هذه الأيام.
- أصبحت تطبيقات قواعد البيانات جزءا هاما من مكونات أي عملية تجارية أو اقتصادية أو سياسية أو تعليمية
- حجر الأساس لتبادل المعلومات في الانترنت،
- مكون رئيسي في أي نظام معلومات محوسب، مستفيدة من التطور المناظر في تقنيات الاتصال وشبكات الحاسوب،

مصطلحات هامة

- **قواعد البيانات (Database):** هي مجموعة مشتركة من البيانات المترابطة والمتجانسة منطقيا،والتي صممت كي تلبي الاحتياج المستمر للمعلومات لمنظمة معينة.

- **Database is a Shared collection of logically related data (and a description of this data),designed to meet the information needs of an organization.**

- **البيانات(Data):** هي حقائق (خام) معروفة يمكن أن تسجل ولها معنى ضمني(implicit)

- **العالم المصغر(Mini-World) :** هو الجزء من العالم الحقيقي الذي نخزن البيانات عنه في قاعدة البيانات ، مثلا الجزء المتعلق بدرجات طالب في النظام الجامعي.

مصطلحات هامة

- **نظام إدارة قواعدالبيانات (Database Management System)(DBMS):**
- هو النظام البرمجي الذي يسهل إنشاء وإدارة وصيانة قواعد البيانات المحوسبة.

- **نظام قواعد البيانات(Database System):** هو عبارة عن برمجيات نظام إدارة قواعد البيانات بالإضافة إلى البيانات نفسها ، بالإضافة إلى التطبيقات التي يمكن ربطها بها،والمستخدمون لها.
- **تعريفات البيانات(Meta-Data):**وهي البيانات المخزنة عن البيانات، أو هي معلومات عن بنية البيانات وليس البيانات نفسها.

مكونات قواعد البيانات

1. **تعريفات البيانات(Data definitions)**

وهي البيانات المخزنة عن البيانات، أو هي معلومات عن بنية البيانات وليس البيانات نفسها، وتسمى أيضا ب meta-data.

2. **قواعد البيانات المخزنة(Stored database)**

وهي البيانات نفسها، التي تفصل عن تعريفاتها، من أجل مرونة إجراء اي تعديل او تحديث للبيانات، مما يقلل من زمن المعالجة ومن مساحة التخزين، أكثر مما يمكن.

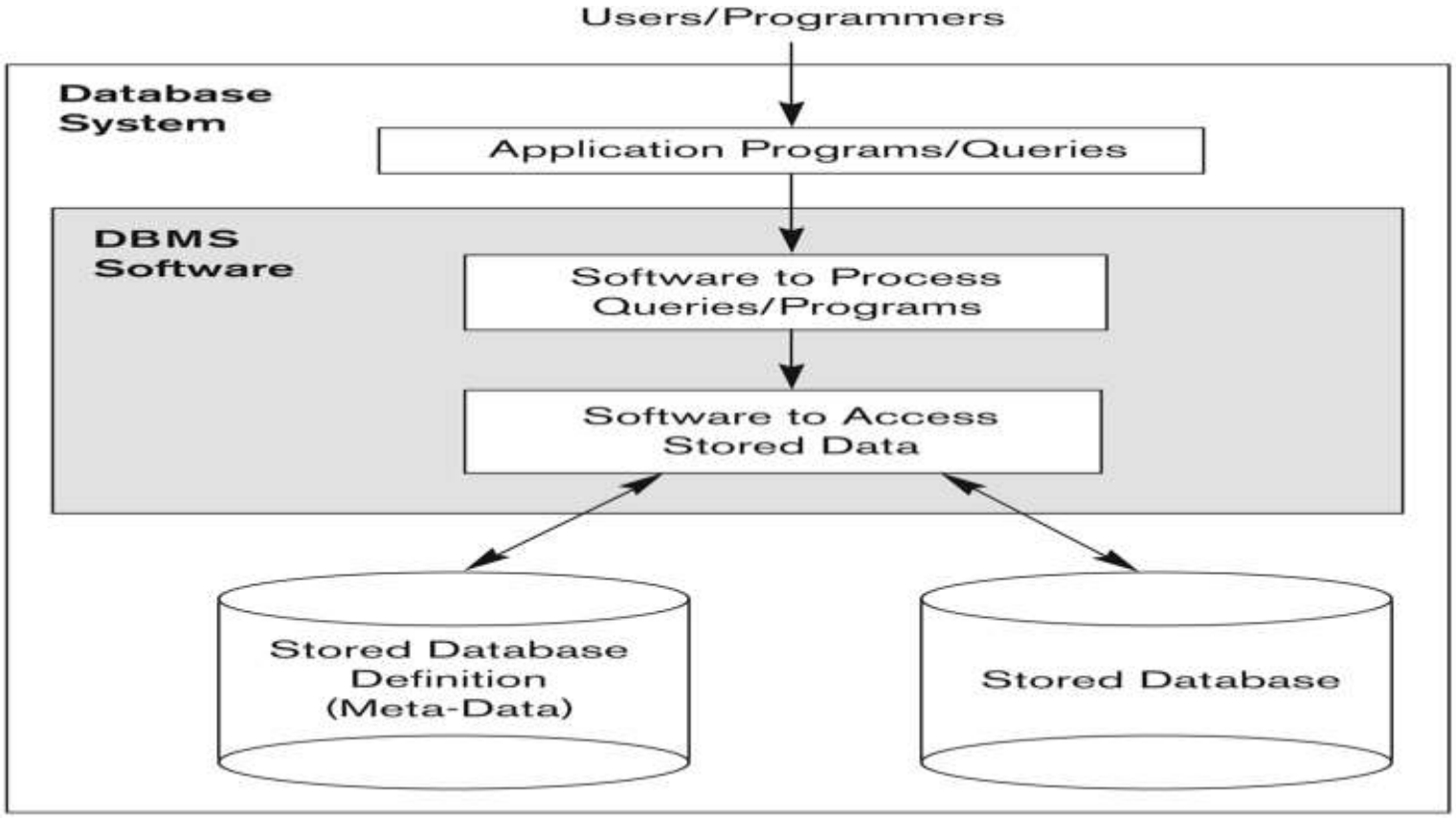
مكونات نظام قواعد البيانات

يتكون نظام قواعد البيانات(Database System) من:

1. المستخدمين والمبرمجين(Users / programmers)
2. التطبيقات البرمجية/ الاستعلامات(Application programs / Queries)
3. نظام إدارة قاعدة البيانات(DBMS)
4. قاعدة البيانات نفسها

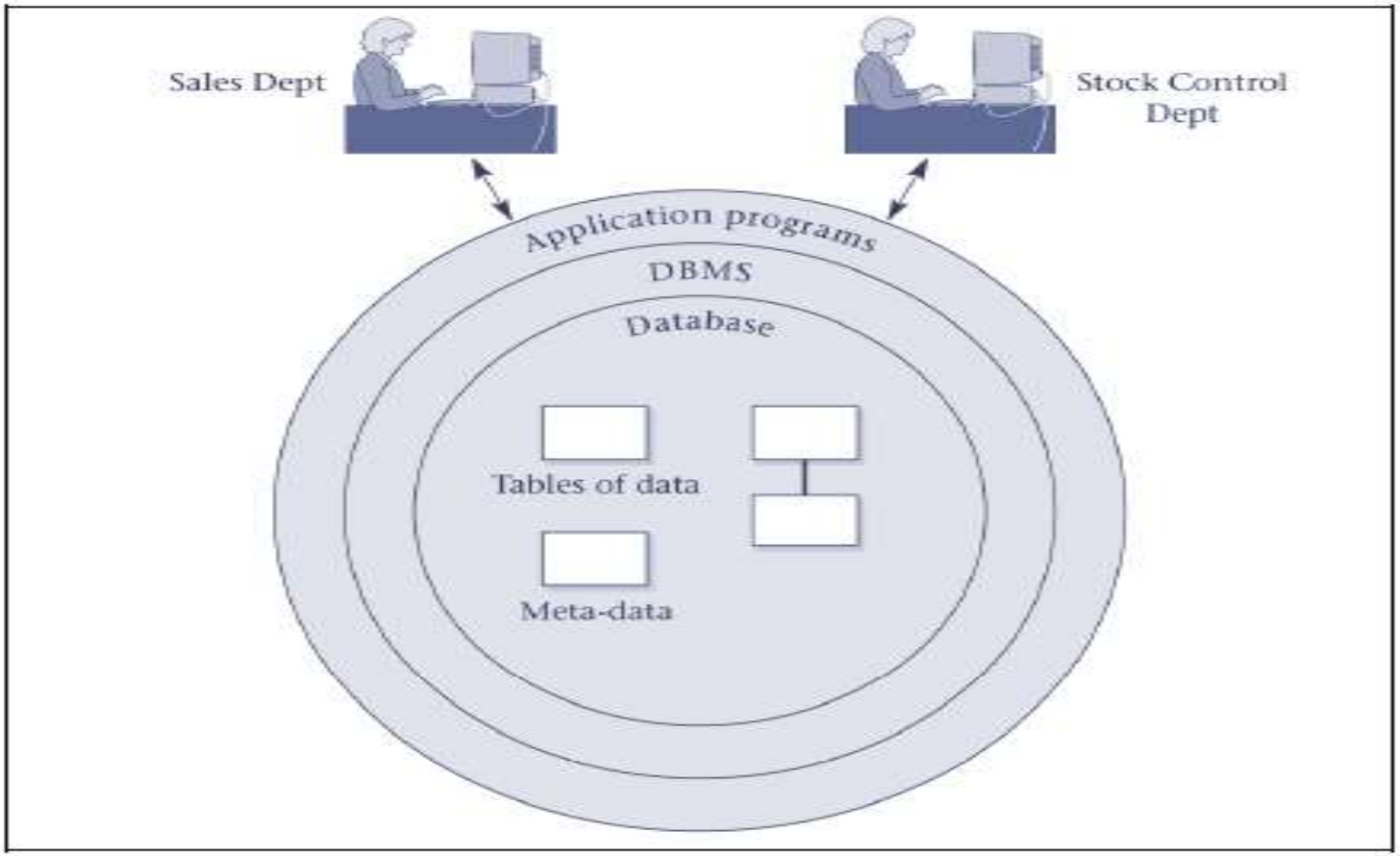
مكونات نظام قواعد البيانات

- **نموذج(1) يوضح نظام قواعد البيانات:**



مكونات نظام قواعد البيانات

- **نموذج(2) يوضح نظام قواعد البيانات:**



أنواع تطبيقات قواعد البيانات

تصنف تطبيقات قواعد البيانات إلى:

1. **التطبيقات التقليدية لقواعد البيانات Traditional Applications**

وهي تطبيقات مضى على وجودها ما يقارب نصف قرن، ومع ذلك فما زال الاحتياج لها مستمرا، ومازالت تقنياتها تتطوريوميا ، ومن أمثلتها قواعد البيانات النصية والرقميةNumeric and Textual Databases، أي قواعد البيانات التي تحتوي على أرقام ونصوص، تخزن فيها، ثم تعالج وتخرج كتقارير حسب التطبيق المستخدم.

أنواع تطبيقات قواعد البيانات

تصنف تطبيقات قواعد البيانات إلى:

2. **التطبيقات الحديثة لقواعد البيانات(More Recent Applications):**

وهي التطبيقات التي ظهرت كتطور طبيعي للتطبيقات التقليدية، ونتجت بعد الاحتياج لصيغ أكثر تعقيدا لخزن البيانات،وتقنياتها تعتمد بالطبع على التقنيات التقليدية، ولكنها أكثر تعقيدا وتحتاج إلى حاسبات أكثر تعقيدا، كما أنها تستخدم في المؤسسات والشركات العملاقة، والدول ومؤسساتها الرسمية.

أنواع تطبيقات قواعد البيانات

2. **التطبيقات الحديثة لقواعد البيانات(More Recent Applications):**

وهي تطبيقات كثيرة نذكر منها على سبيل المثال:

أ- قواعد بيانات الوسائط المتعددة(Multimedia Databases)

ب- نظم المعلومات الجغرافية (Geographic Information Systems (GIS))

ج- مخازن أو مستودعات البيانات(Data Warehouses)

د- قواعد البيانات النشطة وأنظمة الوقت الحقيقي

(Real-time and Active Databases)

مثال حول قاعدة البيانات

- العالم المصغر للمثال هو جزء من بيئة جامعة، وعند تصميم النموذج الأولي للبيانات (conceptual data model) ، تظهر لنا مجموعة من المكونات، حيث يسمى كل مكون بالكينونة entity ، وبتجميع هذه الكينونات وخصائصها نحصل على قاعدة البيانات المطلوبة

مثال حول قاعدة البيانات

- ولدينا هنا بعض كينونات العالم المصغر: طالب، مقرر دراسي، شعب مقرر، قسم علمي، محاضر ..الخ

طلاب	STUDENTs
مقررات	COURSEs
شعب من المقررات	SECTIONs(of COURSEs)
أقسام علمية	DEPARTMENTs(academic)
محاضرين	INSTRUCTORs

مثال حول قاعدة البيانات

يتكون النموذج الأولي للبيانات، بالإضافة إلى الكينونات وصفاتها، من مجموعة من العلاقات، والتي تمثل علاقاتها الحقيقية في الواقع:

- الشعبة تتبع مقرر دراسي .
- المدرس يدرس مقرر دراسي .
- الطالب يسجل في شعبة مقرر .
- المقرر الدراسي يقدم بواسطة قسم علمي .
- الخ...

مثال حول قاعدة البيانات

COURSE

Course_name	Course_number	Credit_hours	Department
Intro to Computer Science	CS1310	4	CS
Data Structures	CS3320	4	CS
Discrete Mathematics	MATH2410	3	MATH
Database	CS3380	3	CS

SECTION

Section_identifier	Course_number	Semester	Year	Instructor
85	MATH2410	Fall	04	King
92	CS1310	Fall	04	Anderson
102	CS3320	Spring	05	Knuth
112	MATH2410	Fall	05	Chang
119	CS1310	Fall	05	Anderson
135	CS3380	Fall	05	Stone

GRADE_REPORT

Student_number	Section_identifier	Grade
17	112	B
17	119	C
8	85	A
8	92	A
8	102	B
8	135	A

PREREQUISITE

Course_number	Prerequisite_number
CS3380	CS3320
CS3380	MATH2410
CS3320	CS1310

وتكون حصيلة النموذج الأولي للبيانات، بعد تطبيق مجموعة معينة من الإجراءات، قاعدة بيانات مادية تتكون من جداول مترابطة فيما بينها وفق النموذج العلائقي، ولذا يحتوي الشكل على أربعة جداول هي :

- مقرر
- شعبة
- درجات
- متطلب سابق

مثال حول قاعدة البيانات

❑**كتالوج نظام إدارة البيانات (DBMS Catalog):**

•هو جزء من ال DBMS الذي يحتفظ بمعلومات عن الجدول، وليس بيانات الجدول نفسه ، أي عن بنية الجدول، مثل اسم الجدول، أنواع البيانات فيه، أي أنه يخزن تعريفات البيانات الموجودة في قاعدة البيانات والتي أسميناها meta-data.

•ويطلق على هذا الكتالوج أكثر من تسمية، منها بنية قاعدة البيانات ومنها تعريفات قاعدة البيانات وكذلك مخطط السكيما لقاعدة البيانات كما سيأتي لاحقا، وبغض النظر عن التسمية، فإن على مصمم قاعدة البيانات الاهتمام بعملية التعريف، التي ينتج عنها هذا الكتالوج، فهي حجر الأساس في تطوير قواعد البيانات الحديثة، مقارنة بالنظم القديمة التي لم تكن تستخدم هذه التقنية.

مثال حول قاعدة البيانات

❑كتالوج نظام إدارة البيانات (DBMS Catalog):

RELATIONS

Relation_name	No_of_columns
STUDENT	4
COURSE	4
SECTION	5
GRADE_REPORT	3
PREREQUISITE	2

COLUMNS

Column_name	Data_type	Belongs_to_relation
Name	Character (30)	STUDENT
Student_number	Character (4)	STUDENT
Class	Integer (1)	STUDENT
Major	Major_type	STUDENT
Course_name	Character (10)	COURSE
Course_number	XXXXNNNN	COURSE
….	….	…..
….	….	…..
….	….	…..
Prerequisite_number	XXXXNNNN	PREREQUISITE

مستخدمي قواعد البيانات

- هم الأشخاص الذين يستخدمون أو يديرون قاعدة البيانات ومحتوياتها ، والذين يصممون ويطورون ويحافظون على تطبيقات قواعد البيانات ، وكذلك بالنسبة لل DBMS
- ويصنف المستخدمون إلى قسمين كل قسم يتكون من مجموعة من التصنيفات هي:

1. **المستخدم غير المباشر ويسمون مجازا workers behind the scene (أي عاملون خلف المنصة)**
2. **المستخدم المباشر ويسمون مجازا actors on the scenes (ممثلون على المنصة)**

مستخدمي قواعد البيانات

1. **المستخدم غير المباشر :**

أ- **مديرو قاعدة البيانات (database administrators):** مسئوليتهم في منح حقوق وشرعية الوصول إلى قاعدة البيانات ، وفي تنسيق ومراقبة الاستخدام ، والحصول على موارد البرمجيات والعتاد ، والتحكم في الاستخدام ومراقبة كفاءة العمليات

ب- **مصممو قاعدة البيانات(database designers):** مسئوليتهم في تعريف المحتوى وكتالوج قاعدة البيانات وكذلك شروطها ووظائفها ومعاملاتها، ويجب أن يتواصل مصممو قاعدة البيانات مع المستخدم النهائي end-user ويجب عليهم فهم احتياجاته، وعكسها في قاعدة البيانات المصممة.

مستخدمي قواعد البيانات

2. **المستخدم المباشر** :وهو المستخدم النهائيend –users

أ- **المستخدم النادرcasual user** : يصل المستخدم إلى قاعدة البيانات نادرا عند الحاجة

ب- **المستخدم العادي naive user** : وهذا هو النوع الذي يقوم بالقسم الأكبر من أعمال المستخدم النهائي حيث يقوم بالإجراءات الثابتة المعرفة مسبقا ، ويقوم بتنفيذها على قاعدة البيانات.

ت- **المستخدم المركب sophisticated user**:وهو المستخدم ذو المهام المعقدة كمحللي النظام والعلماء المتخصصين

ث- **المستخدم الوحيد stand-alone user:** غالبا يقوم هذا المستخدم بصيانة قاعدة البيانات الشخصية بواسطة برمجيات جاهزه

خصائص قواعد البيانات

من خصائص قواعد البيانات:

1. الوصف الذاتي لنظام قواعد البيانات
2. إستقلالية البرنامج عن البيانات
3. تجريد البيانات
4. دعم أكثر من طريقة عرض
5. المشاركة في البيانات، والمعالجة المتعددة للبيانات من أكثر من مستخدم

مزايا نظام إدارة قواعد البيانات

تتميز أنظمة قواعد البيانات ب:

1. -التحكم بتكرار البياناتredundancyعند تخزينها ، وضبط الوصول غير الشرعي (غير المصرح) للبيانات ، وتقديم بنية تخزين مفهرسة لتسريع المعالجة والاستعلام .
2. - تقديم خدمات النسخ الاحتياطي والاستعادة recovery، backup
3. تقديم واجهات متعددة لمختلف المستخدمين ، تمثيل علاقات معقدة بين البيانات
4. الدفع نحو تطبيق شروط سلامة البيانات وغير ذلك من المميزات منها (مرونة تغيير بنية البيانات ، فرصة التوجه نحو المعيارية standard).

