

Section 6

Queries

By the end of this Section you should be able to:

Create, Edit and Delete a Query

Use Sort in a Query

Print Query Results

Query Related Tables

Use Value Ranges in Queries

Use a Query to Find Non-Matches

Use AND & OR Queries

To gain an understanding of the above features, work through the **Driving Lessons** in this **Section**.

For each **Driving Lesson**, read the **Park and Read** instructions, without touching the keyboard, then work through the numbered steps of the **Manoeuvres** on the computer. Complete the **Revision Exercise(s)** at the end of the section to test your knowledge.

Driving Lesson 51 - Querying a Table

P Park and Read

A database management system has the ability to process the information the user has entered and produce meaningful results. Access uses a system known as **Querying**. A query is used to extract data from a table and analyse that data. Simply create a **Query**, selecting which fields to show in the answer and which records to include. The program then processes the data and performs the query.

Once created, a **Query** can be saved and run whenever required. It will always show the latest data. For example suppose a query is created to list all your customers in a certain town. If new customers are added in that town, the query will immediately reflect this, it does not need to be updated. New fields can be added to queries and existing fields can be removed.

Queries can also be used in place of a table in many instances, e.g. as a basis for **Forms** and **Reports**.

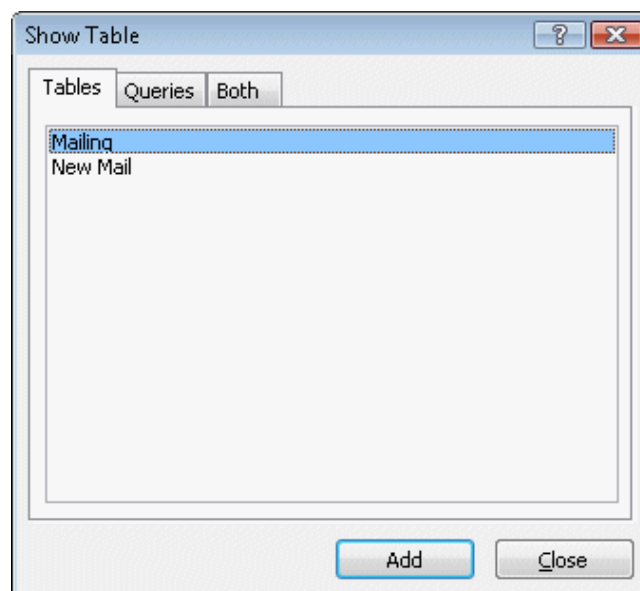
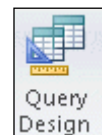


There are several types of Query. This guide will only look at simple Select Queries.



Manoeuvres

1. Open the database **Mailing**, select the **Create** tab and click **Query Design** in the **Other** group. The **Show Table** dialog box is displayed and shows a list of available tables.

