

**Introduction
To
Microsoft Office
Access 2007.**

Contents

1.	INTRODUCTION.....	3
2.	STARTING ACCESS 2007	3
3.	THE USER INTERFACE	6
3.1	THE OFFICE BUTTON.....	6
3.2	THE QUICK ACCESS TOOLBAR	6
3.3	THE RIBBON.....	6
4.	TABLES OF DATA	7
4.1	ENTERING DATA	9
4.2	CREATING ADDITIONAL TABLES	9
4.3	RENAMING TABLES (AND OTHER DATABASE OBJECTS).....	10
5.	QUERYING THE DATABASE.....	11
5.1	COMPLEX QUERIES	12
5.2	FINE-TUNING QUERIES.....	13
5.3	USING QUERIES TO UPDATE TABLES.....	16
5.4	QUERIES THAT MAKE NEW TABLES	17
6.	GOOD PRACTICE IN DATABASE DESIGN.....	18
6.1	CODES OR NAMES	18
6.2	TABLE STRUCTURE	18
6.3	ADVANCED FIELD DESIGN	19
6.4	LOOKUP FIELDS.....	20
7.	IMPORTING AND EXPORTING DATA.....	21
7.1	IMPORTING TABLES.....	21
7.2	LINKING TABLES	22
7.3	EXPORTING DATA	22
8.	CREATING A DATA ENTRY FORM.....	23
9.	CREATING REPORTS.....	24

1. Introduction

Microsoft Access is a database management system. It allows information to be stored, organised and manipulated using a computer.

A database is an organised collection of related information, such as a telephone directory.

- Information is held in a series of tables. The tables can be linked so that information in one table can be related to information in a second table.
- In order to get specific information from a database, you would run a query. A query interrogates tables looking for any information that matches specified criteria.
- Results of queries can be printed out using a report. A report allows you to present information in a well-formatted and clear layout.
- Data-input and querying can be made easier using forms. A form is a well laid out front end to the database that offers input fields for data or query criteria.

Tables, queries, reports and forms are known as database objects. This document outlines the use of these objects to create and use a database.

In Access there are many ways to carry out these tasks. This document covers one way only; in order to get you started. Running through the document is an example database concerned with computer training courses.

Files produced in Access 2007 have the file extension ".mdbx", and the files themselves are very different from previous file formats. If you wish to use an old Access database in Access 2007, Access will switch to compatibility mode, to let you work on the old file with no problems. You can then save this file in Access 97-2003 format, or you can save it in the new Access 2007 format.

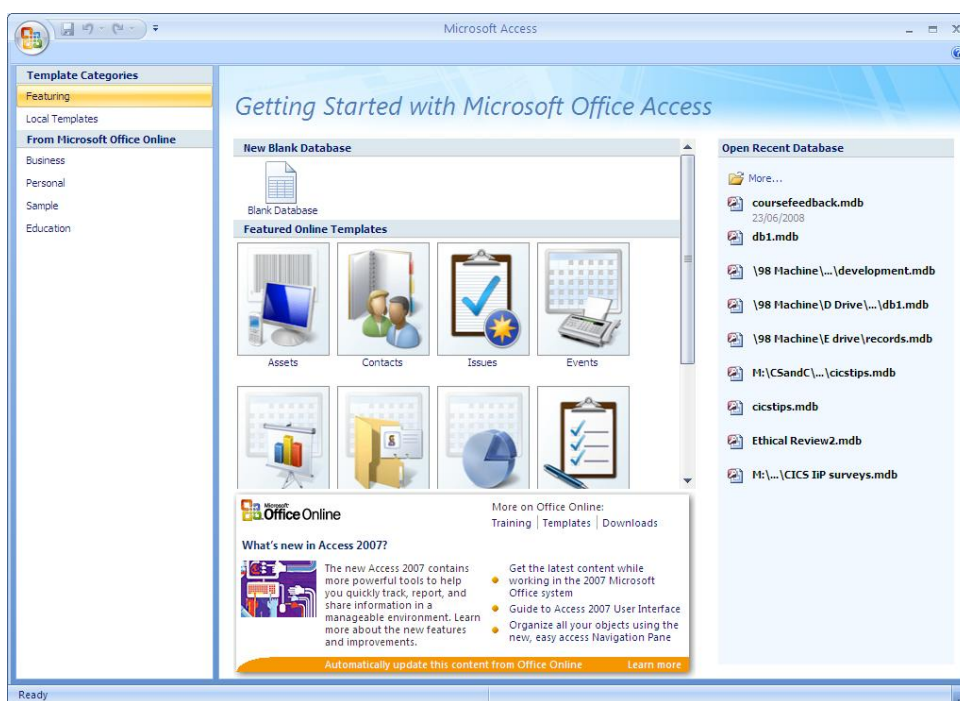
If you give a copy of a new Access file to a colleague who is using an older version, they will not be able to read it initially. However, their Access will detect this new format, and will invite them to download and install a compatibility pack. Once they have done this they will be able to work on Access 2007 files in their older version.

2. Starting Access 2007

Access 2007 is installed on the Managed XP Service. It is available on CD for people to install on personal computers.

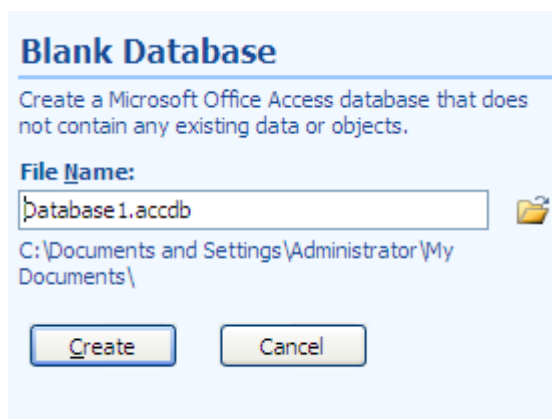
To start Access, click the **Start** button and select **All Programs**, then from the sub-menu, select **Microsoft Office**, then select **Microsoft Office Access 2007**.

When Access starts you will see the Access window, the right-hand pane displays options for selecting files which you have recently worked on, and for creating a new database.



If you will be working on an existing database it will be listed in the right-hand pane. You can click the named database to load it.

