



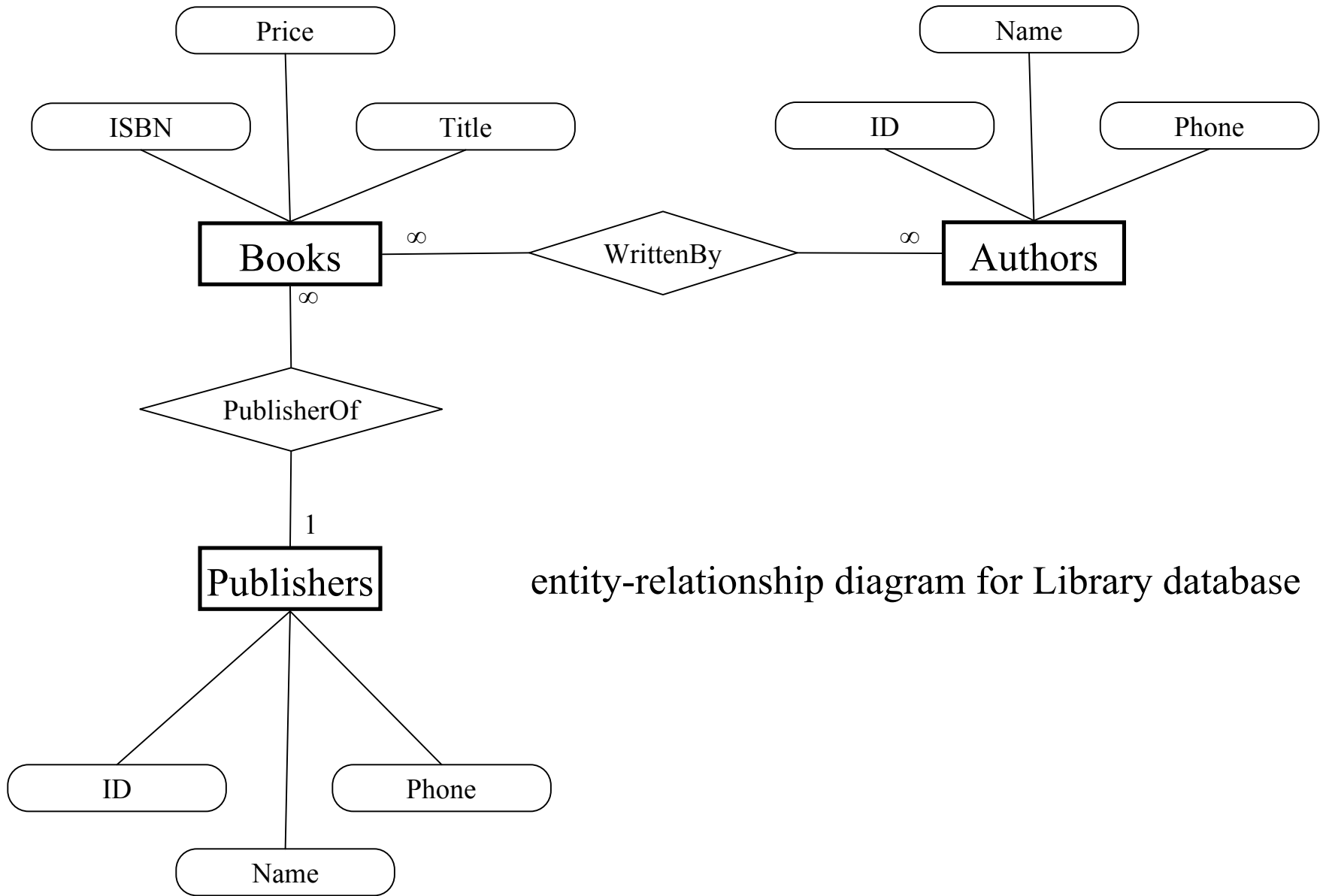
More Views

INFO/CSE 100, Spring 2005
Fluency in Information Technology

<http://www.cs.washington.edu/100>

Readings and References

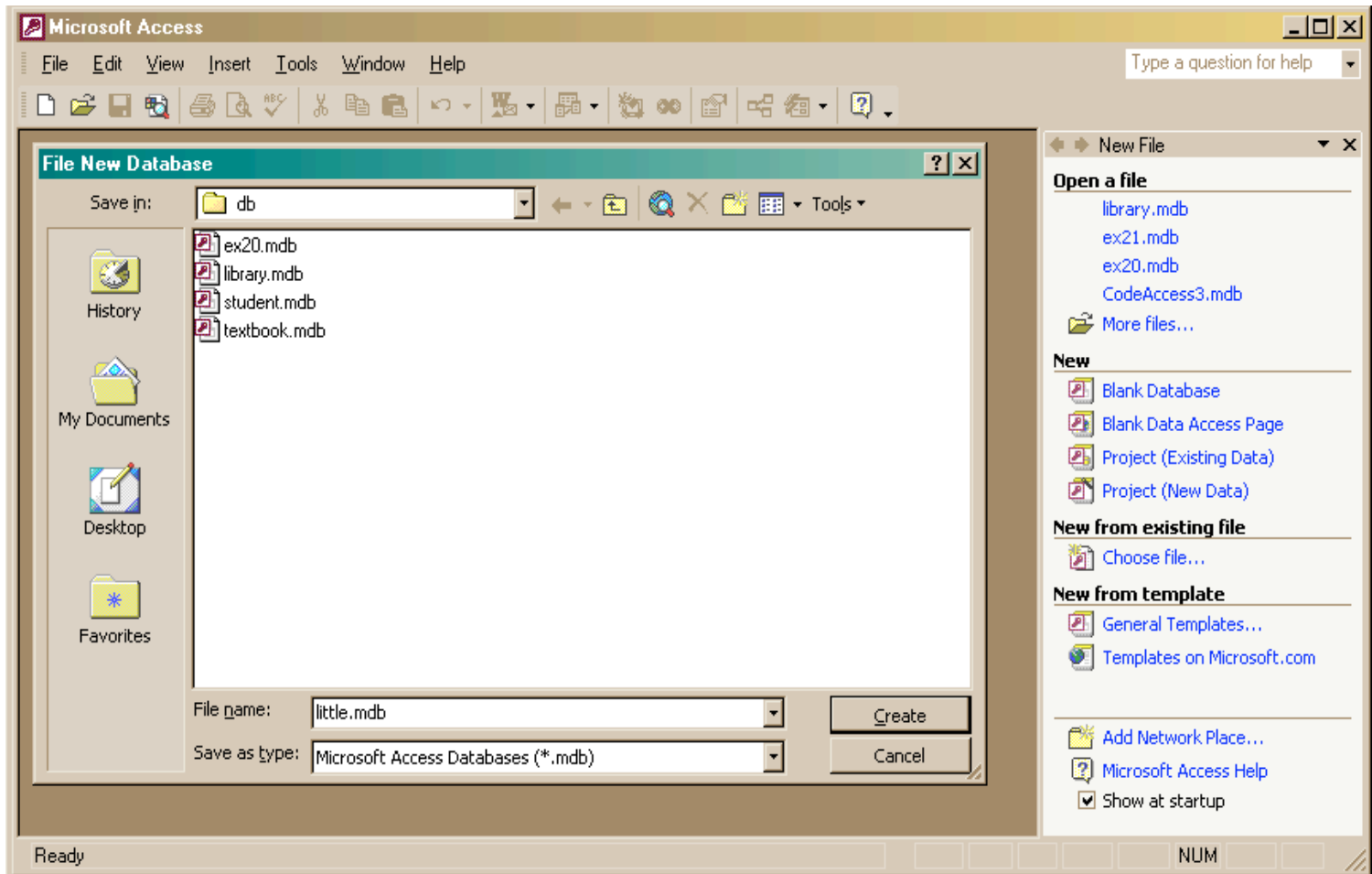
- Reading
 - » *Fluency with Information Technology*
 - Chapter 14, Database Queries
- References
 - » *Access Database: Design and Programming*
 - by Steve Roman, published by O'Reilly



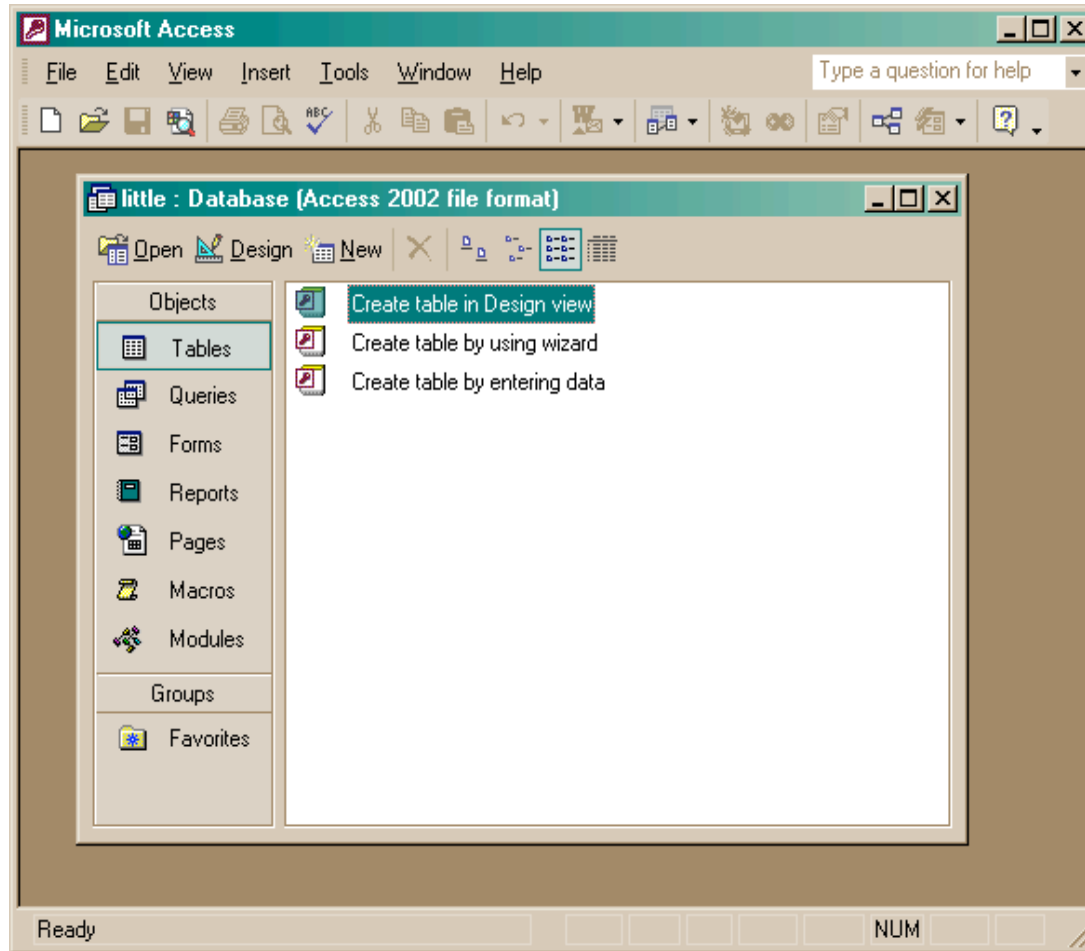
entity-relationship diagram for Library database