....

مزايا نظام إدارة قواعد البيانات

تتميز أنظمة قواعد البيانات ب:

5. المشاركة في البيانات لأكثر من مستخدم
6. منع الوصول غير المخول للبيانات
7. تمثيل العلاقات المعقدة للبيانات
8. المرونة في تغيير بنية البيانات
9. توفير البيانات المحدثة بشكل فوري

التطور التاريخي لقواعد البيانات

التطور التاريخي لقواعد البيانات :

1. تطبيقات قواعد البيانات القديمة : النموذج الشبكي ، النموذج الهرمي التي كانت موجودة في الستينات.
2. نظم قواعد البيانات العلائقية : بدأت في السبعينات وقد طبقت في الثمانينات عبر بعض ال DBMS التجارية
3. التطبيقات الحديثة كائنية التوجة Object-Oriented DB
4. البيانات في التجارة الإلكترونية وتطبيقات الويب XMLو HTMLلغة scripts languages

ورغم التطورات المتسارعة في نظم قواعد بيانات، إلا أن النظام ألعلائقي مازال هو النظام المعمول به في حزم ال DBMSالمختلفة، وحتى النظم الأحدث فقد طورت عنه.

استخدام انظمة إدارة قواعد البيانات

- لا ينصح بالاستخدام:
  - إذا كانت تكلفة برمجيات الـDBMS أغلى من جدوى النظام نفسه.
  - عندما لا تكون برمجيات الـDBMSضرورية كأن تكون قاعدة البيانات واضحة وبسيطة وثابتة .
  - عندما تكون قاعدة البيانات أعقد من الـDBMS المتاح، مثل انظمة الوقت الحقيقي
- ليس ضرورياً الاستخدام:
  - عندما لا تكون هناك حاجة للوصول المتعدد للبيانات من أكثر من مستخدم

قواعد معطيات

المعهد التقاني للحاسوب بحماة
قسم الشبكات

محاضرة رقم 2

مفاهيم ومعمارية نظم قواعد البيانات

Eng. Hana Sabbagh

في هذ المحاضرة

- نموذج البيانات
- تصنيفات نماذج البيانات
- مخطط قواعد البيانات
- مثيل (أو تفاصيل) قواعد البيانات
- المعمارية الثلاثية لمخطط قاعدة البيانات(معمارية الإسكيما)
- إستقلالية البيانات
- لغات قواعد البيانات
- واجهة التطبيق في نظم إدارة قواعد البيانات

نموذج البيانات

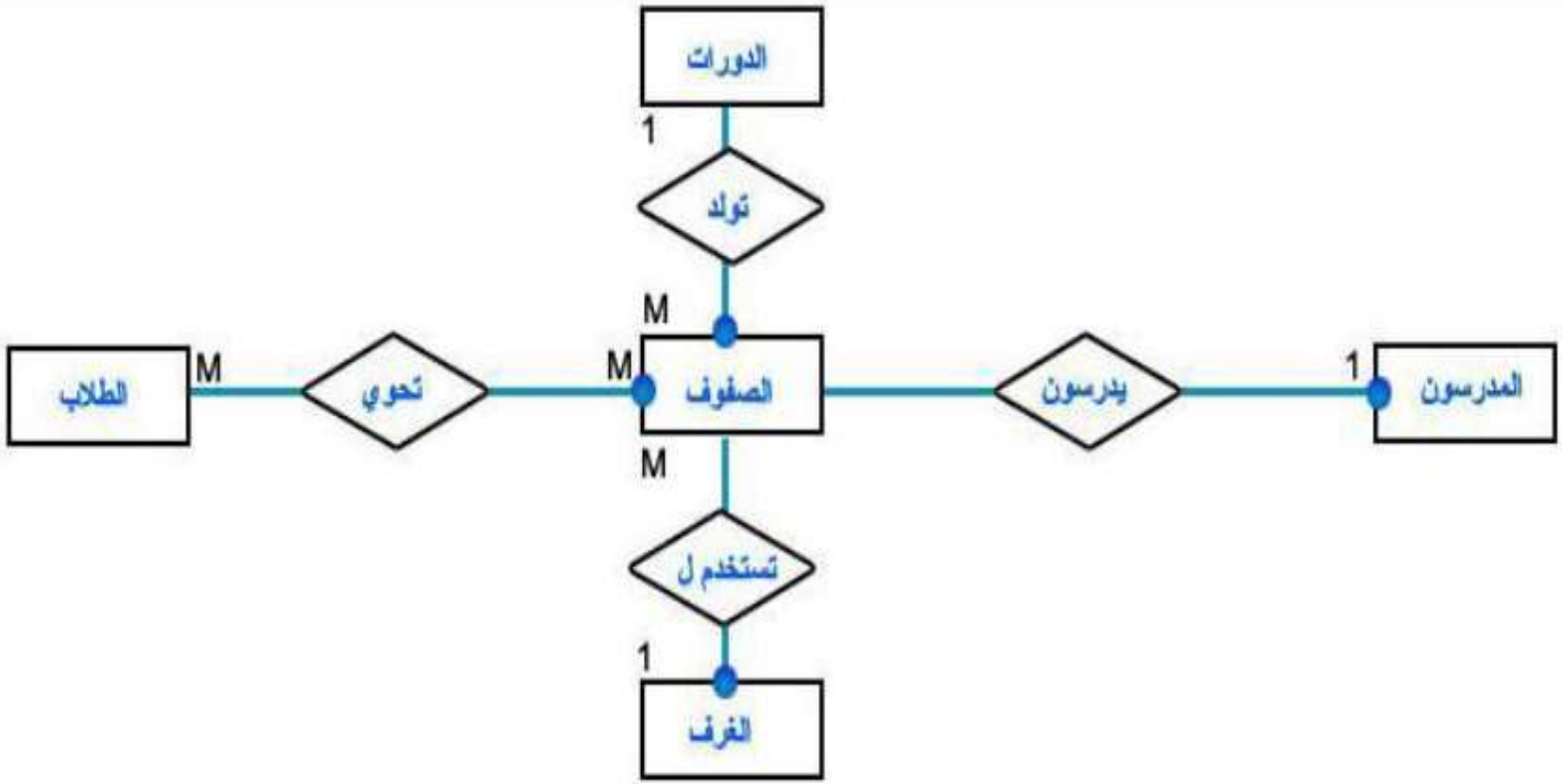
- **نموذج البيانات Data Model:** هو مجموعة من المفاهيم التي تقوم بوصف بنية قاعدة البيانات والشروط أو القيود التي تسير قاعدة البيانات وفقها.
- **عمليات نموذج البيانات Data Model Operations:** هي العمليات التي تحدد آلية استرجاع البيانات وتحديثها من قاعدة البيانات، وذلك بالرجوع إلى مفهوم النموذج التي تسيروفقه، وتشمل هذه العمليات نوعين هما العمليات الاساسية basic operations التي تعرف من ضمن النموذج، والعمليات التي يعرفها المستخدم.

تصنيفات نماذج البيانات

تقسم نماذج البيانات إلى ثلاثة تصنيفات رئيسية هي:

1. **نماذج البيانات الأولية (أو المفاهيمية) Conceptual data models :** تقدم هذه النماذج مفاهيما قريبة من طريقة المستخدم في استقبال البيانات، ولذلك يسمى نموذج البيانات من هذا النوع بنموذج المستوى الأعلى، أي المستوى الذي يفهمه المستخدم، في هذا النموذج يتم تقديم توصيف لقاعدة البيانات بحسب رؤية المستخدم وشروطه وعملياته.

تصنيفات نماذج البيانات



تصنيفات نماذج البيانات

تقسم نماذج البيانات إلى ثلاثة تصنيفات رئيسية هي:

2. **نماذج البيانات المادية (أو الفيزيائية) Physical data models :** تقدم هذه النماذج مفاهيما تصف تفاصيل تخزين البيانات في الحاسوب بصورتها المادية، ولذلك فهي تسمى أيضا بنماذج البيانات منخفضة المستوى، أي المستوى الذي يفهمه الحاسوب، ويحتوي على توصيفات قاعدة البيانات القريبة من لغة الآلة (لا يحتاج القائمون على إدارة قواعد البيانات إلى التدخل بهذا المستوى, تترك المهمة لنظام الادارة).

تصنيفات نماذج البيانات

تقسم نماذج البيانات إلى ثلاثة تصنيفات رئيسية هي:

3. **نماذج البيانات التنفيذية (أو التمثيلية)**

Implementation (representational) data models:

تقدم هذه النماذج المفاهيم المتوسطة، والتي تقع بين النموذجين أعلاه، وذلك بعمل توازن بين رؤية المستخدم التي في النموذج الأولي وبين تفاصيل تخزين البيانات في الحاسوب (النموذج المادي)، وهذه هي مهمة مصمم قاعدة البيانات في تحويل النموذج الأولي إلى نموذج مادي (استخدام لغة قياسية SQL).

مخطط قواعد البيانات

- **مخطط قاعدة البيانات Database Schema :** هو مخطط يصف قاعدة البيانات، حيث يشمل وصف لبنية البيانات فيها، وشروط تلك البيانات أو القيود التي يجب أن تطبق عليها، ويمكن اعتباره كتالوجا أو قاموسا يصف تلك البيانات، بغض النظر عن البيانات نفسها.
- **المخطط الرسومي Schema Diagram :** هو مخطط قاعدة البيانات ولكن بشكل رسومي لعرض بعض مميزات وخصائص المخطط.
- **بنية المخطط Schema Construct :** هي المكونات التي يتكون منها مخطط قاعدة البيانات، أو هي الكائنات التي تكون المخطط مثلا في قاعدة بيانات جامعة لدينا بعض مكونات مخطط قاعدة البيانات مثل الطالب ، المقرر ..وغيره

مثيل (أو تفاصيل) قواعد البيانات

- هي البيانات الفعلية المخزنة في قاعدة البيانات في لحظة معينة من الزمن، وتسمى كذلك حالة أو امتداد قاعدة البيانات (DB State)، ويسمى مثيل لأنه يحتوي على تفاصيل لنفس الكيان المحدد في قاعدة البيانات، فمثيل الكيان طالب هي بيانات طالب معين..وهكذا، وتترجم أيضا إلى وحدة الكيانinstance entity

مثال مخطط قواعد بيانات

STUDENT

Name	Student_number	Class	Major

COURSE

Course_name	Course_number	Credit_hours	Department

PREREQUISITE

Course_number	Prerequisite_number

SECTION

Section_identifier	Course_number	Semester	Year	Instructor

GRADE_REPORT

Student_number	Section_identifier	Grade

مثال عل مثيل قواعد بيانات

COURSE

Course_name	Course_number	Credit_hours	Department
Intro to Computer Science	CS1310	4	CS
Data Structures	CS3320	4	CS
Discrete Mathematics	MATH2410	3	MATH
Database	CS3380	3	CS

SECTION

Section_identifier	Course_number	Semester	Year	Instructor
85	MATH2410	Fall	04	King
92	CS1310	Fall	04	Anderson
102	CS3320	Spring	05	Knuth
112	MATH2410	Fall	05	Chang
119	CS1310	Fall	05	Anderson
135	CS3380	Fall	05	Stone

GRADE_REPORT

Student_number	Section_identifier	Grade
17	112	B
17	119	C
8	85	A
8	92	A
8	102	B
8	135	A

PREREQUISITE

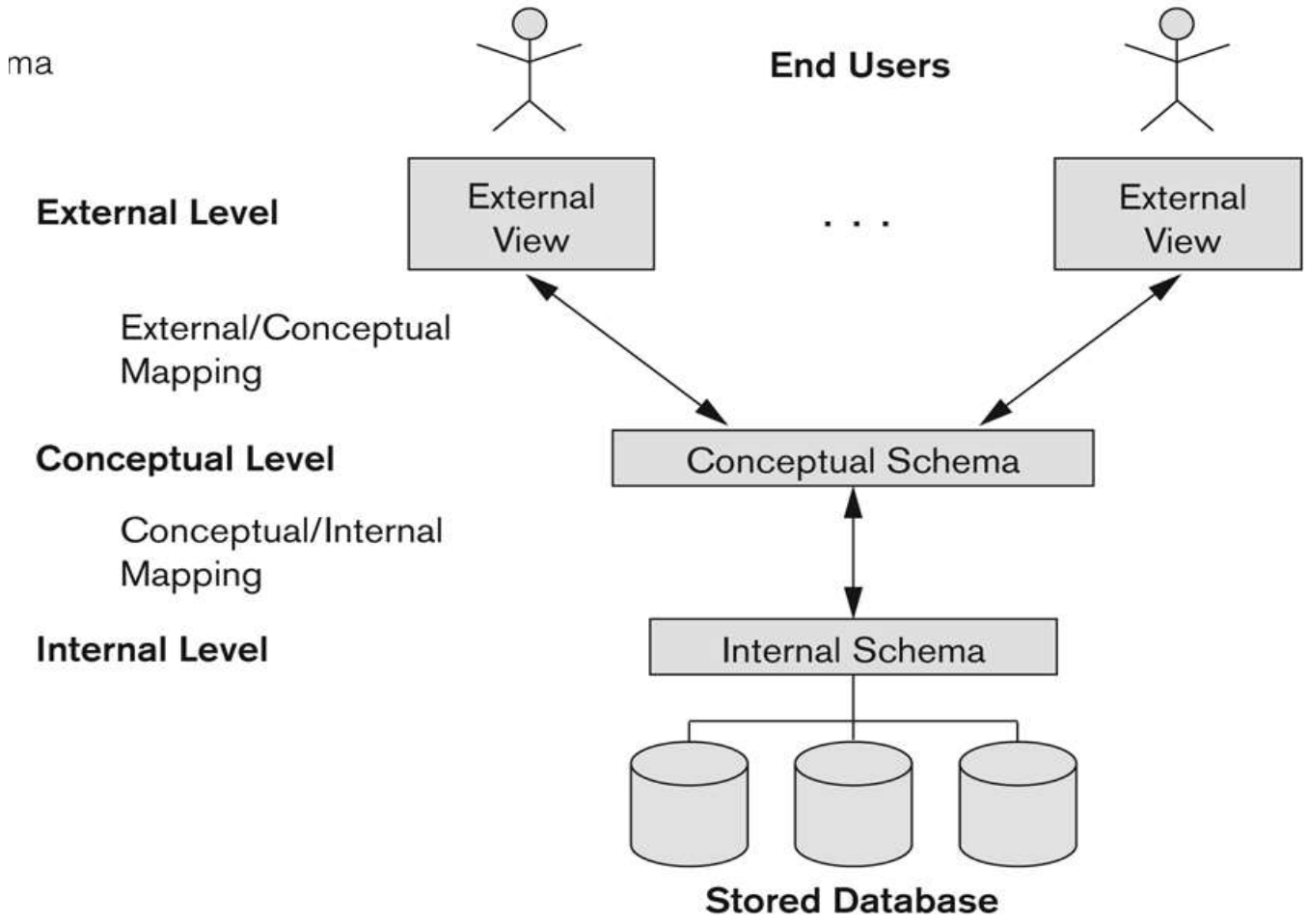
Course_number	Prerequisite_number
CS3380	CS3320
CS3380	MATH2410
CS3320	CS1310

المعمارية الثلاثية لمخطط قاعدة البيانات (معمارية الإسكيما)

يتكون مخطط قاعدة البيانات من ثلاثة مستويات، كل مستوى يستخدم واحدا او إثنين من نماذج البيانات المذكورة سابقا، وهذه المستويات موضحة بالشكل التالي، وهي:

1. **المخطط الداخلي Internal schema**
2. **المخطط الأولي (أو المفاهيمي) Conceptual schema**
3. **المخطط الخارجي External schemas**

المعمارية الثلاثية لمخطط قاعدة البيانات (معمارية الإسكيما)



إستقلالية البيانات

- تتحقق استقلالية البيانات، بالقيام بفصل البيانات التي يراها المستخدم عن بنيتها الداخلية، ويتم ذلك على مرحلتين هما:
  - **الاستقلالية المنطقية للبيانات Logical Data Independence:** هي القدرة على تغيير المخطط الأولي لقاعدة البيانات conceptual schema، بدون تغيير المخططات الخارجية لها ولا تغيير التطبيقات البرمجية عليها.
  - **الاستقلالية الفيزيائية للبيانات Physical Data Independence:** هي القدرةعلى تغيير المخطط الداخلي لقاعدة البيانات internal schemaدون إجراء أي تغيير على المخطط الأولي (المفاهيمي).

إستقلالية البيانات

- يتم دعم إستقلالية البيانات بشقيها المادي والمنطقي بواسطة برمجيات الـ(DBMS) الذي يمكن أن يكون مركزياً أو موزعاً:
- **مدير قاعدة البيانات المركزي Centralized DBMS:** يقوم الـ (DBMS) في هذه الحالة بدمج كل مكوناته في نظام واحد، العتاد والبرمجيات والتطبيقات، وكذلك برمجيات واجهات المستخدم.
- **مدير قاعدة البيانات الموزع Distributed DBMS:** ويسمى في هذه الحالة بنظام قواعد بيانات المزود و العميل Client Server DBMS، لأنه لا يوزع جميع مكونات نظام قواعد البيانات ولكنه يوزع فقط مجموعة من مخدمات (مزودات) قواعد البيانات التي تدعم مجموعة من العملاء أو الطرفيات Terminals

لغات قواعد البيانات

- **لغة تعريف البيانات Data Definition Language (DDL)**: هي جزء من لغة الـSQL التي تهتم بإنشاء وتعديل وحذف جميع كائنات قاعدة البيانات، بما فيها إنشاء وتعديل وحذف قواعد البيانات أو أي من كائناتها كالجدول والمنظورView
- **لغة معالجة البيانات Data Manipulation Language (DML):** تقوم هذه اللغة بمعالجة البيانات داخل الكائن نفسه، دون أن تتدخل ببنية الكائن.

