Introduction to Computer Applications

CISY 1225
Chapter 10

Zahoor Khan, PhD

Last updated: May 2014

Next Week

- No new topics
- I will not be here
  - Mashtura will help you
- Class time is for practice
- Do your own practice
- Projects 10-1 and 10-2 available on Friday
- Try to finish all projects and exercises

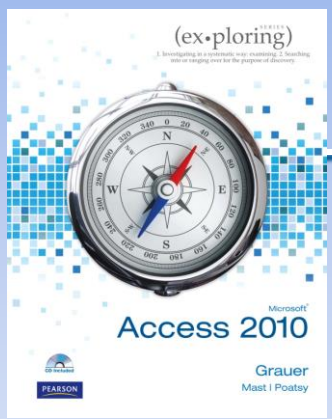
2

CISY 1225 Custom book

Chapter 10
Relational Databases and Queries

Exploring Microsoft Access 2010

by Robert Grauer, Keith Mast, Mary Anne Poatsy



Chapter 2
Relational Databases and Queries

Objectives

- Create tables
- Understand table relationships
- Share data with Excel
- Establish table relationships
- Use the Query Wizard
- Create and modify
  - a single and multiple table query

Table Design: Designing Data

- Table Definition
  - Storage location in a database
- Input vs. Output in Design

Designing Fields Guidelines

1. Include the necessary data
2. Design for now and the future
3. Store data in its smallest parts
4. Add calculated fields to a table
5. Design to accommodate date arithmetic
6. Link tables using common fields

Include Necessary Data

- Determine what data is necessary
- Create a rough draft of reports that may be needed
- Create tables based on fields necessary for reports

Design for Now and the Future

- Organizations evolve over time
- Databases should evolve with the organization
  - Anticipate future needs of the organization
  - Build flexibility into system to satisfy future needs

Store Data in Smallest Possible Pieces

- Creating a name field with the entire name in it violates good database design and reduces the usefulness of the data
- Divide data into the smallest pieces that you're going to need to access
  - Example: Prefix, FirstName, LastName, Suffix
- Provide flexibility for the user

Calculated Fields in a Table

- Produce a value from an expression or function that references one or more existing fields
- Access 2010 allows the user to store calculated fields
  - Can be a benefit or a potential problem
  - Exercise caution when using calculated (derived) fields

 11

Design to Accommodate Date Arithmetic

- Calculated fields can also create date/time data
- Plenty of examples available for using date/time date arithmetic

 12

Link Tables Using Common Fields

- Tables may be joined based on a common field
- Join lines are created
  - Manually by the user, or
  - Automatically by Access when two fields in separate tables share the same name between two related tables
- Avoid Data redundancy errors
  - The unnecessary storage of duplicate data in two or more tables

 13

Creating Tables

- Create fields in Design View
- Import data from another database or application
  - Examples: Excel spreadsheets or Word text files
- Enter data directly into rows in Datasheet view

 14

Creating Fields in Tables

- Field names should be meaningful
- Rules for naming fields:
  - Length can be up to 64 characters
  - Can include letters, numbers and spaces
  - Access uses **CamelCase** notation
    - Use uppercase letters for each first letter of each new word
    - Example: ProductCost

Field Data Types

- Every field has a data type
- Determines:
  - The type of data that can be entered
  - The operations that can be performed on that data
- Access recognizes 10 data types

Access Data Types

- Number
- Text
- Memo
- Date/Time
- Currency
- Yes/No
- OLE
- AutoNumber
- Hyperlink
- Attachment

 17

Foreign Key Review

A field in one table that is also a primary key of another table

SpeakerID is the primary key of the Speakers

SpeakerID is the foreign key in the SessionSpeaker table (duplicates are allowed)