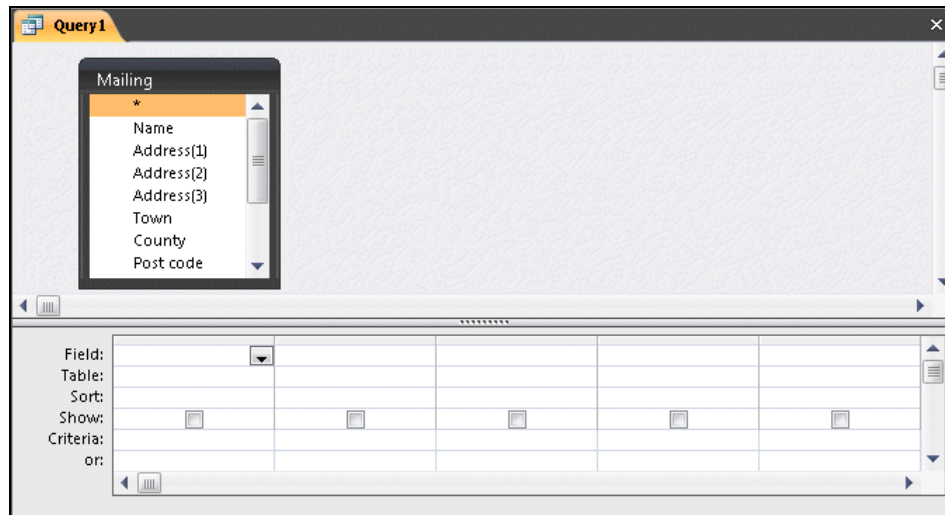


2. Select **Mailing** as the required table, click **Add** and then **Close**.

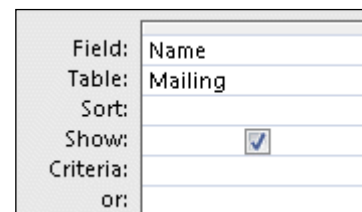


Driving Lesson 51 - Continued

3. The **Select Query** window is now visible. Notice the **Field List** in the top half of the window which shows the available fields, and the **Query Grid** in the bottom half which shows which fields will be included in the output.

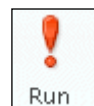


4. There are several ways to move fields from the **Field List** to the **Query Grid**. In the **Field List**, click on **Name** and drag it down to the **Query Grid**. Release it anywhere in the first column.
5. The **Show** box should be checked (have a tick in it) to indicate that this field is required in the output list.



It is possible to use a field in a Query for selection purposes without showing it in the output.

6. To run the query, display the **Design** tab and click the **Run** button. A results table appears. In this case it only consists of the **Name** field because that was the only field placed on to the **Query Grid**.



7. The query results table act exactly like any other table. Use the navigation buttons to navigate between the records as before.



8. To switch back to the query design, make sure the **Home** tab is displayed and click the **Design View** button.
9. In the second column of the **Query Grid**, display the drop down list in the first field and select **Department** from the list of available fields.
10. Click the **Run** button. Both **Name** and **Department** will now be listed. Switch back to the query design, using the **Design View** button and leave the query open for the next Driving Lesson.



Driving Lesson 52 - Selecting in Queries

P Park and Read

One of the main uses of a query is to select certain records from a table based on entered criteria.

Manoeuvres


1. In the query from the last exercise, double click on **Organisation** in the **Field List**. The field is added to the next empty column in the **Query Grid**.
2. To select records for only certain organisations, click in the **Criteria** field for this column, type **school** and press the <Enter> key.

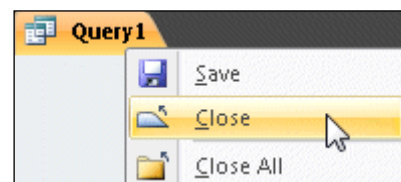
Field:	Name	Department	Organisation
Table:	Mailing	Mailing	Mailing
Sort:			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:			"school"
or:			

3. Notice the quotes around the **school** criteria. Access often changes the criteria to its own language. Run the query. The query lists the three requested fields for all records with an **Organisation** value of **School**.



Querying is not case sensitive. Searching for Fred will also find FRED and fred.

4. Use the **Design View** button to switch back to query design.
5. Click in the **Organisation** column of the **Query Grid** and from the **Design** tab click . The column is removed.
6. Remove the **Department** column from the **Query Grid** in the same way.
7. Use any method to place the fields **Town** and **Post code** into the second and third columns of the **Query Grid**.
8. In the **Criteria** of the **Town** field, type **London**. Run the query. It lists name and postcodes of people living in London. There are **2** records.
9. Right click on the tab header for this query and select **Close**.
10. A message will be displayed asking if you want to save the changes to this query. Click **No**. The query will be closed without saving and therefore cannot be used again.
11. Close the database.




Driving Lesson 53 - Sorting Query Results

P Park and Read

Query results may be easier to understand if they are placed in a certain order or sorted. It is also a useful tool when query results are used in reports.



