

Microsoft Access 2007 – Relationship



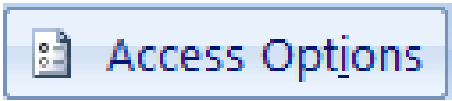
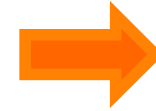
Asst. Prof. Wassana Naiyapo and Kittipitch Kuptavanich
Adapted and Reassembled by Prakarn Unachak
Department of Computer Science, Faculty of Science, Chiang Mai University

Data for This Lab

- ▶ <https://goo.gl/XYTc5p>
- ▶ <https://goo.gl/ShNjmJ>
- ▶ Save the file *13_student_class.accdb* and *13_student_class_rel.accdb* on your computer.



Changing Font Size



Query Design

Datasheet

Access Options

General

Current Database

Datasheet

Object Designers

Proofing

Language

Client Settings

Customize Ribbon

Quick Access Toolbar

Add-ins

Trust Center

Customize the way c

Gridlines and cell effects

Default gridlines showing

☒ Horizontal

☒ Vertical

Default cell effect

☒ Flat

☐ Raised

☐ Sunken

Default column width: 1"

Default font

Size: 12

Weight: Normal

☐ Underline

☐ Italic

Access Options

General

Current Database

Datasheet

Object Designers

Proofing

Language

Client Settings

Customize Ribbon

Quick Access Toolbar

Add-ins

Trust Center

Change the default settings for des
datasheet and layout view.

Table design view

Default field type: Text

Default text field size: 255

Default number field size: Long Integer

AutoIndex on Import/Create: ID;key;code;num

☒ Show Property Update Options buttons

Query design

☒ Show table names

☐ Output all fields

☒ Enable AutoJoin

Query design font

Font: Segoe UI

Size: 14

SQL Server Compatible Syntax (ANSI 92)

☐ This database

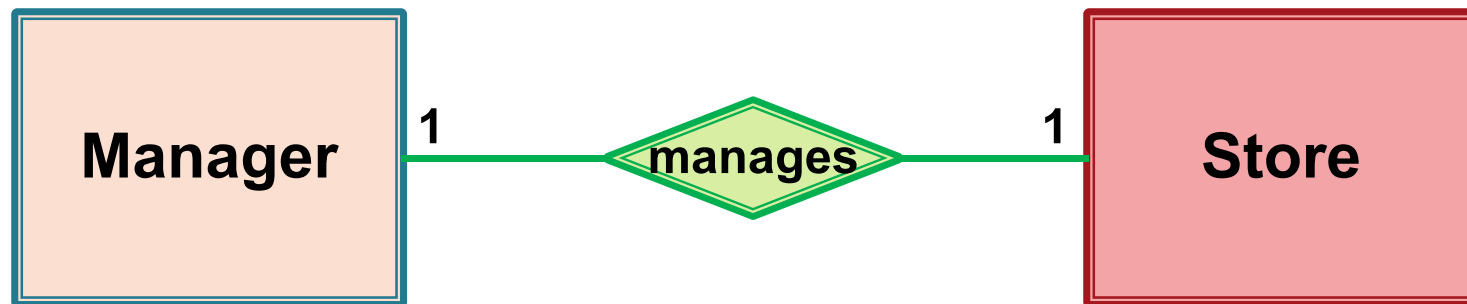
☐ Default for new databases

Review: Relationship

- ▶ **Relationship** tells you how records in one table relate to record in another table.
 - How different data relate.
- ▶ **Type of Relationship:**
 - One to one (1:1)
 - One to many (1:N)
 - Many to many (M:N)

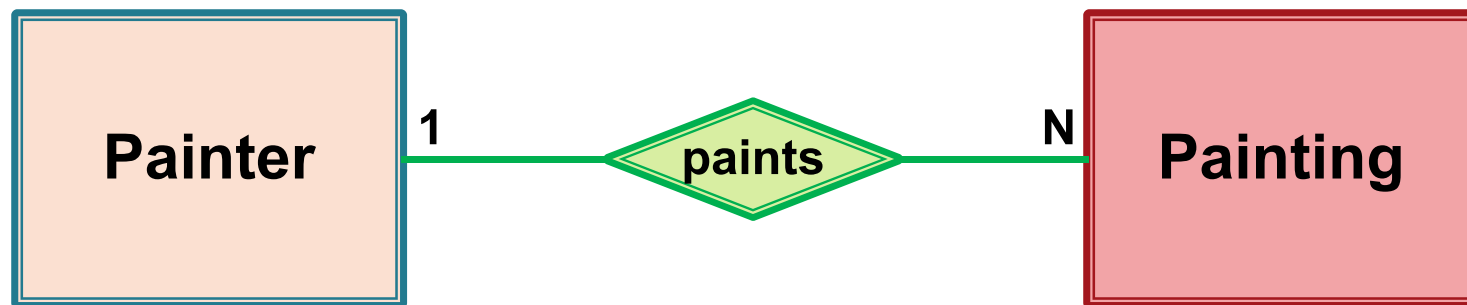
One-to-One (1:1)

- ▶ Store – Manager
 - Each store is managed by a store manager
 - Each store manager manages only one store



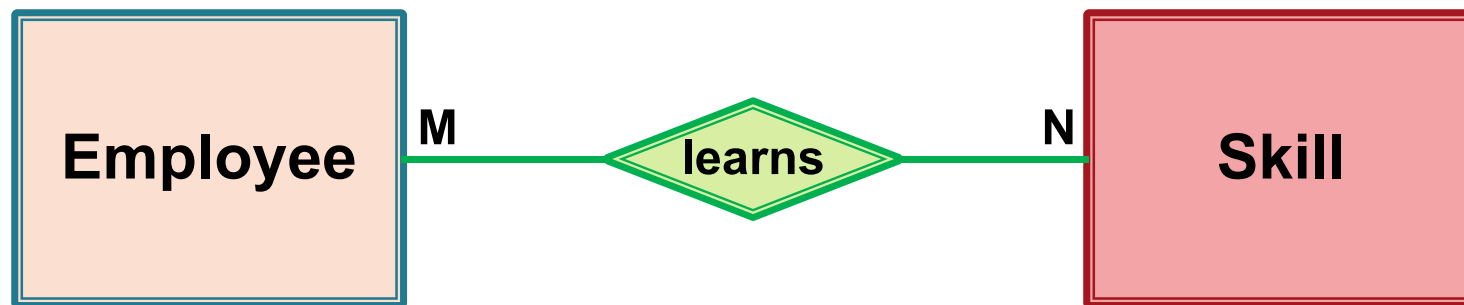
One-to-Many (1:N)

- ▶ Painter – Painting
 - A painter creates many paintings
 - Each painting is painted only by one painter



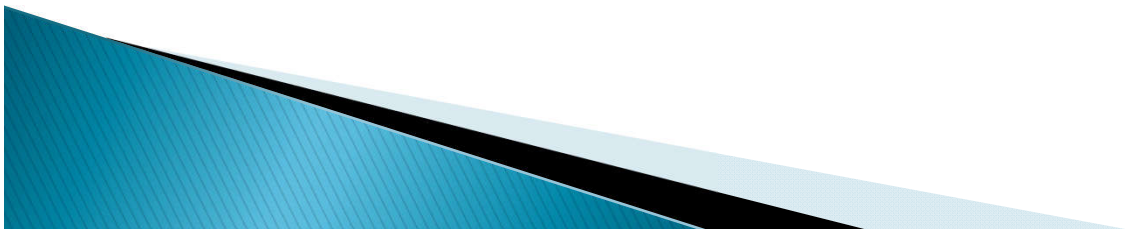
Many-to-Many (M:N)

