8.1 Introduction to database management

Manual and computerised databases

Fig 8.1 Data can be collected on index cards and stored in filing cabinets. In this example of a school database there are three 'tables' (files) of students, teachers and grades

Nowadays, computerised databases are in widespread use, as they help people to quickly find the information that they want. They also vary in size and use depending on what is required. Small databases, such as those that keep information about a music collection, can be run on a personal computer at home. Larger databases now play an important role in how our society works. Industrial, commercial and public organisations use databases to maintain their businesses and services.

Other computerised databases include flight information systems and database systems in public libraries.

Examples of how we use these large databases include:

- booking holidays and airline tickets
- using an online store to search a range of millions of products for a particular item
- accessing a police computer database, with requests from police officers who want information about criminal suspects or stolen items.

Although spreadsheets and databases have similar features, there are three main differences:

- Databases are more often used for applications with a large amount of text, whereas spreadsheets can handle complex numerical calculations more easily.
- Very large applications with thousands of entries are more often handled in databases.
- The way they work in the background is different. When you work in a spreadsheet, you view the data you are entering. In a database, you see only the data you are entering at the time – you have to request a report or different display to see more of the information.

A **database management system (DBMS)** is the term for any program that handles the storage, modification and retrieval of data, as well as controlling who has access to the information. Database programs, such as Microsoft Access, Lotus Approach, FileMaker Pro and Corel Paradox, are available on personal computers and allow people to create their own databases at home, school or work.

Table 8.1 *Advantages and disadvantages of a computerised database*

Advantages	Disadvantages
• Can save enormous amounts of paper as well as filing space	• The computer(s) and peripherals required can be very expensive
• Data can easily be entered by keyboard or scanners	• If the computer, or computer network, is not working, then the database cannot be used
• Speed – data can be found, calculated and sorted very quickly	• Security is very important as some people may attempt to get access to confidential information, sometimes by illegally hacking into the program or data
• Data can easily be changed and updated	• The database file can become corrupted or infected by a computer virus. This can lead to the file not working properly. In some cases, the database may not work at all. Making a backup copy of the database is therefore essential
• Data needs to be entered only once, yet can be presented in many different ways. A whole range of different queries and reports can be produced	• There is often a limit to the size of a database file
• Data can be checked on entry	• Some databases can be complicated to use
• Passwords can be set to allow access only to those with permission to use the database	• Some databases require much time to be spent on staff training, which can be costly
• The data structure of a database can be changed, with new fields added, even after the database has been created. A paper-based system would have to be restarted from scratch	• Data may be stored incorrectly
• Data can be imported and exported to other programs	
• A database file can be automatically linked to others	
• Databases can be shared with other users if the computer is part of a local or wide area network. This includes the Internet	

Questions

1 Give an example of a 'paper database'.

2 Give two examples of large databases that might be used to help members of the public.

3 Annissa is opening a small cake business. Explain whether a word processor, database or a spreadsheet would be suitable for each of the following scenarios:
   a She first needs to store the names and contact information of her customers, types of cakes ordered and the occasion.
   b She then wants to store the amount of money she receives weekly from cake sales so that every three months she can analyse her profits using a graphic.
   c Annissa wants to keep track of the quantities of items used to ensure she has sufficient ingredients available in storage.

Let us first examine what makes up a database.

- The main purpose of a **database** is to store data. It stores this data in a number of related files, more commonly called tables.
- A table consists of a collection of **records**.
- A **record** consists of a number of **fields**.
- A **field** is the smallest piece of data that can be stored.

Fig 8.2 Database components

Figure 8.2 shows a database about a bookshop. There may be other tables, but three are shown in the database relating to the bookshop. One of the tables is called Order. Each table is made up of rows of records. Each table has columns with field names that describe the type of information that is stored in each field. In Figure 8.3, the Order table has three columns with field names CID, PR-ID and QTY, and seven rows of records. Each record therefore has three fields of data. The record in the fourth row of the Order table suggests that the customer with ID 1138 (Ed Yod) ordered 100 units of the product with ID P3745 (Receipt Book).

The power of a database is that information contained in fields can be searched, grouped, sorted and/or exported – often in a matter of seconds. In Microsoft Access, you create a database by giving it a suitable name before you create the tables to enter the data. Table 8.2 illustrates the various components of a database.

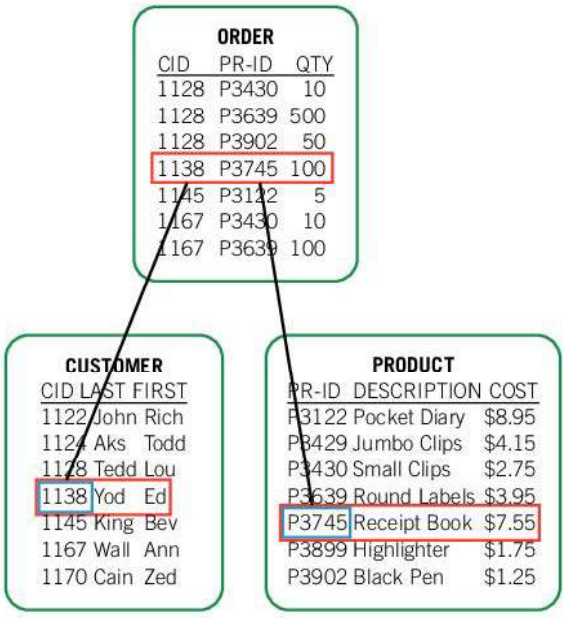


Fig 8.3 The data in the Order table links to data in other tables

Table 8.2 *Database terms and definitions*

Object	Components that make up a database (i.e. tables, forms, queries, reports, macros and modules)
Database	A database can consist of multiple tables
Table	A collection of records about a specific topic, such as students or vehicles
Query	Asks specific questions about the data in the database
Form	A graphical user interface designed specifically for entering, displaying, and searching data. This is an alternative to entering data in the spreadsheet-like view
Report	Summarises and formats data from either table or query data

Creating a table

Before you create the database and its tables, you must first choose an appropriate name for the database that describes its purpose. Then, provide suitable names for the tables based on the type of records that will be stored in them.

For each table in the database, you should consider the following components:

- field name, which identifies the data stored in a field
- field type, also called data type, which tells the database program what kind of data goes in the field, such as text, numbers or dates
- field length, which determines the size of each field in the table
- field description to allow you to describe the purpose of the field
- field properties, which include checks that the data is valid.

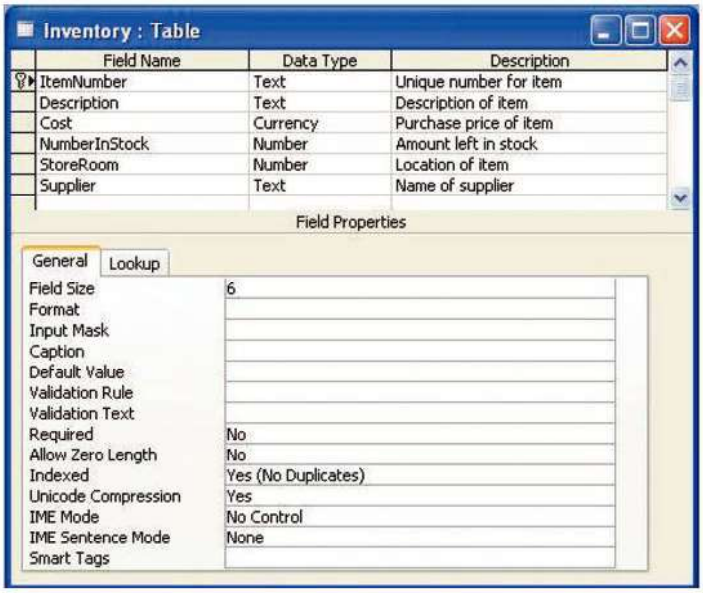


Fig 8.4 When creating a table, it is important to have the correct field name, type, length and description

Field name

The name of each field which identifies the data should be meaningful. For example, you should not have a field labelled 'Name', since you would not know if it is referring to a first name or a last name of someone, a product or a country. The Order table (Fig 8.3) contains three field names – CID, PR-ID and QTY. In some database programs a dash or an underscore is used if more than one word makes up a field name, such as First_Name. In others they can be written together, as FirstName.

