

Access Queries 2: Math & Crosstabs



Access Queries 2: Math & Crosstabs

1.0 hour

Expression Builder	3
Building Join Statement.....	3
Building Equations.....	4
Using Totals	5
Unique Records / No Duplicates.....	5
Summarizing Visit Length by Doctor.....	6
Working with Multiple Tables.....	6
Crosstab Query	7
Crosstab Query Wizard.....	7
Create Query with Crosstab Query Wizard.....	7
Practice: Make another Crosstab Query.....	9



Pandora Rose Cowart

Education/Training Specialist
UF Health IT Training

C3-013 Communicore
PO Box 100152
Gainesville, FL 32610-0152

(352) 273-5051
prcowart@ufl.edu
<http://training.health.ufl.edu>

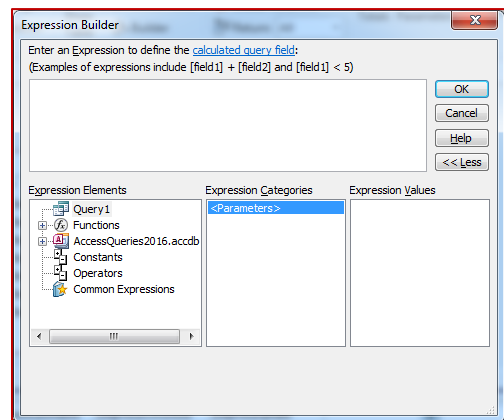
Class Evaluation: https://ufl.qualtrics.com/jfe/form/SV_10jkl6lRskV3XT

Expression Builder

Access has an Expression Builder to help you create your joins and calculations. To use this tool you can right-click in the field box, in the Design view of a Query, and choose Build...

You can find built in functions in this builder. In the left section, Expression Elements, double-click on **Functions**, and open the list of Built-In Functions. These include some statements you may be familiar with from Excel: Sum, Count, Max. Here Average() is Avg, and If() is iif. In Access, we use our field names instead of ranges of cells.

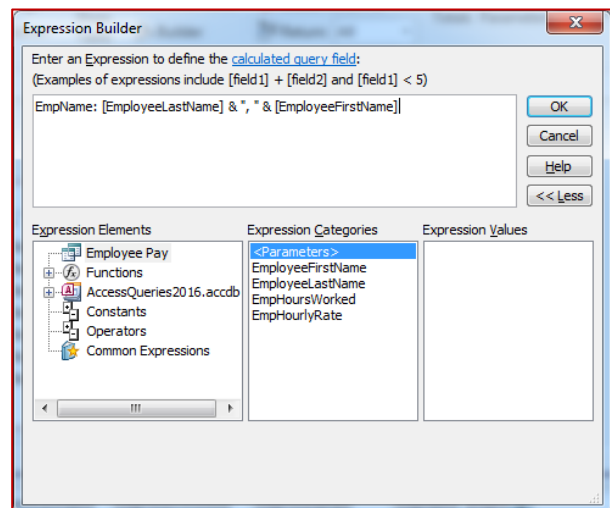
- Sample Derived Expression
 - LastFour: Right([SSN],4) # Find last four characters of the SSN
- Sample Logic Expression
 - Taxes: iif([Exempt]=Yes, 0, [Price]*0.06) # Calculate the taxes on nonexempt items
- Sample Calculated Expression
 - Total: [AmtSold] * [TotalPurchased]



Building Join Statement



1. Create Query in Design View, using the table Employees
2. Add the following Fields:
 - EmployeeFirstName
 - EmployeeLastName
 - EmpHoursWorked
 - EmpHourlyRate
3. Right-click next blank column, and choose **Build**
 - Expression Categories section is blank, click **Cancel**
4. Save Query as **Employee Pay**
5. Try Expression Builder again
 - Double-click EmployeeLastName
 - Type: "&", "&
 - Double-click EmployeeFirstName
6. Click OK and View Datasheet
7. In the Design view, rename Expr1 to EmpName and move this expression to the first column
8. Delete the **EmployeeFirstName** and **EmployeeLastName** fields from bottom of the Query Design



EmpName: [EmployeeLastName] & ", " & [EmployeeFirstName]	EmpHoursWorked	EmpHourlyRate
	Employees	Employees

Building Equations

Calculate **Gross Pay**

- 1. Right-click next blank column, choose BUILD
 - Type - **EmpGrossPay**:
 - Double-click to get **[EmpHoursWorked]**
 - Double-click to get **[EmpHourlyRate]**
 - Erase first <<expr>>, change second one to * (to multiply)
- 2. Expression should read - **EmpGrossPay: [EmpHoursWorked] * [EmpHourlyRate]**
- 3. View Datasheet to see the results