- ▶ Employee – Job Skill
 - An employee may learn many job skills
 - Each job skill may be learned by many employees



Getting Start

Now, we will start
with *13_student_class.accdb*



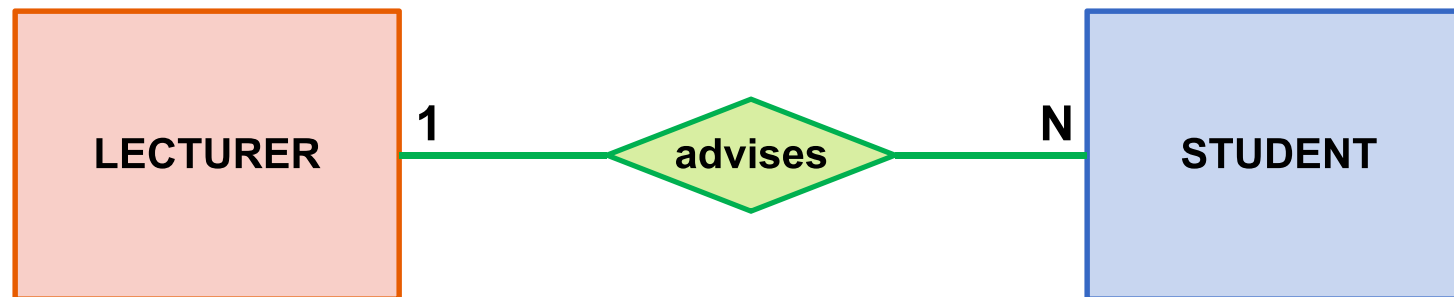
Student Table Revisited



Field Name	Data Type	Field Size
Student ID	Text	9
Title	Text	10
First Name	Text	20
Last Name	Text	20
Birth Date	Date/Time	
GPA	Number	Single
Blood Type	Text	5
Pet	Text	20
Allowance	Currency	
E-mail Address	Text	50

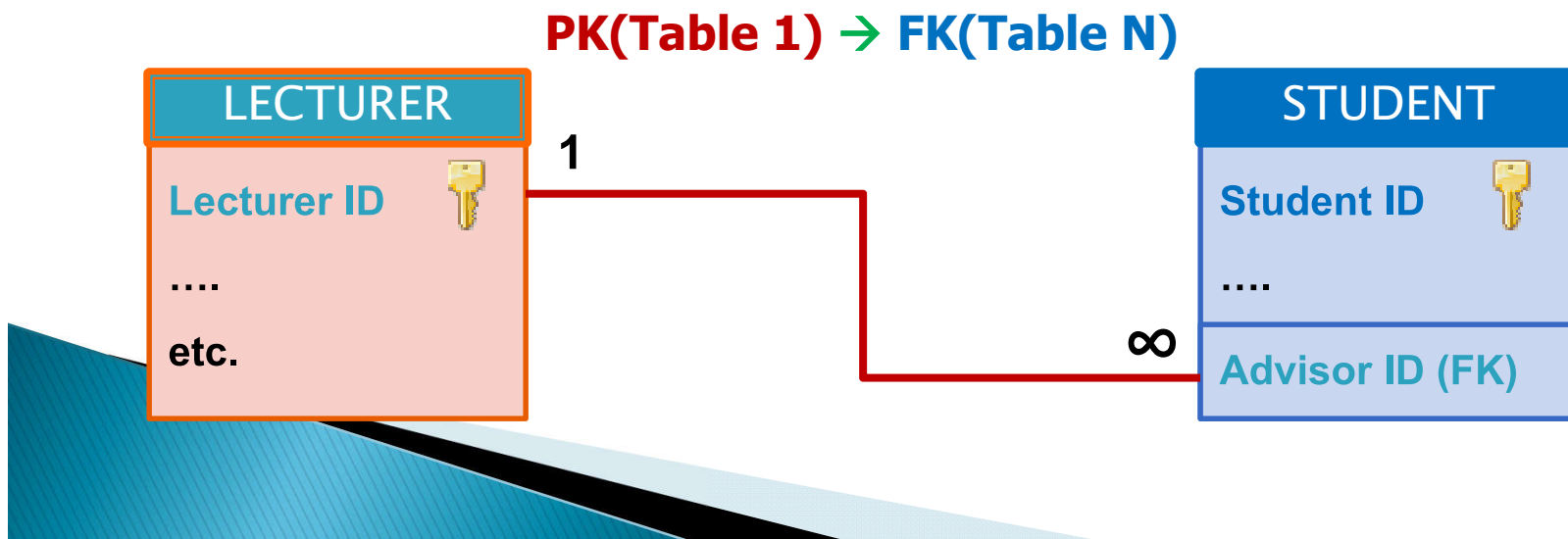
Student – Advisor

- ▶ A Student has only one advisor
- ▶ A lecturer can advise many students
- ▶ 1:N or N:1?