Manoeuvres

1. Open the database **Houses**, select the **Create** tab and click **Query Design**. The **Show Table** dialog box is displayed and shows a list of available tables.
2. Add the **Houses** table to the query and close the dialog box.
3. In the **Field List** for **Houses**, click on **Town**. Move down the list, hold down **<Shift>**, and click on **Price**. Five fields should now be selected. Click in the selection and drag down to the first column of the query grid, then release the mouse button. All of the fields should be visible in the query grid with their **Show** boxes checked.
4. Delete the column for **Occupied**. then in the **Criteria** for **Town**, enter **Cockermouth**. Run the query. There are **5** results.
5. To sort these in order of price, first use the **Design View** button to switch back to query design. 
6. In the **Sort** area for **Price**, drop down the list.

Field:	Town	Address	Type	Price
Table:	Houses	Houses	Houses	Houses
Sort:				
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:	"Cockermouth"			Ascending
or:				Descending
				(not sorted)

7. Select **Ascending** and run the query again. Records for properties in Cockermouth are listed in ascending order of price.
8. Switch back to **Design View** again, add **Bedrooms** to the **Query Grid** and delete the contents of the **Criteria** field for **Town**.
9. In the **Criteria** for **Bedrooms** enter **2** and sort the **Price** field in **Descending** order.

Field:	Town	Address	Type	Price	Bedrooms
Table:	Houses	Houses	Houses	Houses	Houses
Sort:					
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:					2
or:				Descending	
				(not sorted)	

10. Run the query - the table shows 9 houses with exactly two bedrooms, sorted by price, most expensive first. Leave the query open.

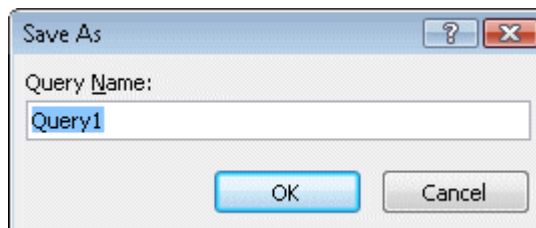
Driving Lesson 54 - Saving Query Results

P Park and Read

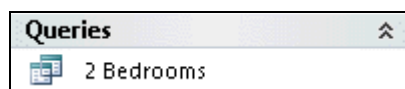
Queries can be saved so that when data changes, the query can be run again to reflect the changes, without the necessity of redesigning it.


Manoeuvres

1. Right click on the tab header for the query created in the previous Driving Lesson and select **Save**. The **Save As** dialog box is displayed.



2. Replace the text **Query1** with **2 Bedrooms** and click **OK**. The query is saved and the title is shown in the tab header for this query. The **Navigation** pane now shows the saved query **2 Bedrooms** (if **All Access Objects** is selected).



3. Return to **Design View** and sort the **Town** field as **Ascending** leaving all other criteria as they are. Run the query.
4. Notice that the first sort is alphabetically by town. The second sort is on price in descending order, so within each town group, the records are sorted with the highest price first.
5. Return to **Design View** and change the **Town** field sort to **Descending**. Run the query to see the towns in reverse alphabetical order.
6. Use the **Close** button, , to close the query. At the dialog box select **No**. The query is closed without saving and the recent changes are lost.
7. Double click on the **2 Bedrooms** query from the **Navigation** pane to open it. It is the version that was saved at step 2.
8. Leave the query open for the next Driving Lesson.


Driving Lesson 55 - Printing Query Results

P Park and Read

When a hard copy of query results is required, it is a simple matter to print out the results.


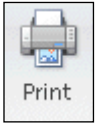


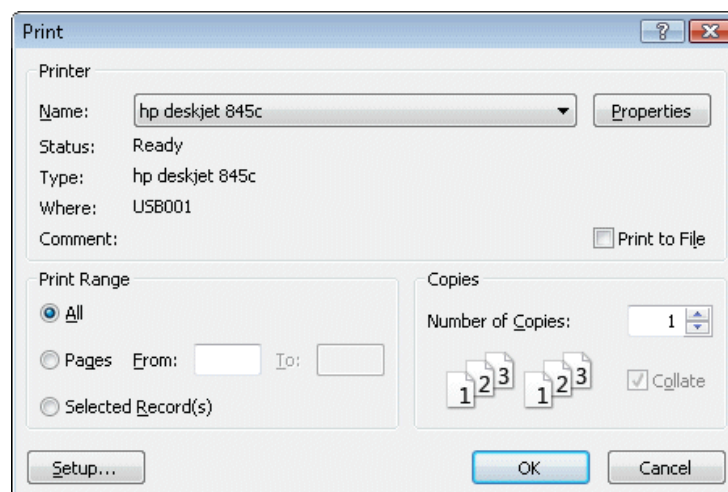
Manoeuvres

1. The **2 Bedrooms** query is open from the previous Driving Lesson. To see a preview of how the query will appear when printed, click the **Office Button** and move the cursor over **Print** to display printing options. 