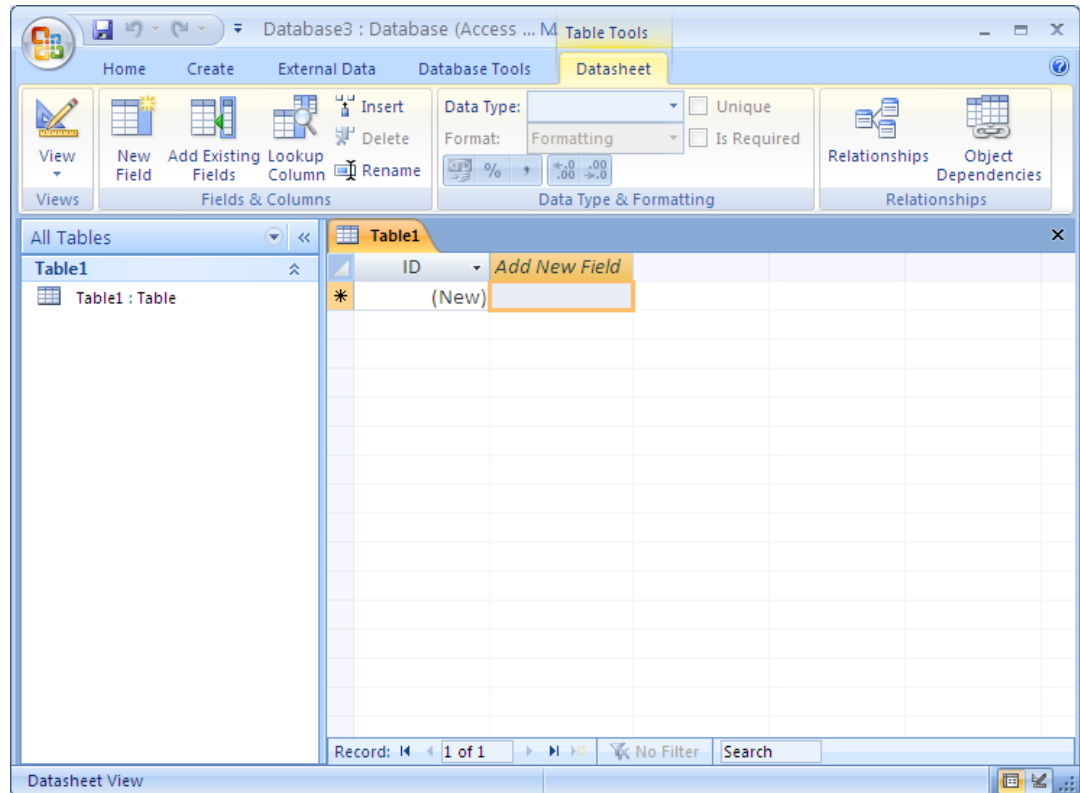
To begin work on a new database, click the **Blank Database** icon. You will need to create a new database file and will see the following appear in the lower-right corner of the screen:



Give your database a meaningful file name. You can use up to 256 characters, use upper and lower case, and include spaces.

By default the database will be created in your **My Documents** folder. You can choose a different folder for your database by using the **Save in** menu.

When you have done this click the **Create** button to create your new database, and to begin creating your first table.



The column on the left gives you access to all of your database components; currently you have one table called Table1. You can hide this column by clicking the left arrows button, or you can change how your database components are listed by clicking the down arrow.

We recommend that you select the **Object Type** options, which will set the column to list all Tables, Queries, Forms and Reports in your database, arranged according to category.

3. The User Interface

The user interface is very different to previous versions of Access. Once you get used to it, it becomes very easy to use. Microsoft has created online training modules that tell you how to find familiar commands in the new interface. You can access these modules from www.shef.ac.uk/cics/office2007 where you will also find a detailed CiCS guide to the new interface.

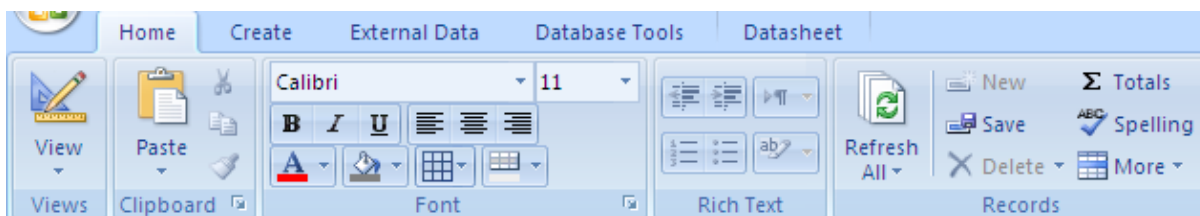
3.1 The Office Button

In the top-left corner of the new interface you will find the Office button, which replaces the **File** menu from previous versions of Access. Click this button to reveal options allowing you to **Save** and **Print** files. It contains a list of your most recently accessed word documents, and it contains the **Access Options** button, which takes you to options previously found in the **Tools** menu under **Preferences**.

3.2 The Quick Access Toolbar

Situated to the right of the Office button, the Quick Access toolbar holds buttons which can be accessed from all views of Access. Whatever you are currently doing, you will be able to access these commands. Initially, the toolbar contains a Save button and Undo and Redo buttons, but you can easily customise the Quick Access Toolbar to hold your favourite commands.


3.3 The Ribbon



The ribbon contains tabs relating to a specific task.

Each tab contains several groups of tools; here the Views, **Clipboard**, **Font**, **Rich Text** and **Records** group are visible.

Each group contains related buttons, menus and lists.

 Some groups have an arrow in the bottom right corner, which gives a dialog box when clicked.

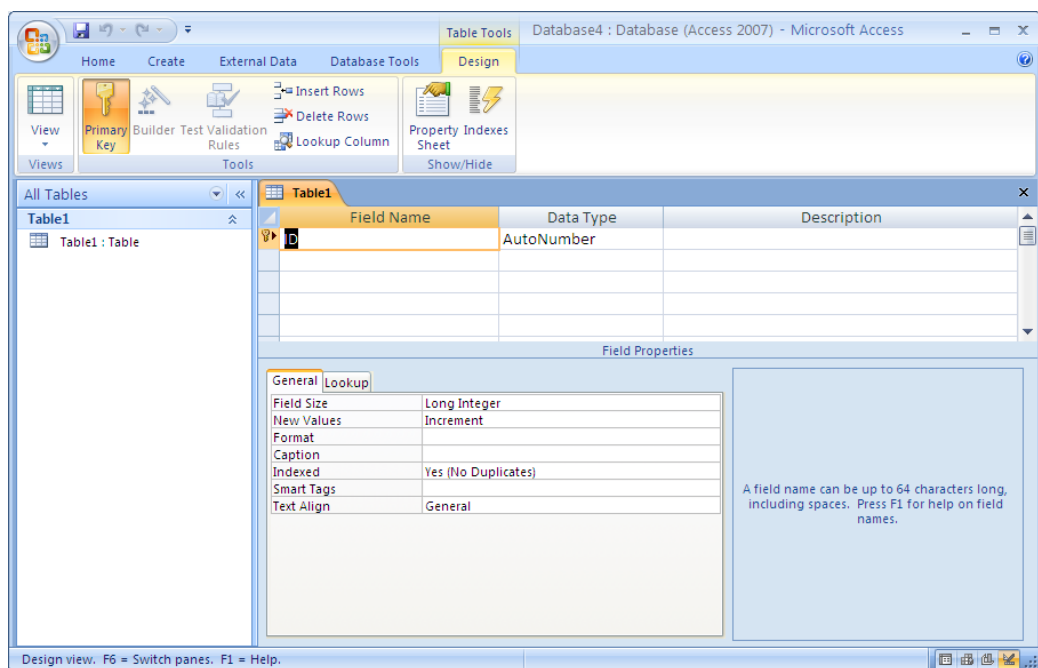
4. Tables of Data

Access holds data in tables.

- Each entry in the database is held as a record
- The individual details of each entry are held in fields.

It is easiest to create a new table in Design view where you first define the fields that you will be using. From the initial screen, click the arrow below the **View** button, found in the **Views** group on the **Home** tab, and from the menu select **Design View**.

A dialog box will require that you name the table, for now leave the default Table1 and click the **OK** button.



In the lower-right area, the grid at the top lets you define the field in the table; the grey area below lets you fine-tune each field. For the time being we will only consider the grid.

First enter a name for the field in the **Field Name** column. The name can be long and include mixed case letters, numbers and spaces. You can overtype the **ID** field that was added automatically.

Press the **Tab** key to move to the **Data Type** column. The majority of fields will hold text data and this will appear by default. If you click to reveal the menu however you will see the following list of data types available to you.