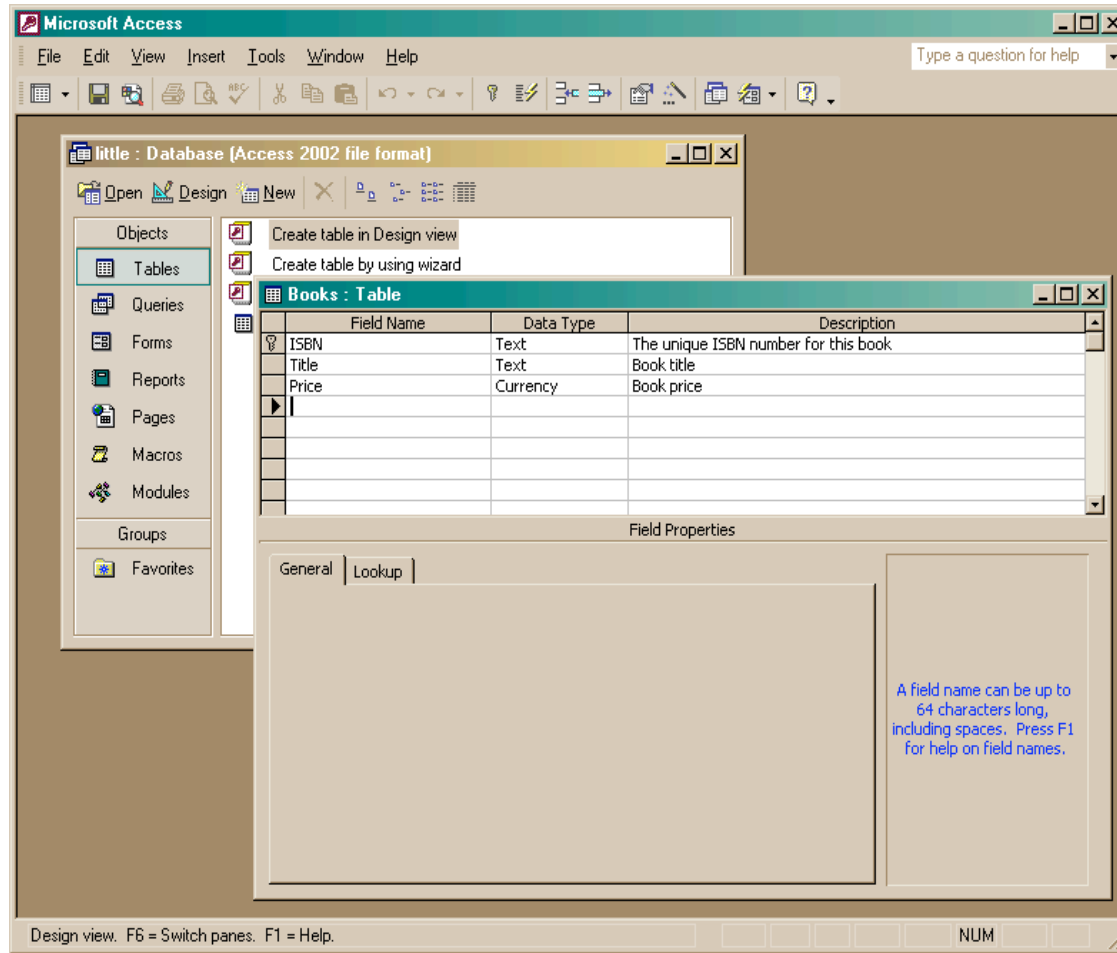
Create a new database



Create a new table in the database

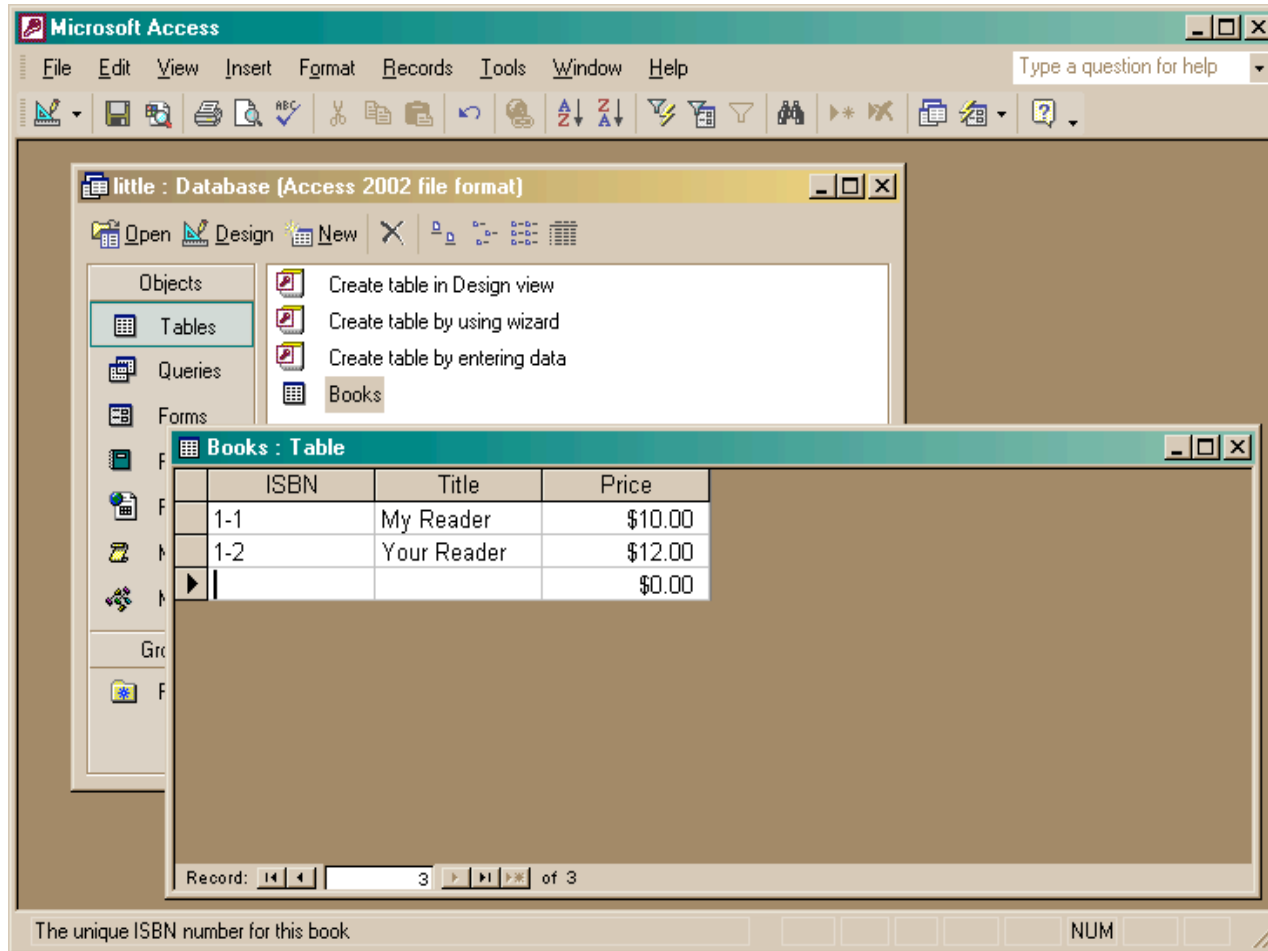


Creating a table in Design view



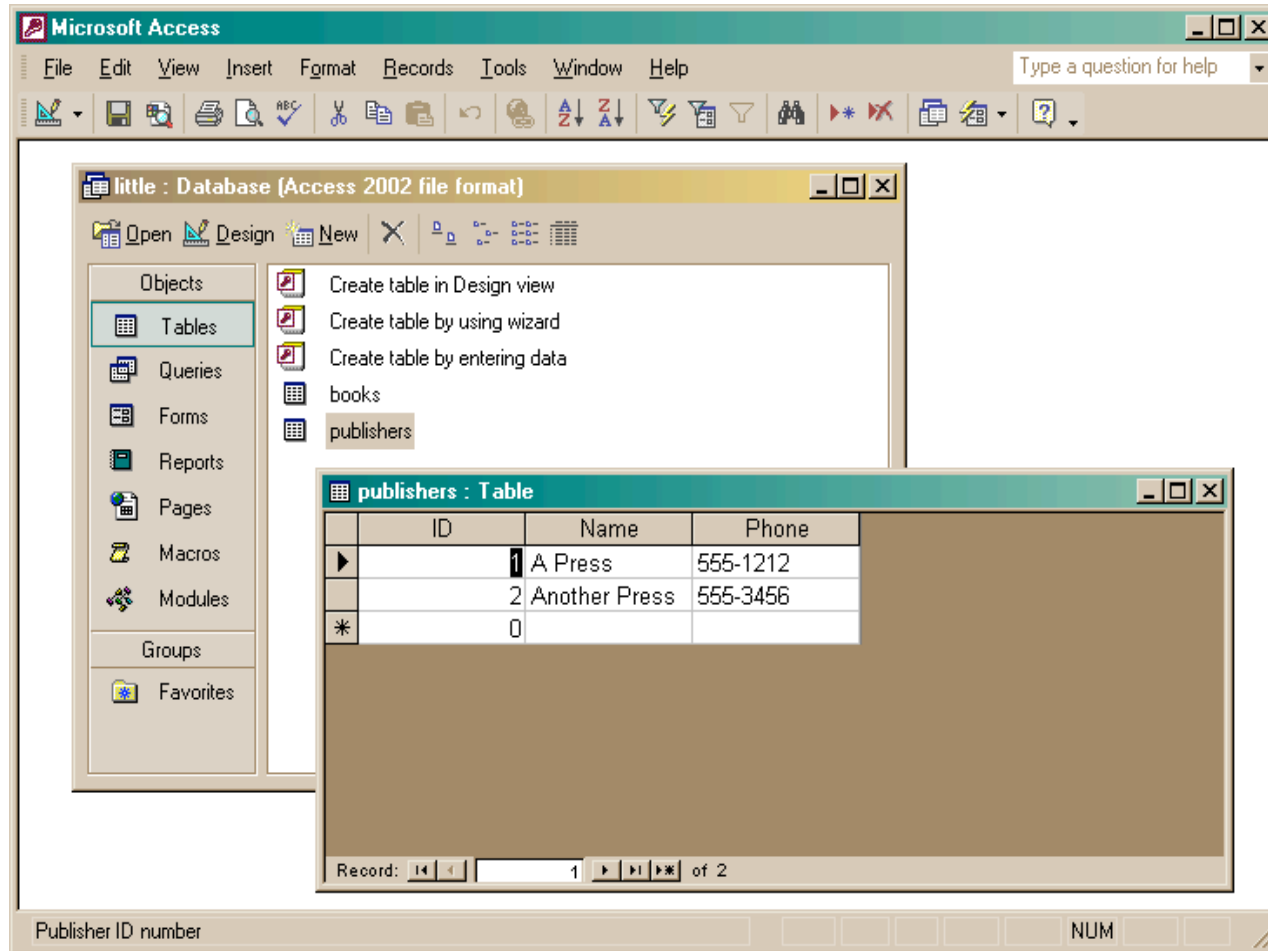


Entering Table Data

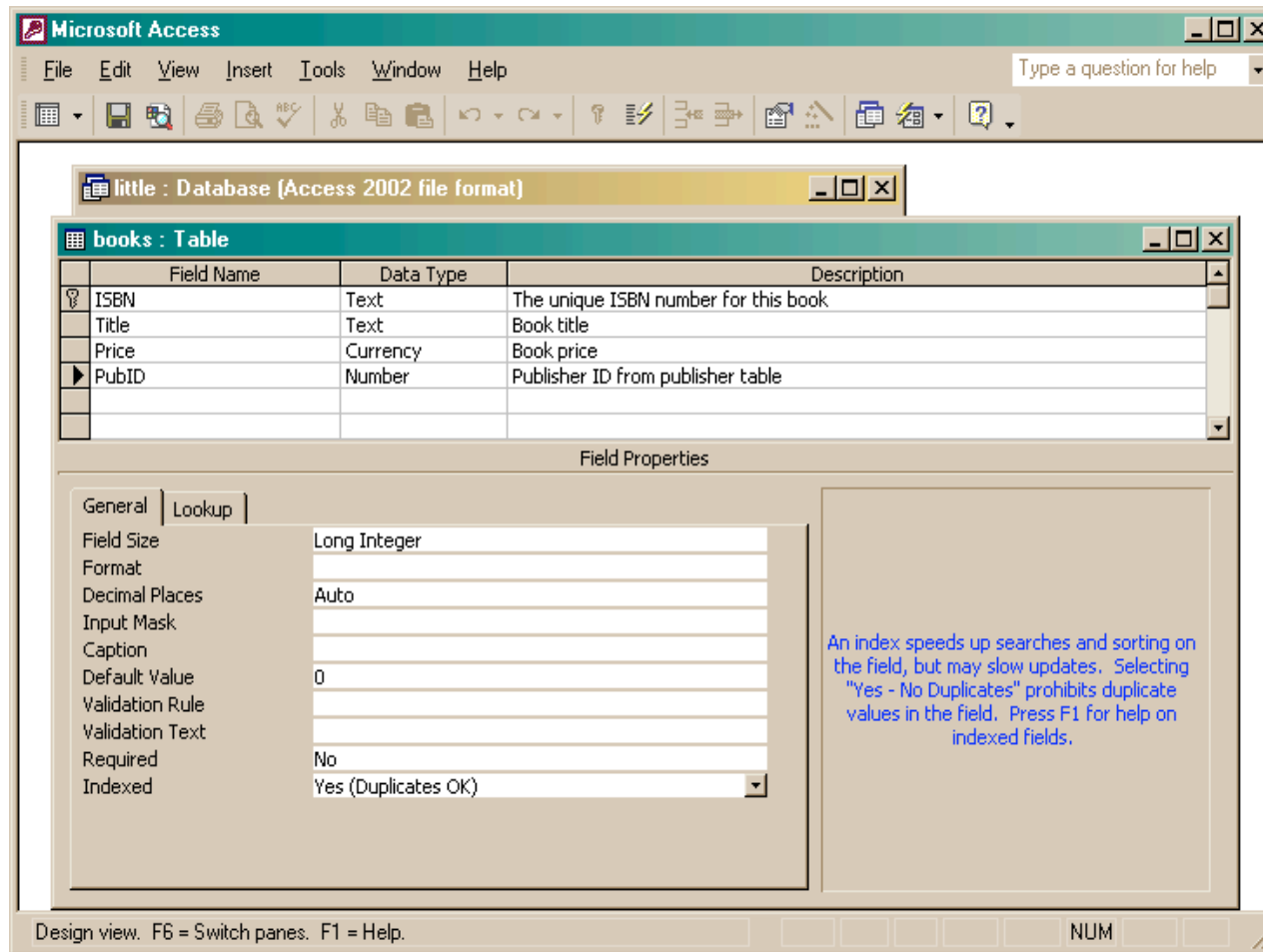




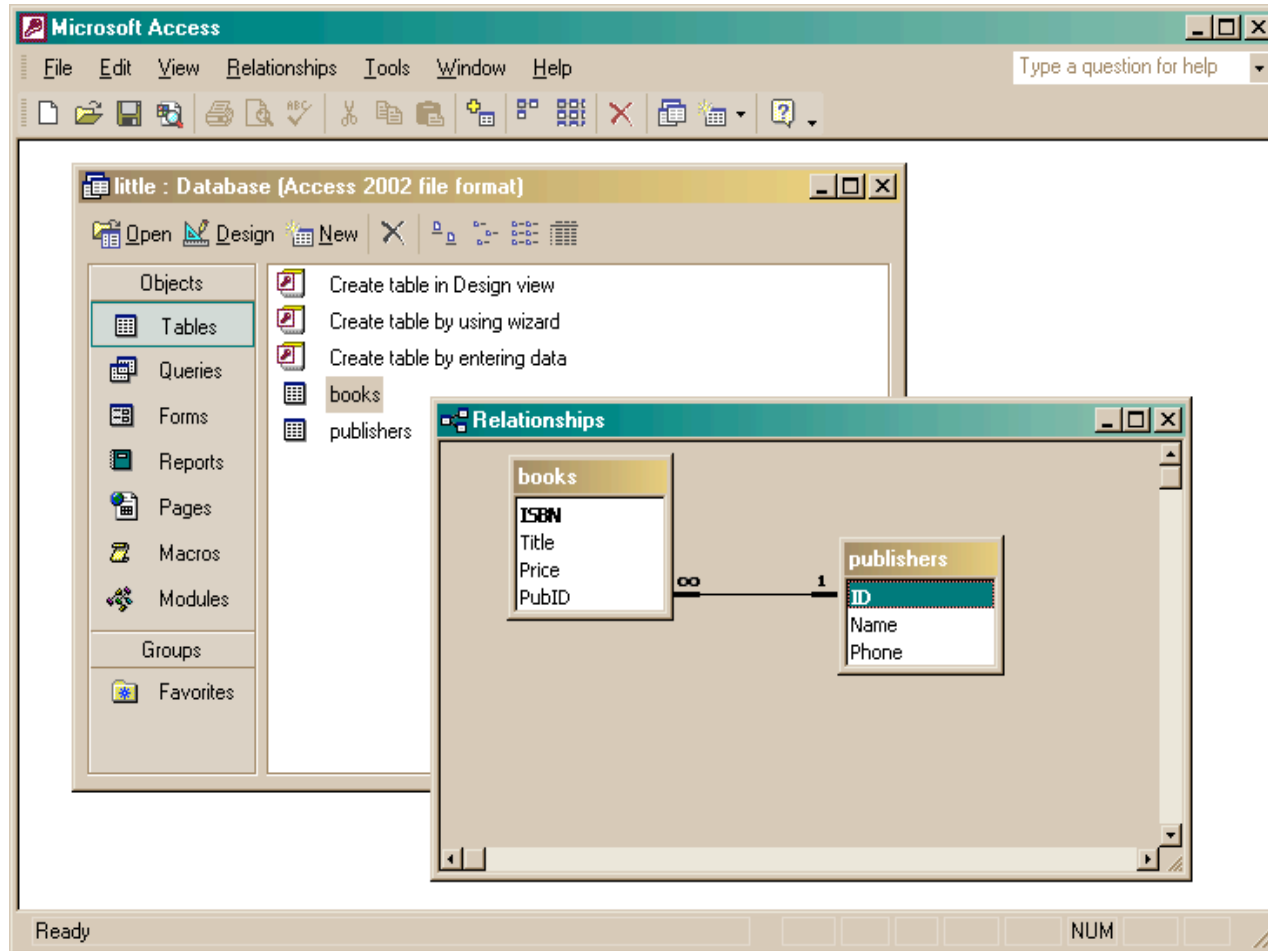
Build another table

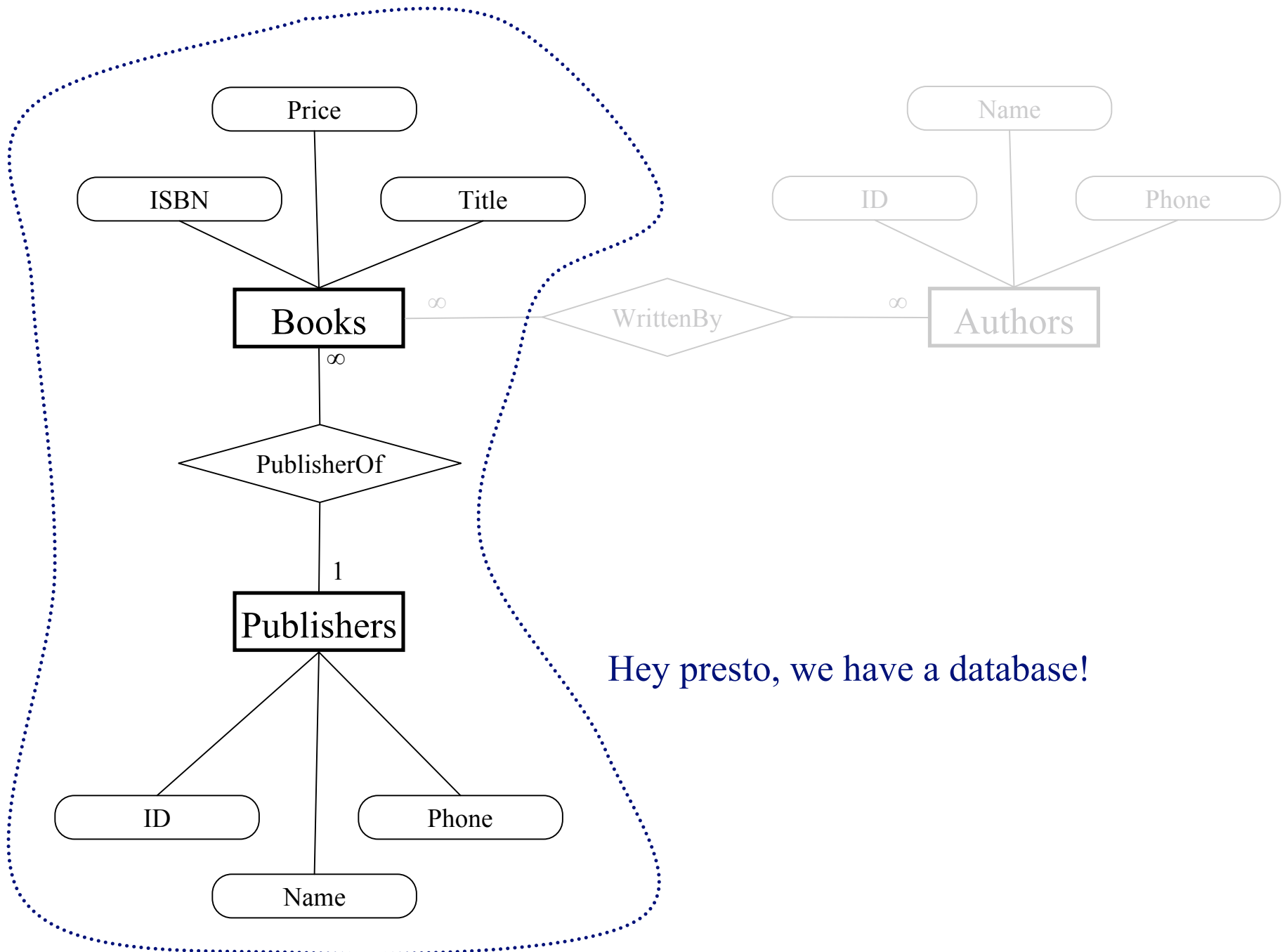