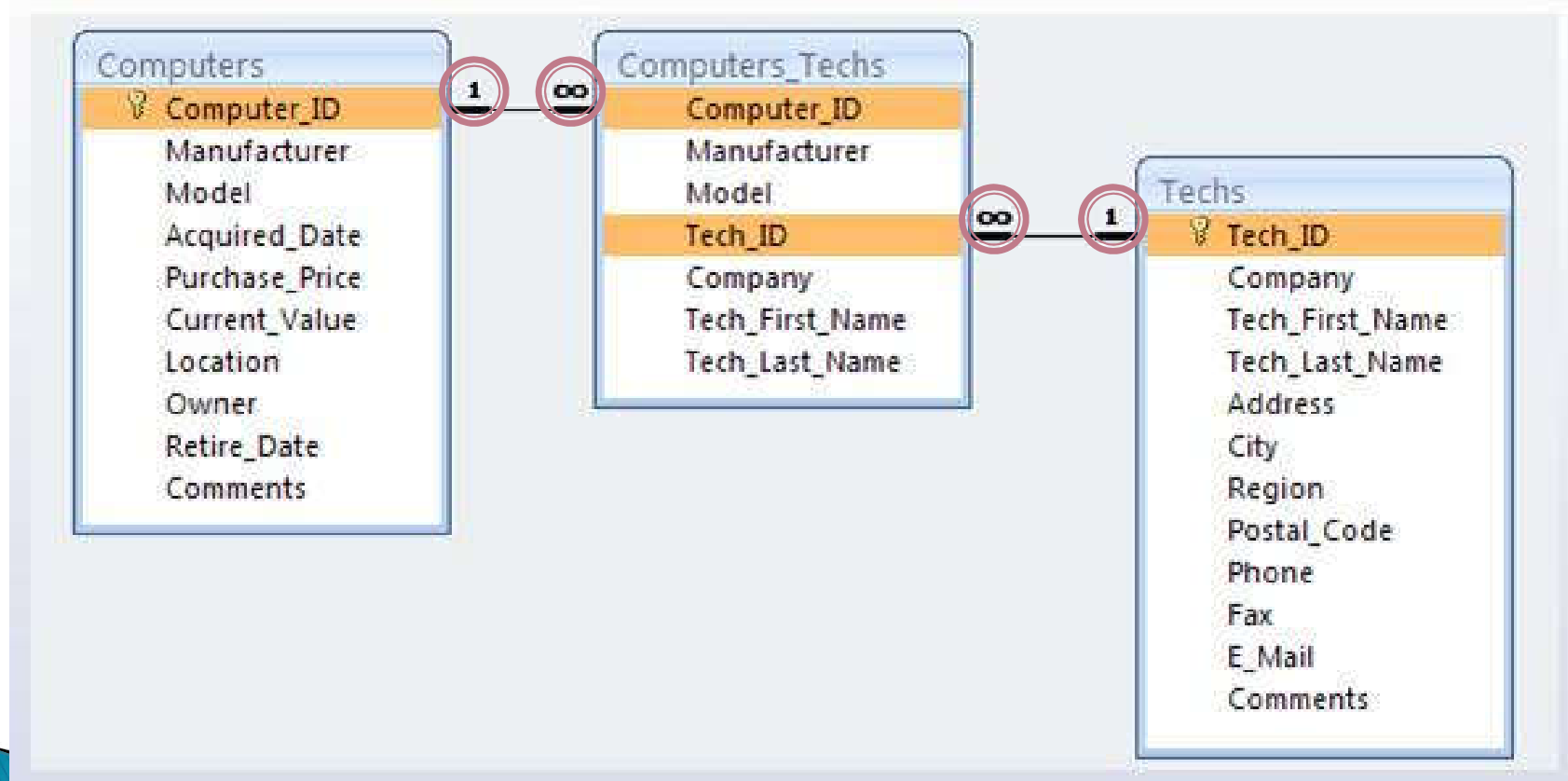
Relationship with Foreign Key

- ▶ One-to-Many Relationship type.
 - Relationship can be created by including a field from the LECTURER table as a new field in STUDENT table. The new field is call a foreign key
- ▶ If the relationship is 1:N, the foreign key will always be on the many side.



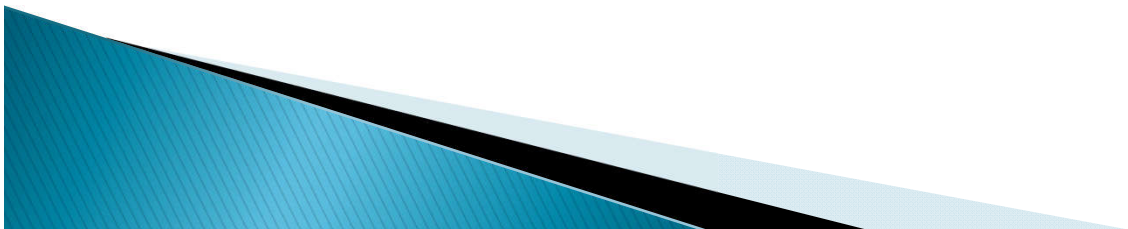
Relationship in MS Access

- Between 2 tables, MS Access can only do 1:1 and 1:N (Not N:M).



Restriction

- ▶ In a database, all table names must be unique
 - Can't have two tables with the same name
- ▶ In a table, all field names must be unique
 - Can't have two columns with the same name.



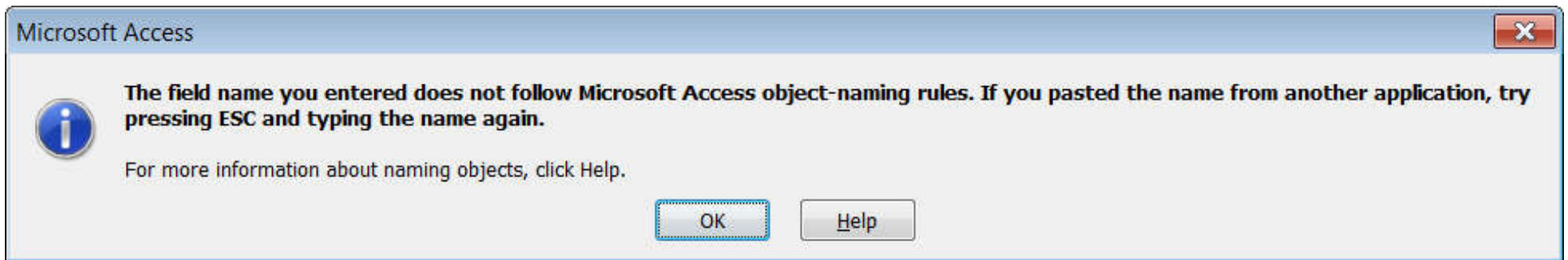
Lecturer Table

Lecturer		
	Field Name	Data Type
🔑	Lecturer ID	Text
	Title	Text
	First Name	Text
	Last Name	Text
	Office	Text
	Phone Number	Text
	E-mail Address	Text

- ▶ Right Click –> Design View
 - Which field is the Primary key?
 - What is the Data Type?
 - How do you find the Field Size?

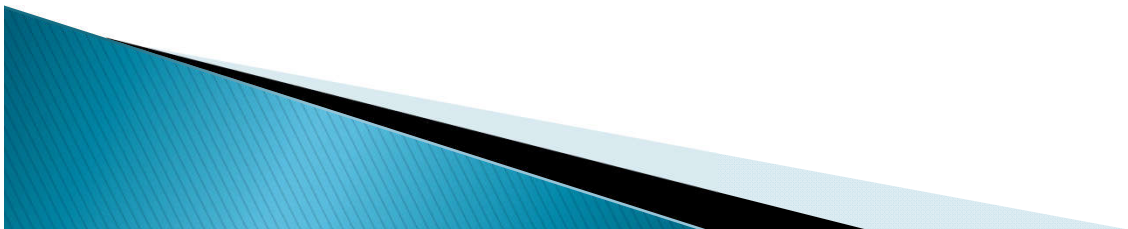
Creating a Foreign Key (FK)

- ▶ Student Table –> **Design View**
- ▶ Create a new Field called “Advisor.ID”
 - Enter



Field Name

- ▶ Can be up to 64 characters long
- ▶ Cannot include
 - ! (an exclamation point)
 - [] (brackets)
 - ` (a grave accent)
 - . (a period)
- ▶ Cannot begin with leading spaces



Creating a Foreign Key (FK)

- ▶ Create a new Field
 - Called “Advisor ID”
 - Data Type: “Text”
 - Field Size: 8
 - Data Type and Size has to be the same as the PK of the related table.
 - The field **name** can be different.

Student		
	Field Name	Data Type
🔑	Student ID	Text
	Title	Text
	First Name	Text
	Last Name	Text
	Birth Date	Date/Time
	GPA	Number
	Blood Type	Text
	Pet	Text
	Allownance	Currency
	E-mail Address	Text
	Advisor ID	Text

The “Indexed” Option

General	Lookup
Field Size	8
Format	
Input Mask	
Caption	
Default Value	
Validation Rule	
Validation Text	
Required	No
Allow Zero Length	Yes
Indexed	Yes (Duplicates OK)
Unicode Compression	Yes
IME Mode	No Control
IME Sentence Mode	None
Smart Tags	

▶ Duplicates (OK)

- One-to-Many (1:N) ✓

▶ No Duplicate

- One-to-One (1:1)

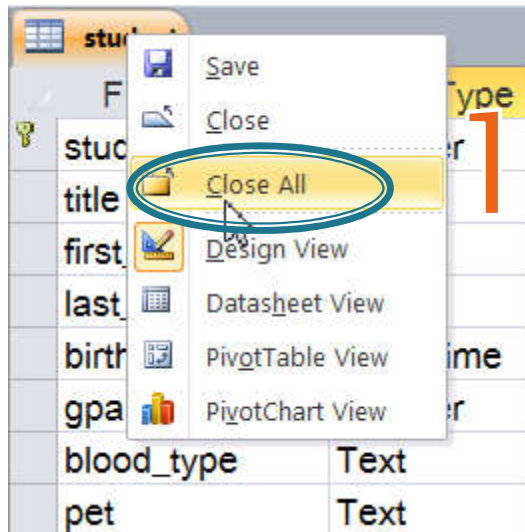
Yes (No Duplicates)