Text	Suitable for small amounts of text such as names addresses and telephone numbers.
Memo	Suitable for large amounts of text such as descriptions or abstracts.
Number	Numeric values (whole or fractional) that can be used in calculations.
Date/Time	Dates and times.
Currency	Monetary values.
AutoNumber	An automatic counter which automatically increments itself for each record.
Yes/No	True-or-false values.
OLE Object	Graphics or other binary information.
Hyperlink	Web page addresses.
Lookup wizard	Creates a list of acceptable values, covered later.

When you have given the first field a name and specified its data type you can then press **Tab** and enter an optional description of that field for your own benefit. Press **Tab** again to move to the next line and define your second field.

When you have defined all the fields in your database you should choose one field to be the Primary Key. The Primary Key is an important feature of Access. Each table should have one field that is unique for each record. A surname, for example is not suitable as a Primary Key as two people may have the same surname, whereas an email address, for example, will be unique and may be used as a Primary Key. When you have identified an appropriate field, select it in the table Design window then click the **Primary Key** button found in the **Tools** group on the **Design** tab of the new **Table Tools** section. A key symbol will appear by that field.

Example Database

In these notes we will construct a database of students registered for fictitious courses in a department. Below is the list of fields used in the first table. Each student has a unique registration number so this has been defined as the Primary Key. As the registration number begins with zero, and will not need to be counted, we have recorded it as a text field.

Table1			
	Field Name	Data Type	Description
🔑	Registration	Text	
	Surname	Text	
	Forenames	Text	
	Title	Text	
	CourseCode	Text	
	StudyYear	Text	
	Email	Text	

4.1 Entering Data

To begin entering data, click the **View** button, found in the **Views** group. If you have not already saved your table you will be required to do so now. When you enter Datasheet View you will see a table with a single blank row. The columns of the table will be labelled with the field names that you have defined. In the example database mentioned above you would see the following screen.

Registrator	Surname	Forenames	Title	CourseCode	StudyYear	Email
*						

To add data to your own table, click in the first field and type the data. Use the right arrow or **Tab** key to move to the next field. When you have entered data for the final field in your first record, press the **Tab** key. The first record will be saved automatically, and a new row will be created for the second record. Continue this process for all records in your data, each record will be saved automatically. In our example database, the first table might look as follows

Registrator	Surname	Forenames	Title	CourseCode	StudyYear	Email
080000001	Booth	Ben	Mr	MCCU01	1st	b.booth
080000002	Grice	Summer	Ms	MCCU02	1st	s.grice
080000003	Coleman	Carol	Ms	MCCU03	1st	c.coleman
080000004	Needham	Rick	Mr	MCCU02	1st	r.needham
080000005	Whalley	Greta	Ms	MCCU01	1st	g.whalley
080000006	Adshead	Miranda	Ms	MCCU02	1st	m.adshead
080000007	Morley	Mike	Mr	MCCU03	1st	m.morley
080000008	McAuley	Jack	Mr	MCCU01	1st	j.mcauley
080000009	Panella	Paulette	Ms	MCCU03	1st	p.panella
080000010	Barkworth	Stephen	Mr	MCCU02	1st	s.barkworth
*						

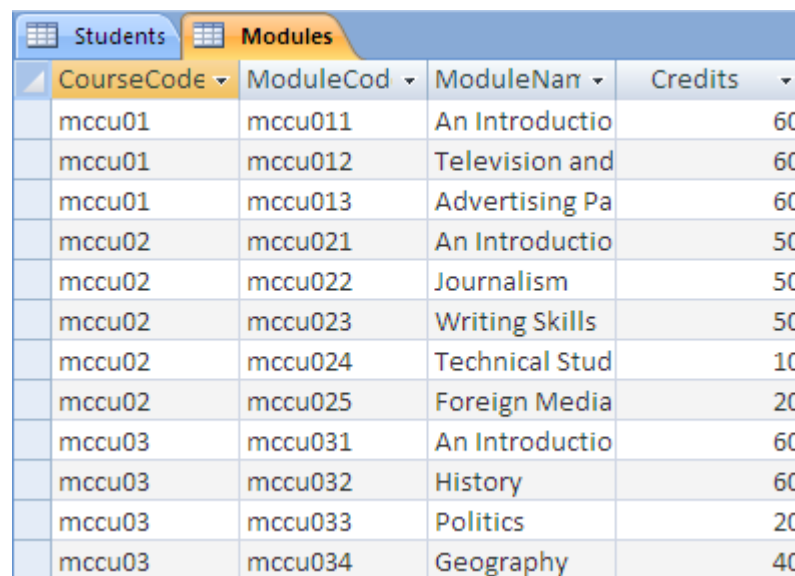
- ✕ When you have completed the first table click the close button in the top-right corner of the window.

4.2 Creating Additional Tables

Although it is often possible to hold all information in a single table, it is preferable to build a series of tables that relate to each other. Each additional table can give detailed information about one of the fields in the first table.

To create additional tables, click the **Create** tab, then from the **Tables** group, click the **Table Design** button. Usually, the second table will duplicate one of the fields from the first table; this is how two tables in a database can be linked together. The second table will also need its own Primary Key.

Below is a second table created in our example database. It contains a list of the modules that are associated with each course. In this table the **ModuleCode** of each module has been defined as the Primary Key.



CourseCode	ModuleCod	ModuleNar	Credits
mccu01	mccu011	An Introductio	60
mccu01	mccu012	Television and	60
mccu01	mccu013	Advertising Pa	60
mccu02	mccu021	An Introductio	50
mccu02	mccu022	Journalism	50
mccu02	mccu023	Writing Skills	50
mccu02	mccu024	Technical Stud	10
mccu02	mccu025	Foreign Media	20
mccu03	mccu031	An Introductio	60
mccu03	mccu032	History	60
mccu03	mccu033	Politics	20
mccu03	mccu034	Geography	40

4.3 Renaming Tables (and Other Database Objects)

If you need to rename a table, for example to give the original Table1 a more meaningful name, make sure that your table is listed in the left-hand column. To switch to the recommended display, click the down arrow in the left column and from the menu select **Object Type**.

Ensure that the table in question is not open in the right-hand pane, then right-click the name in the left column, and from the menu select **Rename**.

You can then type in a new name for the table, then press the **Enter** key to complete the name change.

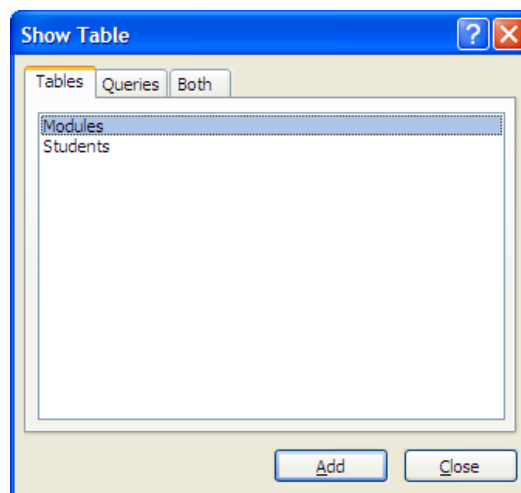
You can also use this technique to rename Queries, Forms and Reports.

5. Querying the Database

Having input a wealth of information into your database you can now begin to extract specific details of interest using queries.

