



CPSC203 – Introduction to Problem Solving and Using Application Software

Winter 2010

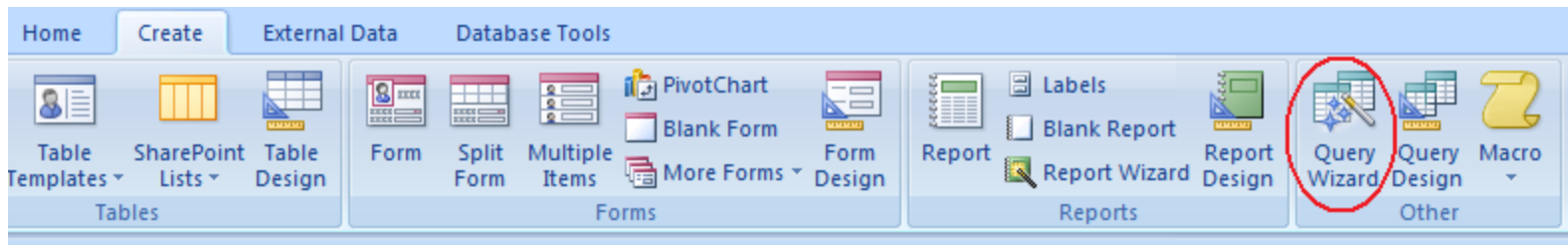
Tutorial 8: Mehrdad Nurolahzade

Introduction

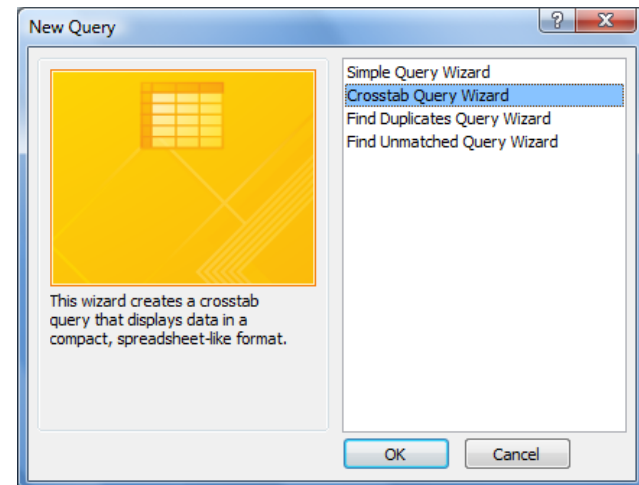
- Cross-tab queries
- Custom calculations

Crosstab Queries (1)

