Microsoft Access 2010
Select Queries and Crosstabs
Microsoft Access: Select Queries and Crosstabs

2.0 hours

Queries compile data from tables into a desired dataset. You can choose multiple fields from one or more data tables, and set criteria to limit the records displayed. By its simplest definition a query is a saved filter, but they can also be used to combine fields, perform calculations, and create summaries. Topics for this workshop include using data filters, setting custom sorts, using criteria, creating joined field expressions, calculating simple math, summarizing with totals, formatting numbers, and creating crosstabs. This workshop highly recommends completion of "Access: Basics".

Sort & Filter ................................................................................................................................. 1
  Filter by Selection ...................................................................................................................... 2
  Data type Filters ..................................................................................................................... 2
  Filter by Form ............................................................................................................................ 2
Creating Queries .......................................................................................................................... 3
  Query Wizards ........................................................................................................................... 3
    Simple Query Wizard ............................................................................................................ 3
    Crosstab Query Wizard ......................................................................................................... 3
    Find Duplicates Query Wizard .............................................................................................. 5
    Find Unmatched Query Wizard ............................................................................................ 5
Design View .................................................................................................................................... 6
  Field Row ................................................................................................................................... 7
  Table Row .................................................................................................................................. 8
  Sort Row .................................................................................................................................... 8
  Show Row ................................................................................................................................... 8
  Criteria Row ............................................................................................................................... 8
Working with Queries ................................................................................................................... 9
  Joining Fields ............................................................................................................................ 9
  Missing Parameter Values ......................................................................................................... 9
Totals ............................................................................................................................................... 10
  Field Properties ......................................................................................................................... 11
Expressions .................................................................................................................................... 12
Class Exercise ................................................................................................................................ 13

Pandora Rose Cowart
Education/Training Specialist
UF Health IT Training

C3-013 Communicore         (352) 273-5051
PO Box 100152              prcowart@ufl.edu
Gainesville, FL 32610-0152  http://training.health.ufl.edu

Updated: 5/22/2016
**Sort & Filter**
In the data view of Tables, Queries, Forms, and Reports you can see the Sort & Filter section on the Home tab.

We have the familiar Ascending (↑) and Descending (↓) buttons, and a Remove Sort ( ― ) option as well; this new option will remove the current sort order and return the dataset to the default, which in a table is sorting by the Primary Key.

The filter tools allow you to create a temporary "query" based on criteria. When you click on the filter arrow next to the field names in any Datasheet view you will see a menu similar to the one shown here. You will also get this menu when you click on the large Filter button on the toolbar based on the current field.

From this menu you can pick and choose which records to see. For example: in this set of data, if we only wanted to see the people in Gainesville and Starke, we could uncheck Micanopy, Waldo and (Blanks), and click OK. Access will hide the records that are not checked, and we will see only the records we want. You can continue to filter by as many fields as you would like, and Access will continue to limit the information based on your filters.

The field that was filtered shows a small funnel next to its filter arrow, and a Filtered appears next to the navigation buttons at the bottom of the window. When you click on the Filtered at the bottom of the table, the all of the records will appear and the data will be Unfiltered ( ). You can then use this button to toggle between your last filter and all the records, or you can use the button on in the Sort & Filter section of the Home tab.
**Filter by Selection**
The Selection button allows you to apply a filter based on the selected cell. For example, if we want to see only the customers who live in "Waldo", we can place our cursor in the "City" column where the value is "Waldo" and click on our Filter by Selection button. We can then choose how to filter by "Waldo". Equals implies that the match will be exact, contains allows for other characters, such as "Where’s Waldo?"

**Data type Filters**
When you try to filter with the large Filter button, or the drop down option on the field, Access gives you a list of logic values to help limit your data. The logic values will be different depending on the type of data.

<table>
<thead>
<tr>
<th>Text Filters</th>
<th>Number Filters</th>
<th>Date Filters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Equals...</td>
<td>Equals...</td>
<td>Equals...</td>
</tr>
<tr>
<td>Does Not Equal...</td>
<td>Does Not Equal...</td>
<td>Does Not Equal...</td>
</tr>
<tr>
<td>Begins With...</td>
<td>Less Than...</td>
<td>Before...</td>
</tr>
<tr>
<td>Does Not Begin With...</td>
<td>Greater Than...</td>
<td>After...</td>
</tr>
<tr>
<td>Contains...</td>
<td>Between...</td>
<td>Between...</td>
</tr>
<tr>
<td>Does Not Contain...</td>
<td></td>
<td>All Dates In Period</td>
</tr>
<tr>
<td>_ends With...</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Does Not End With...</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The Dates in Period include Quarters and Months.