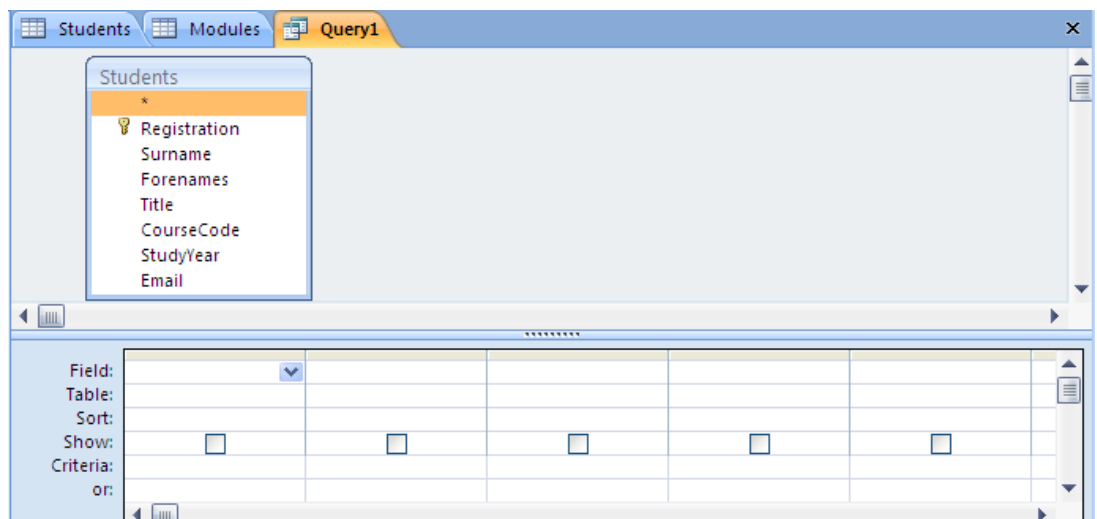
Any existing queries will be available under the **Queries** section of the left-hand column window. If you don't see this, click the down arrow in the left-hand column and select **Object Type** from the menu. To run an existing query, simply double-click it. The query will run and the extracted data will be displayed on screen.

To design a new query click the **Create** tab, then from the **Other** group, click the **Query Design** button. You will be offered a list of tables on which to base your query.



First we will consider a simple query, which is used to extract information from a single table in a database.

Select the table of interest, then click the **Add** button. Click the **Close** button to remove this window. You will then enter the Query Design window containing your chosen table.



Your table will be displayed in the upper pane with all its fields available. In the lower pane is an area to which you can add specific fields from your table. To add a field to the lower pane simply double-click its entry in the upper pane.

For example, in our student database, the first table contains lots of details about the students. If we only need to know their names and email addresses, for a mailing list, we should double-click in turn the entries for **Forename**, **Surname** and **Email**. These fieldnames will appear in the lower pane.

Field:	Forenames	Surname	Email
Table:	Students	Students	Students
Sort:			
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:			
or:			

To view the data in the query we need to switch to Datasheet View. Click the **View** button and the results of the query will be displayed in a table.

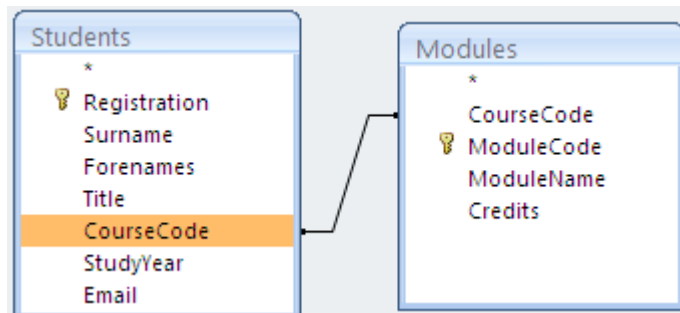
Forenames	Surname	Email
Ben	Booth	b.booth
Summer	Grice	s.grice
Carol	Coleman	c.coleman
Rick	Needham	r.needham
Greta	Whalley	g.whalley
Miranda	Adshead	m.adshead
Mike	Morley	m.morley
Jack	McAuley	j.mcauley
Paulette	Panella	p.panella
Stephen	Barkworth	s.barkworth

Once you have seen the results you can switch back to Design View to enhance your query in any way. If you are happy with your query and wish to save it, return to Design View, click the Save button, and enter a long name as usual. Bear in mind that when you save a query you do not save the data, but save the specifications of the query. Therefore, if the data in the tables changes, then running the query a second time may produce different results.

5.1 Complex Queries

Complex queries are queries that combine data from two or more related tables. To build a complex query create a new query as above, but when you are then offered your list of tables select each table required and click the **Add** button. This will place a number of tables in the query window. When you have added all the tables you need to, click the **Close** button.

In the Query Builder window you will now see all your selected tables displayed. In order for complex queries to work all tables should have at least one link. You can set up your own links by clicking on a field in one table then dragging across to a corresponding field in a different table.



In the students database example above, we can now display the students from the first table and list all the modules they are taking from the second table. Once we have created a link, we can switch to Datasheet View to list the result.

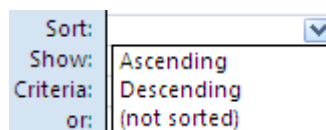
Field:	Forenames	Surname	Email	ModuleName	ModuleCode
Table:	Students	Students	Students	Modules	Modules
Sort:					
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:					
or:					

At any time you can add more tables to a query by clicking the **Show Table** button, available in the **Query Setup** group in the **Design** tab of the **Query Tools** section.

5.2 Fine-Tuning Queries

Sorting the Dynaset

Having obtained the information we require we may choose to sort it into a sensible order. To sort your results, choose the field that you want to be ordered. Then in the row labelled **Sort** click the box. A menu will appear offering you a choice of **Ascending**, **Descending** or **(not sorted)**.



Choose the option as appropriate. When you go to Datasheet View your results will be sorted accordingly.

Selecting Sub-Sets by Criteria

If you don't want to view every record in your database, but want to look at details related to a specific case, you can restrict your results by adding criteria to the query in Design View.

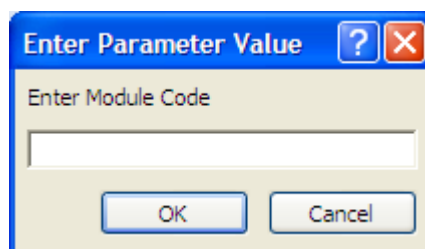
For example, in the student database we may only need to know the names of students taking a particular module (for an attendance list for example). To find these students we can add a criterion stipulating that only students taking a certain module should be listed. We type the criteria into the grid as follows.

Field:	Forenames	Surname	Email	ModuleName	ModuleCode
Table:	Students	Students	Students	Modules	Modules
Sort:					
Show:	✓	✓	✓	✓	✓
Criteria:					= "mccu01"
or:					

To develop this further, we can design the query so that each time it is run it asks you which module you are interested in. This is called a parameter query. Rather than type in text in the **Criteria** row we enter a pair of square brackets []. This will produce a dialog box asking the user for a parameter. To put a meaningful instruction in the dialog box we need to type a message between the square brackets.

Field:	Forenames	Surname	Email	ModuleName	ModuleCode
Table:	Students	Students	Students	Modules	Modules
Sort:					
Show:	✓	✓	✓	✓	✓
Criteria:					[Enter Module Code]
or:					

When the query above is run it will produce the following dialog box.



You could then type in a module code and click **OK**. The query would then list all students taking that module.

To find out about criteria formats click in the **Criteria** row then press the **F1** key. The Office Online Help will start, and offer a list of Help topics. You can click on the topic relating to criteria.

Hiding Superfluous Fields

In the above example the **ModuleCode** is crucial to identifying an exact unit. However, as the query lists the **ModuleTitles** we don't really need to see the code. In this situation you can choose to remove such a field from the query results. In the query builder grid there is a row labelled **Show** containing a ticked box for each field. You can click on any box to remove the tick and that field will not be listed when you run the query.

