1. Queries

One of the most useful features of a database is that you can select exactly the data you want. You can filter the data in a table in Datasheet view, but the advantage of a Select Query is that you can save it and use it whenever you need it. There are a number of different types of Query, but the most common type is a Select Query. As the name implies, this type of Query selects the data you specify. You can use a Select Query as the data source for reports and forms.

The benefits of a Select Query include:

1. Displaying only the records you want.
2. Displaying only the fields you want.
3. Displaying the top number of records eg the top 10 items by Value
4. Creating calculated fields eg Quantity * Cost to show the Value of stock.
5. Using Parameters to greatly extend the flexibility of queries.

Other types of Query can delete, change and append records.

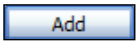
Although there is a 'Query Wizard' in Access, most people find it just as easy to work in Design View without using the wizard so the following section is concerned with creating a Query in Design View.

In general terms, the steps you need to follow are:

1. Tell Access which tables you want to extract data from.
2. Define the fields you want to work with.
3. Tell Access what you want it to do with the fields, e.g. sort in a certain order, or select according to given criteria.
4. Test to check that the query does what you want it to do.
5. Save the query.

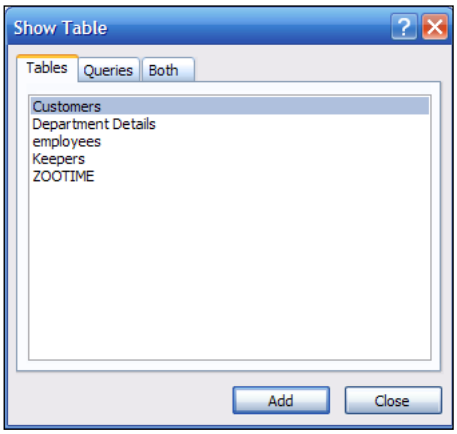
1.1. Creating a Select Query in Design View

- Click 

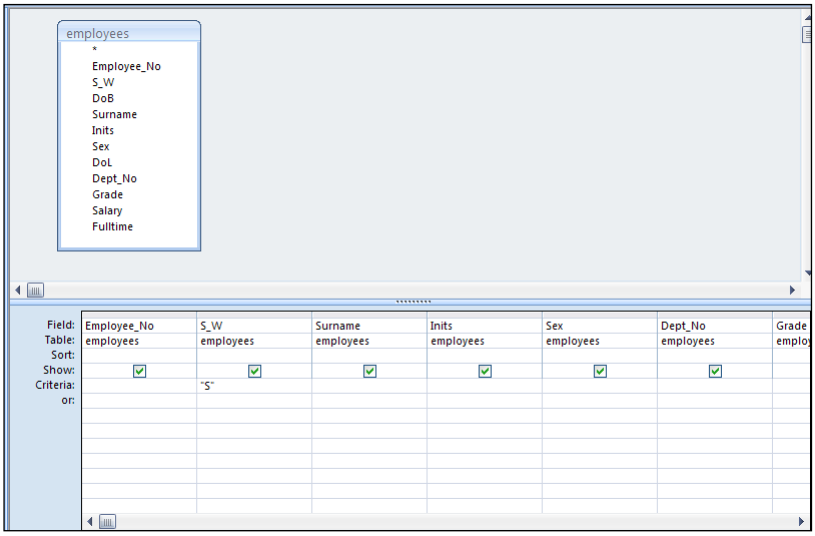
in the *Other Section* of the *Create Ribbon*.
- Select the required Table(s) or Query (ies) from the *Show Table Panel*.
- Click the 

button.
- Click the 

button when you have added all the tables and queries you need.



The *Show Table* window closes leaving the Query window open as shown below:



The query window is driven by a method sometimes called 'Query-By-Example' (QBE). The area at the bottom of the window, as seen above, is the 'QBE grid'**.** This is where you need to enter your selection criteria and any sorting details.

The QBE row details are as follows:

Heading	***Description***
Field:	The name of the field which you would usually display when you run the query and on which you may select or sort.
Table:	The name of the table or query from which the selected field comes.
Sort:	Ascending, or Descending or leave blank.
Show:	A tick in the Check Box indicates the field will display when you run the query. Remove the tick from the Check Box if you do not want to display the field.
Criteria:	Enter the selection criteria. For example, to select all people with surname Smith, you would type ***Smith***. Of course, the entry depends upon the field.
or:	Used to extend the criteria

1.1.1. Defining Query Fields

There are a number of ways of defining the required fields:

- Double-click the required field in the Table 'frame' towards the top of the window. The new field appears in the next available blank column in the QBE grid at the bottom of the screen. Or…
- Click and drag the required field from the Table frame to the next available Field cell in the QBE grid. Or…
- Click the next available Field cell in the QBE grid then click on the right of the cell and select the required field.

1.1.2. Comparison or Relational Operators

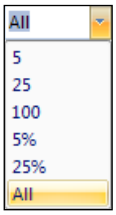
Queries often use Comparison or Relational Operators to define selection criteria. For example, you may wish to select employees who earn more than £25,000. This is done by using one of the comparison operators such as **>** (greater than).

