

Software Department

Lecture 8

.....
Redundancy exists when the same data are stored unnecessarily at different places. **Data anomaly** develops when all of the required changes in the redundant data are not made successfully.

يوجد التكرار عندما يتم تخزين نفس البيانات دون داع في أماكن مختلفة.
يتطور الانحراف في البيانات عندما لا يتم إجراء جميع التغييرات المطلوبة في البيانات الزائدة عن الحاجة بنجاح.

NORMALIZATION تطبيع

It is a process for evaluating and correcting table structures to minimize data redundancies, thereby reducing the likelihood of data anomalies.
إنها عملية لتقييم وتصحيح هياكل الجدول لتقليل التكرار في البيانات ، وبالتالي تقليل احتمالية حدوث حالات شاذة في البيانات.

Normalization works through a series of stages called normal forms. The first three stages are described as first normal form (1NF), second normal form (2NF) and third normal form (3NF).

يعمل التطبيع من خلال سلسلة من المراحل تسمى النماذج العادية. توصف المراحل الثلاث الأولى بالشكل العادي الأول (1NF) ، والشكل العادي الثاني (2NF) والشكل العادي الثالث (3NF).

From a structural point of view, 2NF is better than 1NF and 3NF is better than 2NF.

من وجهة نظر هيكلية ، 2NF أفضل من 1NF و 3NF أفضل من 2NF.

Denormalization عدم التطابق

Produces a lower normal form, that is a 3NF will be converted to a 2NF through denormalization

A successful design must also consider end-user demand for fast performance. Therefore you will occasionally be expected to denormalize some portions of database design in order to meet performance requirements

ينتج نموذجًا عاديًا أقل ، أي 3NF سيتم تحويله إلى 2NF من خلال عدم التطابق

يجب أن يأخذ التصميم الناجح في الاعتبار أيضًا طلب المستخدم النهائي للأداء السريع. لذلك ، يُتوقع منك أحيانًا إلغاء تسوية بعض أجزاء تصميم قاعدة البيانات من أجل تلبية متطلبات الأداء

THE NEED FOR NORMALIZATION الحاجة إلى التطبيع

Software Department

Lecture 8

.....

In following example

PROJ_NUM	PROJ_NAME	EMP_NUM	EMP_NAME	JOB_CLASS	CHG_HOUR	HOURS
15	Evergreen	103	June E. Arbough	Elect. Engineer	84.50	23.8
		101	John G. News	Database Designer	105.00	19.4
		105	Alice K. Johnson *	Database Designer	105.00	35.7
		106	William Smithfield	Programmer	35.75	12.6
		102	David H. Senior	Systems Analyst	96.75	23.8
18	Amber Wave	114	Annelise Jones	Applications Designer	48.10	24.6
		118	James J. Frommer	General Support	18.36	45.3
		104	Anne K. Ramoras *	Systems Analyst	96.75	32.4
		112	Darlene M. Smithson	DSS Analyst	45.95	44.0
22	Rolling Tide	105	Alice K. Johnson	Database Designer	105.00	64.7
		104	Anne K. Ramoras	Systems Analyst	96.75	48.4
		113	Delbert K. Joenbrood *	Applications Designer	48.10	23.6
		111	Geoff B. Wabash	Clerical Support	26.87	22.0
		106	William Smithfield	Programmer	35.75	12.8
25	Starflight	107	Maria D. Alonzo	Programmer	35.75	24.6
		115	Travis B. Bawangi	Systems Analyst	96.75	45.8
		101	John G. News *	Database Designer	105.00	56.3
		114	Annelise Jones	Applications Designer	48.10	33.1
		108	Ralph B. Washington	Systems Analyst	96.75	23.6
		118	James J. Frommer	General Support	18.36	30.5
		112	Darlene M. Smithson	DSS Analyst	45.95	41.4

We see in that example the structure of data set does not conform to the requirements of table nor does it handle data very well.

