18

Database: SQL, MySQL, LINQ and Java DB

Chapter 18 Database: SQL, MySQL and Java DB

Internet & World Wide Web How to Program, 5/e

OBJECTIVES

In this chapter, you'll:

- Learn fundamental relational database concepts.
- Learn Structured Query Language (SQL) capabilities for retrieving data from and manipulating data in a database.
- Configure a MySQL user account.
- Create MySQL databases.
- Learn fundamental concepts of Microsoft's Language Integrated Query (LINQ)

18.1 Introduction

- A database is an organized collection of data.
- A database management system (DBMS) provides mechanisms for storing, organizing, retrieving and modifying data for many users.
- Today's most popular database management systems are *relational databases*.
- SQL is the international standard language used almost universally with relational database systems to perform queries and manipulate data.
- Programs connect to, and interact with, relational databases systems via an interface—software that facilitates communications between a database management system and a program.

18.2 Relational Databases

- A relational database stores data in tables. Tables are composed of rows, and rows are composed of columns in which values are stored.
- A primary key provides unique values that cannot be duplicated in other rows of the same table.
- Each column of a table represents a different attribute in a row of data.
- The primary key is a column (or group of columns) with a *unique* value that cannot be duplicated in other rows.

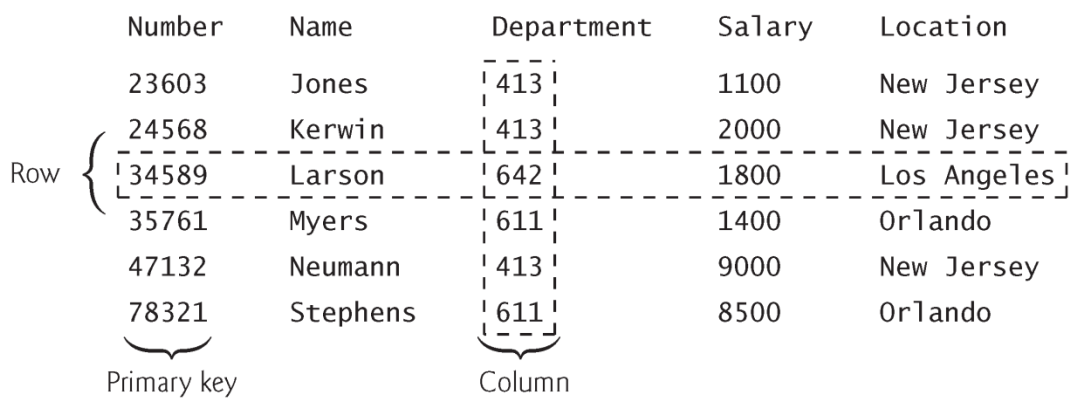

Fig. 18.1 | `Employee` table sample data.

Department	Location
413	New Jersey
611	Orlando
642	Los Angeles

Fig. 18.2 | Result of selecting distinct `Department` and `Location` data from table `Employee`.

18.3 Relational Database Overview: A books Database

- The combined columns of the `AuthorISBN` table represent the table's *primary key*—thus, each row in this table must be a *unique* combination of an `AuthorID` and an ISBN.
- This is known as the Rule of Entity Integrity.
- A one-to-many relationship between tables indicates that a row in one table can have many related rows in a separate table.
- A foreign key is a column in a table that matches the primary-key column in another table.
- The foreign key helps maintain the Rule of Referential Integrity: Every foreign-key value must appear as another table's primary-key value.

18.3 Relational Database Overview: A books Database (Cont.)

- Foreign keys also allow related data in multiple tables to be selected from those tables for analytic purposes—this is known as joining the data.

Column	Description
`AuthorID`	Author's ID number in the database. In the `books` database, this integer column is defined as autoincremented—for each row inserted in this table, the `AuthorID` value is increased by 1 automatically to ensure that each row has a unique `AuthorID`. This column represents the table's primary key.
`FirstName`	Author's first name (a string).
`LastName`	Author's last name (a string).