Data types

The data type determines what kind of data can be entered as well as what operations the database can perform with the data. Again, using the Bookshop example in Figure 8.3, the field CID would be of type Text (also known as character or alphanumeric), but QTY would be a Number. It is therefore important to work out the data types for each field. The most common data types are shown in Table 8.3.

Table 8.3 *Common data types found in tables*

Use this type ...	*When the field's data is ...*
Text (Memo has been renamed as Long Text)	Long or short text – letters, numbers and special characters
Number (also known as a numeric field)	Numbers, for example 12345
Date/Time	Date – day, month and year information (for example, 26/08/1992) including time information (for example, 9:32)
Currency	Dollar-and-Amounts of money – $ (Dollar) or £ (Pound) or € (Euro)
Autonumber	A number that increases automatically as each record is added
Yes/No (also known as a Boolean or logical field)	Only one of two values (for example, a checkbox to tick, Yes/No, True/False, On/Off)
OLE Object	Picture, video clip, sound file or object from another program (Windows only)
Lookup Wizard	A drop-down box that offers you a limited choice of options for a data entry
Hyperlink	Web address that links to a web page
Attachment	An image, spreadsheet file, document, chart or other type of supported file that can be attached to a record; similar to attaching a file to an email message

Field length

Another important field attribute is field length. Not all database tables require you to set a maximum field length, but many do. This is because, if you decide in advance the maximum length of a field, the file size can be kept as small as possible and there is no wasted storage space. Also, the time taken to process data is kept to a minimum.

Think what you would have to do when setting the field length for a surname field. A field length of four characters would be right for those whose surname contains four characters, such as 'Glen', but not suitable for 'Ronald' (six characters). Some tables allow fixed field lengths – for example, 30 characters. The field stays 30 characters long even when the name Smith (5 characters) is entered.

Field description

You can enter an optional description for each field in the Description column in the design grid. The description appears on the status bar when the field is accessed on a form. Examples of a description include entering brief comments about the purpose of the field or the data that should be stored in it.

Field properties

Each field has a set of properties that control the way it stores, handles and displays data. You normally set properties when you create a table in Design view  . However, you can display the Design view at any time to set or change any property settings.

The properties available in the Field Properties pane of the Design view window (Fig 8.4) depend on the data type selected in the design grid. Some of the property types are listed in Table 8.4.

Primary key

Usually one particular field of each record contains an item which is used to identify the record. This field is called the **primary key field**  . This value in the key field must uniquely identify each record. Using the Bookshop example, the CID field in the Customer table can be used to uniquely identify each customer. You would not use First or Last field names, as more than one person may have those names.

The need for a key field becomes clear when you think of what might happen if you wanted to search a database for John Smith. Smith is a common surname and you may find that there are lots of Smiths in your database – and more than one John! It is better, therefore, to have a unique reference such as an identification number (key field) on which to search.

Table 8.4 *Property types for fields in a database table*

Property type	*Description*
Field Size	Limits a Text field to a specific number of characters or a Number field to the range of numbers it can store
Format	Controls the way data appears in Datasheet view
Decimal Places	Displays a set number of decimal places in Number and Currency fields only
Input Mask	Sets a pattern that determines the input format of data, such as the hyphens in a telephone number, for example 224-5860
Caption	Specifies a label other than the field name that appears in the table and on forms and reports
Default Value	Displays a specified value for a field in new records
Validation Rule	Limits the data entered to meet a certain requirement. For example, you can specify that the CustomerID field cannot be less than 1000
Validation Text	Specifies the text you want to appear in an error message if the data entered violates the validation rule. For example, the error message 'Customer IDs start at 1000' will pop up if you enter, say, 900 in the CustomerID field
Required	Specifies that the field cannot be left empty when you enter data into a record
Allow Zero Length	Determines whether or not you can enter quotation marks (" ") in a field to indicate that there is no data for that field in the record
Indexed	Speeds up retrieval of data in a field. All primary key fields are automatically indexed

Other unique keys

A **candidate key** can also be a primary key if it is unique. However, only one field must be chosen as the primary key. Candidate keys are entirely optional, so a table may contain none, one or several of them.

A **composite key** is a primary key that made up of two or more fields. In the Bookshop database, CID cannot be a primary key in the Order table as there are multiple occurrences of 1128 and 1167 in that column. Also, PR-ID cannot be a primary key in the Order table either because P3430 and P3639 are also listed more than once. Therefore, Figure 8.5 shows a composite key using two fields CID and PR-ID. This is done so that a customer can order more than one product and more than one customer can order the same product.

Order

	Field Name	Data Type	
🔑	CID	Short Text	Customer ID
🔑	PR-ID	Short Text	Product ID from inventory
	QTY	Number	Quantity ordered

Fig 8.5 A composite key is a primary key that can comprise multiple fields

In a database where the same field is found in two or more tables, if it is a primary key in one table, then it is a **foreign key** in the other tables.

Questions

1 Put the following terms in order from largest to smallest: database, field, record, table.

2 Explain what descriptions can be used for the following and whether they are suitable field names: last, first, PR_ID, CID.

3 State whether use of the following field names represent use of a primary key, candidate key, composite key or foreign key:
- **a** student ID printed on a student's school report
- **b** student ID printed on a student's exam card
- **c** product code used to identify the name of product
- **d** subject code printed next to each subject on a student's exam card
- **e** school code on a list of the subjects offered at the school
- **f** passport ID to collect student's exam card
- **g** driver's licence ID to collect student's school report
- **h** student's ID and subject code requested to receive student's grade for that subject.

4 Explain why a primary key (such as StudentID) that comprises all numbers would be given a data type of text and not number.

5 Identify a suitable data type for the following fields:
- **a** subject ID
- **b** mobile number
- **c** name of school
- **d** days absent from school
- **e** lives with mother?
- **f** centre code
- **g** day of exam
- **h** vehicle registration number
- **I** receipt number
- **J** quantity ordered.

6 It is important to plan the structure of a database carefully before you start creating one. The following table is to contain information on members of your family. 'Relation' represents what relation this member is to you, such as sister, uncle or self (meaning yourself!). 'Adult' states whether this member is an adult or a child.
- **a** Choose the most suitable field type for each field (see Table 8.3, which shows different field types).
- **b** Choose the maximum field length for each field.
- **c** Explain whether a field in the list can be used as a primary key. If not, explain what field can be created as the primary key field.

Field	Field type	Field size	Key field?
LastName			
FirstName			
Relation			
Month of Birth			
Male?			
Adult			

Practical exercises using Microsoft Access

Exercise 1: Creating a database

1 Start the Access program by double-clicking on the Microsoft Access icon. These icons may look different depending on your version of Access.

2 Access requires you to provide a name for the new database, or you can browse to open an existing database. Select the option to create a new database and type the name Bookshop. Note that this database is saved in your Documents folder if you do not specify another location.

