



جامعة القادسيه

كلية علوم الحاسوب ونقانة المعلومات

الفصل الدراسي الرابع [ علوم + نقانة ]

أساسيات

نظم قواعد البيانات

**Fundamentals of**  
**Database Systems**

ترجمة الفصول (2،3،5،7)

إعداد / عثمان هادي

## أساسيات نظم قواعد البيانات

### ترجمة الفصول (2، 3، 5، 7)

((سبحانك لا علم لنا إلا ما علمتنا)) /// صدق الله العظيم

الإخوة الزملاء هذا ما وفقني الله أن أقوم به ولم يسعفني الزمن لأقوم بالترجمة الكاملة للفصول (3، 5، 7) وأملئ أن أفيد وأستفيد وفي نهاية المطاف أسألكم خالص الدعاء وليوفقنا الله تعالى في هذه الإمتحانات . // // // عثمان صدى فبراير 2011م .

### الفصل الثاني :

مفاهيم نظام قاعدة البيانات والهندسة المعمارية :

#### نماذج البيانات Data Models

تعريف نموذج البيانات : مجموعة من المفاهيم لوصف هيكل قاعدة بيانات ، وبعض القيود التي تتقيد بها قاعدة البيانات .  
نموذج بيانات العمليات : هو عمليات لتحديد قاعدة البيانات واسترجاع التحديثات من خلال الإشارة إلى مفاهيم نموذج البيانات.  
العمليات على نموذج البيانات قد تشمل العمليات الأساسية والعمليات المعرفة من قبل المستخدم.

#### أنواع الـ Data Models

(1) القاعدة الافتراضية Conceptual افتراضى ، semantic ، high :

وهو أهم جزء يختص بالمستخدم (User) وهو يوفر مفاهيم قريبة من تعامل المستخدم مع البيانات ويسمى (Object Based)

وتتكون القاعدة من طبقتي : المستخدم (User) والبيانات (data) / internal

الرسم التالى يوضح مفهوم الـ Object Based :

كمثال جدول الطلاب (Students) به عدة حقول وخصائص . هذا الجدول يخزن فى القاعدة ويتعامل المستخدم مع البيانات الداخلية (Internal) فقط على أنها عناصر مخزنة بطريقة افتراضية . ولنفترض أم لدينا 200 طالب

Name الإسم	No الرقم	Age العمر	Semester الفصل

(2) القاعدة الفيزيائية Physical (القاعدة الحقيقية) ويسمى أيضاً Low level أو internal

وهو الجزء الذى يتعلق بالبيانات المخزنة فى القاعدة كالوصف والجداول وغيرها . وهو يوفر مفاهيم وصف التفاصيل لكيفية تخزين البيانات فى الحاسوب .

(3) القاعدة الهجين أو المدموجة (implement) :

وهو الجزء المختلط الذى يوازى القاعدة الافتراضية والقاعدة الفيزيائية أى أنه يوازى بين المستخدمين وتفاصيل البيانات المخزنة .

تاريخ الـ History of Data Models :

القاعدة العلائقية ، النموذج العلائقى (Relationall) المقترح فى عام 1970 من قبل كود بالانكليزية لشركة (آي بي إم) ، والنظام التجارى الأول فى 1981-82. الآن فى العديد من المنتجات التجارية (DB2 ، وأوراكل ، مزود خادم ، سايببىس ، ينفورميكس).

أساسيات نظم قواعد البيانات الفصل الرابع (علوم + تقانة) / ترجمة عثمان صدى /// الأجزاء (2، 3، 5، 7) صفحة رقم (1)

## القاعدة الشبكية (النموذج الشبكي) : Network data Model

المحاسن :

- (1) له القدرة على وضع العلاقات المعقدة ويوفر بعض العلاقات مثل (إضافة / حذف) وغيرها . كمثال الطالب له علاقة بالقسم ..
- (2) يستخدم جمل وله لغة بسيطة محورية .

## القاعدة الهرمية (النموذج الهرمي) : Hierarchical data Model

- المحاسن : (1) سهل البناء ، بسيط التصميم (2) يعمل في مجال المنظمات التي تختص في مجال المترجمات وغيرها .
- (3) سهل اللغة ويستخدم بعض الجمل السهلة مثل Next وغيرها
- المساوى : (1) طبيعة العالجة الحوارية (2) صغير المجال

## Schemas versus instances : وتكتب في بعض الأحيان كلمة versus بالاختصار VS وتعنى الفرق

\* ال **Schemas Database** وهو وصف لقاعدة البيانات كالشكل والقيود التي تفرض على القاعدة .