Consider the following deficiencies:

نرى في هذا المثال أن بنية مجموعة البيانات لا تتوافق مع متطلبات الجدول ولا تتعامل مع البيانات جيداً. ضع في اعتبارك أوجه القصور التالية:

1. The project number (PROJ_NUM) is apparently intended to be primary key or at least a part of a PK, but it contains nulls.
2. The table entries invite data inconsistencies. For example the JOB_CLASS value "Elect. Engineer" might be entered as "Elect. Eng."
3. The table displays data redundancies. Those data redundancies yield the following anomalies:

1. يبدو أن المقصود برقم المشروع (PROJ_NUM) أن يكون مفتاحاً أساسياً أو على الأقل جزءاً من PK ، ولكنه يحتوي على قيم خالية.
2. تدعو إدخالات الجدول إلى تناقضات البيانات. على سبيل المثال ، يمكن إدخال قيمة JOB_CLASS "Elect. Engineer" على أنها "Elect. Eng."
3. يعرض الجدول تكرار البيانات. ينتج عن هذا التكرار في البيانات الحالات الشاذة التالية:

Software Department

Lecture 8

-
- Update anomalies. Modifying the JOB_CLASS for employee number 103 requires (potentially) many alterations, one for each EMP_NUM=103.
 - Insertion anomalies. Just to complete a row definition, a new employee must be assigned to a project. If the employee is not assigned, a phantom project must be created to complete the employee data entry
 - Deletion anomalies. Suppose that only one employee is associated with a given project, if that employee leaves the company and the employee data are deleted, the project information will also be deleted. To prevent the loss of the project information, a fictitious employee must be created just to save the project information.

أ. تحديث الحالات الشاذة. يتطلب تعديل JOB_CLASS للموظف رقم 103 (يحتمل) العديد من التعديلات ، واحد لكل EMP_NUM = 103.

ب. تشوهات الإدراج. فقط لإكمال تعريف الصف ، يجب تعيين موظف جديد للمشروع. إذا لم يتم تعيين الموظف ، فيجب إنشاء مشروع وهمي لإكمال إدخال بيانات الموظف

ج. تشوهات الحذف. لنفترض أن موظفًا واحدًا فقط مرتبط بمشروع معين ، إذا غادر هذا الموظف الشركة وتم حذف بيانات الموظف ، فسيتم أيضًا حذف معلومات المشروع. لمنع فقدان معلومات المشروع ، يجب إنشاء موظف وهمي فقط من أجل حفظ معلومات المشروع.

عملية التطبيع THE NORMALIZATION PROCESS

We will learn how to use normalization to produce a set of normalized tables to store the data that will be used to generate the required information. The **objective of normalization** is to ensure that each table conforms to the concept of well-formed relations, that is, tables that have the following characteristics:

سوف نتعلم كيفية استخدام التسوية لإنتاج مجموعة من الجداول الموحدة لتخزين البيانات التي سيتم استخدامها لإنشاء المعلومات المطلوبة. الهدف من التطبيع هو التأكد من أن كل جدول يتوافق مع مفهوم العلاقات جيدة التكوين ، أي الجداول التي لها الخصائص التالية:

- Each table represents a single subject. For example, a course Table will contain only data that directly pertains to courses. Similarly, a student table will contain only student data.
- No data item will be unnecessarily stored in more than one table (in short, tables have minimum controlled redundancy). The reason for this requirement is to ensure that the data are update in only one place.
- All nonprime attributes in a table are dependent on the primary key. The reason for this requirement is to ensure that the data are uniquely identifiable by a primary key value.
- Each table is void of insertion, update or deletion anomalies. This is to ensure the integrity and consistency of the data.

- يمثل كل جدول موضوعا واحدا. على سبيل المثال ، سوف يحتوي جدول الدورة التدريبية على البيانات التي تتعلق مباشرة بالدورات التدريبية فقط. وبالمثل ، سيحتوي جدول الطالب على بيانات الطالب فقط.

- لن يتم تخزين أي عنصر بيانات دون داع في أكثر من جدول واحد (باختصار ، تحتوي الجداول على حد أدنى من التكرار الخاضع للرقابة) والسبب في هذا المطلب هو ضمان تحديث البيانات في مكان واحد فقط.

