الفصل الثاني
تصميم قاعدة البيانات

1. أهمية تصميم قاعدة البيانات

إن عملية بناء قاعدة بيانات جيدة ليست بالعملية البسيطة، إذ لا بد من بذل مجهود كبير للحصول على قاعدة بيانات جيدة. فالتصميم الجيد لقاعدة البيانات يسهل عملية استخدام وإدارة تلك القاعدة. أما التصميم السيء فسيؤدي إلى تكرار البيانات (وجود البيانات نفسها في أكثر من مكان)، وبالتالي يصعب عملية الحفاظ على توافقية البيانات ويؤدي غالباً تكرار البيانات إلى نتائج غير صحيحة عند طلب المعلومات، وهذا بدوره يؤدي إلى أن القرارات ستكون خاطئة وبالتالي التخطيط المستقبلي سيكون خاطئاً لاعتماده على بيانات مخزنة بطريقة غير صحيحة في قاعدة بيانات صُممت بطريقة سيئة لتعطي معلومات خاطئة.

2. دورة حياة نظام قاعدة البيانات

2.1. الدراسة المبدئية للنظام القائم

تشمل الدراسة المبدئية للنظام القائم المراحل الآتية:

1. تحليل الوضع الحالي ومعرفة طبيعة العمل والإجراءات المستخدمة وقواعد العمل.
2. تحديد القيود والمشاكل التي تواجه النظام القائم بما في ذلك القيود المادية (العامل البشري – الأنظمة والقوانين – التمويل).
3. تحديد المتطلبات الواجب تحقيقها والمزايا المطلوب إبرازها في النظام المطلوب.

2.2. تصميم قاعدة البيانات

تعدّ هذه المرحلة من أهم المراحل في دورة حياة النظام فلا بد من بذل جهد كبير لتصميم النظام. وسندرس في هذا الفصل المبادئ الأساسية لتصميم قاعدة البيانات من خلال مخطط الكينونات العلائقية.

دورة حياة نظام قاعدة البيانات :

1. الدراسة المبدئية
2. دراسة الجدوى
3. تحليل نظام
4. تصميم نظام (يتضمن تصميم قاعدة بيانات)
5. كتابة Code
6. اختبار
7. مرحلة تنفيذ (إطلاق) 8. صيانة

Scanned by CamScanner

2.3. هيكل (مخطط) قاعدة البيانات (Database Schema)

هو مخطط يوصف قاعدة البيانات حيث يشمل وصف لبنية البيانات فيها، وشروط تلك البيانات أو القيود التي يجب أن تُطبق عليها. يمكن عدّ مخطط البيانات دليلاً أو قاموساً يصف تلك البيانات، بغض النظر عن البيانات نفسها.

إذا ما تم عمل التصميم المنطقي لقواعد البيانات فإنه من السهل وضع التصميم الفيزيائي لقواعد البيانات تلك، وبالتالي فإن الخطوات الأساسية التي ممكن اتباعها لتصميم قاعدة بيانات هي:

1. جمع البيانات ومتطلبات المستخدمين (Requirement Gathering)
2. تحليل البيانات (Data Analysis)
3. إنشاء التصميم المفاهيمي لقواعد البيانات (Conceptual Model)
4. إنشاء التصميم المنطقي لقواعد البيانات (Logical Model)
5. إنشاء التصميم الفيزيائي لقواعد البيانات (Physical Model)

إن عملية جمع البيانات وتحديد المتطلبات للمستخدمين هي خطوة مهمة جداً وذلك لأنها تحدد الاحتياجات المختلفة للمستخدمين من قاعدة البيانات وتعطي وصفاً للنظام القائم داخل البيئة التي تتم عليها الدراسة. هنا لا بد من ذكر أن هذه العملية يجب أن تكون عملية تفصيلية وشاملة لأن أي نقص في المعلومات يؤدي إلى تصميم خاطئ أو خلل في التصميم وهذا يؤدي إلى مشاكل في قواعد البيانات. بناءً على هذه المعلومات

Scanned by CamScanner

يتم بناء مخطط قواعد البيانات الذي يمثل المعلومات التي تم تجميعها في المرحلة السابقة وهذه المعلومات هي وصف للبيانات المختلفة وعلاقتها مع بعضها ببعض وكذلك تحديد القيود والمحددات المفروضة على هذه البيانات. إن هذه المرحلة تسهل التفاعل بين المستخدمين وبين المصمم حيث يتم عمل مخططات سهلة الفهم وتعكس الواقع قيد الدراسة.