No

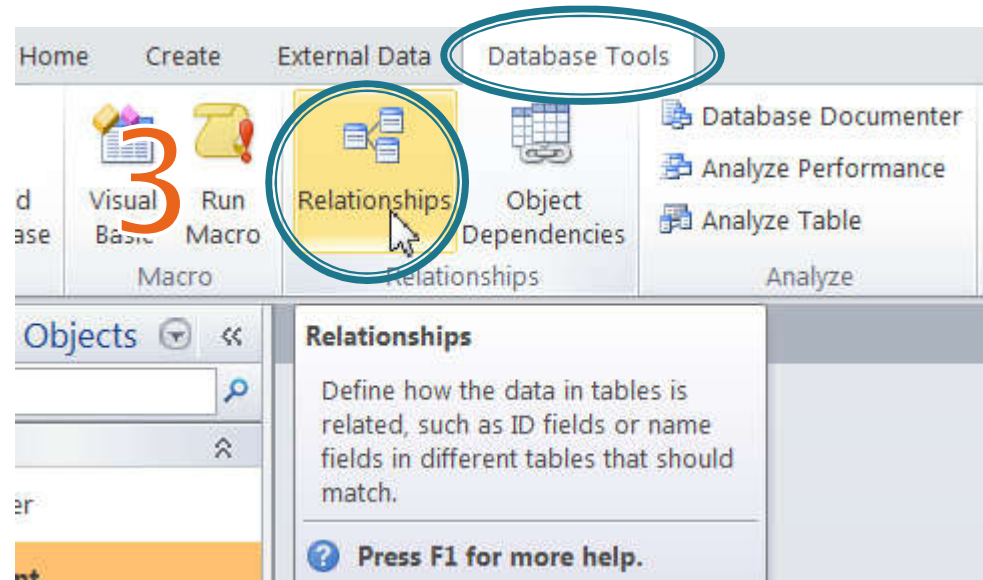
Yes (Duplicates OK)

Yes (No Duplicates)

Creating a Relationship [1.1]

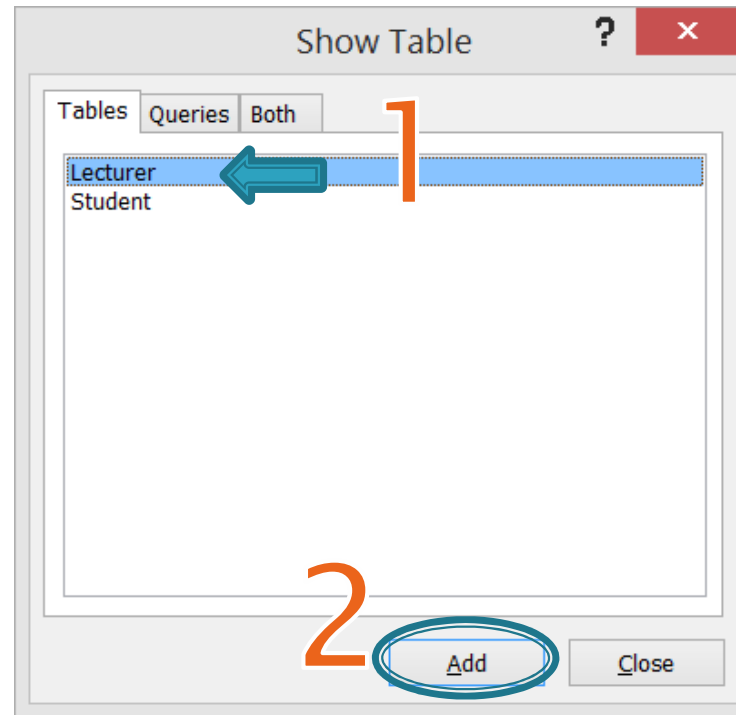


- ▶ Right Click -> Close All
- ▶ Database Tools
 - Relationships



Creating a Relationship [1.2]

- ▶ Adding the tables
 - Click on “Lecturer”
 - Add
 - Click on “Student”
 - Add
 - Close



Lecturer	
🔑	Lecturer ID
	Title
	First Name
	Last Name
	Office
	Phone Number
	E-mail Address

Student	
🔑	Student ID
	Title
	First Name
	Last Name
	Birth Date
	GPA
	Blood Type
	Pet
	Allowance
	E-mail Address
	Advisor ID

Creating a Relationship [1.3]

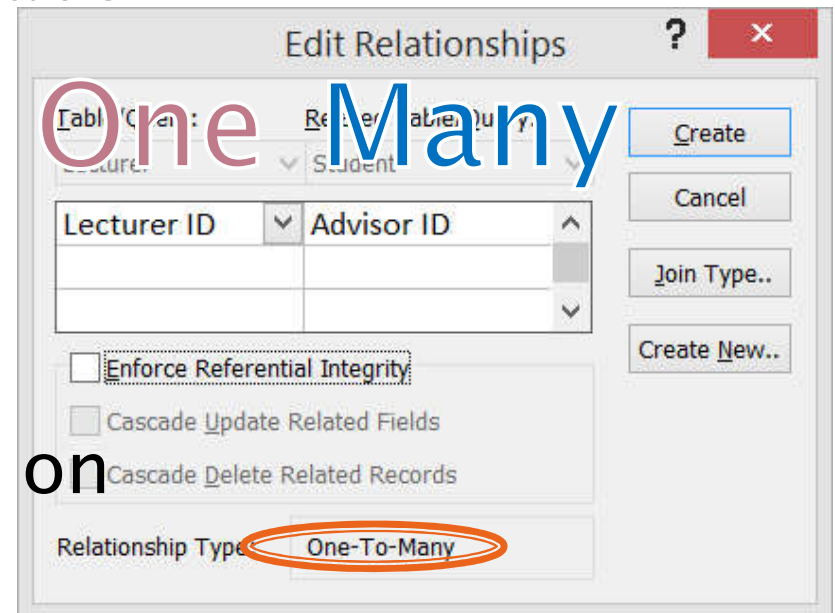
- ▶ Click on “**Advisor ID**” [Student]
 - Drag and drop on “**Lecturer ID**” [Lecturer]
- ▶ A Dialog Box “**Edit Relationship**” will show up

The screenshot shows the 'Edit Relationships' dialog box. It contains the following elements:

- Table/Query:** Lecturer
- Related Table/Query:** Student
- Primary Key:** Lecturer ID
- Foreign Key:** Advisor ID
- ☐ **Enforce Referential Integrity**
- ☐ **Cascade Update Related Fields**
- ☐ **Cascade Delete Related Records**
- Relationship Type:** One-To-Many
- Buttons:** Create, Cancel, Join Type.., Create New..