Software Department

Lecture 8

-
- تعتمد كافة السمات nonprime في الجدول على المفتاح الأساسي. سبب هذا المطلب هو التأكد من أن البيانات يمكن التعرف عليها بشكل فريد من خلال قيمة المفتاح الأساسي.
 - يخلو كل جدول من إدخال أو تحديث أو حذف الشذوذ. هذا لضمان سلامة واتساق البيانات.

التحويل إلى النموذج العادي الأول (1NF) Conversion to FIRST NORMAL FORM

الخطوة 1: استبعاد المجموعات المتكررة STEP 1: Eliminate the Repeating Groups

Start by presenting the data in tabular format, where each sale has a single value and there are no repeating groups. A **repeating group** derives its name from the fact that a group of multiple entries of the same type can exist for any single key attributes occurrence. To eliminate the repeating groups, eliminate the nulls by making sure that each repeating group attribute contains an appropriate data value.

ابداً بتقديم البيانات بتنسيق جدولي ، حيث يكون لكل عملية بيع قيمة و لا توجد مجموعات متكررة. تستمد المجموعة المكررة اسمها من حقيقة أن المجموعة من عدة إدخالات من نفس النوع يمكن أن توجد لأي سمات مفتاح واحد القضاء على المجموعات المكررة ، وإزالة الأصفار بالتأكد من أن كل مجموعة مكررة السمة تحتوي على قيمة بيانات مناسبة.

PROJ_NUM	PROJ_NAME	EMP_NUM	EMP_NAME	JOB_CLASS	CHG_HOUR	HOURS
15	Evergreen	103	June E. Arbough	Elect. Engineer	84.50	23.8
15	Evergreen	101	John G. News	Database Designer	105.00	19.4
15	Evergreen	105	Alice K. Johnson *	Database Designer	105.00	35.7
15	Evergreen	106	William Smithfield	Programmer	35.75	12.6
15	Evergreen	102	David H. Senior	Systems Analyst	96.75	23.8
18	Amber Wave	114	Annelise Jones	Applications Designer	48.10	24.6
18	Amber Wave	118	James J. Frommer	General Support	18.36	45.3
18	Amber Wave	104	Anne K. Ramoras *	Systems Analyst	96.75	32.4
18	Amber Wave	112	Darlene M. Smithson	DSS Analyst	45.95	44.0
22	Rolling Tide	105	Alice K. Johnson	Database Designer	105.00	64.7
22	Rolling Tide	104	Anne K. Ramoras	Systems Analyst	96.75	48.4
22	Rolling Tide	113	Delbert K. Joenbrood *	Applications Designer	48.10	23.6
22	Rolling Tide	111	Geoff B. Wabash	Clerical Support	26.87	22.0
22	Rolling Tide	106	William Smithfield	Programmer	35.75	12.8
25	Starflight	107	Maria D. Alonzo	Programmer	35.75	24.6
25	Starflight	115	Travis B. Bawangi	Systems Analyst	96.75	45.8
25	Starflight	101	John G. News *	Database Designer	105.00	56.3
25	Starflight	114	Annelise Jones	Applications Designer	48.10	33.1
25	Starflight	108	Ralph B. Washington	Systems Analyst	96.75	23.6
25	Starflight	118	James J. Frommer	General Support	18.36	30.5
25	Starflight	112	Darlene M. Smithson	DSS Analyst	45.95	41.4

الخطوة 2: تحديد المفتاح الأساسي Step 2: Identify the primary key

Even a causal observer will not that PROJ-NUM is not an adequate primary key because the project number does not uniquely identify all of the remaining entity (row) attributes. To maintain a proper primary key that will uniquely identify any attribute value, the new key must be composed of a combination of a PROJ_NUM and EMP_NUM

حتى المراقب السببي لن يقول أن PROJ-NUM ليس مفتاحاً أساسياً مناسباً لأن رقم المشروع لا يحدد بشكل فريد جميع سمات الكيان (الصف) المتبقية. للحفاظ على مفتاح أساسي مناسب يحدد بشكل فريد أي قيمة سمة ، يجب أن يكون المفتاح الجديد سماً من مجموعة من PROJ_NUM و EMP_NUM