بعد هذه المرحلة يتم التحول إلى تصميم منطقي لقواعد البيانات باستخدام إحدى نظم إدارة قواعد البيانات.

3. نماذج قاعدة البيانات : النموذج العلائقي The Relational Model

إنك تشارك العديد من العلاقات مع أفراد عائلتك .فعلى سبيل المثال، في العائلة الواحدة سيكون لكل منا شكل صلة وعلاقة مع الأم. فأنت ووالدتك ذات صلة (علاقة)، فلديك أم واحدة فقط، لكنها قد تكون لها عدة أطفال. أنت وإخوتك ذات صلة قد يكون لديك العديد من الإخوة والأخوات، وبطبيعة الحال، سيكون لديهم الكثير من الإخوة والأخوات أيضاً.
ومن هنا تفهم العلاقة على أنها وصف دقيق للحالة التشاركية مع الكائن الآخر.

يعدّ النموذج العلائقي لقواعد البيانات The Relational Model بأنه الدعامة الأساسية لقواعد البيانات المعاصرة. في هذا الفصل سنعرض الأفكار الأساسية لهذا النموذج، والتي تركز على تنظيم البيانات و إمكانية استرجاعها.

يتكون النموذج العلائقي من :

- مجموعة من هياكل البيانات تُعرف بالجداول .
- مجموعة من الأساليب لبناء الجداول بدءاً من المخطط الأولي للجداول و نشير هنا إلى عمليات الجبر العلائقية.
- مجموعة من القيود المفروضة على البيانات الواردة في الجداول.

تعدّ قواعد البيانات العلائقية من أهم قواعد البيانات وتتصف بأنها قاعدة بيانات يستقبلها المستخدمون على هيئة جداول وليس شيئاً آخر سوى الجداول.

Scanned by CamScanner

تذكرة: (مفهوم العلاقة رياضياً)

- العلاقة هي عبارة عن مجموعة جزئية من الجداء الديكارتي بين مجموعتين (أو أكثر)
- فمثلاً إذا كانت لدينا المجموعتان:

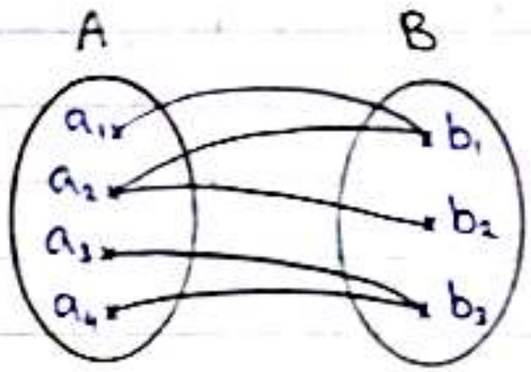
$$A = \{a_1, a_2, a_3, a_4\}$$
$$B = \{b_1, b_2, b_3\}$$

وكانت العلاقة:

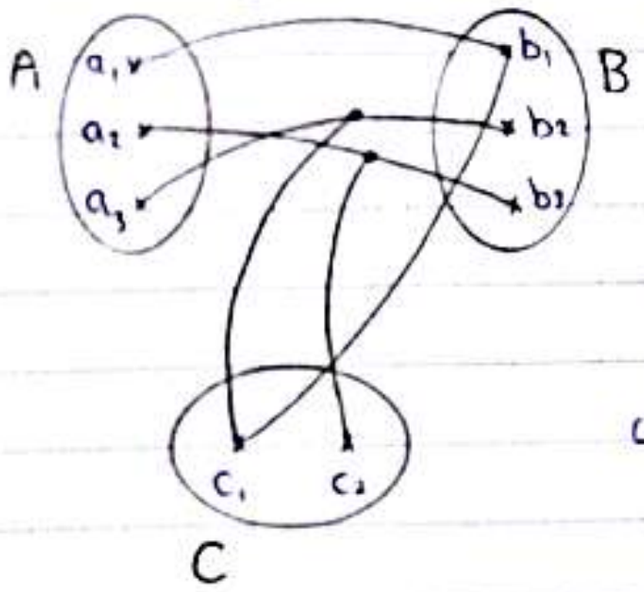
$$\mathcal{R} = \{(a_1,b_1), (a_2,b_1), (a_2,b_2), (a_3,b_3), (a_4,b_3)\}$$