Speakers

SpeakerID	First Name	Last Name
1	YourName	YourName
2	Warren	Brasington
3	James	Shindell
4	Edward	Wood
5	Kristine	Park
6	William	Williamson
7	Holly	Davis
8	David	Tannen
9	Jeffrey	Jacobsen
10	Jerry	Masters
11	Kevin	Kline
12	Jessica	Withers
13	Betsy	Allman

SessionSpeaker

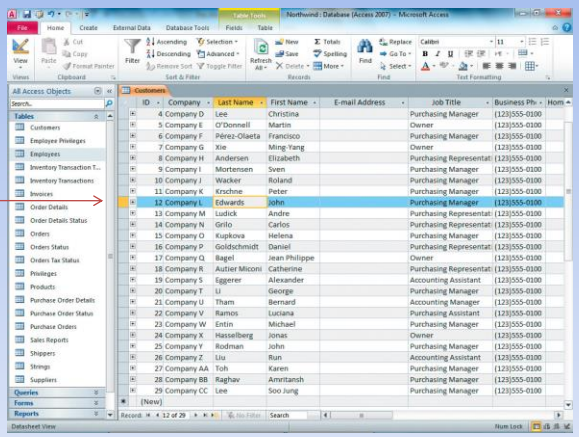
SpeakerID	SessionID
1	S01
1	S04
1	S09
1	S10
1	S13
2	S01
3	S02
3	S06
3	S09
4	S09
4	S12
5	S02
5	S07

 18

Using Table Views

Datasheet View

Active record



 19

Work with Field Properties

- Field property
- Text data type
- Number data type
- Caption property
- Validation rule

 20

Access Field Properties

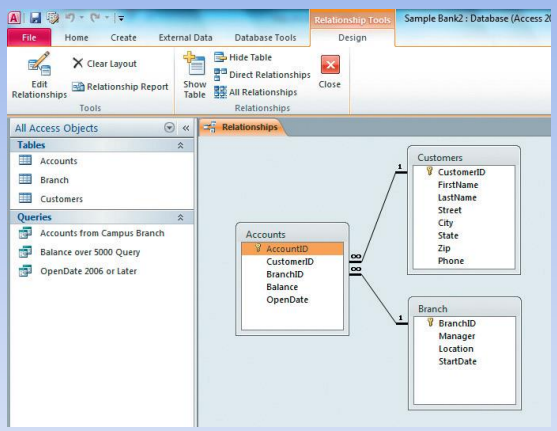
- Field Size
- Format
- Input Mask
- Caption
- Default Value
- Validation Rule
- Validation Text

Access Field Properties (continued)

- Required
- Allow Zero Length
- Indexed
- Expression
- Result Type

Understanding Table Relationships

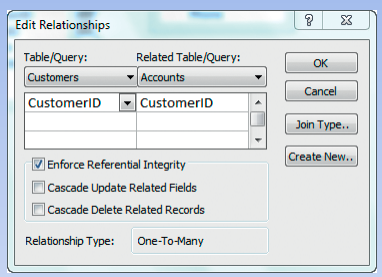
- Efficiently combine data from related tables
- Create queries, forms, and reports



 23

Establishing Referential Integrity

- Edit Relationships dialog box
- Select Enforce Referential Integrity checkbox



 24

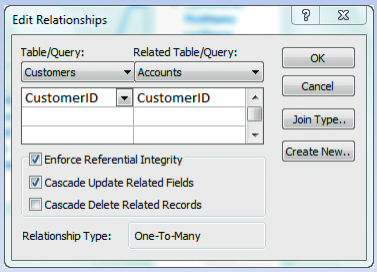
Set Cascade Options

- Cascade Update Related Fields
- Cascade Delete Related Records

Click Enforce Referential Integrity

Click Cascade Update in case the primary key changes

Click Cascade Delete with caution



Indexing to Retrieve Data Quickly

- Provides quick sorting based on the primary key
- Provides quick retrieval of data based on the primary key