Depending on the Filter option you choose you will get a small dialog box such as these:

**Filter by Form**
The final option can be found under the Advanced button (Advanced ...) on the Sort & Filter section – Filter by Form ( ). This option allows you to apply a filter based on one or more selections set up on a filter form. This is similar to Microsoft Excel's AutoFilter feature. When you choose this option, you will see a window similar to the one below:

Each field has a dropdown list containing the values of that column. Such that, if you wanted to filter for the customers in "Gainesville", "FL" and "32608", you could do so by choosing those items from their respective lists. Click on the Toggle Filter button to apply your filter.
**Creating Queries**

To create a permanent "filter", one that can then be used as a data source for the rest of the database, you can create a Query. Queries allow us pull from many related tables, create specific criteria to select our data and even create different levels of grouping within our data. To create a query, click on the Create tab; in the Other section we have two options for new queries, creating with a wizard or in through the query design view.

**Query Wizards**

There are four query wizards available.

**Simple Query Wizard**
This wizard creates a select query based on the fields you choose from one or more data source. It truly is a simple wizard, in that it only has two steps. If you want to choose any sorting levels, custom fields or criteria you will need to change the design view after the query is created.

**Crosstab Query Wizard**
This wizard sets up your data such that one field becomes a column heading, and it totals the data within each row-column (cell) match and creates a total field.

**Example:**

<table>
<thead>
<tr>
<th>Animal</th>
<th>Color</th>
<th>Sold</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bunnies</td>
<td>Brown</td>
<td>5</td>
</tr>
<tr>
<td>Bunnies</td>
<td>Black</td>
<td>10</td>
</tr>
<tr>
<td>Mice</td>
<td>Brown</td>
<td>3</td>
</tr>
<tr>
<td>Mice</td>
<td>Black</td>
<td>12</td>
</tr>
<tr>
<td>Pythons</td>
<td>Brown</td>
<td>1</td>
</tr>
<tr>
<td>Pythons</td>
<td>Black</td>
<td>5</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Animal</th>
<th>Brown</th>
<th>Black</th>
<th>TotalSold</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bunnies</td>
<td>5</td>
<td>10</td>
<td>15</td>
</tr>
<tr>
<td>Mice</td>
<td>3</td>
<td>12</td>
<td>15</td>
</tr>
<tr>
<td>Pythons</td>
<td>1</td>
<td>5</td>
<td>6</td>
</tr>
</tbody>
</table>
There are five steps to creating a cross tab query.

1. Choose your data source. You can only pull in data from one place, so you may need to create a query first.
2. Choose which fields will be along the left side of your results.
3. Choose which field will be along the top of your results. You can only use one field for this option.
4. Choose which field and function you wish to summarize. If you check the "Yes, include row sums" option, Access will create an extra column totaling each row.
5. Name the query and choose how you wish to view the results.
Find Duplicates Query Wizard
This wizard is a valuable tool to help maintain your data. It allows you to search for duplicate values, by field, within your data sources. This is especially important if you do not have a primary key or are using AutoNumbers. The wizard will walk you through choosing a data source and which fields may have the duplication. If you choose more than one field it will look for both to be duplicated (i.e. LastName and BirthDate).

Find Unmatched Query Wizard
This wizard is useful to compare two data sources to find data is in one source that is not in the other. Unmatched data is the most common reason Referential Integrity cannot be enforced between two tables. This wizard is also very useful to compare two imported Excel spreadsheets.

Find Matching Data (normal query)
- What data appears in both tables

Find unmatched data
- What is in the 1st table that is NOT in the 2nd
At the beginning of every semester we have a list of students from the previous semester and a list of students from the current semester. If we need to find who was a student last semester and this semester, a simple select query can retrieve that dataset.

But if we want to know who was a student last semester who is NOT a student this semester, we would use a FIND UNMATCHED QUERY.

**Design View**

When you choose to create a query in the Design View you will need to select your data source from the list of tables and queries. To add a table or query, choose it from the list and either click on the Add button or double on the item. Once you have added the tables and queries needed, you can close this window.