Field:	Forenames	Surname	Email	ModuleName	ModuleCode
Table:	Students	Students	Students	Modules	Modules
Sort:					
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Criteria:					[Enter Module Code]
or:					

Counting Results

It may be that you do not wish to display results, only to count them. For example you may need to know the number of students taking a certain module.

To do this, click the **Totals** button, in the **Show/Hide** group in the **Design** tab of the **Query Tools** section. An extra row labelled **Total** will be added to the lower pane. In the field you wish to count click in the **Totals** box and select **Count** from the menu. When you then run the query you will see the total rather than the data.

Field:	Forenames	Surname	Email	ModuleName	ModuleCode
Table:	Students	Students	Students	Modules	Modules
Total:	Group By	Group By	Group By	Group By	Group By
Sort:	Group By				
Show:	Sum	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Criteria:	Avg				[Enter Module Code]
or:	Min				
	Max				
	Count				
	StDev				
	Var				
	First				
	Last				
	Expression				
	Where				

In the example above we would select **Count** from the menu. When the query is run, the user will specify a module code and then the surnames of the people attending the module will be counted and displayed next to the module name.

Using Expressions in Queries

Sometimes in queries you do not wish to simply list the data, but rather to manipulate it in some way, this is common with date fields where you may wish to calculate elapsed time.

The first query we created in our example listed students' names and email addresses.

Forenames	Surname	Email
Ben	Booth	b.booth
Summer	Grice	s.grice
Carol	Coleman	c.coleman
Rick	Needham	r.needham
Greta	Whalley	g.whalley
Miranda	Adshead	m.adshead
Mike	Morley	m.morley
Jack	McAuley	j.mcauley
Paulette	Panella	p.panella
Stephen	Barkworth	s.barkworth

It may be that we wish to extract the full address with the @sheffield.ac.uk suffix. To do this we can create an expression that takes the data in the **Email** field and appends the text @sheffield.ac.uk To do this we simply click in the **Email** field and add the required text in quotes

Field:	Forenames	Surname	Email+"@sheffield.ac.uk"

As soon as we move out of that field Access will convert this to an expression

Field:	Forenames	Surname	Expr1: [Email]+"@sheffield.ac.uk"

When we run the query the email addresses will be listed in full.

To find out about expression formats click in the **Field** row then press the **F1** key. The Office Online Help will start, and offer a list of Help topics. You can click on the topic relating to expressions.

5.3 Using Queries to Update Tables

Now that we know how to extract specific details from a database we can use our knowledge to automatically update fields if need be. As a simple example we can imagine in the student database that a year has passed and we need to list the students and record that they are now in the second year. To do this we create an Update Query.

To create an update query we need to create a new query and base it on the table we need to update. We might need to include other tables if we need to set certain criteria. In the query builder we should first build a query with

criteria that lists only those records we want to update. So in the example we would list all students whose registration number started with 08 as follows

Field:	Registration	StudyYear
Table:	Students	Students
Sort:		
Show:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Criteria:	>"080000001" And <"089999999"	
or:		

When we are satisfied with the results we can click the **Update** button in the **Query Type** group in the Design tab of the Query Tools section. This will add a new row labelled **Update to** in the query grid. In the **StudyYear** field we can now type **2nd** to represent second year.

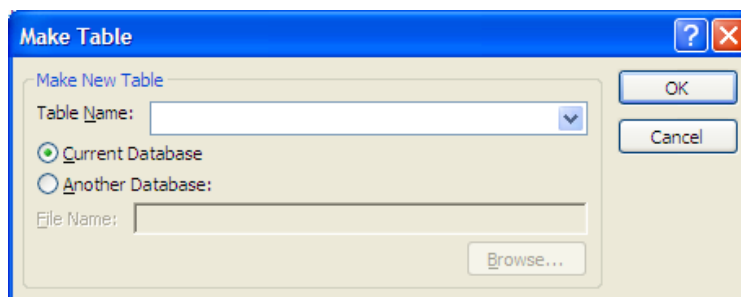
Field:	Registration	StudyYear
Table:	Students	Students
Update To:		"2nd"
Criteria:	>"080000001" And <"089999999"	
or:		

We can now run the query by clicking the **Run** button in the **Results** group in the **Design** tab of the **Query Tools** section. Access will ask you to confirm that you wish to update the records. Click the **Yes** button to update the table.

5.4 Queries that Make New Tables

All of the queries above produce results that only appear when the query is run. If you want to keep the results of a query permanently, you can tell Access to create a new table containing the query results.

Build and test the query as usual, then click the **Make Query** button in the **Query Type** group in the **Design** tab of the **Query Tools** section to get the following dialog box.



Supply a new table name then click **OK**

We can now run the query by clicking the **Run** button in the **Results** group in the **Design** tab of the **Query Tools** section. Access will ask you to confirm that you wish to update the records. Click the **Yes** button and the table will then be created alongside your existing tables.

6. Good Practice in Database Design

Your database should be structured so that the person using it needs to input the minimal amount of information.

6.1 Codes or Names

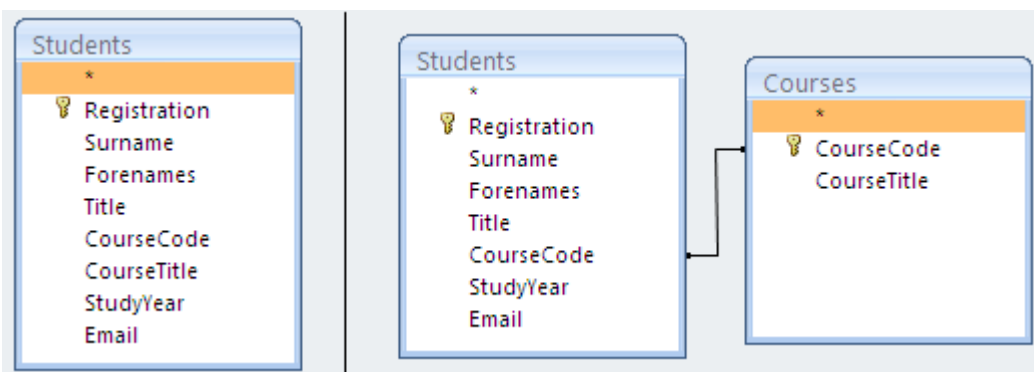
In our example database each student was registered on a course that had a unique programme code set by University administration. Each course, naturally, would also have a unique name. Below we have created a table containing the programme code and name for each fictitious course.

CourseCode	CourseTitle
MCCU01	Communication Studies
MCCU02	Media Studies
MCCU03	Cultural Studies

Clearly, when inputting data or writing query criteria it is much quicker to enter codes than type in full titles, with the correct capitalisation.