Options on External Data Tab

- Import & Link
- Export
- Collect Data
- Web Linked Lists

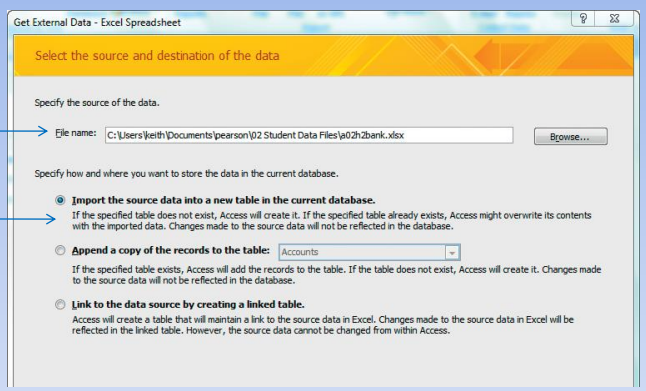
Click Excel to import spreadsheet data

External Data Tab

Import Data from Excel

Click Browse to find a spreadsheet

Decide what you want to do with the data

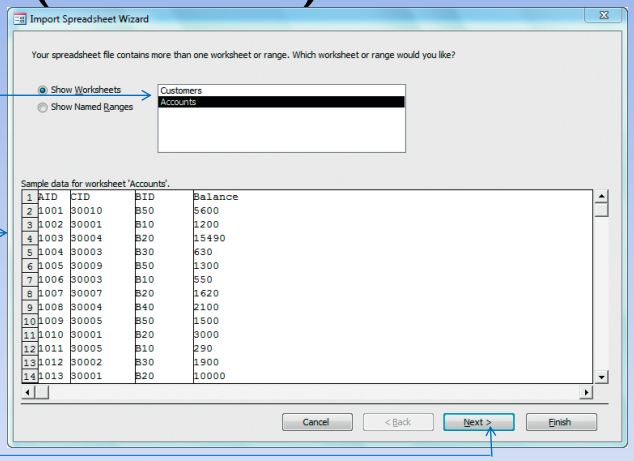


Import Data from Excel (continued)

Choose the worksheet to import

Preview of the worksheet data

Click Next to continue

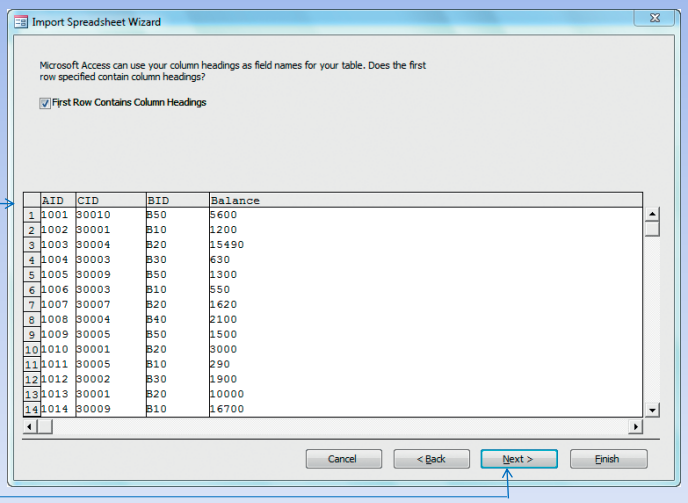


 29

Import Data from Excel (continued)

Column headings

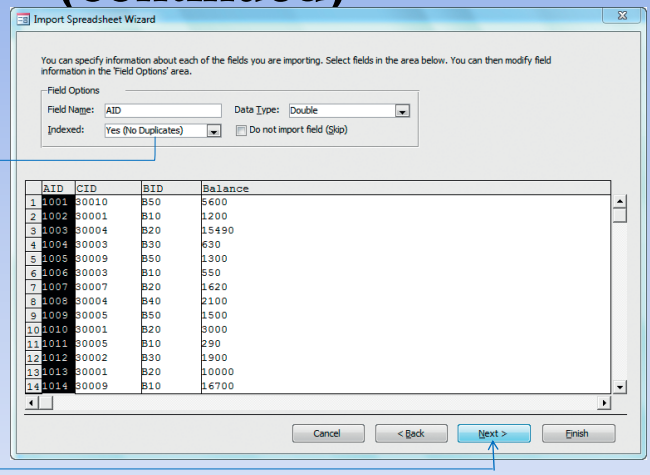
Click Next to continue



 30

Import Data from Excel (continued)