Operator	Use	Example
=	Equals	***="Smith"*** - the name Smith.
>	Greater than	***>200***, or ***>"S"*** - Greater than 200, or later in the alphabet than s.
<	Less than	***<200*** - Less than 200.
>=	Greater than or equal to	***>=200*** - Greater than or equal to 200.
<=	Less than or equal to	***<=200*** - Less than or equal to 200.
<>	Not equal to	***<>25000*** - All values except 25,000.
Between	Between two values	***Between "A" and "M"*** (inclusive) - "Between" is also very useful for numeric and date fields.
Is Null	An empty field	***Is Null***
Is Not Null	A field containing data	***Is Not Null***
Like	Pattern match	***Like "Smi*"*** - Where the first 3 letters are Smi.

1.1.3. Logical Operators

Operator	Use	Example
AND	Both criteria are true	***>200 and <300*** - Greater than 200 and less than 300.
OR	One or the other criteria is true	***"Smith" or "Jones"***
NOT	The criteria is not true	***Not "Smith"*** - Everything except Smith.

1.1.4. Defining the Number of Records eg Top 10



For example, you may want to display the 10 employees with the highest salary…

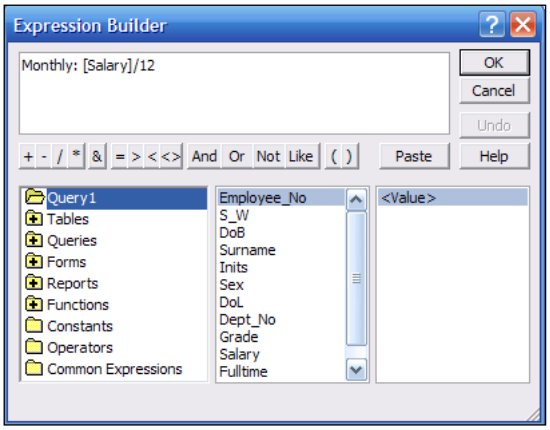
- Sort the data into the required order by clicking the Sort cell in the QBE grid of the relevant field (*Salary* for this example) then clicking on the right of the cell and select *Descending*.
- Use the Combo Box to the right of the *Return:* button in the *Query Setup* section of the *(Query Tools) Design Ribbon* to select the number of records you want.

> **The values are the "presets" but you can type any number you want. For example, type 10 to display the Top 10.**

1.1.5. Calculated Fields

You should not include calculated fields in a table as there is no foolproof way of getting Access to update the field. However, it is easy to add calculated fields to a Select Query which recalculates the values whenever you run the query. To add a calculated field…

- Click in the *Field:* cell in a blank column. (You may need to insert a column so that the calculated column appears in the correct place).
- Write the name you want to assign to the calculated column. If you forget this, Access assigns the default name of *Expr1:*
- Write the required expression. This might be easier if you use the Builder button to display the *Expression Builder Panel* which allows you to select the fields and arithmetical symbols, and inserts some of the required punctuation such as square brackets around the field names.



The Expression Builder Panel illustrated shows the expression Monthly: [Salary]/12 which divides the (annual) salary by 12 to calculate monthly pay.

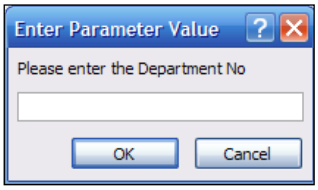
1.1.6. Parameter Queries

You may want to run a query for an employee file for only one department. Although you could have a query for every single department, this would mean that you would have a large number of almost identical queries in a large organisation. It would be difficult to locate the right one in such a long list, and awkward to work with. The solution is to create a Parameter Query which prompts the user to type a value…

- Click in the *Criteria:* cell in the relevant column.
- Enter the prompt that you want the user to see eg:

  [Please type a Department Number]

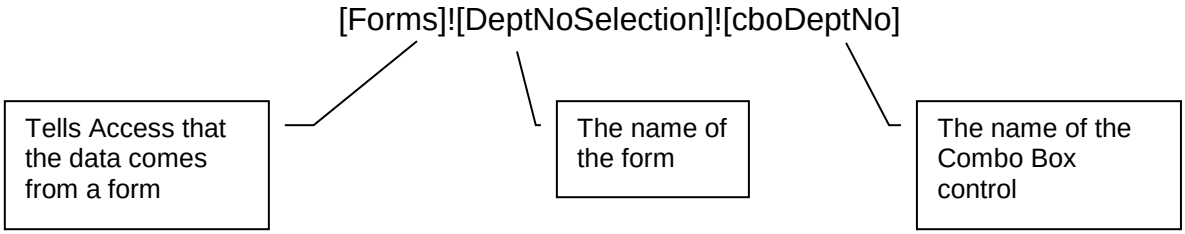
  Remember to use square brackets at the beginning and the end of the phrase.
- When you run the query the panel prompting the user will appear.



