Introduction to Access

(V1.0)

copyright © University of Glasgow

Course content created by Blair Thompson

Last edited by Blair Thompson on 22/02/23

Contents

Introduction

This course is intended to impart basic database skills using *MS Access*.

The course is designed to teach you how to update records, search records, create queries forms and reports in *MS Access*. This will familiarise you with how databases work in *Access* and make it easier when you design your own database towards the end of the course.

The course is designed to give you a good overview of *MS Access* and some of the basic skills required for developing a database.

Objectives

On successful completion of this course participants will be able to:

- Search and Sort database records.
- Edit and update fields in database records.
- Create Forms to facilitate data entry and access.
- Create Reports based on Tables and Queries.
- Run Queries and edit and create new Queries.
- Decide how you want to store data on a database.
- Set up a database and a table within it.
- Identify the key which uniquely identifies an element in your table
- Understand Table relationships in Database design.

Course Work

Although the course work is usually voluntary, you are advised to complete it as it will help you to judge whether you have mastered what was taught on this course.

Sometimes, these units are offered as part of IT induction packages for specific university schools. In these instances, the school often requires that their students complete and submit the course work for assessment.

Introduction to Access
Session 1

Session objectives: by the end of this session you should have mastered:

- Familiarising with *MS Access*;
- Working with tables;
- Navigating through records;
- Adding new data to a table;
- Editing a record;
- Deleting records;
- Finding and replacing data;
- Filtering data;
- Creating forms and reports.

1   What is a Database?

A database is a collection of data, *regardless of format*. A filing cabinet, a library catalogue and a telephone directory are examples of databases. *Raw data itself does not have any meaning, but when it is structured and put into a context, it becomes information.*

Microsoft Access is a *relational database application* that comes as part of the *MS Office* suite which can be used to organize data and then be used to sort, search and query the data – in effect to turn it into information.

When *MS Access* opens it initially looks like this:
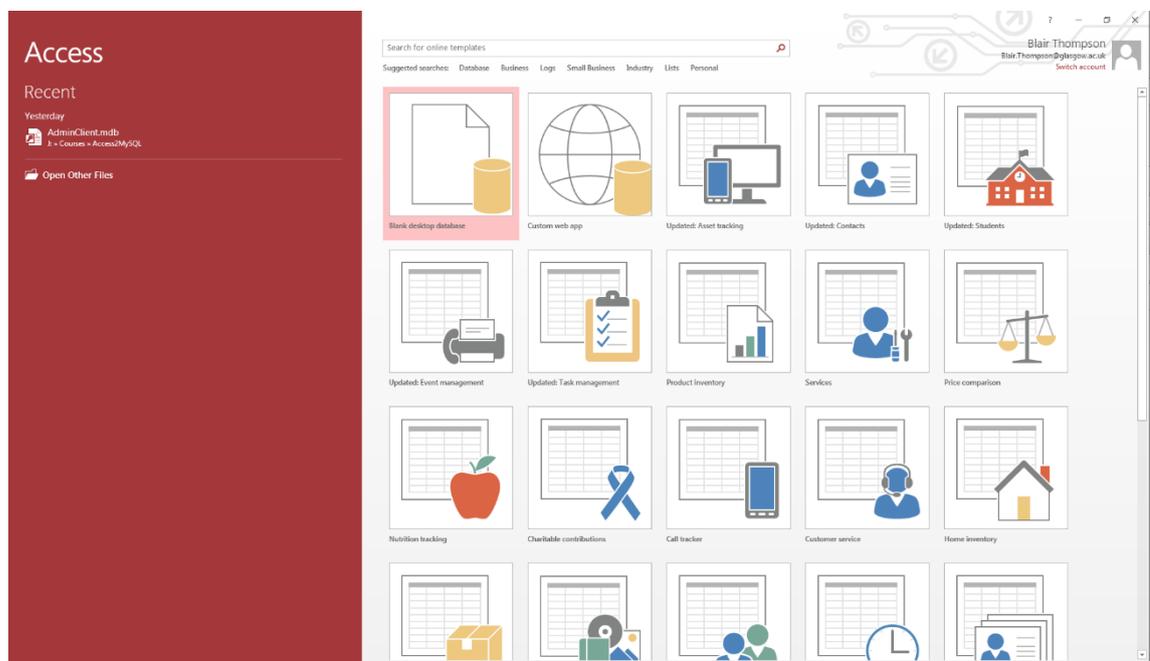
The default on opening is to display the **Start Screen** along with the **Blank desktop database** option selected. Clicking on the latter displays a box to enter a file name and create a new file. Once you have worked with some databases, the most recent will appear under the **Recent** title of the left hand pane

Microsoft Access has a number of templates that are available online, you can use the search box to find a template that matches your needs.

When you click on any template a dialog box will appear that will give you a more in-depth description of what that template does and controls that allow you to create a database based on it
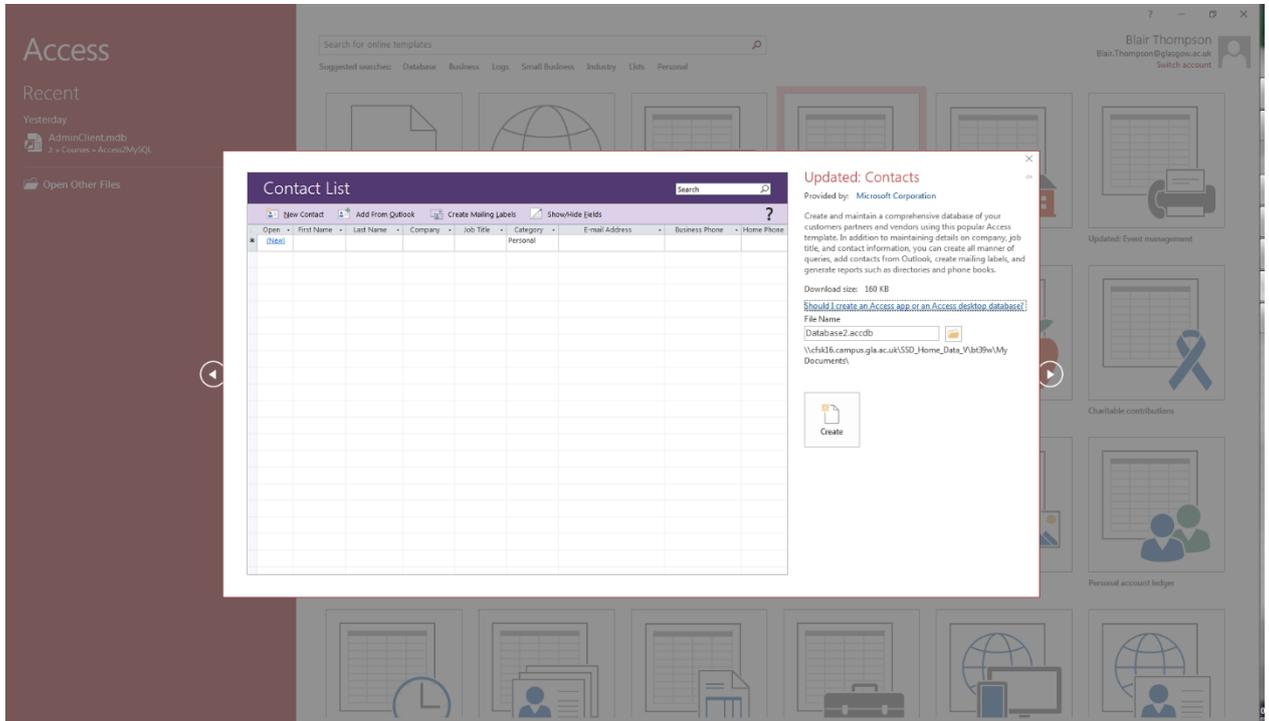
To open a database file stored elsewhere on your computer or on another network, click the **Open other files** link and browse to the file you want to open, and then click file to open it.

Note: because older versions of applications may still exist on campus, you may need to access older file types occasionally. *Access 2007/2010/2016* file types are interchangeable and all versions will let you use older files. However older versions of *Access* will not open *Access 2007/Access 2010/Access 2016* files.

Working with databases

When you open a database or begin to create a new one you will see a new View. We'll look now at the features here and what they can do.

a.   Task 1

1   Open *MS Access* from the **Start** menu. Next, open a web browser and go to IT Training's **Course resources** web page.

2   Download the Microsoft Access zip file

3   Save the zip file to a location of your choice

2   The Interface

a.   Command Tabs:

Along the top of the window (on the **Ribbon**) are the command tabs:
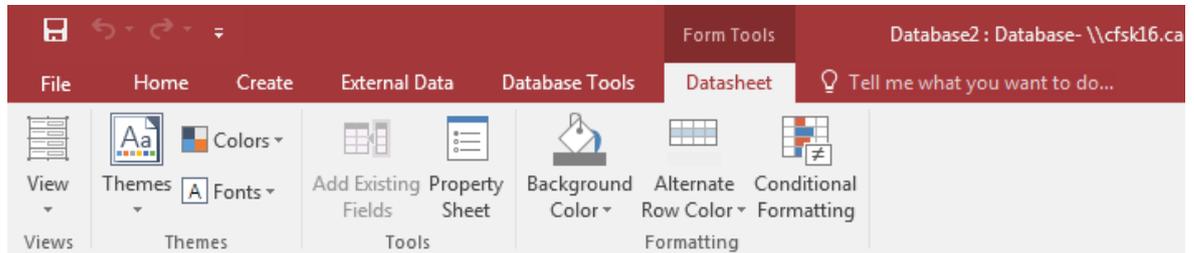

The command tabs are essentially the same thing as menus. The commands listed under each tab are also the commands that are applicable to your current view of the database. This goes one step further with the addition of *contextual tabs*: e.g. the tab labeled **Form Tools** appears only when you have selected a Table in **Datasheet** view.

Below the command tabs' ribbon is the main window, which is split into two panes. To the right is the **Work** area, where you can design, edit or update a database. To the left is the **Navigation** pane: here you can navigate quickly to any of the **Tables**, **Queries** or **Forms** you wish to work with.

**Navigation pane**   **Work area**
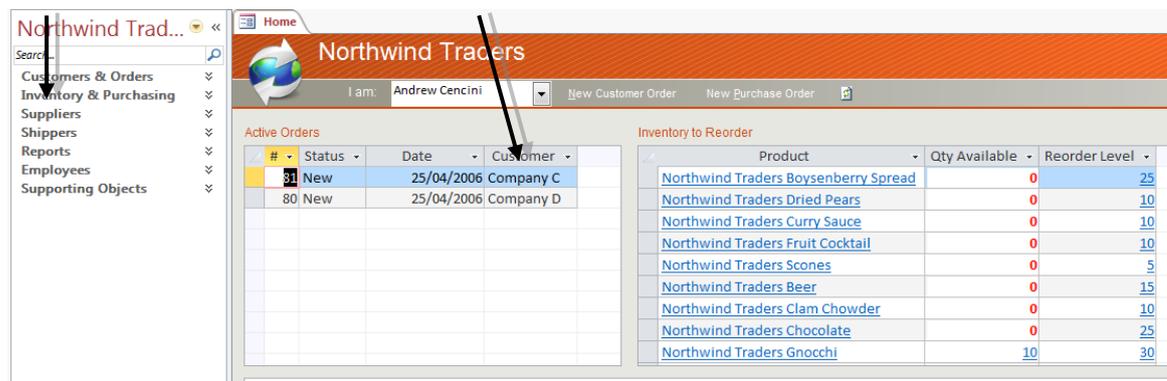
Note: for some objects, a separate window opens when you choose to edit/update them.

b.   Views

Click the **Views** command on the **Home** tab to cycle through the different views available for each object. (The type of views available will differ depending on the object that is currently open).
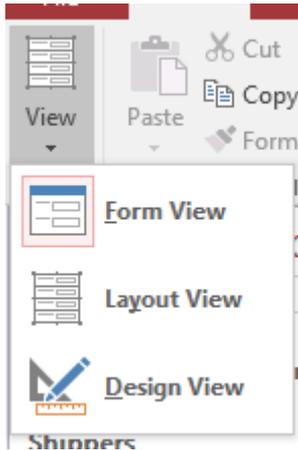
You can also click the small down arrow underneath the word **View** to see all of the available views.

The current **View** can also be changed by clicking on the **View** buttons in the lower-right corner of the window.
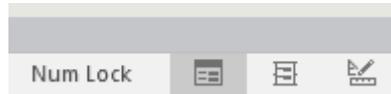
c.   Northwind: A Sample Database

We will now look at the sample database you downloaded called '*Northwind Traders'*. This shows how a database can organise a collection of data in a structured way.

All the necessary data on the company's products, suppliers and employees is held in the database.

This database therefore contains examples of all the objects a database uses to structure this data and extract information, and will be used to help you familiarise yourself with *MS Access*.

d.   Objects in a Database
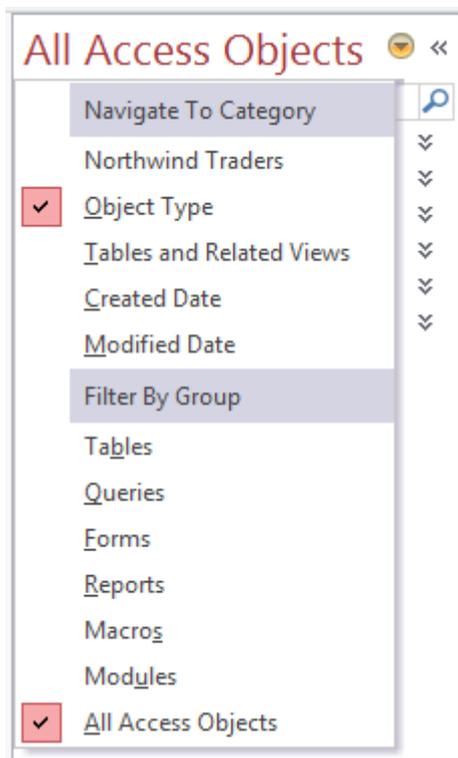
A database object is defined as some individual piece of a database that can be used on its own. The major database objects are: **Tables**, **Queries**, **Forms**, **Reports**, and **Macros**.

These can be accessed via the **Navigation Pane** – which is in a "Shutter Bar". Open or close this by clicking the double arrow button at the top left of the work area window.

The **Navigation Pane** is used to control and use the objects of an *MS Access* database.

- Expand the **Navigation Pane** by clicking the pull-down arrow beside the title to show the full **Navigation Pane** toolbar.
- Then, click **Object Type** to display all objects currently in the database.

All objects in the database are categorized by their object type.

For instance, if you clicked on **Tables and Related Views**, you'll see a list of all the Tables that are in the database which is currently open.

When you wish to work with another type of object, simply expand the **Navigation Pane** again to select the new category.

Now we'll look at one of these object categories – the **Table**, which is key to creating a database. (Although we won't be creating a database until much later: first you'll become familiar with all the features of a database by working with an existing one).

Tables

A Table is where the raw data is held, laid out in *columns* and *rows* and consisting of individual cells.

Record ID	Field Name 1	Field Name 2	Field Name 3
Record 1	Data	Data	Data
Record 2	Data	Data	Data
Record 3	Data	Data	Data

The *columns* of cells contain the individual units of data arranged according to category or *Field*. Thus every cell in a column will contain the *same type* of data. This could be text, or currency figures or date or even a file of some kind.

The *rows* contain all the information about a *single database item*, known as a *Record*.

So every record will have the same structure (same types of *Field* in the same order) though individual data items may differ.

Complex databases can contain many tables.

The data in tables is organised into distinct groupings, as in the "*Northwind Traders*" example – *employees*, *products*, etc., to make retrieving useful information easier.

Note: every table in a database must contain a **Primary Key Field**. That is, a field whose data will *always be unique*.

Without this you will not be able to distinguish, e.g. between employee records with the same name.

Usually a *numeric ID* is created and there is an option in *MS Access*, called

AutoNumber which will automatically create a unique number for each record as you enter data.

Primary keys allow the different tables to be joined together in order to extract information that you require.

3   Working with Tables

We'll demonstrate Editing (or Updating) **Tables** using the *Northwind traders* database which you should have open.