Change Indexed property to *Yes (No Duplicates)*

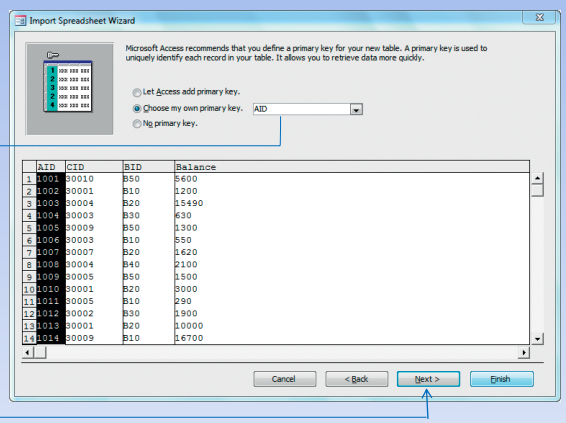


Click Next to continue

 31

Import Data from Excel (continued)

AID becomes the primary key

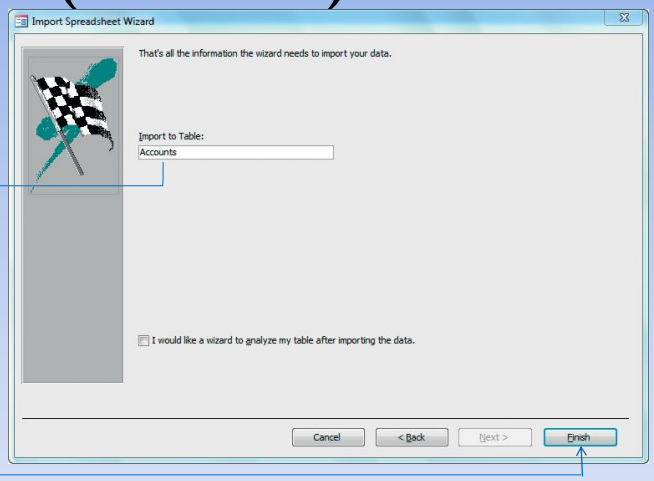


Click Next to continue

 32

Import Data from Excel (continued)

Accounts becomes the table name



Click Finish to import the data

 33

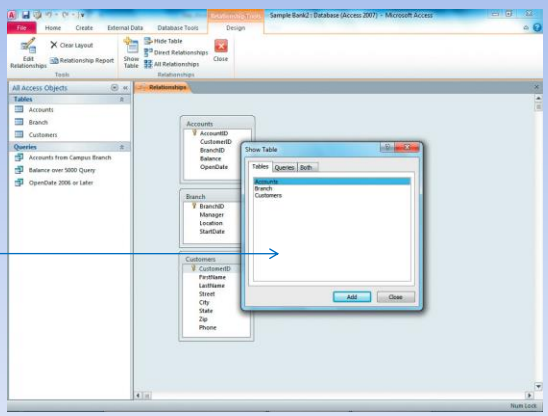
Types of Relationships

- One-to-one relationship
  - Example: you have one birth certificate
- One-to-many relationship
  - You are taking multiple courses
- Many-to-many relationship
  - Artificially constructed relationship