\* ال **Schema Diagram** : وهو عرض لوظائف ال **Schemas** .

\* ال **Construct Diagram** : وهو يقوم بوصف الخصائص للعلاقات مثل (العمر ، الرقم ، الإسم ، القسم ، غيرها )

وتسمى الكينونات أو ال **Entity** .

\* ال **Database Instance** : وهي البيانات الفعلية التي تخزن في القاعدة عند لحظة زمنية معينة وتسمى (حالة القاعدة )

أو ال **database state** أو ال **occurrence** .

توضيح : لنفترض أنك تعمل على قاعدة بيانات معينة (مثلاً قاعدة بيانات الجوازات) فعندما تفتح القاعدة لإدخال جواز جديد وعند

البدء في الإدخال أو أثناءه أو بعده في هذه اللحظة تسمى القاعدة **Database Instance** .

## Database Schema Vs. Database State (الفرق)

\* ال **Database state** ترمز لمحتوى قواعد البيانات عند لحظة زمنية معينة .

\* ال **Initial Database State** : حالة القاعدة قبل إدخال البيانات (الحالة الابتدائية)

وهي حالة القاعدة عند لحظة تحميل البيانات.

\* ال **Valid State** : الحالة الصحيحة : وهي الحالة التي تطابق الهيكل والقيود للقاعدة .

ال **Distinction** :

الوصف للقاعدة لا يتغير سريعاً / يتغير حسب تحديث القاعدة (تكرار) **database state** تسمى : **intension**

وتسمى **extension**

## Three-Schema Architecture (معمارية)

تقسم إلى ثلاثة أجزاء وقد صممت لدعم خصائص ال **DBMS** مثل :

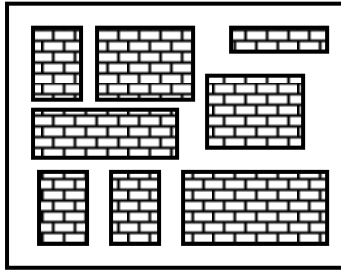
– يجب فيها فصل البرامج عن القاعدة (**programs**) – تدعم أكثر من وجهة نظر للبيانات (**suport**)

أنواع المعماريات : تم تعريفها في ثلاث مستويات هي :

(1) **الداخلي internal** : وهي لوصف الشكل الفيزيائي التخزيني للبيانات والوصول لها بأقصى طريقة وتستخدم القاعدة الفيزيائية

لأنها تتعامل مع البيانات .

الـ **Access paths** : لأن البيانات في الذاكرة تخزن كما في الرسم التالي (متباعدة) وعندما نريد إسترجاع بيانات معينة من



خلال إستعلام فإن أقصر وأسرع طريقة للوصول لها يسمى بـ **Access paths** .