Task 2

Open the **Employees Table** as follows:

1   Expand the "Shutter Bar" if necessary, then open the full list of *Access* objects.

2   Select **Tables** and then open the **Employee List** Table.

b.   Navigating Through Records

To navigate around the records, use the navigation buttons at the bottom of the **Table** pane:
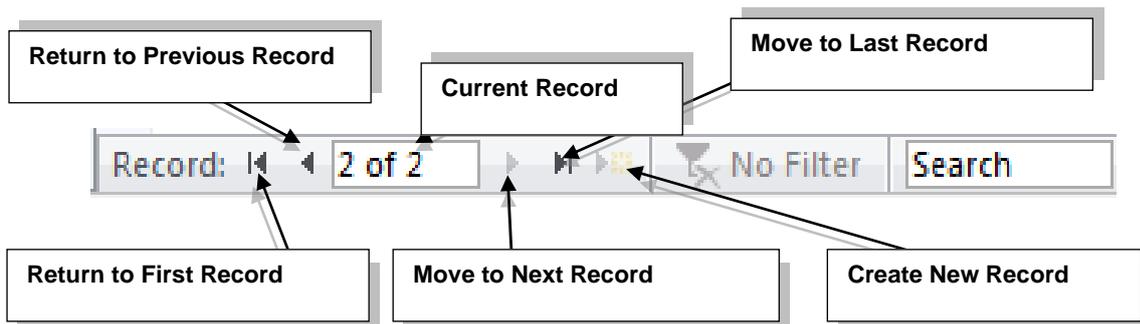
Or, simply click with the mouse into a cell of the table. Depending on the size of the Table, it may have horizontal or vertical scroll bars.

c.   Adding New Data to a Table

Click on the **Create New Record** button shown above. Ignore **AutoNumber** fields, as their new, automatically generated ID numbers will be added by *MS Access*.

Place the cursor in the next field of the new record and enter the data. Check that it is correct then move to the next field: enter data and continue entering data and moving along until you have completed all the fields in the record.

A small pencil icon will appear to the left of the record you are currently writing.

If you are entering data using the keyboard, pressing the **Tab** key will also advance you to the next field in the row. However, if you have reached the end of the record and press **Tab** again you will move to a new record.

Note: Moving to another **Record** in the **Table** or closing the **Table** will ensure that the changes to the record have been saved. *Access* automatically saves such edits to data.

Task 3

Add the following information (just leave the other **Fields** blank):

1   **Company:** Northwind Traders

2   **First Name:** John

3   **Last Name:** Jones

4   Did you note the order of these names?

5   **Job Title:** Sales Representative

6   **Business Phone:** (123)555-0100

7   **Home Phone:** (123)555-0103

8   **Address:** 123 10th Avenue

9   **City:**   Redmond

d.   **Selecting a Record**

Moving the mouse to the shaded box at the left of the record changes it to a black arrow.

	ID	Company	Last Name	First Name	E-mail Address	Job Title
⊞	1	Northwind Tra	Freehafer	Nancy	nancy@northwindtraders	Sales Representative
⊞	2	Northwind Tra	Cencini	Andrew	andrew@northwindtrade	Vice President, Sales
⊞	3	Northwind Tra	Kotas	Jan	jan@northwindtraders.cc	Sales Representative

Clicking in this position selects all the data in the record (simultaneously highlighting it) and allows it to be copied, pasted or deleted.

4	West	40
5	Northeast	55
6	Northwest	56
7	Southeast	57
8	Southwest	58
*	(New)	

By dragging up or down you can select any number of records together (and then copy, paste or delete them.

e. Deleting a Record
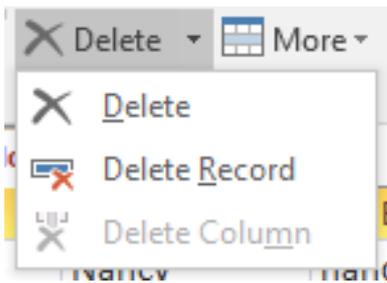
After selecting the **Record** as above, click the small pull down arrow beside the **Delete** command in the **Home** tab (in the **Records** group) and click **Delete Record**.

Attempting to delete a record, contained in a **Table** that is related to another **Table**, will result in an error message. This is so that other data will not be put at risk. (Table relationships will be explained later).

f. Editing a Record

Editing is simply a case of clicking inside the **Field** in a record to be edited and changing the data in a similar way as would be done in *Word*. Moving to another record in the **Table** or closing the **Table** will ensure that the changes to the record have been saved.

	ID	Company	Last Name	First Name	E-mail Ad
	1	Northwind Tra	Freehafer	Nancy	nancy@north
	2	Northwind Tra	Cencini	Andr	andrew@nor
	3	Northwind Tra	Kotas	Jan	jan@northwi
	4	Northwind Tra	Sergienko	Mariya	mariya@nort
	5	Northwind Tra	Thorpe	Steven	steven@nort
	6	Northwind Tra	Neipper	Michael	michael@nor
	7	Northwind Tra	Zare	Robert	robert@nortl
	8	Northwind Tra	Giussani	Laura	laura@northv
	9	Northwind Tra	Hellung-Larser	Anne	anne@northv
*	(New)				

The record being edited becomes highlighted and marked with the pencil icon.

Task 4

Open the **Customers** Table.

- Change "Carlos Grilo's" **Job Title** to "Purchasing manager".
- Change "Karen Toh's" **Business Phone** number to (123)555-0100.
- Delete record **19** (Company S).

4 Finding and Replacing Data

You can use the **Find** and **Replace** commands to search Tables. You can find both commands on the **Home** tab. The **Find** command will search through an object and locate all instances of a keyword.

The **Find** command also gives you the ability to search only specific columns of data and flexibility in how it searches. If you only know part of a word or phrase, you can search based on what you know.
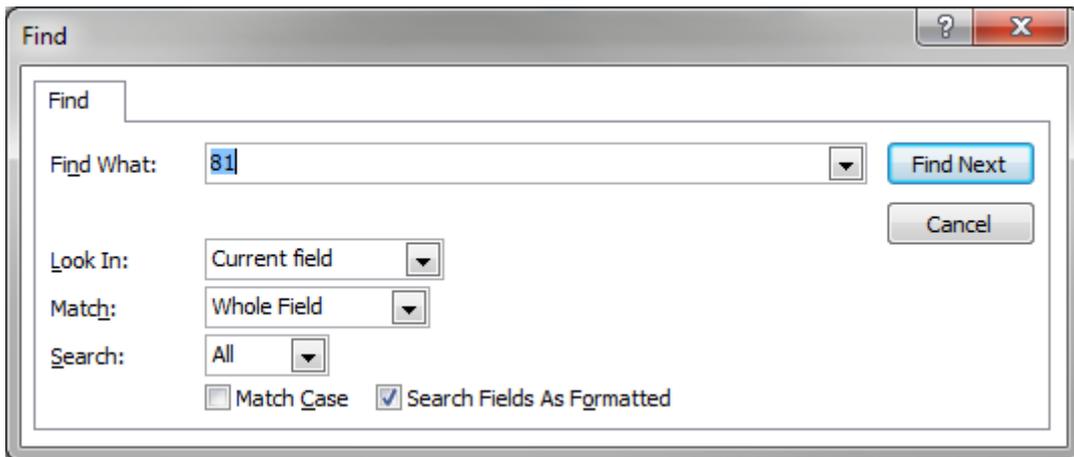
After typing in the text to be found, there are two options that can be used to refine the search.

If you have no selection made in the table **Find** will search the *whole table* but if you place the mouse cursor in a particular column you can change the **Look In** field to search only in the selected column.

Don't forget to change it back if you wish to search the rest of the Table.
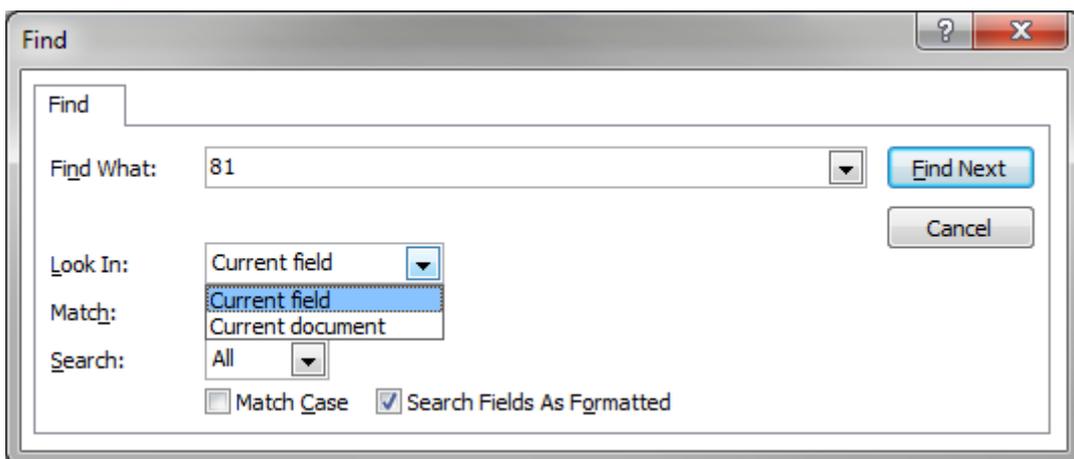
The **Match** drop-down menu allows for selecting to search the whole field for each record, the start of the field or any part of the field.
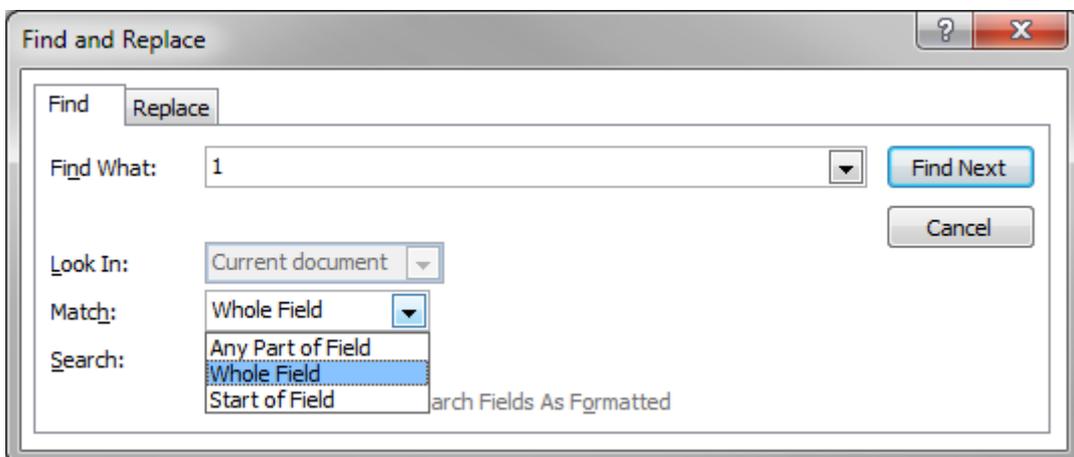
After choosing the most useful combination of options, clicking on the **Find Next** button will locate the desired record.


Task 5

Open the **Customers** table. Use **Find** to search for particular values:

1   Find the records for *Accounting Manager*

2   Find the records for customers in *Chicago*

3   What is the name of the contact for *Company L*?

b.   **Find and Replace**

If the text found is to be replaced, click on the **Replace** tab at the top of the dialog box. This produces another box into which the replacement text can be entered, and it is then a question of either replacing one instance of the found text or all instances:
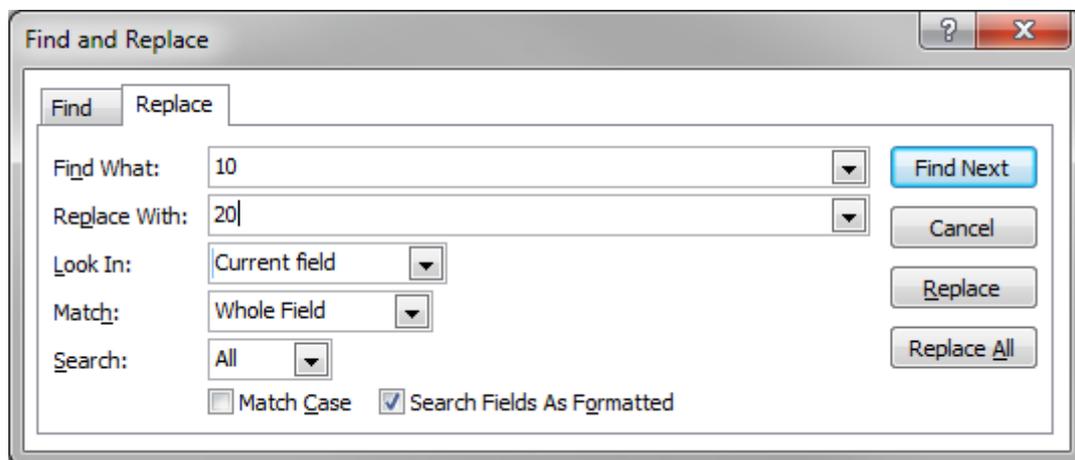

Task 6

Open the **Products** table.

1   Find "Northwind Traders Peas" and **Replace** with "Northwind Traders Garden Peas"

2   Find the **Product Code** "NWTD-72" and **Replace** with "NWTD-12"

3   Find the **List Prices** which match £10.00 and **Replace All** instances with £12.50.

Note: You can use **Find** repeatedly, to find different records *with the same data* in one field but when you want to replace data items in a field with one particular item of data you should just use **Replace All**.

5 Filtering

If you have multiple records which match the **Find** criteria, these records can be displayed together by means of filtering using these buttons in the **Sort & Filter** group on the **Home** tab.
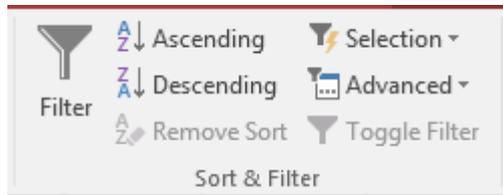

a. Filter and Sort

A search for a term such as **Sales Representative** can find several records. To view them all together,

1   First click on any cell *in the column containing the appropriate Fields*, then choose the **Filter** command
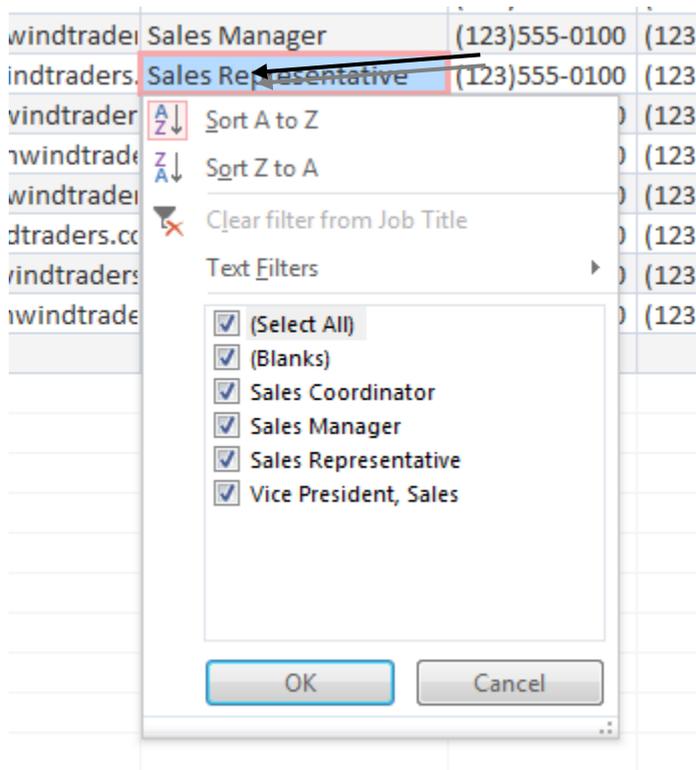
2   The **Filter** pop-up box will appear (this also gives the option to **Sort** on this field).

3   This will display the different values (but not duplicate values) the field contains.

4   Deselect all the values that you don't require and then click **OK**.

5   The table will then display only those records containing the field values you selected.

6   To select a few items from a large list, click on **Select All** to deselect everything. Then select the items you wish to view.

7   To remove the filter, use the **Toggle Filter** button (the Filter will still be in memory and you can activate it again by clicking the **Toggle Filter** command).

8   To *permanently* remove a Filter, activate the **Filter** command and choose the **Clear filter from** option in the **Filter** pop-up box.
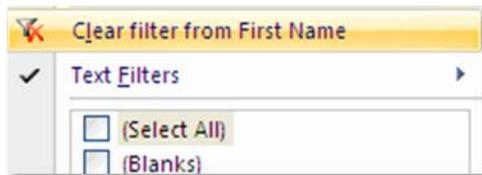
Task 7

Open the **Products** table.