الخطوة 3: تحديد كل التبعية Step 3: Identify All Dependencies

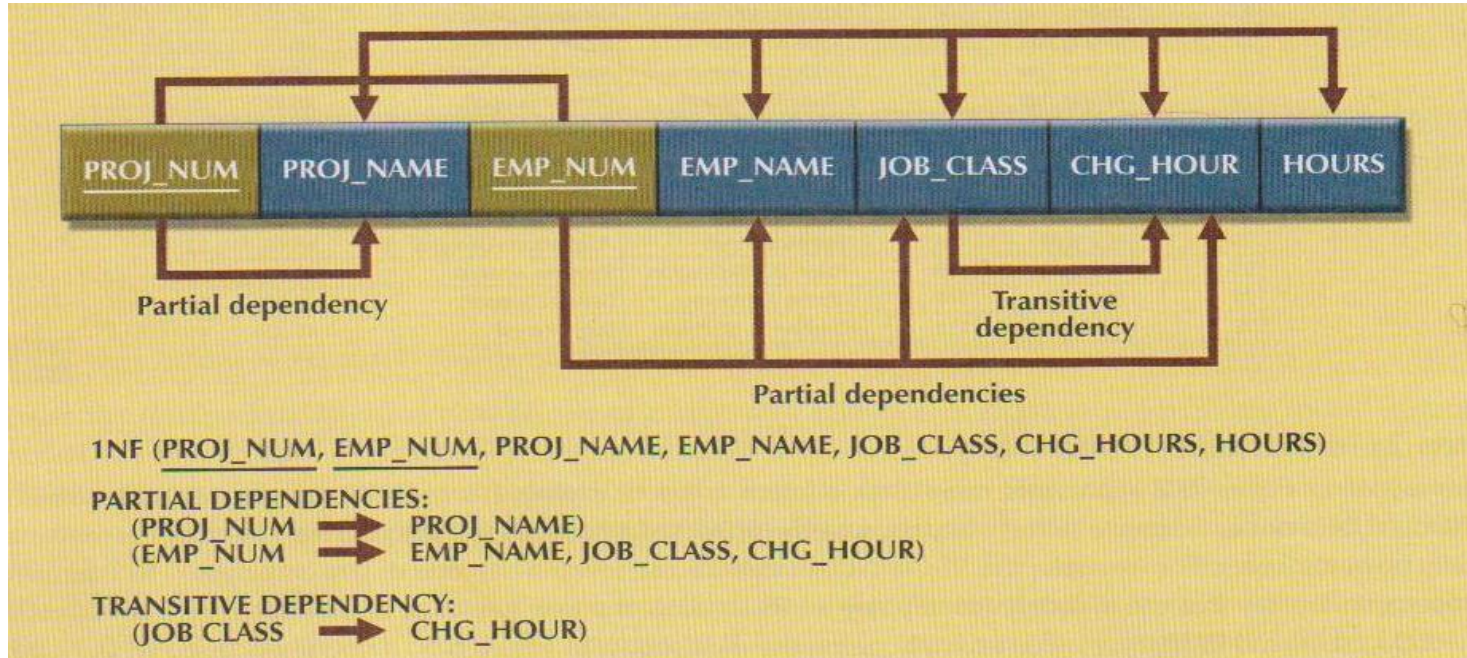
The identification of the PK in Step 2 means that you have already identified the following dependency:

تحديد PK في الخطوة 2 يعني أنك قد حددت التبعية التالية بالفعل:

Software Department

Lecture 8

PROJ_NUM, EMP_NUM \rightarrow PROJ_NAME, EMP_NAME, JOB_CLASS, CHG_HOUR, HOURS
 PROJ_NUM \rightarrow PROJ_NAME
 EMP_NUM \rightarrow EMP_NAME, JOB_CLASS, CHG_HOUR
 JOB_CLASS \rightarrow CHG_HOUR



The dependencies you have just examined can also be depicted with the help of the diagram known as dependency diagram

Partial dependency a dependency based only a part of a composite primary key

يمكن أيضاً تصوير التبعية التي قمت بفحصها للتو بمساعدة الرسم التخطيطي المعروف باسم مخطط التبعية