Add publisher ID to books



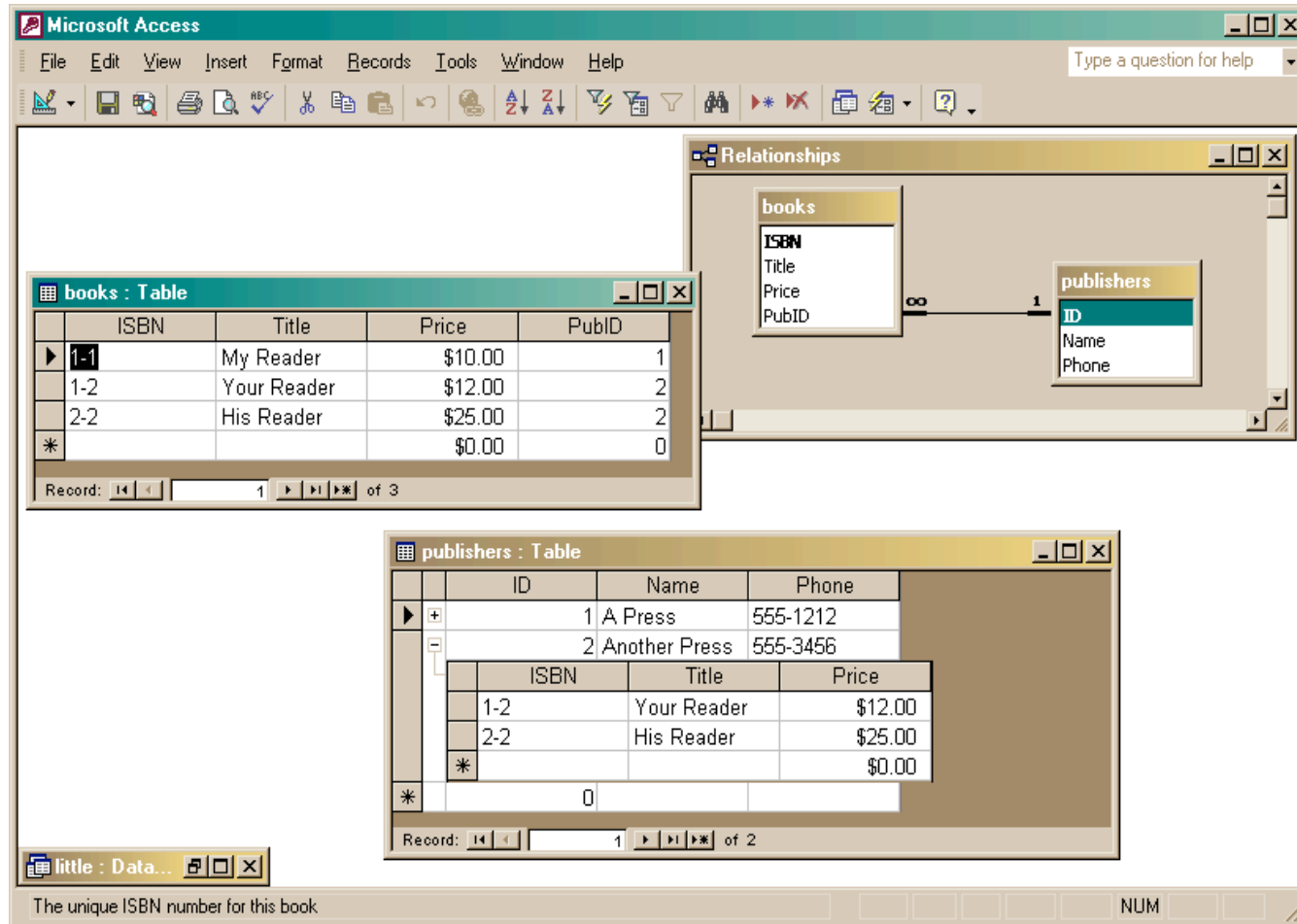
Create the link between the tables





Hey presto, we have a database!

Two tables with a relationship



The screenshot shows Microsoft Access with three windows open:

- books : Table**

	ISBN	Title	Price	PubID
▶ 1-1		My Reader	\$10.00	1
1-2		Your Reader	\$12.00	2
2-2		His Reader	\$25.00	2
*			\$0.00	0