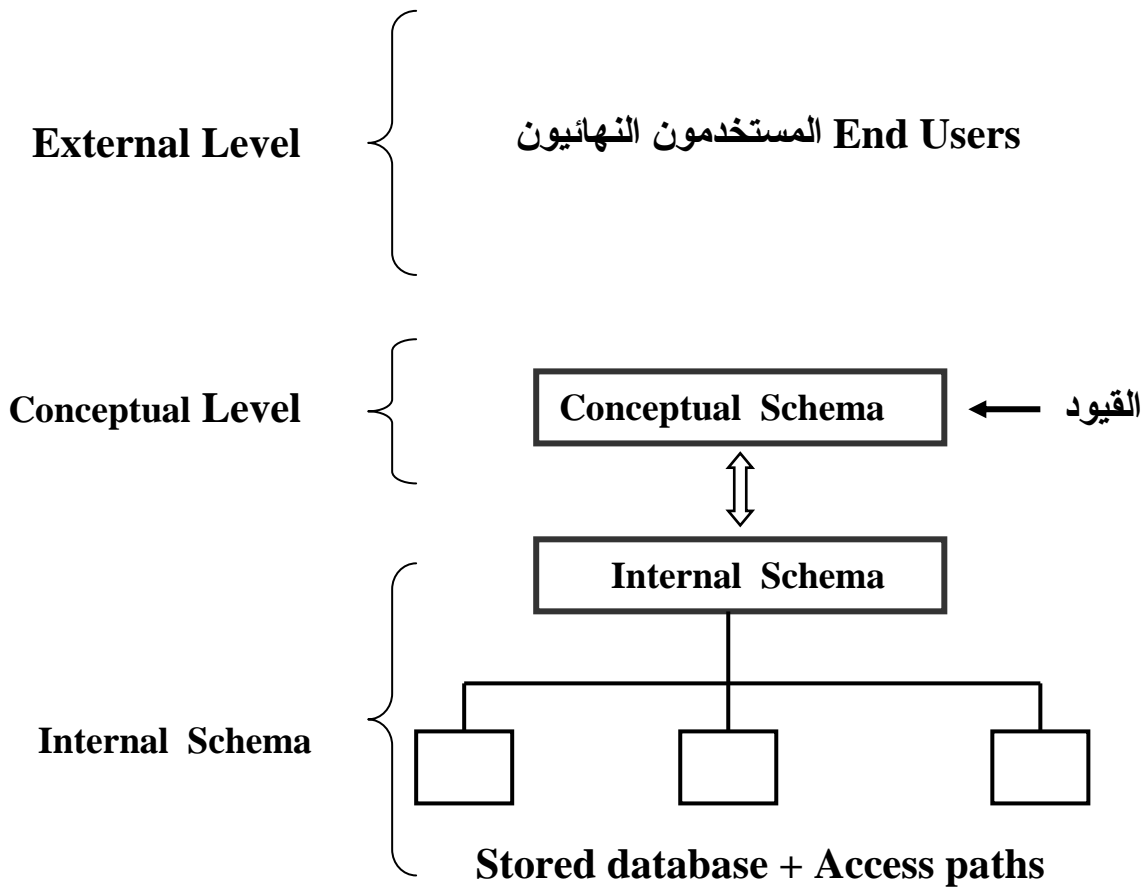
(2) الـ **Conceptual** : توجد في المستوى الإفتراضى وتقوم بوصف الشكل والقيود

لكل القاعدة على مجتمع من المستخدمين وتستخدم نموذج البيانات .

(3) الـ **External** : وهى التى تقوم بوصف المستخدمين الآخرين (المستخدمين) والمستخدمين

الإعتياديين وكل ما له علاقة بنموذج البيانات .

الشكل التالى يبين المعماريات الثلاثة السابقة : (يوجد هذا الشكل بالكتاب بتفاصيل أكثر)



الـ **Mappings** : التنقل او التحويل بين الطبقات : هو الذى يختص بالتحويل من مرحلة إلى أخرى للتبادل بين المطالب كتمثال

البرامج ترمز إلى طبقة الـ **External Level** ويتم تحويلها إلى mapped بواسطة القاعدة إلى **Internal** للتنفيذ .

إستقلالية البيانات : **data independence** :

**Logical data independence** : المقدرة على تغيير وصف القاعدة بدون الحاجة لتغيير الـ **External** والبرامج التطبيقية الخاصة .

**physical data independence** : المقدرة على تغيير الوصف الفيزيائى بدون تغيير الـ **Conceptual** مثل تغيير التراكيب الخاصة

بالبيانات المخزنة **structures** وتغيير الـ **Access paths** .

## لغات نظم إدارة قواعد البيانات DBMS Language :

(1) لغة الـ **DDL** : لغة تعريف البيانات **Data Definition Language** :

وهي لغة تستخدم بواسطة مدير قاعدة البيانات والمصممون وذلك لتحديد الوصف للقاعدة .

وفي كثير من لغات DBMS يتم إستخدامها :

\* لتحديد الـ **External** والـ **internal**

\* وفي بعضها تستخدم كل لغة لكل **Schema**

\* وفي بعضها تستخدم معاً للـ **Conceptual** والـ **External** والـ **internal**

(2) لغة الـ **DML** : **Data Manipulation Language** : وهي تستخدم لتحديد عمليات التحديث والإسترجاع .