If the list of values is short, you may be able to include the values in the prompt.

1.1.7. Parameter Queries Related to a Form

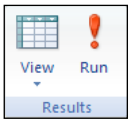
Unless the list of values is so short that you can include them in the prompt, there is no way to give the user a list of the valid values within the query, but the parameter can be related to a field such as a Combo Box in a form. Instead of running the query directly, the user runs the form, selects a value using the form and Access uses this value automatically. Forms are covered in a later section of this document but an example of the parameter entry in the query could read…



Like the calculated fields described in Section 7.1.5. above, this is much easier to write if you use the Expression Builder.

1.2. Running and Testing the Query

When you are satisfied you have specified your selection and output criteria correctly, run the query:



- Click either *View* or *Run* from the *Results Section* of the *(Query Tools) Design Ribbon*.

Your Query results will be displayed in *Datasheet View*, very much like a table. If you want to change the Query after viewing your output:

- Click 

in the *Views Section* of the *Home Ribbon* to return to *Design View*.

1.3. Saving the Query

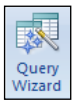
- Click 
- Type the name you want to give the Query, overtyping the default name (such as Query1).

It is good practice to name objects in such a way that you (and other people) can later identify them easily. Many databases have a large number of queries, (and tables, reports and forms). It is very difficult to work effectively with lists of queries called Query1, Query2 and so on. Always try to use meaningful names for your queries and other Access objects to help you recognise easily what each one does.

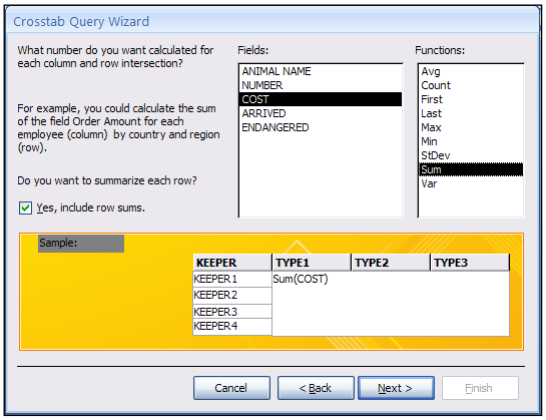
One of the most important advantages of Queries is that they can be used as the basis of reports and forms.

1.4. Creating a Crosstab Query

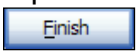
As it is usually easy to create a Select Query, there is often no benefit in using the Query Wizard to create such a query. However, it is helpful to use the Query Wizard if you want to create a Crosstab Query.

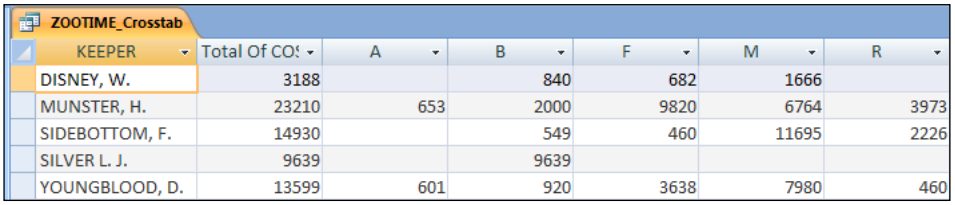
- Click 

in the *Other Section* of the *Create Ribbon.*
- Select *Crosstab Query Wizard* then as prompted by the wizard, select:
  a) A table or query
  b) One or more field(s) for Row Headings
  c) A field for Column Headings
  d) The field you want to summarize as illustrated on the right.



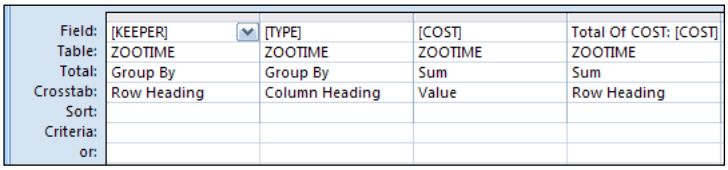
The fields you use as Row and Column Headings must be "repeating" fields where lots of records have the same value.

- On the final panel of the wizard, change the default name of the query if you wish then click 



KEEPER	Total Of COS	A	B	F	M	R
DISNEY, W.	3188		840	682	1666	
MUNSTER, H.	23210	653	2000	9820	6764	3973
SIDEBOTTOM, F.	14930		549	460	11695	2226
SILVER L. J.	9639		9639			
YOUNGBLOOD, D.	13599	601	920	3638	7980	460

The completed Crosstab Query and below, the corresponding Query Design screen.



Field:	[KEEPER]	[TYPE]	[COST]	Total Of COST: [COST]
Table:	ZOOTIME	ZOOTIME	ZOOTIME	ZOOTIME
Total:	Group By	Group By	Sum	Sum
Crosstab:	Row Heading	Column Heading	Value	Row Heading
Sort:				
Criteria:				
or:				

- Close the Query.

You have already told Access the name of the query on the final step of the CrossTab Wizard.