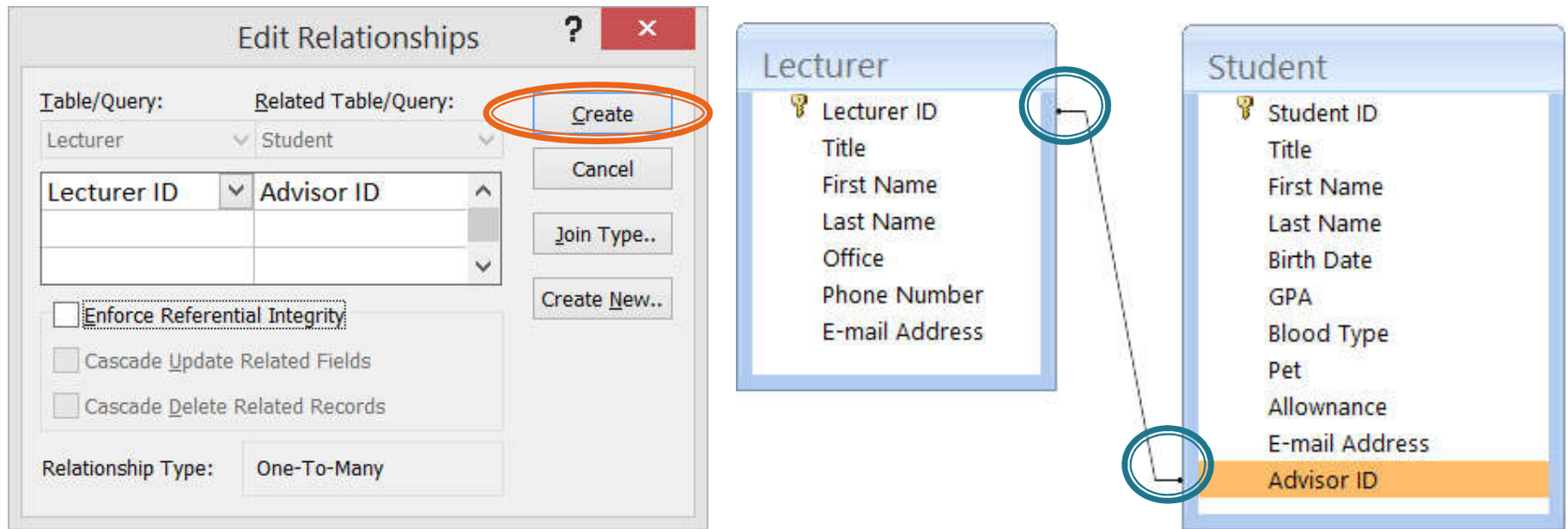
Creating a Relationship [1.4]

- ▶ Notice the “One-To-Many” label
 - Which table is on which side?
 - What if we reverse the dragging direction?
(lecturer_id -> advisor_id)
- ▶ The table on the “**One**” label of the relationship is always on the left (with the **PK**)
- ▶ The table on the “**Many**” label is always on the right (with the **FK**)



Creating a Relationship [1.5]

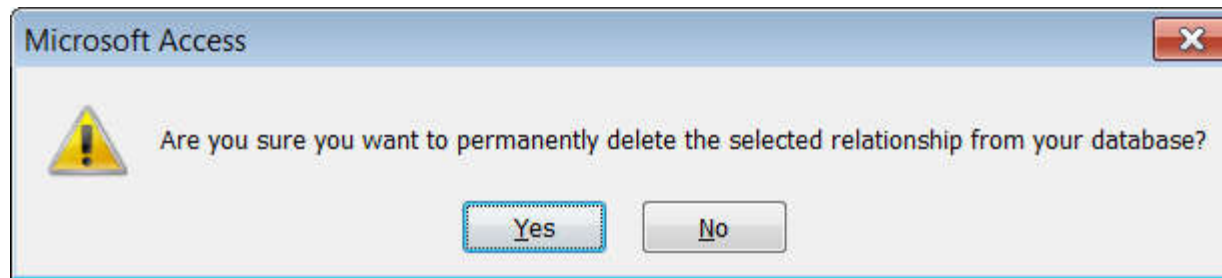
- ▶ Click “Create”



- ▶ Now we have a relationship
 - Notice that the 1:N (or 1:∞) is missing

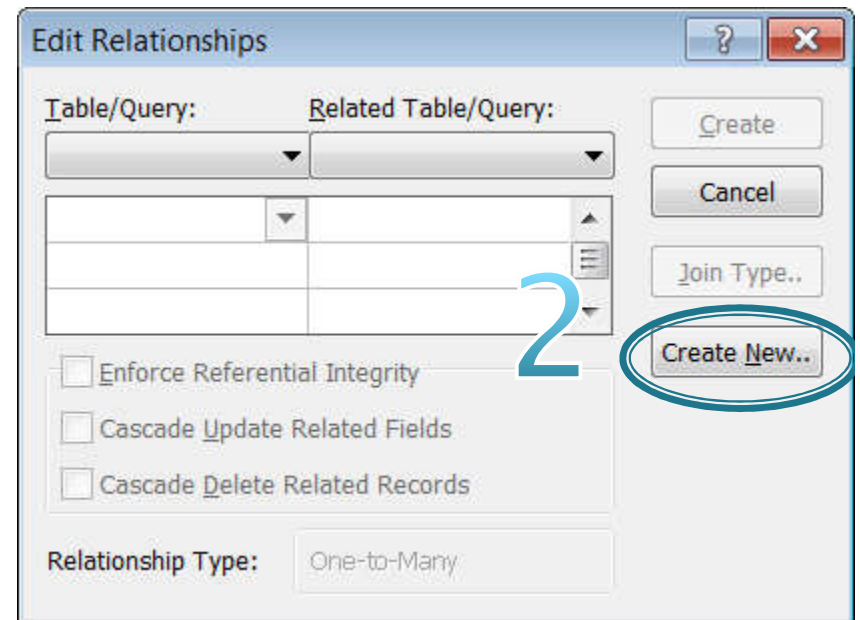
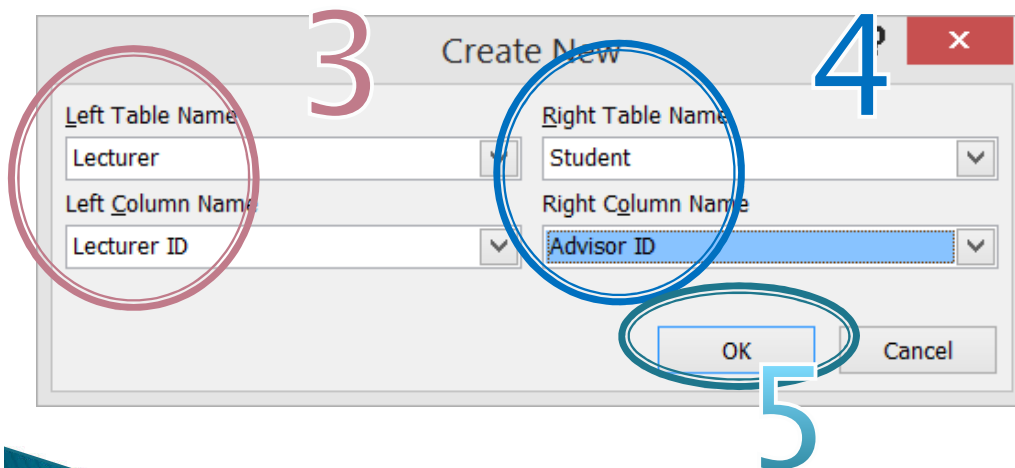
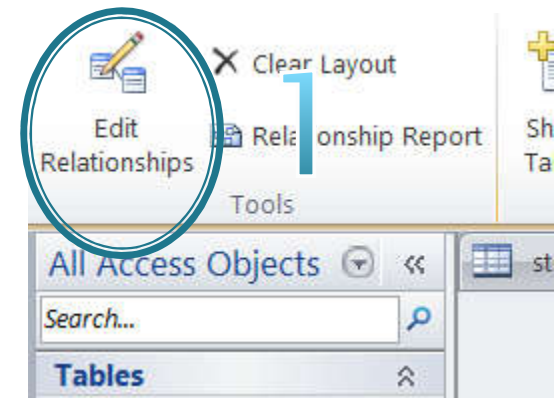
Removing a Relationship

- ▶ Click on the line and then “Delete”
 - Click “Yes”



Creating a Relationship [2.1]