Select **Quick Print**, from the **Office Button** menu to print directly without displaying the **Print** dialog box.

2. Select **Print Preview** from the options. A **Print Preview** tab is displayed in the **Ribbon** and the query results are displayed as they will print.
3. Note that the print has a header showing the title of the query and the date, and a footer showing the page number. Zoom in to the page by clicking on it with the magnifier.
4. Click the **Landscape** button on the **Print Preview** tab to change the orientation of the print on the page. 
5. Click the **Print** button, or use the key press **<Ctrl P>**, to open the **Print** dialog box. The dialog box has options to print selected parts of the query result, in exactly the same way as for a table. There is also an option to print more than one copy. 



6. Click **OK** to print a single copy of the whole list.
7. Use the button on the **Ribbon** to **Close Print Preview**.
8. Close the **2 Bedrooms** query and close the database.

Driving Lesson 56 - Querying Related Tables

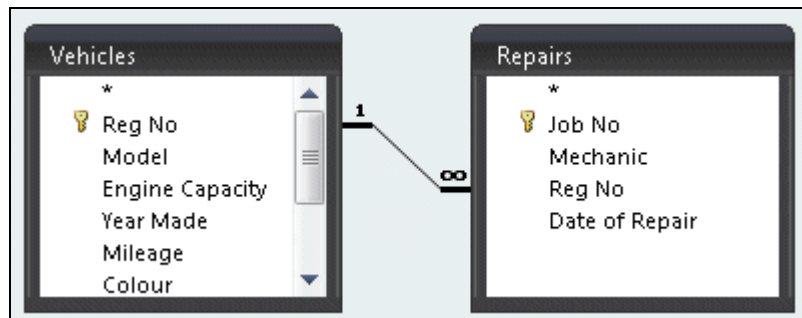
P Park and Read

The technique for querying related tables is exactly the same as for one, except that at the **Show Table** stage, more than one table may be added.

Fields can then be chosen from both tables to give a result that would not be possible from an individual table.

Manoeuvres

1. Open the database **Daley**.
2. Display the **Create** tab and select **Query Design**.
3. The **Show Table** dialog box lists all available tables. Double click on **Vehicles** and then on **Repairs** to add them to the query. Close the **Show Table** dialog box.
4. The relationships between the tables were set in Driving Lesson 28 and are shown here.



If Driving Lesson 28 has not been completed and there is no existing relationship, it can be defined on this screen but will only apply to this query.

5. From the **Vehicles** field list add the following fields: **Reg No**, **Model** and **Manufacturer**.
6. From the **Repairs** field list add the following fields: **Mechanic** and **Date of Repair**.

Field:	Reg No	Model	Manufacturer	Mechanic	Date of Repair
Table:	Vehicles	Vehicles	Vehicles	Repairs	Repairs
Sort:					
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:					
or:					

7. Note that the **Table** name for each field is displayed in the query, indicating the source of each of the fields.



Driving Lesson 56 - Continued

8. In the **Mechanic** criteria enter **David**. This will display only David's jobs. Run the query. There are **2** records.
9. There really is no need to display David's name in the result as all the information applies to him. Switch back to **Design View** and click in the **Show** box for **Mechanic**. The tick is removed.

Mechanic	<input type="checkbox"/>
Repairs	<input type="checkbox"/>
	<input type="checkbox"/>
"David"	<input type="checkbox"/>

10. Run the query again. The results are the same, except that David's name is not shown.



The field has not been removed from the query. It can still be used for selection or sorting purposes.