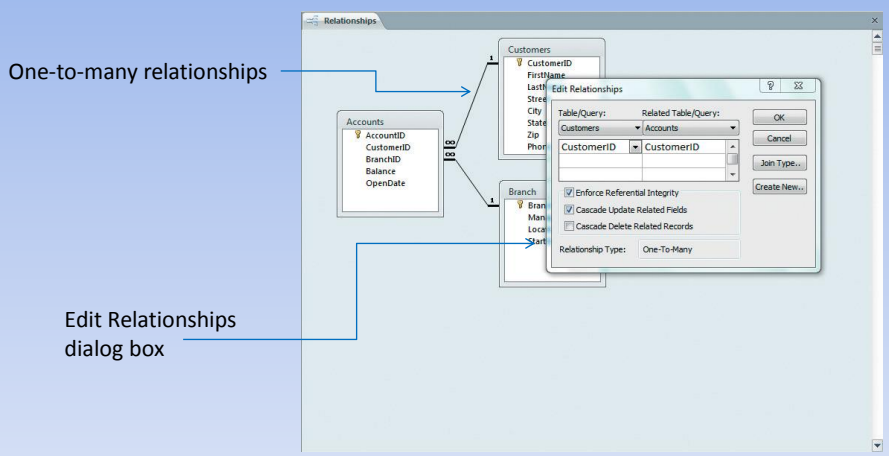
 34

Establishing a One-to-Many Relationship

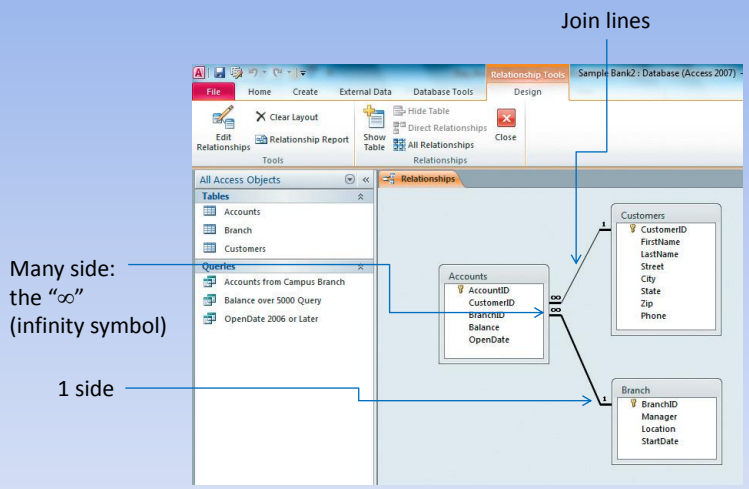
- Open Relationships window
- Add tables
- Establish relationships



Relationships Between Tables

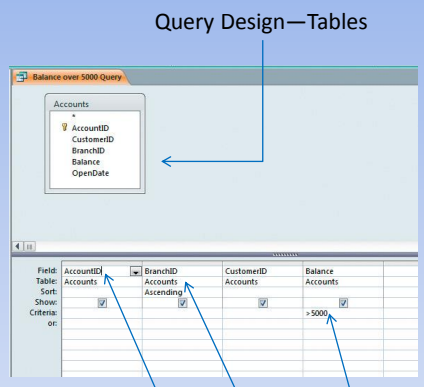


Relationships Window



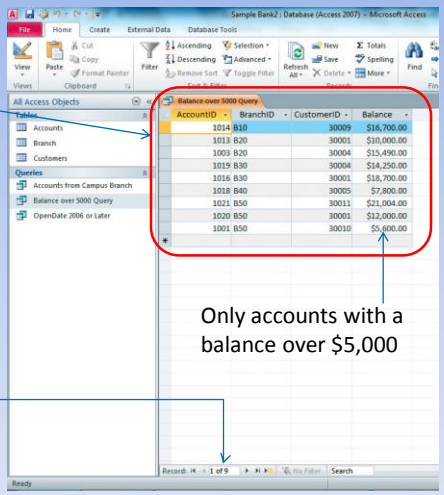
Single-Table Queries

- Show Table
- Design Grid to add
  - Field row
  - Table row
  - Sort row
  - Show row
  - Criteria