1   Sort the table records on List Price from lowest to highest price. Then sort from highest to lowest.

2   Undo the sort and sort on Category. Undo this sort.

3   Open the Customers table.

4   Filter the records to show all instances of the Job Title of "Owner".

5   Remove the Filter.

6   Display all the records for customers from "Memphis".

7   Remove the Filter.

b. Filter a Range of Values

MS Access makes it easy to filter a table of data quickly based on one criterion. For example, consider the **Standard Cost** column in the **Products** table of the *Northwind* sample database.

1   Click the first price in the list to highlight that particular field and then click the **Filter** command in the **Sort & Filter** section of the **Home** tab. The **Filter** pop-up box will appear.
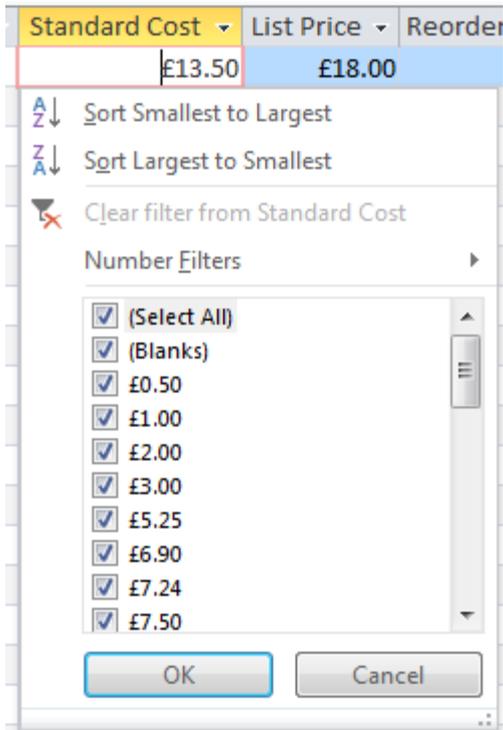
2   Click on the heading, **Number Filters**, and new menu appears, i.e.

3   **Equals**, **Does Not Equal**, **Less Than**, **Greater Than**, **Between**.

4   (If you have *text data* it will display **Text Filters**, if you have *dates* it will display **Date Filters**, etc. They will all give access to a similar menu).

5   When you choose an option, a small dialog box will open.

6   The examples seen just below here are for **Equals** and **Does Not Equal**.
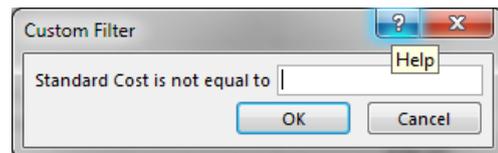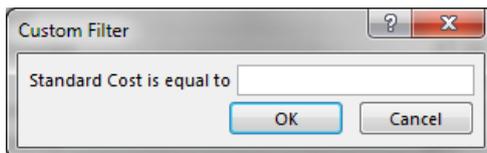
Note that for **Less Than** and **Greater Than** the filter is actually **Less Than *or equal to***, **Greater Than *or equal to***.
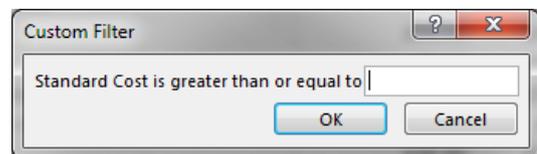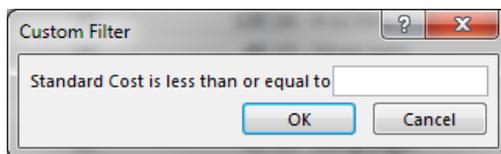
The final option is **Between**. The dialog box for this requires you to enter the smallest and largest figures. The filter will display all the records with **Standard Cost** fields containing values within this range and *including the two values that you enter*.
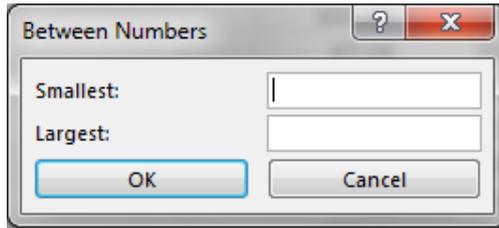
7   To undo the Filter click the **Toggle Filter** button; it is a **Toggle** button which switches the Filter on and off. To *permanently* remove a Filter, activate the **Filter** command and choose the **Clear filter from** option in the **Filter** pop-up box.

Task 8

Go to the **Products** table. Use a **Number Filter** to:

1   Display all the **Standard Costs** greater than £10. Remove the filter and then use a new one to display all the **Standard Costs** between £5.50 and £12.75.

2   Open the **Orders** table.

3   List the **Shipping Fees** that cost £100 or more. What are the **Order IDs** for the highest **Shipping Fees**? (**Sort** can help with this).

c.   Text Filters
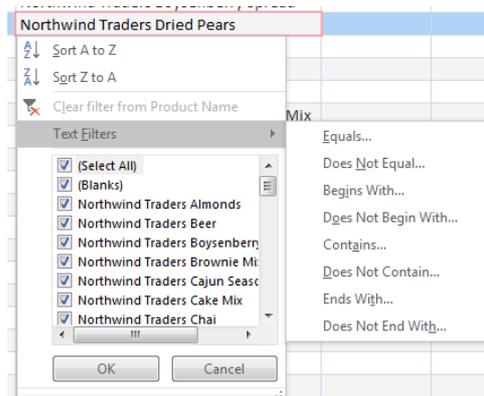
The **Text Filter** options are somewhat different to the numeric filters. While it has **Equals** and **Does Not Equal**, it does NOT have **Less Than**, **Greater Than**, or **Between**.

Instead it has **Begins With**, **Does Not Begin With**, **Contains**, **Does Not Contain**, **Ends With**, **Does Not End With**.

If you chose **Equals** or **Does Not Equal** you must enter an exact match for all the text in the field.

The other options however allow you to enter only part of the text in a field. You can enter a number of characters – for instance a surname or product name. Or if you can't remember the full name, you can enter part of a name.

Task 8

Go to the **Products** table. Use a **Text Filter** to:

1   Display all the products from the **Category** *Beverages*. How many **Beverages** have a reorder level greater than **20**?

2   Remove the filter and then display all the records with a **Category** containing the word *Fruit*.

The **Date Filter** displays a **Calendar** icon which opens a calendar from which you can choose a specific date.

Your options here are **Equals**, **Does Not Equal**, **Before**, **After** and **Between**.

Make sure you choose the correct year, as well as month and day. You can of course, if you prefer, still enter a date by typing it into the text box.
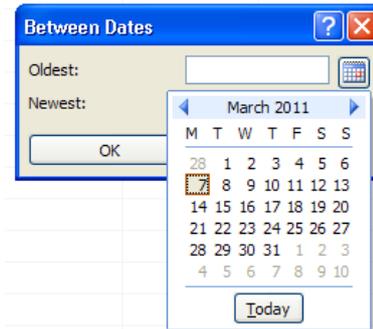
Task 8

Go to the **Orders** table. Use a **Date Filter** to:

1   Display all the orders that were paid between 30th April and the 31th May 2006 (inclusive).

2   Display all the orders made before the end of March 2006. How many of these have not been paid yet?

6   Forms

So far you've used **Tables** to enter and edit data (though only briefly) and to Sort and Filter data to find information. While Tables are fine for the latter, they can be problematical for data entry.

A **Form** is a way of inputting data into a **Table** in a database in a user-friendly way, and in a way that will not affect how a **Table** is organised.

Forms are also useful for reducing data entry errors as the clarity of the form makes it easier for the user to enter the correct data into the correct field.

Note that in this form the employee data is split into two sections (each accessible by clicking on the appropriate tab) that is, company data and personal data. This allows the form to be in a smaller and more easily readable format.

The view of the second tab (**Orders**) is shown below.
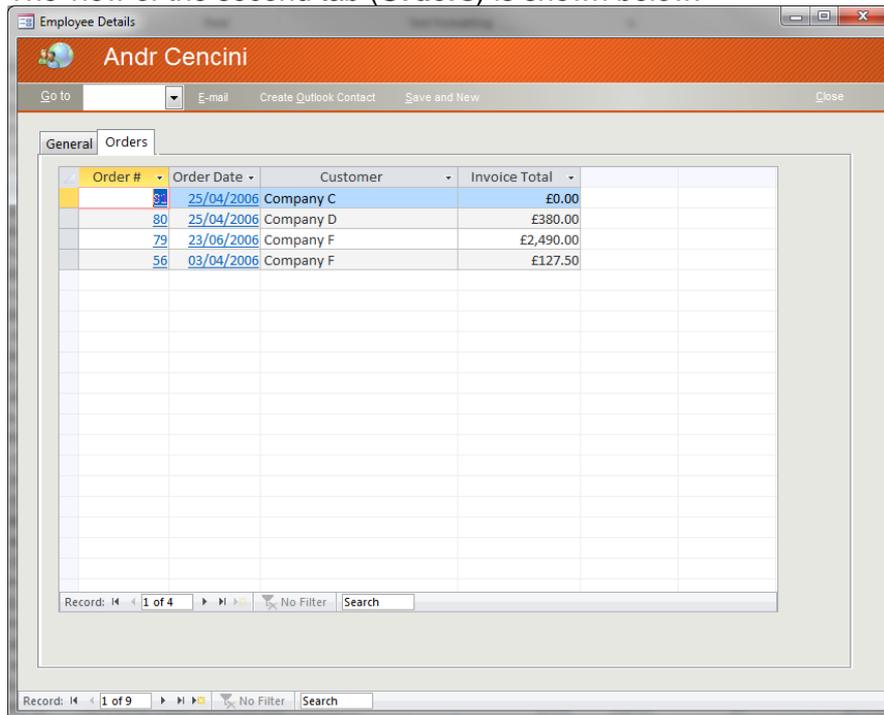
Notice also that the database can hold image file data and that the Order numbers and dates are links which will open more order information in another form.

At the bottom of the **Form** are the buttons used to **Navigate** through the individual **Records** which are part of the **Table** on which it is based.
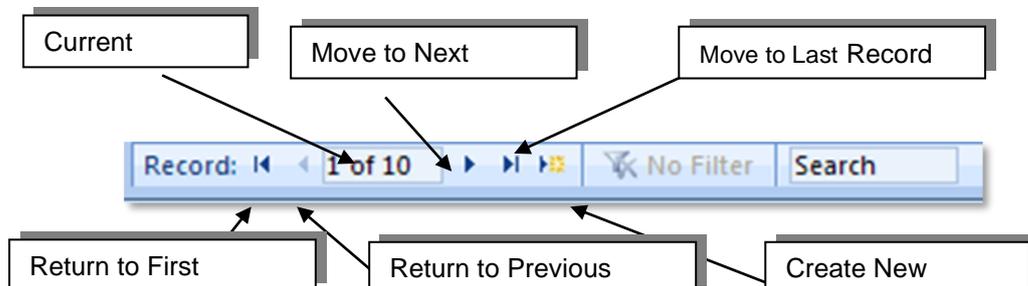
To add a **New Record** to the **Table** using the **Form**, click on the arrowhead with the asterisk* and complete the now empty **Fields**.

Task 9

1   Open the **Employee Details Form**. Add the following information:

First Name: Jane

Last Name: Jones

Title: Sales Representative

Business Phone: (123) 555-0105

Home Phone: (123) 555-0103

Address: 123 7th Avenue, Seattle, WA, 99999, USA

2   Open the **Employees Table** to see if the data has been added.

b.   Creating Forms

Forms are a user-friendly way to enter data into a Table. The simplest way to create a **Form** is to use the **Form Wizard** under the **Create** tab in the **Forms** group:
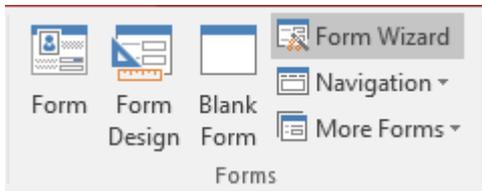

Task 10

Complete this task by following the guidance below:

1   Select the **Form Wizard** option in the **Forms** group under the **Create** tab.

2   This will open the **Form Wizard** dialog box.

Note the two selection options. The first is used to select the **Table**, in this example the **Shippers Table**, on which the **Form** is to be based, and the second to select which **Fields** to include in the **Form**.
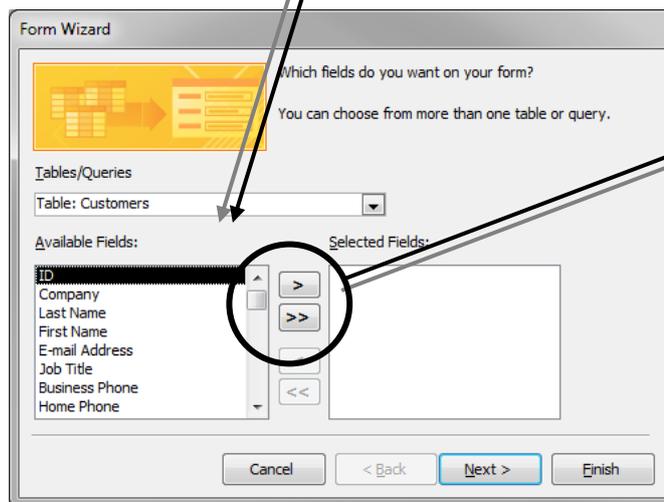
Using the topmost of the two magnified buttons includes the highlighted **Field** in the **Form**.

Using the bottom button will include all the **Fields** in the **Form**.

3   Select the **Table** on which you wish to base your Form.

4   From the fields now displayed, select those you wish to appear in your Form.

5   The next option is used to choose the layout of the **Form**, with each of the options previewed on the left-hand side.
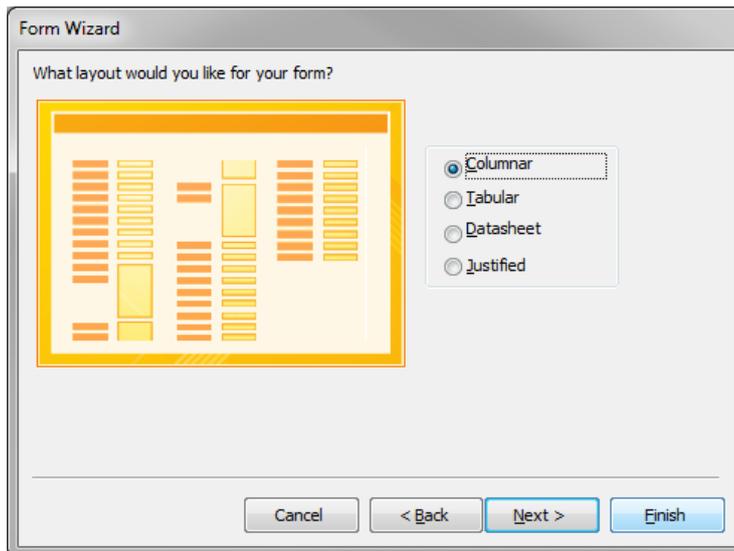
6   Select the layout you prefer for your Form.

7   Click **Next** to confirm your selection and move on.
The final part of the **Form** creation is choosing a name for the **Form**:
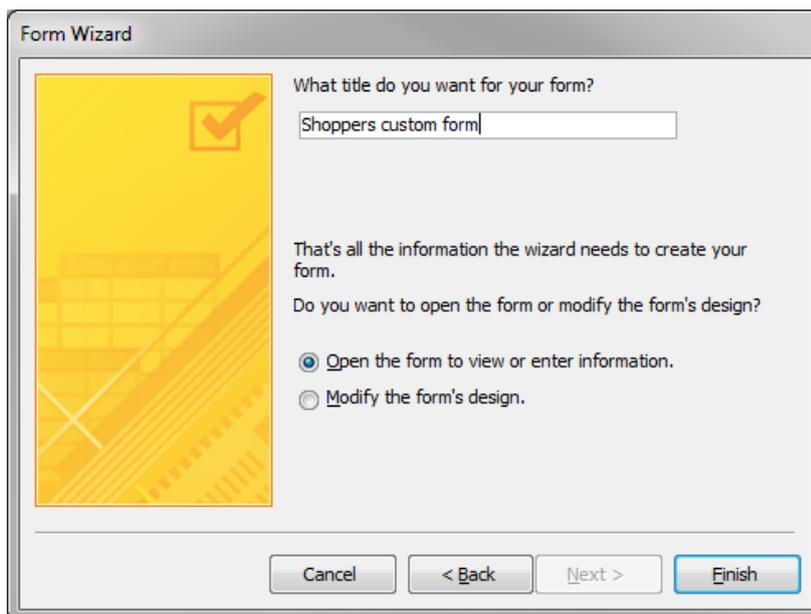
8   Type in a suitable name for the Form.

