

<b>Databases</b>	<b>Day 3</b>
	<b>2 hours</b>
<b>TUTORIAL 6</b> <b>Relational Theory / Relational Model / Data Dictionary</b>	

**Learning Objectives**

- (i) Learn the basic theories of Relational Model
- (ii) Learn the basics of Data Dictionary
- (iii) Learn how to construct data dictionary information for database tables

**Task 1 – Apply Basic Theories of Relational Model**

In DB Design process, a designer first creates the ER model by studying the business, processes and information requirements. The ER model is then mapped into relations via Logical DB design and these relations are finally created into DB tables in implementation.

We will examine the Relational Model this week and learn to map ER model into relations in activities next week.

**1.1 Basic Terms in Relational Model** (refer to pp 1-7 to 1-12)

The key terms used in Relational Model include: relation, attribute, degree, cardinality, tuple and domain.

**Relation** – a named table that consists of fixed number of columns and any number of rows. Each relation's name must be unique within the database.

**Attribute** – a named column in a relation. Name of each attribute must be unique within a relation.

**Degree** – the number of attributes in a relation.

**Cardinality** – number of tuples (or rows) in a relation.

**Tuple** – a row in a relation (just like an instance in an entity)

**Domain** – set of allowable values for an attribute (e.g. gender has value 'M' or 'F')

## 1.2 Apply Relational Model Terms

Use Relational Model terms on the following relations:

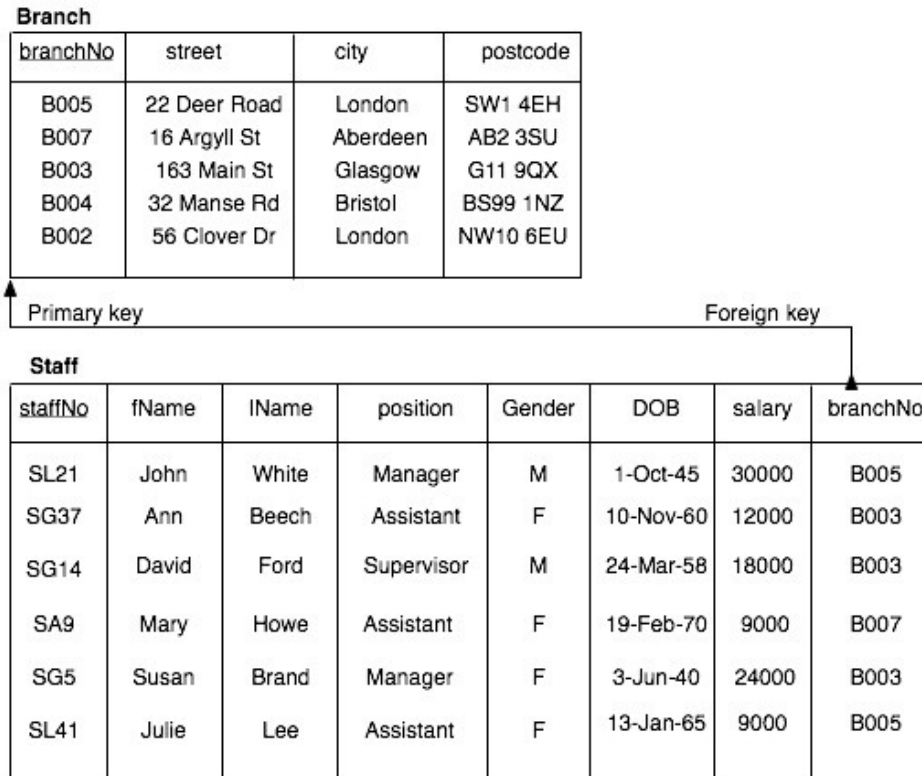


Figure 1 – Sample Relations

Source: Database Systems, A Practical Approach to Design, Implementation and Management, by Thomas Connolly, Carolyn Begg

- (i) Name the relations: \_\_\_\_\_
- (ii) Identify all of the attributes for Branch relation: \_\_\_\_\_
- (iii) What is the degree for Branch relation? \_\_\_\_\_
- (iv) What is the cardinality for Branch relation? \_\_\_\_\_
- (v) What is the domain for City attribute in Branch relation? \_\_\_\_\_
- (vi) Identify anyone tuple in Branch relation: \_\_\_\_\_

### 1.3 Properties of Relations (refer to pp1-13)

In Relational Model, the relations have to observe these properties:

- (i) A relation has a name that is different from other relations in the same database
- (ii) Each attribute name must be distinct (or unique) within a relation
- (iii) Each attribute is allowed to store one value (or atomic value)
- (iv) The values of the attribute must be from the same domain
- (v) Each tuple in a relation is distinct (in other words, tuples cannot be duplicated)
- (vi) The order of attributes in a relation is not important
- (vii) The order of tuples in a relation is not important

**1.3.1** Answer the following questions by referring to relations in Fig 1. For each question, you should answer whether the action is allowed or not allowed. If it is not allowed, explain which property in a relation that may be violated (in other words, which relation property is not followed).

- (i) Add a new attribute "Manager Name" to Branch relation \_\_\_\_\_
- (ii) Add the following tuple to Branch relation: "889Z, 1 Orchard Road, Liverpool, AB3 4TU"  
\_\_\_\_\_
- (iii) Add a new attribute "Nationality" to Staff relation \_\_\_\_\_
- (iv) John White is given another appointment, "Acting Senior Mgr". Add this appointment to his position  
\_\_\_\_\_
- (v) Management has decided to open branches overseas in Europe. A new relation named "Branch" is created to hold information on these overseas branches  
\_\_\_\_\_

### 1.4 Integrity Rules in Relational Model (refer to pp 1-20 to 1-24)

There are 2 integrity rules in Relational Model: **Entity Integrity** and **Referential Integrity**.