6.2 Table Structure

Consider the following alternative table structures for our example database with programme titles



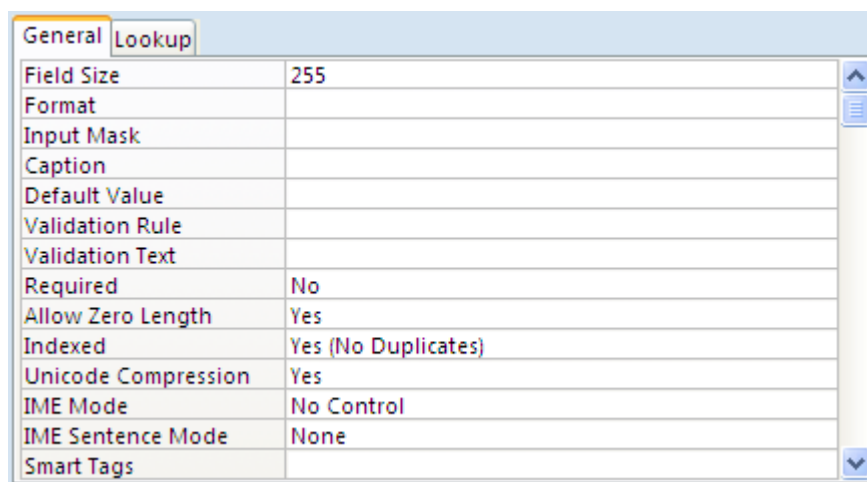
For each student entered into the first structure we would have to input both the programme code and title. However, for each student entered into the second structure we need only supply the programme code and the database will associate the correct title.

Every database should be structured to avoid needless typing.

6.3 Advanced Field Design

When we first designed our tables we selected each field's data type as **Text**, **Number** etc and left it at that. We can greatly improve the database by being more specific about the data in each field.

When you choose any data type in the table design window a collection of properties appears. Below is the collection for the **Text** data type.



There are similar properties for other types of data. The most useful properties are summarised below.

Field Size specifies the number of characters allowed in a **Text** field, so **CourseCode** could be set to 6 and **CourseCode** could be set to "1st".

Default Value can be used to automatically fill in a field for each new record. This value can be changed manually for any exceptions.

Validation Rule can be used to check the accuracy of data. **CourseCode** could be set to

```
= "mccu01" Or = "mccu02" Or = "mccu03"
```

If the user attempts to input anything different an error message will appear.

Validation Text can be used to set a meaningful error message when the validation rule above is contravened.

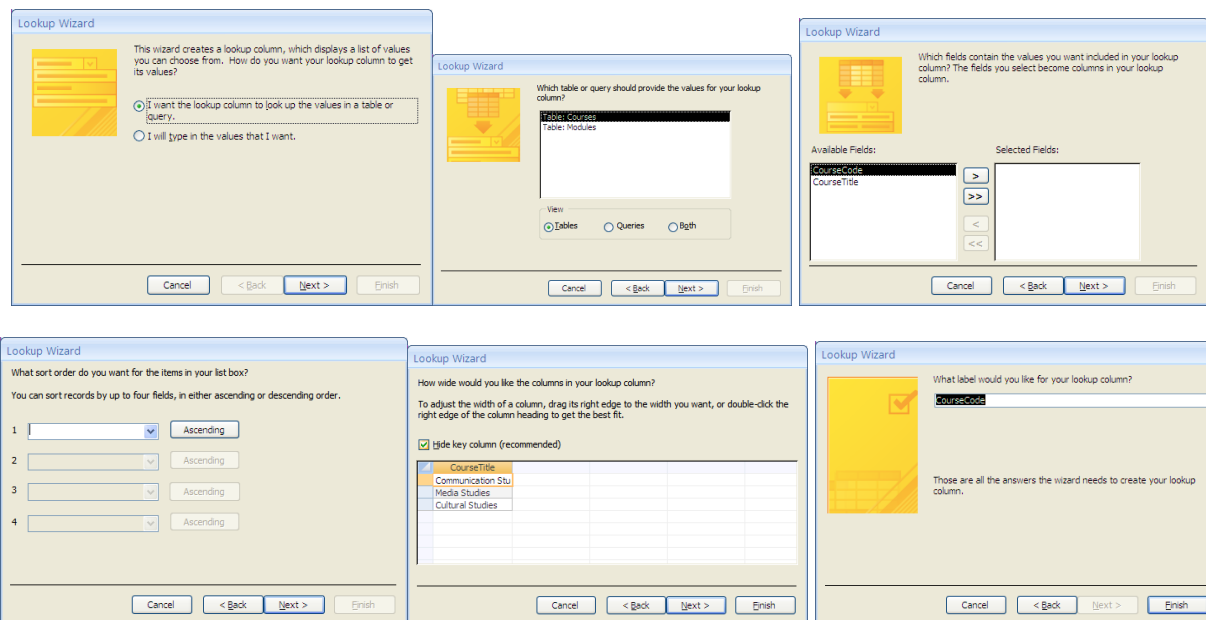
Required can be used to specify that the field **MUST** contain data, and would be appropriate for **Registration**, **CourseCode** and **Year**.

Indexed can be used to specify that the field contents must be unique in that table. This is set automatically for the Primary Key but can be set manually for other fields, such as the **Email** field.

6.4 Lookup Fields

If a field should only contain certain specified values, as in the programme code field, you can build a menu in the table that contains the choice of values. In this way the person inputting data does not need to type each code in but can select it from a list.

To create such a lookup list, open the table in Design View, and set the **Data Type** for the field in question to **Lookup Wizard**. You will then be taken through a series of steps to build the lookup menu.



First we specified that the list values should come from a table rather than be entered manually. Then we selected the table, and the relevant field from that table. We could then set the width for the menu and give it a name.

When the wizard finishes, it asks you to save the layout of the table. You can then switch to Datasheet View, where you will see that clicking in the appropriate field produces a menu of values to choose from.

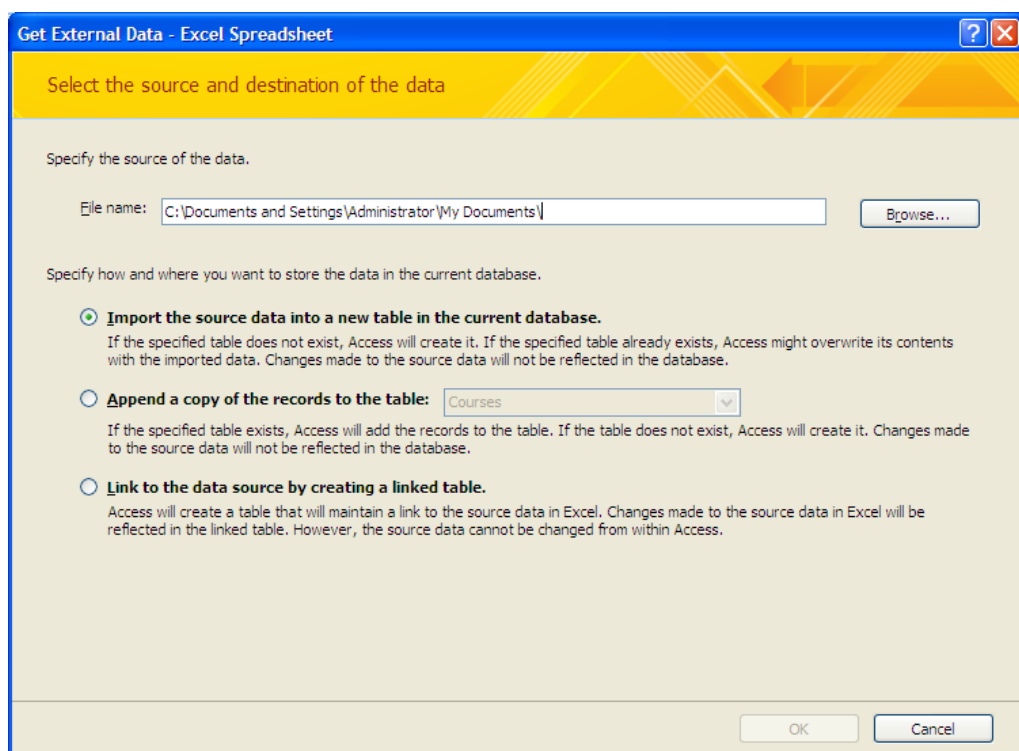
Registration	Surname	Forenames	Title	CourseCode	StudyYear	Email
080000001	Booth	Ben	Mr	MCCU01	1st	b.booth
080000002	Grice	Sharon	Ms	MCCU02	1st	s.grice
080000003	Coleman	Carol	Ms	MCCU03	1st	c.coleman
080000004	Needham	Rick	Mr	MCCU02	1st	r.needham
080000005	Whalley	Greta	Ms	MCCU01	1st	g.whalley
080000006	Adshead	Miranda	Ms	MCCU02	1st	m.adshead
080000007	Morley	Mike	Mr	MCCU03	1st	m.morley
080000008	McAuley	Jack	Mr	MCCU01	1st	j.mcauley
080000009	Panella	Paulette	Ms	MCCU03	1st	p.panella
080000010	Barkworth	Stephen	Mr	MCCU02	1st	s.barkworth
*				<div style="border: 1px solid black; padding: 2px;"> MCCU01 MCCU02 MCCU03 </div>		