9   Click **Finish**. The wizard by default will open the Form for you to view or edit.

10   The finished **Form** will look similar to this:

Note that the form you created displays the chosen fields in text boxes in which you can add or edit data. The typical database navigation buttons will be seen at the bottom of the Form (though are not shown here).

7 Reports

A **Report** is a way of presenting the information extracted from a database in a well-organised fashion. Reports can be used to produce printed documents or online documents which can be downloaded and read.

Reports, like forms, are completely customizable and easy to create by using a Wizard.

If the data in your database has changed, you don't need to design a whole new report. Simply reissue the report and when *MS Access* runs the background query again, the data changes will be taken into account automatically.

We'll look at some examples of Reports so you can see how useful they can be.

Task 11

1   Access the **Reports** section in the **Navigation** pane. Open up the **Report** "*Employee Address Book*".

2   Click on any **Employee** name: an **Employee Details** form will open up for that employee, allowing you to edit the employee data.

3   Open and view the **Quarterly Sales Report**. Choose **File** tab ➔ **Print** ➔ **Print Preview**. This demonstrates how a report creates a print-ready document.

4   Choose **Save As** and select to save as a **PDF**. The Report will be saved as a document that can be viewed on a web site or used for printing.

b. Creating Reports

Many of the reports you create will simply be an exercise in displaying the data in a certain way. The **Report Wizard** command can be found in **Reports** group under the **Create** tab.
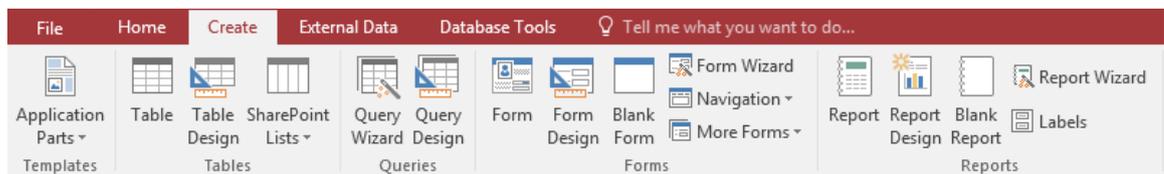
Since reports are made from **Queries**, and most of the queries will have already been built, creating reports using the **Wizard** is easy. (We look at **Queries** in detail next session but there are examples in the Northwind database we can use meanwhile).

Firstly, the **Table** or **Tables** the **Report** will be based on must be chosen, and then the required **Fields**. It is important in the selection process to choose the Fields in the order that they should be displayed.

Task 12

1   For this example, we will make a Report based on the full results from the **Customers Extended** query.

2   Choose **Company**, **Last Name**, **First Name**, **E-Mail Address**, **Job Title**, **Business Phone**, **Address**, **City** and **State**.
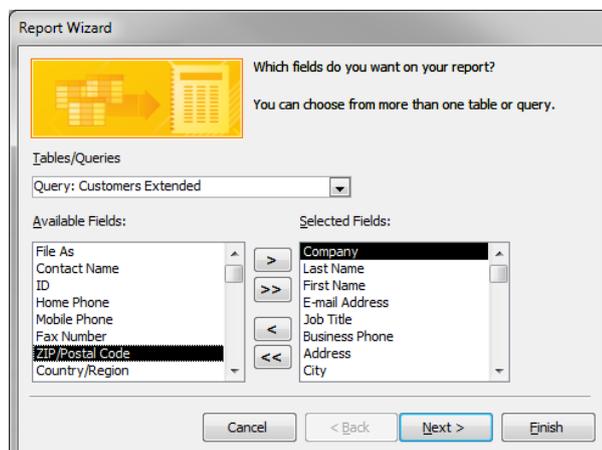
3   The next screen of the Report Wizard allows you to apply levels of grouping to the report.
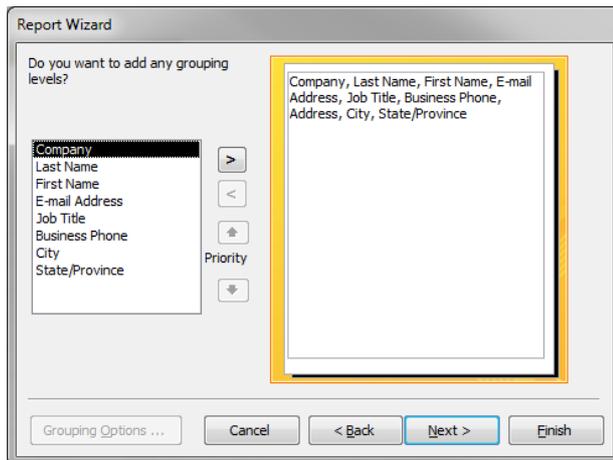
The next window allows your data to be grouped according to any **Field/s** chosen. You should always think carefully about what is the logical field for your data to be grouped under.

Often it will be the key field but you may want to highlight other elements: e.g. group all the employees by birth date (so that they are arranged by age).

4   Select **City** to be your **Grouping** level.

This means customers will be listed by their city.

5   Click **Next** to continue.
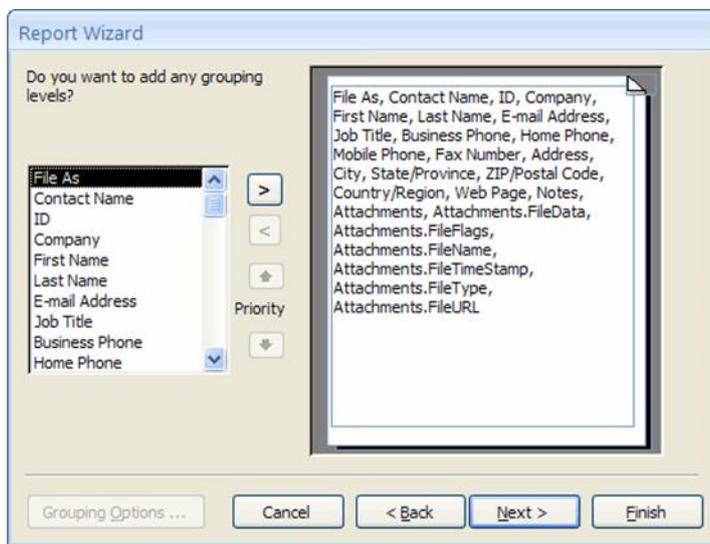
Your report will now list your data under this ID – in numerical order.

In your own time, later on, you should try grouping under a different field and view the result.

The next options deal with the layout of the information, first with a sort order for the records. By selecting a grouping level you already imposed one type of order on the data.
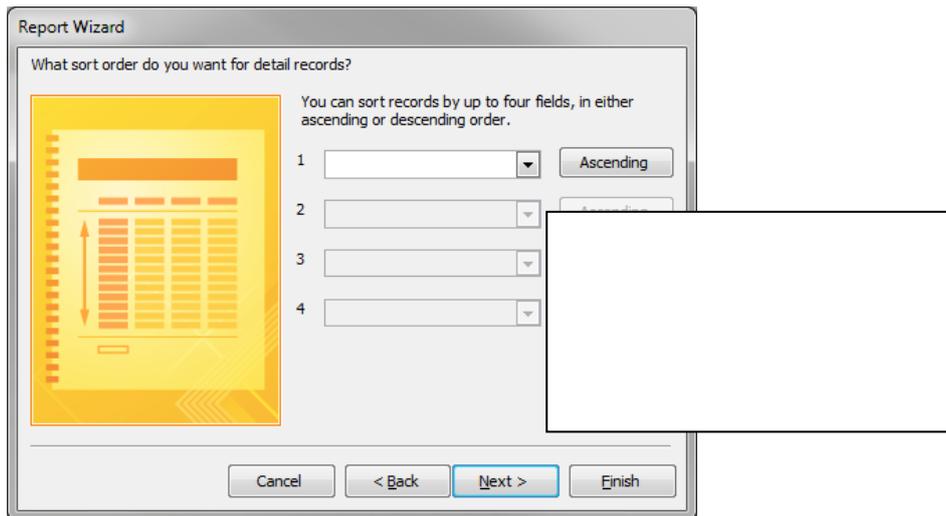
6   The next window is for **Sort** order. Select **Last Name**.

How do you think this will affect the order?

*7   Click **Next***
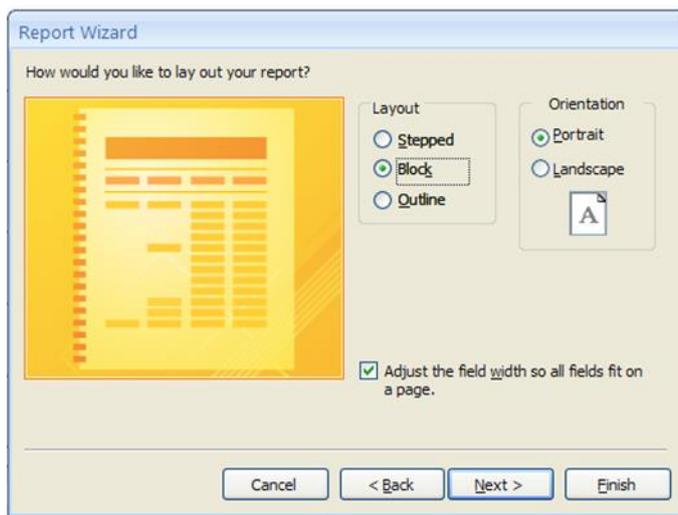
8   The next window is for report layout. Click on each option to view a preview.

9   Select the Layout, **Stepped** and the Orientation **Landscape**.

10   Click **Next**

11   The final window will appear: enter a title for the report.

12   Choosing the **Finish** button will then allow for the report to be previewed and saved.

      The preview window will show how the report will look when printed out. This allows you to see any corrections that may be need to be made to the design.

Task 13

1   Examine the report preview. Has sorting on **Last Name** made any difference to the order of items?

2   Look at the position of all the fields in the report. Do you see anything wrong?

      Sorting on **Last Name** has made one difference – **First Name** has been separated from **Last Name** by **Company**.

      This is because **Company** was added to your selection of fields before any name but sorting on it has made it move ahead of **Company** while **First Name** has remained where it was.

      We need to adjust the position of the **Company** number on the page. You will also note that some **Job Titles** and **Phone Numbers** don't display fully.

3   Close **Print Preview**.
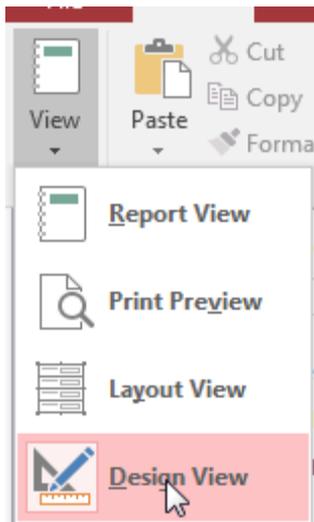
Note that in the left hand corner of the ribbon on the **Home** tab there is the **View** command button.

When this button is clicked a selection of **View** options will appear. What we want here is **Design View** and selecting this will open a new view in which all the elements of the report (but not the data) are displayed on a grid.

It is quite easy to adjust a report layout if it is not quite as you want it: there is no need to recreate the report all over again.
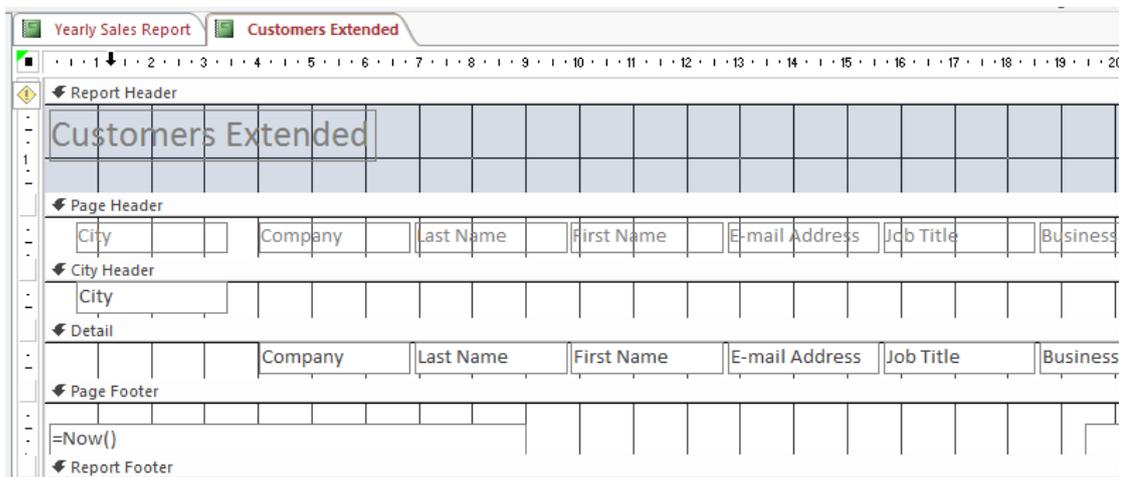
Task 13 (cont.)

1   With the report still open, select **Design View** from the **View** command on the left corner of the ribbon.

2   The report design layout will appear (see above). Select **Company** and move it to the right of the **First Name** box.

3   Select the **Business Phone** and **Job Title** boxes. Drag the edge of the boxes outward until they are long enough to display the longest item.

4   You may have to repeat this, checking the result in **Report View**.

5   When you have completed these tasks and returned to **Report View** you should see that the report now displays all the fields properly.

8   Using Print Preview

Print Preview is used to view a document in full form before actually printing it. To open this view, click the **File** tab, select the **Print** command, and then click **Print Preview**.
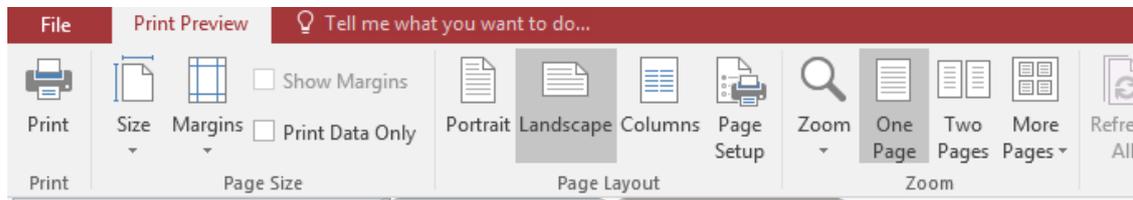
In the **Page Size** group you can choose from a number of paper output sizes and choose a normal, wide, or thin margin; in the **Page Layout** group you can choose a page orientation or set columns. The **Page Setup** button opens the full **Page Setup** dialog box containing all of the above functionality and more.
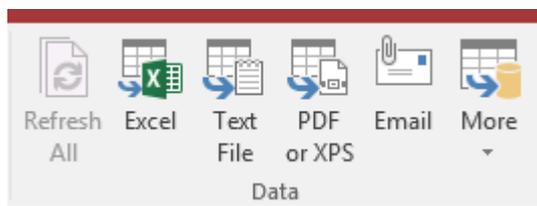
The **Data** section of the tab allows you to save a digital copy of a database object instead of printing a paper hard copy. The report can then be organised or analysed further by exporting a copy of it, e.g. to *Word* or to *Excel* by selecting an **Export** option via the **Data** group on the tab.

Task 14

1   Display the report you created in **Print Preview** view. Examine how it looks in different page layouts and different sizes.

2   Return everything to its original page setup, including **Landscape** view. Export the report as a PDF.

3   Now export the report as a *Word* file. Open it and examine it. Save and close.

9   Summary

Your tutor will remind you of what you have covered in this session, and answer any final questions.

Save your work and make back-up copies of any files you have used in this session.

Session 2

1 Session objectives:

by the end of this session you should have mastered:

- creating queries;
- creating a blank database;
- creating a new table;
- data types;
- primary keys;
- using the lookup wizard;
- saving a table;
- relationships between tables.

2 Queries

A query is a question that is asked of a database control program about the data it contains. We specify what particular fields we are interested in finding out, tell the database where to look for those fields, and specify any conditions under which to search.

Unlike searches using Filter options, you can choose that only those fields you are interested in will be displayed.