11. To 'unhide' the field, return to **Design View** and replace the tick in the **Show** box. Run the query again, the name will be shown.
12. Return to **Design View**, remove the tick from the **Show** box in the **Mechanic** field and save the query as **David**.
13. Run the query and **Print Preview** the query results. David's name will appear as the title.
14. Print a copy.



When a query includes data from 2 tables, records will only be included in the results where there is corresponding data from both tables. In the example above, if there is a vehicle that has no repair jobs, it will not appear in the results.

15. Return to **Design View**, move the cursor above the **Date of Repair** field until the cursor changes to a black downward arrow, and click once to select the whole column.
16. Press the <**Delete**> key to remove the field. It can no longer be used for selection or sorting. Run the query to see the effect.
17. Close the query without saving but leave the database open.

Driving Lesson 57 - Editing Queries

P Park and Read

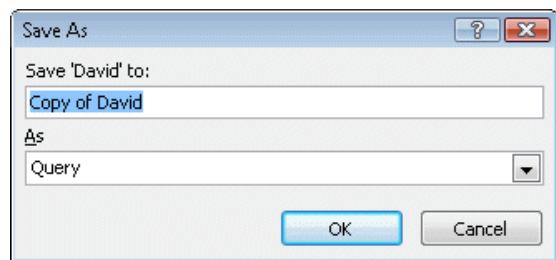
There may be occasions when several similar queries are required. It is usual in these circumstances to edit the original query, rather than set up and save new ones. For example, if in an address query the only criterion that changes is the beginning of the postcode, then it is acceptable to run the query for one postcode, print or preview the result, before editing the query for another postcode. The amended queries can be saved with a different name if required.

Fields appear in the results in the order they appear in the Query Grid and they can be moved to a more suitable position in the grid if required.

Manoeuvres

1. The **Daley** database is open, open the **David** query.
2. Switch to **Design View**. Exactly the same information is required for Richard. Change the criteria for the **Mechanic** to **Richard**.
3. Run the query. This time Richard's details are shown. If a printout is required however, the printout will still show David as the title.

4. Click the **Office** button and select **Save As**.
5. Change the name from **Copy of David** to **Richard** and click **OK**. A second query is saved with a name of **Richard**.



6. Switch to **Design View**. It makes more sense to have the Manufacturer's name before the Model name. Move the cursor above the **Manufacturer** field until the cursor changes to a black downward arrow, and click once to select the column.
7. Point the cursor in the narrow black area above the field name until it changes to a white arrow and drag the field to the position between **Model** and **Reg No**. A thick black line will help you.

Field:	Reg No	Manufacturer	Model	Mechanic
Table:	Vehicles	Vehicles	Vehicles	Repairs
Sort:				
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Criteria:				"Richard"
or:				

8. Once the mouse button is released the field is dropped in its new position.
9. Run the query then close it without saving but leave the database open.

Driving Lesson 58 - Ranges of Values and Wildcards

P Park and Read

When you want to perform a query that does not contain precise values, or contains an unknown element, you can use **ranges of values** and **wildcards**. The following search rules apply to queries run on single or related tables.

A range of values can be specified in a query using the following operators:

>	Greater than	<	Less than
>=	Greater than / equal to	<=	Less than / equal to

These operators allow the user to process data depending on its order. As with exact matches, the required range is typed into the appropriate field of the query grid. For example, to discover which items, if any, are in short supply in a table of stock records, the user could type <**100** in the **Stock Level** field of the query grid to display all the items with less than 100 in stock.

Wildcards can be used in queries to search for patterns in data, e.g. all employees whose last names begin with **Mac** or end with **son**. The following wildcard characters are used:

* to replace one or more characters (% if using ANSI-92 SQL)

? to replace a single character (_ if using ANSI-92 SQL)

For example, wildcards could be used to select 'Sales' people from their job titles, which may be sales assistant, sales supervisor, etc.



Manoeuvres

1. Use the database **Daley** and create a query using the **Vehicles** table.
2. Place the **Reg No**, **Model** and **Price** fields on to the query grid.
3. To find all the cars costing less than £3000, type <**3000** in the criteria box of the **Price** field. Make sure that the **Show** boxes are checked.