تبعية جزئية هي تبعية تعتمد فقط على جزء من مفتاح أساسي مركب

Transitive dependency is a dependency of one nonprime attribute on another nonprime attribute

التبعية متعددة هي تبعية لسمة غير رئيسية واحدة على سمة غير رئيسية أخرى

The term first **normal form (1NF)** describes the tabular format in which:

يصف المصطلح النموذج العادي الأول (1NF) التنسيق الجدولي الذي:

- All of the key attributes are defined.
- There are no repeating groups in the table. in other words, each row/column intersection contains one and only one value, not a set of values.
- All attributes are dependent on the primary key.
- يتم تحديد كافة السمات الرئيسية.
- لا توجد مجموعات متكررة في الجدول. بمعنى آخر ، كل صف / عمود يحتوي التقاطع على قيمة واحدة فقط ، وليس مجموعة من القيم.
- جميع السمات تعتمد على المفتاح الأساسي.

Software Department

Lecture 8

The problem with the 1NF table structure is that it contains partial dependencies.

While partial dependencies are sometimes used for performance reasons, they should be used with caution

تكمّن مشكلة بنية الجدول NF1 في احتوائه على تبعيات جزئية. بينما تُستخدم التبعيات الجزئية أحياناً لأسباب تتعلق بالأداء ، يجب استخدامها بحذر

Conversion to Second Normal Form (2NF) التحويل إلى النموذج العادي الثاني (2NF)

Converting to 2NF is done only when the 1NF has a composite primary key. if the 1NF has a single attribute primary key, then the table is automatically in 2NF. The 1NF-to-2NF conversion is simple starting with

يتم التحويل إلى NF2 فقط عندما يحتوي NF1 على مفتاح أساسي مركب. إذا كان NF1 يحتوي على سمة أساسية واحدة ، فسيكون الجدول تلقائياً في NF2. تحويل NF1 إلى NF2 بسيط يبدأ بـ

Step 1: Write Each Key Component on a Separate Line

الخطوة 1: اكتب كل مكون رئيسي على سطر منفصل

Write each key component on a separate line; then write the original (composite) key on the last line.

اكتب كل مكون رئيسي في سطر منفصل ؛ ثم اكتب الاصل (مركب) مفتاح في السطر الأخير.

PRO_NUM
EMP_NUM
PROJ_NUM EMP_NUM

Each component will become the key in a new table. In other words, the original table is now divided in to three tables

سيصبح كل مكون هو المفتاح في جدول جديد. بمعنى آخر ، تم تقسيم الجدول الأصلي الآن إلى ثلاثة جداول (PROJECT ,EMPLOYEE, and ASSIGNMENT).