Task 15

1   Open up the query **Customers Extended** using the navigation pane

2   Choose **Design View**.

Notice this consists of two main parts: the Table on which the query is based (top part) and the query itself below. Don't attempt to change anything.
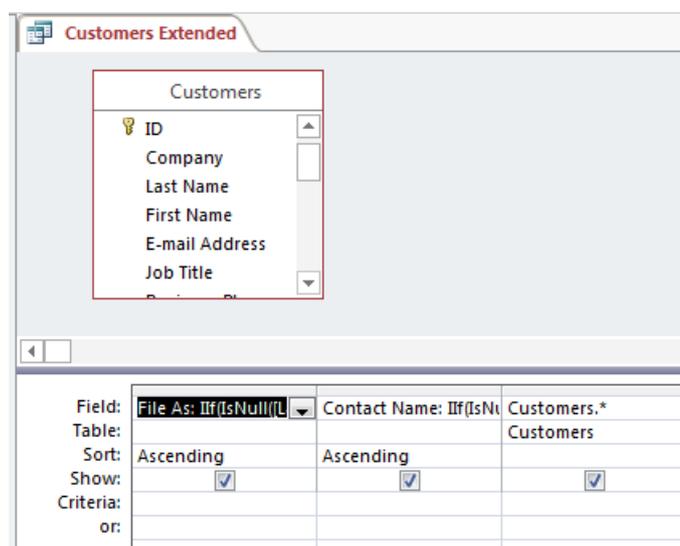
3   Now open the query **Order Details Extended** in **Design View**. This shows the
     same kind of display. However now we have two Tables in the query.
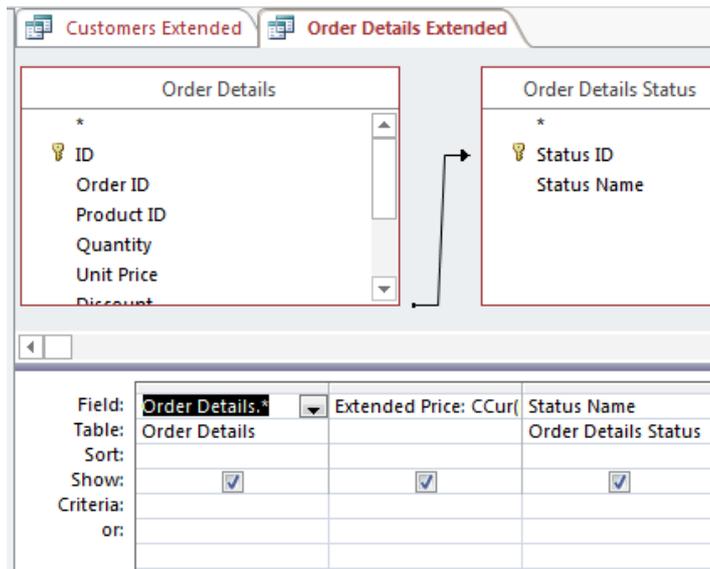
Notice the tables in the second query, above, are linked by an arrowed line: this connects
the tables through the same ID field which exists in both Tables. This is what's known as
a table Relationship: we will discuss this further, later in the session.

3   Creating Queries

We'll look at how queries work by creating a **Simple
Query** – that's a query that selects data from a single
Table.

To start working with a new blank query, click the **Query
Design** command which is in the **Queries** group (in
Access 2007, the **Other** group) under the **Create** tab.
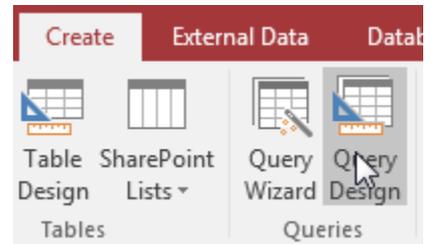
Task 16

1   Choose **Queries Design** on the **Create** tab and the **Show Table** window opens,
     where you select the database table you want to base the query on

     A **Query** window will appear with a dialog box asking which **Table** or **Tables** contain
     the data to create the **Query** (see illustration on next page).

2   Choose **Customers** from the list of Tables in the **Show Table** window.
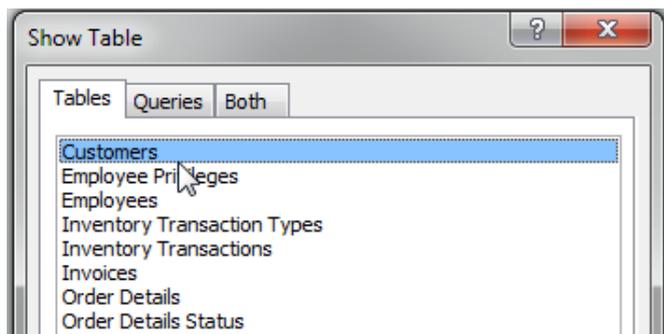
Having chosen the **Table** or Tables, it is then a question of choosing the **Fields**. This can be done by dragging from the **Table** onto the grid, double-clicking on the required **Field**, or by using the Drop-down menu that appears in each **Field** cell on the grid. You can add the **Fields** in any order you wish; this allows you to view the query result in whatever layout you prefer.
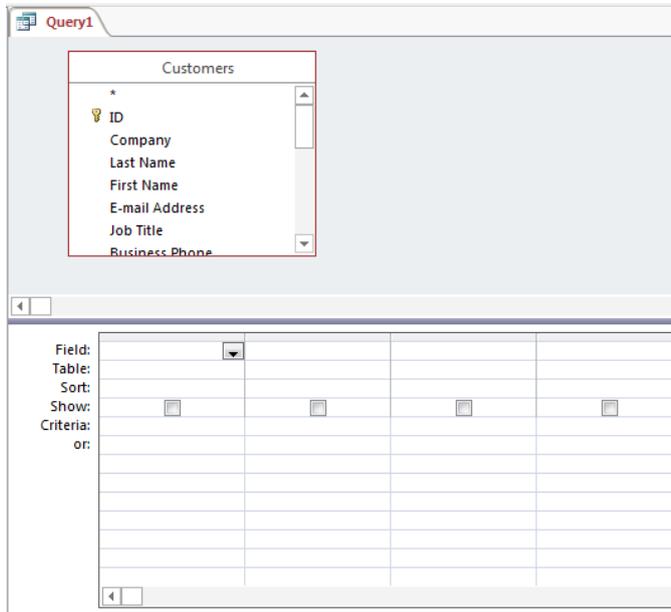
3   Choose Last Name, First Name, Company, Job Title and City from the Customers Table.

4   Under Last Name, click in the Sort row and select Ascending.

This **Simple Query** will produce a result either by using the **Run** button to actually perform the query or the query **View** button to preview the result. If you don't see this, make sure that the **Design** tab is active.

Last Name	First Name	Company	Job Title	City
Eggerer	Alexander	Company S	Accounting Assistant	Los Angelas
Raghav	Amritansh	Company BB	Purchasing Manager	Memphis
Ludick	Andre	Company M	Purchasing Representati	Memphis
Bedecs	Anna	Company A	Owner	Seattle
Gratacos Solso	Antonio	Company B	Owner	Boston
Tham	Bernard	Company U	Accounting Manager	Minneapolis
Grilo	Carlos	Company N	Purchasing Representati	Denver
Autier Miconi	Catherine	Company R	Purchasing Representati	Boston
Lee	Christina	Company D	Purchasing Manager	New York
Goldschmidt	Daniel	Company P	Purchasing Representati	San Francisco
Andersen	Elizabeth	Company H	Purchasing Representati	Portland

5   To save the result, simply click on the **Save** button and name the **Query**.

b. Using Criteria

Queries can produce more complex results by using the **Criteria** option in the **Query** grid. This allows for the use of text, numbers, wildcards and of the combining of criteria using

Task 17

1   You should follow the examples below as they are demonstrated by your tutor. If time allows, try one or more of them yourself afterwards.

c. AND plus OR.

Within a single column you can use **AND** to specify a range of values that will be contained in the records accessed by the query on the **Products** table.
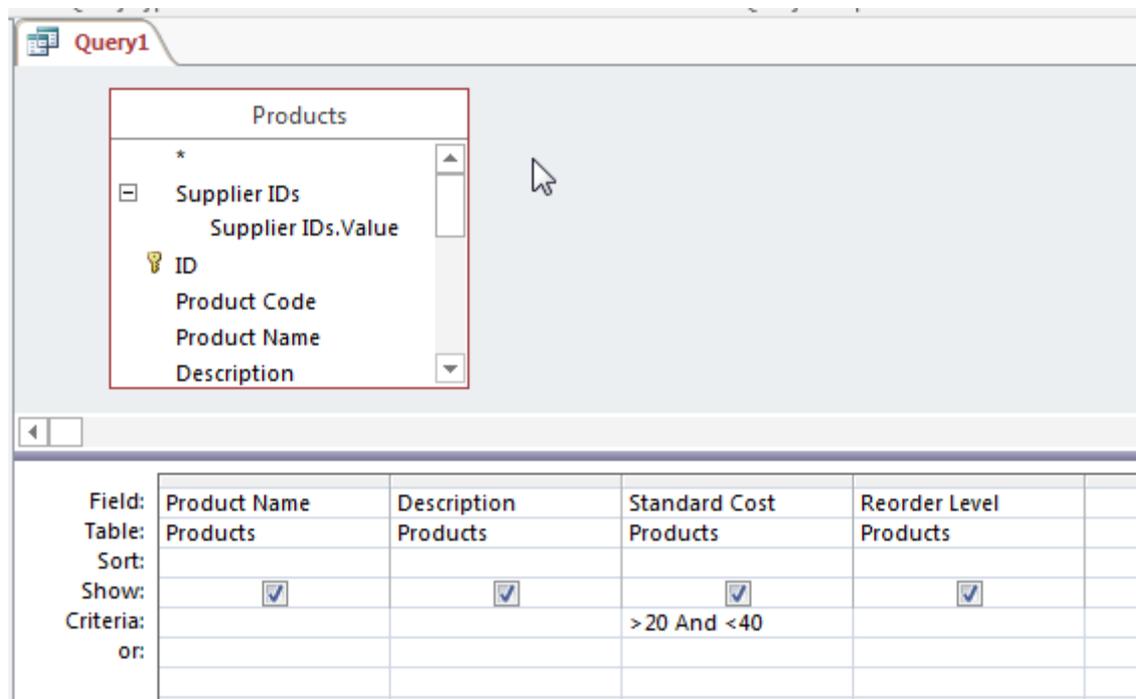
For instance, here we query Standard Costs between £20 and £40. This is much the same kind of analysis as you make using a filter but queries can do more.

Here is a query to find results between two different numerical fields using *comparison symbols* such as **AND** (if you enter criteria in different columns they will be **AND**ed):

Field:	Product Name	Description	Standard Cost	Reorder Level	
Table:	Products	Products	Products	Products	
Sort:					
Show:	☑	☑	☑	☑	
Criteria:			>15	>20	
or:					
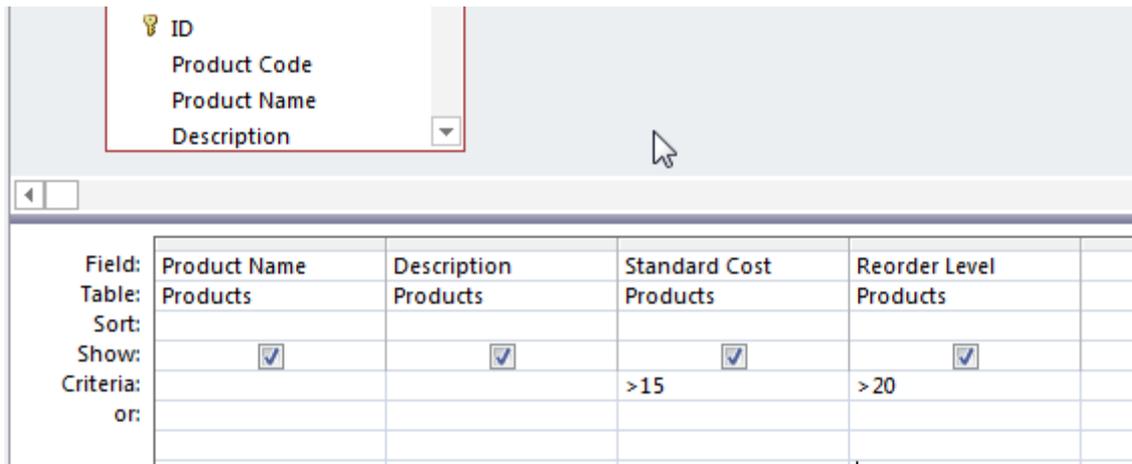
This example finds all **Employees** in two locations using **OR**:

Field:	First Name	Last Name	E-mail Address	City	
Table:	Employees	Employees	Employees	Employees	
Sort:					
Show:	☑	☑	☑	☑	
Criteria:				="Denver"	
or:				="Seattle"	
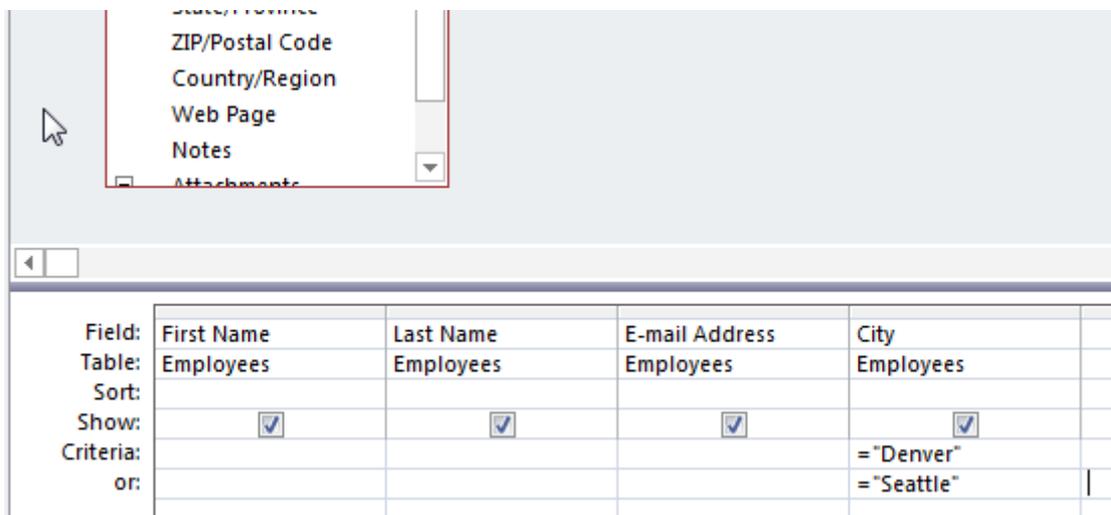
In the following examples, a **Query** is run to display all **Customers** who work in their firms'
accounting departments.

Field:	First Name	Last Name	E-mail Address	Job Title	
Table:	Employees	Employees	Employees	Employees	
Sort:					
Show:	☑	☑	☑	☑	
Criteria:				Like "*accounting"	
or:					
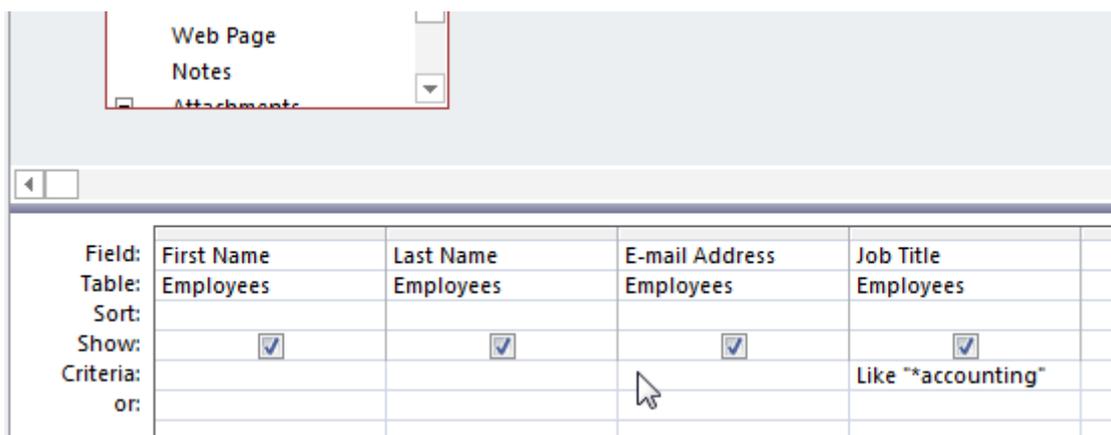
Note: the expression *like "*accounting*"* is the equivalent to *contains*, used in the last
session. The "*" is a wild card character used to indicate that any text can appear in its
place – here, either before or after the other text entered.