Calculate **Deductions**

- 1. In the next blank column type -- Deductions:0.175
 - Change Format properties to **Percent** (right-click to get property window)
- 2. View the Datasheet, we now have a constant in the Deduction column
- 3. Use Expression Builder to modify the equation to show the deduction amount instead of the deduction rate; don't forget to save first.
- 4. Change the expression to **Deductions: 0.175*[GrossPay]**
- 5. Right-click Properties, Format -> Currency

Calculate **NetPay**

- 1. NetPay: **[GrossPay] -[Deductions]**
- 2. View Datasheet
- 3. Close and Save query

GrossPay: [EmpHoursWorked]*[EmpHourlyRate]	Deductions: 0.175*[grosspay]	NetPay: [GrossPay]-[Deductions]

EmpName	EmpHoursWorked	EmpHourlyRate	GrossPay	Deductions	NetPay
Appleton, Annie	40	\$10.00	\$400.00	\$70.00	\$330.00
Blackthorne, Billie	40	\$15.00	\$600.00	\$105.00	\$495.00
Carson, Carly	40	\$12.50	\$500.00	\$87.50	\$412.50
Dawson, Devon	35	\$7.00	\$245.00	\$42.88	\$202.13
Edwards, Edgar	25	\$6.50	\$162.50	\$28.44	\$134.06

Using Totals

By default, the **Total** row does not appear in the Design view of a Query. To display this row, click on the **Totals** button on the right side of the Design tab, or right-click in the bottom of the Query and choose **Totals**.

Without Totals

Field:	Visit Date
Table:	Visit Info
Sort:	
Show:	<input checked="" type="checkbox"/>
Criteria:	
or:	

Visit Date	4/28/2009
	6/17/2009
	4/28/2009
	4/28/2009
	6/17/2009

With Totals (group by)

Field:	Visit Date
Table:	Visit Info
Total:	Group By
Sort:	
Show:	<input checked="" type="checkbox"/>
Criteria:	
or:	

Visit Date	4/28/2009
Total	6/17/2009

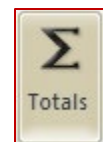
Total functions include:

- **Group By:** Group this field (Hide Duplicate Values)
- **Sum:** Find Sum within each Group
- **Avg:** Find Average within each Group
- **Min:** Find Minimum Value within each Group
- **Max:** Find Maximum Value within each Group
- **Count:** Find Number of Items within each Group
- **StDev:** Find Standard Deviation within each Group
- **Var:** Find Variance of each Group
- **First:** Find First Value entered of each group
- **Last:** Find Last Value entered of each Group
- **Expression:** No Totals, just evaluate expression within the field
- **Where:** Find the dataset based on this criteria, by default this will not show this field

Unique Records / No Duplicates



1. Create Query in Design view using the table Doctors
2. Add the field **DocFirst**
3. View Datasheet (3 records)
4. In the Design view, drag the Table **Visit Info** into the top half of the Query window
 - Because a relationship is set, the tables will already be linked
5. Don't add anything to the query, but view the Datasheet (162 records)
 - There are still only three doctors, but adding the **Visit Info** table, which has 162 records, changes the Query result to show the doctor for each Visit Info record.
6. From the Design tab turn on the **Totals** (look for the sigma on the right side)
7. View Datasheet (3 records)
8. In the Design view, add the field **Visit Length**
9. View Datasheet (15 records)
 - 3 Doctors, 5 Length options. $3 * 5 = 15$ records



Summarizing Visit Length by Doctor

- 1. In the Design view, change the **Visit Length's Totals** row from **Group by** to **Sum**
- 2. View Datasheet (3 records)
- 3. In the Design view, add a title to Visit Length
 - **Total Time: Visit Length**
- 4. Add another **Visit Length** field to the bottom of the Query
 - Change the second **Visit Length** field from Group by to **Average**
 - Add the title **Avg Time**
 - Change the **Format** property to **Standard**
 - ~ If needed, right-click on the field to open the Properties
 - ~ Access ignores the number of decimal places until you choose a **Format**
- 5. View Datasheet (3 records)
- 6. Close and Save as **Doctor Time Summary**

Field:	DocFirst	Total Time: Length	Avg Time: Length
Table:	Doctors	Visit Info	Visit Info
Total:	Group By	Sum	Avg
Sort:			

DocFirst	Total Time	Avg Time
Sallye	2250	41.67
Samuel	2205	40.83
Sidney	2295	42.50