فيمكن تمثيل العلاقة $\mathcal{R}$ بيانياً أو على شكل جدول كما يلي:

A	B
a_1	b_1
a_2	b_1
a_2	b_2
a_3	b_3
a_4	b_3



- في الحقيقة ليس من الضرورة أن تكون العلاقة ثنائية، بل ثلاثية ورباعية و...



فمثلاً: إذا كان لدينا مثل العلاقة الثلاثية الموضحة بيانياً، فإن تمثيلها بهذا الشكل يعتبر معقداً جداً، وكلما زاد عدد عناصر العلاقة أو عدد مجموعاتها زاد تعقيد الشكل البياني لها.
لذا نلجأ لتمثيل العلاقات على شكل جداول مما يخفف من التعقيد.

Scanned by CamScanner

/ /

- تتكون قاعدة البيانات العلائقية من مجموعة من العلاقات Relations
حيث كل علاقة هي عبارة عن مشاهدة لمجموعة بيانات أو أشياء من الواقع
ومن هنا جاءت تسمية (قواعد البيانات العلائقية)
ويتم تمثيل العلاقة في قاعدة البيانات بجدول.

الجدول: هو بنية ثنائية البعد تتألف من أعمدة وأسطر.
لكل عمود اسم وهيد ونمط بيانات محدّد، ويمثل العمود صفة لكيان معين
يُدعى الواصفة Attribute.
أما الأسطر في الجدول فندعوها السجلات Tuples - record
فإذا كان الجدول يمثل بيانات موظفي شركة ما، فكل عمود يمثل صفة للموظف
(اسمه - رقمه - تاريخ توظيفه ....) وكل سطر يمثل موظف محدد.

مثال:

Stud N°	Stud Name	Stud Date
13	Tolo	1992
11	JoJo	1995
32	Nana	1993

- عدد الأسطر في الجدول يمثل درجة العلاقة Cardinality
وعدد الأعمدة يمثل درجة العلاقة degree

- بشكل عام يمكن تمثيل العلاقة مثلاً وفق الترميز التالي: R(A,B,C,D)
حيث R اسم العلاقة، A,B,C,D هي الواصفات.
ويُدعى هذا الترميز بـ 'schema' (مخطط) ويكون ثابتاً لا يتغير مع الزمن.
والـ schema للعلاقة في الجدول السابق هي:
Student (Stud N°, Stud Name, Stud Date)
أما Student (34, Lolo, 1994) هو سجل محدّد ويُدعى instance
وهو متغير مع الزمن.

Scanned by CamScanner

/ /

خواص الجداول في قاعدة البيانات العلائقية:

١- لكل جدول اسم وحيد .
٢- القيم المدرجة تحت عمود واحد لها نفس النوع
٣- كل عمود له اسم يختلف عن بقية الأعمدة في نفس الجدول .
٤- كل سطر في الجدول يختلف عن بقية الأسطر .
٥- ترتيب الصفوف ، وكذلك الأعمدة في الجدول ليس ذو أهمية .
٦- يوجد لكل جدول حقل يسمى المفتاح الأساسي Primary key
يمكننا من الوصول لسجل معين في هذا الجدول .
٧- كل القيم مُعبَّر عنها مباشرة ، أي أنها قيم صريحة وليست متغيرات .