- ▶ We can also create a Relationship by using the “Edit Relationship” button



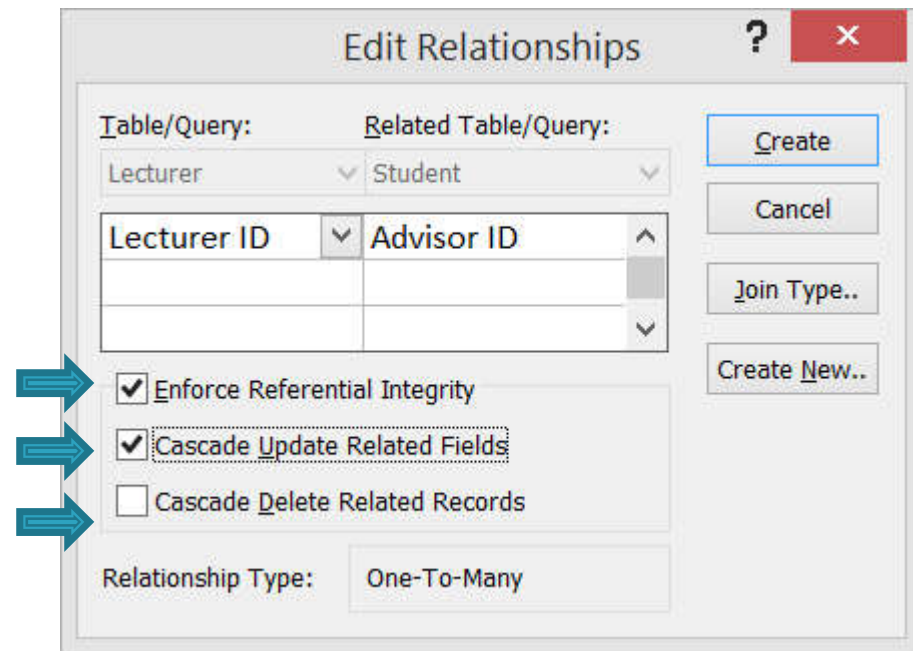
Creating a Relationship [2.2]

▶ Enforce Referential Integrity

- The value in FK field has to map to a valid PK in another table

▶ Cascade Update Related Field

- If **Lecturer ID** is changed all related records in the **Student** table will get updated.



• Cascade Delete Related Field

- If any record in the **Lecturer** table is deleted the related records in the **Student** table will also be **deleted**

Creating a Relationship [2.3]

► In this lab we check all 3 options

► Now we have Relationship 1:N (1:∞)

Edit Relationships

Table/Query: Lecturer Related Table/Query: Student

Lecturer ID Advisor ID

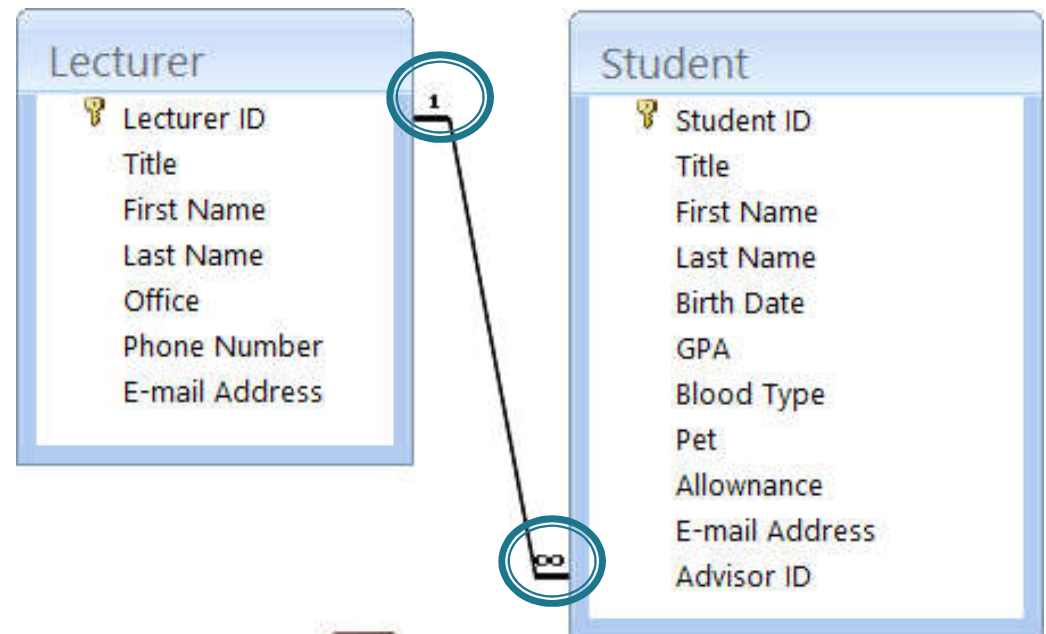
☒ Enforce Referential Integrity

☒ Cascade Update Related Fields

☒ Cascade Delete Related Records

Relationship Type: One-To-Many

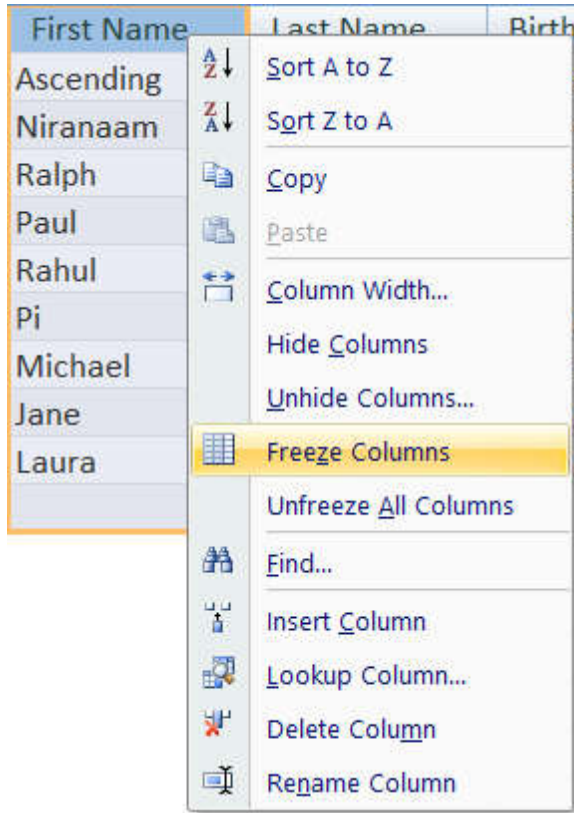
Buttons: OK, Cancel, Join Type.., Create New..



• Save and Close



Assigning Relationship [1]



First Name	Last Name	Birth
Ascending		
Niranaam		
Ralph		
Paul		
Rahul		
Pi		
Michael		
Jane		
Laura		

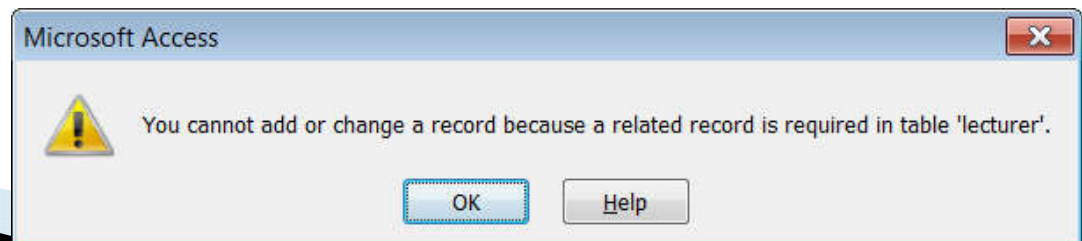
- Sort A to Z
- Sort Z to A
- Copy
- Paste
- Column Width...
- Hide Columns
- Unhide Columns...
- Freeze Columns**
- Unfreeze All Columns
- Find...
- Insert Column
- Lookup Column...
- Delete Column
- Rename Column

- ▶ Open the student table
 - Right Click on First Name
 - Freeze Columns
 - Then we move to Advisor ID

Assigning Relationship [2]

- ▶ Fill in the following data
- ▶ a dialog box saying “a related record is required” will show if entering wrong data
 - Save & Close

First Name	Advisor ID
Ascending	LEC23720
Niranaam	LEC23720
Ralph	LEC11348
Paul	LEC36231
Rahul	LEC66451
Pi	LEC11348
Michael	LEC11348
Jane	LEC36231
Laura	LEC36231



Assigning Relationship [3]

- ▶ In the lecturer table
 - By clicking on the plus sign (+) at the front of each record, the related records from the student table will unfold.