- To create a cross-tab query, select **Query Wizard** from **Create** menu

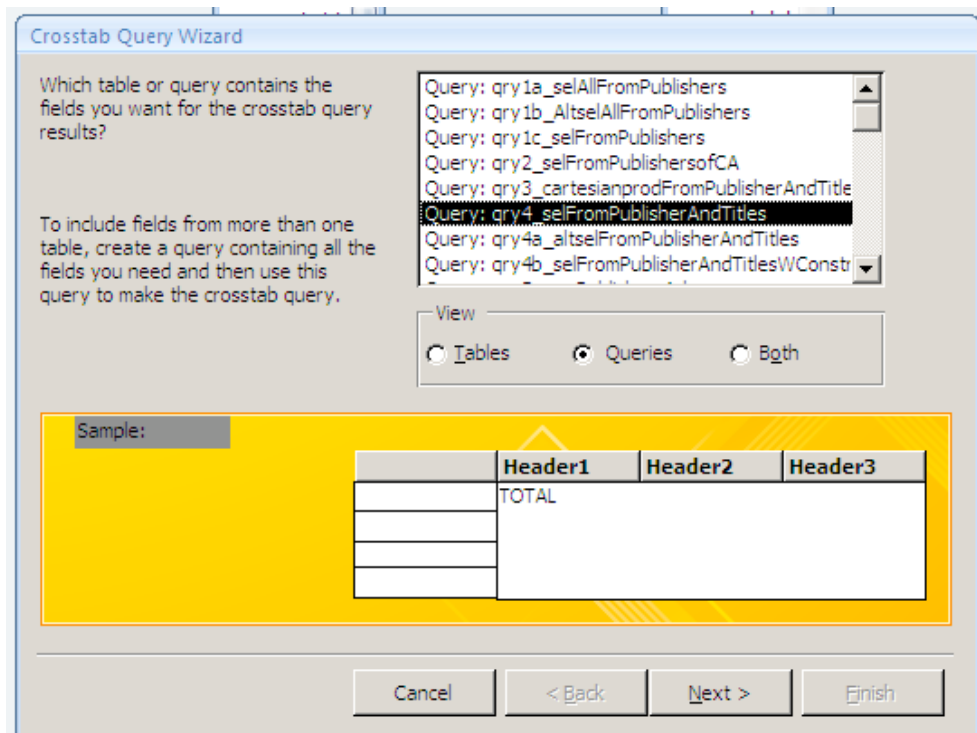


- In the New Query window select **Crosstab Query Wizard** and press **OK**



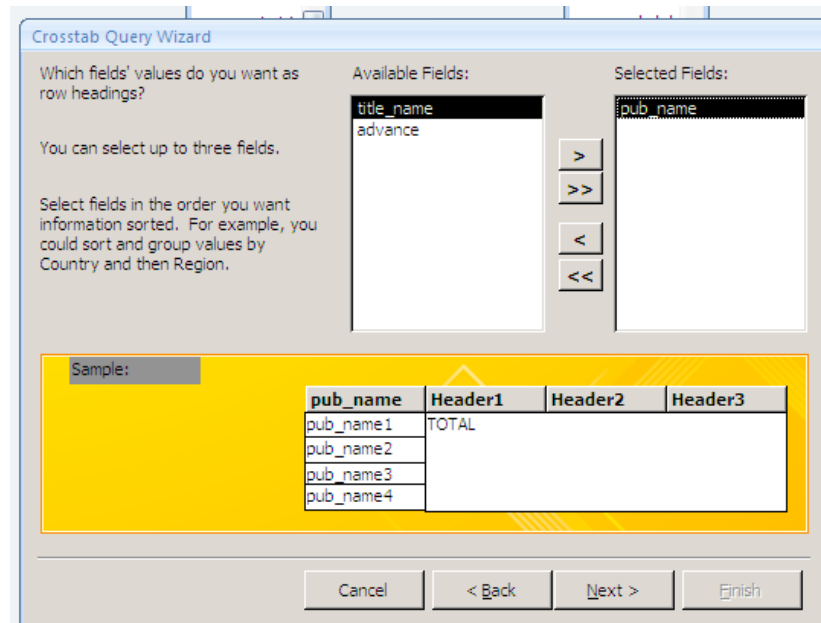
Crosstab Queries (2)

- Select the table/query that contains the fields you want and press **Next**



Crosstab Queries (3)

- Select from the **Available Fields** list the fields you wish to display as row headings by adding them to the **Selected Fields** list and press **Next**



Crosstab Queries (4)

- Specify a field to display values for the column heading and press **Next**

Which field's values do you want as column headings?

For example, you would select Employee Name to see each employee's name as a column heading.

title_name
advance

Sample:

pub_name	advance1	advance2	advance3
pub_name1	TOTAL		
pub_name2			
pub_name3			
pub_name4			

Cancel < Back Next > Finish

Crosstab Queries (5)

- Specify the type of calculation to perform and which field and press **Next**

Crosstab Query Wizard

What number do you want calculated for each column and row intersection?

For example, you could calculate the sum of the field Order Amount for each employee (column) by country and region (row).

Do you want to summarize each row?

Yes, include row sums.