Symbols used in Queries	
>	Greater than
<	Less than
=	Equal to (not usually used because assumed if no other criteria symbol present)
>=	Greater than or equal to
<=	Less than or equal to
Between	Tests whether data falls between two values (Note this is same as > X AND < Y)
#DD/MM/YY#	**#** a sign used to signify date.
Wildcards	
*	Matches any number of letters or numbers, used in this form: *Like "Lon*"* to find all words beginning with Lon
#	Matches any single digit, used in this form: *Like "###"* finds all three-digit numbers
?	Matches any single character, used in this form: *Like "M??t"* finds Meat, Meet, Moat, etc.

Task 18

1   Open the query, Product Category Sales by date, in Design view.

d. More Complex Queries

So far we've only looked at queries on single tables. What if you wish to extract information from more than one table? Let's look at an example of this.

Note that *three tables* have been selected to form this *new query* (see next page). The tables are linked by lines: these lines show the fields in each table *by which they are connected*. If common fields like this don't exist (e.g. like **Products ID** in the example here) then you can't join tables to create queries.

Any fields used to link tables *must be unique for at least one of the tables*. This prevents duplication of records selected when the query is run. Thus *key fields* are used to link tables: these are marked with a **key icon**. 🔑

The primary key field in one table must match up with a similar field in at least one of the other tables in a query. For instance, the primary key field in **Products** (in the example shown below) matches a field in the **Order Details** table. Note that this is not the key field for **Order Details** but they both refer to the same data.

Such keys are known as *Foreign Keys* and may or may not be unique (i.e. may or may not be primary key fields). Tables with foreign keys will always also have their own *Primary Key* which uniquely identifies the records in the table. Note here that **Products** and **Order Details** are joined by one key field while **Order Details** and **Orders** are joined by a different one.
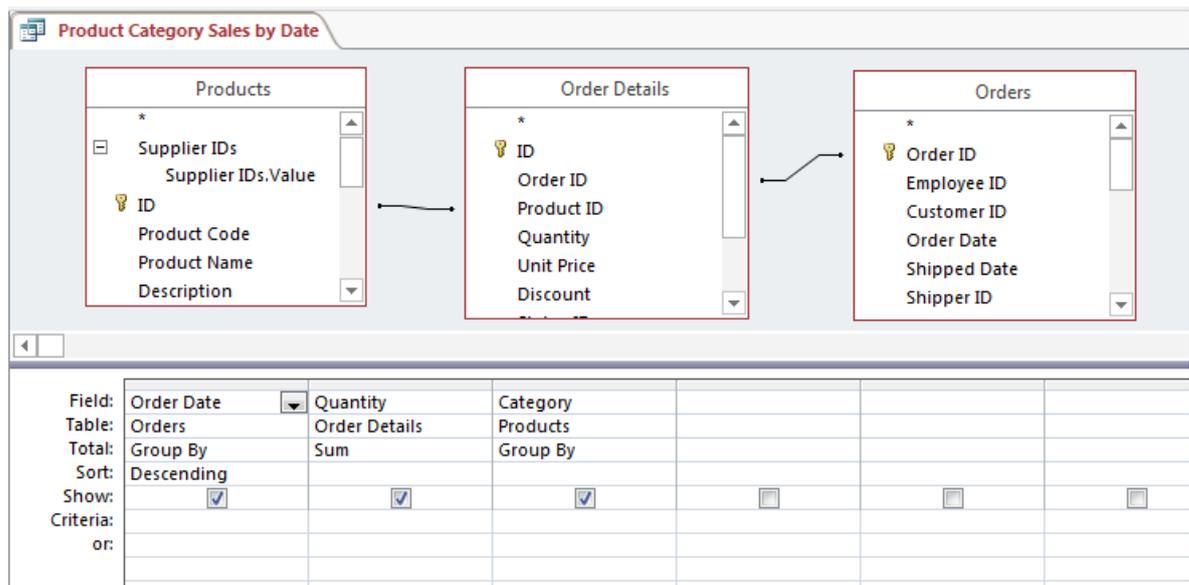
The key fields don't need to be selected for the query fields, unless they form part of the information you wish to extract. You select those fields from the tables that will give you the data you require. The query will display the data all together as if from one table.

Now you will create a complex query for yourself. Before you start look at the following tables quickly: **Customers**, **Order Details** and **Products**. Take a note of two product names. View each table in Design view and try to see where there are shared key fields.

The query you will create is to find some details about customers who have ordered particular products. You will create the query using the tables you've just looked at and you will design it so that a user can input any product name they want to look up.

Task 19

1 Open a new query in **Design** view. Add the tables to the query in the order given above: **Customers**, **Order Details** and **Products**.

2 As you add the tables, you should see *Access* automatically "join" the tables at each relevant key field. If this does not happen then tables have not been added properly.

3 Now drag some fields into the lower window. You can make your own choice except that one field must be **Company Name** and another **Product Name** (it should be obvious why you should do this).

e. Sorting Results in a Query

To arrange the results of a **Query**, you can use the **Sort** option in the **Query** grid. Chose the field you want to sort on and then choose whether you want ascending or descending order.

Field:	Product ID	Order ID	Order Date	Qu
Table:	Order Details	Order Details	Orders	Or
Total:	Group By	Group By	Group By	Su
Sort:	⊡		Descending	
Show:	Ascending	☑	☑	
Criteria:	Descending			
or:	(not sorted)			

If you don't choose this option fields will appear in unsorted order. The field you choose will depend on which part of the information extracted is most important. For instance if you need to know names of employees, sort on the last name; if you were interested in shipping dates you would sort on the dates.

You can of course always sort or filter the results of a query in the same way as you would any normal table. Query results are usually output as a table.

f. Creating a Parameter Query

A parameter query lets you enter specific search criteria every time you run a query. For example, let's say you are looking for a product in the **Northwind** sample database that is a particular price. Consider a simple product query in **Design View**:

Field:	ID	Product Name	Standard Cost	
Table:	Products	Products	Products	
Total:	Group By	Group By	Group By	
Sort:				
Show:	☑	☑	☑	
Criteria:			[Enter product price]	
or:				

This query will retrieve the **ID**, **Name**, and **Cost** of a product. We can transform this query into a parameter query by adding a new type of command to the **Criteria** row of a particular field. Add the text "[*Enter product price:*]*"* into the **Criteria** row.

As you will see in the next lesson, this text will appear in a special dialog box that will prompt the user to enter a specific value.

g. Using Parameter Queries

Once parameter text within square brackets has been entered into the Criteria cell of a field, running the query will produce a dialog box:
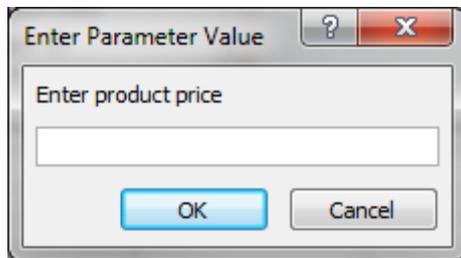
If you enter 7.50 into the text box and click **OK**, all products that are £7.50 in price will be displayed:

That's all there is to using a parameter query. Parameter queries will incorporate the user input directly into the background design of the query before it is executed. You can have multiple parameters inside a query; they are filled in from left to right (if looking at the layout of the fields in Design view).

Task 20

1   If you haven't already done so, create the Query described above and save it as *Products Price Query*.

2   Open the **Product Purchases** Query then add a *Criterion* to allow input of a creation date: "*[Enter creation date:]*". Test your query by running it.

3   Save your query and name it *Products Date Query*.

Note: what you've learned so far about creating queries will stand you in good stead for creating some useful, though basic, queries. But we've only really scratched the surface of queries.

4 Create a New (Blank) Database

Now that you are familiar with Access and what it can do, you will create your own, new database. Close your current database then select **File ➔ New**. You will see the window view that was displayed when you first opened *MS Access 2016*.
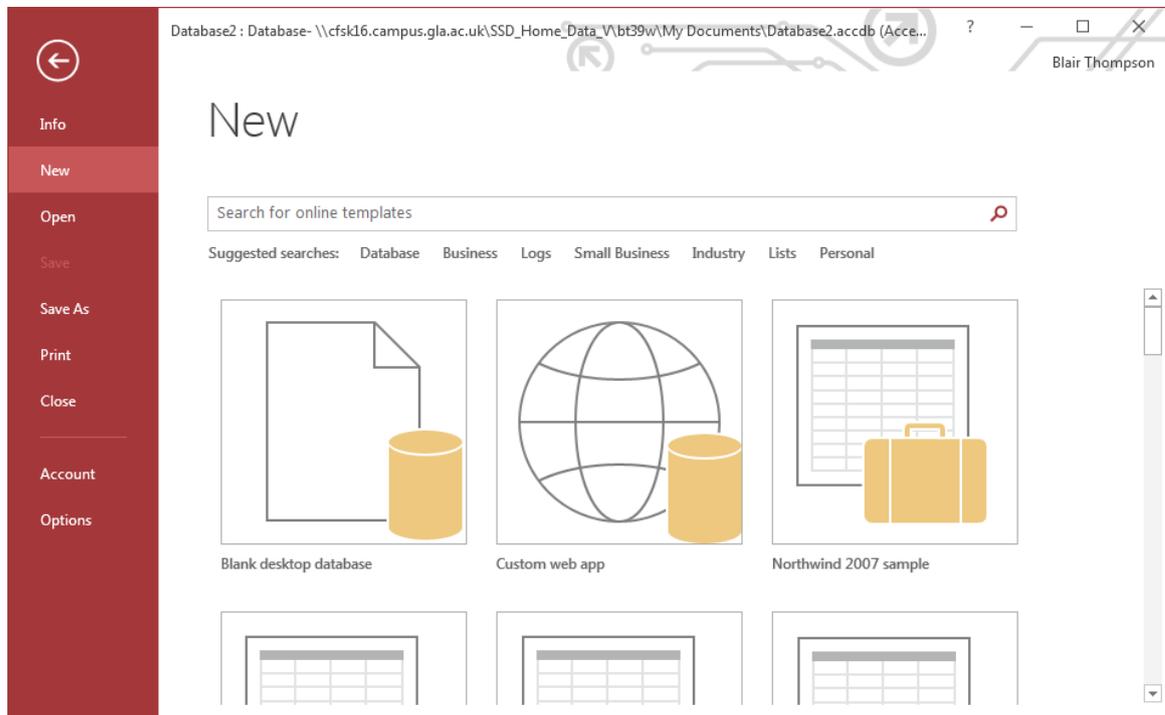
There are several options but just click on the **Blank desktop database** option.
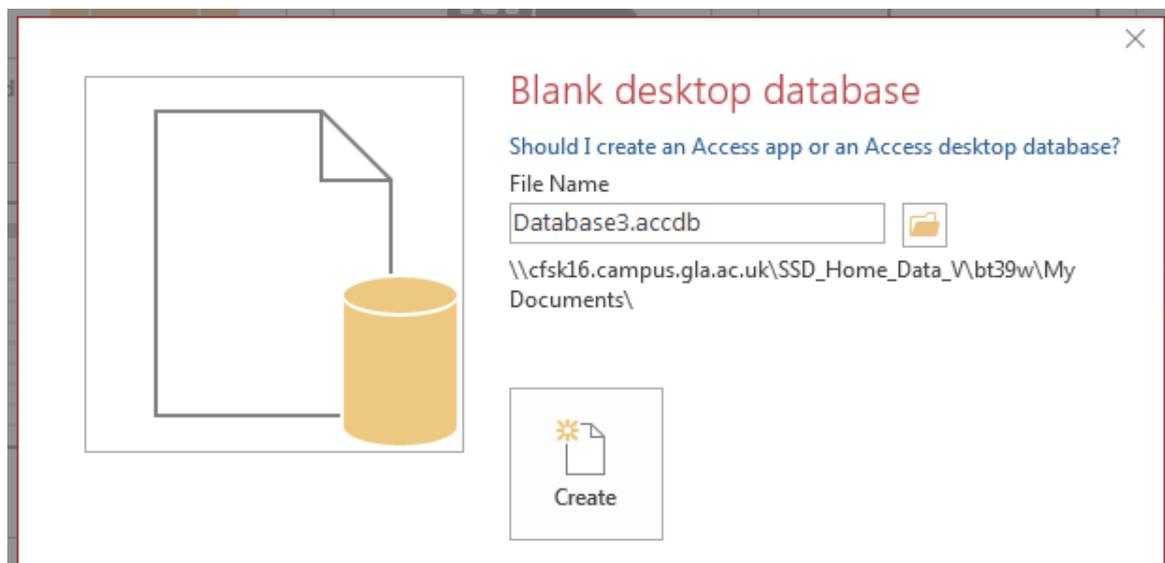
When you choose this option, the **Database Properties** pane will display a filename. This is usually something like"Database1" but it allows you to change this name (giving a more meaningful name) and store it in a suitable location. Next click the **Create** button to create the blank database file.

When the **Blank** database option is chosen, all its basic elements must be set up immediately, even before any records to be included in the database have been created.

This means creating any tables you wish to be included (the simplest database has only one table, most will have two or more). By default the blank database gives you a new table – you can rename this and set its properties. And you can add as many other new tables as you require.

5 Creating a New Table

a. Tables

Tables are the basis of all databases. Tables can be created in a number of ways. Use the **Tables** section of the **Create** tab to make a table:
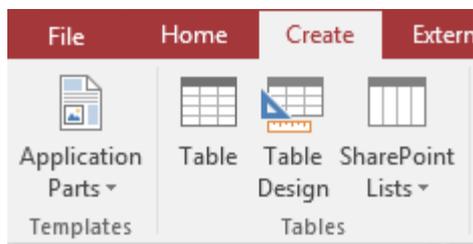
Clicking the **Table** command creates a new table. A new tab will open, containing the table in **Datasheet View**: you must save the table right away giving it a suitable name.
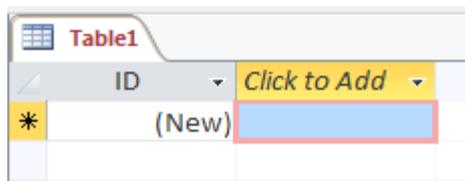
Note: when you use the **Table** command, *Access* creates a **Primary Key** field for you with an **AutoNumber** format. It names this field **ID**.

Click inside the **Click to Add** column and start entering data. Press **Enter** to keep adding fields to the record, or press **Tab**. As soon as you begin to add data to a new field, a new record is created below the record you are entering data into.

Once you stop entering data into a record and move to the next record, that will set the number of fields in all the records (unless you use the **Add New Field** option to create a new field – when you must go back and enter data in earlier records for the new field).

Field Properties: *Access* will make a best guess at the field properties based on what you enter into a field. For instance, if you enter numbers only it will set it as a numeric field, if you enter text or a mixture of text and number it will set it as a text field.

Or, if you enter a date, it will set it as a date field (and the next time you go to add or edit data in this field a small calendar icon will appear). And so on.

b. Design View

Using the **Design View** gives the greatest level of control over table creation, and is therefore to be recommended. Once **Design view** has been chosen, **Access** displays a blank table.
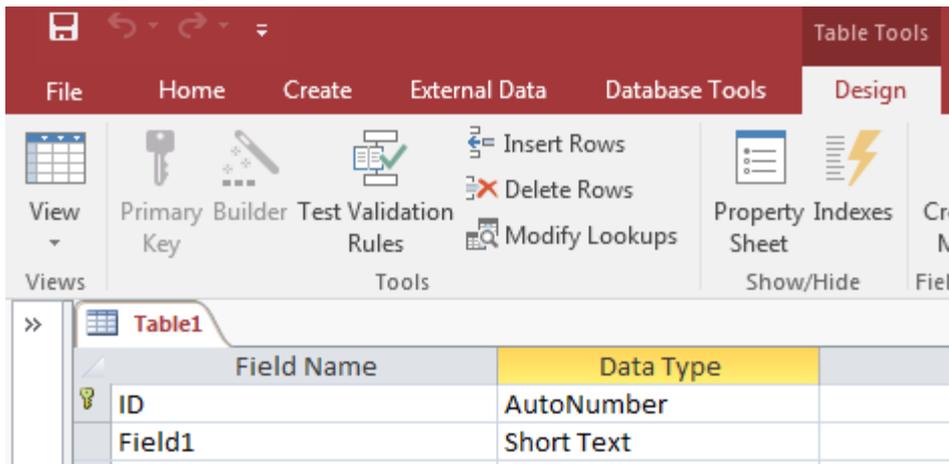
Design View includes its own **Design** tab in a contextual tab. You have the ability to add your own choice of **Primary Key** field, construct custom formulas, insert or delete different fields, and more.