وأوامر لغة **DML** : **DML Commands** وتسمى لغة البيانات الفرعية ويمكن تضمينها في بعض لغات البرمجة عامة

الأغراض مثل كوبول ، السي ، الأسمبلى . وبديلاً عن ذلك فإن بعض أوامر هذه اللغة تطبق بصورة مباشرة بواسطة لغة الإستعلامات

. **query language**

(3) لغة الـ **High Level or Non-procedural Languages** كمثال لها لغة الإستعلامات البنوية **SQL**

• وهي لغة تقوم بتحديد البيانات التي يمكن إسترجاعها وتسمى لغة الـ **declarative languages** .

(4) لغة الـ **Low Level or Procedural Languages** :

• وهي التي تقوم بتحديد كيفية إسترجاع البيانات بواسطة الحلقات .

## واجهات نظم إدارة قواعد البيانات DBMS Interfaces :

\* الواجهة القياسية : مثل لغة **SQL** .

\* واجهة البرمجة لتضمين الـ **DML** في بعض لغات البرمجة .

\* واجهة المستخدم : كمثال القوائم ، النماذج ، الرسومات ، ..... إلخ .

\* واجهات أخرى : واجهات التحديث ، الإدخال والإخراج ، الويب ، البنوك ، إلخ .

\* أنواع واجهات الـ **DBA** : إعطاء صلاحيات الوصول ، وضع معاملات النظام ، تغيير الـ **Schema** .

## Database System Utilities الأشياء المساعدة في القاعدة (المعينات في القاعدة)

هنالك بعض المعينات أو الوظائف للقاعدة هي :

(1) تحميل البيانات وتخزينها في الـ **DB** : **Loading data stored** : كمثال الملفات النصية **Text**

(2) النسخ الإحتياطي للقاعدة بصورة دورية : **Backing up the database periodically on tape**

وتسمى أيضاً **Back up** : وهذه العملية على حسب أهمية المعلومات (يومية ، إسبوعية ، شهرية ، غيرها )

أنواع النسخ الإحتياطي :

أ/ نسخ كامل لكل القاعدة مع تحديثها .

ب/ نسخ للتعديلات المحدثّة فقط داخل القاعدة كمثال حسابات البنوك وهو نوع معقد من النسخ .

(3) إعتبار أشكال الملفات للقاعدة : **Reorganizing database file structures**

(4) إنشاء التقارير عن القاعدة : **Report generation utilities**

(5) مراقبة كفاءة القاعدة : *Performance monitoring utilities* // الكفاءة تسمى (*Performance*)

وظائف أخرى : مراقبة المستخدم ، الترتيب ، الضغط ، إلخ

Other functions, such as *sorting, user monitoring, data compression, etc*

### أدوات أخرى : Other Tools

\* قاموس البيانات : *Data dictionary* ويسمى *repository* : ويستخدم لـ

- لتخزين ووصف الـ *Schema*
- وصف البرامج التطبيقية
- لإستخدام الصور القياسية
- لقرارات التصميم
- لوضع معلومات عن المستخدم

### أنواع قاموس البيانات DD

\* *Active data dictionary* : وهو نوع يمكن الوصول له عن طريق (البرامج *software* في الـ *DBMS*) *DBMSSW* أو عن طريق المستخدم أو مدير القاعدة *DBA* .

\* *Passive data dictionary* : ويتم الوصول له عن طريق مدير القاعدة *DBA* أو المستخدم .