Fields:

title_name

Functions:

Count
First
Last
Max
Min

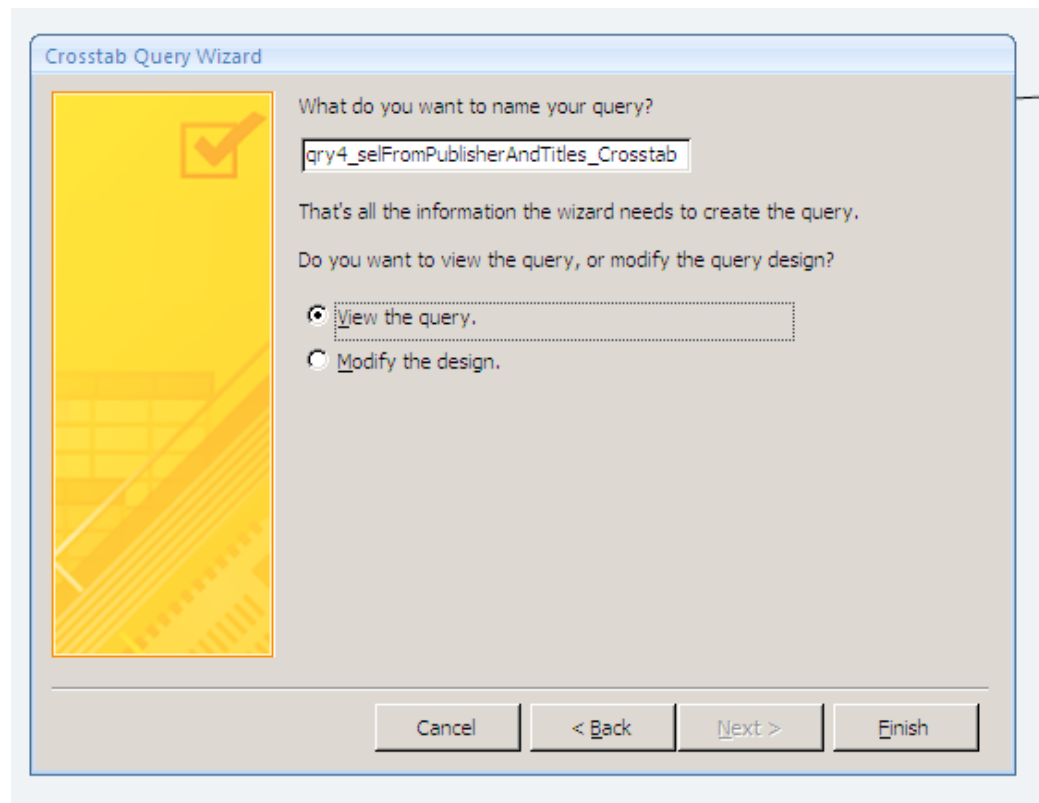
Sample:

pub_name	advance1	advance2	advance3
pub_name1	Count(title_name)		
pub_name2			
pub_name3			
pub_name4			

Cancel < Back Next > Finish

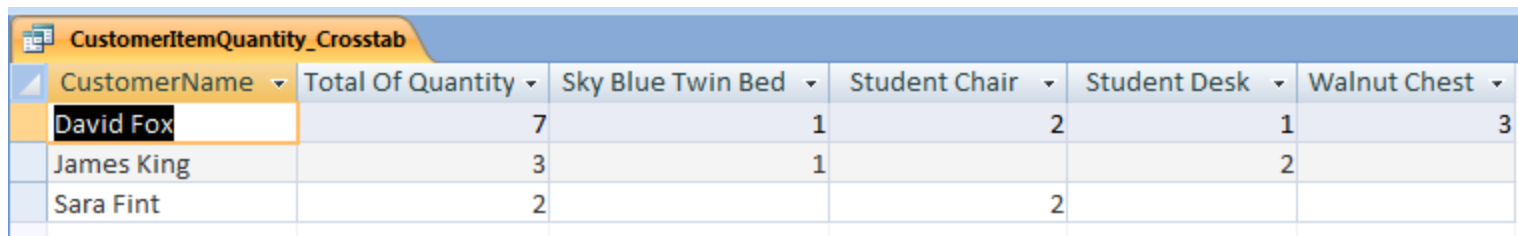
Crosstab Queries (6)

- Specify a name for your query and press **Finish**



Crosstab Example (1)

- Create a new query that contains all the fields from tables **Customer**, **Item**, and **Order** in the example database
- Create a new crosstab query from the above query
- Select **Customer.CustomerName** for row heading
- Select **Item.ItemName** for column heading
- Select **Order.Quantity** for cell value
- Select **Sum** for function