If you need to add another table/query, you can return to this window by clicking on the Show Table button on the query design tab, or right-click in the empty area in the top half of the window.

You can also drag the table/query from the navigation pane into the top half of the query design.
If there are relationships established between the tables, you will be able to see them here in the design view of the Query.

![Query Design View](image)

The data sources (where we will pull our data from) appear in the top half of the window. The bottom half is dedicated to what will be displayed. The query shown here will produce no data, because it is displaying no fields.

**Field Row**

To add a field to the bottom half of the screen, you can:

- Click and drag the field name from the table to the field row
- Double-click on the field from the table (Access will place it at the end of all the fields)
- Choose the field name from the drop down list provided (notice the arrow next to "LAST" below).
- To grab several fields at once: from within the table you can click on the first field, hold down the Shift key, and click on the last field, release the shift key, and click on the selection and drag down to the field row below.
- To pick and choose several fields at once: from within the table you can click on the first field you want and hold down the Ctrl key and click on the next field. Continue to hold down the control key until you have all the fields you desire and then drag the selection to the field row below.

The * field at the top of the tables and queries, will select all the fields in that table or query as one item, such that you will only see Patients.* on the field line, but the datasheet view of the query will show every field within that table.
Table Row
Below the Field: row is the Table: row. This shows the data source, that is which table or query the field originates from. This is important if the same field names are used across multiple tables, or if the field names are vague.

Sort Row
The Sort: row allows the choice of Ascending, Descending, and (not sorted). You can sort by an individual column in the data view, but if you want your data to maintain its sort level throughout the database you will need to use the sort row in the design view.

Show Row
The Show: row gives you the choice to display that field or not. You can apply criteria and sort orders to fields but not have it displayed in the query. For example, if you wanted all the "Florida" customers, but you didn’t want to see "Florida" repeated on every single record, you could set the criteria for State to "Florida" and uncheck the Show: box so it will not appear in your datasheet view.

Criteria Row
The power of queries lies in the Criteria: row. This is where you define what you are requesting from the data. For example, if you only want the appointments where Mr. Jones scheduled, we can set the criteria to ="Jones" for the last name field.

If this is a text field the quotes will be added automatically, and if you leave off a logic sign, an equal sign will be assumed, so for this criteria line we could have just typed Jones and gotten the same answer. Access looks at the field type to determine the format of the criteria. By default, text fields have quotes ("Text"), date fields the pound signs (#10/19/2011#) and numbers will be left alone.

In Access 2010, when you type in the criteria row you will get a helper that offers expression you may wish to use. You can ignore the helper, or use the quotes so it won’t pop up.

Logic Values
Access is flexible in how criteria are built; here is a list of the most common logic values:

<table>
<thead>
<tr>
<th>Logic Value</th>
<th>Meaning</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;</td>
<td>Less Than</td>
<td>&lt; 25 &lt;#1/1/2005#</td>
</tr>
<tr>
<td>&gt;</td>
<td>Greater Than</td>
<td>&gt;12 &gt;Now()</td>
</tr>
<tr>
<td>&lt;&gt;</td>
<td>Not Equal</td>
<td>&lt;0 &lt;&gt;&quot;Monday&quot;</td>
</tr>
<tr>
<td>And</td>
<td>Combinational And</td>
<td>&gt;12 And &lt;32 &gt;#1/1/05# And &lt;#2/1/05#</td>
</tr>
<tr>
<td>Or</td>
<td>Combinational Or</td>
<td>=&quot;M&quot; or =&quot;F&quot; =&quot;FL&quot; or =&quot;GA&quot;</td>
</tr>
<tr>
<td>Like</td>
<td>Used for non-exact values</td>
<td>Like &quot;A????&quot; <em>(Field must have five characters and start with the letter &quot;A&quot;)</em></td>
</tr>
<tr>
<td></td>
<td>Use ? to represent one Char</td>
<td>Like &quot;FL***&quot; <em>(Field must start with the letters &quot;FL&quot; but can be any length)</em></td>
</tr>
<tr>
<td></td>
<td>Use * to represent many</td>
<td></td>
</tr>
<tr>
<td>Not</td>
<td>Negates the Logic Values</td>
<td>Not =0 Not Like &quot;P***&quot;</td>
</tr>
<tr>
<td>Between/And</td>
<td>Finds a Range</td>
<td>Between 1 and 25 Between #1/1/06# and #2/1/06#</td>
</tr>
<tr>
<td>Is Null</td>
<td>Is Blank</td>
<td></td>
</tr>
<tr>
<td>Is Not Null</td>
<td>All non-blanks</td>
<td></td>
</tr>
</tbody>
</table>
Rows vs. Columns
As it goes across, Access will read the criteria rows as **Ands**, the columns as **Ors**.