Step 2: Assign Corresponding Dependent Attributes

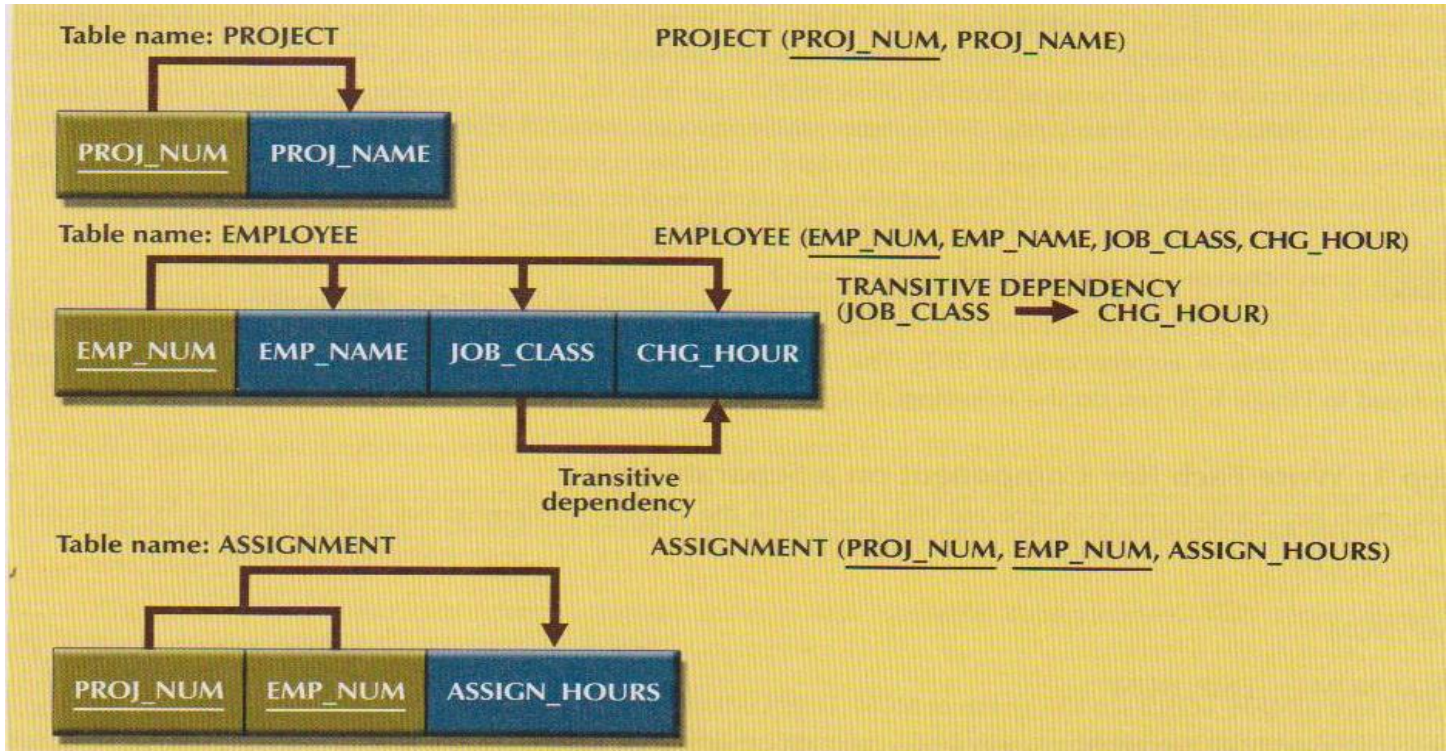
Use dependency diagram to determine those attributes that are dependent on other Attributes

الخطوة 2: تعيين السمات التابعة المقابلة
استخدم مخطط التبعية لتحديد تلك السمات التي تعتمد على الآخر
صفات

PROJECT(PROJ_NUM, PROJ_NAME)

EMPLOYEE(EMP_NUM, EMP_NAME, JOB_CLASS, CHG_HOUR)

ASSIGNMENT(PROJ_NUM, EMP_NUM, ASSIGN_HOURS)



A table is in second normal form (2NF) when the table is in 1NF and it includes no partial dependencies; that is, no attribute is dependent on only a portion of the primary key. Note that it is still possible for a table in 2NF to exhibit transitive dependency; that is, one or more attributes may be functionally dependent on non-key attributes.

- it is in 1NF

And

- it includes no partial dependencies; that is, no attribute is dependent on only a portion of the primary key. Note that it is still possible for a table in 2NF to exhibit transitive dependency; that is, one or more attributes may be functionally dependent on non-key attributes.

□ إنه في NF1

و
□ لا يتضمن أي تبعيات جزئية؛ أي أنه لا توجد سمة تعتمد على جزء فقط من المفتاح الأساسي. لاحظ أنه لا يزال من الممكن لجدول في NF2 أن يحمل تبعية متعددة؛ أي أن سمة واحدة أو أكثر قد تعتمد وظيفيًا على سمات غير أساسية.

Conversion to Third Normal التحويل إلى الوضع العادي الثالث

Step 1: Identify the Determinant Attributes

For every transitive dependency, write its determinant as PK for a new table.

JOB_CLASS

الخطوة 1: تحديد السمات المحددة

لكل تبعية متعددة، اكتب محددها كـ PK لجدول جديد.

JOB_CLASS

Step 2: Identify the Dependent Attributes

Identify the attributes that are dependent on each determinant identified in Step 1 and identify the dependency.

الخطوة 2: تحديد السمات التابعة

حدد السمات التي تعتمد على كل محدد تم تحديده في الخطوة 1 وحدد التبعية.