واجهة التطبيق في نظم إدارة قواعد البيانات

- تختلف واجهة التطبيق في نظم الـDBMS،ويمكن تصنيفها إلى:

1. واجهة لغة الإستعلام SQLالمستقلة،حيث يقوم المستخدم بكتابة جمل الإستعلام SQL.
2. واجهة برمجة خاصة بالمبرمج' يقوم من خلالها بتنفيذ برامج خاصة تتضمن لغة معالجة البيانات DML
3. واجهات مستخدم سهلة الإستخدام، تتضمن قوائم منسدلة ضمن نماذج سهلة الإستخدام، وتتميز بكونها واجهات رسومية سهلة الإستخدام

قواعد معطيات

المعهد التقاني للحاسوب بحماة
قسم الشبكات

محاضرة رقم 3

نمذجة البيانات باستخدام مخطط الكينونة العلائقي

Eng. Hana Sabbagh

في هذ المحاضرة

- ما هو مخطط الكينونة العلائقي؟
- مكونات مخطط الكينونة العلائقي
  - الكيان
  - الصفة وانواعها
  - المفتاح الرئيسي والمفتاح الجزئي
  - الكيان الضعيف
  - العلاقات و أنواع العلاقات
  - أنواع القيود على العلاقات
- مثال1(مخطط كينونة علائقي لجامعة)
- مثال2 (مخطط كينونة علائقي لشركة)

ما هو مخطط الكينونة العلائقي؟

هو نموذج عالي المستوي يقوم بعرض بناء البيانات، ويتم استخدام هذا النموذج أثناء مرحلة التصميم المفاهيمي للنموذج الأولى،وينتج عن ذلك النموذج ألأولي ، لقاعدة البيانات، والذي عن طريقة نقوم بتصميم مخطط قاعدة البيانات، ويتم تمثيل بناءالبيانات والقيود المطلوبة عليها باستخدام إشكال رسومية سهلة ومحددة.

مكونات مخطط الكينونة العلائقي

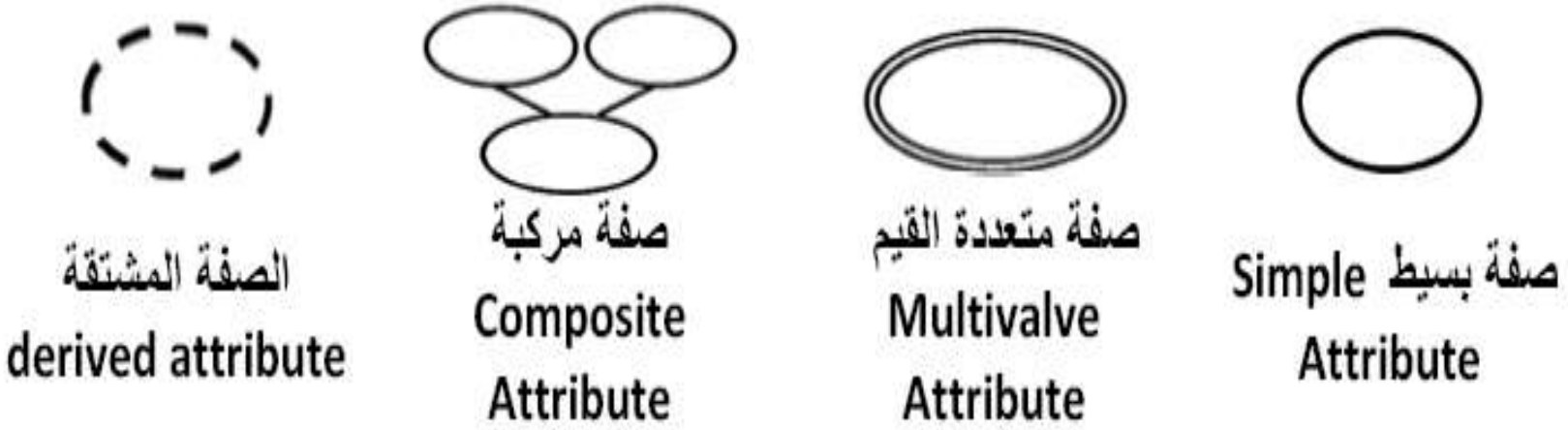
- **الكيان أو الكينونة(Entity):** هو الوحدة الأساسية التي يتم تمثيلها بنموذج الكينونة/العلاقة ويشير هذا الكيان إلي "شيء" حقيقي في الحياة سواء كان له وجود فعلي مثل (طالب – موظف – سيارة - ... الخ) أو وجود منطقي مثل (شركة – وظيفة – مقرر -... الخ).
- ويتم تمثيل الكيان باستخدام شكل مستطيل يكتب داخله اسم الكيان أو الكينونة

مكونات مخطط الكينونة العلائقي

- **الصفة (Attribute):** هي صفة معينة تصف الكيان وتكون تابعة له في المخطط ، مثل اسم الموظف ، عمر الطالب، مرتب موظف، درجة طالب، عدد الساعات الدراسية لمقرر، لاحظ أن كل صفة تتبع كينونة أو 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

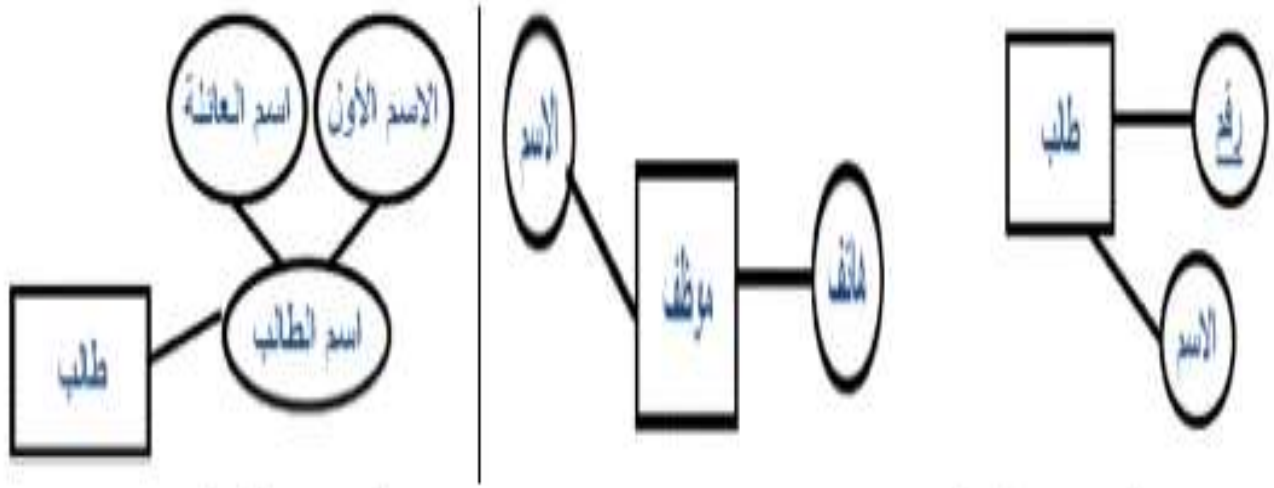
مكونات مخطط الكينونة العلائقي

- يتم تمثيل الصفة باستخدام شكل بيضاوي، والصفة متعددة القيم يتم تمثيلها بالشكل البيضاوي المزدوج



شكل تمثيل الصفات المختلفة للكيان

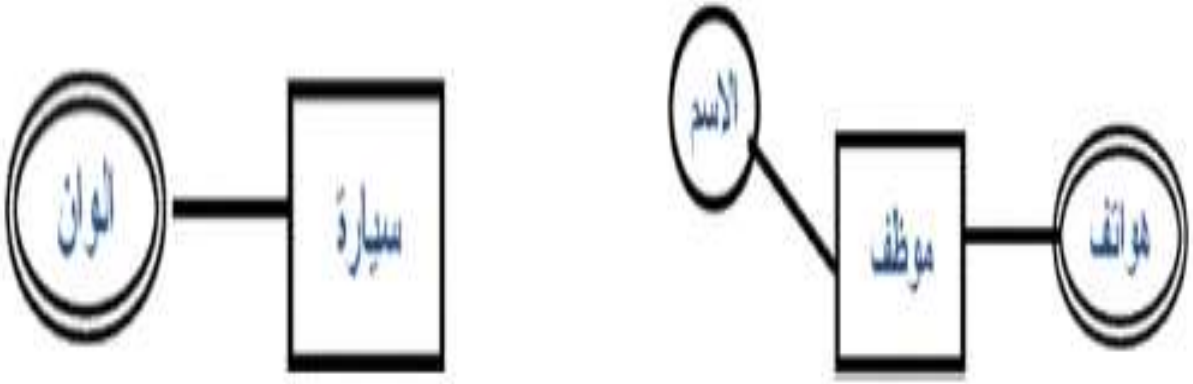
مكونات مخطط الكينونة العلائقي



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

شكل مثال توضيحي

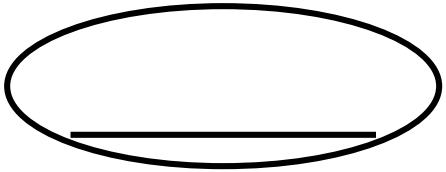
مكونات مخطط الكينونة العلائقي



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

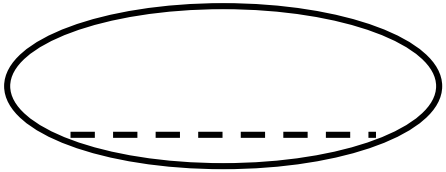
مكونات مخطط الكينونة العلائقي

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



مكونات مخطط الكينونة العلائقي

- **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

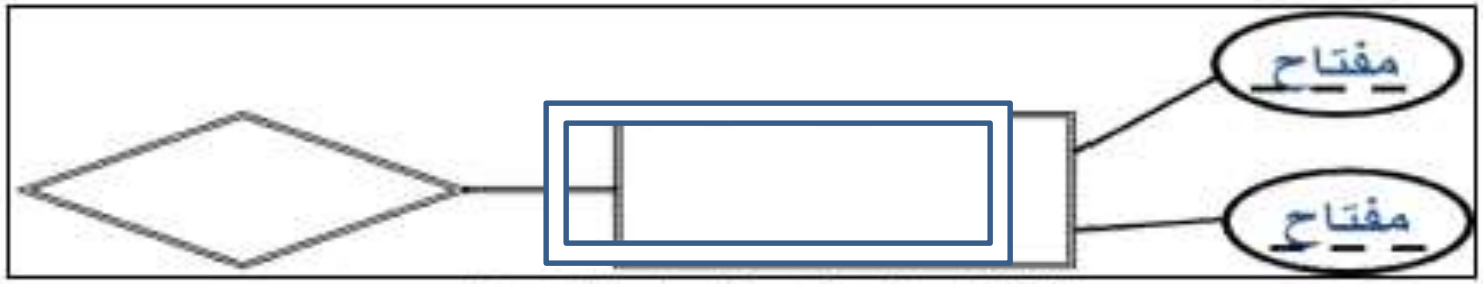


مكونات مخطط الكينونة العلائقي

- **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 تقوي، ويتم تمثيل الكيان الضعيف بمستطيل مزدوج

مكونات مخطط الكينونة العلائقي

- يرتبط المفتاح الجزئي بالكيان الضعيف، ليساعد فيما بعد بتكوين مفتاح رئيسي للكيان الضعيف كما سنرى في محاضرة لاحقة.



المفتاح الجزئي للكيان الضعيف

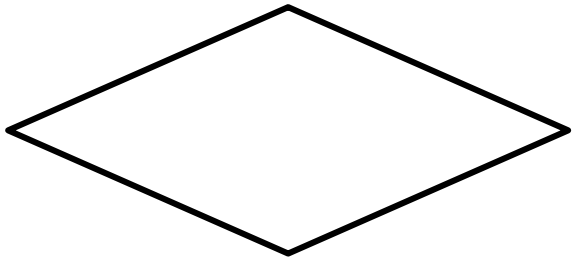
مكونات مخطط الكينونة العلائقي

- يرتبط المفتاح الجزئي بالكيان الضعيف، ليساعد فيما بعد بتكوين مفتاح رئيسي للكيان الضعيف كما سنرى في محاضرة لاحقة.

مثال توضيحي

مكونات مخطط الكينونة العلائقي

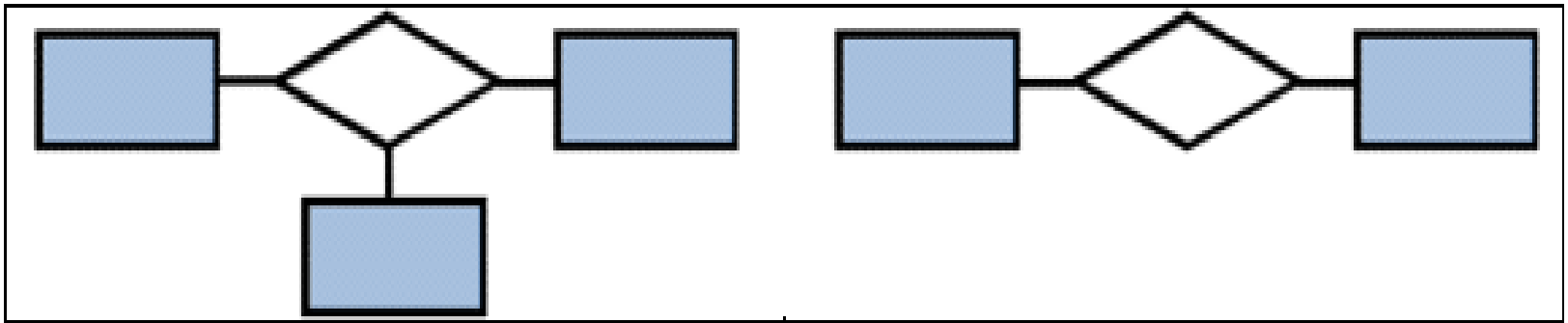
- **العلاقات (Relation):** العلاقة (R) بين مجموعة من الكيانات (E1,E2, ...En) هي مجموعة تمثل الارتباطات بين هذه الكيانات، كل وحدة في العلاقة (R) هي عبارة عن اتحاد أو ارتباط بين الكيانات المرتبطة بهذه العلاقة، بحيث أن هذه الوحدة تمثل بصف واحد من كل كيان مشارك في العلاقة.
- في نموذج الكينونة/العلاقة، يجب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



.

مكونات مخطط الكينونة العلائقي

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



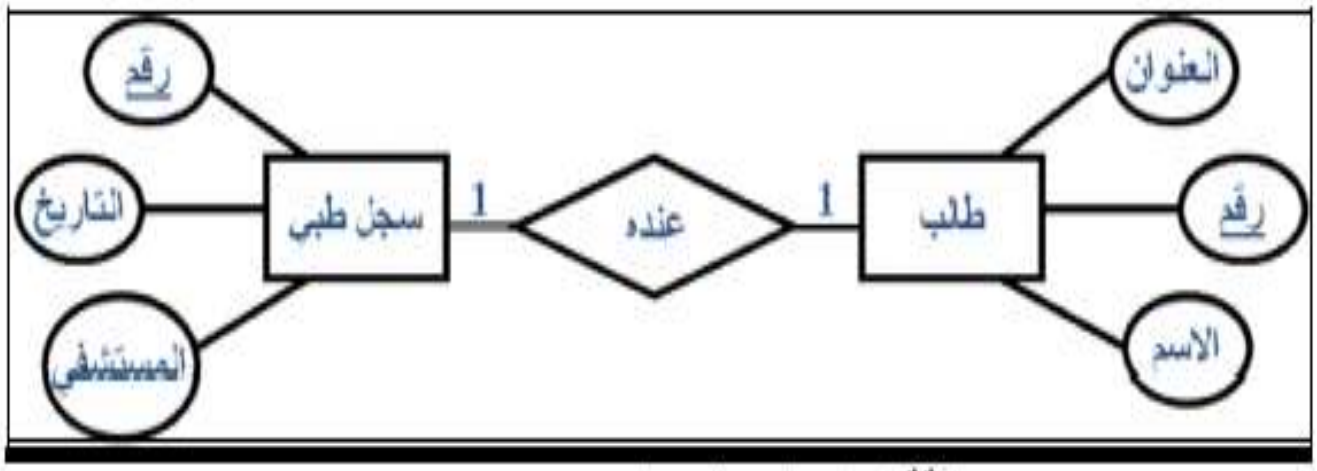
العلاقة الثنائية Binary والعلاقة الثلاثية Ternary

مكونات مخطط الكينونة العلائقي

- **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:

1. **علاقة واحد- الى-واحد(one-to-one):** وفيها ترتبط وحدة واحدة من الكيان الأول بوحدة واحدة من الكيان الآخر على الأكثر، ويرمز لها بالرمز1:1
2. **علاقة واحد- الى-كثير(one-to-many) :** وفيها يمكن أن ترتبط وحدة واحدة من أحد الكيانات بأكثر من وحدة في الكيان الآخر، والعكس غير صحيح، ويرمز لها بالرمز 1:N
3. **علاقة كثير- الى-كثير(many-to-many) :** وفيها يمكن أن ترتبط أكثر من وحدة من الكيان الأول بأكثر من وحدة في الكيان الآخر، والعكس، أي يمكن لأي وحدة في الكيان الآخر أن ترتبط بأي وحدة في الكيان الأول، ويرمز لها بالرمز M:N

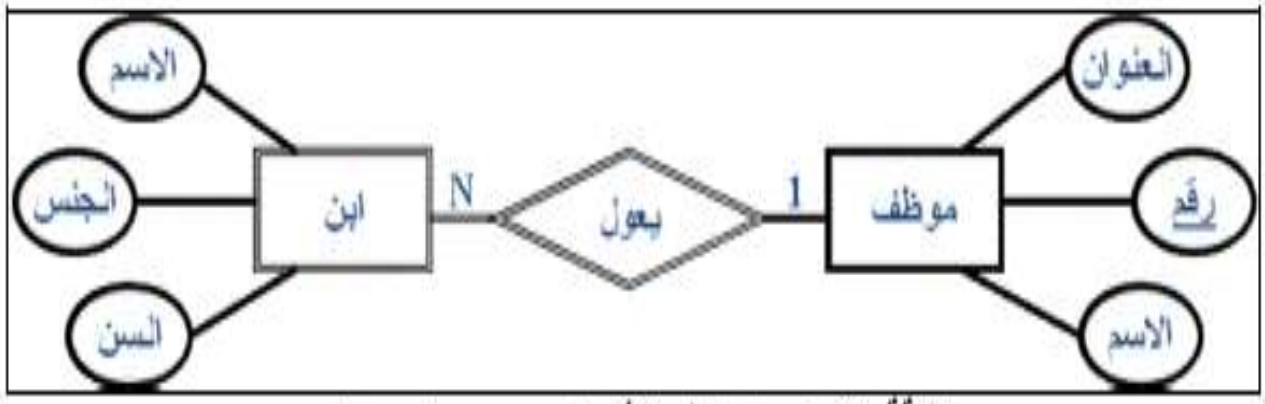
مكونات مخطط الكينونة العلائقي



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).

