

Module 2

Designing a Database

Materials Required for Module 2

r *Microsoft Access User's Guide*

Note:

A computer is not required to complete this module.

Materials Required for Instructor-Facilitated Class

r *Examples of tables that violate 1st Normal Form, meet 1st and violate 2nd, meet 1st and 2nd but violate 3rd*

r *Recommendations for further reading on database design and normalization*

Module Objectives

Lesson 1 - Relational Database Design

Upon completion of this lesson, you will be able to:

- r Define the terms Primary Key, Foreign Key, and Relationship.
- r Identify One-to-Many, Many-to-Many, and One-to-One relationships.
- r Given the needs of a database user, identify the tables, fields, and relationships that are needed.

Lesson 1

Relational Database Design

Reading Assignment

- r *Microsoft Access User's Guide*
Chapter One: *Designing a Database*

As You Read

- r How important is good database design? Consider a telephone company that stores its customer information according to telephone numbers. If you installed a second line for a fax machine, how many bills would you receive each month?

If the company tracked by phone number > Two
If the company tracked by address > One

How many records would need to be modified if you moved to a new address?

If tracked by phone number > Two

Anyone could claim to have a calling card and give a random number.

Exercises

- 1) What is information and how does it differ from data?
See page 2 - Facts presented in a useful manner..
- 2) What is a relationship in a database?
See page 12 - Information contained in the database which associates data stored in different tables.
- 3) Describe a One-to-Many relationship.
Example: Employees <-> Orders
See page 14 - A relationship where a single record in Table A can be related to several records in Table B.
- 4) Describe a Many-to-Many relationship.
Example: Classes <-> Students
See page 15 - A record in Table A can have more than one matching record in Table B, and a record in Table B can have more than one match in Table A.
- 5) Describe a One-to-One relationship.
Example: Products <-> Descriptions
See page 18 - For each record in Table A there is at most one related record in Table B. For each record in Table B there is at most one related record in Table A.
- 6) Why should data like a suppliers' address or the current tax rate appear only once in your database?
See Page 6 - a) You will have to find and update multiple locations of the data. b) Data integrity. c) You are wasting memory and disk space.
- 7) When should you include calculated or derived data in your tables?
See page 9 - There is no need if it can be computed from existing data, however, there may be times when it is preferable in order to decrease access time.
- 8) If you were tracking each personal phone call made while at work, would the date of the call make a good primary key?
See page 11 - No, you might receive more than one call in any given day.
- 9) If you were tracking the time that you arrived to work each morning would the date make a good primary key?
See page 11 - Yes, since you only arrive at work once each morning.

Points to Ponder

Let's go back to the telephone company example discussed at the beginning of this module.

What would be the drawbacks to tracking customers and telephone numbers by addresses?

On-Your-Own Lab

Through the completion of self-study labs you will be creating your own Address Book database in MS Access. Each time you learn how to do something new in this course you will practice what you have learned by adding features to your own database.

Keep in mind that there is not only one "right" way to implement a feature in a relational database. You may find that you will include fields or tables that no other participant feels are necessary. During the instructor-led labs that follow each module, you will compare your design and implementation with those of other participants so that you have a chance to discuss the benefits and limitations of different designs; just as you will discuss different options with customers.

You may personalize your database so that it suits your needs, however, in order to complete future exercises you must include certain tables, fields, relationships, and forms. At the end of each lab, you will be given an example database which meets these requirements. You may either use it for the following labs or add to your design.

1. List ten questions that you would like to be able to answer by querying your database:

Hint: 1) Who in your Address Book has a birthday in January ?

Hint: 2) How many cats, dogs, and birds does each family have ?

Hint: 3) How many ski? (You want to invite them to watch a slide show of your last trip to Whistler.)

Ask students for the questions they listed here.

On-Your-Own Lab

2. List the tables that you need in order to answer the questions above. Fill in the spaces provided below:

Hint: Although you could implement this database with as few as one table, four are recommended. In order to answer the first three questions in step one you need to have a table for the people you want to track, a table for their dependents, one for their hobbies, and a table to track their pets.