CustomerName	Total Of Quantity	Sky Blue Twin Bed	Student Chair	Student Desk	Walnut Chest
David Fox	7	1	2	1	3
James King	3	1		2	
Sara Fint	2		2		

Crosstab Example (2)

- Create a crosstab query that shows number of ordered items per order status

ItemName	Total Of Quantity	Cancelled	Delivered Successfully	Delivered Successfully	Processing
Sky Blue Twin Bed	2	1		1	
Student Chair	4		2		2
Student Desk	3	1	1		1
Walnut Chest	3	3			

- Answer:
 - Rows: *Item.ItemName*
 - Columns: *Order.Status*
 - Cells: *Order.Quantity*
 - Function: *Sum*

Crosstab Example (3)

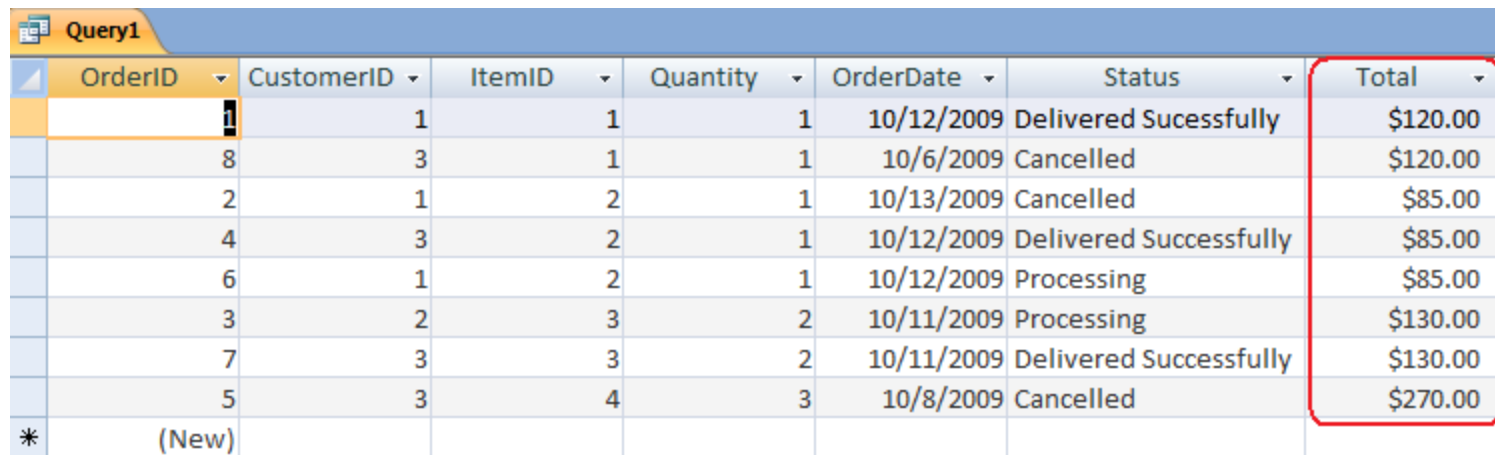
- Create a crosstab query that shows number of number of times customers ordered each item

CustomerName	Total Of OrderID	Sky Blue Twin Bed	Student Chair	Student Desk	Walnut Chest
David Fox	4	1	1	1	1
James King	3	1		2	
Sara Fint	1		1		

- Answer:
 - Rows: *Customer.CustomerName*
 - Columns: *Item.ItemName*
 - Cells: *Order.OrderID*
 - Function: *Count*

Custom Calculations (1)