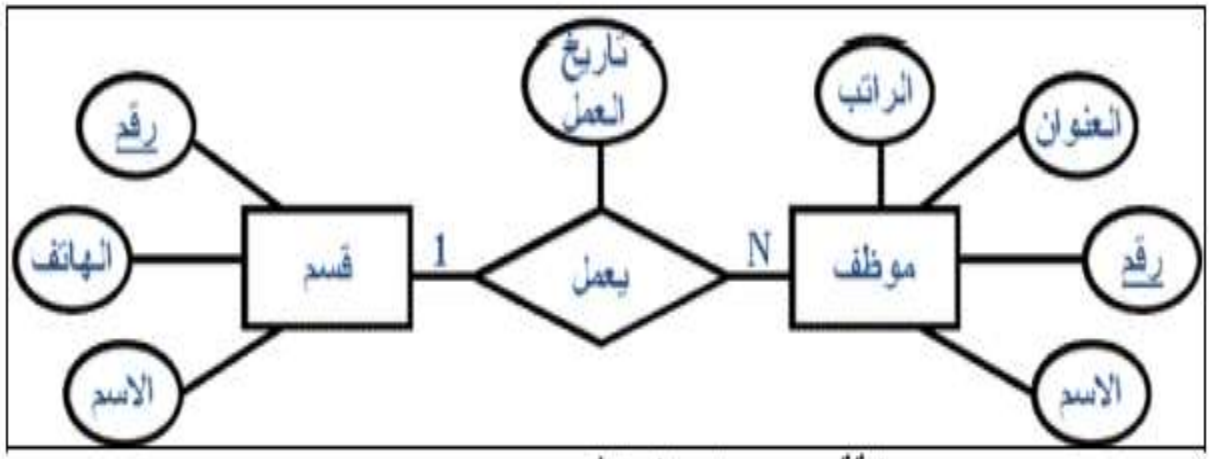
مكونات مخطط الكينونة العلائقي



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.

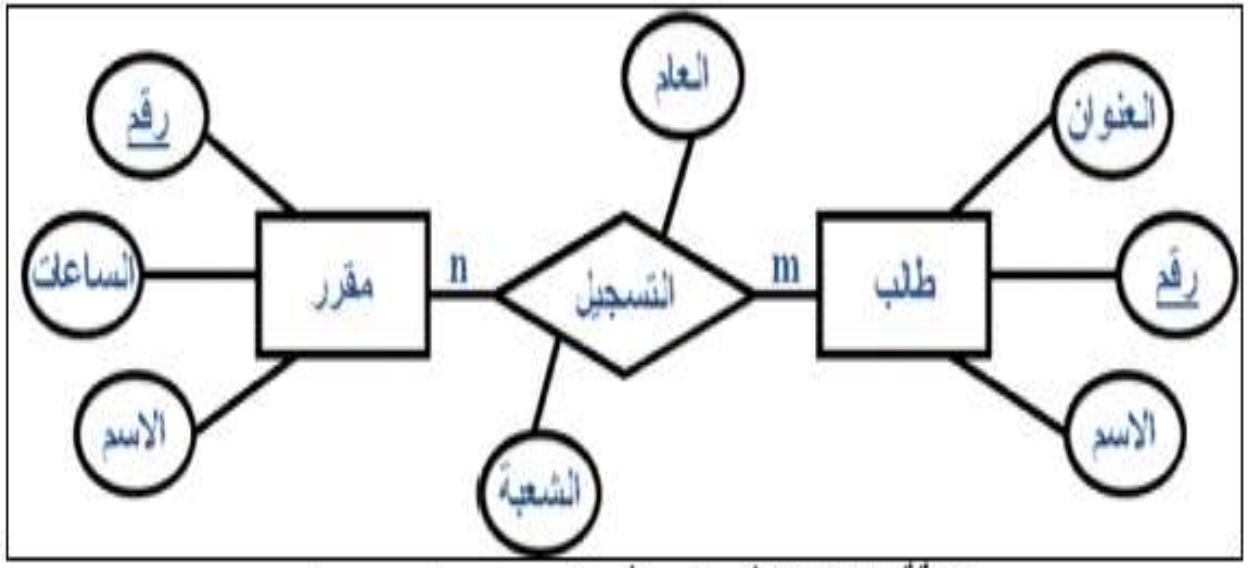
مكونات مخطط الكينونة العلائقي



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.

مكونات مخطط الكينونة العلائقي



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

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

مكونات مخطط الكينونة العلائقي

- **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.

مكونات مخطط الكينونة العلائقي

- **أنواع القيود على العلاقات Relationship Constraints:**

1. **الاشتراك الكلي(Total participation) :** نقول أن العلاقة علاقة الاشتراك الكلي ، إذا كان كل وحدة في الكيان ألأول 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 طبي) لطالب في نظام معلومات الجامعة، إذ لا بد أن يرتبط أي قيد في السجل الطبي بطالب ما في كيان الطلاب، وإلا فلا معنى للعلاقة هذه.

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

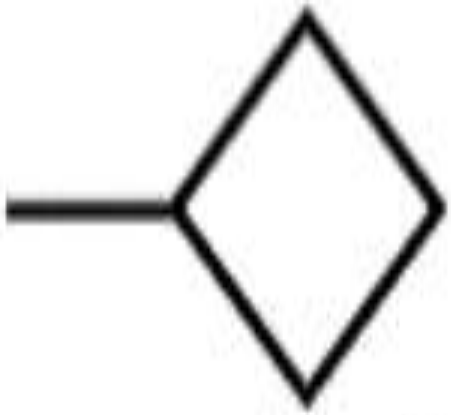
مكونات مخطط الكينونة العلائقي

- **أنواع القيود على العلاقاتRelationship Constraints :**

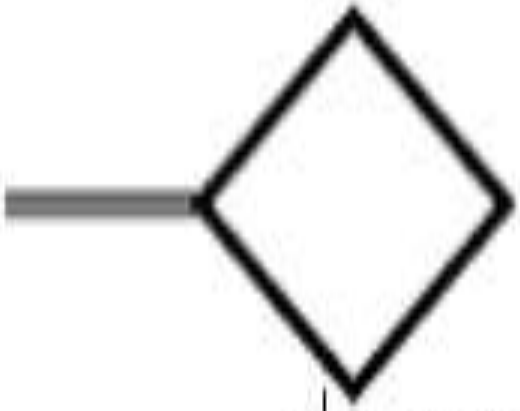
2.الاشتراك الجزئي(Partial participation) :نقول أن العلاقة علاقة اشتراك جزئي، إذا كانت بعض الوحدات في الكيان المشترك بالعلاقة ترتبط ببعض الوحدات في الكيان الآخر ضمن العلاقة، ويتم تمثيل قيد الاشتراك الجزئي برسم خط مفرد يربط الكيانات المرتبطة.

مكونات مخطط الكينونة العلائقي

- **أنواع القيود على العلاقاتRelationship Constraints:**



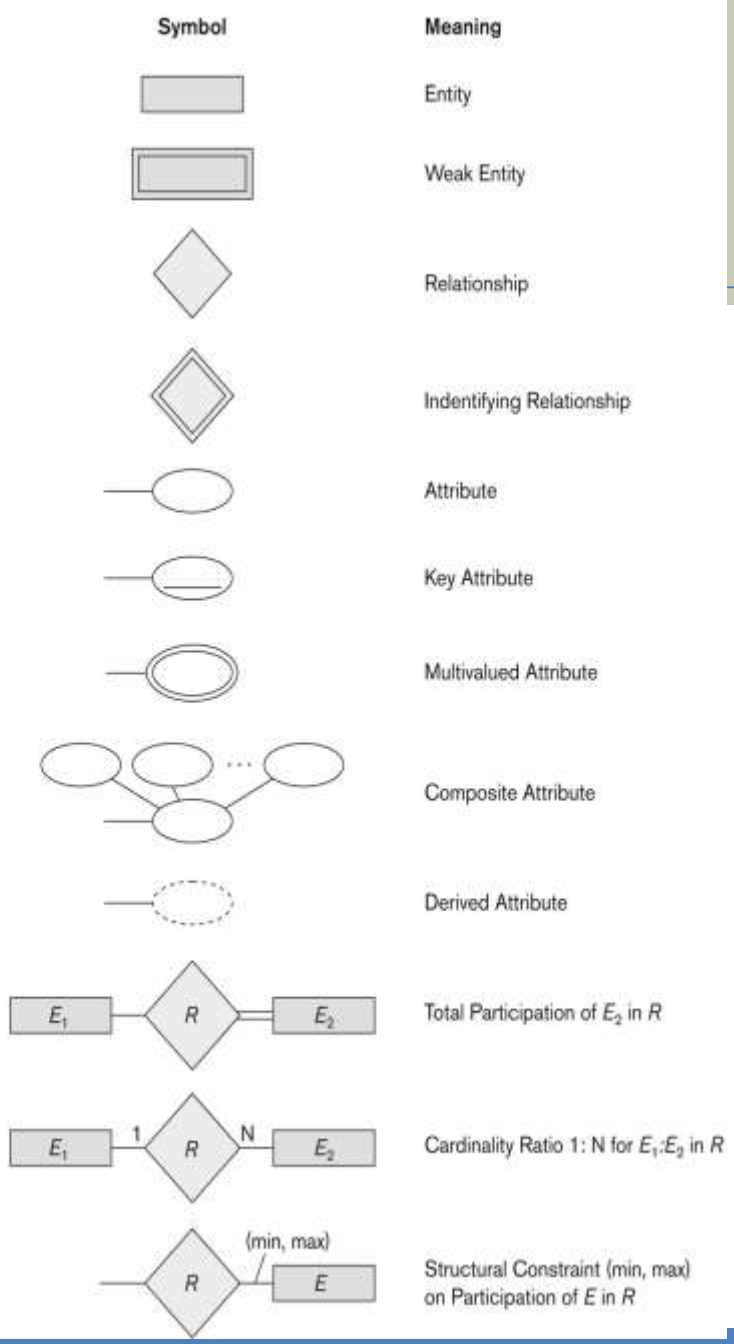
شكل علاقة اشتراك جزئي



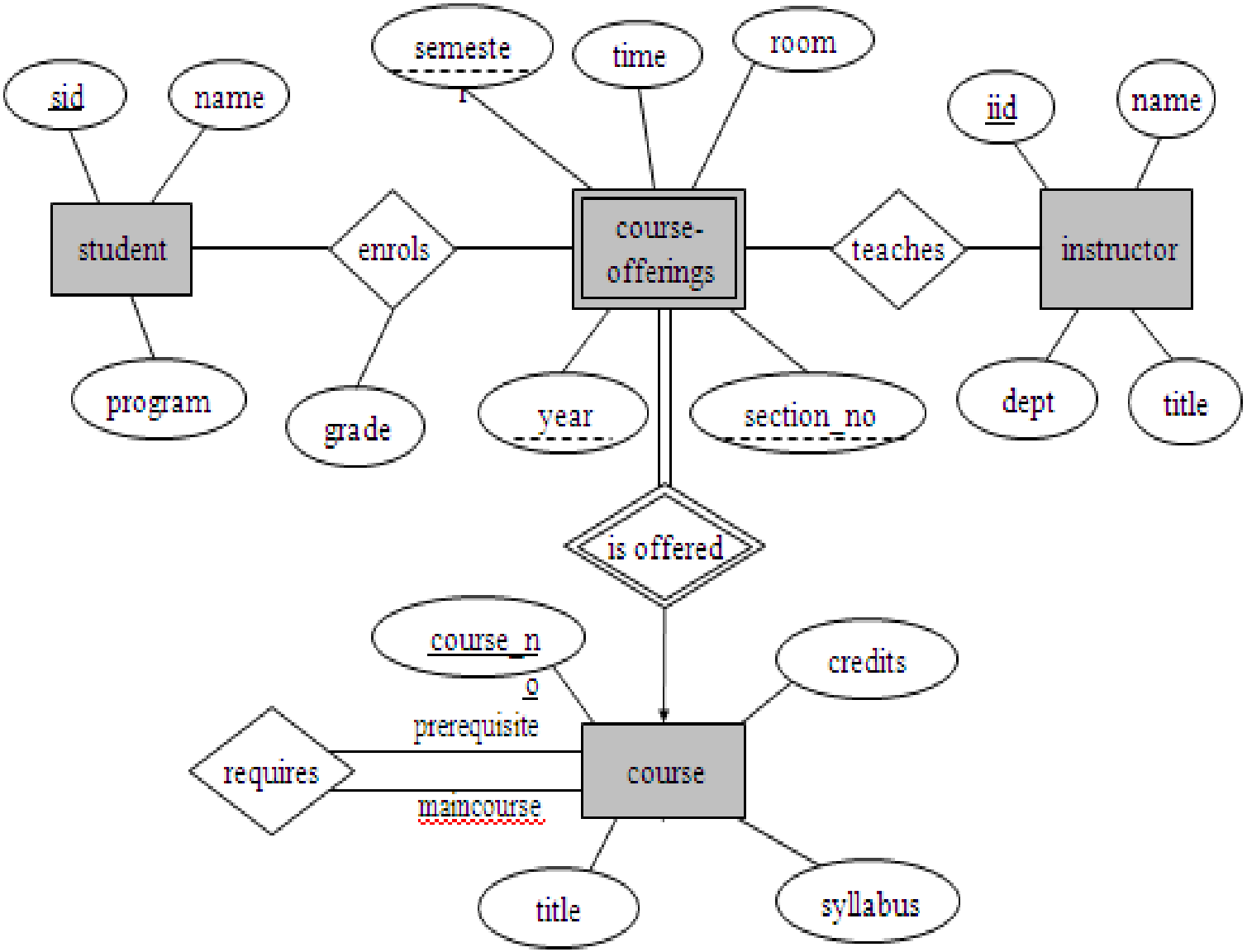
شكل علاقة اشتراك كلي

مكونات مخطط الكينونة العلائقي

Figure 3.14
Summary of the notation for ER diagrams.

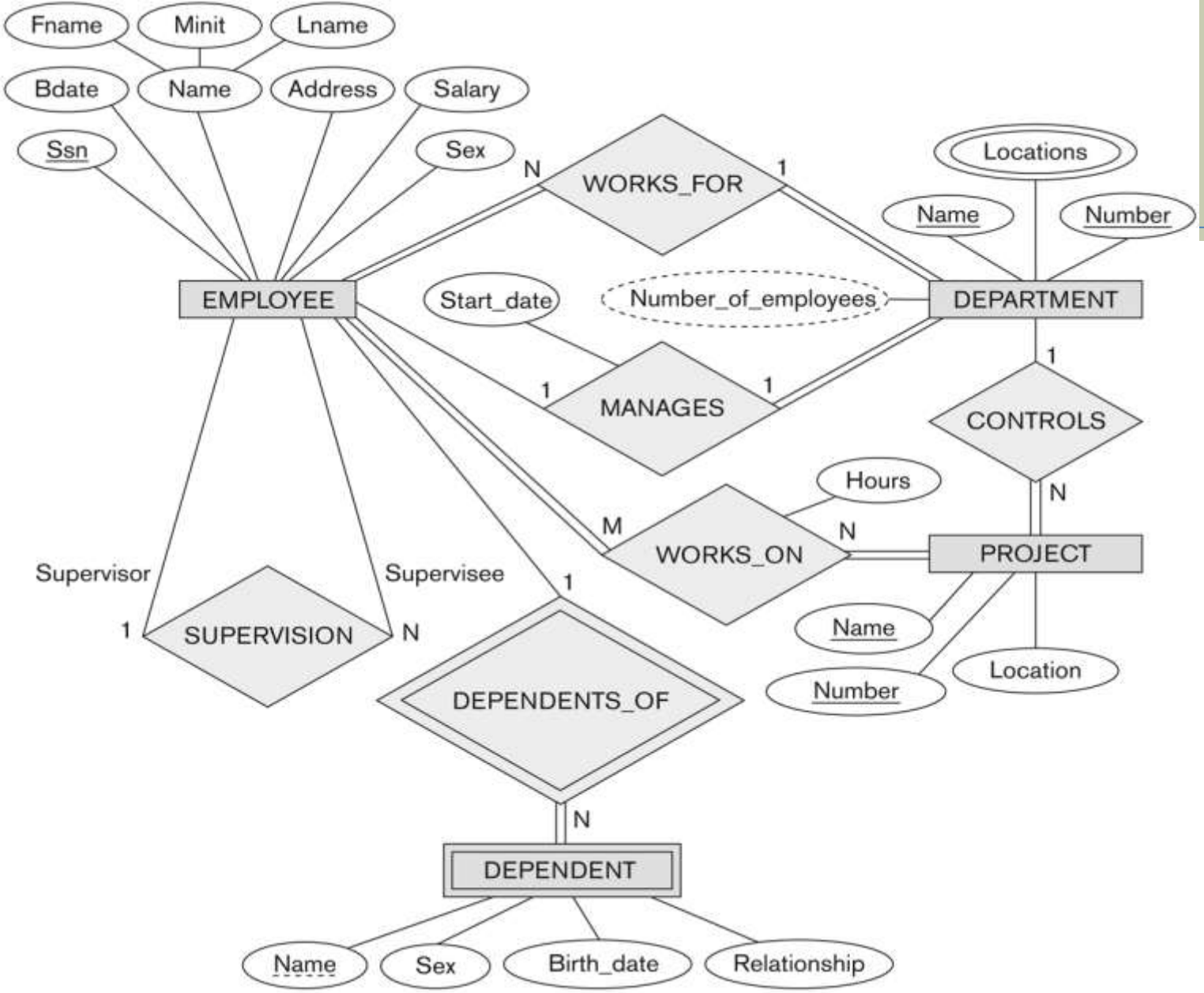


مثال 1



E-R diagram for a university.