Field:	Reg No	Model	Price
Table:	Vehicles	Vehicles	Vehicles
Sort:			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:			< 3000
or:			

4. **Run** the query. The answer table should contain the vehicles costing less than £3000 and display their **Reg No**, **Model** and **Price**. Click on **Design View** to return to the **Query Grid**.



Driving Lesson 58 - Continued

5. Multiple selection criteria can be entered in the same query. Add the **Previous owners** field to the query grid. To look for all cars with four or more owners, and which also cost less than £3000. Type **>=4** in the criteria box for **Previous owners**.
6. Run the query. This is a multiple query, it has more than one selection criteria. Because both criteria must be met for a record to be selected, this is an example of an AND query.
7. Save the query as **Owners** and click **OK**. Close the query and the database.
8. Open the **Houses** database and create a new query in **Design View** based on the **Houses** table.
9. Place the **Town**, **Address**, **Type**, **Price** and **Comment** fields into the query grid.
10. We need to search for properties where **detached** or **semi-detached** is a feature in the **Type** field. The properties may be houses, bungalows, etc. The text may appear in a variety of forms, e.g. **semi-detached house**, **detached bungalow**, and may be in the middle of other text.
11. In the criteria row for the **Type** field, type ***detached***. This is enough information to find all required occurrences because the wildcard characters, (*), denote that any amount of text may occur before or after the selected characters (If this doesn't work, use **%detached%**).
12. Press **<Enter>** and note that Access changes the criteria to **Like " *detached* "**, which is the code it uses to perform a wildcard search.
13. Run the query. Records are found with a variety of relevant **Type** fields.

Town	Address	Type	Price	Comment
Cockermouth	8 Blackpool Court	Detached House	£38,000	Quiet Road
Aspatia	15 Waldram Way	Semi-Detached House	£39,000	Quiet Road
Great Broughton	The Castle	Detached House	£47,000	Conservatory
Workington	1 Whitburn Street	Semi-Detached House	£17,000	Includes Carpets
Whitehaven	9 Deepdene	Detached House	£49,950	Listed Building
Whitehaven	25 Lilac Close	Detached House	£58,950	Panoramic Views
High Seaton	6 Stonehills	Detached Bungalow	£55,000	Inc. Carpets & Curtains
Workington	265 Harrison Road	Semi-Detached House	£16,500	Poor Decorative Order
Stainburn	138 Swaledale	Detached House	£47,500	Includes Carpets
Workington	21 Chapel Road	Semi-Detached House	£36,950	Well Maintained

Record: 1 of 10 | No Filter | Search

14. Switch back to **Design View**. Delete the criteria in the **Type** field and type ***r?ad** in the criteria for the **Address** field.
15. Run the query. All records where the address ends in **road** (or read or riad, etc.) will be listed. (If this doesn't work, use **%r_ad**).
16. Close the query without saving and close the **Houses** database.

Driving Lesson 59 - Non-Matches

Park and Read

In order to produce results which exclude the matches made using other searches, the user may precede any criteria with the operator <> or **NOT**, as in not London, or not *sales*, or <> Scotland.



Manoeuvres

1. Open the database **Houses** and create a new query using both the **Location** and **House details** tables.
2. Place the **Town** and **Type** from the **Location** table and **Price** from the **House details** table on to the query grid. Ensure that they have their **Show** boxes checked.
3. In the **Criteria** box for the **Town** field, type **Not Cockermouth** to find all the properties except those in Cockermouth.
4. **Run** the query. The answer table contains 25 records.
5. Switch to the **Query Design**. In the **Town Criteria** box change the entry to <> **Workington**
6. Add the **Address** field to the query grid and sort the query by **Town** (ascending) and then **Price** (ascending). **Run** the query. There are now 24 results.
7. Save the query as **Not Workington**.
8. Preview the results before printing a copy.
9. Close the query and the database.