- publishers : Table**

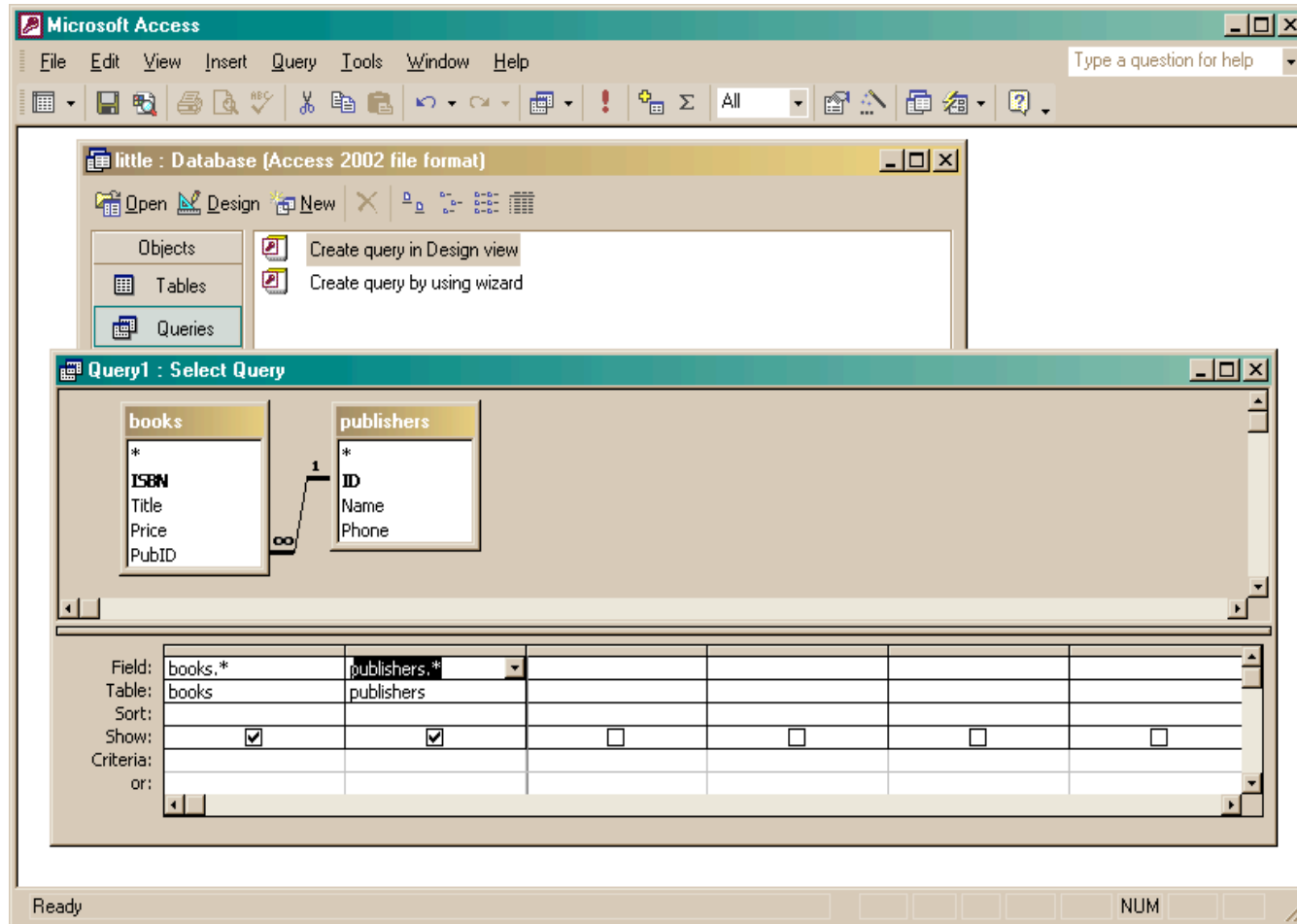
	ID	Name	Phone
▶ +	1	A Press	555-1212
-	2	Another Press	555-3456
	ISBN	Title	Price
	1-2	Your Reader	\$12.00
	2-2	His Reader	\$25.00
*			\$0.00
*			0
- Relationships**

Diagram showing a one-to-many relationship between 'publishers' (ID) and 'books' (PubID). The 'publishers' side has a '1' and the 'books' side has an '∞'.