مثال2


Fname
Minit
Lname
Name
Bdate
Address
Salary
Ssn
Sex
N
WORKS_FOR
1
Locations
Name
Number
EMPLOYEE
Start_date
Number_of_employees
DEPARTMENT
1
1
MANAGES
1
CONTROLS
Hours
N
M
WORKS_ON
N
PROJECT
Supervisor
Supervisee
1
1
SUPERVISION
N
Name
Location
Number
DEPENDENTS_OF
N
DEPENDENT
Name
Sex
Birth_date
Relationship

قواعد معطيات

المعهد التقاني للحاسوب بحماة
قسم الشبكات

محاضرة رقم 4

قواعد البيانات العلائقية

التحويل من مخطط الكيان العلائقي إلى مخطط قواعد بيانات

Eng. Hana Sabbagh

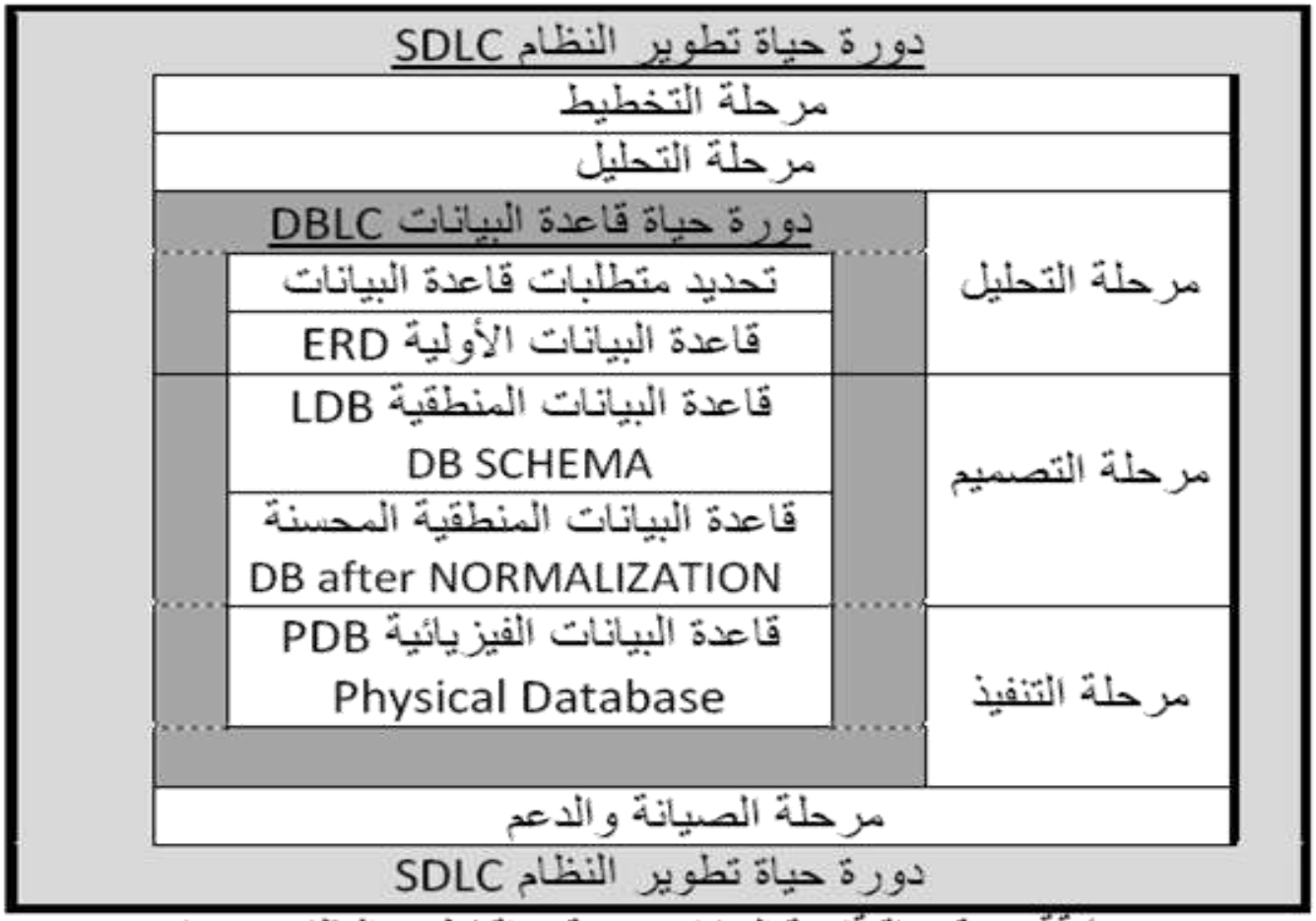
في هذ المحاضرة

- دورة حياة قاعدة البيانات Database Life Cycle (DBLC)
- مخطط قواعد البيانات Database Schema
- التحويل من مخطط الكيان العلائقي إلى مخطط قواعد بيانات
  - تحويل الكيان
  - تحويل الصفة المركبة والصفة متعددة القيمة
  - تحويل الكيان الضعيف
  - تحويل العلاقات
- مثال1(تحويل مخطط كينونة علائقي لجامعة إلى مخطط قواعد بيانات)
- مثال2 (تحويل مخطط كينونة علائقي لشركة إلى مخطط قواعد بيانات)

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:

Database Life Cycle (DBLC) دورة حياة قاعدة البيانات



علاقة دورة حياة قاعدة البيانات بدورة حياة تطوير النظام عموما

دورة حياة قاعدة البيانات Database Life Cycle (DBLC)

تتكون **دورة حياة قاعدة البيانات من المراحل التالية:**

1. **تحديد المواصفات والمتطلبات الخاصة بقاعدة البيانات،** وهي مرحلة جزئية ضمن جمع مواصفات ومتطلبات نظام المعلومات في مرحلة التحليل.
2. **إعداد قاعدة البيانات الأولية،** وفيها يتم تصميم نموذج اولي للبيانات بواسطة مخططات علاقة-كينونة (E-RD).
3. **تصميم قاعدة البيانات المنطقية،** تحويل قاعدة البيانات الأولية، أو مخطط علاقة-كينونة إلى مخطط الاسكيما، وذلك بإتباع قواعد التحويل

دورة حياة قاعدة البيانات Database Life Cycle (DBLC)

تتكون دورة حياة قاعدة البيانات من المراحل التالية:

4. **تحسين قاعدة البيانات المنطقية**، وذلك بتطبيق قواعد تطبيع البيانات Normalization التي تهدف إلى تقليل تكرارية البيانات، من أجل رفع كفاءة قاعدة البيانات ما أمكن.
5. **تنفيذ قاعدة البيانات الفيزيائية physical database**: وفي هذه المرحلة يتم كتابة أكواد إنشاء قاعدة البيانات بلغةSQL ، ويحدد فيها بنية الجداول ونوع بيانات الحقول والمفاتيح الاساسية والاجنبية وباقي شروط تصميم قاعدة البيانات، ثم تنفيذ ذلك ضمن مدير قاعدة بيانات DBMS مناسب، مثل (oracle, access, sqlserver, mysql .... etc)

مخطط قواعد البيانات Database Schema

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

مخطط قواعد البيانات Database Schema

- **لوصف مخطط قواعد البيانات ،نستخدم المصطلحات التالية:**
- **العلاقة (جدول السكيما) relation:** أو يمكن أن نطلق عليها اسم 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، ويتكون 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 الصف row الذي يمثل وحدة instance من وحدات الكيان، بعد تحويله إلى جدول، ويتكون الصف من الخلايا الناتجة عن تقاطعه مع الأعمدة المكونة للجدول.

مخطط قواعد البيانات Database Schema

- **لوصف مخطط قواعد البيانات ،نستخدم المصطلحات التالية:**
- **المفتاح الرئيسي Primary Key(PK):** هو حقل في جدول يتميز بأن قيمه وحيدة في جميع صفوف الجدول، وتكون قيمتة مميزة لكل صف عن أي صف آخر.
- **المفتاح الأجنبي foreign key(FK):** هو حقل موجود في جدول وهو لا يمثل واحدة من صفاتة، ولكنه يعتبر مفتاح أجنبيا لأنه يمثل جدولا آخر، ويجب أن يكون هو نفسه المفتاح الرئيسي في ذلك الجدول، أو على الأقل تكون قيمتة وحيده unique value ويقوم المفتاح الأجنبي بتمثيل العلاقة relationship والربط بين جدولين.

AUTHOR

au_id	au_lname	au_fname	address	city	state
172-32-1176	White	Johnson	10932 Bigge Rd.	Menlo Park	CA
213-46-8915	Green	Marjorie	309 63rd St. #411	Oakland	CA
238-95-7766	Carson	Cheryl	589 Darwin Ln.	Berkeley	CA
267-41-2394	O'Leary	Michael	22 Cleveland Av. #14	San Jose	CA
274-80-9391	Straight	Dean	5420 College Av.	Oakland	CA
341-22-1782	Smith	Meander	10 Mississippi Dr.	Lawrence	KS
409-56-7008	Bennet	Abraham	6223 Bateman St.	Berkeley	CA
427-17-2319	Dull	Ann	3410 Blonde St.	Palo Alto	CA
472-27-2349	Gringlesby	Burt	PO Box 792	Covelo	CA
486-29-1786	Locksley	Charlene	18 Broadway Av.	San Francisco	CA

TITLE

title_id	title	type	price	pub_id
BU1032	The Busy Executive's Database Guide	business	19.99	1389
BU1111	Cooking with Computers	business	11.95	1389
BU2075	You Can Combat Computer Stress!	business	2.99	736
BU7832	Straight Talk About Computers	business	19.99	1389
MC2222	Silicon Valley Gastronomic Treats	mod_cook	19.99	877
MC3021	The Gourmet Microwave	mod_cook	2.99	877
MC3026	The Psychology of Computer Cooking	UNDECIDED		877
PC1035	But Is It User Friendly?	popular_comp	22.95	1389
PC8888	Secrets of Silicon Valley	popular_comp	20	1389
PC9999	Net Etiquette	popular_comp		1389
PS2091	Is Anger the Enemy?	psychology	10.95	736

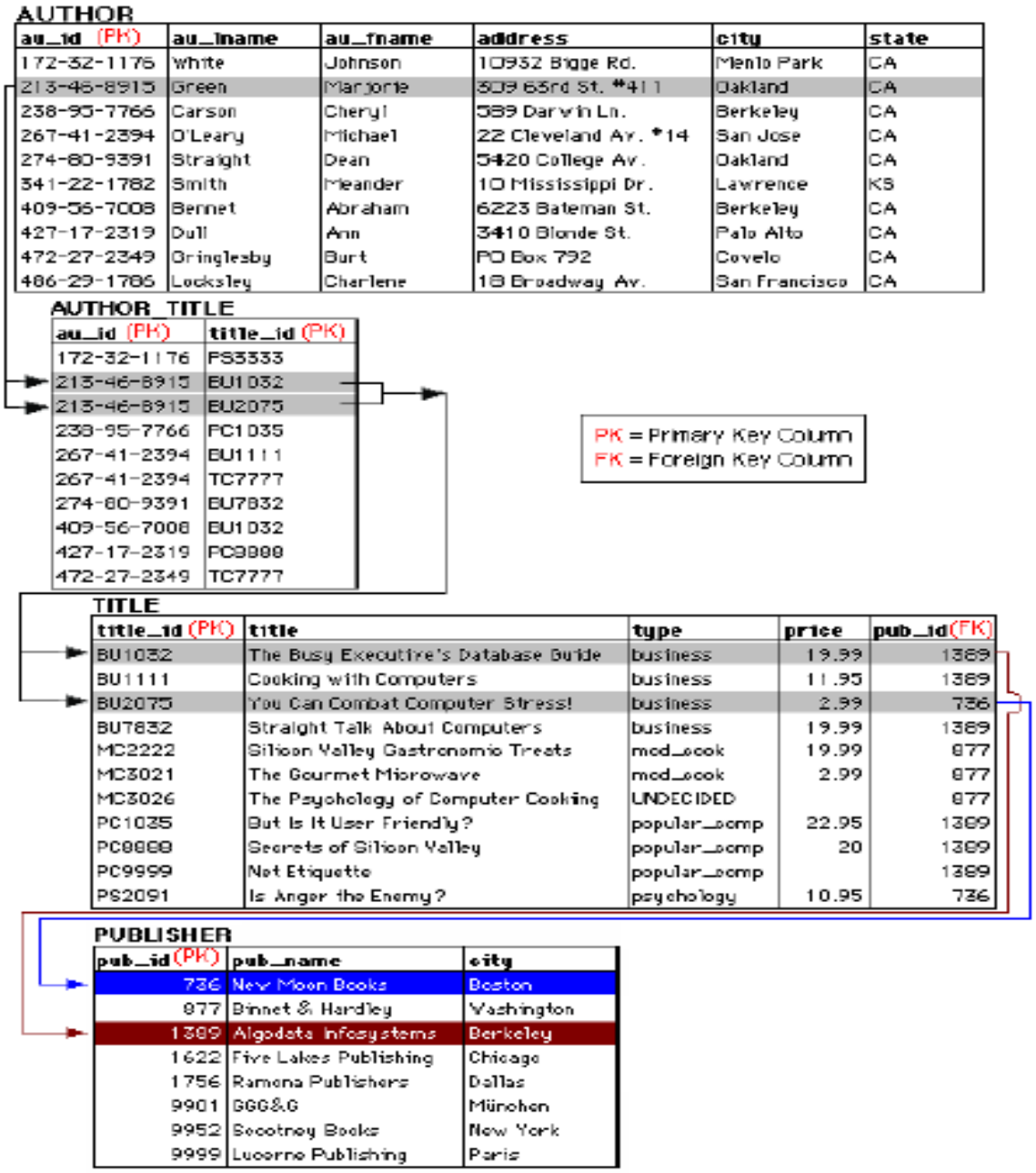
PUBLISHER

pub_id	pub_name	city
736	New Moon Books	Boston
877	Binnet & Hardley	Washington
1389	Algodata Infosystems	Berkeley
1622	Five Lakes Publishing	Chicago
1756	Ramona Publishers	Dallas
9901	GGG&G	München
9952	Scootney Books	New York
9999	Lucerne Publishing	Paris

AUTHOR_TITLE

au_id	title_id
172-32-1176	PS3333
213-46-8915	BU1032
213-46-8915	BU2075
238-95-7766	PC1035
267-41-2394	BU1111
267-41-2394	TC7777
274-80-9391	BU7832
409-56-7008	BU1032
427-17-2319	PC8888
472-27-2349	TC7777

AUTHOR	(au_id, au_lname, au_fname, address, city, state, zip)
TITLE	(title_id, title, type, price, pub_id)
PUBLISHER	(pub_id, pub_name, city)
AUTHOR_TITLE	(au_id, title_id)



AUTHOR

au_id (PK)	au_lname	au_fname	address	city	state
172-32-1176	White	Johnson	10932 Bigge Rd.	Menlo Park	CA
213-46-8915	Green	Marjorie	309 63rd St. #411	Oakland	CA
238-95-7766	Carson	Cheryl	589 Darwin Ln.	Berkeley	CA
267-41-2394	O'Leary	Michael	22 Cleveland Av. #14	San Jose	CA
274-80-9391	Straight	Dean	5420 College Av.	Oakland	CA
341-22-1782	Smith	Meander	10 Mississippi Dr.	Lawrence	KS
409-56-7008	Bennet	Abraham	6223 Bateman St.	Berkeley	CA
427-17-2319	Dull	Ann	3410 Blonde St.	Palo Alto	CA
472-27-2349	Gringlesby	Burt	PO Box 792	Covelo	CA
486-29-1786	Locksley	Charlene	18 Broadway Av.	San Francisco	CA

AUTHOR_TITLE

au_id (PK)	title_id (PK)
172-32-1176	PS3333
213-46-8915	BU1032
213-46-8915	BU2075
238-95-7766	PC1035
267-41-2394	BU1111
267-41-2394	TC7777
274-80-9391	BU7832
409-56-7008	BU1032
427-17-2319	PC9999
472-27-2349	TC7777

PK = Primary Key Column
FK = Foreign Key Column

TITLE

title_id (PK)	title	type	price	pub_id (FK)
BU1032	The Busy Executive's Database Guide	business	19.99	1389
BU1111	Cooking with Computers	business	11.95	1389
BU2075	You Can Combat Computer Stress!	business	2.99	736
BU7832	Straight Talk About Computers	business	19.99	1389
MC2222	Silicon Valley Gastronomic Treats	mod_cook	19.99	877
MC3021	The Gourmet Microwave	mod_cook	2.99	877
MC3026	The Psychology of Computer Cooking	UNDECIDED		877
PC1035	But Is It User Friendly?	popular_comp	22.95	1389
PC8888	Secrets of Silicon Valley	popular_comp	20	1389
PC9999	Net Etiquette	popular_comp		1389
PS2091	Is Anger the Enemy?	psychology	10.95	736

PUBLISHER

pub_id (PK)	pub_name	city
736	New Moon Books	Boston
877	Binnet & Hardley	Washington
1389	Algodata Infosystems	Berkeley
1622	Five Lakes Publishing	Chicago
1756	Ramona Publishers	Dallas
9901	GGG&G	München
9952	Scootney Books	New York
9999	Lucerne Publishing	Paris

التحويل من مخطط الكيان العلائقي 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، ويتم تطبيق هذه الخوارزمية كاملة، مع تجاوز الحالات التي لم تظهر في النموذج الأولي

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

خوارزمية التحويل Mapping Algorithm:

Step 1: Mapping of Regular Ent y Types	١. تحويل الكيانات العادية (القوية).
Step 2: Mapping of Weak Ent y Types	٢. تحويل الكيانات الضعيفة.
Step 3: Mapping of Binary 1:1 Relaon Types	٣. تحويل العلاقات الثنائية من النوع ١:١.
Step 4: Mapping of Binary 1:N Relaonshi p Types.	٤. تحويل العلاقات الثنائية من النوع ١:N.
Step5: Mapping of Binary M:N Relaonshi p Types.	٥. تحويل العلاقات الثنائية من النوع N:M.
Step 6: Mapping of Mulval ued ari bute s.	٦. تحويل الصفات متعددة القيم.
Step 7: Mapping of N-ary Relationship Types.	٧. تحويل العلاقات فوق الثنائية.

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

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، وتسميته بالمفتاح الرئيسي primary key(PK) وإذا كانت الصفة التي تمثل المفتاح من النوع المركب فإن المفتاح الرئيسي سيكون مجموعة الحقول التي تنشأ من الصفة المركبة.

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

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، ويكون المفتاح الرئيسي PK للجدول الجديد، عبارة عن مفتاح مركب مكون من المفتاح الأجنبي FK بالإضافة إلى المفتاح الجزئي (Partial Key) الخاص به.

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

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 ويفضل أن يكون الجدول الذي يحتوي على المفتاح الأجنبي، هو الجدول الذي يكون نوع قيد اشتراكه في العلاقة، من نوع (الاشتراك الكلي)

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

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 (1) إلى الجدول الآخر المرتبط بالعلاقة (M)، بغض النظر عن نوع قيد الاشتراك.

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

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 ويضم الجدول الثالث حقلين كمفتاحين أجنبيين يمثلان المفتاحين الرئيسيين في الجدولين، ويمكن إضافة أي حقل آخر يكون له مغزى، كأن تكون العلاقة لها صفة بذاتها، فتتحول الصفة إلى حقل في الجدول الجديد.