Fig 8.6 Creating a database called Bookshop

Exercise 2: Creating a table

1 Now that you have created the Bookshop database, it is time to create a database file called a table. Depending on the database program you are using, an empty table, labelled as Table1, may be created for you. If so, then right-click on its label to close and delete it.

2 Microsoft Access provides options to create a table, query, form or report. The Bookshop database will require three tables, called Customer, Product and Order. Each table is created in Design view.

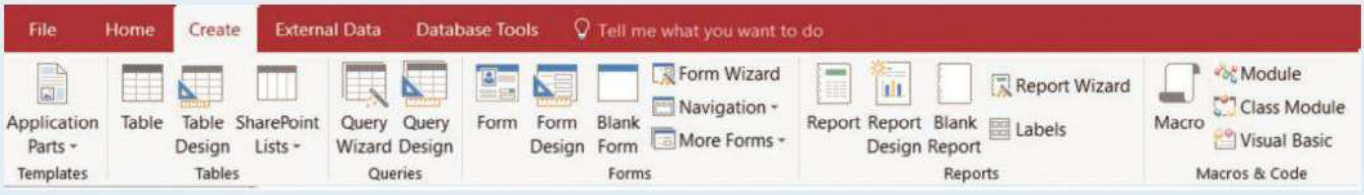


Fig 8.7 Microsoft Access provides options to create a table, query, form or report

3 To create the first table (called Customer), locate the icons or labels that will allow you to Create a Table and view its design. For now, the new table is called Table1.

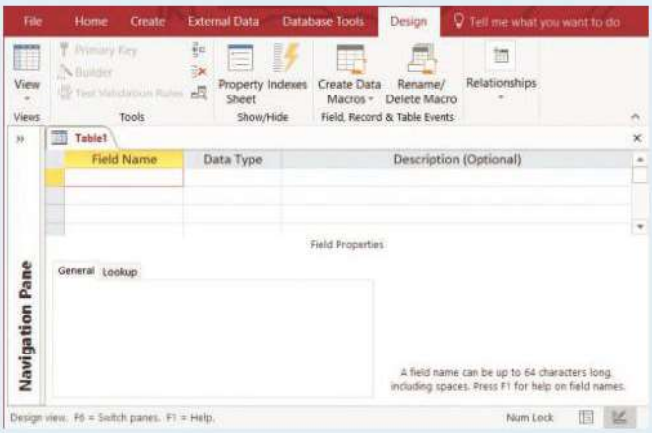


Fig 8.8 Start with a blank table in Design view

4 Click in the first Field Name box and type 'CID', then press *Enter*. In the Data Type column, choose the Text type, press *Enter* and then type the description as 'Customer ID'. Click below to the Field Properties area to type 4 for the field size. Enter the following field names, types and descriptions in your table.

Field name	Field type	Description
CID	Text	Customer ID Number
Last	Text	Customer's Last Name
First	Text	Customer's First Name

5 Note that the CID field is Text, since calculations are not performed on this field, otherwise the Number data type would be chosen. Click again on the CID field name and enter the following field properties:
Field Size: 4
Validation Rule: >999
Validation Text: Customer IDs start at 1000

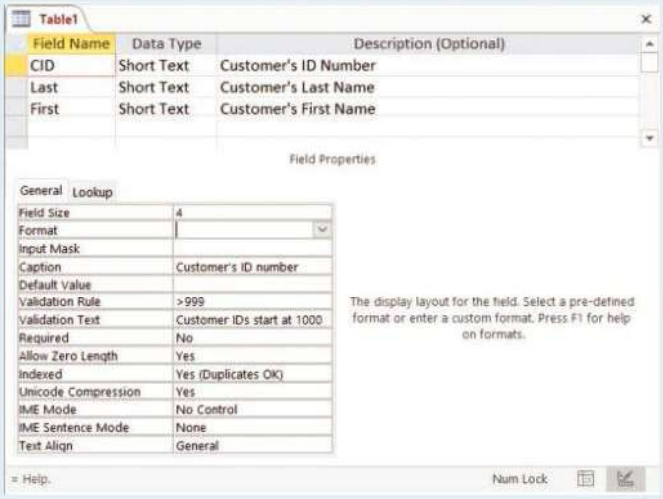


Fig 8.9 Entering the design of the Customer table

6 At this point, you should choose one field whose value uniquely identifies each record in a table. If you do not define a primary key, Microsoft Access asks you if you would like to create one when you save the table. For this exercise, make the CID field the primary key, meaning that every item has a four-digit number and no two are the same. To do this, select the CID field and select the Primary Key icon . Then click the Save icon or *Ctrl* + *S* to save the table. Name the table as Customer and click OK to close the dialogue box. You have created your Customer table.

7 Use the same Bookshop database. Locate the icons or labels that will allow you to Create a Table and view its design.

8 Enter the following field names, types and descriptions in this table.

Field name	Field type	Description
PR-ID	Text	Product ID
Description	Text	Description of product
Cost	Currency	Unit cost of the product

9 Make PR-ID a primary key and save the table as Product.

10 Use the same Bookshop database to create another blank table in Design view. This will be used to create the Order table.

11 Enter the following field names, types and descriptions in this table.

Field name	Field type	Description
CID	Text	Customer ID
PR-ID	Text	Product ID from inventory
QTY	number	Quantity ordered
DISCOUNT	Yes/No	10% discount offered on Product

12 Make CID and PR-ID the composite primary key. Select the CID row, then while pressing the *Shift* key, select the PR-ID row. Then click the Primary Key icon in the Menu bar. Both fields will have the Primary Key icon next to them.

13 Save the table as Order.

Tables should be joined, so that you can access and coordinate information in all the fields of the connected tables. Joining tables saves you having to enter the same information in several tables. In addition, it allows you to create reports, forms and queries from the related data tables in the database file. This means you can create smaller, more efficient tables that can be related when you need access to the data.

Relationships

The linking of two tables can occur in one of two ways:

- one-to-one
- one-to-many.

One-to-one (1:1)

This type of linking usually takes place when the primary key in one table matches the primary key in the second table. An example of *one-to-one* linking is that each employee has a personnel record, or a personnel record is created on each employee. In Figure 8.10 there is one record in the Employee table that links with one other record in the PersonalRecord table.

One-to-many (1:M)

This occurs when one primary key in one table links with a foreign key or a combined key in another table. That is, a record in one table matches many records from the other table. For example, one department has many employees, or many employees are assigned to one department. This is shown in Figure 8.10, where one record in the Department table can have many matching records in the Employee table.

In applications such as Microsoft Access, the linking of tables is called a relationship. A one-to-one relationship is shown in Figure 8.10 by a line connecting the two entities with 1 denoting the 'one' ends of the relationship. A one-to-many relationship is shown with M or an infinity sign (∞) denoting the 'many' end of the relationship. Although there are other conventions in many texts, you should use one that you understand best.

Using the Bookshop database example with the Customer, Product and Order tables, how do you identify who has ordered which product? This is achieved by linking pairs of tables that have a field common to both tables. There is related data in these tables, since a customer can place one or more orders for a product. Figure 8.11 shows three tables again with their fields. The primary key field is underlined in each table.