At the bottom, a status bar shows: "The unique ISBN number for this book NUM"



Create a query





The query produces a new (virtual) table

The screenshot shows the Microsoft Access interface. The main window displays a query named "All Books w/ all fields : Select Query". The query results are shown in a table with the following data:

ISBN	Title	Price	PubID	ID	Name	Phone
1-1	My Reader	\$10.00	1	1	A Press	555-1212
1-2	Your Reader	\$12.00	2	2	Another Press	555-3456
2-2	His Reader	\$25.00	2	2	Another Press	555-3456
*						

At the bottom of the window, there is a status bar with the text "The unique ISBN number for this book" and a field labeled "NUM".



Project (select particular columns)

The screenshot shows Microsoft Access with a database named 'little : Database (Access 2002 file format)'. The 'Queries' pane on the left shows a query named 'Title & Publisher'. The main window displays the query design view for 'Title & Publisher : Select Query', showing a relationship between the 'books' table (with fields ISBN, Title, Price, PubID) and the 'publishers' table (with fields ID, Name, Phone). A one-to-many relationship is indicated between the 'books' table and the 'publishers' table.

Below the design view is a table showing the query criteria:

Field:	ISBN	Title	Name
Table:	books	books	publishers
Sort:			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:			
or:			

Overlaid on the right is a data view window titled 'Title & Publisher : Select Query' showing the following data:

	ISBN	Title	Name
▶ 1-1		My Reader	A Press
	1-2	Your Reader	Another Press
	2-2	His Reader	Another Press
*			

Record: 1 of 3



Select particular rows

The screenshot shows Microsoft Access with a database named 'little : Database (Access 2002 file format)'. The 'Queries' pane on the left shows a query named 'Costly books'. The main window displays the 'Costly books : Select Query' design view, which is linked to two tables: 'books' and 'publishers'. The 'books' table has fields: ISBN, Title, Price, and PubID. The 'publishers' table has fields: ID, Name, and Phone. A relationship line connects the 'PubID' field in 'books' to the 'ID' field in 'publishers'.

Below the design view is a data grid with the following fields and criteria:

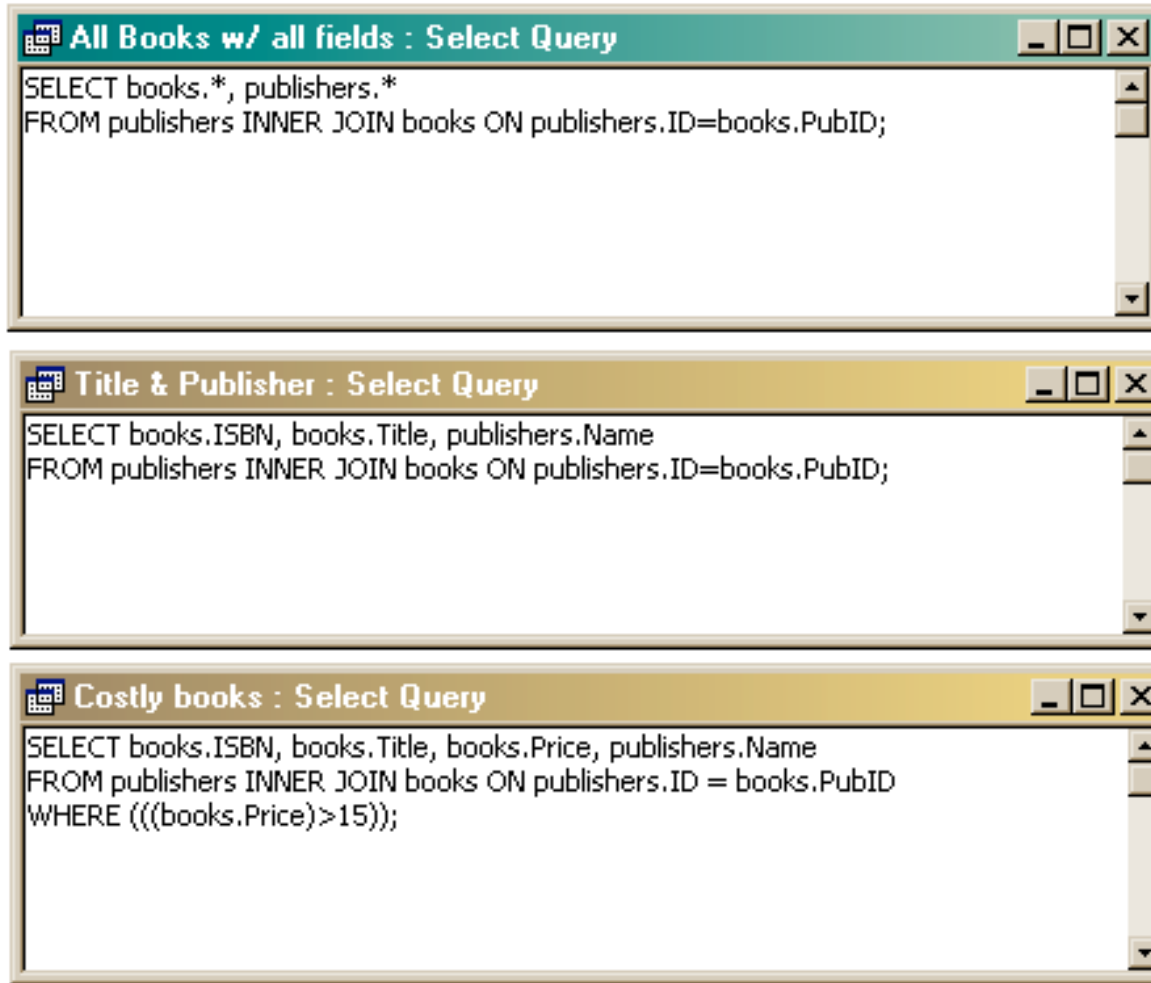
Field:	ISBN	Title	Price	Name	
Table:	books	books	books	publishers	
Sort:					
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Criteria:			>15		
or:					

An inset window titled 'Costly books : Select Query' shows the data view of the query. It displays a table with the following data:

ISBN	Title	Price	Name
2-2	His Reader	\$25.00	Another Press

The status bar at the bottom of the inset window shows 'Record: 2 of 2'.

SQL behind the scenes



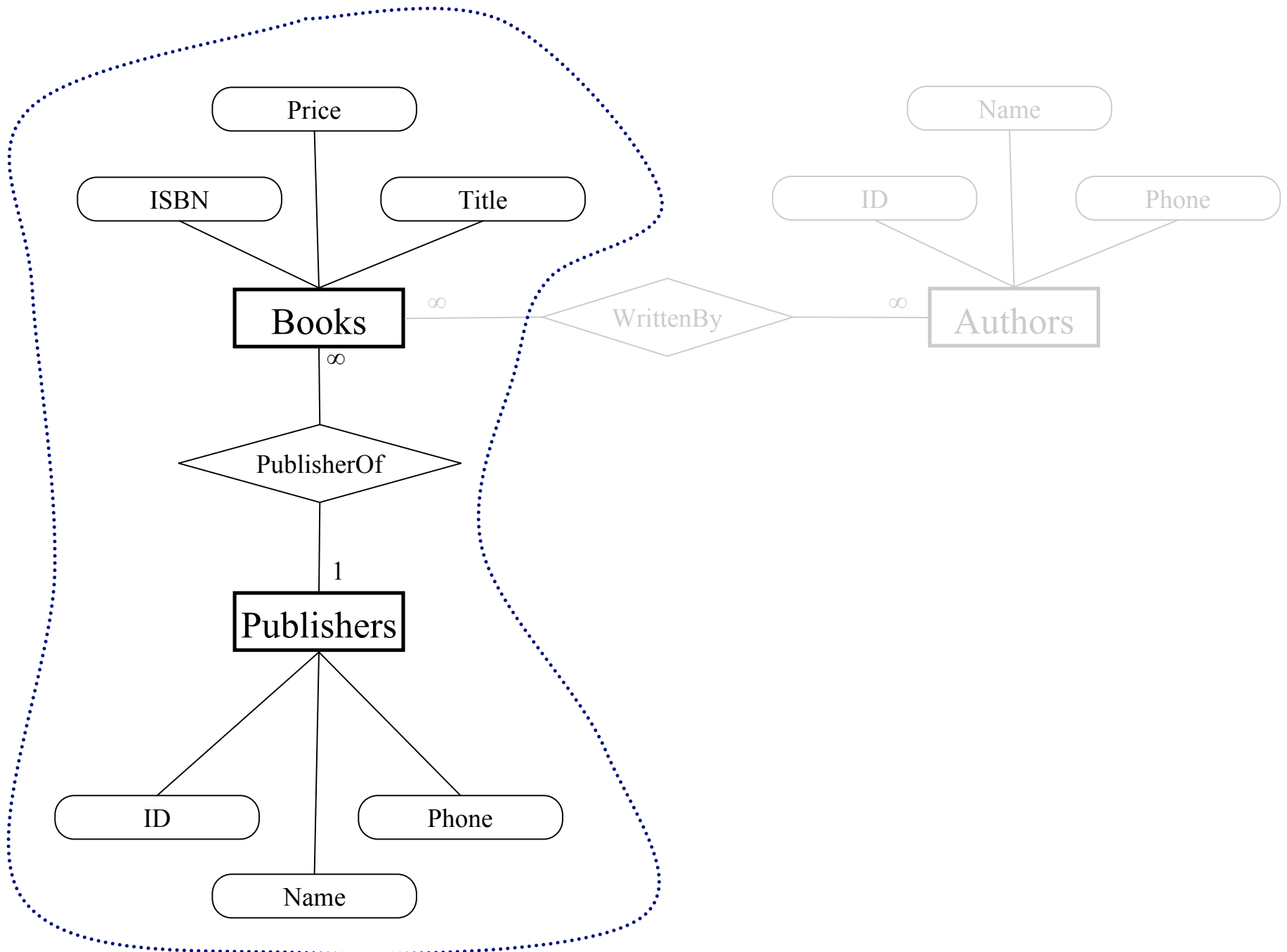
```
All Books w/ all fields : Select Query
SELECT books.*, publishers.*
FROM publishers INNER JOIN books ON publishers.ID=books.PubID;

Title & Publisher : Select Query
SELECT books.ISBN, books.Title, publishers.Name
FROM publishers INNER JOIN books ON publishers.ID=books.PubID;

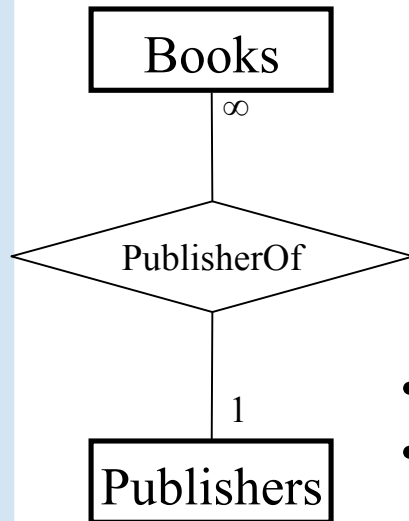
Costly books : Select Query
SELECT books.ISBN, books.Title, books.Price, publishers.Name
FROM publishers INNER JOIN books ON publishers.ID = books.PubID
WHERE (((books.Price)>15));
```