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

٦. **تحويل الصفات متعددة القيم:** يتم في هذه الحالة، عادة، إنشاء جدول جديد يضم الصفة المتعددة القيم كحقل، ويضاف 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، والصفات ذات القيم المشتقة تلغى من الجدول، لأنها صفات قابلها للاشتقاق من صفات أخرى، فلا داعي لوجودها

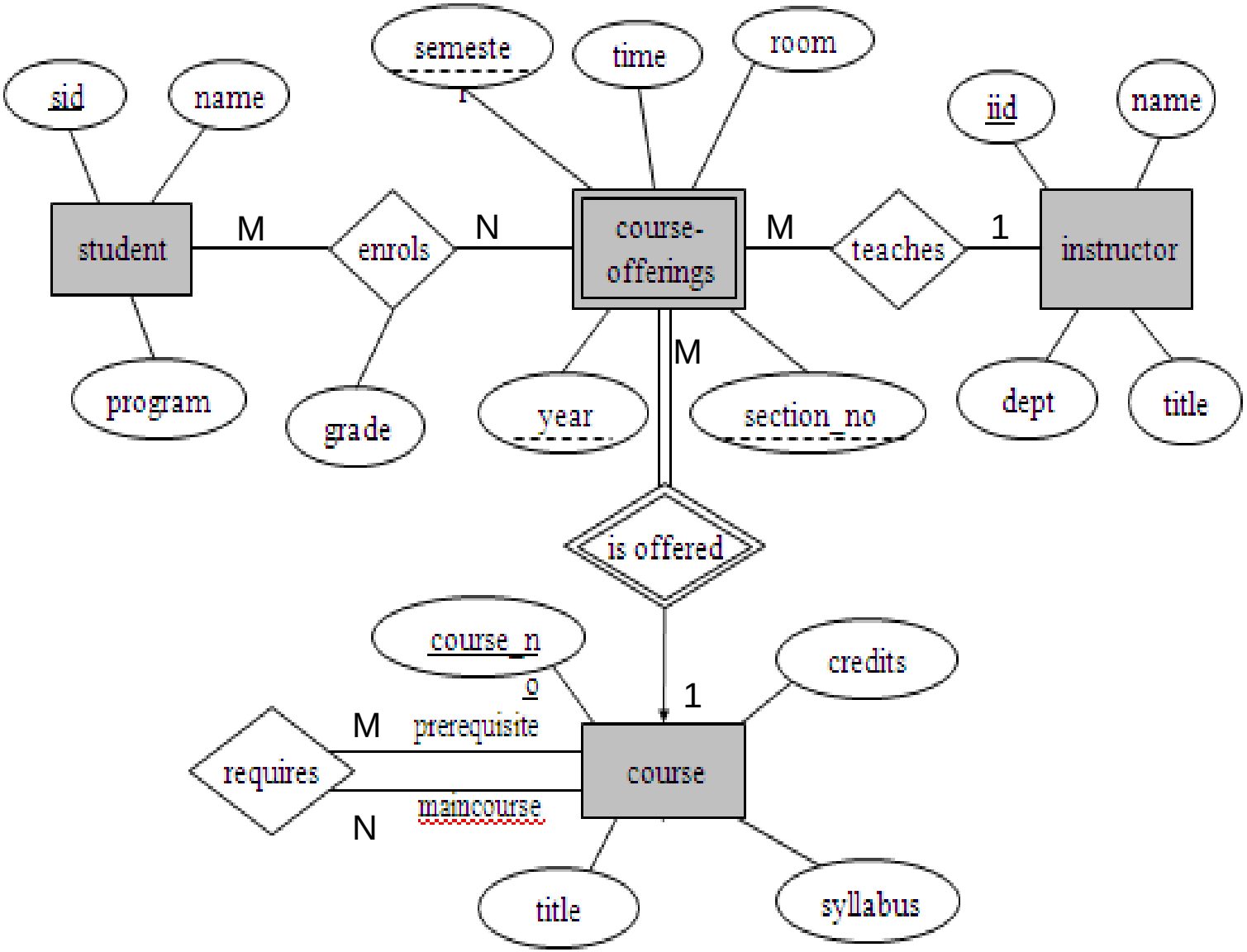
التحويل من مخطط الكيان العلائقي إلى مخطط قواعد بيانات
Mapping ERD to DB schema

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

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

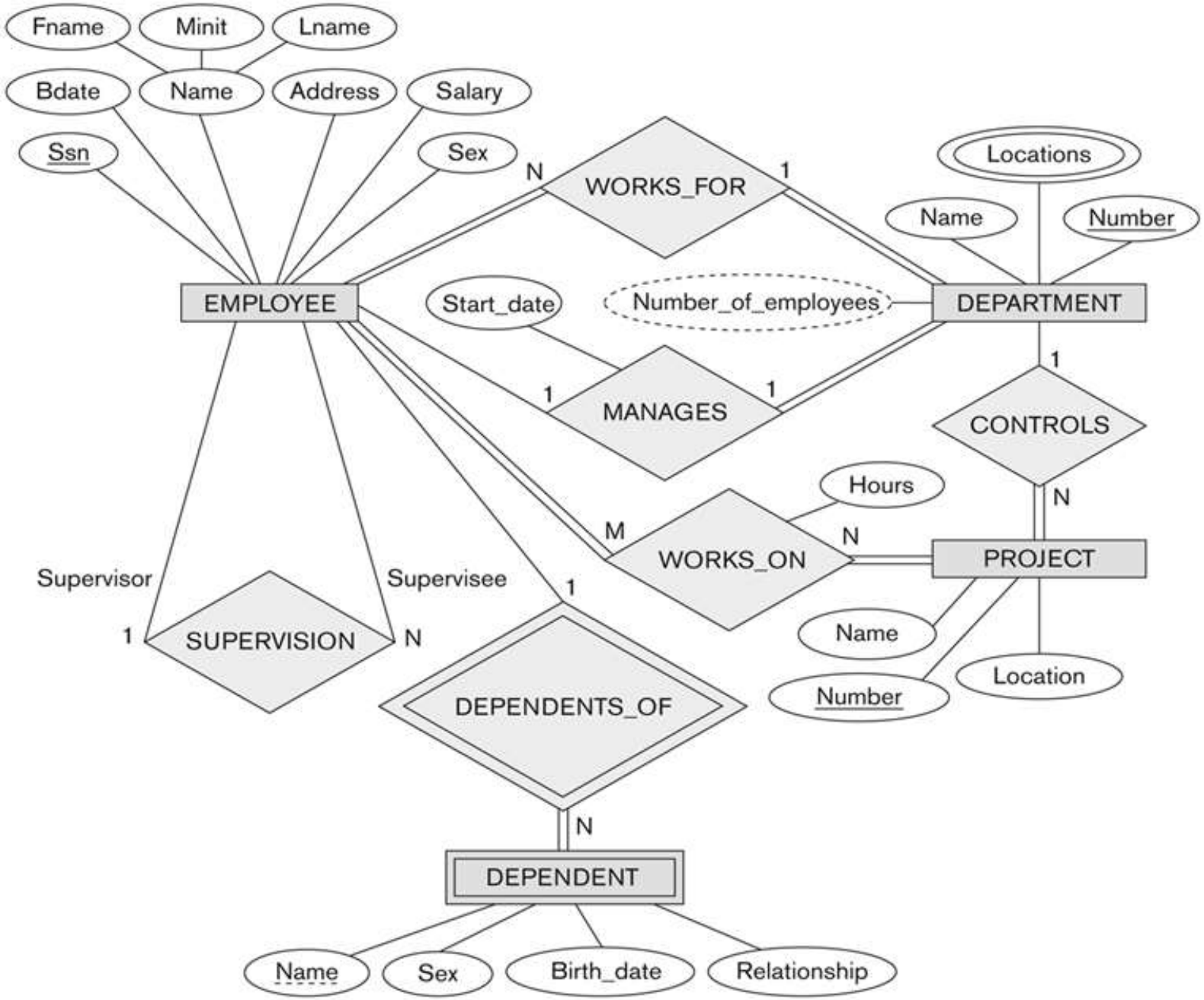
تمرين: حول كل من نماذج الكيان العلائقي التاليين إلى ما يقابلها من مخطط قواعد البيانات Database Schema

مثال 1



E-R diagram for a university.

مثال2


Fname
Minit
Lname
Bdate
Name
Address
Salary
Ssn
Sex
N
WORKS_FOR
1
Locations
Name
Number
EMPLOYEE
Start_date
Number_of_employees
DEPARTMENT
1
1
MANAGES
1
CONTROLS
Hours
N
M
WORKS_ON
N
PROJECT
Supervisor
Supervisee
1
1
SUPERVISION
N
Name
Location
Number
DEPENDENTS_OF
N
DEPENDENT
Name
Sex
Birth_date
Relationship

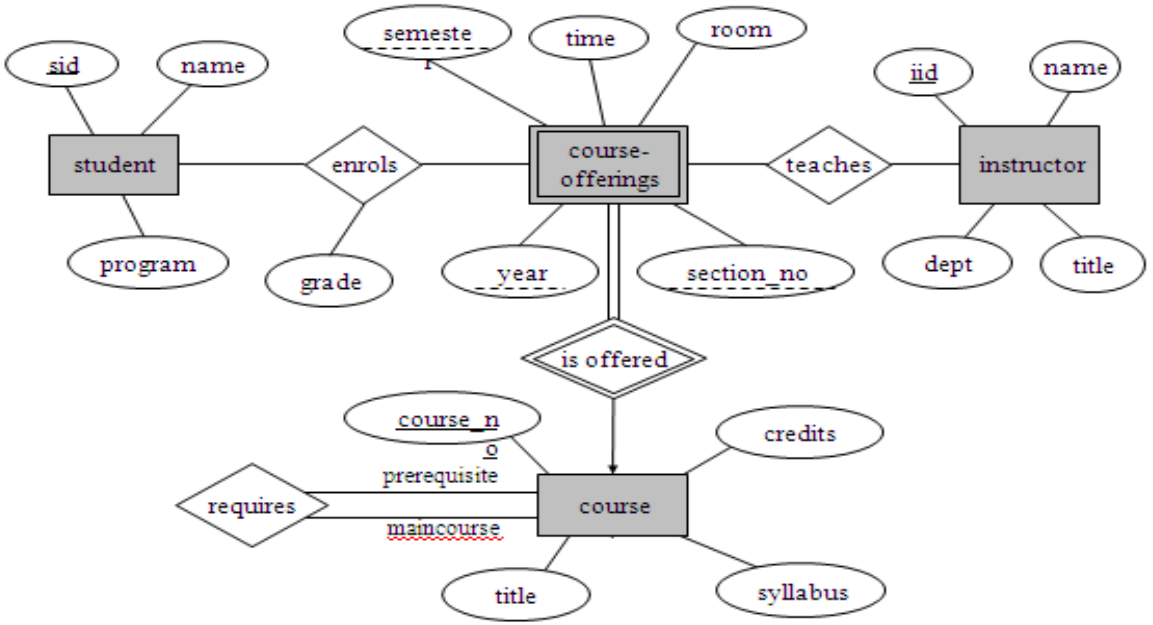
محاضرة رقم 5

أمثلة على تصميم قاعدة البيانات

التحويل من مخطط الكيان العلائقي إلى مخطط قواعد بيانات

أمثلة للتحويل من مخطط الكيان العلائقي إلى مخطط قواعد بيانات

مثال 1 (مخطط الكيان العلائقي لجامعة)



E-R diagram for a university.

حول كل من نموذج الكيان العلائقي السابق إلى ما يقابله من مخطط قواعد البيانات Database

1. **نبدأ بتحويل الكيان العادي:**

يتم تمثيل الكيان العادي (القوي) بشكل المستطيل أحادي الإطار، ويحتوي مخطط الكيان العلائقي السابق على ثلاثة كيانات هي:

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

نلاحظ في هذا المثال أن كل الصفات من النوع البسيط

ويكون المفتاح الرئيسي (Primary Key) للجدول هو مجموعة صفات المفتاح الرئيسي المرتبطة بالكيان

وينتج عن عملية التحويل الجداول التالية:

STUDENT	***sid***	***name***	***program***

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

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

2. تحويل الكيان الضعيف:

يتم تمثيل الكيان الضعيف بشكل مستطيل مزدوج الإطار، وسبب 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)

ويحتوي هذا الكيان على صفات المفتاح الجزئي التالية:

- السنة (Year)
- الفصل (Semester)
- رقم الشعبة (Section–no)

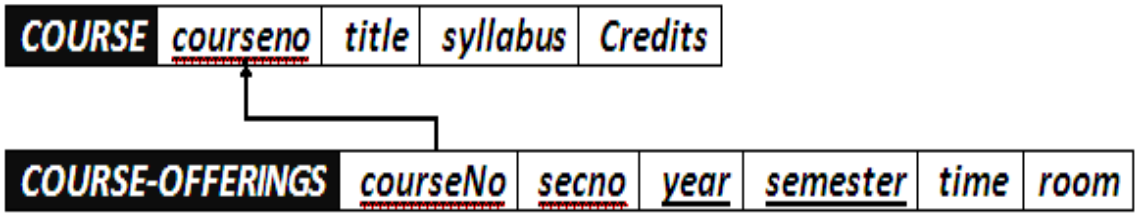
ويجب أن يرتبط الكيان الضعيف بكيان قوي بواسطة علاقة تعريف تمثل علاقة التعريف بشكل معين مزدوج الإطار، وذلك في سبيل تقوية الكيان الضعيف.

وتتم عملية تحويل الكيان الضعيف بتحويله إلى جدول يحمل اسم الكيان الضعيف، ويحتوي حقولاً من الصفات المرتبطة به، بالإضافة 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 ويكون المفتاح الرئيسي للجدول هو مفتاح الكيان القوي بالإضافة الى المفاتيح الجزئية في الكيان الضعيف.

وينتج عن عملية التحويل الجدول التالي:

COURSE-OFFERINGS	courseno	secno	year	semester	time	room

وبذلك يرتبط جدول الشعب الفصلية المقترحة (Course–Offerings) بجدول المقرر (Course) بوجود المفتاح الأجنبي (Foreign Key) رقم المقرر (courseno)

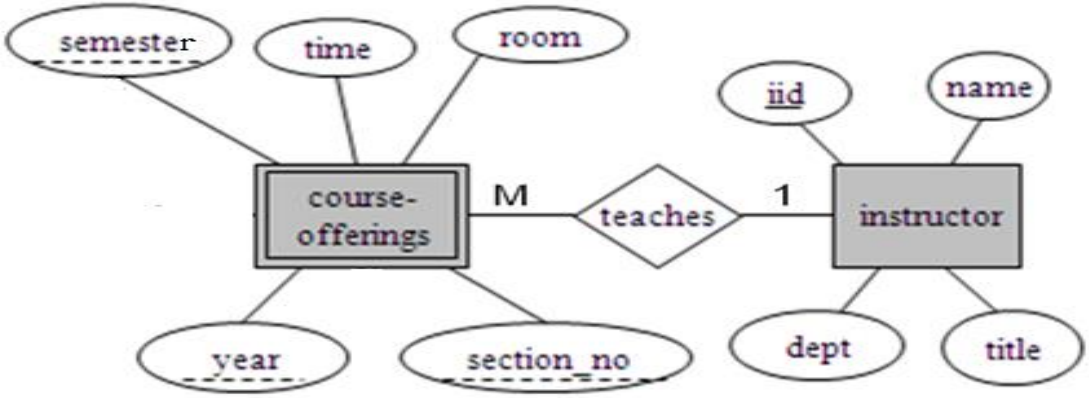


3. تحويل العلاقات :

يتم تمثيل العلاقة بشكل معين أحادي الإطار. في هذه الحالة يتم التعامل مع العلاقة حسب نوعها كل على حده. ويتم تجاهل علاقة التعريف (شكل معين مزدوج الإطار)، لأنه تم بناؤها مسبقا عند تحويل الكيان الضعيف.

أ. تحويل علاقة واحد إلى كثير:

في هذا المثال لدينا علاقة يُدرس(teaches) بين كيان المحاضر (Instructor) وكيان الشعب الفصلية المقترحة(Course–Offerings)،

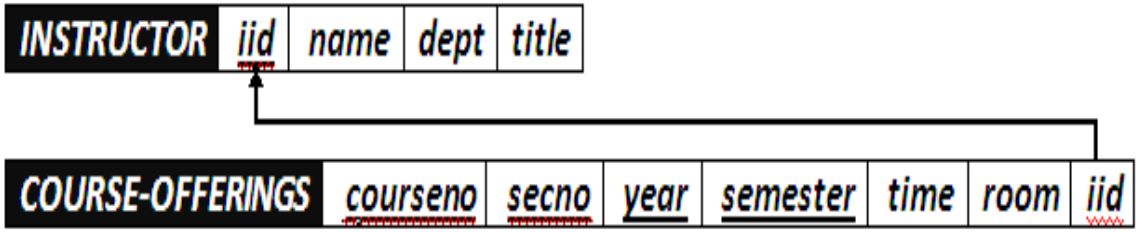


وعليه يتم أخذ المفتاح الرئيسي (iid) من جدول Instructor كونه طرف العلاقة واحد، ويضاف كمفتاح أجنبي في جدول course–offerings، وبالتالي يتم تعديل جدول

Course– Offerings ليصبح بالشكل التالي:

COURSE-OFFERINGS	courseno	secno	year	semester	time	room	iid

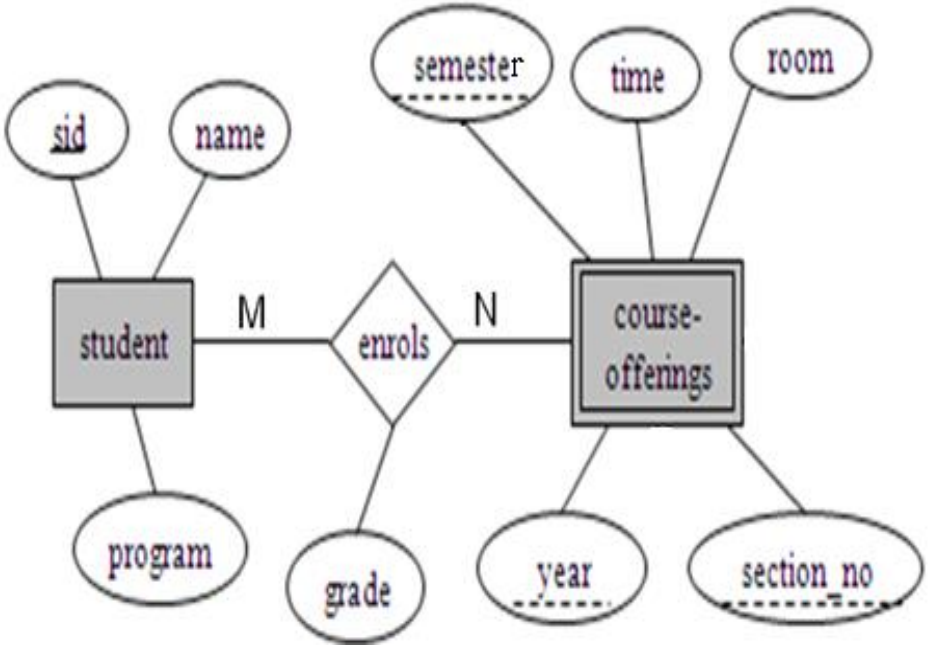
وبذلك يرتبط جدول الشعب الفصلية المقترحة (Course–Offerings)بجدول المحاضر (Instructor) بوجود المفتاح الأجنبي رقم المحاضر (iid)



ب. تحويل علاقة كثير إلى كثير:

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

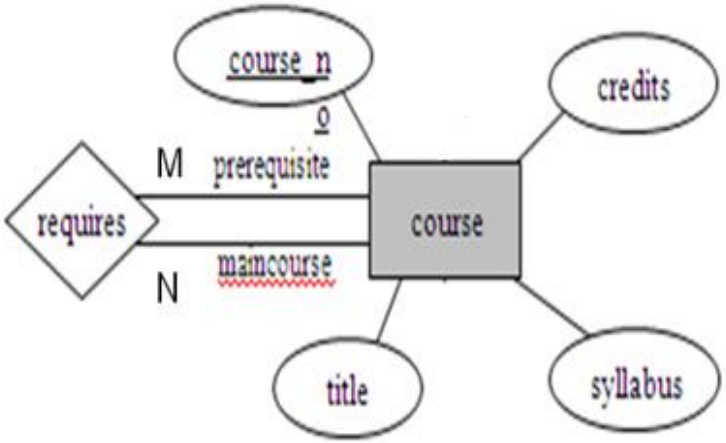
- علاقة يُسَجِّل (enrolls)



ينتج عن هذه العلاقة الجدول التالي:

ENROLS	***sid***	***courseno***	***secno***	***year***	***semester***	***grade***