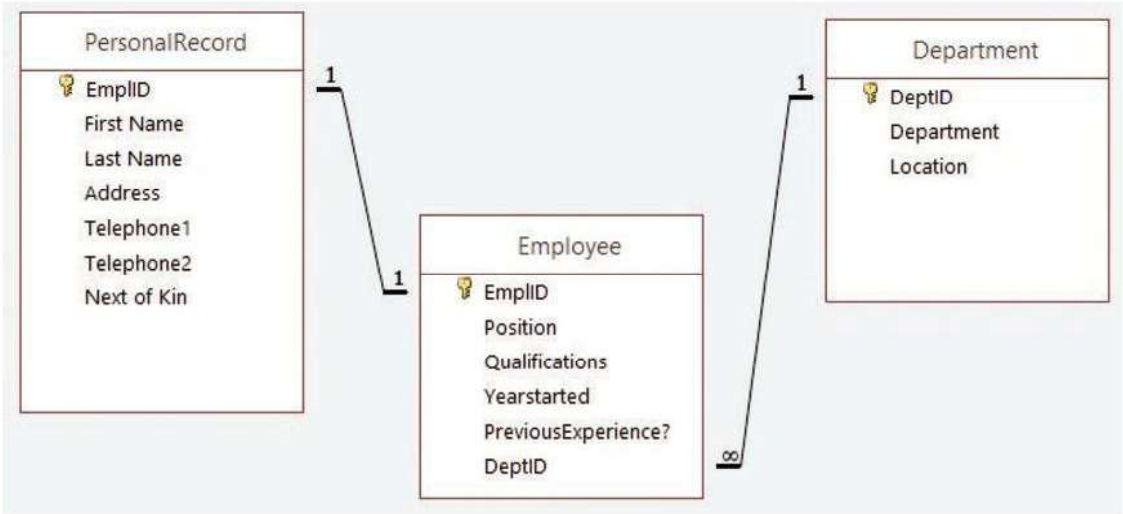


Fig 8.10 Linking common fields between pairs of tables

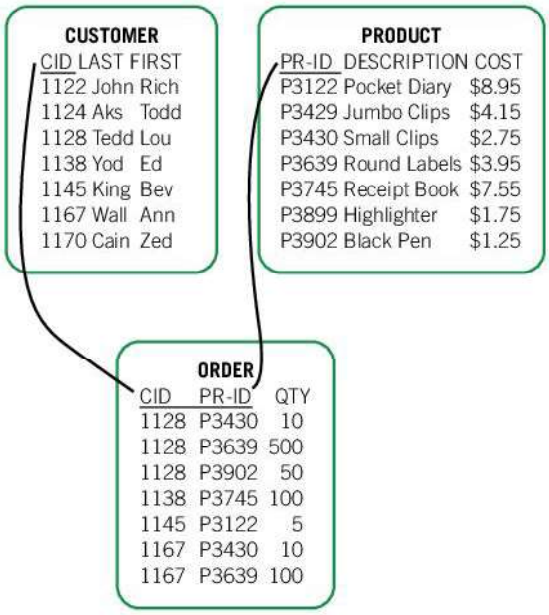


Fig 8.11 Creating a relationship between pairs of tables for the Bookshop database

Linking tables ensures that the data in the database remains as accurate as possible. For example, if you wish to delete a customer who has placed orders, the database will not allow you to do so until the orders for that customer are deleted. That way, no order can be placed without having a customer's data linked to it.

Questions

1 Use Figure 8.11 to answer the following questions:

- **a** State whether pairs of tables have a one-to-one relationship or a one-to-many relationship.
- **b** State the primary key in each table.
- **c** State the first names of the customers who ordered round labels.
- **d** State the description of the product that was ordered by the customer with ID 1145.
- **e** How many customers placed orders?

2 Answer the questions based on these two tables in a Sports database:

ATHLETE			
AthleteID	**NameOfAthlete**	**Code**	**Gender**
121	Jade Boyce	U13	F
231	Shade Skeete	U13	F
351	Neil Hall	U20	M
142	Figman McJig	O20	M
187	Eli Jarad	U20	M

DIVISION	
Code	**Category**
U13	Under 13
U20	Under 20
O20	Seniors

- **a** State the names of the tables.
- **b** How many records are in the Athlete table?
- **c** How many fields are in the Division table?
- **d** In each table, state the most appropriate field that can be used as a primary key.
- **e** State the name of the table and the field that is a foreign key.
- **f** Identify the field that is used to link the tables.
- **g** Explain whether the tables are linked as one-to-one or one-to-many.
- **h** Explain whether the row for the Under 13 division can be deleted from the Division table.
- **i** Explain whether the row of data for athlete 142 can be deleted from the Athlete table.
- **j** Write the name(s) of the seniors.
- **k** What division is Athlete 231 in?

Practical exercises using Microsoft Access

Exercise 3: Linking tables in the Bookshop database

1 Open the Bookshop database that was created in exercises 1 and 2. Locate the Database toolbar that contains the Relationships icon 

.

2 Click on the Relationships icon, then click the Show Table icon 

on the toolbar to make it appear.

3 Double-click on the Customer, Product and Order tables. When you have finished adding the tables, click Close.

4 Now click on the CID field in the Customer table and drag it to the CID field in the Order table.

5 An Edit Relationships window pops up (Fig 8.12). Make sure the Enforce Referential Integrity option is ticked in the checkbox and press OK. This means that Access will help you enforce rules so that your data is valid to start with and remains valid throughout its use.

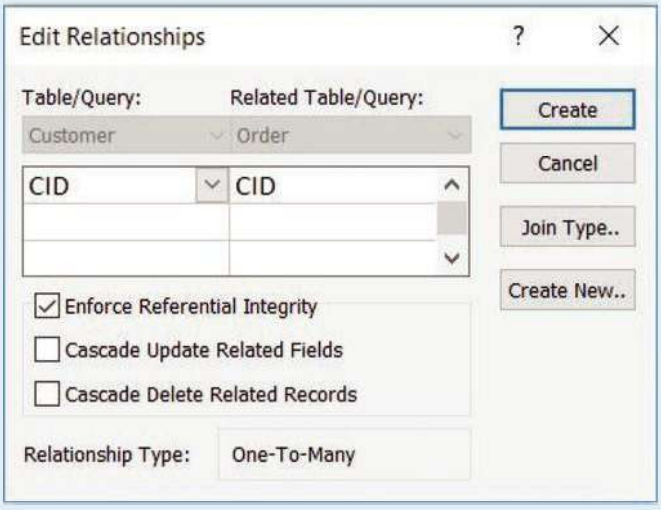


Fig 8.12 Enforcing rules on your data

6 You have created a relationship between the Customer and Order tables.

7 Repeat the procedure in steps 4 to 6 to create a relationship between P-ID in the Product table and P-ID in the Order table.

Exercise 4: Linking tables in other databases

1 Return to the previous example on the Sports database.

2 Create the database called Sports, and then create the two tables.

3 Link the tables using a common primary key, and enter the data in each table.

You can start entering data once you have completed the table design. Design view allows you to enter field names, data types and descriptions into your database table. Another view, Datasheet view allows you to enter raw data into each field of your database table. If you have selected a field as a primary key, make sure that there is always data in this field. To switch views between the Datasheet and the Design view, simply click the button in the top-left hand corner of the Access program window.

A form is another way to enter data (Fig 8.13). It is a graphical representation of a table where you can also add, update and delete records in your table as with the datasheet. You can give the form a different name from your table, but they both still work on the same information and the same data. This means if you change a record in a form, it will also be changed in the table.

A form is very good to use when you have many fields in a table. It means you can see all the fields in one screen. If you were in the Datasheet view, you would have to keep scrolling to see a field at the far left or right.

Sometimes you may need to see one record and the related data on that record in one form. For example, you may wish to see all orders for one customer. It is useful to create a form of the customer data with a sub-form (Fig 8.14) showing the products ordered along with their costs, quantities ordered and whether they are on discount.