Integrity rules are important in database as this ensures that your data is valid. For example, integrity rule will ensure that a customer order is placed by an existing or valid customer; otherwise, you will not be able to trace who actually places this order.

- (i) What is Entity Integrity? \_\_\_\_\_
- (ii) What is Referential Integrity? \_\_\_\_\_
- (iii) Are the following actions applied to relations in Fig 1 allowed?
  - a. Add a new tuple to Staff relation with these values:  
"SL33, Margaret, Day, Supervisor, F, 1-Jun-60, 1500, B987"
  - b. Close the branch "B005" in London as it is not doing well. Remove this tuple from Branch relation.
  - c. Open a new branch at Manchester. Management has not decided on the branch code and left the branch code as "NULL".

## **Task 2 – Understand the Basics of Data Dictionary (OPTIONAL)**

In designing a database, the designer has to grapple with a host of information: database tables, their attributes or columns, the data type for attributes and constraints. What tool can a database designer use to help him organise and retrieve the relevant information in a database? The tool is the **Data Dictionary**.

Just like in everyday language usage, you refer to a dictionary when you need to check the word spelling, meaning or origin. A good DBMS (database management software) has an integrated data dictionary component to provide information and answers to the database designer or application programmer on the database details.

### **2.1 What are the terms used to refer to Data Dictionary?**

“DBMS maintains a **system catalog**, a **repository** of information containing data that describes the data in a database” Other terms used to refer to system catalog are Data Dictionary or Data Directory.

### **2.2 Why use a Data Dictionary?**

As mentioned in the preamble, a database designer grapples with thousands of pieces of information describing a database – database tables, their columns, the data types, the constraints and other relevant information. It is important that this information on database (or commonly referred to as *metadata*) is integrated into a DBMS software for easy reference.

### **2.3 What are the key information items in a Data Dictionary?**

Let's look at the basic information in a data dictionary. Below is an extract of Book Table from NP40Book:

#### **Book**

<b>Attribute</b>	<b>Description</b>	<b>Data Type</b>	<b>Constraint</b>	<b>Null Value</b>
ISBN	Unique identifier of each book	char(10)	Primary key	No
Title	Title of Book	Varchar(200)		No

**Attribute** – name assigned for each attribute in the table (this comes from the attribute attached to the entity when you are modelling the database). Attribute name must be unique in a table.

**Description** – a brief explanation or description of this attribute

**Data Type** – the data type assigned for this attribute

**Constraint** – defines whether this attribute contains a constraint like Primary Key, Foreign Key or contains certain permissible values (such as gender must be 'M' or 'F')

**Null Value** – is it permitted for the attribute to hold a Null value at the point of creating the database record? Values must be “Yes” or “No”

**Task 3 - Exercise on Data Dictionary Terms**

(a) Why must the attribute name be unique in a database table?

Ans: \_\_\_\_\_

(b) Is it permitted to have the same attribute name in another table?

Ans: \_\_\_\_\_

(c) Besides char and varchar, what are the other commonly used data types?

Ans: \_\_\_\_\_

(d) What is the difference between char(20) and varchar(20)?

Ans: \_\_\_\_\_

\_\_\_\_\_

(e) When will you define an attribute as data type number (e.g. Int, decimal) instead of char?

Ans: \_\_\_\_\_

\_\_\_\_\_

(f) Why is it important to define Constraints in a data dictionary?

Ans: \_\_\_\_\_

\_\_\_\_\_

(g) When Null value is defined as "No", what does it mean? State an example.

Ans: \_\_\_\_\_

\_\_\_\_\_

(h) When Null value is defined as "Yes", what does it mean? State an example.

Ans: \_\_\_\_\_

\_\_\_\_\_

### Task 4 – Define Items for Uniqlo's receipt

Recall the activity in week 2 when we examined the documents and identified the attributes in those documents.

Re-examine the Uniqlo's receipt document and complete the table with information for these identified attributes listed below. You should examine carefully the attributes, their format and length before filling up the table with items.



WWW.UNIQLO.COM

UNIQLO Somerset  
 313 Orchard Road #03-27/34  
 Singapore 238895  
 TEL:6834 4073  
 GST REG NO:200816544K

2000040797241 x	1	34.90 T
W's denim shorts		
2000041816897 x	1	24.90 T
W's chino shorts		
<b>Total</b>	<b>2</b>	<b>59.80</b>
Cash		60.00
Change		
Cash		-0.20

GST 3.91

Date 06/04/2013 Time 17:27 Store 0003 POS 5 Emp 5236 Tran No 197  
 You were served by CHAI LING, S



**Uniqlo's Receipt**

Attribute	Description	Data Type	Constraint	Null Value
Receipt_No	Number assigned to receipt	Char(22) E.g. 2013040617270 003050197	Primary Key	No
Receipt_Date				
Receipt_Time				
Store_No				
POS				
Employee_No				
Tran_No				
Employee_Name				
Store_Name				
Store_Address				
Store_Phone_No				
Item_Code				
Item_Name				
Item_Qty				
Item_Price				
Total_Items				

---

Attribute	Description	Data Type	Constraint	Null Value
Receipt_Amt				
Payment_Mode				
Paid_Amt				
Change_Amt				
GST_Amt				