Working with Multiple Tables

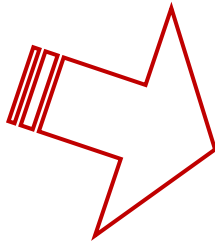
- 1. Create Query in Design view using the tables Doctors and Visit Reasons
 - There is no relationship line between the two tables. There is no inherent link between these two datasets. Any fields used here will give you every possible combination of records, even if they don't exist in the actual datasets.
 - When does the doctor treat a reason? When is the reason treated by the Doctor? During the visit. The **Visit Info** table must be part of the Query to get an accurate result on your data.
- 2. In the Design view, drag the Table **Visit Info** into the top half of the Query window
- 3. Add the following Fields:
 - From Table Doctors: **DocFirst**
 - From Table Visit Reasons: **VisitReason**
 - From Table Visit Info: **Visit Length**
- 4. Turn on Totals
- 5. Change the **Visit Length** Total to **Sum**
- 6. View Datasheet (25 records)
 - 3 Doctors * 9 Reasons = 27 records, but we only get 25. This is because at this time, there is only one *Headache* record. As the other Doctors treat Headaches, they will appear in this datasheet.
- 7. Turn off Totals (162 records)
- 8. Close and Save as **Doctors by Reason**

Crosstab Query

Crosstab queries will put the unique values of a field across the top of the Query result as Field Headers, and summarizes a field for each data point. Think of it as a simple version of a PivotTable. We will use the wizard in class, but you can build your own once you become comfortable.

Example:

<u>Item</u>	<u>Color</u>	<u>Sold</u>
Blouses	Blue	5
Blouses	Red	10
Pants	Blue	3
Pants	Red	12
Socks	Blue	1
Socks	Red	5



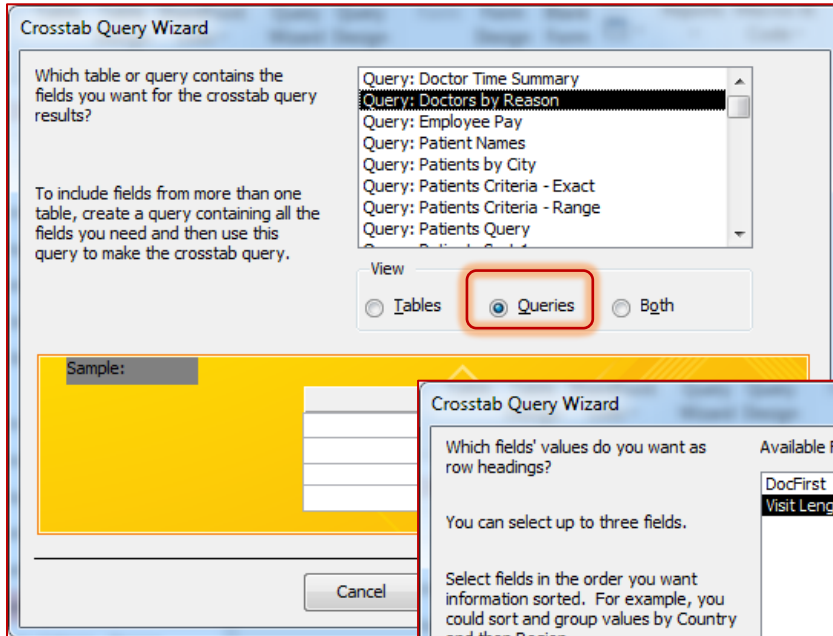
<u>Item</u>	<u>Blue</u>	<u>Red</u>	<u>Total Sold</u>
Blouses	5	10	15
Pants	3	12	15
Socks	1	5	6