Field options

Data entry should be as simple and quick as possible. This is particularly important if a database has hundreds or thousands of records. Data also needs to be accurate. To help with this, you can set the entry options for fields. One entry option is to get data to be automatically entered into a field. For example, the Quantity field could automatically enter the number 0 for each new record entered.

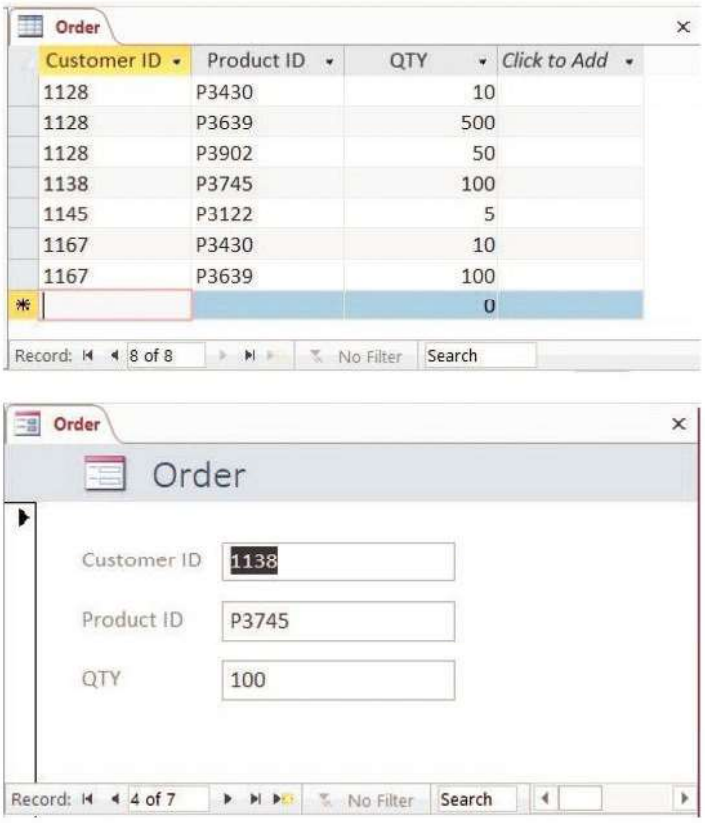


Fig 8.13 You can enter data using a datasheet (top) or a form (bottom)

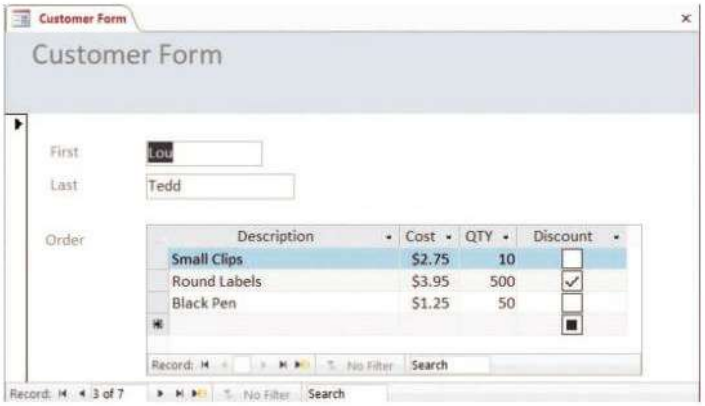


Fig 8.14 A form showing the customer's name along with a sub-form of the products ordered, their costs, quantities and whether they are on discount

In most databases, you will also need to enter the same values into a field repeatedly. For example, it is much quicker and more accurate to choose a product from a list, rather than having to type it in each time. A value list (or combo box) lets you do this by choosing a value from a list (Fig 8.15).

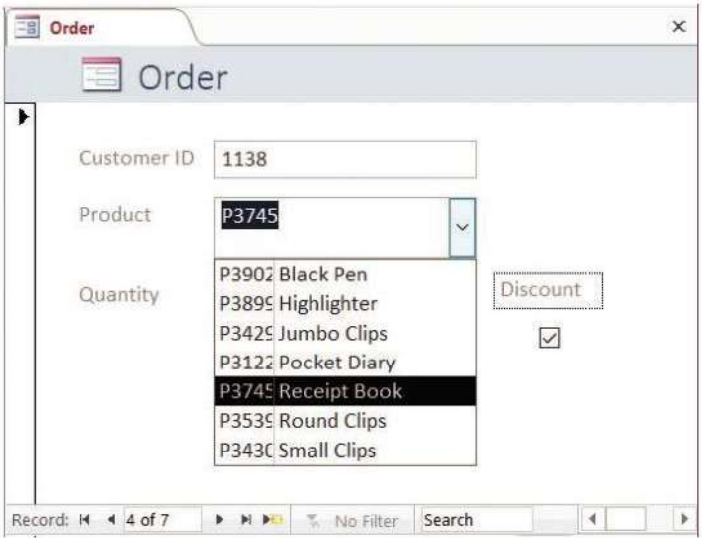


Fig 8.15 Value lists are a quick and accurate way to enter data. Here a list is shown where only one product can be selected. A check box determines whether a product is eligible for a 10% discount

Other entry options include a check box (Fig 8.15). Instead of typing Yes or No in a field, you can have a check box where a tick means 'Yes' and the absence of a tick means 'No'. Another option is to have radio buttons, also known as option buttons. These methods of data entry reduce the possibility of error when inputting data.

Questions

1. Explain the difference between Datasheet view and Design view when creating a table.
2. State the name of the graphical representation of a table that allows you to quickly add records.
3. State two data entry options that can be used to quickly and accurately enter data in a database.

Practical exercises using Microsoft Access

Exercise 5: Entering data

1. Open the Bookshop database that was created in Exercise 1.
2. Double-click on the Customer table to open it.
3. To switch views between the datasheet and the Design view, simply click the button in the top-left hand corner of the Access program window.
   - Datasheet view : allows you to enter raw data into your database table
   - Design view : allows you to enter fields, data types and descriptions into your database table.
4. Click on Datasheet view and enter the data shown below into the Customer table and save it.

CID	Last	First
1122	John	Rich
1124	Aks	Todd
1128	Tedd	Lou

   - To add a new row, press *Enter* or select the next line and enter the information.
   - To modify a record if you have made an error, select the record and field you want to update, and replace it with the text you want.
   - To delete a record, right-click on the row and select Delete Record (Fig 8.16).

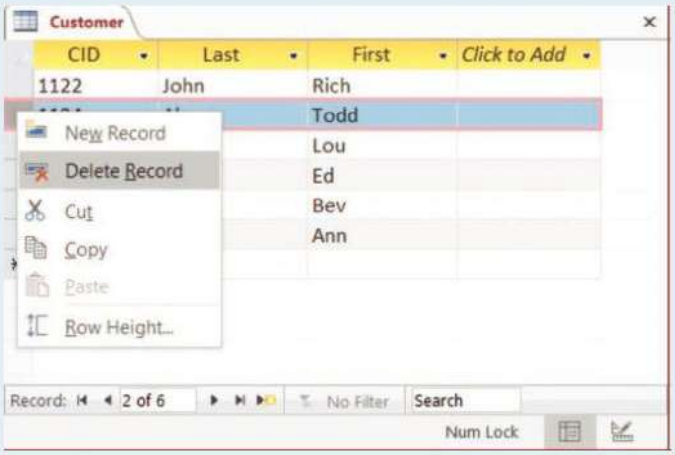


Fig 8.16 Deleting a record from the Customer table

5. Click on the first record again and change the CID to 112. Press *Enter* and you should see your validation rules working (Fig 8.17).