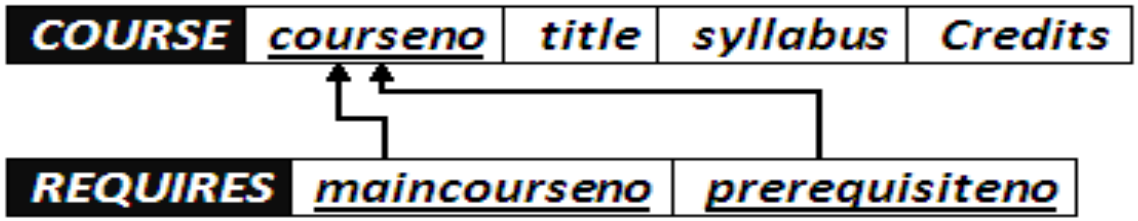
- علاقة يتطلب (Requires)



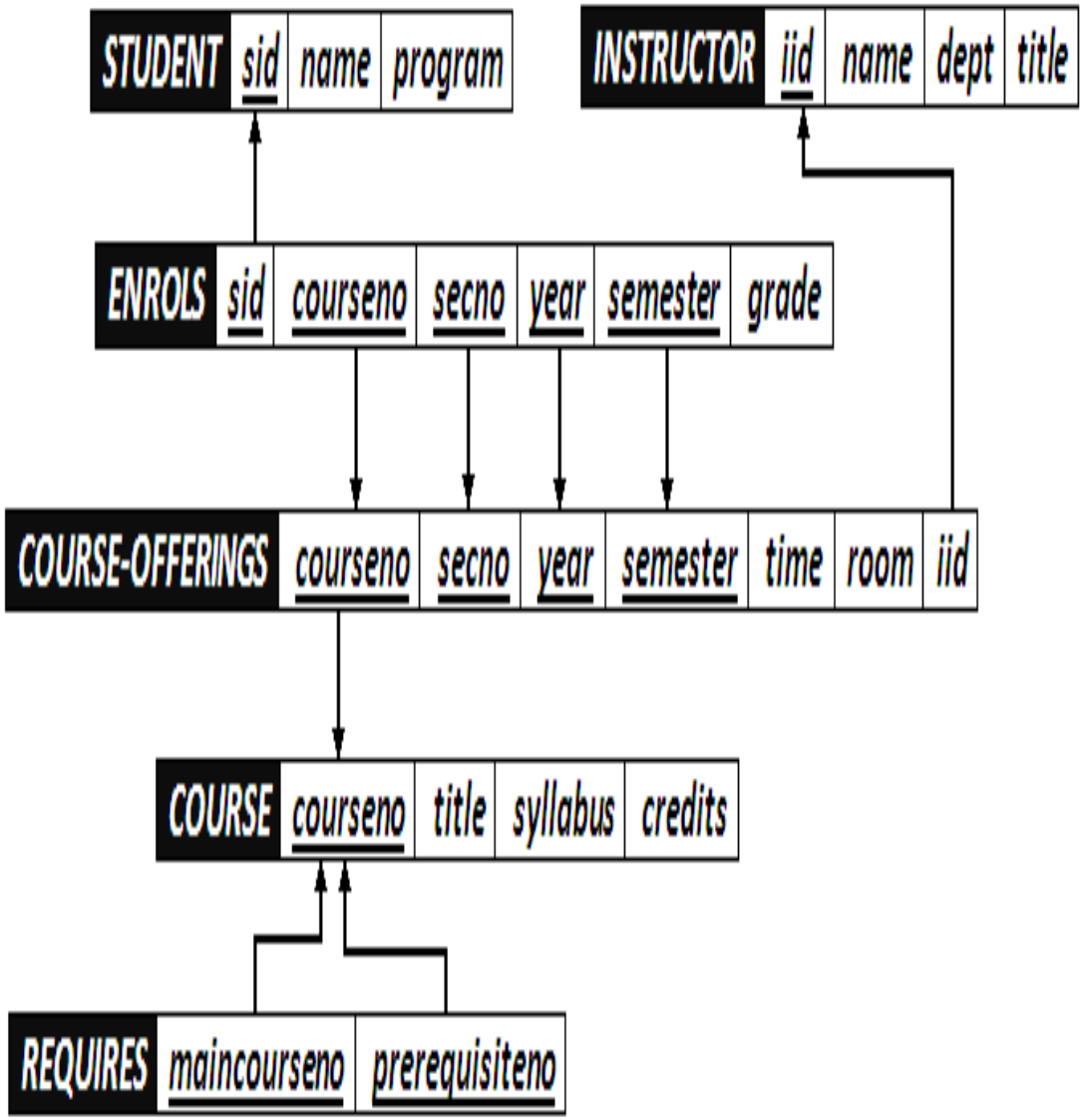
وينتج عن هذه العلاقة الجدول التالي:

REQUIRES	***maincourseno***	***prerequisiteno***

وبذلك يرتبط جدول يتطلب (Requires) مع جدول المقرر (course)

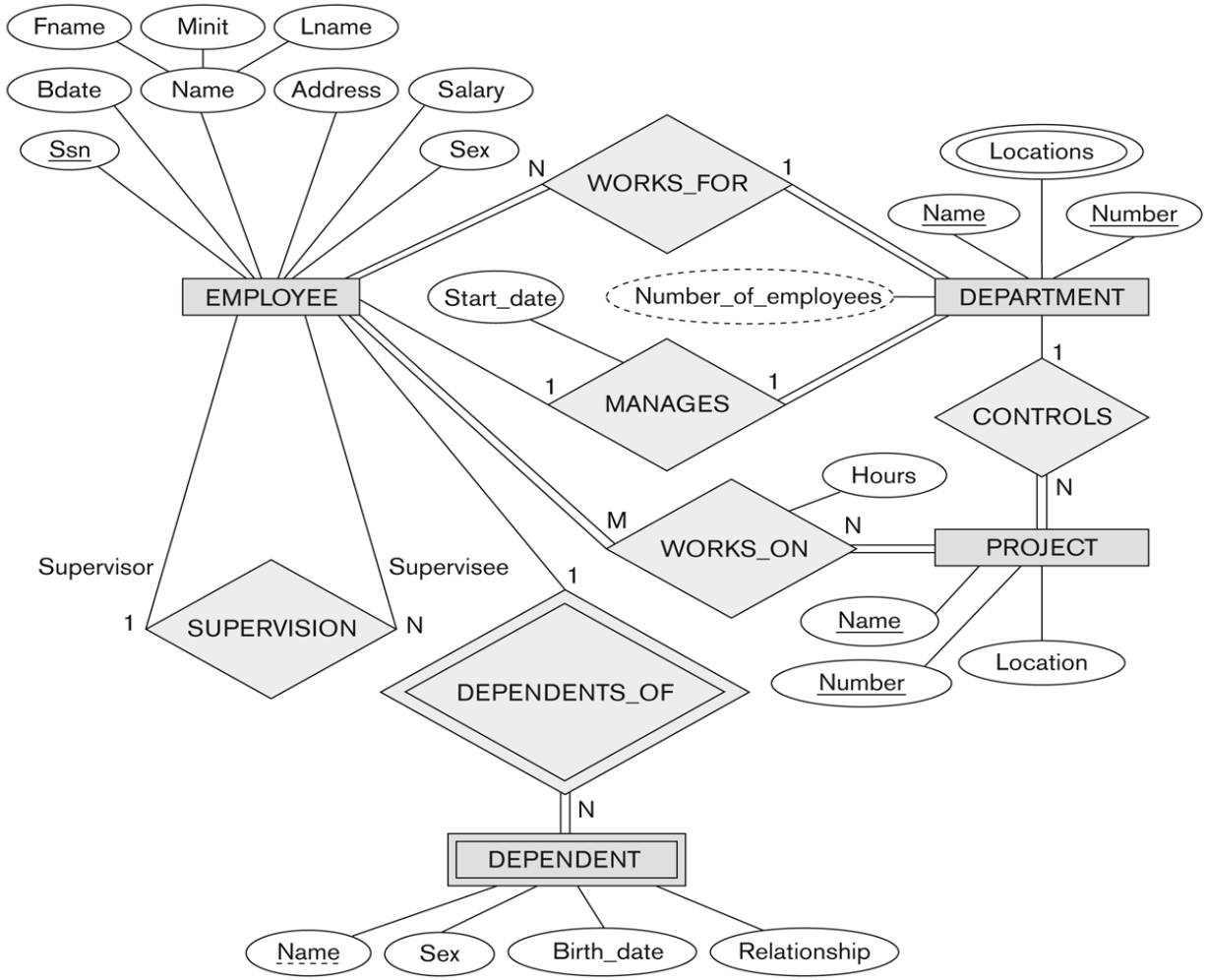


وبذلك نكون قد أنشأنا مخطط قواعد البيانات الناتج عن التحويل للمثال رقم (1)، وهو على الشكل التالي:


STUDENT sid name program
INSTRUCTOR iid name dept title
ENROLS sid courseno secno year semester grade
COURSE-OFFERINGS courseno secno year semester time room iid
COURSE courseno title syllabus credits
REQUIRES maincourseno prerequisiteno

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.

مثال 2 (مخطط الكيان العلائقي لشركة)



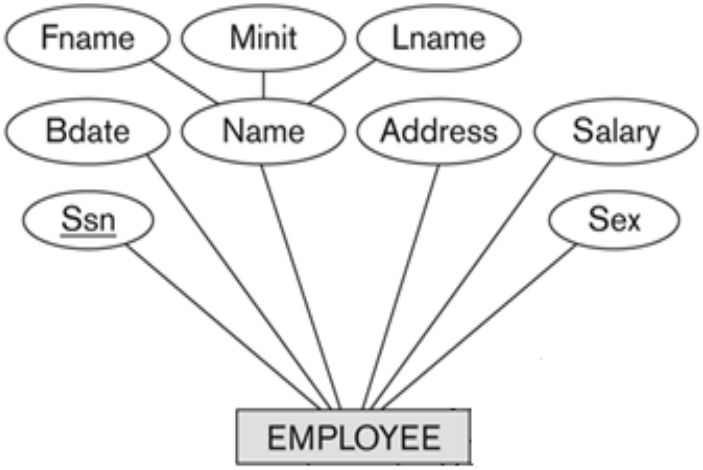
1. **نبدأ بتحويل الكيان العادي:**

يحتوي مخطط الكيان العلائقي السابق على ثلاثة كيانات عادية هي:

- الموظف (Employee)
- القسم (Department)
- المشروع (Project)

أ. **كيان الموظف (Employee)**

نلاحظ في هذا الكيان وجود صفة مركبة هي صفة الاسم (Name) والتي تتكون من الصفات الجزئية الاسم الأول(Fname)، وحرف الاسم الأوسط (Minit)، والاسم الأخير (Fname) وكما اوضحنا سابقا، فإن الصفة المركبة في عملية التحويل تدرج بصفاتها الجزئية فقط.



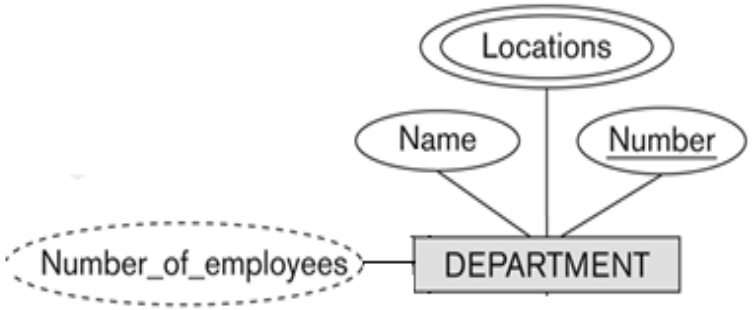
وينتج عن عملية التحويل الجدول التالي:

EMPLOYEE	Ssn	Fname	Minit	Lname	Bdate	Address	Salary	Sex

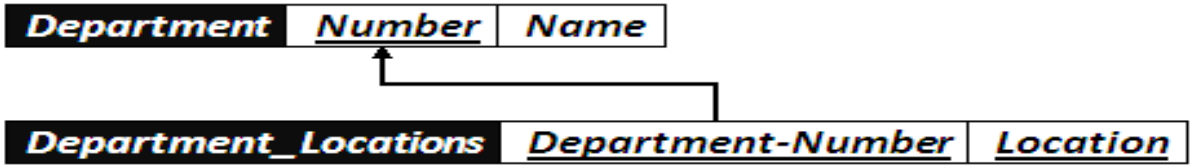
ب. كيان القسم (Department)

نلاحظ في هذا الكيان وجود صفة مشتقة هي عدد الموظفين (Number of employees)، وذكرنا سابقا أننا في عملية التحويل نتجاهل هذه الصفة، ليتم بناؤها لاحقاً بجملة استعلام.

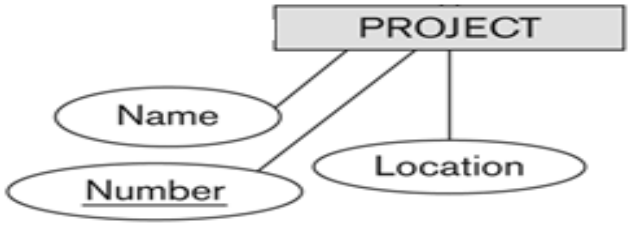
كما نلاحظ وجود صفة متعددة القيمة وهي المواقع (locations) وذكرنا سابقا أن الصفة متعددة القيمة يتم تحويلها إلى جدول مستقل مع المفتاح الرئيسي للكيان، ويكون اسم الجدول مكون من اسم الكيان مضافا إليه اسم الصفة متعددة القيمة.



وينتج عن عملية التحويل الجدولين التاليين :



ت. كيان المشروع (Project)



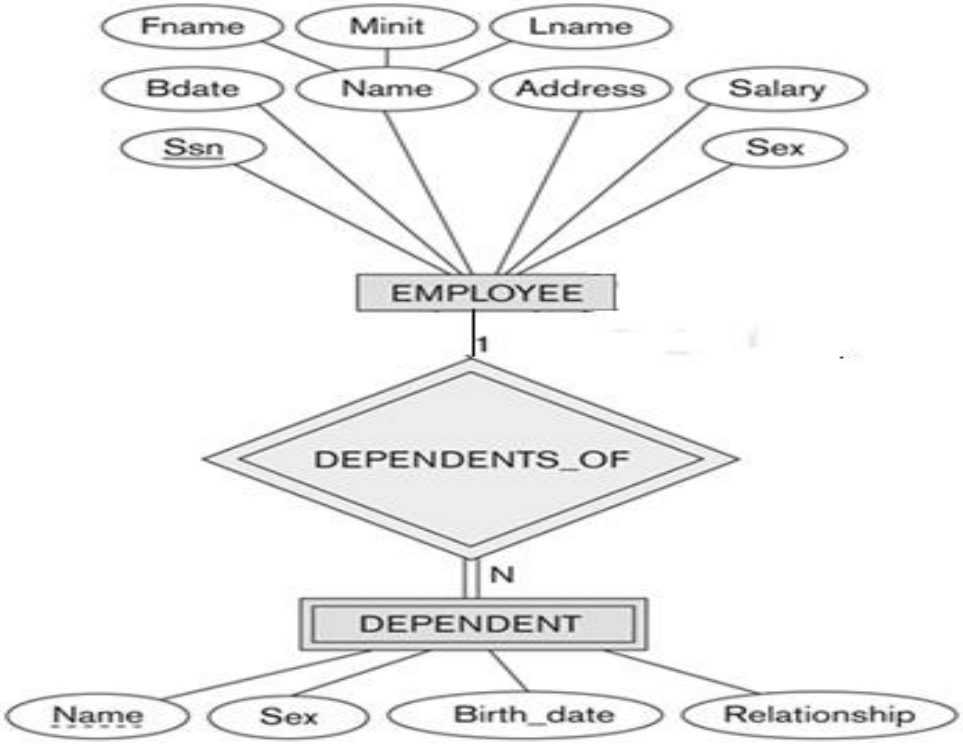
هذا الكيان يحتوي على صفات بسيطة فقط وبالتالي ينتج عن عملية التحويل الجدول التالي:

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

2. **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.



ويحتوي هذا الكيان على صفة المفتاح الجزئي الاسم (Name)، التالية:

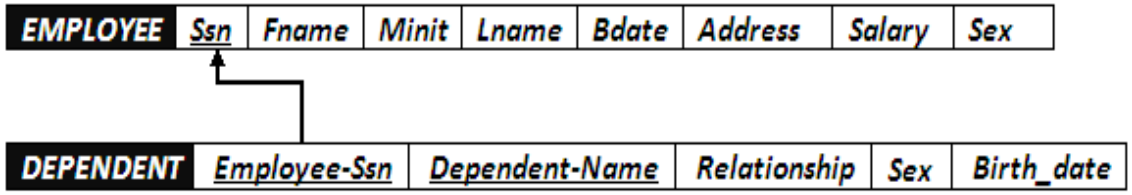
ويرتبط هذا الكيان الضعيف بكيان قوي هو كيان الموظف (Employee) بواسطة علاقة التعريف يعتمد على (Dependent_of).

وتتم عملية تحويل الكيان الضعيف بتحويله إلى جدول يحمل اسم الكيان الضعيف(Dependent)، ويحتوي حقولا من الصفات المرتبطة به، بالإضافة إلى حقل المفتاح الرئيسي من جدول الكيان القوي (Employee)المرتبط معه بعلاقة التعريف (Dependent–of) يكون المفتاح الرئيسي للجدول هو مفتاحا للكيان القوي (Empoyee–Ssn) بالإضافة إلى المفتاح الجزئي في الكيان الضعيف (Dependent–Name).

وينتج عن عملية التحويل الجدول التالي:

DEPENDENT	Employee-Ssn	Dependent-Name	Relationship	Sex	Birth_date

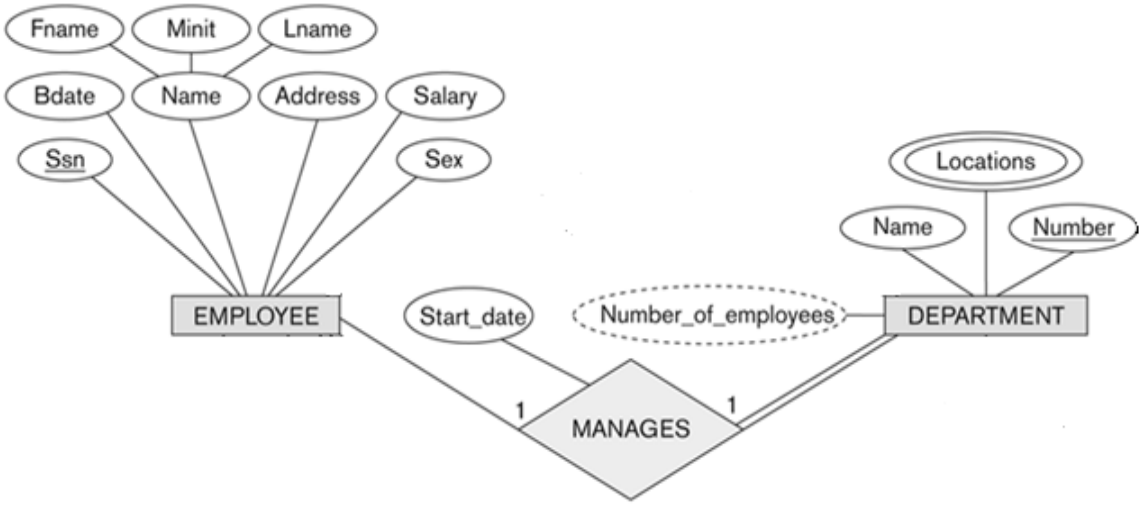
وبذلك يرتبط جدول (Dependent) بجدول (Employee) بوجود المفتاح الأجنبي رقم الموظف (Employee–Ssn).