Ask two different students to write their tables on the board and discuss any problems or omissions in those tables.

Discuss the drawbacks to this typical design:

<u>Main Contacts</u>	<u>Dependents</u>	<u>Hobbies</u>	<u>Pets</u>
Contact ID	Dependent	Contact ID	Pet ID
Last Name	Main Contact ID	Hobby Name	Pet Name
First Name	Relationship	Description	Owner ID
Address	Last Name		Pet Type
City	First Name		
State	Date of Birth		
Zip Code			
Work Phone			

Home Phone

Date of Birth

1) What if you have two freinds who are related? They will be repeated in the Contacts table and the Dependants table.

2) What if you want a listing of everyone with the last name Smith? You will need a query on Contacts and then one on Dependants.

3) Hobbies are repeated. It would be better to have a Hobby Assignments table.

4) Zip Code is not directly dependant on the Contact ID. It is related to the City and State. It would be better to have the zip code as a foreign key to a City and State table.

Two good solutions to the Contacts/Dependants issue:

1) Use a higherarchical structure like the Employees table in Nwind (the Manager ID column lists the Employee ID of their manager.

2) Have the following three tables:

<i>Contact</i>	<i>Association</i>	<i>Associate Grouping</i>
<i>Contact ID</i>	<i>Contact ID</i>	<i>Grouping ID</i>
<i>etc.</i>	Grouping ID	Description

<i>Conta</i>	<i>Last</i>	<i>First</i>
<i>ct ID</i>	<i>Name</i>	<i>Name</i>

<i>1</i>	<i>Smith</i>	<i>Joe</i>
----------	--------------	------------

2 *Doe* *Jane*

3 *Jones* *Sue*

<i>Association Grouping</i>	<i>Description</i>
<i>MyFam</i>	<i>My Close Reletives.</i>
<i>SpouseFam</i>	<i>My Spouses Close Reletives.</i>
<i>Adams</i>	<i>The Adams Familly</i>
<i>Newtons</i>	<i>The next door neighbors</i>
<i>MyWork</i>	<i>People I work with.</i>
<i>Jones</i>	<i>The familly we used to live next to.</i>
<i>Swim</i>	<i>The members of my swim team</i>

Conta *Groupi*
ct ID *ng ID*

1 *MyFam*

2 *Newton*
 s

2 *Swim*

3 *Spouse
Fam*

3 *Adams*

3. Each of the subjects you listed above will be a table in your database. Under each subject list the information you want in each table.
4. Now that you have decided on what data to put in each table you need to determine the relationships that will help you to obtain the information to answer your ten questions above. Draw a line from the Primary Key to the Foreign Key for each relationship.

Relationships:

1) *Main Contacts.Contact ID <---> Dependents.Main Contact ID*

2) *Main Contacts.Contact ID <---> Hobbies.Contact ID*

2) *Main Contacts.Contact ID <---> Pets.Owner ID*

5. Now that you have a design, check to make sure that it meets your needs. Can you answer each of your ten questions by looking at the data in your tables?

The PSS Challenge

In the As You Read and Points To Ponders sections of this module you were asked to consider the drawbacks of tracking telephone customers by their name, household, or phone number. By applying what you have read in chapter one of the User's Guide design a solution that will allow the telephone company to obtain the following information: the names of all customers it has done business with, the addresses of each household it has done business with, all phone numbers the company can assign, and a list of the phone numbers that are currently assigned to a household which includes the name of the person(s) responsible for paying the bill for that number.

A Customer table for tracking name, address, and complaint history. A phone number table for tracking numbers, the regions they are valid in, and the services accessible from those numbers. An assignments table for tracking which numbers are assigned to which customers.

Instructor Led Module Review

1) *Ask participants if they have any questions. If none are asked, ask students some of the questions from the exercise sections:*

What is a relationship?

What is a Foreign Key?

When would you store calculated values in you tables.?

2) *Review As You Read & Points to Ponder*

3) *Present Lab Solution*