Recall: Structure of the database

- A database contains one or more *tables*
 - » Tables include *entities* with *attributes*
 - » There are *relationships* defined between the entities in the various tables
 - » Retrieve information from the tables using *queries*
- We designed and partially implemented a simple library database in the previous lecture



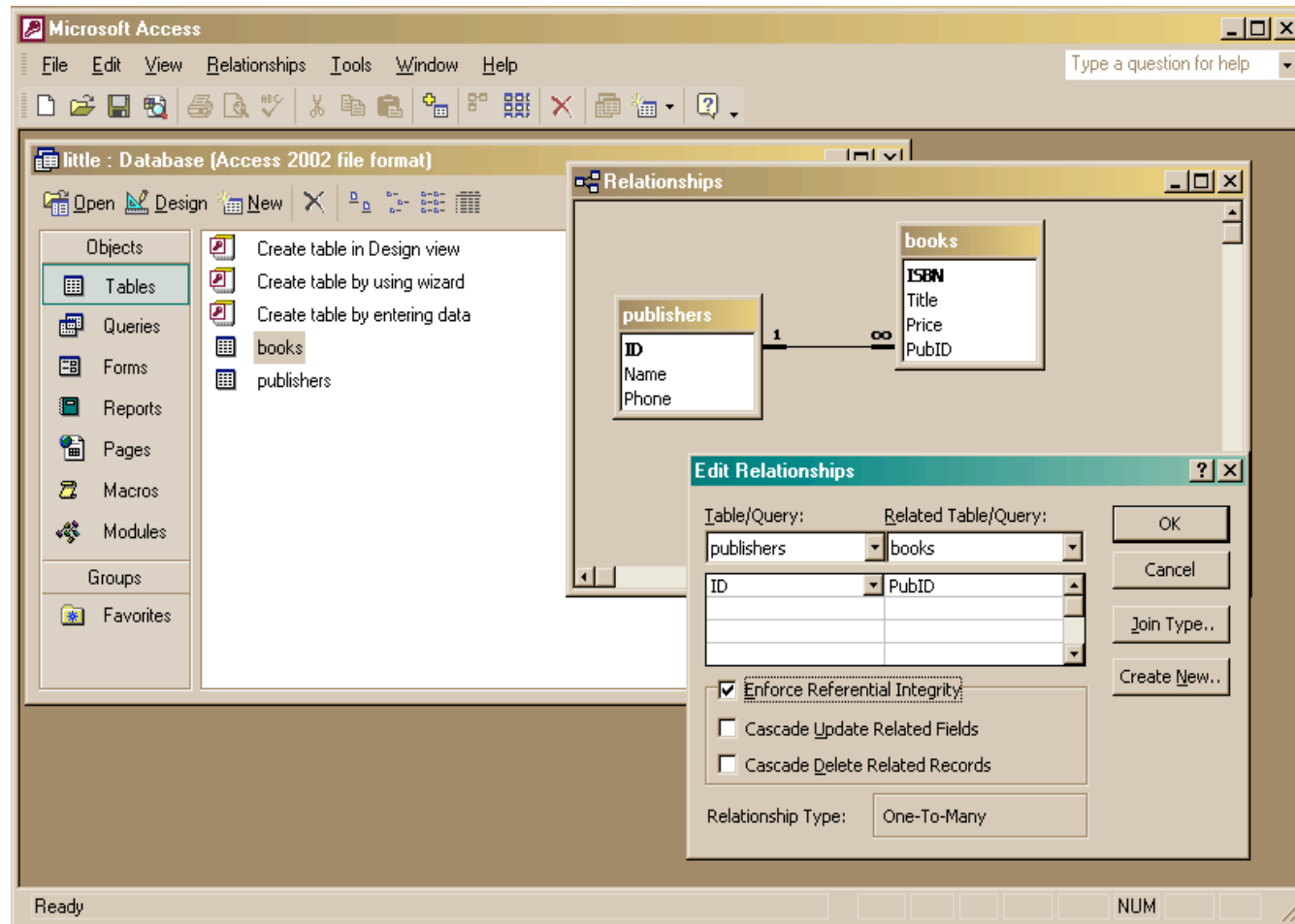
What is the relationship?



This relationship is 1-to-many:

- One publisher is responsible for many books.
 - Each book has only one publisher.
-
- The two tables are joined using the publisher ID number.
 - The publisher ID is the *primary key* for each entry in the publishers table.
 - Therefore, each publisher must have a unique publisher ID.
 - The publisher ID is a *foreign key* for each entry in the books table and we have requested *referential integrity*
 - Therefore, the given publisher ID must exist in the publishers table.

Referential Integrity



PubID must reference an actual publisher

All Books w/ all fields : Select Query

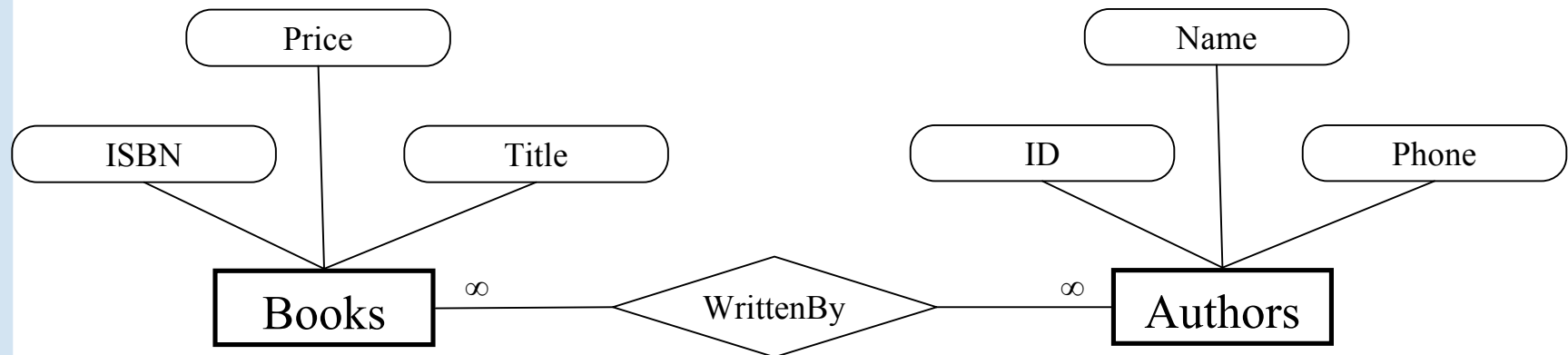
	ISBN	Title	Price	PubID	ID	Name	Phone
▶	1	My Reader	\$10.00	1	1	A Press	555-1212
	1-2	Your Reader	\$12.00	2	2	Another Press	555-3456
	2-2	His Reader	\$25.00	2	2	Another Press	555-3456
*							

Record: 1 of 3

All Books w/ all fields : Select Query

```
SELECT books.*, publishers.*
FROM publishers INNER JOIN books ON publishers.ID=books.PubID;
```

What is the relationship?