Using **Design View** is more in-depth than simply entering data into fields. You can specify the field name, its data type, and give the field some sort of description if you like. It is also a lot easier to specify certain fields, such as YES/NO fields.

Task 21

1   Create a new, blank database using **Design View**. Name it *Courses 2016*, saving it to your **My Documents** folder.

2   Create a new table in **Design View**: you will add fields and set properties for the fields you add following the guidance given in the following pages.

When your table is ready, you will add your records to it. To ensure that records are added properly, you must first design their layout and format. You do this by creating **Fields** – these **Fields** must contain data of a specified type, such as *Short Text*, *Number*, *Date*.

At the bottom of **Design View** is the **Field Properties** section. Here you can modify all of the properties of a particular field.
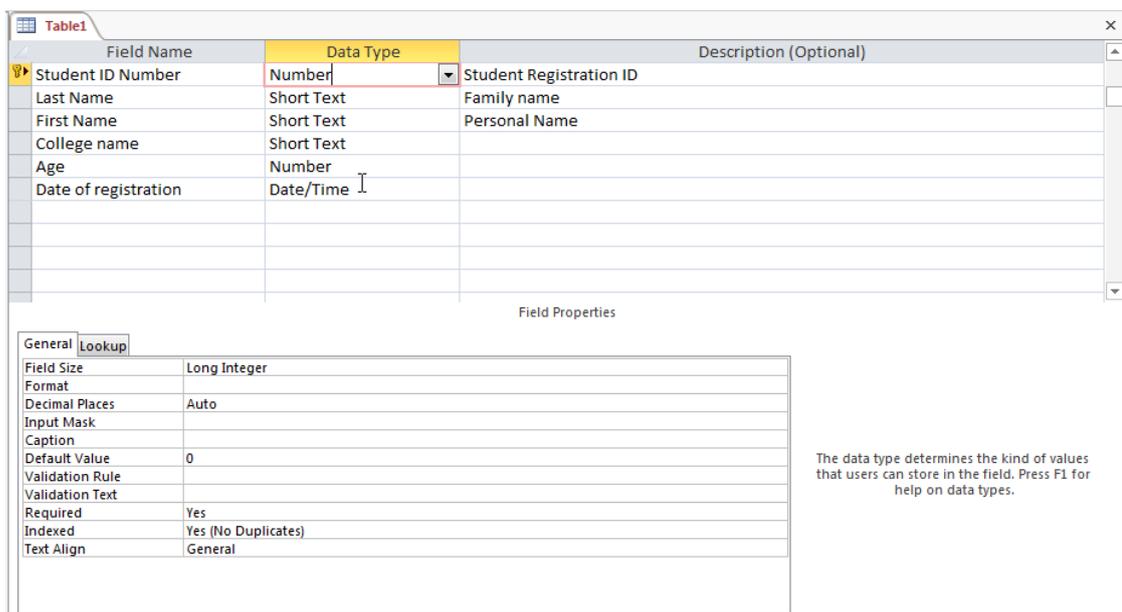
Fields should contain data about *one thing*; for example a **Table** may contain information about students. Therefore, the kinds of **Fields** it can contain are: student registration (ID) number, name, telephone number, and so on. In other words, any information related to the student and required for contacting them.

Before you create your fields, you should think carefully about *the types of data* you require: do you need names? Do you need addresses, dates, exam marks? Some require text, some number, some date format.

The available **Fields** options and a description of these options are contained in the table in the appendix at the end of this document.

This also show maximum sizes of data for each field type: this can be limited in the **Field Properties** options, which is useful for saving space in your database. For instance, why leave a text field at the default 50 characters when it should never have more than 15 characters entered into it?
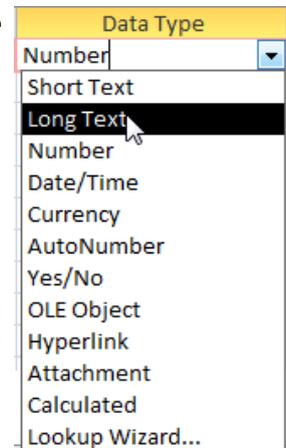
Task 22

1   In the new **Table** you created, enter 6 fields: Student ID Number, Last Name, Forename, College Name, Date of birth and Date of Registration.

2   Select the appropriate data type for each field: but set Student ID Number to **Number** format, not **AutoNumber**.

3   Set a suitable field size for each item you enter.

c. Primary Key

Primary Key: The student example is a very useful example for examining this. As was mentioned earlier, every table should have a **Primary Key** field and this field must produce only unique values. Since every student has a different ID number, this will make a natural **Primary Key** field, uniquely identifying each student.

If you set a field as the **Primary Key**, *Access* will prevent you adding any record where the key is duplicated – *it will force you to use unique data for that field*. **Last Name**, for instance, is no use since many people have the same name. It is best to use a **Field** with a number *that will not change or be used in calculations*.

Where there is no natural **Primary Key** field available, you can use *Access's* **AutoNumber** data type which will automatically create a unique number every time you add a new record.

Where there is a suitable field to act as **Primary Key**, select it and then click on the **Primary Key** button on the toolbar.

This will automatically alert *Access* to treat it as an Indexed field, that is, it will check any values input and reject them if they are duplicated in existing records.
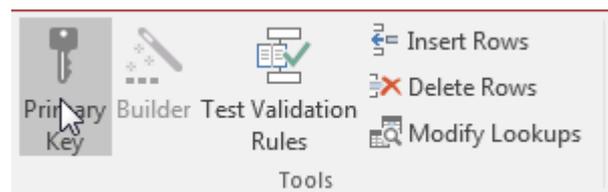
Task 23

1   Select the *Student ID Number* field. Set this field as the **Primary Key**.

2   Note that when you do this, the **Indexed** row under **Field Properties** changes to say "*Yes (No Duplicates)*".

	Field Name	Data Type
🔑	Student ID Number	Number
	Last Name	Short Text
	First Name	Short Text

The key icon will now appear beside the field name.

d.   Validating Data

Specifying **Field** types is not just a matter of logical consistency; it can help validate data that is input by users by limiting what they can actually enter. *This can cut down on errors in data entry.* For instance, if you specify a **Number** type, a database user will not be able to enter text.

The **Field** properties can be used for further validation, e.g. by limiting a number to percentage or currency or limiting the number of decimal places. Another way of validating data is to use an **Input Mask**.

Input Mask Example

When you create or update a table in **Design View**, select the **Field** which you require to have an input mask. (You can only apply a mask to text or date fields).

General	Lookup	
Field Size	255	
Format		
Input Mask		...
Caption		
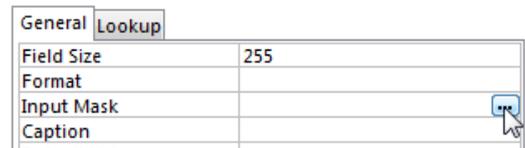
Then, under **Field Properties**, select **Input Mask**. A button will appear at the side with 3 little dots (**…**). Click this to open the **Input Mask Wizard**.
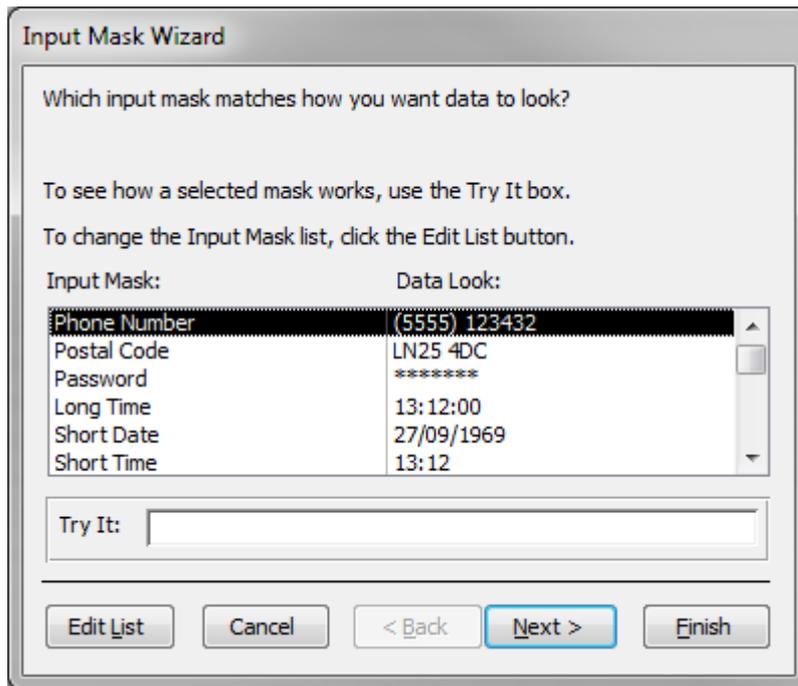
Select the type of mask you want (e.g. telephone number, post code, password).

You can edit the mask to some degree: here, a different place holder for data entry is selected. If, for instance, you selected a date input mask of **Medium Date**, you would only be able to enter dates in one of several date forms that you select: (e.g.) *27-Jan-99*. Other typical masks are for post codes and phone numbers.

Note: the input mask is a different thing from a field format. The format controls how the field will be displayed and what type of data is entered. The input mask controls how the data is actually entered.

Task 24

1   Now add an input mask to your date field. Choose the **Medium Date** input mask.

e.   Validation Rule

Validation Rule: In addition to adding protection like required values and input masks, you can also add validation rules to your database to ensure that data entered makes sense.

By clicking in the **Validation Rule** box, you can enter a simple expression, e.g. "*>=0",">0 AND <10*". This is very useful when you want to limit input to a small range of numbers.

Always remember though that no validation can be perfect and users entering data must take care to ensure that correct data is entered. For instance, limiting input between 1 and 10 will not stop the user entering 5 instead of 6 out of carelessness.

Task 24

1   Now add a validation rule to your **Age** field. Allow only numbers *between 17 and 25* (this is a very ageist institution!).

f.   The Lookup Wizard

The **Lookup Wizard** can limit the data entry user to a limited number of choices which they select from a drop-down menu – this reduces error by both limiting options and ensuring correct spelling/number order. You will perform a task making use of the **Lookup Wizard** shortly.

Vehicle ID #	Manufacture Date	Manufacturer's N	Model	Country ID	Cylinders	Add New Field
1	1982	Chevrolet	Corvette	1	8	
2	2003	Aston Martin	V12 Vanquish	2	12	
3	2000	Honda	S2000	4	4	
4	2003	Hyundai	Tiburon	5	4	
5	2002	Chevrolet	575 Marinello	3	12	
6	1979	Aston Martin	Spider	3	4	
7	1965	Honda	Falcon	1	8	
* (New)		Hyundai			6	
		Ferrari				
		Fiat				
		Ford				

In the **Lookup Wizard**, you can enter either a static list of values or specify a source for the values that you want to retrieve, such as a field in a table. The data type of a Lookup field is either **Text** or **Number**, depending on the choices that you make in the wizard.

From the **Data Type** column choose **Lookup Wizard**. This brings up a dialog box that gives the default option of using values already used in the database, or of entering new values (the option taken in this example).

Data Type
AutoNumber
Number
Number
Text
Memo
Number
Date/Time
Currency
AutoNumber
Yes/No
OLE Object
Hyperlink
Attachment
Lookup Wizard...
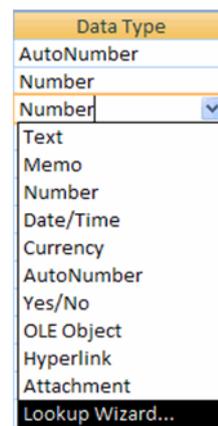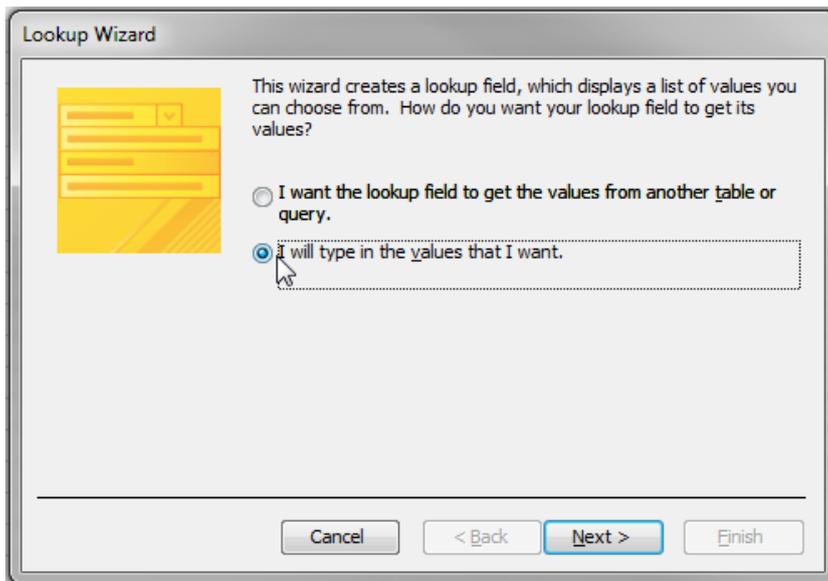
The next dialog box of the **Wizard** is where you enter the values you want to use for the value list. For instance, here we'll enter the range of car manufacturers.
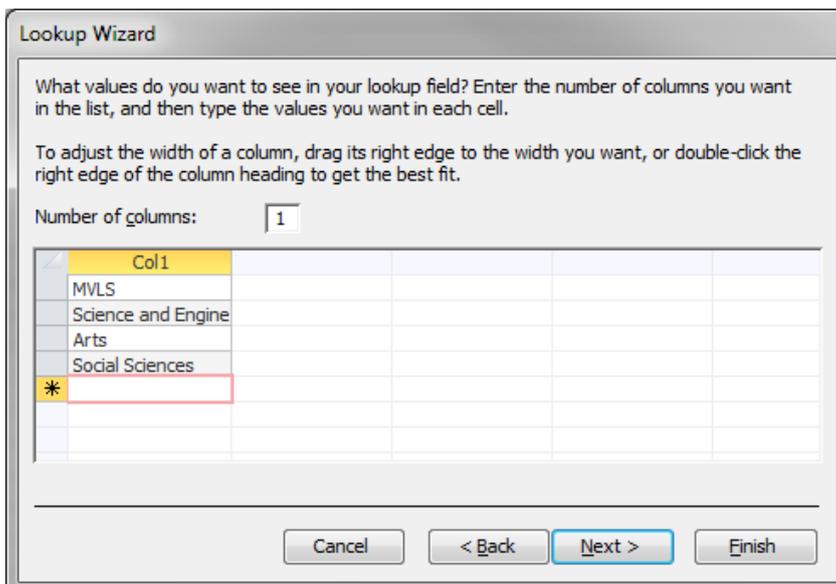
In this page of the **Wizard** you can specify the number of columns for the value list and which values you want to include in the list. (The majority of lookup fields/value lists you will use will only be a single field at a time.)

Click your mouse inside the first cell, type a value, and press **Tab** on your keyboard to move to the next cell. When you have entered the list of values you want to use, click the **Next** button.

The final step of the **Wizard** asks you to name the lookup column (value list). The default name is the same name as the field, but you can name it whatever you like. Click **Finish** to complete the Wizard.

Now, when you open **Datasheet View** for the table, you will be able to use the drop-down list to fill in a value for the field.

Task 24

1   Now create a **Look-up list** for one of your fields. If you have added the fields suggested earlier, an obvious choice will be *College Name*. Four or five options will be enough for the exercise.

g. Saving a Table

Once all the fields have been entered into the **Table**, choose the **Save** icon from the toolbar and give the **Table** a name in the dialog box.

Task 25

1   Make a last check that all your fields are entered and formatted as you want them to be and check you have set a **Primary Key** field.

2   Save the table, naming it *Student Details Table* (use the save button on the **Quick Access** toolbar).

3   If you should forget to set a **Primary Key**, a warning message will appear when you save your file.

4   If you see this message, cancel the save action and set a Primary Key. If you don't think you have a field that is suitable, then insert a **Field** with an **AutoNumber** format, to give each record a *unique number* to *identify* it.

6 Relationships between Tables

You have already seen how **Primary Keys** link tables when creating queries. You may have wondered whether it might not have been better to have just one big table so you didn't have to do this.

A well-organised database will be divided up into **Tables** that contain different categories of data (*fields*). For instance, take the example of a database holding information about people and the dwellings (houses, flats, etc.) they lived in.