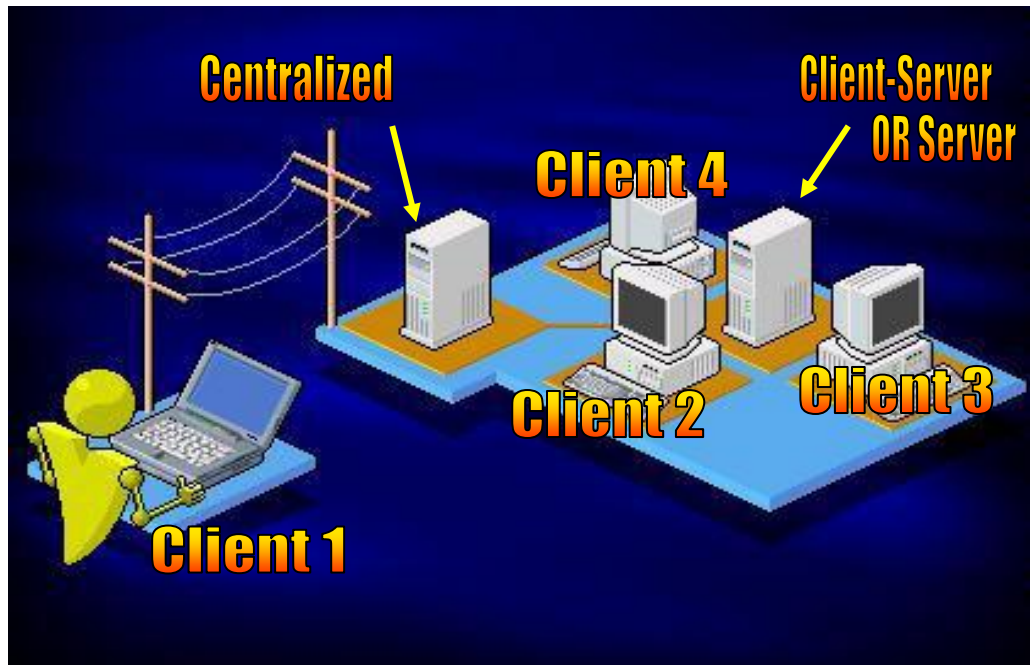
بيئات تطوير التطبيقات والـ *Case* : *Application Development Environments and CASE*

مثل للـ *Case* لغة أو أداة الـ *Power builder* و الـ *builder*

الفرق بين المعمارية المركزية ومعمارية العميل : *Centralized and Client-Server Architectures*

- معمارية المركزية *Centralized* تختص بتخزين الـ *DBMS* والـ *DB* في مكان واحد (مركزي) الأجهزة والبرامج ، إلخ

- معمارية *Client-Server* تعنى وجود جهازين فأكثر الأول يسمى *Server* وهو المستقبل وتوجد به القاعدة كمثال عندما نطلب الرصيد في الهاتف فإننا نتصل بالـ *Server* والجهاز الآخر يسمى *Client* وهو المرسل . الرسم التالي يوضح ذلك :



في الـ *Client* : الأجهزة دائماً ما تكون مرتبطة بالخدمة وتوفر واجهات معينة ونظام الحاسوب المستخدم (نظام التشغيل) يسمى *Client-Server* للوصول للبيانات الموجود في المخدم *Server* وقد يكون نوع الحاسوب إما شخصي أو عملي أو كبير أو محطة عمل وكل هذه الحواسيب تستخدم نظام الخادم . وتعمل كل الحواسيب بالشبكة السلكية أو اللاسلكية .

**نظام الـ DBMS Server** : هو نظام يوفر المعاملات التي يحتاج لها العميل Client مثل معاملات الإستعلامات DB-Q

ويسمى نظام الـ DBMS Server أحياناً بـ transaction أو الـ query Server .

**تصنيفات الـ Classification of DBMSs** : تقسم إعتياداً على عدة خصائص منها :

(1) الـ data model : وهو تصنيف قياسي أساسي ويستخدم :

\* الـ Traditional : وهو إما علائقي أو شبكي أو معماري Relational, Network, Hierarchical

وهي تصنيفات قديمة .

\* الـ Emerging : وهي تصنيفات حديثة مثل Object-oriented, Object-relational

(2) عدد المستخدمين :

\* مستخدم مفرد (وحيد) Single-user : وهي تستخدم من قبل شخص واحد .

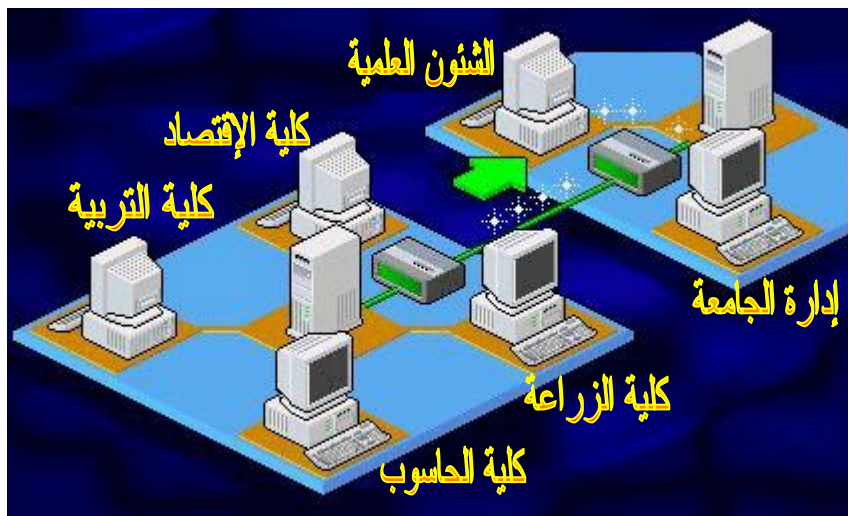
\* مستخدمين متعددين multi-user : وهي تستخدم أكثر من شخص وهو أغلب الأستخدام .