Student		Lecturer		
	Lecturer ID	Title	First Name	Last Name
+	LEC11348	Associate Professor	Brandon	Fowler
-	LEC23720	Professor	Noah	Howard
		First Name	Student ID	Title
		Niranaam	555555555	Mr.
		Ascending	500123456	Mr.
	*			
+	LEC36231	Lecturer	Florence	Allison
+	LEC66451	Assistant Professor	George	Palmer

Cascading Update

- ▶ Now change “LEC11348” to “LEC11349”

Lecturer	
Lecturer ID	Title
LEC11348	Associate Profes
LEC23720	Professor
LEC36231	Lecturer
LEC66451	Assistant Profess



LEC11349

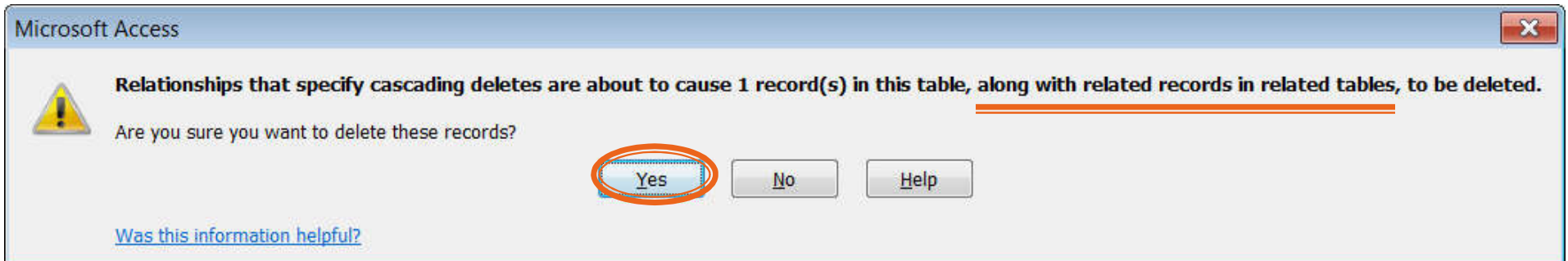
- After saving the table, the related records in the student table also change

Student		
Student ID	First Name	Advisor ID
500123456	Ascending	LEC23720
555555555	Niranaam	LEC23720
570883775	Ralph	LEC11349
570899998	Paul	LEC36231
571123456	Rahul	LEC66451
571314159	Pi	LEC11349
571634633	Michael	LEC11349
571645508	Jane	LEC36231
571888555	Laura	LEC36231

Cascading Delete [1]

- ▶ Delete the Record with PK = LEC66451
 - Notice the related **Student** Record (Rahul)

Lecturer		
	Lecturer ID	Title
+	LEC11349	Associate Pro
+	LEC23720	Professor
+	LEC36231	Lecturer
-	LEC66451	Assistant Prof
	Student ID	First Name
	571123456	Rahul
*		



along with **related records in related tables**, to be deleted.

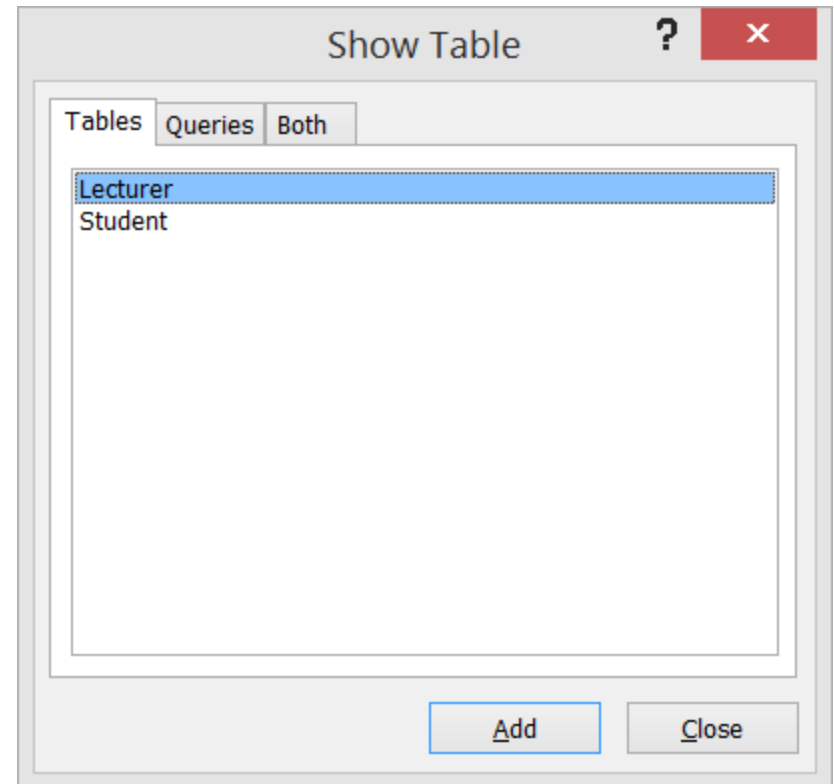
Cascading Delete [2]

- ▶ Now the related record is also deleted

Lecturer		Student	
Student ID	First Name	Advisor ID	
500123456	Ascending	LEC23720	
555555555	Niranaam	LEC23720	
570883775	Ralph	LEC11349	
570899998	Paul	LEC36231	
571314159	Pi	LEC11349	
571634633	Michael	LEC11349	
571645508	Jane	LEC36231	
571888555	Laura	LEC36231	


Creating a Query

- ▶ To create a query: at the **Show Table** dialog box, add only the tables related to the query.
 - If the query involves data in **Student** table only without referring to the **Lecturer** table, add only the **Student** table