*Access does not select blank fields when searching for non-matches. For example a record with no data in the **Town** field, would not be included in a **Not Workington** query.*

Driving Lesson 60 - And Queries

P Park and Read

Previous Driving Lessons have demonstrated **AND** queries with selections in different fields. When it is necessary that two conditions must be met in a single field, e.g. a value must be higher than one value and lower than another, then the word **and** is used. Simply enter the two queries in the field, separated by the word **and**. As an alternative, the function **Between** can be used.



Manoeuvres

1. Open the database **Staff** and create a new query using the **Staff** table.
2. Place the fields **Surname** and **Date of Birth** on to the grid.
3. To list all the employees born in the 1960s, type the following information into the **Date of Birth Criteria** box and press **<Enter>**.

>=01/01/60 and <=31/12/69

4. Widen the **Date of Birth** field by clicking and dragging the right border to the right, until all the data is visible.

Field:	Surname	Date of Birth
Table:	Staff List	Staff List
Sort:		
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:		>=#01/01/1960# And <=#31/12/1969#
or:		

5. **Run** the query. The list contains the names of all the people born in the 1960s (about 8 records, depending on values used on additional records).
6. Switch back to the query design. Remove the **Date of Birth** field from the grid by highlighting the column and selecting **Delete Columns**.
7. Place the **Salary** field on to the query grid.
8. To find everyone who earns between £15,000 and £20,000 type **between 15000 and 20000** in the **Criteria** box of the **Salary** column.

Field:	Surname	Salary
Table:	Staff List	Staff List
Sort:		
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:		Between 15000 And 20000
or:		

9. **Run** the query. The list contains all of the people who earn between £15,000 and £20,000 (6 records).
10. Close the query without saving and close the database.

Driving Lesson 61 - Or Queries

P Park and Read

If a value within a particular field needs to match one of two or more conditions, then the conditions may be entered into the appropriate field on the query grid separated by **Or**. They may also be on displayed on separate lines in the query grid. **AND** and **OR** queries are examples of **logical operators**.



Manoeuvres

- Using the **Houses** database, create a new query grid based on the **Houses** table, including the fields **Town**, **Address** and **Bedrooms**.
- To search for all properties in Maryport **Or** Aspatria, enter **Maryport or Aspatria** in the **Criteria** box for **Town**.

Field:	Town	Address	Bedrooms
Table:	Houses	Houses	Houses
Sort:			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:	"Maryport" Or "Aspatria"		
or:			

- Run** the query. The answer table contains all properties in Maryport or Aspatria. There are 5. Switch back to the query design.
- Clear the **Town** criteria and in the **Criteria** box of the **Bedrooms** field enter **3**. In the line below, the **or** line, enter **4**. This will search for three or four bedroom houses.

Field:	Town	Address	Bedrooms
Table:	Houses	Houses	Houses
Sort:			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:			3
or:			4

- Run** the query. The answer table contains eighteen properties, all of which have either three or four bedrooms.
- Switch back to the query design and change the criteria as shown.

Field:	Town	Address	Bedrooms
Table:	Houses	Houses	Houses
Sort:			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:	"Maryport"		
or:			4

- Run the query. The results contain all properties which are either in Maryport or have 4 bedrooms.
- Save the query as **Complex** and close it, but leave the database open.

Driving Lesson 62 - Deleting a Query

P Park and Read

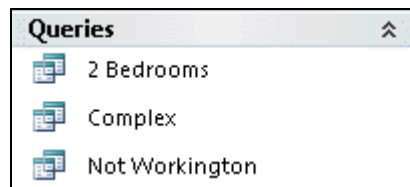
When several queries have been run and saved, the **Database Window** may become cluttered. In this instance, it is a good idea to delete those no longer required.

Care should be taken, however, that the queries being deleted are not used elsewhere, i.e. a form or report may be based on a particular query, deleting it will render the form/report unusable.