(3) المركزية Centralized : التي تستخدم حاسوب واحدة للقاعدة وقاعدة واحدة فقط. والموزعة distributed

تستخدم عدة حواسيب وعدة قواعد .

### تقسيمات الـ DBMS الموزعة

كمثال إدارة جامعة القصارف موزعة على الكليات المختلفة .



وتعتمد في هذا التوزيع على :

(1) التوزيع المتشابه : Homogeneous DDBMS : وتشابه فيه القاعدة في كل الأماكن .

(2) التوزيع المختلف : Heterogeneous DDBMS : وتختلف فيه القاعدة .

(3) التوزيع المتعدد : Federated or Multidatabase Systems : تتعد فيه القاعدة .

انتهى الفصل الثاني بحمد الله وتوفيقه ،،،





## الفصل الخامس :

### Relational Model Concepts مفاهيم النموذج العلائقي

Relational Model Constraints and Relational Database Schemas القيود على العلاقة ووصف قواعد البيانات

Update Operations and Dealing with Constraint Violations تحديث العمليات والتعامل مع القيود

### Relational Model Concepts : مفاهيم النموذج العلائقي :

- النموذج العلائقي للبيانات الذي يبني على مفهوم العلاقة . - العلاقة عبارة عن مفهوم رياضي يبني على أساس المجموعات
- المنهج العلائقي يعطي من إدارة البيانات الرسمية المعطاة التي توفر بواسطة النظرية العلائقية .
- النموذج طرح أولاً من قبل الدكتور E.F. Codd في الورقة العلمية الثالثة عام 1970م لمشاركة المعلومات الكبيرة في الإتصالات
- الورقة أعلاه ليست ثورة في حقل قاعدة البيانات .

### تعريفات غير رسمية INFORMAL DEFINITIONS

- العلاقة عبارة عن جدول . - يمكن اعتبار العلاقة مجموعة صفوف set of rows
- بطريقة بديلة أن العلاقة مجموعة من الأعمدة . - كل صف يمثل حقيقة تقابل كينونة في العالم الحقيقي مثل الطالب .
- كل صف يختلف عن الآخر ولكل صف قيمة لمجموعة من العناصر هذه القيمة تعرف الصف في الجدول بطريقة فريدة .
- في بعض الأحيان مجموعة أرقام تسلسلية تعرف الصف في الجدول .
- كل عمود له إسم (إسم العمود أو إسم الخاصية)

### تعريفات رسمية FORMAL DEFINITIONS

- يمكن تعريف العلاقة بأكثر من طريقة كمثال للزبون (إسم ، رقم ، عنوان ، تلفون ) والزبون يعتبر علاقة تعرف مجموعة من الخصائص وله مجال أو مجموعة ولكل خاصية منها قيم صحيحة . فمثلاً يمكن تعريف الخاصية Cust-id في الخانات الرقمية 6 خانات .
- هو مجموعة من القيم المرتبة . - كل قيمة هي مشتقة من مجال مناسب كمثال الإسم من سلة الحروف (حرفي) string
- كل صف من صفوف الزبون يجب أن ترجع له قيمة في الجدول - العلاقة بمعنى مجموعة من الجداول
- كل اعمدة الجدول تسمى خصائص العلاقة . - للمجال تعريف منطقي كمثال رقم تلفون (USA) 10 خانات رقمية
- للمجال صيغة تعريف أو نوع بيانات فمثلاً في أمريكا رقم التلفون يمثل بـ dddd وكل d تمثل خانة عشرية وكذلك التاريخ ويمتلك صيغ متعدد مثل : yyyy-mm-dd أو dd-mm-yyyy
- العلاقة تمثل الخاصية الكارتيزية للمجموعات . - كل مجموعة لها قيم من المجال تستخدم في الـ role