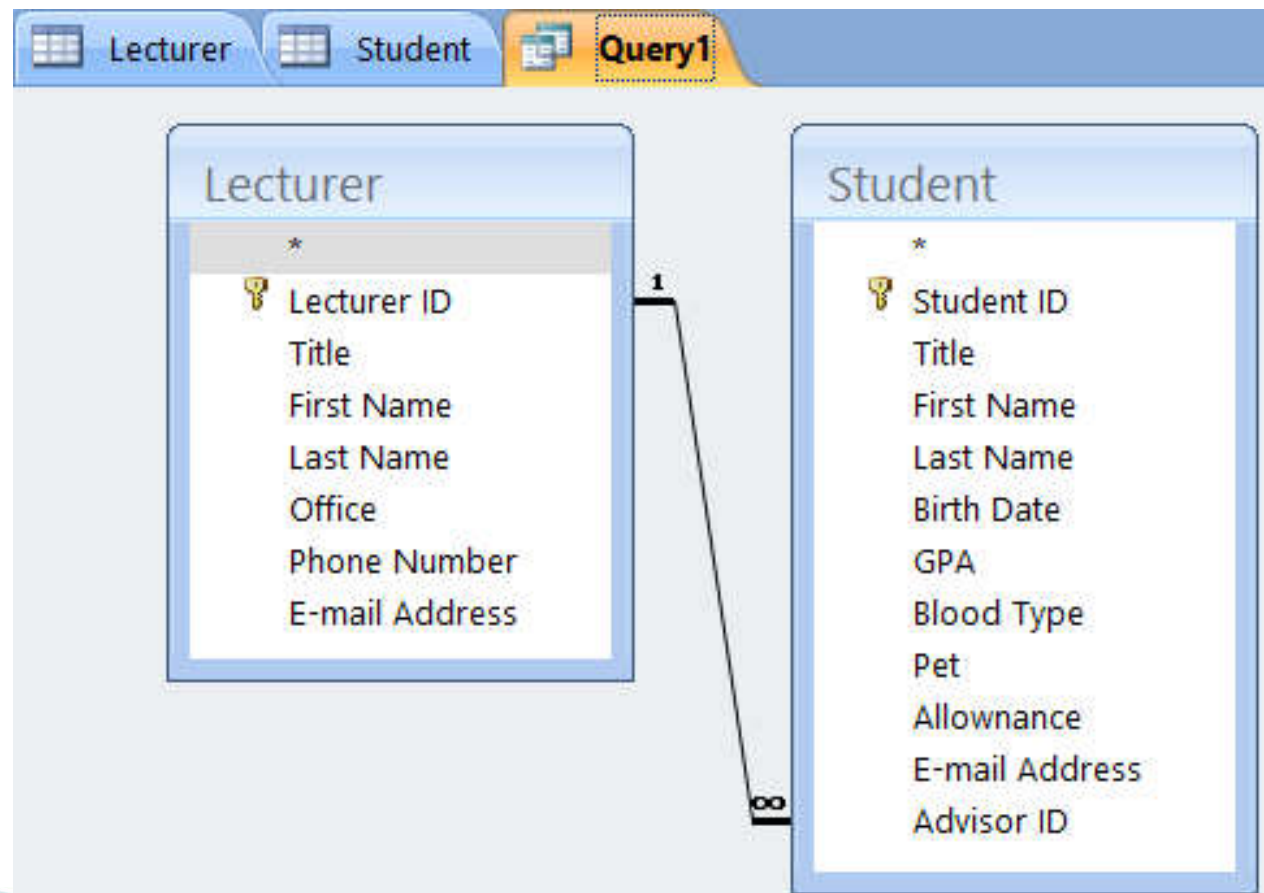
Query Practice V

- ▶ Create a query to find all female students and male students from the Faculty of Law(20)
 - Showing
 - Names, student ID, and GPA
 - Sorted by GPA (ascending)
 - Need wildcard (? and *)

Field:	First Name	Last Name	Student ID	GPA
Table:	Student	Student	Student	Student
Sort:				Ascending 
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:			Like "??20*"	
or:				


Creating a Query [2]

- ▶ If querying involves data in the lecturer table, we need to add the **Lecturer** table also.



Query Practice VI

- ▶ Create a query to list the name of students whose advisors are professors
 - Showing student ID, name, and GPA
 - Sorted by last name – descending

Field:	Student ID	First Name	Last Name	GPA	Title
Table:	Student	Student	Student	Student	Lecturer
Sort:			Descending 		
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Criteria:					"Professor"
or:					

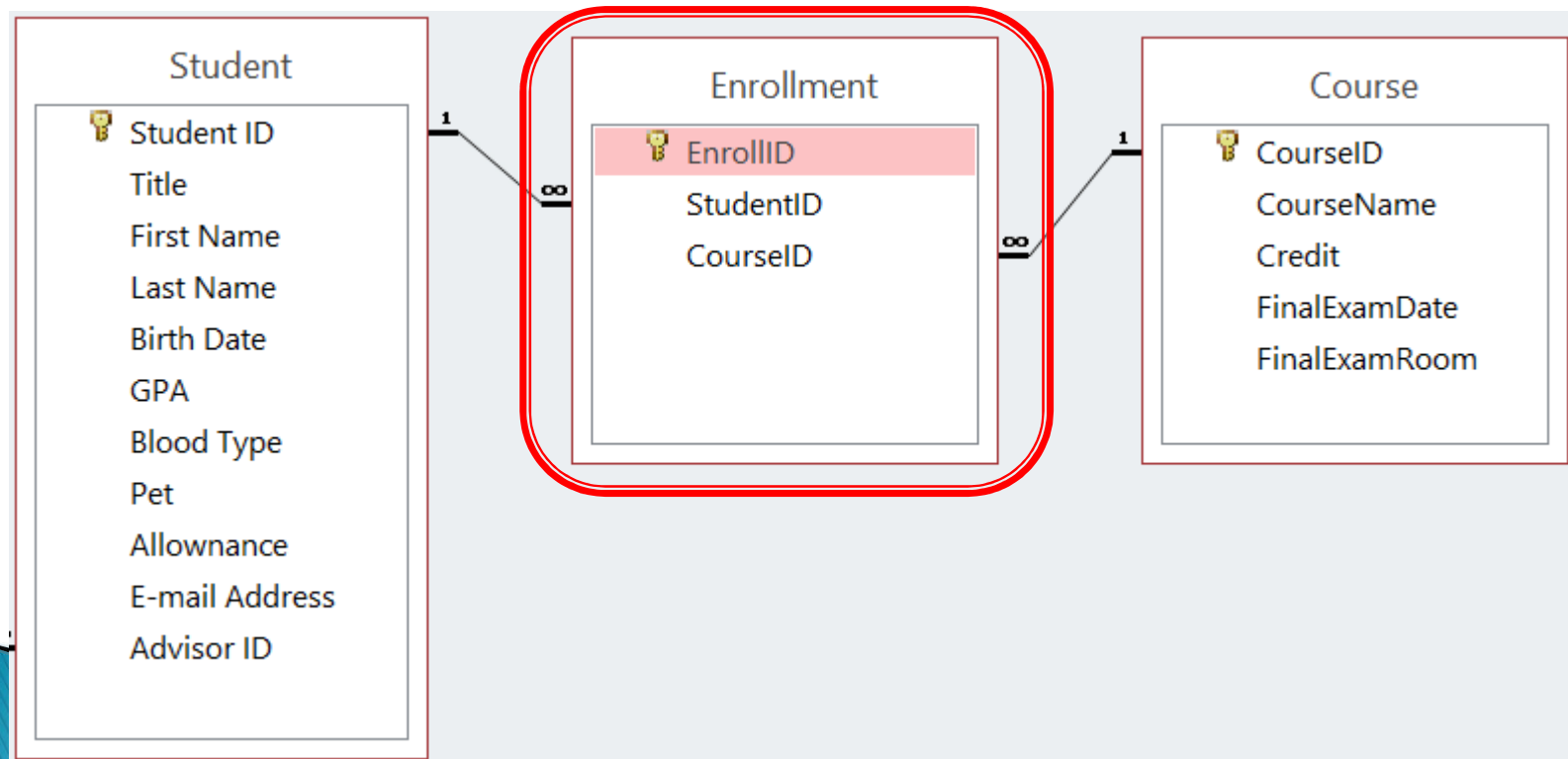
Query Practice VII

- ▶ Create a query for students with GPA higher than 2.75 and having advisors w/ office in building 4 (Office:A4XX)
 - Showing (Students') Name, and the advisors' office number and GPA
 - Remember wildcard? (* and ?)

Field:	First Name	Last Name	Office	GPA
Table:	Student	Student	Lecturer	Student
Sort:				
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:			Like "A4??"	>2.75
or:				

Many-to-Many Relationship

- ▶ Need to create an intermediate table to handle the relationship
- ▶ In *13_Students_Class_rel.accdb*:



Summary

- ▶ Creating a Relationship
 - Where to put foreign key?
- ▶ Removing a Relationship
- ▶ Enforcing Referential Integrity
- ▶ Cascading Update/Delete
- ▶ Querying on related tables