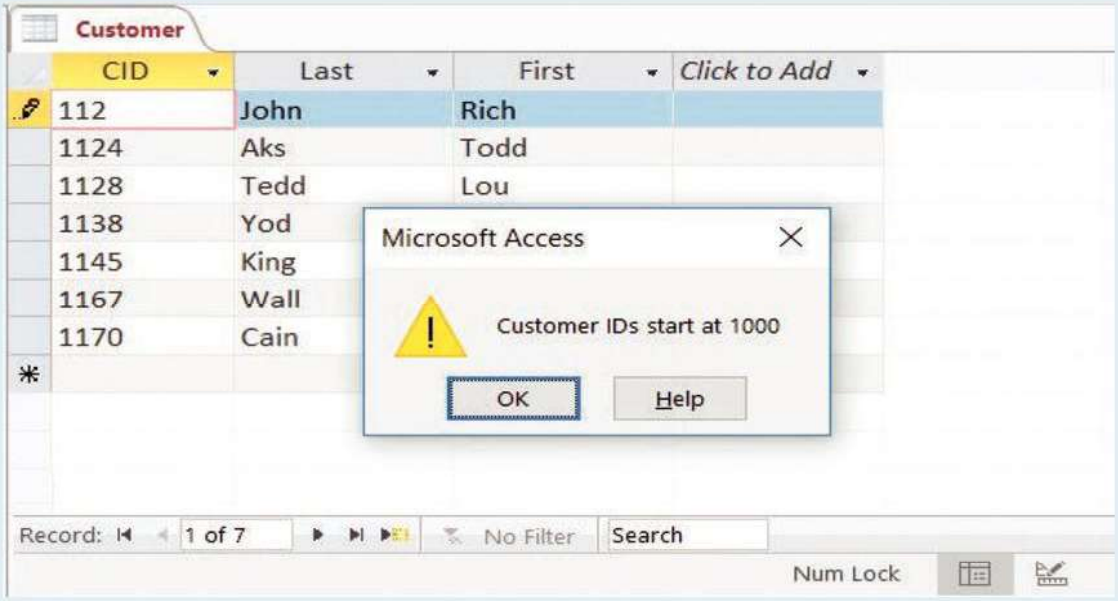


Fig 8.17 Entering a number that is not valid results in an error message

6 Select the Product table and double-click to open it.

7 Enter the following records:

PR-ID	Description	Cost
P3122	Pocket Diary	$8.95
P3429	Jumbo Clips	$4.15
P3430	Small Clips	$2.75

8 Select the Order table and double-click to open it.

9 Enter the following records:

Customer ID	Product ID	QTY	Discount
1128	P3430	10	
1128	P3639	500	Y
1128	P3902	50	

Exercise 6: Create a basic form

Microsoft Access does a very good job of creating a form, and even provides a Form Wizard for creating more complex forms. First, follow these steps to create a basic form:

1 Open the Bookshop database with the Customer, Product and Order tables. Each table should have three records of data.

2 Select the Customer table. Then in the menu or ribbon, select the Create tab and then the Form icon 

.

3 This option creates a simple form using all of the fields in the table.

4 Use the form to enter the following records in the Customer table:

CID	Last	First
1138	Yod	Ed
1145	King	Bev
1167	Wall	Ann
1170	Cain	Zed

Exercise 7: Create a form using the wizard

1 Select the Product table. Then activate the Form Wizard. In some versions of Access, there may be an icon Form Wizard, which can also be found in the More Forms option on the ribbon.

2 Select the fields needed for the form by selecting a table or query from the Tables/Queries drop-down menu. To select the fields you want to view on your form use the > arrow to move them from the Available Fields window in the left pane to the Selected Fields window in the right pane. You would use >> if you are selecting all of them (Fig 8.18).

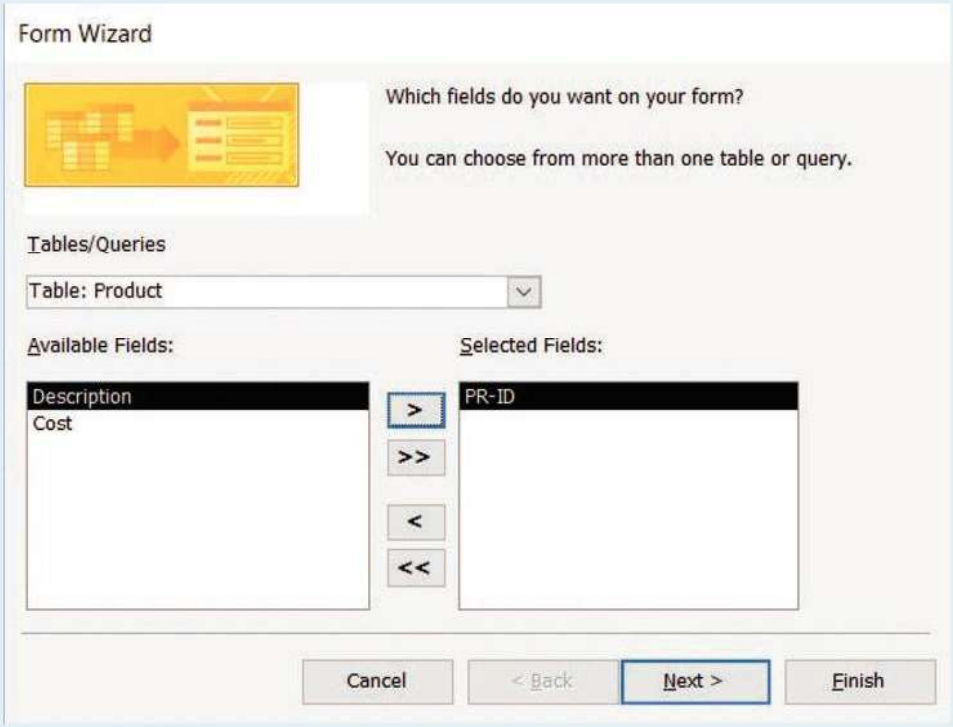


Fig 8.18 Using the Form Wizard to select fields

3 Click Next.

4 Select the layout you wish and click Next.

5 You may be asked to select the style you desire. Use a light background if you are going to print your form.

6 Click Next.

7 Give your form the name Product and click Finish.

8 You should see your form.

9 Enter the following data using the form:

PR-ID	*Description*	*Cost*
P3639	Round Labels	$3.95
P3745	Receipt Book	$7.55
P3899	Highlighter	$1.75
P3902	Black Pen	$1.25

Exercise 8: Completing the data entry with the Order table

1 Use the Form Wizard or another method to enter the following data in the Order table:

Customer ID	Product ID	QTY	Discount
1138	P3745	100	
1145	P3122	5	
1167	P3430	10	
1167	P3639	100	Yes

Exercise 9: Creating a form with a sub-form

1 Select the Customer table.
2 Activate the Form Wizard.
3 Select the fields needed for the form by selecting a table or query from the Tables/Queries drop-down menu. Use the > arrow to move the CID, then First and Last field names from the Available Fields window in the left pane to the Selected Fields window in the right pane.
4 Practice adding additional fields from the tables so that the fields are selected in the following order:
   a Select the Order Table and add the CID and PR-ID fields (Fig 8.19a).
   b Select the Product Table and add the Description and Cost fields.
   c Select the Order Table again and add the QTY and Discount fields.
5 Click Next.
6 If prompted, view the data by Customer. You will be able to see the form and sub-form in the preview pane (Fig 8.19b).
7 Click Next.
8 Leave the layout option of the sub-form as Datasheet and click Next.
9 Type Customer Form as the name of the main form and leave the name of the sub-form as Order sub-form (Fig 8.19c).
10 Click Finish to see the final form (Fig 8.19d).
11 Locate record 3 to view the orders for the customer Lou Tedd.
12 At the bottom of the form you will see Record: 3 of 7.
13 Add the following order for Lou Tedd. Click in the PR-ID and enter P3745. The Customer ID, description and cost of that product should fill in those fields. Enter 25 for the quantity and leave the discount blank.
14 Click the icon to add a new record. Enter the following record of data for a new order:
   CID: 1199
   First: June
   Last: Jarway
   PR-ID: P3745
   QTY: 25