ملاحظة:
- إذا كانت الواصفة A هي المفتاح الأساسي فنكتب الـ schema بالشكل :
R (A , B , C , D)
- إذا كانت الواصفتان A,B معاً هما المفتاح الأساسي نكتب :
R (A,B , C , D)

Scanned by CamScanner

3.2. أهم مفاهيم قواعد البيانات العلائقية

3.2.1. المفتاح الأساسي للجدول: Primary Key

الجدول يتألف من صفوف (أو سجلات Records)، و في قواعد البيانات العلائقية يجب ألا يتكرر الصف كاملاً بل يجب أن توجد قيمة فريدة تميز كل صف من الصف الآخر. هذا ما نسميه بالمفتاح الأساسي للجدول. كمثال على ذلك في جدول المستعيرين يمكن أن يكون لدينا أشخاص لهما الإسم نفسه وتاريخ الميلاد نفسه وكل البيانات عنهما متماثلة لذلك نميز بينها عن طريق رقم الهويه مثلاً ولكن هذا التمييز صعب بعض الشيء وأحياناً غير ممكن. لذلك سهلت لنا قواعد البيانات ميزة الترقيم التلقائي بحيث عندما نضيف سجلاً جديداً فإنه يأخذ رقماً خاصاً به تلقائياً ودون أن ندرج له رقماً في عمود المفتاح الأساسي و بالتالي نميز بين كل صف في الجدول عن طريق رقمه الذي يميزه.

3.2.2. المفتاح الغريب(الأجنبي) Foreign Key

في قواعد البيانات هناك مصطلح اسمه التكامل المرجعي Referential Integrity أي إنه لو فرضنا أنه يُوجد لدينا جدولان مثل Addresses، Employees يتم الربط بينهما عن طريق وضع عمود في أحدهما، هذا العمود له اسم عمود المفتاح الأساسي نفسه في الجدول الآخر، وله نوع البيانات نفسها Data Type والصفات نفسها Attributes. هذا العمود نسميه (مفتاح غريب) بحيث يصبح هذا العمود هو صلة الوصل بين الجدول الأول والجدول الثاني. هذه العملية نسميها بالتكامل المرجعي.

إذاً كتعريف نهائي للتكامل الرجعي: التكامل المرجعي هو علاقة بين جدولين يحوي الجدول الأول على مفتاح أساسي والجدول الثاني يرتبط مع الجدول الأول بمفتاح غريب له اسم المفتاح الأساسي نفسه وبياناته نفسها وصفاته نفسها. أما المفتاح الغريب فهو عمود في جدول يشير إلى عمود في جدول آخر هو المفتاح الأساسي للجدول الأخير و يكون لكل من العمودين الأساسي والغريب الاسم نفسه ونوع البيانات المدخلة فيهما والصفات نفسها. فعند إضافة بند إلى الجدول ذي المفتاح الغريب وليس موجوداً

Scanned by CamScanner

في الجدول ذي المفتاح الأساسي فسيقوم ملقم قواعد البيانات بتنبيهي إلى أنني أدرج شيئاً غير موجود في الجدول الأساسي.

من الأشياء المفيدة التي يقدمها لنا التكامل المرجعي هي عملية الحذف المتسلسل Deletes Cascading. ففرضاً أن لديك موظفاً مسجلاً في الجدول Employees و لدى هذا الموظف ثلاثة عناوين مسجلة في الجدول Addresses، ثم قدم هذا الموظف طلب استقالة ووافقت أنت عليه عندها ستقوم بحذف سجله من قاعدة البيانات الخاصة بشركتك. تخيل مدى الصعوبة والإرباك الذي ستتعرض له عند كل عملية حذف و ولاسيما مع الزبائن أو مع الأشخاص الذين يسجلون على بريد إلكتروني مجاني ثم لا يعودوا إلى البريد والكثير من العمليات المشابهة. وهنا تبرز أهمية الحذف المتسلسل فعندما أطلب من ملقم قواعد البيانات أن يحذف سجل هذا الموظف من الجدول Employees فإنه سيقوم آلياً بحذف جميع السجلات المقابلة في الجدول الذي يحتوي على مفتاح غريب.

Scanned by CamScanner