3. تحويل العلاقات :

3.أ. تحويل علاقة واحد إلى واحد:

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

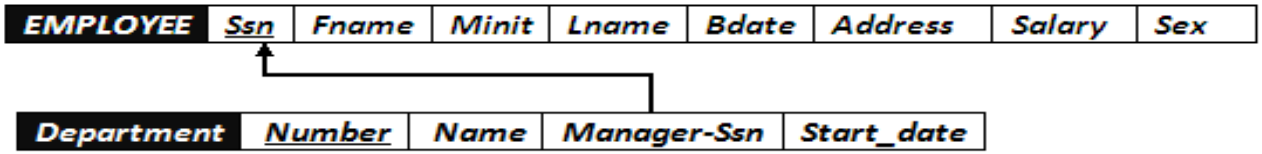


وبناءً على النقطة السابقة، يتم أخذ نسخة من المفتاح الرئيسي لجدول الموظف (Employee_Ssn)، مضافا إليها الصفة على العلاقة (start_date)، ووضعهما كتعديل على جدول القسم (department)، ويفضل إعادة تسمية المفتاح الرئيسي ليدل على العلاقة وهي علاقة ادارة ، فهو مدير (Manager)، وبالتالي بدلا من التسمية Employee_Ssn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

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

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

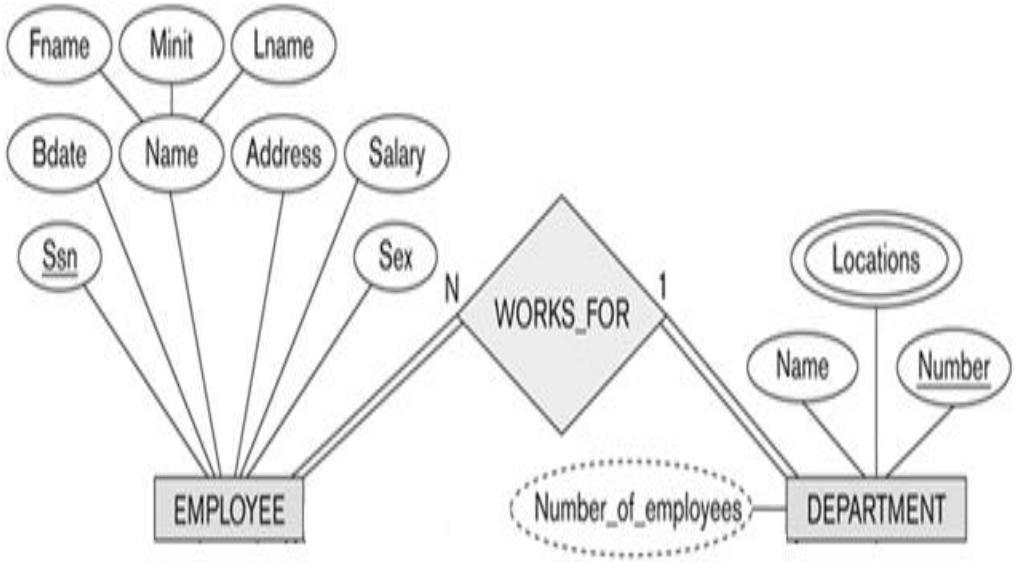
وبذلك يرتبط جدول (Department) بجدول (Employee) بوجود المفتاح الأجنبي رقم الموظف (Employee-Ssn).



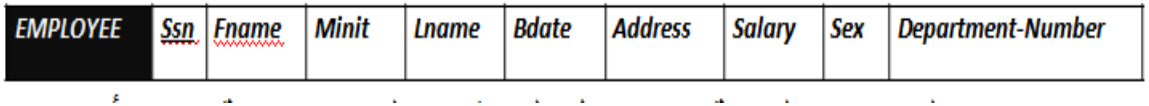
3.ب. تحويل علاقة واحد إلى كثير:

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

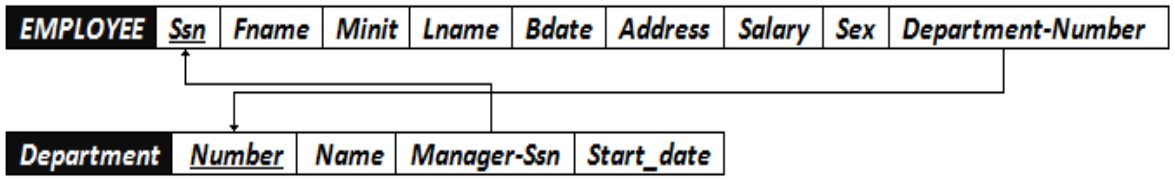
- **علاقة موظف يعمل في قسم (Works_for)**



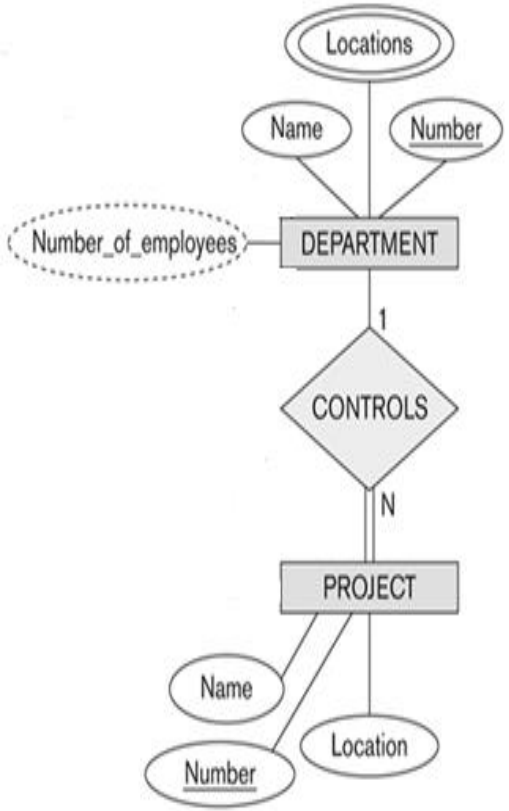
نتيجة لهذه العلاقة يتم التعديل على جدول الموظف (Employee) ليحتوي على رقم القسم (Department_Number) الذي يعمل فيه، ليصبح جدول الموظف (Employee) بالشكل التالي:



وعليه تتوثق العلاقة بين جدول الموظف والقسم، بإضافة مفتاح أجنبي جديد، وهو رقم القسم في جدول الموظف، كما يظهر في الشكل التالي:



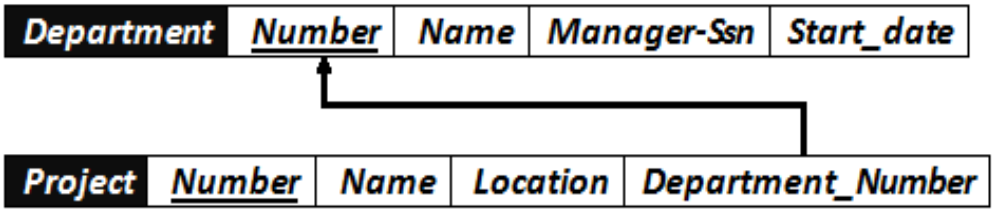
- **علاقة قسم يتحكم بمشروع (Controls)**



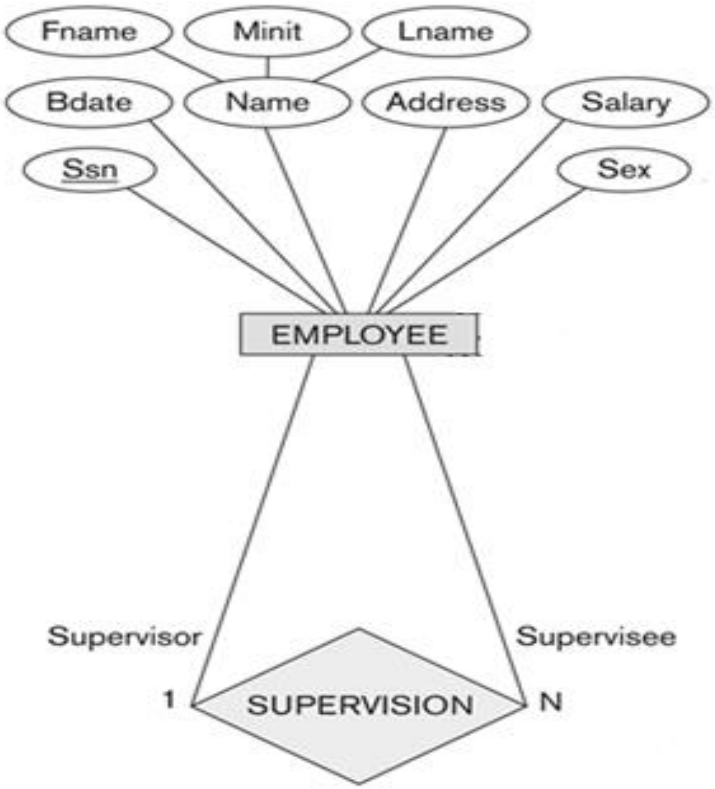
نتيجة لهذه العلاقة يتم التعديل على جدول المشروع (Project) ليحتوي على رقم القسم (Department_Number) الذي يتحكم في المشروع، ليصبح جدول المشروع (Project) بالشكل التالي:

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:

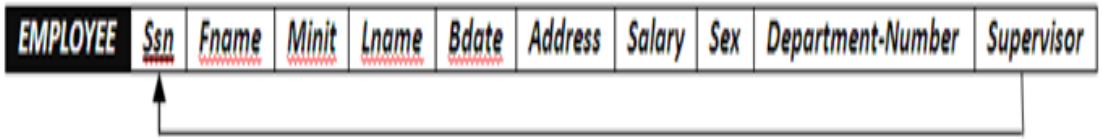


- **علاقة إشراف موظف على موظف (Supervision)**



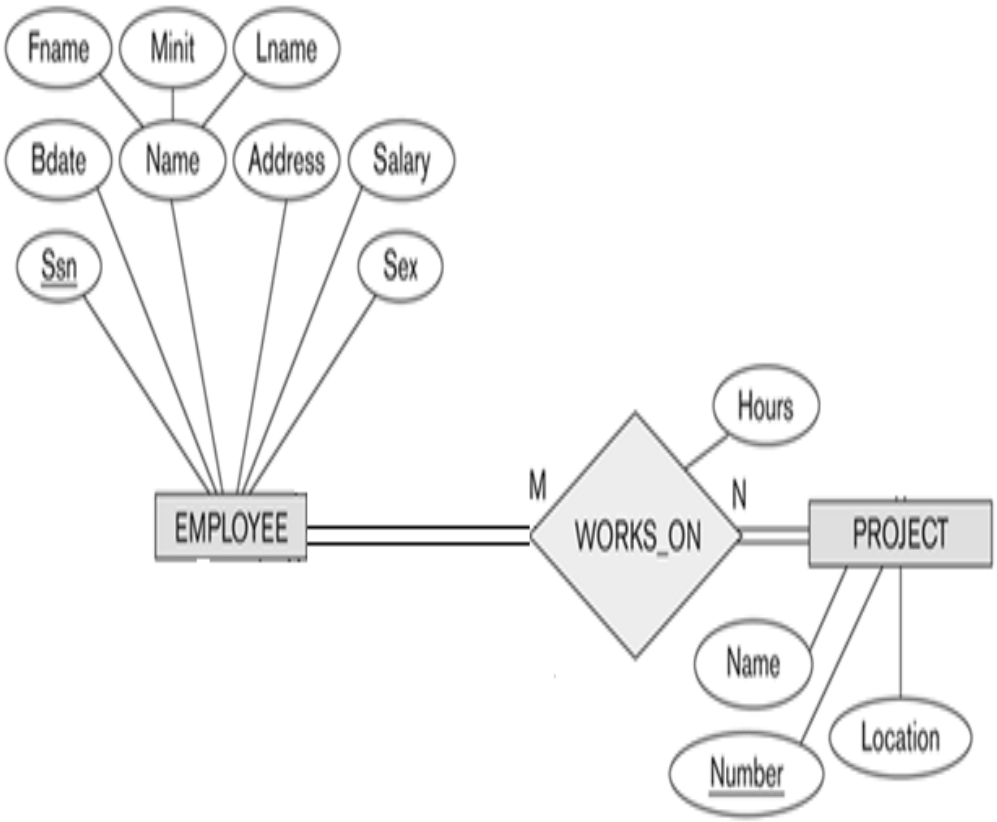
هذه العلاقة هي علاقة أحادية ذات تغذية راجعة أي علاقة كيان على نفسه.

نتيجة هذه العلاقة 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:



3.ج. تحويل علاقة كثير إلى كثير :

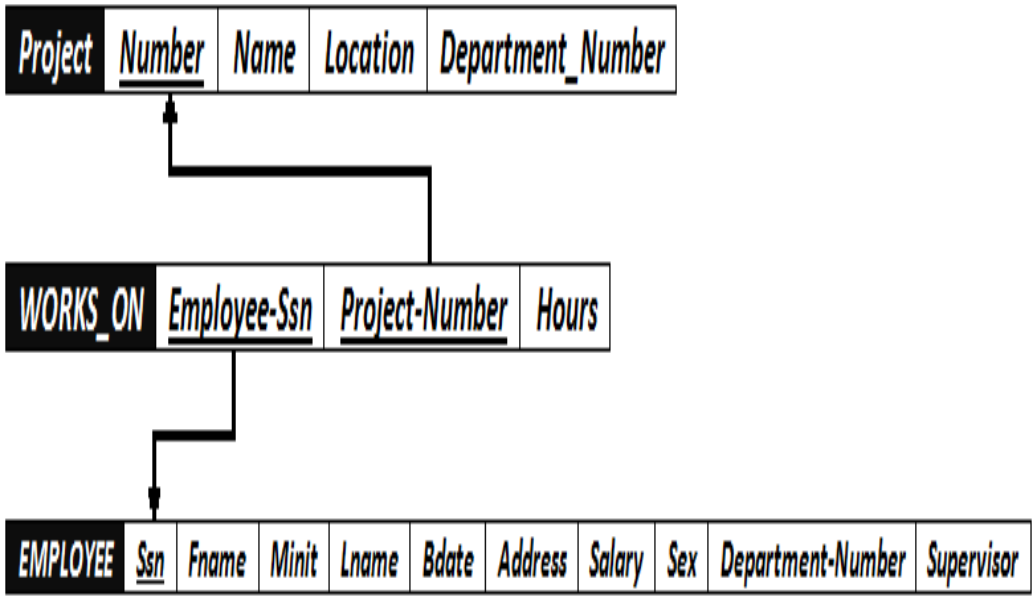
في هذا المثال، يوجد لدينا علاقة واحدة من نوع كثير إلى كثير، هي علاقة موظف يعمل على مشروع (Works_on).



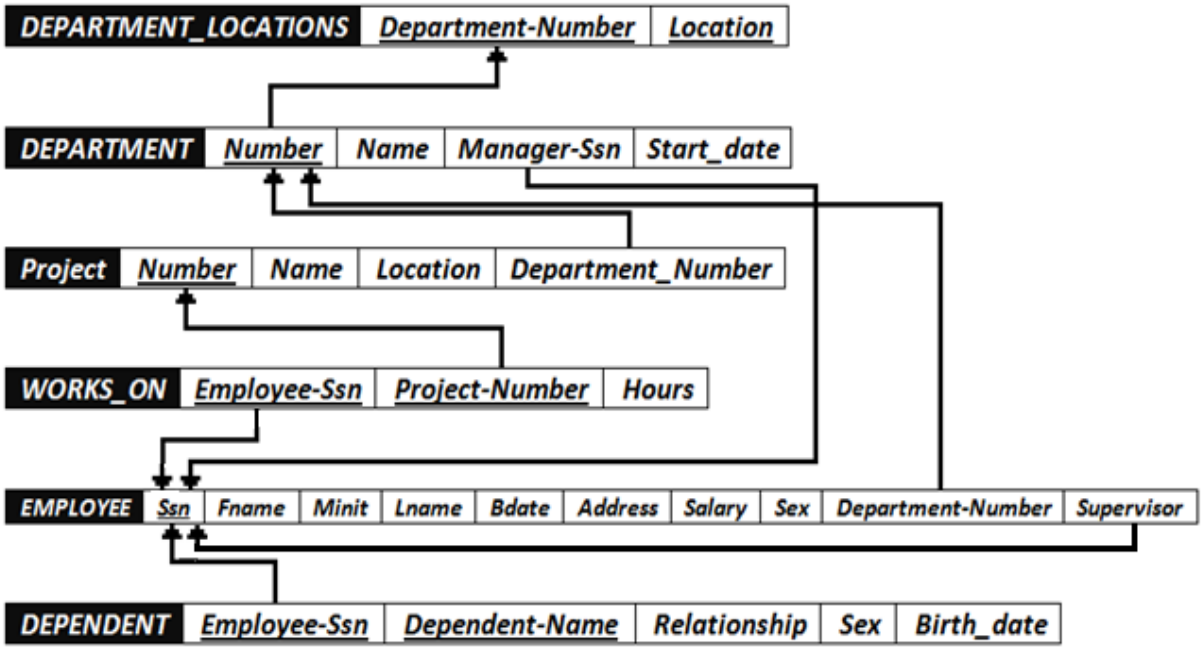
وفي هذه الحالة يتم إنشاء جدول جديد تحت اسم العلاقة (Works–on)، ونقوم بأخذ المفتاح الرئيس من كل الجدولين، الموظف (Employee) والمشروع (Project)، بالإضافة إلى الصفة عدد الساعات (Hours)المرتبطة بالعلاقة، ويصبح لدينا الجدول التالي:

WORKS_ON	Employee-Ssn	Project-Number	Hours

وبذلك يرتبط جدول الموظف (Employee) وجدول المشروع (Project) عن طريق الجدول الجديد (Works_on) عن طريق المفاتيح الأجنبية رقم الموظف (Employee–Ssn) من جدول الموظف Employee، ورقم المشروع (Project–Number) من جدول المشروع (Project).



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


DEPARTMENT_LOCATIONS
Department-Number
Location
DEPARTMENT
Number
Name
Manager-Ssn
Start_date
Project
Number
Name
Location
Department_Number
WORKS_ON
Employee-Ssn
Project-Number
Hours
EMPLOYEE
Ssn
Fname
Minit
Lname
Bdate
Address
Salary
Sex
Department-Number
Supervisor
DEPENDENT
Employee-Ssn
Dependent-Name
Relationship
Sex
Birth_date