7. Importing and Exporting Data

7.1 Importing Tables

Earlier we created a table by defining its fields then typing in the data. It may be, however, that the data is already on a computer, in an existing database, a spreadsheet or a text file. Access includes wizards which can import data from these formats.

To import data click the **External Data** tab, then from the **Import** group, click the button for the file type that contains the import data.



In the dialog box that appears, click the **Browse** button, then in the File Open dialog box select the file you wish to import and click **Open**.

In the lower section of the dialog box, specify whether you wish to create a new table with the imported data, whether you wish to add the data to an existing table, or whether you wish to create a dynamic link to the data source, rather than import it.

Click the **OK** button and the wizard will analyse the data in the file and identify the fields. It may ask you to choose field headings and select a Primary Key. When the wizard finishes, the data will be imported into your database as a new table.

7.2 Linking Tables

If the data that you need already exists, you can import it. This means, however, that a whole dataset is duplicated. If one version of the data changes, the two versions may contain conflicting data.

A technique that may be preferable is linking your database to the data source. By doing this, the data remains in its original location, but it can be accessed and interrogated by your database. This is the technique used to run Access Reports from the central Oracle database.

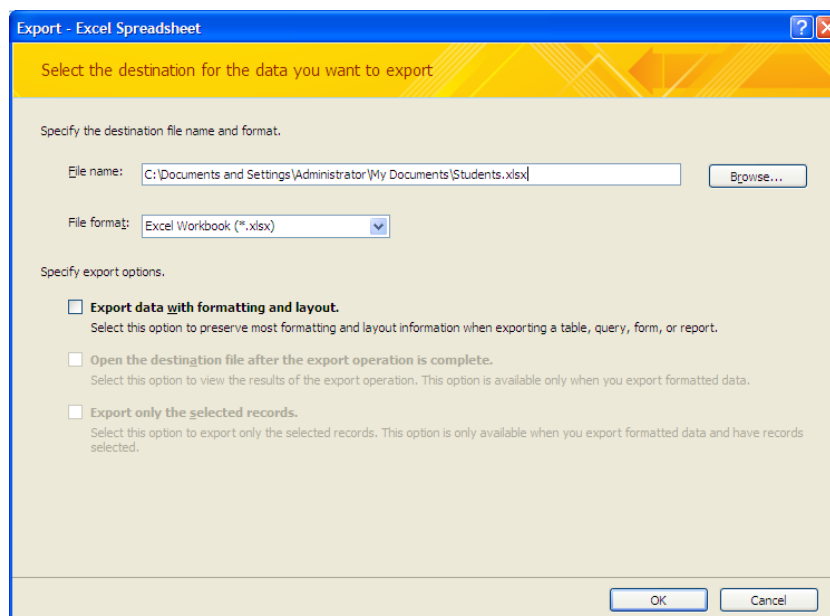
To link your database to an external data source, repeat the process outlined in section 7.1, but in the first dialog box, select the third option: “**Link to the data source by creating a linked table.**” When the wizard finishes, a new table will be listed with your own tables, but it will display an arrow icon indicating it is a linked table.

7.3 Exporting Data

At some stage colleagues may need to access part of your data. Rather than give them a copy of the whole database you can export a single table or even the results of a query. You can then give them the data as a database table, a spreadsheet or a text file.

To export your data first display the table or query results on-screen, then click the **External Data** tab, and from the **Export** group, click the button for the file type that you wish to export to.

In the dialog box that appears, rename the file if you need to, and click the **Browse** button if you wish to specify a different location. Finally click the **OK** button to export the data into its new format.



8. Creating a Data Entry Form

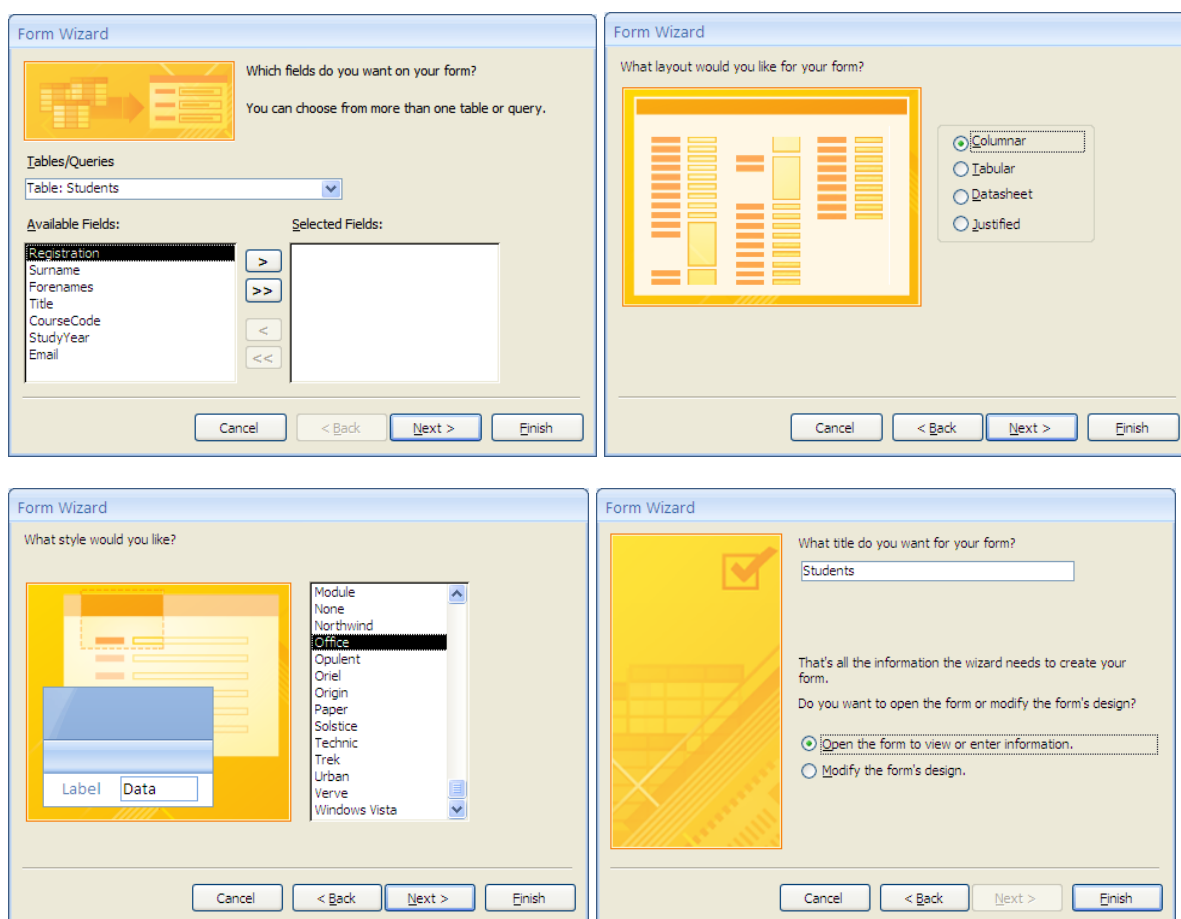
Typing data into a table is fine, but the fields are very close together and the screen becomes awash with data. For a more user-friendly method of entering data you can quickly and easily create a data entry form. To create such a form click the **Create** tab, then in the **Forms** group, click the **More Forms** button, then from the menu select **Form Wizard**.