This Query reads "From Patients Table, display the all the records where:

- City equals "Gainesville" **AND** Zip equals "32608"
- **OR** City equals "Waldo"

We will get all the Patients in who live in "Gainesville" and have a zip code of "32608", and in the same datasheet result, all the customers who live in "Waldo" no matter what their zip code.

**Working with Queries**
No matter how you initially create your query you can customize it in the design view.

**Joining Fields**
One of the basic design rules when planning your database is to break the data into its smallest logical parts. It is much simpler to join the fields together than it is to break them apart.

On the **field line**, we use the name of the fields and the joining character ampersand (&).

**Example:**

```
[FirstName]&[LastName]
```

We need to add a title at the front of the expression as well as a space between the two fields. We add a title in front of the field by putting any text we want within the naming convention of field names followed by a colon (:) and any extra text that should be added needs to be included in quotes (").

```
[FirstName]"&"[LastName]
```

**Missing Parameter Values**
Instead of changing the criteria every time we want a different set of data, we can set Access to prompt us for criteria but setting a missing parameter value. We do this by putting a field that doesn't exist into the criteria. Anytime we use a field outside of standing alone on the field line we need to put in it brackets [ ].
Since there is no field in any of our data sources called **Which city?**, so Access will prompt us with a window similar to the one shown here, asking for our input. As long as we get the spelling correct, it will return the values we need.

**Examples:**
- Between [Enter First Date] and [Enter Last Date]
- Like [First letters of Last Name]"*"

---

**Totals**

By default **Total** row does not appear in the design view of your query. To display this row, click on the **Totals** button on the toolbar (the sigma, looks just like the sum button from Excel) or right-click and choose **Totals**.

*Without Totals:*

*With Totals (group by):*

By default the fields are set to "Group By", which means each unique record for that data set is shown one time. We can add the Visit Date again, and do a "Count" to find out how many visits occurred for each day:

---

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 of each group
- **Last**: Find Last Value of each Group
- **Expression**: No Totals, just evaluate expression within the field
- **Where**: Find data based on this criteria, by default this will not show this field
Field Properties
The design view of Access Tables displays the properties of the fields at the bottom of the window. The other database objects hide their properties, but they can usually be found on the right-click (shortcut) menu. We may want to change the format of our numbers to only have two decimal places, or our times to exclude the seconds. By default the query will pull in the format settings from the table, but we always have the option to reformat within each database object.

Example Query:

![Example Query Image]

Result:

![Result Image]

To format the Average of the Length field we have to see the properties of that field. In the design view of the query, right-click on the field and choose Properties.

This will open the Property Sheet. The Tables and Field Properties handout explains how the number formats behave. The Decimal Places property setting has no effect if the Format property is blank or is set to General Number. Standard is equivalent to Excel's Comma Style.
**Expressions**

One of the basic design rules when planning your database is to make sure you are not using any calculated or derived fields. We can do math within your queries, forms, and reports. We have seen that Access does some math for us with the Totals line, but we can build our own custom expressions.

As with joining fields together, titles of the column are text followed by a colon (:) and all fields should be in square brackets ([ ]). Access, like Excel, will follow the order of operations (parenthesis first, then powers, then multiplication/division, then addition/subtraction) so remember to use proper parenthesis placement.

Sample Calculated Fields:
- Total: [AmtSold] * [TotalPurchased]  # Calculate Amount Sold multiplied by Total Purchased
- Taxes: [Total] * 0.0625  # Calculate Total multiplied by 0.0625
- GrandTotal: [Total] + [Taxes]  # Calculate Total plus Taxes

* The above samples build on top of each other, if you plan to do this make sure you save the query before you run it, otherwise it may not recognize the fields.