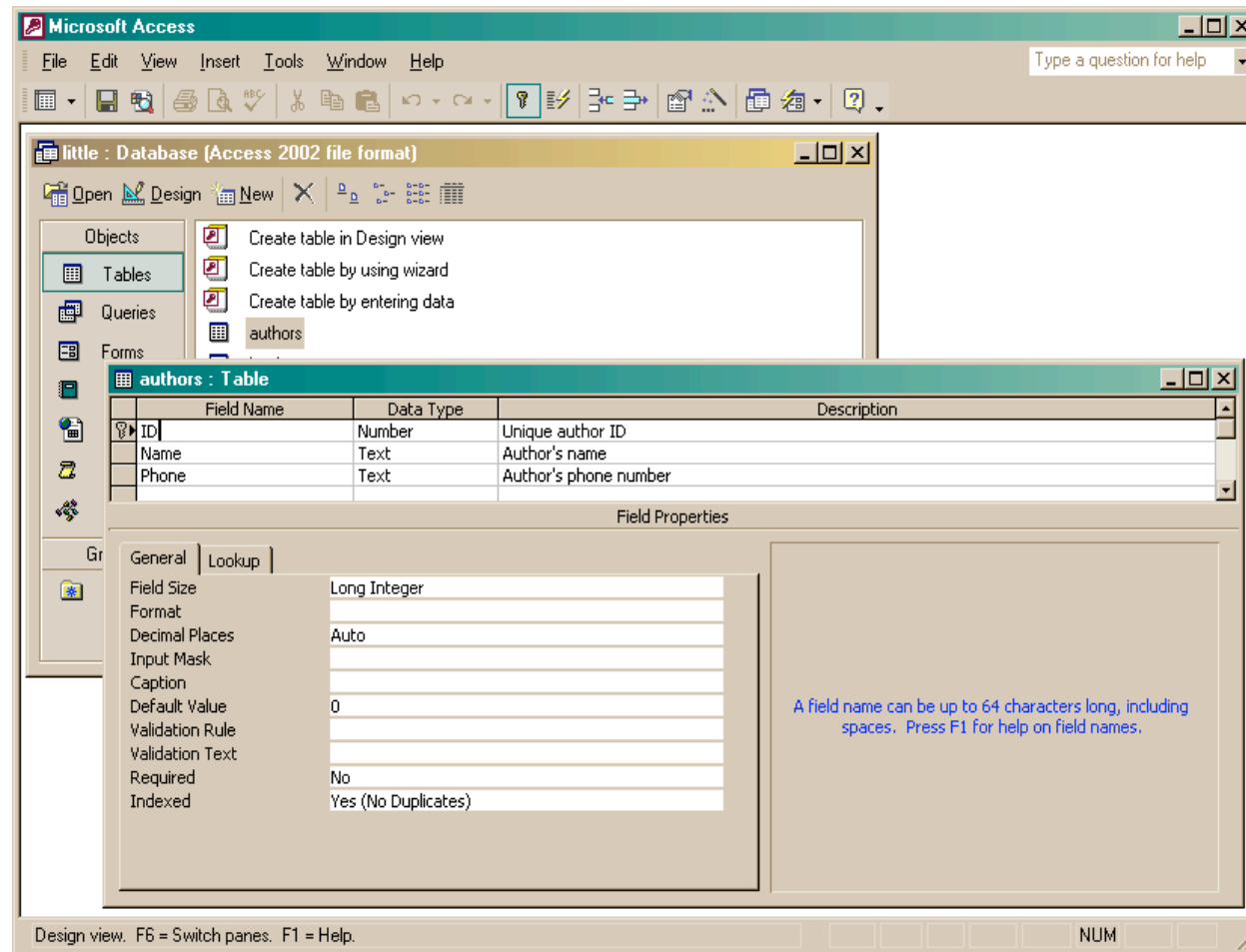


This relationship is many-to-many:

- One book may have several authors.
 - One author may have written several books.
-
- We need a unique identifier for each book.
 - We already selected the ISBN as the primary key and asked Access to make sure that there are no duplicates
 - We need a unique identifier for each author
 - We will define an author table with a unique ID for each author



authors table



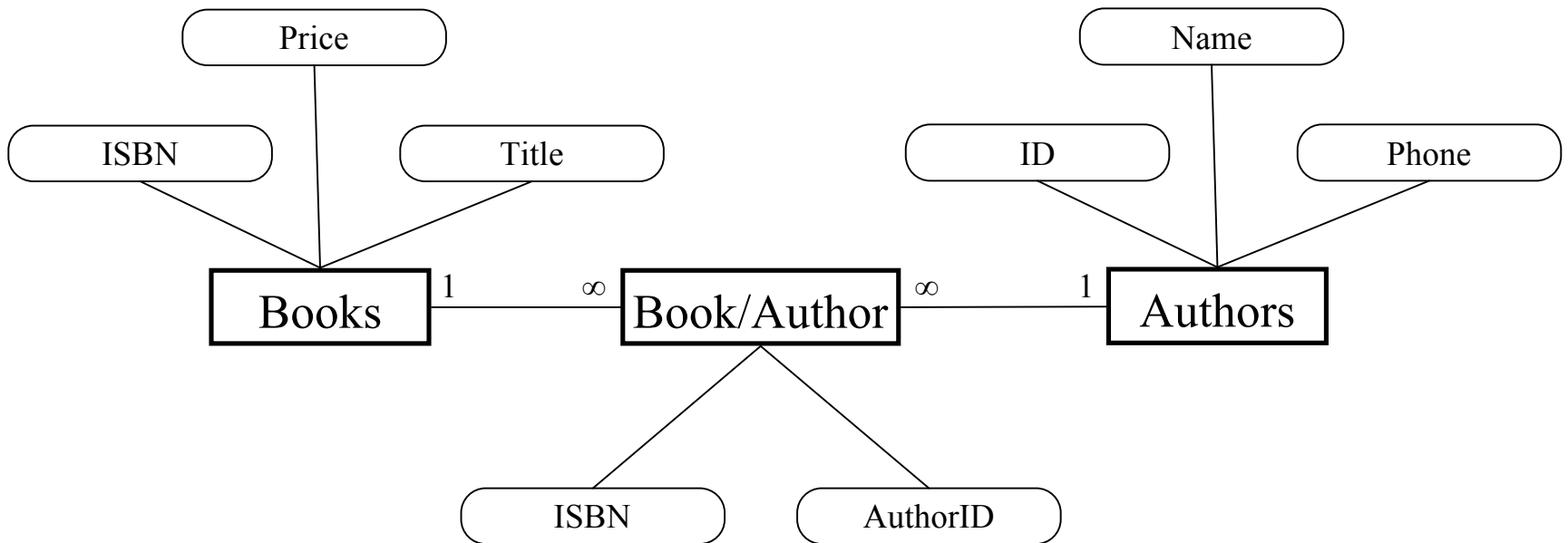
Link one book with many authors?

- We DO want:
 - » to link each book to one or more authors
- We DON'T want
 - » to specify extra fields (author1, author2, author3,...)
 - this is wasteful and limits the max number of authors
 - » to specify each book entry several times, naming a different author in each row
 - this duplicates all the other information about the book

Add a cross-reference table!

- Refine the design so that it includes another table that is a book-author cross reference
 - » Each entity in the table is a single cross reference
 - Attribute: ISBN
 - Attribute: Author ID
 - » No primary key
 - Now we can break the many-to-many relationship into two 1-to-many relationships that we already know how to implement
-

Define new cross-reference entities





book-author table

The screenshot shows Microsoft Access with a database named 'little : Database [Access 2002 file format]'. The 'Objects' pane on the left shows 'Tables' selected, with a list containing 'authors', 'book-author', 'books', and 'publishers'. The 'authors' table is displayed in the top right pane, and the 'books' table is displayed in the middle right pane. A third table, 'book-author', is displayed in a separate window in the foreground.

authors : Table

ID	Name	Phone
1	Alex	555-0256
2	Bill	555-0512
3	Charlie	555-1024
*		

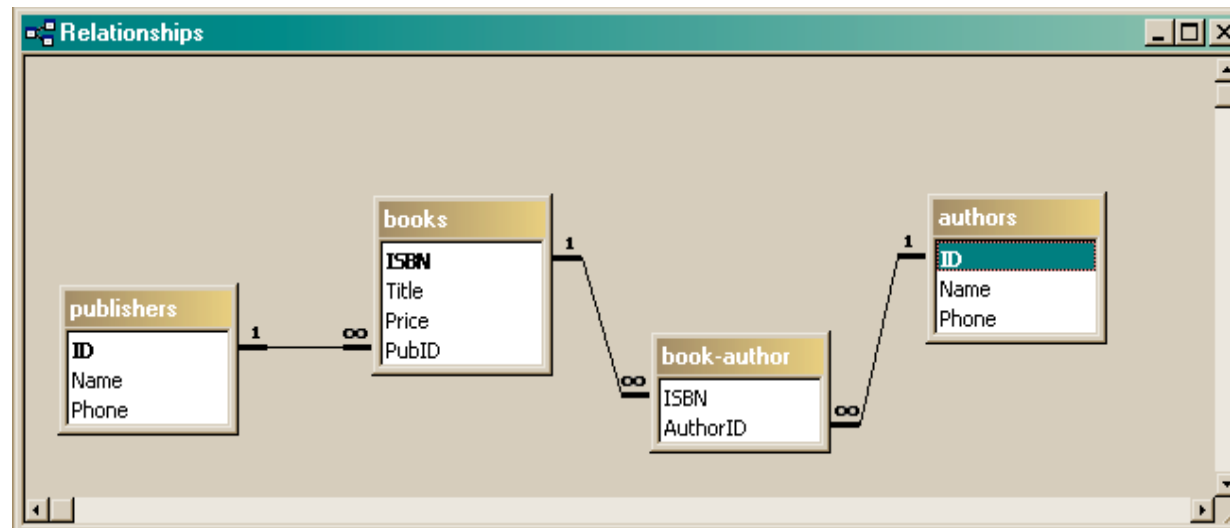
books : Table

ISBN	Title	Price
1-1	My Reader	\$10.00
1-2	Your Reader	\$12.00
2-2	His Reader	\$25.00
*		\$0.00

book-author : Table

ISBN	AuthorID
1-1	1
1-2	1
2-2	2
2-2	3
*	

Define the new relationships



Define a query that uses the relationship

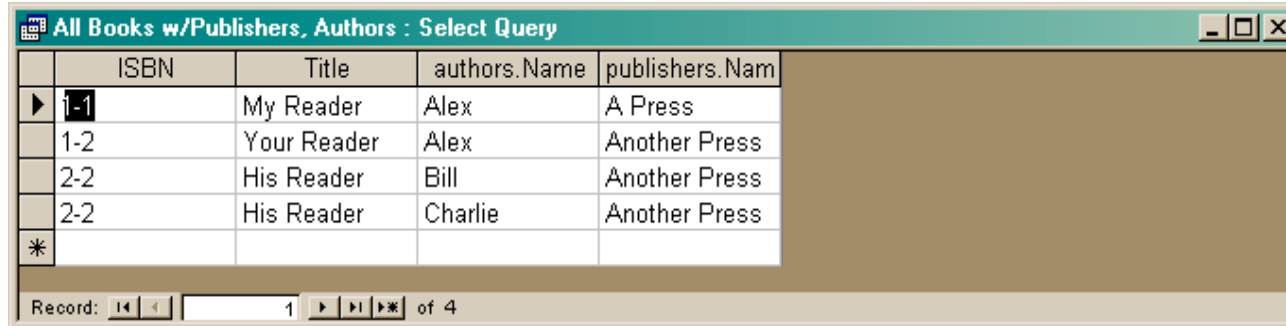
The screenshot shows a database query builder interface. At the top, a window titled "All Books w/Publishers, Authors : Select Query" displays a relationship diagram. The diagram includes four tables: publishers, books, book-author, and authors. Publishers is connected to books (1 to ∞), books is connected to book-author (1 to ∞), and book-author is connected to authors (∞ to 1). Below the diagram is a table with columns for Field, Table, Sort, Show, Criteria, and or. The Field column contains ISBN, Title, Name, and Name. The Table column contains books, books, authors, and publishers. The Show column has checkboxes for each field. Below this is another window titled "All Books w/Publishers, Authors : Select Query" containing the following SQL query:

```
SELECT books.ISBN, books.Title, authors.Name, publishers.Name
FROM publishers INNER JOIN (books INNER JOIN (authors INNER JOIN [book-author] ON authors.ID = [book-author].AuthorID) ON
books.ISBN = [book-author].ISBN) ON publishers.ID = books.PubID;
```