The details of a dwelling (*post code*, *number of rooms*, etc.):

Household I	Postcode	Street	Number	Town	Tenure	Click to Add
1	G31 7SP	Nimmo Crescent	23	Glasgow	HSG Assocn Rented	
2	G31 8CH	Mainwairing Drive	401	Glasgow	Council Rented	
3	PA4 7BH	Abbey Road	11	Paisley	PBD HA Rented	
4	EH43 6AB	Nether Bakehouse	23	Edinburgh	Privately Owned	
5	EH12 3NT	Victory Road	127	Edinburgh	Privately Owned	
6	DD8 2BN	Peenbovril Street	20	Dundee	Council Rented	
*	(New)					

- would be in a separate **Table** from the details of the people who *live* in the dwelling (*name, sex, age,* etc.):

Person ID	Surname	Forename	Sex	Age	Occupation	dwellingID
1	Smith	Allan	Male	34	Analyst	1
2	Smith	Betty	Female	29	Secretary	1
3	Mathie	Irene	Female	23	Therapist	2

For a dwelling to be identified with an individual, the *two* **Tables** should be *related*. This is accomplished by having a **Field** in the *People* **Table** which is common to both tables.

Field Name	Data Type	Description
Person ID	AutoNumber	
Surname	Text	surname of housedweller
Forename	Text	forneame of housedweller
Sex	Text	male/female
Age	Number	age in years
Occupation	Text	employment details/houseperson/child
dwellingID	Number	match for houseID in Dwelllings table

A **Field** identifying **dwellings** should be inserted into the **Table** containing data about *people* – in the example, **DwellingID**. This field would have the same format and properties of, and would refer to, a matching field in the **Dwellings Table**.

Both fields are *key fields*. As you can see, the field is a *Primary Key* in the **Dwellings Table** (**Household ID**) and a *Foreign Key* in the **People Table** (**DwellingID**). This will allow *Access* to link them together so that they can be queried and related data extracted from both simultaneously.

How does this help, apart from being more logical? Well suppose you only had one table for this data – the first one – and all the details about the people living in the dwellings were included in this.

You would have to have a field in the **Dwelling** table for *every person* in the dwelling – and how do you know in advance how many people will live in a dwelling? Large families are rare nowadays but if your table is likely to have some records like that included you must design it so that *every record* can include them. That would mean a lot of extra fields which in most cases would remain empty.

And if you added the dwelling data to the people data there would then be the problem of having to add all the dwelling data to the records of every person who lived in that dwelling. This would mean a lot of duplication. So whatever way you did it, having a single table would waste space and slow down search time.

To make this clearer, we'll examine an example database with tables set up as described above – but with little actual data.

Task 26

1   Download the *Dwelling sample database* from the web site where you previously found the *Northwind* database. Open the table, **Addresses**, in **Design View** and examine it. (There is no data in this table).

2   Note there are two separate fields for *occupation* – one for main tenant and one for their partner. There is a problem here: what if there is only one person in the flat or what if there are three or more sharing (e.g. as with students).

3   Now think of the space wasted if any adults living in a dwelling have no children or are not renting rooms to other people – the database must preserve that space in case it's required. Or what if someone has 4 or 5 children? There is only room for 3.

4   Open each of the tables, **Dwelling** and **People**. The first thing to note is that, *combined*, these two tables have fewer fields than the first table you looked at.

5   Not only that but despite having fewer fields they can hold more data without wasting space. For instance you can now have any number of occupants for a dwelling with

no wasted space. And other duplicated fields (in **Addresses**) are no longer required – you only need *Sex, Age* and *Occupation* to occur once for each record in the **People** table.

b.   Creating a Relationship

How do we ensure that these tables will work together properly? We do this by creating a *relationship* through a **Primary Key**.

1   Select the **Database Tools** tab on the *Access* ribbon.

2   Next, click on the **Relationships** button on the toolbar.
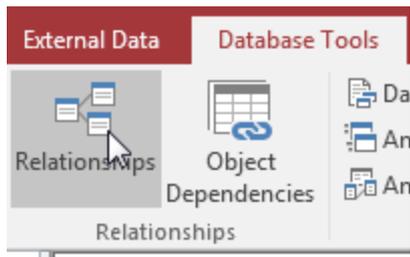
3   The **Show Table** dialog box will appear. Here, choose the tables that you wish to join in relationship.
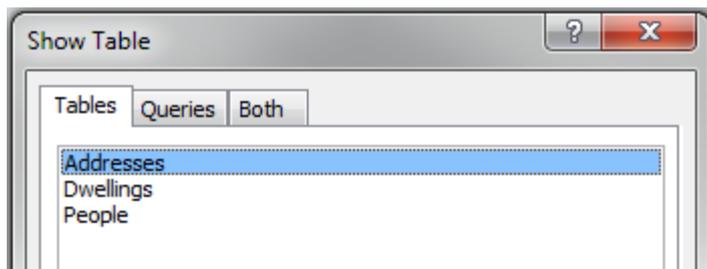
4   Select each table in turn and click the **Add** button.

5   When you've selected all the tables, click **Close**.

6   Now we must decide on a field that both tables share which can be used to join them in the relationship.

7   To do this, choose **Edit Relationships** from the **Relationships** menu – this is available when the **Relationships** window is open. This opens the **Edit Relationships** dialog box: since we have none to edit we create a new one by clicking the **Create New** button.
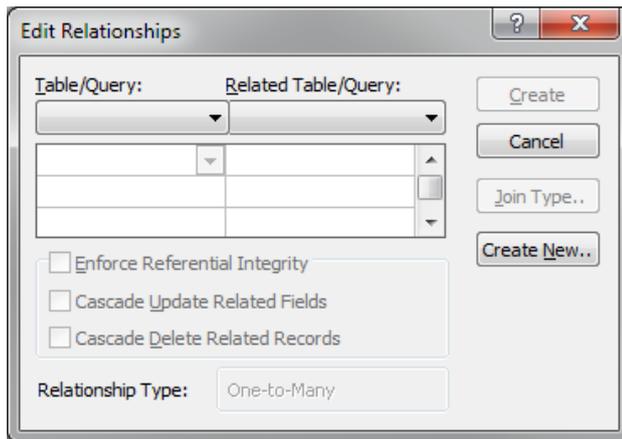
8   Next select the fields from each table which will be the "join" of the relationship. Use the **Table Name** options to select the tables to be joined and the Column name options to select the Fields that will make the join. Remember: they must be **Key Fields**!
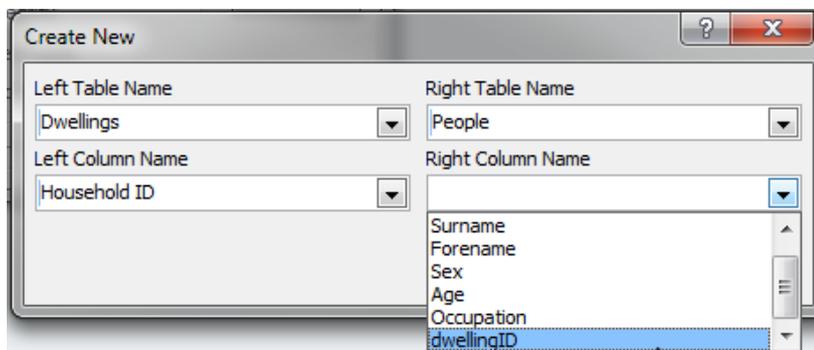
The join must make sense – for instance there is no point in joining *Household ID* in **Dwellings** with *Age* in **People**. This is meaningless and will never produce useful information. In fact in this example there is only one possible join – *Household ID* in **Dwellings** with *Dwelling ID* in **People**. Only these two fields have similar formats and data; and, in effect, the same data.

Not only that, *Household ID* is a **Primary Key** field for **Dwellings** and *Dwelling ID* in **People** is a **Foreign Key** because its data is the same as that of the *Household ID* primary key. Thus two key fields are linked to create the relationship.

9   Click **OK** to confirm your selection and the window will appear like this.
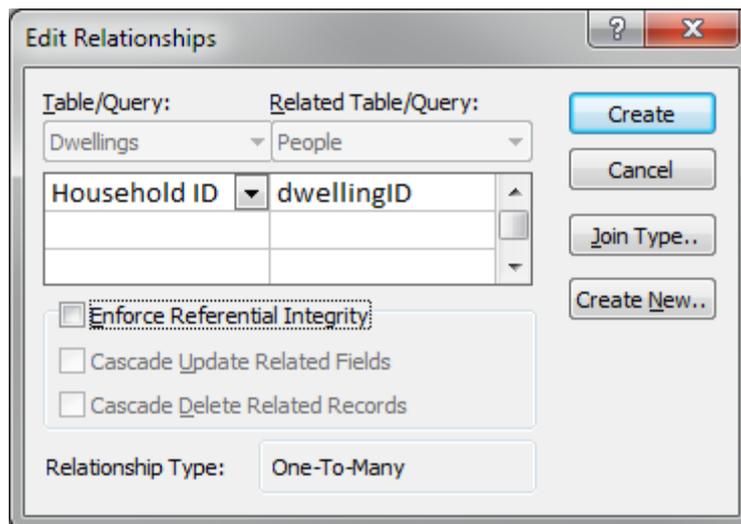
10   Tick the **Enforce Referential Integrity** box and then click the **Create** button.

You will now see the relationship marked by a line joining the two tables. The two symbols "**1**" and "**∞**" (one and infinity) represent the join type "one-to-many". Each dwelling can have many people in it but each person only has one dwelling. This is the most common type of relationship and the only one we can cover in this course.
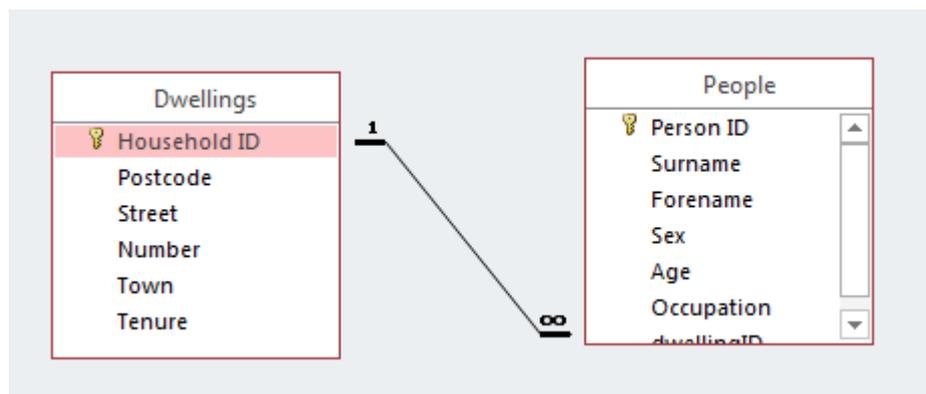
The following links will explain table relationships in more detail:

https://goo.gl/5c1ZHH

databases.about.com/b/2010/10/03/creating-relationships-in-access-2010.htm

www.addictivetips.com/microsoft-office/ms-access-2010-relationship-query-interconnect/

http://goo.gl/c1cbgI

7   Course Review and Revision

What you have learned should be enough to allow you to create and do useful work with relatively simple databases which incorporate a small number of tables. You should make sure you have mastered the material here before attempting more advanced options in *Access*.

Help and online tutoring can be found at:

http://www.functionx.com/access/Lesson01.htm

http://www.gcflearnfree.org/access2010

http://www.youtube.com/watch?v=_0RSuJZTDQc

http://www.youtube.com/watch?v=ijsDlEbqS4g&feature=related

You should now have completed all the tasks in this course. Make sure you have submitted the assignments (if this is required of you), which are the database files you worked on during this course. If still find you have a problem with anything, please ask your tutor for advice.

You will now be asked to complete a short, follow-up questionnaire, which can be found at: *www.gla.ac.uk/services/it/training/coursefeedback/*

Make sure you have saved all the work you did on the course. Log off from your PC and make sure you take away any materials related to the course and remove any disks or other portable storage devices you may have been using.

8 Appendix I: Data Types

The following table summarizes all the field data types available in **Microsoft Access**, their uses, and their storage sizes.

a. Microsoft Access 2016 Data Types

Data Type	Description
AutoNumber	An AutoNumber field creates unique values automatically when Access creates a new record. The AutoNumber field is primarily used for Primary Keys in Access.
Short Text	A Text field can contain values that are text, numeric or a combination of both. A text field can contain a maximum length of 255 characters.
Long Text	A much larger version of the text field, allowing storage of up to 1 GB of data.
Number	The Number field can store numeric values up to 16 bytes of data.
Date/Time	The Date/Time field allows storage of date and time information. The date/time field now also includes the Auto Calendar feature.
Currency	The Currency data type stores values in a monetary format. This can be used with financial data as 8-byte numbers with precision to four decimal places.
Yes/No	Boolean data storage of true/false values.
OLE Object	The OLE Object field stores images, documents, graphs etc. from Office and Windows based programs. The maximum data size is 2 GB, although this will slow down a database.
Hyperlink	The Hyperlink field type is used to store web addresses. This has a maximum size limit of 1 GB of data.
Attachment	The Attachment field type is used to store images, spreadsheet files, documents, charts and other types of supported files to the records in your database. This is a new feature that has been introduced with Microsoft Access 2007.
Calculated	Can store the calculated result using one or more fields from within the table

b. Lookup fields

You can set a field's data type to **Lookup Wizard**. Doing this starts the **Lookup Wizard**, which helps you create a Lookup field. A Lookup field displays either a list of values that is retrieved from a table or query, or it displays a static set of values that you specified when you created the field.

In the **Lookup Wizard**, you can enter either a static list of values or specify a source for the values that you want to retrieve, such as a field in a table. The data type of a Lookup field is either **Text** or **Number**, depending on the choices that you make in the wizard.

Note: Lookup fields have an additional set of field properties, which are located on the **Lookup** tab in the **Field Properties** pane.

c. Validation Rules for fields

To do this ...	Validation Rule for Fields	Explanation
Accept letters (a - z) only	Is Null OR Not Like "*[!a-z]*"	Any character outside the range A to Z is rejected. (Case insensitive.)
Accept digits (0 - 9) only	Is Null OR Not Like "*[!0-9]*"	Any character outside the range 0 to 9 is rejected. (Decimal point and negative sign rejected.)
Digits and letters only	Is Null OR Not Like "*[!((a-z) or (0-9))]*"	Accepts A to Z and 0 to 9, but no punctuation or other characters.
Exactly 8 characters	Is Null OR Like "????????"	The question mark stands for one character.
Exactly 4 digits	Is Null OR Between 1000 And 9999	For Number fields.
Exactly 4 digits	Is Null OR Like "####"	For Text fields.
Positive numbers only	Is Null OR >= 0	Remove the "=" if zero is not allowed either.
No more than 100%	Is Null OR Between -1 And 1	100% is 1. Use 0 instead of -1 if negative percentages are not allowed.
Not a future date	Is Null OR <= Date()	
Email address	Is Null OR ((Like "*?@?*.?*") AND (Not Like "*[,;]*"))	Requires at least one character, @, at least one character, dot, at least one character. Space, comma, and semicolon are not permitted.
You must fill in Field1	Not Null	Same as setting the field's Required property, but lets you create a custom message (in the Validation Text property.)
Limit to specific choices	Is Null OR "M" Or "F"	It is better to use a lookup table for the list, but this may be useful for simple choices such as Male/Female.
Limit to specific choices	Is Null OR IN (1, 2, 4, 8)	The IN operator may be simpler than several ORs.
Yes/No/Null field	Is Null OR 0 or -1	The Yes/No field in Access does not support Null as other databases do. To simulate a real Yes/No/Null data type, use a Number field (size Integer) with this rule. (Access uses 0 for False, and -1 for True.)

Useful Shortcut keys

Using keyboard shortcuts can help you become more efficient when creating documents in Microsoft applications. Most keyboard shortcuts require you to use two or more keys at the same time. To use a keyboard shortcut first press and hold down the modifier key or keys (i.e. SHIFT, CTRL, ALT) and then press the corresponding standard key on your keyboard.

Function	Shortcut
Go to "Tell me what you want to do"	ALT+Q
Open	CTRL+O
Save	CTRL+S
Close	CTRL+W
Cut	CTRL+X
Copy	CTRL+C
Paste	CTRL+V
Select all	CTRL+A
Bold	CTRL+B
Italic	CTRL+I
Underline	CTRL+U
Cancel	Esc
Undo	CTRL+Z
Re-do	CTRL+Y