Fig. 18.3 | `Authors` table from the `books` database.

AuthorID	FirstName	LastName
1	Paul	Deitel
2	Harvey	Deitel
3	Abbey	Deitel
4	Michael	Morgano
5	Eric	Kern

Fig. 18.4 | Sample data from the `Authors` table.

Column	Description
`AuthorID`	The author's ID number, a foreign key to the `Authors` table.
`ISBN`	The ISBN for a book, a foreign key to the `Titles` table.

Fig. 18.5 | `AuthorISBN` table from the `books` database.

AuthorID	ISBN	AuthorID	ISBN
1	0132152134	2	0132575663
2	0132152134	1	0132662361
1	0132151421	2	0132662361
2	0132151421	1	0132404168
1	0132575663	2	0132404168
1	013705842X	1	0132121360
2	013705842X	2	0132121360
3	013705842X	3	0132121360
4	013705842X	4	0132121360
5	013705842X		

Fig. 18.6 | Sample data from the `AuthorISBN` table of books.

Column	Description
`ISBN`	ISBN of the book (a string). The table's primary key. ISBN is an abbreviation for "International Standard Book Number"—a numbering scheme that publishers use to give every book a unique identification number.
`Title`	Title of the book (a string).
`EditionNumber`	Edition number of the book (an integer).
`Copyright`	Copyright year of the book (a string).

Fig. 18.7 | `Titles` table from the `books` database.

ISBN	Title	EditionNumber	Copyright
0132152134	Visual Basic 2010 How to Program	5	2011
0132151421	Visual C# 2010 How to Program	4	2011
0132575663	Java How to Program	9	2012
0132662361	C++ How to Program	8	2012
0132404168	C How to Program	6	2010
013705842X	iPhone for Programmers: An App-Driven Approach	1	2010
0132121360	Android for Programmers: An App-Driven Approach	1	2012

Fig. 18.8 | Sample data from the `Titles` table of the books database .

18.3 Relational Database Overview: A books Database (Cont.)

- An entity-relationship (ER) diagram shows the *database tables* and the *relationships* among them.
- Every row must have a primary-key value, and that value must be unique in the table. This is known as the Rule of Entity Integrity.
- An infinity symbol (∞) indicates a one-to-many relationship, in which an entry from a table can have an arbitrary number of entries in another table.
- A *many-to-many* relationship indicates that multiple entries can be related between tables.

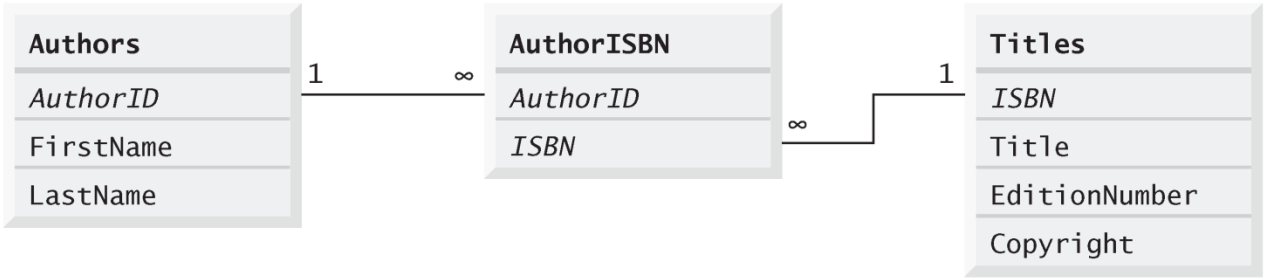

Fig. 18.9 | Table relationships in the books database.

Common Programming Error 18.1

Not providing a value for every column in a primary key breaks the Rule of Entity Integrity and causes the DBMS to report an error.

Common Programming Error 18.2

Providing the same primary-key value in multiple rows causes the DBMS to report an error.

Common Programming Error 18.3

Providing a foreign-key value that does not appear as a primary-key value in another table breaks the Rule of Referential Integrity and causes the DBMS to report an error.