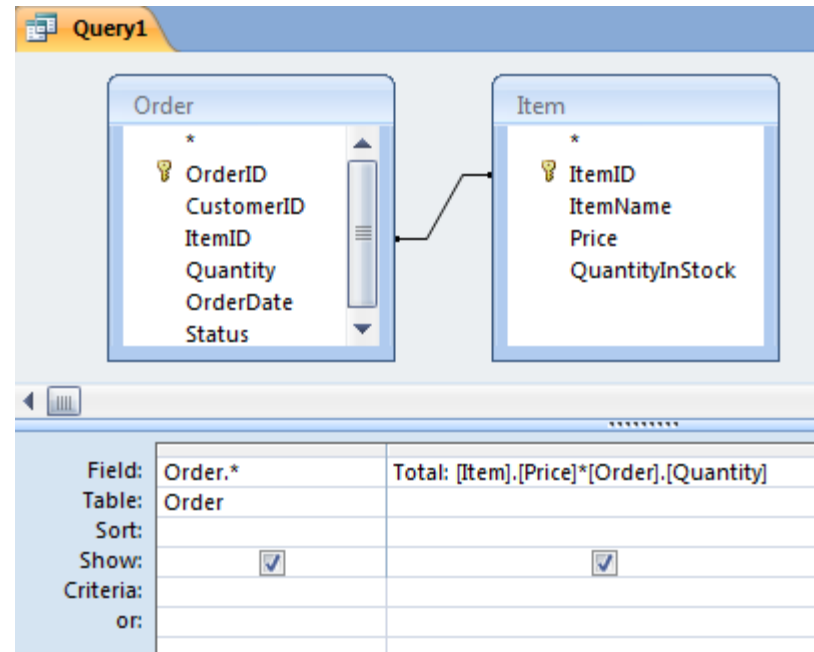
- New fields can be added to a query result based on a calculation on existing fields
- Example: Create a new query that shows all orders together with their **Total** value



The screenshot shows a query result window titled "Query1". The table contains the following data:

OrderID	CustomerID	ItemID	Quantity	OrderDate	Status	Total
1	1	1	1	10/12/2009	Delivered Successfully	\$120.00
8	3	1	1	10/6/2009	Cancelled	\$120.00
2	1	2	1	10/13/2009	Cancelled	\$85.00
4	3	2	1	10/12/2009	Delivered Successfully	\$85.00
6	1	2	1	10/12/2009	Processing	\$85.00
3	2	3	2	10/11/2009	Processing	\$130.00
7	3	3	2	10/11/2009	Delivered Successfully	\$130.00
5	3	4	3	10/8/2009	Cancelled	\$270.00
*(New)						

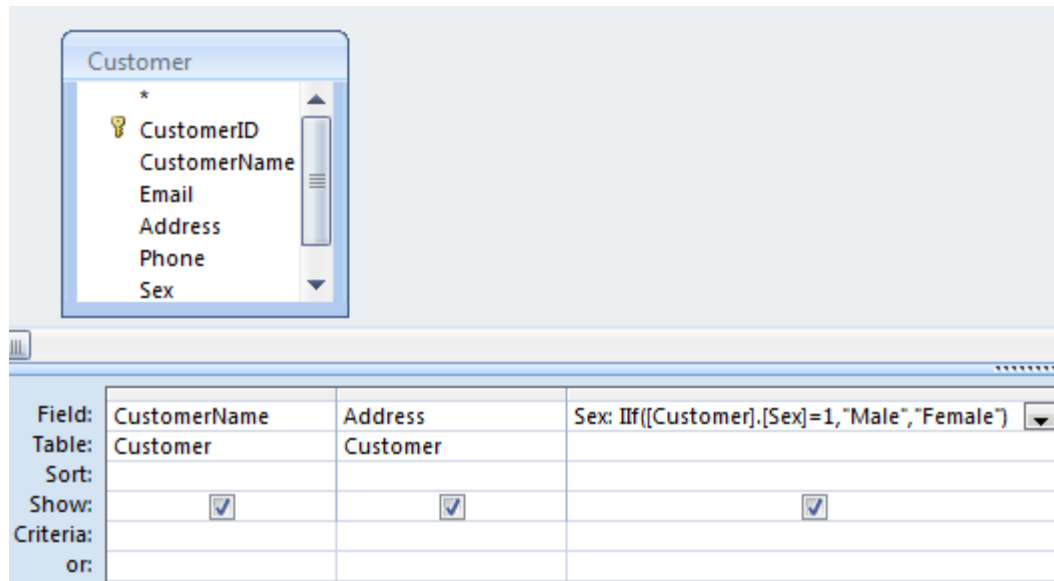
Custom Calculations (2)



```
SELECT Order.*, Order.Quantity*Item.Price AS Total
FROM Order, Item
WHERE Order.ItemID=Item.ItemID
```

IF Statement (1)