### خصائص العلاقات CHARACTERISTICS OF RELATIONS

- ترتيب الـ tuples في العلاقات . وهي مرتبة لكنها تظهر غير مرتبة .
- ترتيب الخصائص في العلاقة (R) مع القيم في كل جدول ، أي ترتيب القيم مثل إسم تكتب أمامها الإسم ، ورقم يكتب أمامها الرقم .
- القيم في الجدول : كل قيمة في الجدول غير قابلة للإنقسام والقيمة الخاصة null والتي تعني فارغ تستخدم لتمثيل القيم المجهولة أو الغير ملائمة في الجدول .
- الترقيم : Notation تشير إلى القيم المكونة في الجدول بنفس الطريقة .



## الفصل السابع :

### خوارزمية التخطيط العلائقي ER-to-Relational Mapping Algorithm

#### خطوات تحويل الخوارزمية إلى ER

الخطوة 1 : رسم خرائط أنواع الكينونات العادية

الخطوة 2 : رسم خرائط أنواع الكينونات الضعيفة

الخطوة 3 : رسم خرائط أنواع العلاقات الثنائية 1 : 1

الخطوة 4 : تعيين من 1 إلى متعدد

خطوة 5 : تعيين عمليات الاندماج الثنائي : أنواع العلاقة ن.

خطوة 6 : رسم خرائط متعددة القيم سمات.

خطوة 7 : رسم الخرائط من أنواع العلاقة ن آري.

\* الخطوة (1) أى كينونة منتظمة فى ER Schema تنشأ علاقة R تتضمن كل الخصائص البسيطة للعلاقة R .

– نختار أحد الخصائص بحيث يكون مفتاح أساسى للعلاقة .

– إذا كان المفتاح المختار مركب من مجموعة الخواص البسيطة التى تشكلها ستشكل المفتاح الأساسى سوية .

\* الخطوة (2) أى weak entity فى ER تابعة لـ owner فى E تنشئ علاقة R تتضمن كل الخصائص البسيطة أو مكونات

بسيطة من الخواص الدمجة بخواص للعلاقة R بالإضافة إلى التى تتضمن خاصية المفتاح الأجنبى فى العلاقة R أى المفتاح الأساسى

حق الموظف فى ER .

– إن المفتاح الأساسى للعلاقة R هو تجمع من المفتاح الأساسى حق الموظف الذى يسمى المفتاح الأجنبى والمفتاح الآخر حق weak entity

\*الخطوة (3) فى العلاقة واحد لواحد تعرف العلاقة S و T الذى تطابق entity type للعلاقة المفضلة R هنالك ثلاثة طرق

:

1– الطريقة الأولى : نختار المفاتيح الأساسى لك entity ونضم فى entity الأخرى بحيث يكون مفتاح أجنبى . ويفضل أن يكون

المفتاح الأجنبى فى العلاقة العامة total .

2– إذا كان كلاهما عام total يتم دمجهما فى علاقة واحدة تسمى بإسم الإثنين معاً .

3– دمج العلاقة بالمفتاحين الأساسيين لك entity الأول والثانى ونمثلهما فى entity type .

\*الخطوة(4) : أى كينونة منتظمة regular entity تعرف بالعلاقة S تمثل تعريف entity بعلاقة n .

وتتضمن العلاقة T مفتاح أجنبى بالنسبة للمفتاح الأساسى لـ S .

\* الخطوة(5) : أى regular entity علاقة من m إلى n تنشئ علاقة حيث R تتضمن المفتاح الأساسى للكينونة الأولى

والمفتاح الأساسى للكينونة الثانية كمفاتيح أجنبية فى العلاقة الجديدة ثم ندمجها مع بعض كمفتاح أساسى بالإضافة إلى تضمين أى

خاصية تتعلق بالعلاقة فى الجدول مع تضمين أسماء المفاتيح الأساسية فى العلاقة الجديدة .

\* الخطوة (6) : لكل خاصية متعددة القيم تشكل علاقة جديدة والعلاقة الجديدة تتضمن الخاصية المتعددة القيم بالإضافة للمفتاح

الأساسى بالنسبة للخاصية كمفتاح أجنبى فى العلاقة الجديدة . المفتاح الأساسى للعلاقة هو نتيجة من المفتاح الأجنبى والخاصية .