Query By Example

actual SQL

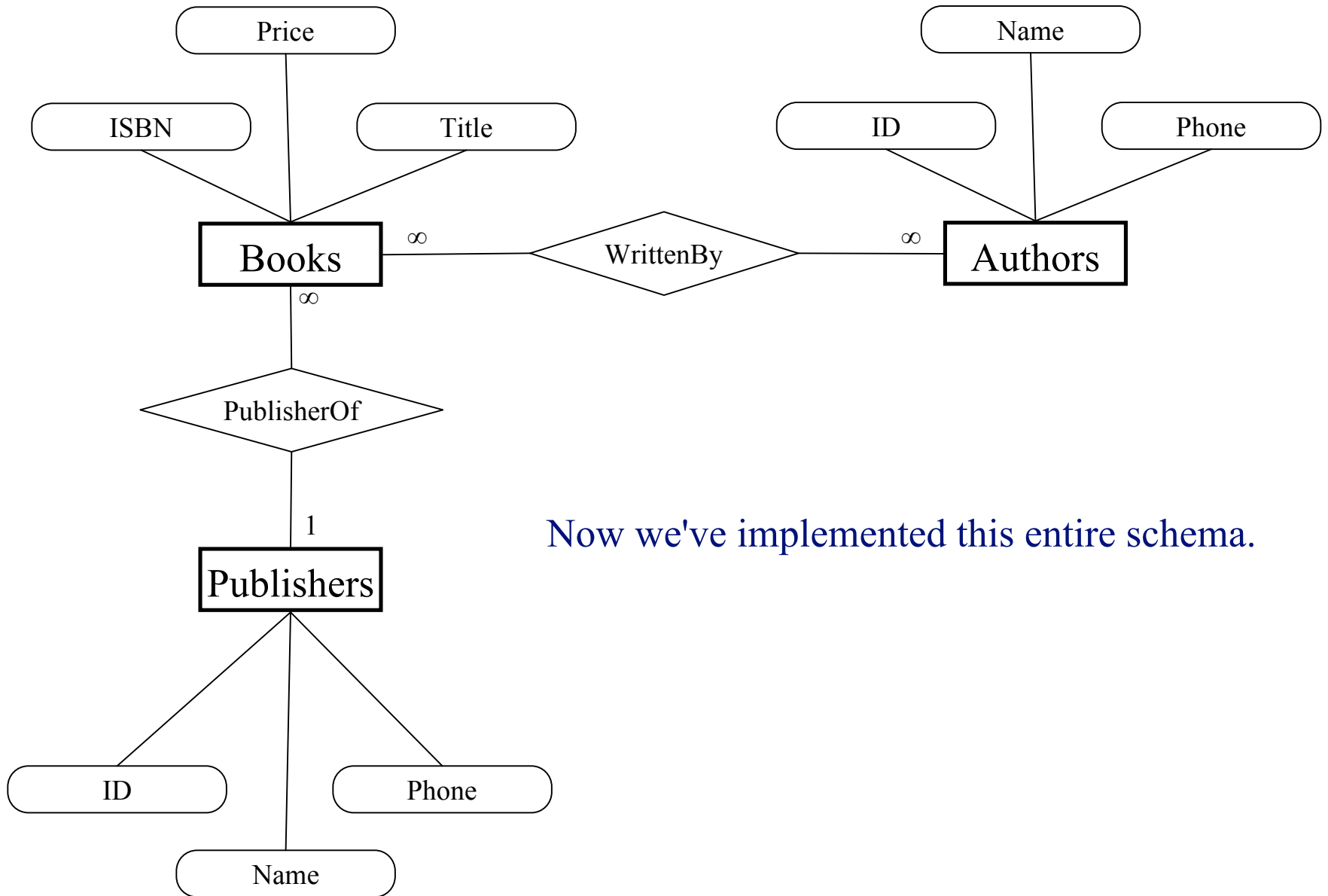
Get the new view of the data



	ISBN	Title	authors.Name	publishers.Nam
▶	1-1	My Reader	Alex	A Press
	1-2	Your Reader	Alex	Another Press
	2-2	His Reader	Bill	Another Press
	2-2	His Reader	Charlie	Another Press
*				

Record: 1 of 4

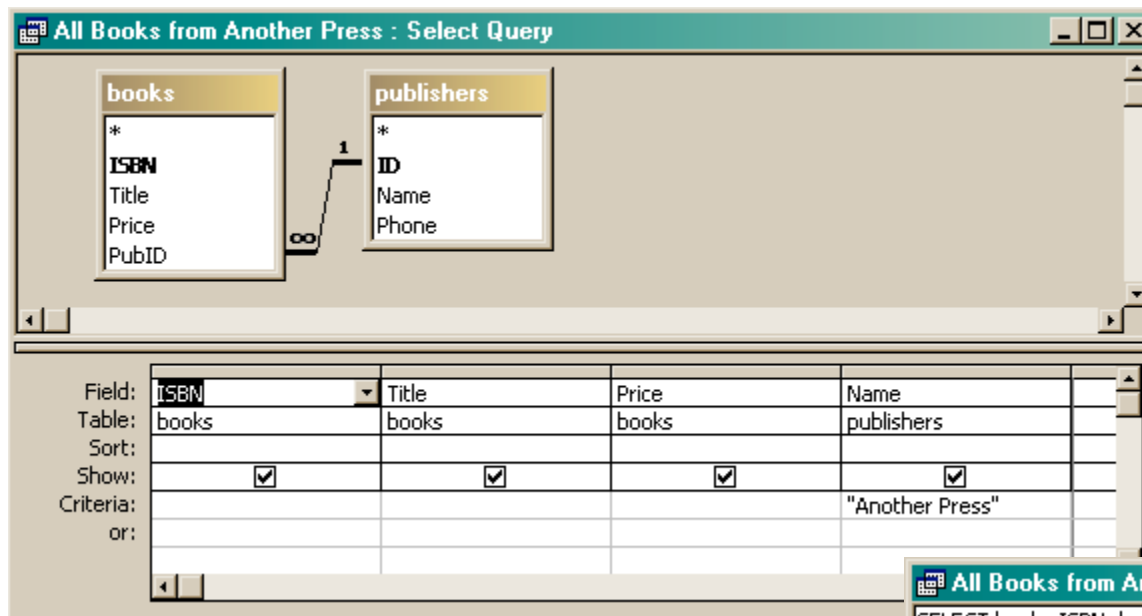
- Notice that this view has redundant data
 - » That's okay, because we are not storing it this way, just presenting it
 - » The redundant items (Alex, Another Press) came from a single entry in a table – they are guaranteed to be identical



Now we've implemented this entire schema.

View: All Books from “Another Press”

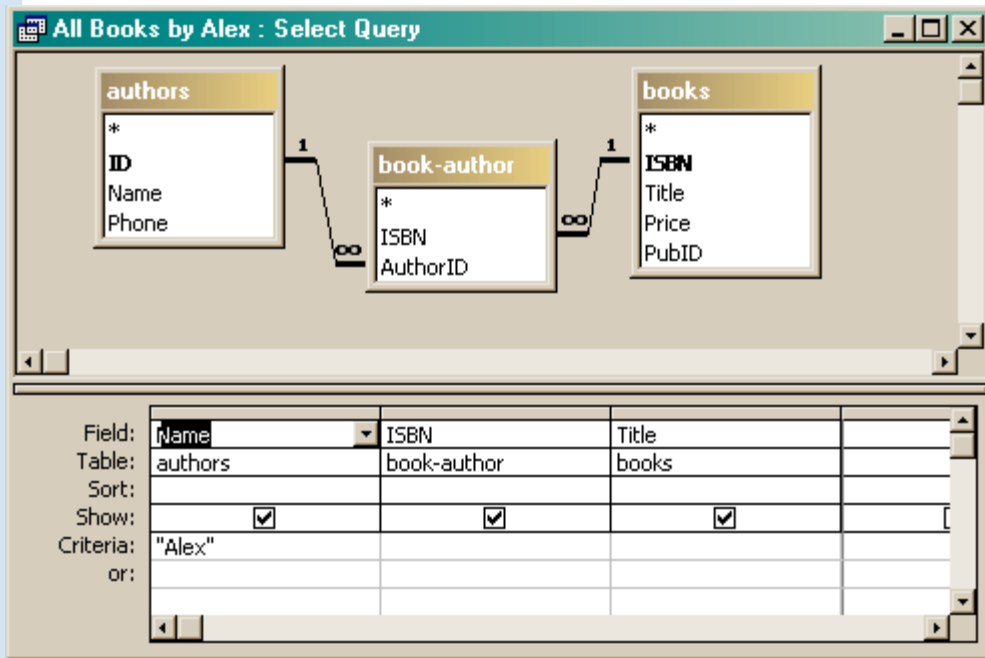
ISBN	Title	Price	Name
1-2	Your Reader	\$12.00	Another Press
2-2	His Reader	\$25.00	Another Press



```
SELECT books.ISBN, books.Title, books.Price, publishers.Name
FROM publishers INNER JOIN books ON publishers.ID=books.PubID
WHERE (((publishers.Name)='Another Press'));
```



View: All Books by Alex



Name	ISBN	Title
Alex	1-1	My Reader
Alex	1-2	Your Reader

Record: 3 of 3

```

SELECT authors.Name, [book-author].ISBN, books.Title
FROM books INNER JOIN (authors INNER JOIN [book-author] ON authors.ID=[book-author].AuthorID) ON books.ISBN=[book-author].ISBN
WHERE (((authors.Name)="Alex"));
    
```

View: All info about a given ISBN

Enter Parameter Value

ISBN?

1-1

OK Cancel

Book Info for Given ISBN : Select Query

ISBN	Title	Price	authors.Name	publishers.Nam
1-1	My Reader	\$10.00	Alex	A Press
*				

Record: 1 of 1

Book Info for Given ISBN : Select Query

publishers: ID, Name, Phone

books: ISBN, Title, Price, PubID

book-author: ISBN, AuthorID

authors: ID, Name, Phone

Field:	ISBN	Title	Price	Name	Name
Table:	books	books	books	authors	publishers
Sort:					
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:	[ISBN?]				
or:					

Book Info for Given ISBN : Select Query

```
SELECT books.ISBN, books.Title, books.Price, authors.Name, publishers.Name
FROM publishers INNER JOIN (books INNER JOIN (authors INNER JOIN [book-author] ON authors.ID = [book-author].AuthorID) ON
books.ISBN = [book-author].ISBN) ON publishers.ID = books.PubID
WHERE (((books.ISBN)=[ISBN? ]));
```