Datasheet View of Results

Query results in Datasheet view



Only accounts with a balance over $5,000

Nine records match the criteria

Specifying Criteria for Different Data Types

- Field data type
- Delimiters
- Criteria

Wildcards

- The question mark (?)
  - Example: *H?LL* will return *Hall*, *Hill*, and *Hull*
- The asterisk (*)
  - Example: *S*nd* will return *Sand*, *Stand*, and *StoryLand*

Operators & Operands

- Operators
- Operands

Null and Zero-Length Strings

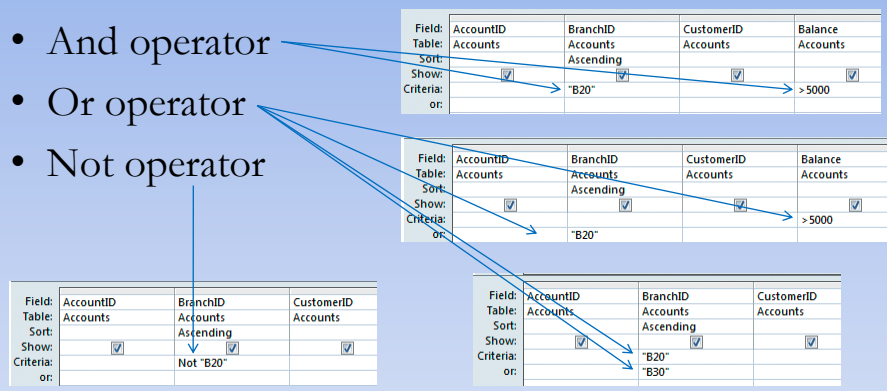
- Null
- Is Not Null

Query Sort

- Determines the order of records
- Determined from left to right

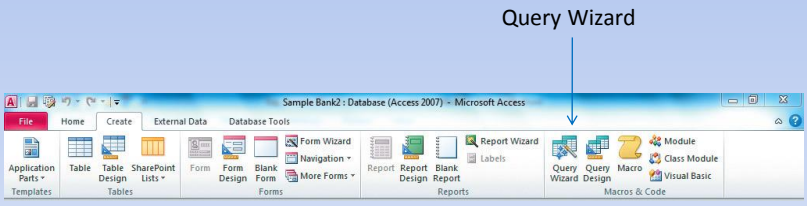
AND, OR, and NOT Criteria

- And operator
- Or operator
- Not operator

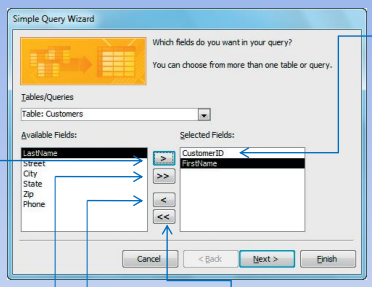


Using the Query Wizard

- Launch Query Wizard
- Modify Query in Design view



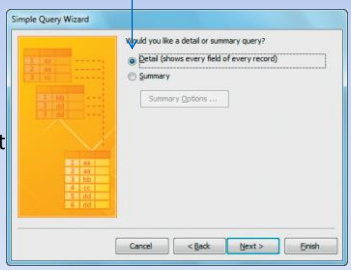
Query Wizard



Fields already moved to the Selected Fields list

Select a Table or Query

Select detail or summary data



Remove all fields from the Selected Fields list

Remove a single field from the Selected Fields list

Move all fields to the Selected Fields list

Move a single field to the Selected Fields list

Multi-Table Queries

- Permit multiple tables
- Use related tables
- Similar to creating a single-table query
- Fixing a common problem

Summary

- This chapter introduced the concept of tables and query design.
- Tables and forms are used to input data, and to create queries and reports to extract information from the database in an organized and useful way.
- The information to be extracted, though, is dependent on the quality of the underlying tables.

Questions

Copyright