3 Relational Database

A relational database consists of a collection of tables, each of which assigned a unique name. The table consists of a number of rows . A row in a table represents a relation among a set of values.

A relational database is a collection of data items organized as a set of tables from which data can be accessed in many different ways without having to reorganize the database tables. The relational database was invented by E. F. Codd at IBM in 1970.

A relational database is a set of tables containing data fitted into predefined categories. Each table (which is sometimes called a *relation*) contains one or more data categories in columns. Each row contains a unique instance of data for the categories defined by the columns.

Consider the account table as shown in figure 3.1 . It has three column header (three fields) :

- Branch-name
- Account-number
- Balance

Branch-name	Account-umber	Balance
A	A-101	500
B	A-215	234
C	A-234	800

Figure 3.1

Account table

3.1 Instance and schemas

Database changes over time as information is inserted and deleted.

Instance : is the collection of information stored in the database at a particular moment is called an *instance* of the database.

Schema : the overall design of the database is called the *database schema*. Schemas are changed infrequently.

The database schema is the logical design of the database, database instance is a snapshot of the data in the database at a given instance of time.

Schemas, like tables and fields, have names. For example the schema that describe the table in figure 3.1 could be :

Account table-Schema= (Branch-name , Account-number , Balance)

3.2 Keys

As you may already know, databases use tables to organize information. Each table consists of a number of rows, each of which corresponds to a single database record. So, how do databases keep all of these records straight? It's through the use of keys.

Primary Keys

The first type of key we'll discuss is the primary key. ***Every database table should have one or more columns designated as the primary ke****y*. The value this key holds should be unique for each record in the database. For example, assume we have a table called Employees that contains personnel information for every employee in our firm. We'd need to select an appropriate primary key that would uniquely identify each employee. Your first thought might be to use the employee's name.

This wouldn't work out very well because it's conceivable that you'd hire two employees with the same name. A better choice might be to use a unique employee ID number that you assign to each employee when they're hired

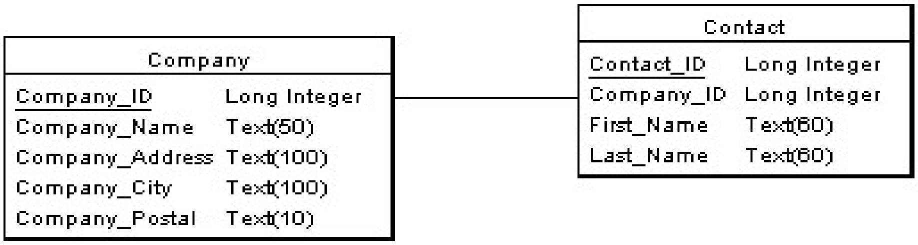
Once you decide upon a primary key and set it up in the database, ***the database management system(DBMS)*** will enforce the uniqueness of the key. If you try to insert a record into a table with a primary key that duplicates an existing record, the insert will fail.

Foreign Keys

The other type of key that we'll discuss in this course is the foreign key. These keys are used to create relationships between tables. Natural relationships exist between tables in most database structures

The foreign key link is set up by matching columns in one table (the child) to the primary key columns in another table (the parent).

Example 1 :In the example of figure 3.2, there is a link between the Company and Contact tables. The Company table is the parent table in the link. The Contact table is the child: the Company ID field in the Contact table indicates which Company a Contact belongs to.



Company	
Company_ID	Long Integer
Company_Name	Text(50)
Company_Address	Text(100)
Company_City	Text(100)
Company_Postal	Text(10)

Contact	
Contact_ID	Long Integer
Company_ID	Long Integer
First_Name	Text(60)
Last_Name	Text(60)

Figure 3.2 Example 1 of a relation between two tables

Example 2 : suppose there is a need to design a database to hold the history of each student of a school. The information of each student should have :

- Student name
- Student phone
- Information about the student in each year in the school.