Crosstab Query Wizard

Create Query with Crosstab Query Wizard



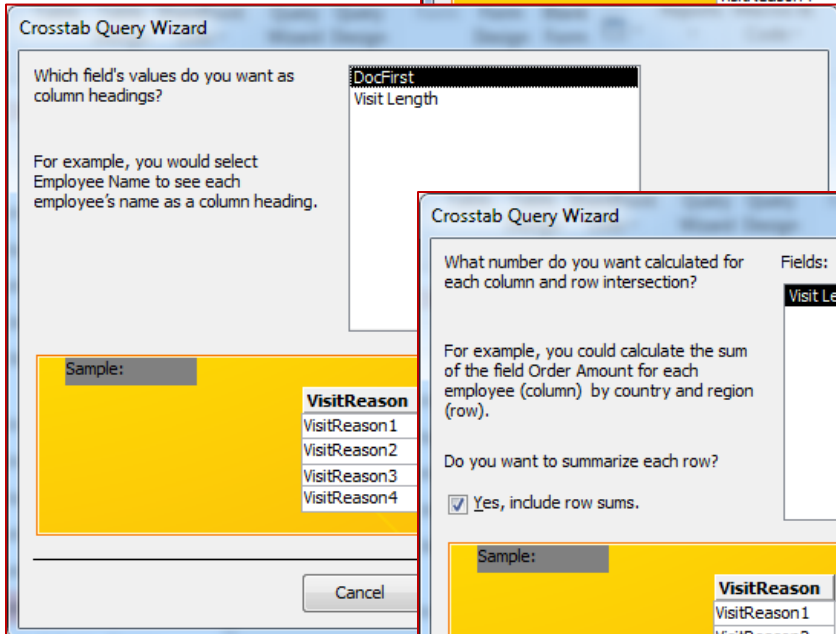
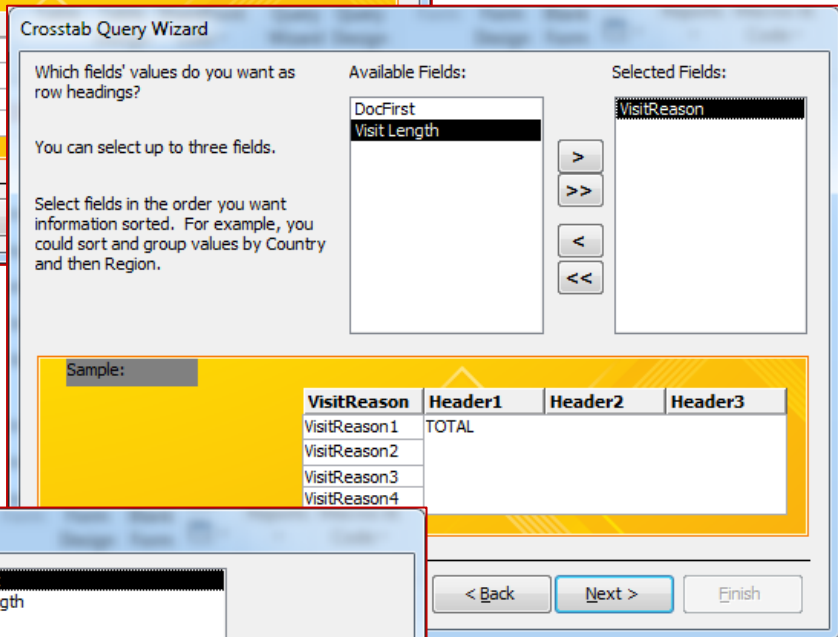
1. From the Create tab choose **Query Wizard**
2. Select the **Crosstab Query Wizard**
3. Since we want information from three different tables, we have to create the query with all the fields we need before we run this wizard. We created this query as **Doctors by Reason**. You won't see this query right away because the wizard is only displaying the tables.
 - Change the **View setting** to show the **Queries**, choose **Query: Doctors by Reason (Next)**
4. You can use up to three fields as your Row Headings. Watch the Preview to see if you're getting the desired fields along the side of the Query result.
 - Select the **VisitReason** as your Row Heading (*Next*)
5. Select the **DocFirst** field as your Column Heading (*Next*)
6. Select the **Sum** option to summarize our **Visit Length**
7. Name the Query: **Doctors by Reason_Crosstab Sum**



Set the View to Queries or Both and from the list choose

Query: Doctors by Reason

Choose the **VisitReason** as the Row Heading.
 Choose **DocFirst** as the Column Heading.



Use the Sum Function for the **Visit Length** field

8. Turn on the **Totals** from the **Home** Tab in the Datasheet view to calculate column Totals.

- Under each column change the Total to **Sum**

9. Go to the Design view. Notice the new row in the bottom of the query for **Crosstab**.

- There can only be one **Value** field, and it must have a mathematical summary: Sum, Avg, Count
- There can only be one **Column Heading** field

The screenshot shows the Microsoft Access ribbon with the 'Totals' button highlighted in a red box. Below the ribbon is a table titled 'Doctors by Reason_Crosstab Sum'.

Reason	Total Of Visi	Sallye	Samuel	Sidney
Allergies	825	240	285	300
Backache	885	345	330	210
Cold/Flu	960	405	240	315
Followup	795	240	240	315
Headache	45		45	
Heartburn	360	120	180	60
Nausea	915	390	285	240
Physical	915	240	345	330
Sore Throat	1050	270	255	525
Total	6750	2250	2205	2295

The screenshot shows the Design view of a query named 'Doctors by Reason_Crosstab Sum'. A field list box is open, showing fields from the 'Doctors by Reason' table: DocFirst, VisitReason, and Visit Length. Below the field list is a table defining the query structure.

Field:	VisitReason	DocFirst	Visit Length	Total Of Visit Length: Visit Length
Table:	Doctors by Reason	Doctors by Reason	Doctors by Reason	Doctors by Reason
Total:	Group By	Group By	Sum	Sum
Crosstab:	Row Heading	Column Heading	Value	Row Heading
Sort:				
Criteria:				
or:				

Practice: Make another Crosstab Query

- Based on Query: Doctors by Reason
- Row Heading: VisitReason
- Column Heading: DocFirst
- Value Summary: Avg
- Name the Query: **Doctors by Reason Crosstab Avg**
- Once the query is created, change the format property of the Visit Length fields to Standard
 - There are two of them. The Visit Length and the Total of Visit Length