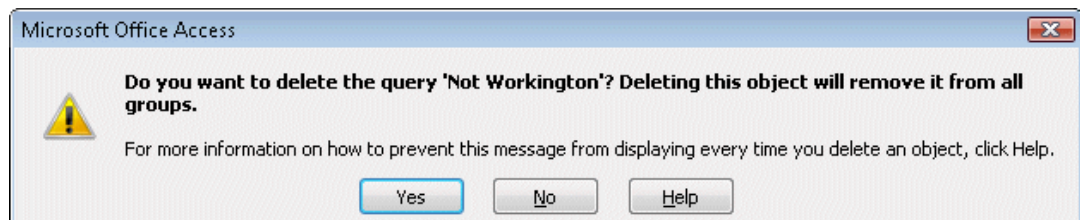


Manoeuvres

1. The **Houses** database is open from the previous driving lesson. If the **Navigation** pane is showing **All Access Objects**, the three queries created in this section will be listed.



2. From the list of **3** saved queries, click on **Not Workington**.
3. Press the **<Delete>** key or click the **Delete** button in the **Records** group of the **Home** tab. A dialog box is displayed.



4. Select **Yes** to confirm the deletion.
5. Leave the database open for the next Driving Lesson.

Driving Lesson 63 - Revision

This Driving Lesson covers the features introduced in this section. Try not to refer to the preceding Driving Lessons while completing it.

1. The database **Houses** is open from the previous Driving Lesson. Create a new query using the **Houses** table.
2. In the **Field List**, select all the fields and place them on the query grid.
3. Search for all properties in the table located in **Cockermouth**.
4. **Run** the query and view the answer table.
5. Switch back to the query design. Clear the grid. Now try the following questions. The fields to use are shown in brackets after each question.
6. How many properties include a garage in the sale? Place **Yes** in the **Garage** column. (Show **Town, Address** and **Garage**).
7. How many properties are for sale? Put **For Sale** in the **Status** field. (Show **Town, Address, Price** and **Status**).
8. How many three-bedroom properties are there? (Show **Town, Address, Bedrooms** and **Price**).
9. How many properties are there in **Whitehaven**? (Show **Town, Address** and **Price**).
10. How many properties cost less than £22,000? (Show **Town, Address, Price**).
11. How many properties cost between £40,000 and £50,000? (Show **Town, Address, Price**).
12. Use the criteria ***list*** in the **Comment** field. What comment does it find, and what is the Address? (Show **Town, Address, Comment**).
13. Use **<> None** in the **Offers** column, to discover how many houses have offers placed on them. (Show **Town, Address, Offers**).
14. Close the query without saving.
15. Close the database.



Answers to this exercise can be found at the end of this guide.

If you experienced any difficulty completing the Revision, refer back to the Driving Lessons in this section, then re-do the revision.

Driving Lesson 64 - Revision

This Driving Lesson covers the features introduced in this section. Try not to refer to the preceding Driving Lessons while completing it.

1. Open the **Football Agent** database and create a query that will show the **Surname, First name, Current Club** and **Annual salary** of any player earning more than £100,000 per year. Do not show the actual salary.
2. Move the **First name** field so that it will be displayed before the **Surname**.
3. Print a copy of the result but do not save the query.
4. Close the database.
5. Open the database **Daley** and create a new query.
6. All tables are to be included in the query.
7. Create a query to show the manufacturer and model of all cars repaired on 2nd January 2002 and who the mechanic was for each job.
8. Save the query as **2 Jan**.
9. Do not show the date in the answer, as it will now be the title of the query result. Print a copy.
10. Delete this query and close the database.
11. Open the **Staff** database and create a new query based on the **Staff** and **Courses** tables.
12. Find out how many staff have been on a course after the 1st April 1995. Show **Title, First name, Surname, Course** and **Date**.
13. Using the same fields, find out how many staff have been on an introductory course of some kind – use ***intro*** as the criteria.
14. Find out the only member of staff with a salary of more than £20,000 has been on an Intermediate course.
15. Close the query without saving and close the database.



Answers to this exercise can be found at the end of this guide.

If you experienced any difficulty completing the Revision, refer back to the Driving Lessons in this section, then re-do the revision.

Once you are confident with the features, complete the Record of Achievement Matrix referring to the section at the end of the guide. Only when competent move on to the next Section.