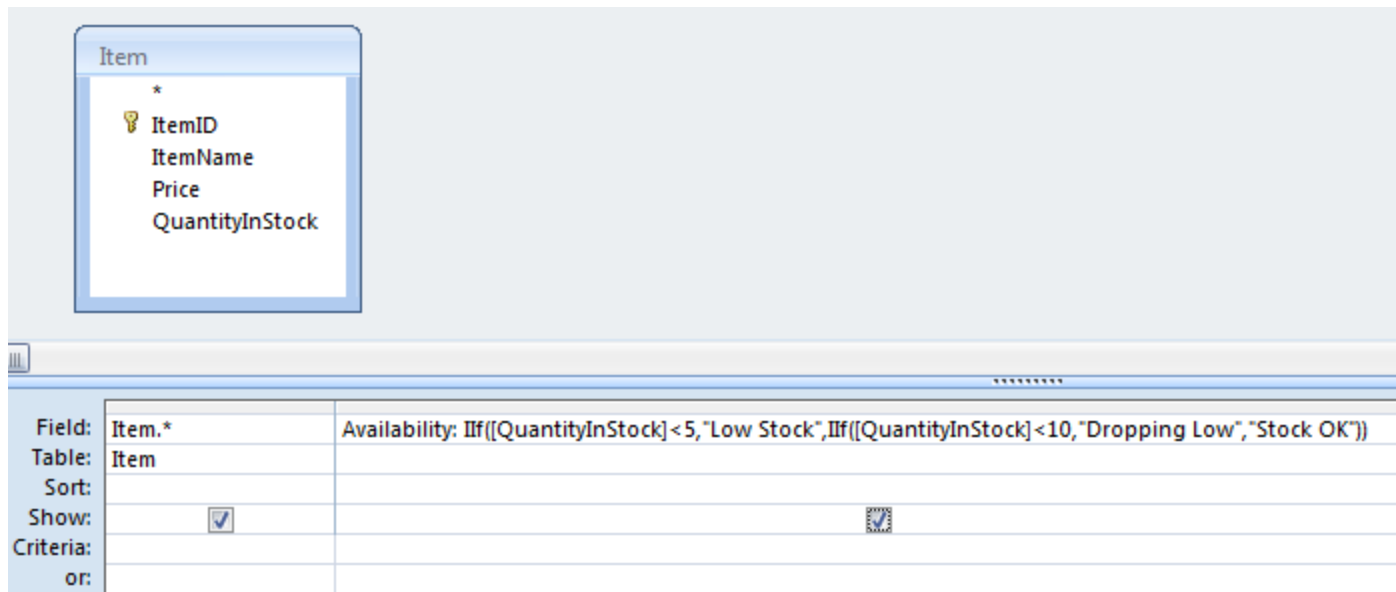
- *IIf(Condition, True Clause, False Clause)*



```
SELECT CustomerName, Address, IIf(Sex=1, "Male", "Female") AS Sex  
FROM Customer
```

Nested IF Statement

- If item quantity in stock less than 5, then item availability is “Low Stock”; if item quantity in stock less than 10, then item availability is “Dropping Low”; otherwise item availability is “Stock OK”



The screenshot shows a database management tool interface. At the top, a table definition for 'Item' is displayed, listing fields: ItemID (primary key), ItemName, Price, and QuantityInStock. Below this, a query result is shown in a table format. The 'Field' column contains 'Item.*' and the 'Table' column contains 'Item'. The 'Criteria' column is empty. The 'or:' column is empty. The 'Show:' column has a checked checkbox. The 'Availability' column contains the nested IF statement: 'Iif([QuantityInStock]<5,"Low Stock",Iif([QuantityInStock]<10,"Dropping Low","Stock OK"))'. The 'Sort:' column is empty.

Field:	Item.*	Availability: Iif([QuantityInStock]<5,"Low Stock",Iif([QuantityInStock]<10,"Dropping Low","Stock OK"))
Table:	Item	
Sort:		
Show:	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Criteria:		
or:		

```
SELECT Item.*, Iif(QuantityInStock<5,"Low Stock",Iif(QuantityInStock<10,"Dropping  
Low","Stock OK")) AS Availability  
FROM Item
```