Sample Derived Fields:
- LastFour: Right([SSN],4)  # Find last four characters of the SSN
- MiddleInit: Left([MiddleName],1)  # Find the first character of MiddleName

Sample Logic Fields:
- iif([Class] Is Null, "Unknown", [Class])  # If Class is blank, put in "Unknown"
- Taxes: iif([Exempt]=Yes, 0, [Price]*0.0625)  # Calculate the taxes on nonexempt items

Most of the expressions you use in Microsoft Excel can be used here in Access. For example, Average in Excel is Avg in Access, and IF in Excel is IIF in Access.

There is an Expression Builder to help you create your expressions. Right-click in the field box, in the design view of a query, and choose Build...

On the left hand side there is a list of database objects. Double-click on the Functions, and open the list of Built-In Functions. From here you can get more details on any function.

In Access 2010, you can create a calculated field in a table. You would use this same expression builder.
Class Exercise

1. View Tables

2. Table Filters
   - Open Patients
   - Click on Filter Arrow for City
     - Uncheck Gainesville, OK .............................................. (21 records)
   - Right-click Micanopy
     - Does Not Equal "Micanopy" ........................................... (12 records)
   - Toggle Filter ............................................................... (76 records)
   - Select & Right-click on AVE in an address
     - Contains "Ave" ............................................................. (13 records)
     - Close and Save
   - Open Patients ............................................................. (76 records)
   - Right-click Micanopy
     - Does Not Equal "Micanopy" ............................................ (12 records)
   - Toggle Filter ............................................................... (76 records)
   - Select & Right-click on AVE in an address
     - Contains "Ave" ............................................................. (13 records)
   - Close and Save

3. Create Query in Design view, Patients
   - Double-click Med Rec #
   - Double-click Last
   - Drag and Drop First Before Last
   - Change next blank field to Address from drop down list
   - Click on City, Ctrl click on Zip
     - Click in an empty space next to table to drop selection
   - Click on City, shift click on Zip,
     - Drag all three to next blank
   - Datasheet View ............................................................ (76 records)
     - Sort by Last Ascending
     - Sort by City Descending
     - Clear Sort
   - Design View
     - Sort by Last Ascending, City Descending
     - Make sure City is before Last
   - Close and save as Patients Sort 1

4. Create Query in Design view, Patients
   - Move * field down (double-click or drag)
   - Move down City Field
     - Sort by City Descending
   - Datasheet ................................................................. (76 records)
     - notice 2 City Fields (Patient.City and Field0)
   - Design View
     - Don't Show City
     - Sort by Last within each City
   - Close and save as Patients Sort 2
5. **Create Query with Simple Query Wizard**
   - From Table Patients, Bring in all fields
   - Name: *Patients Query*
   - Design View
     ~ Notice brackets around field names
   - Close Query

   - Right-click on Patients Query
     ~ Choose Design View
     ~ Notice brackets are gone
   - Criteria
     ~ City = Waldo ............................................ ( 9 records)
   - Look at SQL View
     ~ SELECT Patients.[MED REC], Patients.[LAST], Patients.[FIRST], Patients.[ADDRESS], Patients.[CITY], Patients.[ST], Patients.[ZIP]
       FROM Patients
       WHERE ((Patients.[CITY])="waldo");
   - Criteria: City = Waldo OR Starke ......................... (12 records)
     • OR – Separate lines
   - Sort by City
   - Clear and set new Criteria
     ~ Zip = 32608 ............................................ (12 records)
     ~ AND City= Gainesville .................................. ( 9 records)
       • Same Lines
     ~ (Gainesville AND 32608) OR Waldo .................... (18 records)
   - Press F12 to Save As...
   - Save as *Patients Criteria - Exact*
   - Close Query