JOB_CLASS → CHG_HOUR

Name the table to reflect its contents and function. In this case, JOB seems appropriate.

قم بتسمية الجدول ليعكس محتوياته ووظيفته. في هذه الحالة ، يبدو JOB مناسبًا.

Step 3: Remove the Dependent Attributes from Transitive Dependencies

Eliminate all dependent attributes in the transitive relationship(s) from each of the tables that have such a transitive relationship.

EMP_NUM → EMP_NAME, JOB_CLASS

الخطوة 3: إزالة السمات التابعة من التبعية متعدية

قم بإزالة كافة السمات التابعة في العلاقة (العلاقات) متعددة من كل من الجداول التي لها مثل هذه العلاقة متعددة.

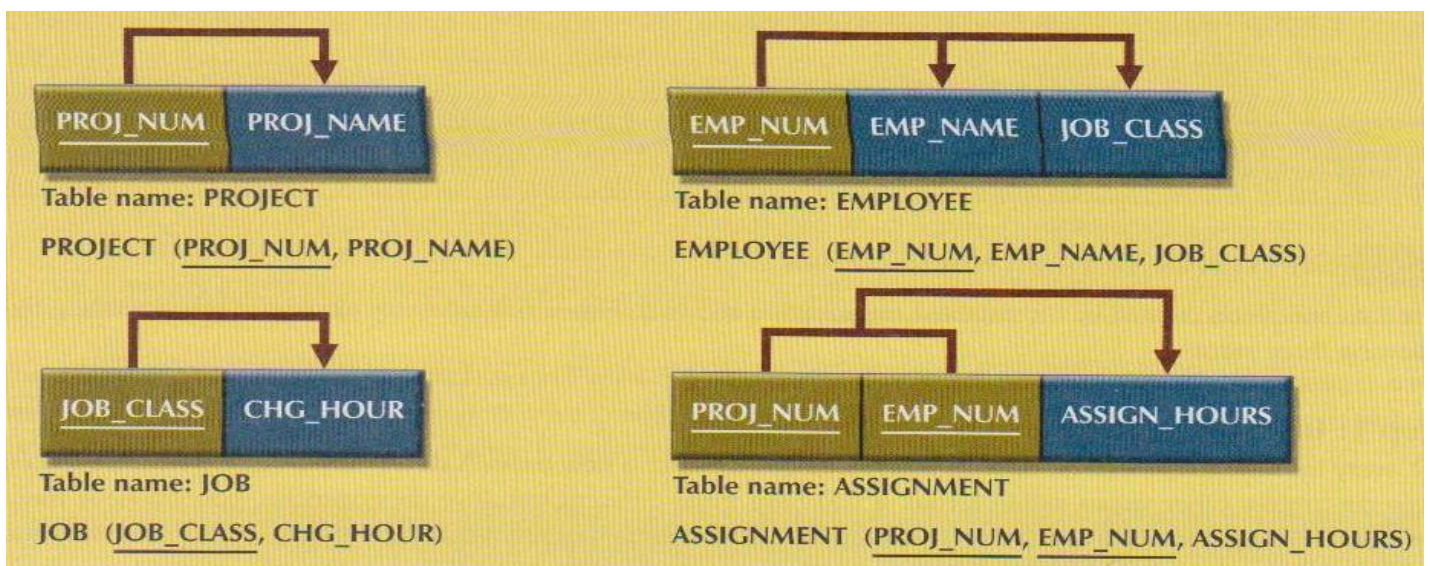
JOB_CLASS, EMP_NUM □ EMP_NAME

Note that the JOB_CLASS remains in the EMPLOYEE table to save as FK.

After the 3NF conversion has been completed, your database contains four tables:

لاحظ أن JOB_CLASS يبقى في جدول الموظف لحفظه كـ FK.

بعد اكتمال التحويل NF3 ، تحتوي قاعدة البيانات الخاصة بك على أربعة جداول:



عندما NF الجدول في 3NF A table is in 3NF when

- It is in 2NF
- لا يحتوي على تبعية متعددة متعدية It contains no transitive dependencies