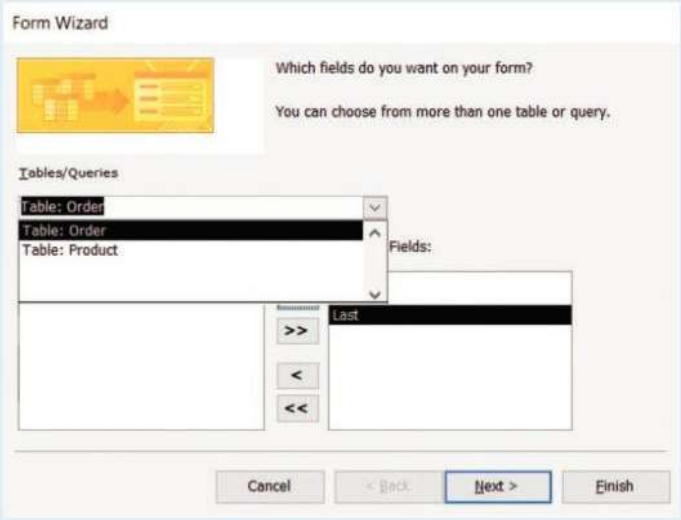


Fig 8.19a Selecting the Order table

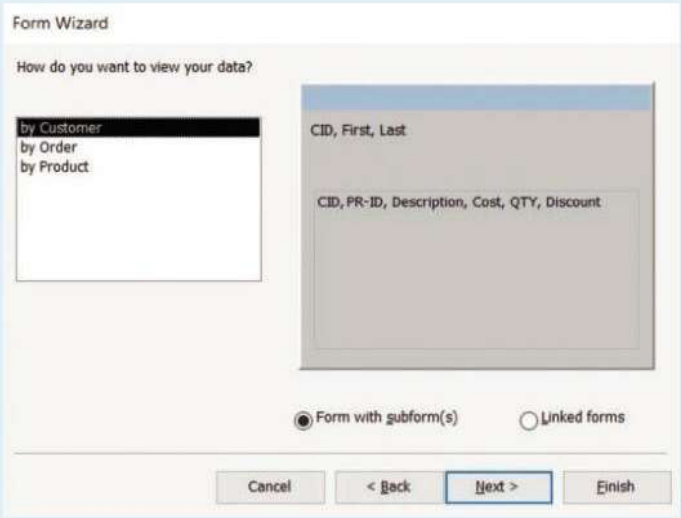


Fig 8.19b Viewing the data by Customer – you can see the form and sub-form in the preview pane

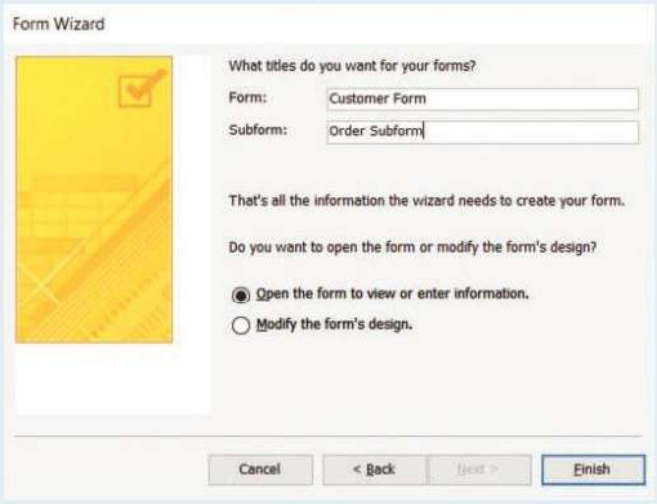


Fig 8.19c Naming the form and sub-form

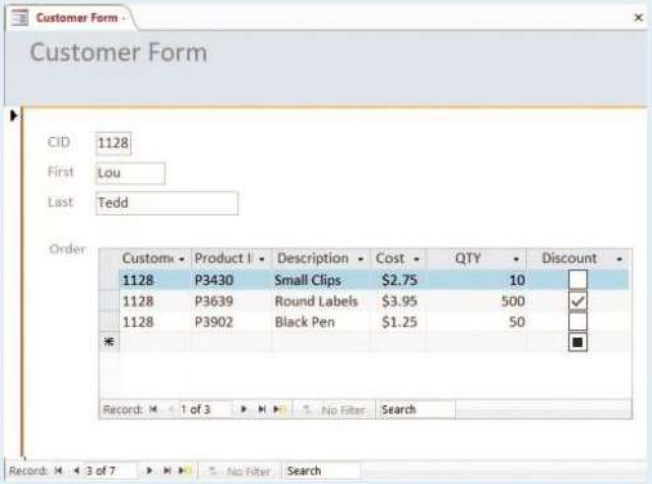


Fig 8.19d The final form showing the orders for customer Lou Ted

Exercise 10: Create a form for the Sports database that contains the Team and Division tables

TEAM			
AthleteID	**NameOfAthlete**	**Code**	**Gender**
121	Jade Boyce	U13	F
231	Shade Skeete	U13	F
351	Neil Hall	U20	M
142	Figman McJig	O20	M

DIVISION	
Code	**Category**
U13	Under 13
U20	Under 20
O20	Seniors

1. Create a database called Sports.
2. Create the two tables, choosing the most appropriate data types in each table.
3. Select a suitable primary key for each table.
4. Create a form to enter the data in the tables as follows:
   - Main form: Code and Category from the Division table
   - Sub-form: AthleteID, NameOfAthlete and Gender from the Team table.

Computers have important advantages over manual systems since they can work extremely quickly, and automatically, to make calculations and to retrieve and sort data into some useful form. All these functions are known as data processing.

Searching

Searching a database is reasonably straightforward once you realise that you must tell the program precisely what to search for. You do this by giving clear criteria (conditions) for the search. You might know what you want to find, but the software does not – until you tell it.

A **query** (also known as a filter or search) is used to answer a question using the data in a database. The database is searched to find all the records that match a particular condition. A query is the question you ask, such as 'List the first names of the customers who ordered round labels.'; it is not the results (for example, Ann and Lou).

You may make several queries to ask different specific questions. When a query is 'run' it produces as output a list of all the records that match the condition that defined the query. Once you have created a query, you can run it as often as needed. Even the results of the query will be updated if you add more data to the table.

When you begin to design your queries, ask yourself:

- What specific question do I want to ask?
- What data (fields) do I need displayed?
- What sort order will help me the most?

Queries can be as simple or as complex as you like. To search a database, you must first enter the search criteria into one or more fields for their tables. Then you instruct the program to find all the records that match the conditions that you have set. For example, to find the products with orders that are less than 50, you would type <50 in the QTY field of the Order table.