In the first window, choose the table you want to enter data into then click the **>>** button to include all fields in the form. Click the **Next** button.

In the second window select **Columnar** layout and click the **Next** button.

In the third window select the **Office** style and click the **Next** button.

In the fourth window supply a name for the form and click the **Finish** button.



Your form will be created, and will be ready for data entry.

To add records click the new record button and type away. If you had set up a lookup menu for the CourseCode field, then this would be translated to the form.

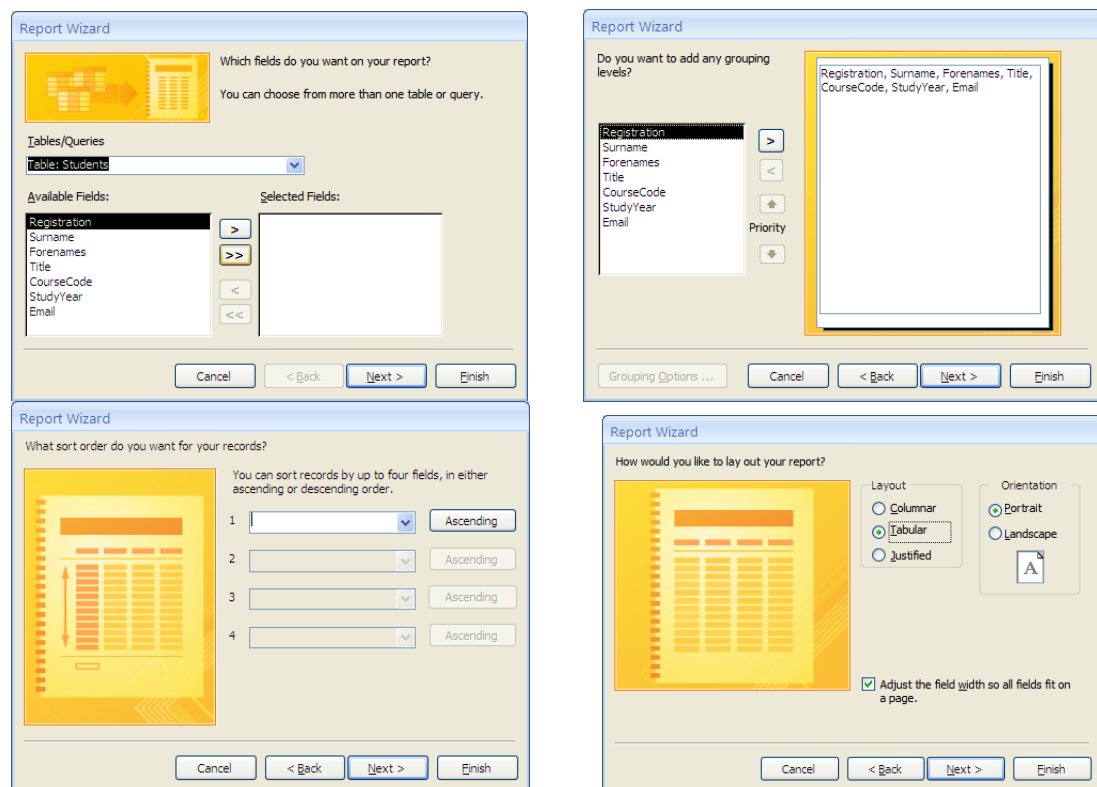
To use the form in future sessions locate it in the **Forms** section of the left-hand column, then double-click it.

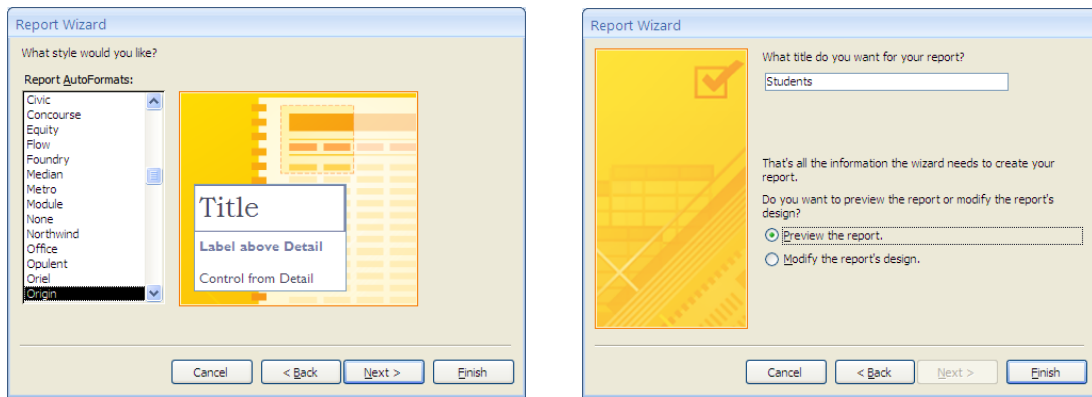
9. Creating Reports

Although you can print off a query to have a paper copy of the information you require, the results will be tabulated simply and cannot be formatted. Instead, you can design a report, which allows the results of queries to be laid out as you require and can include formatted headings and page breaks.

To create a report we can use the wizard as we did for forms. Click the **Create** tab, then in the **Reports** group, click the **Report Wizard** button.

- In the first dialog box choose the table or query that contains the data, then click the **>>** button to include all fields. Click the **Next** button.
- Ignore the groupings options and click the **Next** button.
- Specify a field to sort by and click the **Next** button.
- Choose tabular and click the **Next** button.
- Choose a formatting option and click the **Next** button.
- Specify a name for your report and click the **Finish** button





The tabular report will be generated. Below we have used the query that lists students attending a specified module.

Registration	Surname	Forenames	Title	CourseC	StudyYear	Email
080000001	Booth	Ben	Mr	MCCU01	1st	b.booth
080000002	Grice	Summer	Ms	MCCU02	1st	s.grice
080000003	Coleman	Carol	Ms	MCCU03	1st	c.coleman
080000004	Needham	Rick	Mr	MCCU02	1st	r.needham
080000005	Whalley	Greta	Ms	MCCU01	1st	g.whalley
080000006	Adkins	Miranda	Ms	MCCU02	1st	m.adkins

Use **Page Setup** from the **File** menu to alternate between Portrait and Landscape, and then use the **Print** button to print out your report.

If you wish to add professional design to your report, within the additional **Report Layout Tools** section, click the **Format** tab, then in the **AutoFormat** group, click the **AutoFormat** button. You can then choose a colourful format from the options displayed.