\* الخطو (7) : العلاقة التى تشترك فيها أكثر من إثنين التحويل ينتج علاقة جديدة ويتم أخذ المفتاح الأساسى ويدمج فى العلاقة الجديدة

والمفتاح الأساسى هو تجمع المفاتيح الأجنبيين وكذلك إذا كانت ثلاثة يكون المفتاح الأساسى هو تجمع ثلاثة مفاتيح وتضمن الخصائص بالعلاقة

أساسيات نظم قواعد البيانات الفصل الرابع (علوم + تقانة) / ترجمة عثمان صدى /// الأجزاء (5،2،3،7) صفحة رقم (10)

شرح مثال (2) :

تنشأ علاقة تعتمد على أنواع العلاقات المعتمدة تتضمن المفتاح الأساسي للموظف (SSN) في العلاقة ويسمى في العلاقة R مفتاح أجنبي في الإيعتماد (dependent) يسمى (ESSS) المفتاح الأساسي للـ dependent هو تجمع من المفتاح الأجنبي والمفتاح الجزئي بالنسبة للـ dependent (ESSN, dependent name) وذلك لأن الـ dependent name مفتاح جزئي في الـ dependent .

خطوات عمل الـ Mapping للخوارزميات : ER-to-Relational Mapping Algorithm

الخطوة الأولى : إذا أردنا تحويل الـ regular entity أى أن كل واحدة منها تتضمن كل خصائص الكينونة entity للعلاقة R ثم نختار واحد من الخصائص بحيث يكون مفتاح أساسي للعلاقة R إذا كان المفتاح الأساسي هو عبارة عن خصائص مركبة فإن المفتاح الأساسي هو مجموع هذه الخصائص .

الخطوة الثانية : إذا كان لدينا weak entity مع woner ننشئ علاقة تتضمن كل الخصائص البسيطة أو المركبة بالإضافة إلى تضمين المفتاح الأساسي بالنسبة للموظف (الكافل) كمفتاح أجنبي بالنسبة للمكفول مع تغيير تسمية المفتاح الأساسي مثل essn والمفتاح الأساسي لهذه العلاقة هو جميع المفتاح الأجنبي مع المفتاح الجزئي بالنسبة للعلاقة .

الخطوة الثالثة :

العلاقة واحد لواحد توجد في ثلاثة احتمالات :

- (1) foreign key approach : وهو أن نضع المفتاح الأساسي للعلاقة الخاصة في العلاقة العامة كمفتاح أجنبي .
- (2) merged relation option : دمج العلاقات وهو إذا كانت كلتا العلاقات total ندمجهم في علاقة واحدة نسميها بإسمها في العلاقاتين كمثال employee , department .

(4) cross-reference or relationship relation : وهو دمج المفتاح الأساسي option للعلاقة الأولى

والمفتاح الأساسي للعلاقة الثانية في علاقة واحدة .

( هذا ما إستطعته من ترجمه لهذا الفصل ،،،،،،،،،،، لكم العتبي )

(الحمد لله الذى هدانا لهذا وما كنا لنهتدى لولا أن هدانا الله)

(اللهم لا تؤاخذنا إن نسينا أو أخطأنا)

إن أصبت فمن الله وإن أخطأت فمنى والشيطان

مع الشكر  
رحمة الله

تذكرة :

علمتني الحياة أن أجمل قلبي مدينة بيوئها المحبة وطرقها التسامح والعفو وأن أعطي ولا أنتظر الرد على العطاء ..

وأن أصدق مع نفسي قبل أن أطلب من أحد أن يفهمني .. وعلمتني أن لا أندم على شئ وأن أجعل الأمل مصباحاً يرافقني كل مكان فأرجو أن أتعلم منها أكثر وأكثر

أخيراً : الضمير لا ينعنا من فعل الأشياء السيئة .. ولكنه يزعج ما تبقى فينا من أخلاق .. ويعكر المتعة!

وإذا افتقرت إلى الذخائر لم تجد .. ذخراً يكون كصالح الأعمال

(اللهم صلى على سيدنا محمد الأمين بقدر ما خط الفلمح في الورق وبقدر ما أشرق نوراً أو برق)