6. **Create Query in Design view, Patients**
   - Double-click on title of the Patients table and move down all fields
   - Set Criteria for City: [Which city?]
   - Datasheet View
     ~ Use Refresh button to try another city
       • Gainesville ............................................ (55 records)
       • Micanopy ............................................. ( 9 records)
       • Waldo ................................................ ( 9 records)
       • Starke ............................................... ( 3 records)
   - Close and Save as *Patients by City*
   - Open Query to test missing parameter value
7. Create Query in Design view, Patients
   - Double-click on title of the Patients table and move down all fields
   - Set Criteria
     ~ Range Set criteria for Zip
       • >=32600 OR <= 32699 ....................... (76 records)
       • >=32600 AND <= 32699 .................... (51 records)
       • Between 32600 And 32699 ............... (51 records)
     ~ Wild Cards criteria for Last:
       • j* (Like "j*") ........................................ (10 records) Begins with J
       • *s (Like "*s") ...................................... (21 records) Ends with S
       • j*s (Like "j*s") ................................... (6 records) Begins J & Ends S
       • *e* (Like "*e*") .................................. (42 records) Contains E
       • Like "?e*" ........................................... (11 records) 2nd char is an E
     ~ Criteria for Address
       • Like *box* ........................................ (14 records)
       • Like *box* ........................................ (12 records) with space
       • Not Like *box* ................................. (64 records)
     ~ Final Criteria
       • Address: Not Like *box *
       • Zip: Between 32600 and 32699
       • Last: Like "e*" ..................................... (21 records)
   - Save as Patients Criteria – Range

8. Create Query in Design view, Patients
   - In the first blank field blank type: [First]&[Last]
   - See Datasheet view
   - Design view, Expression 1
     ~ Expr1: [First] & [Last]
     ~ Right-click on expression choose ZOOM
       • Change Font... to comfort level (~14)
       • Patient Name: [First] & " " & [Last]
   - Close and save as Patient Names

9. Create Query in Design View, Employees
   - Fields:
     ~ EmployeeFirstName
     ~ EmployeeLastName
     ~ EmpHoursWorked
     ~ EmpHourlyRate
   - Right-click next blank column, choose BUILD
     ~ Expression Builder Blank, Cancel
     ~ Save Query “Employee Pay”
   - Try Expression Builder again
     ~ Double-click EmployeeLastName
       ~ Type: & ", " &
     ~ Double-click EmployeeFirstName
   - View Datasheet
- Rename Expr1 to EmpName and move to the first column
- Delete EmployeeFirstName & EmployeeLastName from bottom of the Query Design

```
<table>
<thead>
<tr>
<th>EmpName: [EmployeeLastName] &amp; &quot;&quot;, &quot; &amp; [EmployeeFirstName]</th>
<th>EmpHoursWorked</th>
<th>EmpHourlyRate</th>
</tr>
</thead>
</table>
```

- **GrossPay** 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)
  ~ Click OK and View Data

- **Deductions** In the next blank column Type -- Deductions:0.175
  ~ Format properties to Percent (right-click to get property window)
  ~ Change to Deductions: .175*[GrossPay]
    - Use expression builder, don't forget to save.
    - Right-click Properties, Format -> Currency

- **NetPay** In the next blank column
  ~ NetPay: [GrossPay]-[Deductions]

- Close and save query

10. **Create Query in Design view Doctors**
- Bring down DocFirst........................................... ( 3 records)
- Add Table: Visit Info
  ~ Datasheet view ........................................... (162 records)
- Turn on Totals .............................................. ( 3 records)
  ~ Turn off Totals ........................................... (162 records)
- Add a field: Length
  ~ See 5 different visit times (15, 30, 45, 60, 75)
- Turn on Totals
  ~ 3 doctors, 5 lengths ....................................... (15 records)
  ~ Change Length's total to SUM............................. ( 3 records)
~ Change Title of Sum of Visit to Total Time
- Add a field: Length (again)
~ Average new Length of Visit
~ Title Average Time
~ Format Field, Standard
- Close and Save as *Doctors Time Summary*

11. Create Query in Design view, Doctors and Reasons
- Note the lack of relationship
- Add Table Visit Info
- Bring Down fields .................................... (162 records)
  ~ Doctors: DocFirst
  ~ Reasons: Reasons
  ~ Visit Info: Length
- Turn on Totals ........................................ (25 Records)
  ~ Group by Doctors
  ~ Group by Reasons
  ~ Sum Length
- Turn off Totals ........................................ (162 records)
- Close and Save as *Doctors by Reason*

12. Create Query using Cross Tab Query Wizard based on Doctor by Reason
- Wizard (See [Page 3](#))
  ~ Row Headings: Reasons
  ~ Column Heading: Doctors
  ~ Calculating *Sum* of Length
- Turn on Totals in Datasheet view

13. Create Query using Cross Tab Query Wizard based on Doctor by Reason
- The same but for the *Average* of Length