First	Description	QTY
Lou	Small Clips	10
Bev	Pocket Diary	5
Ann	Small Clips	10

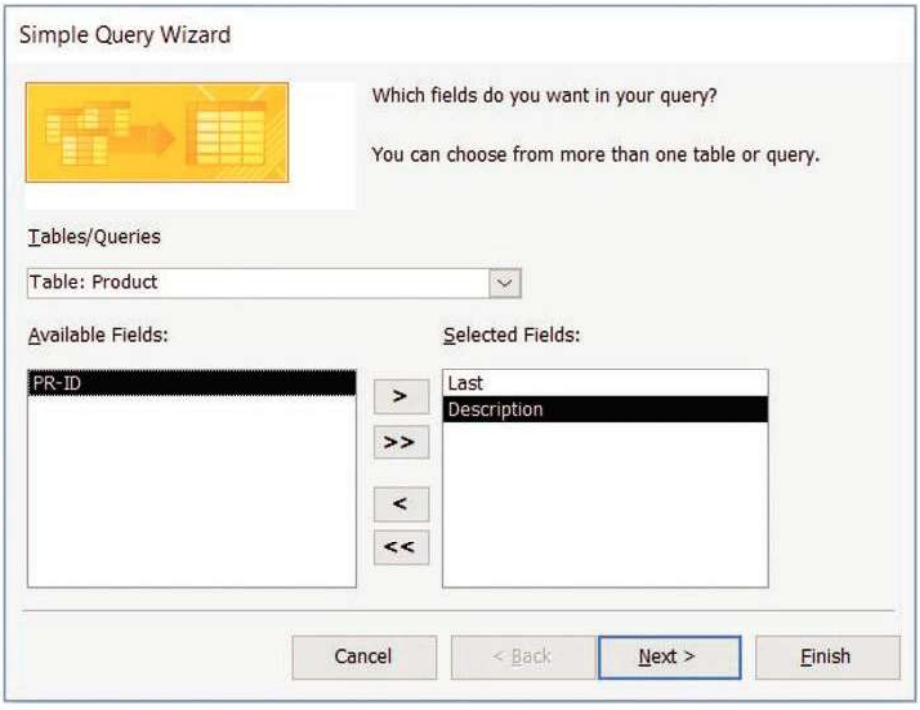


Fig 8.20 Using the Query Wizard to select fields in a query

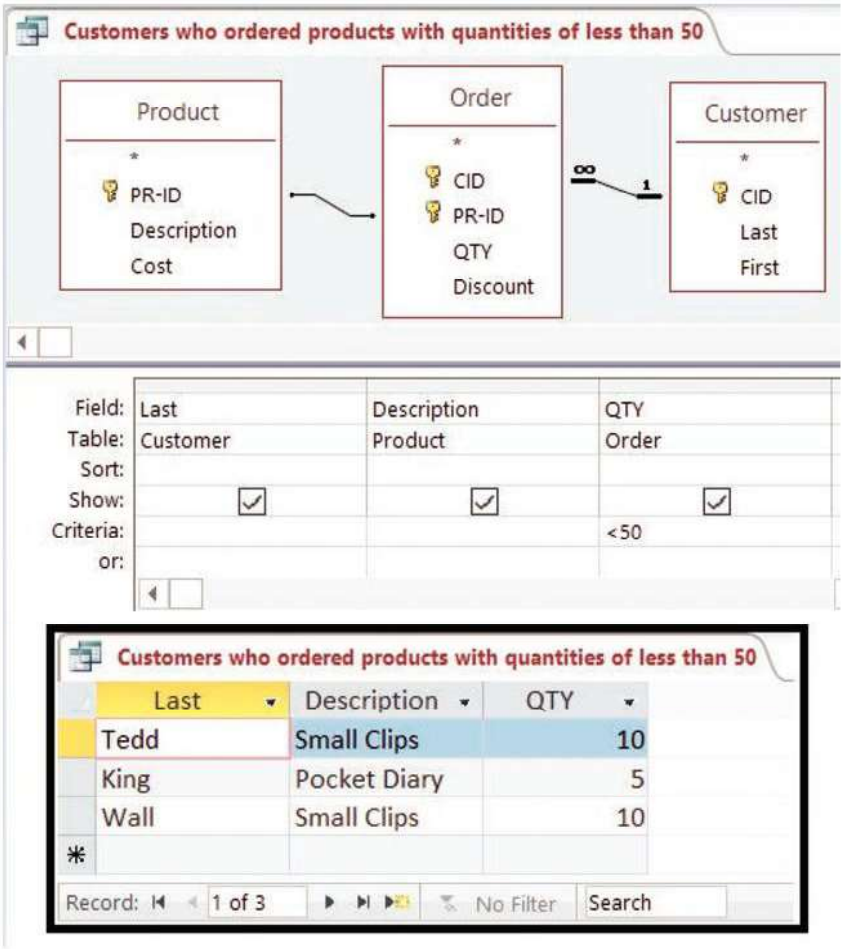


Fig 8.21 Creating a query to list the products with orders of less than 50

After executing the query, you may see one or more records that were produced from your search.

The comparisons that you can make in queries will depend on which database program you are using. Table 8.5 lists some common operators used.

Searching for specific records

The word 'criteria' in a database means you want to find only certain records and ignore others (the ones that do not meet the criteria conditions). If you want to limit the results of your query, you can set up specific criteria. If a record meets those criteria, it is included in the results. Note also in Table 8.5 that the criteria does not include any formatting such as the dollar sign ($) or percentage symbol (%). Any search using text as the criteria is enclosed in double quotation marks. For example, suppose you wish to find the product with ID P3430. This produces one result: Small Clips costing $2.75. Figure 8.22 illustrates how you place P3430 in the criteria row below the field name where you wish to find the specific records. The other fields can be selected (ticked) to indicate that you wish to see their results as well. If you only wish to see the product description and the costs, make sure only those two fields are selected.

Table 8.5 *Operators used in searching databases*

Operator	*Meaning*	*Example*
=	Equal to	"P3122" searches the product ID number field to find all products with that identical product ID
>	Greater than	> 5 searches the cost field to find all products that cpst more than $5.00
<	Less than	< 5 searches the cost field to find all products that cost less than $5.00
>=	Greater than or equal to	>= 5 searches the cost field to finds products whose cost is greater than or equal to $5.00
<=	Less than or equal to	<= 5 searches the cost field to find products that cost less than or equal to $5.00
*	Asterisk, known as a wild-card character, can be used to represent one or more characters	"R*" or like "R*" will find all products whose first character or letter begins with R, for example Receipt books and Round labels

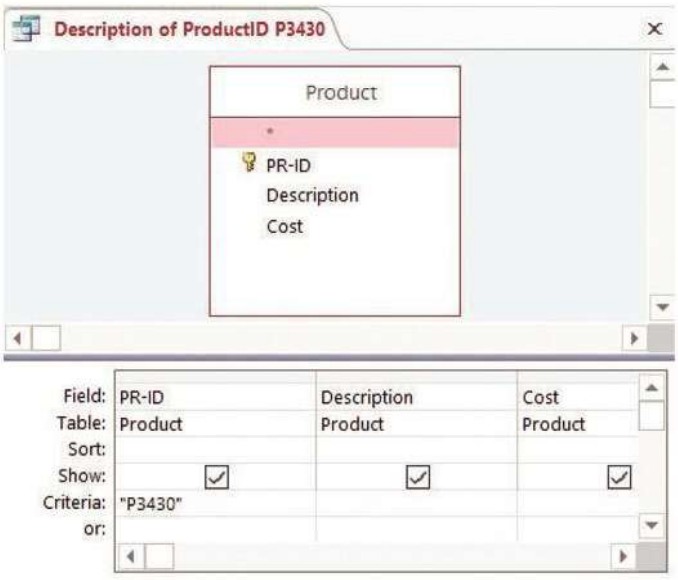
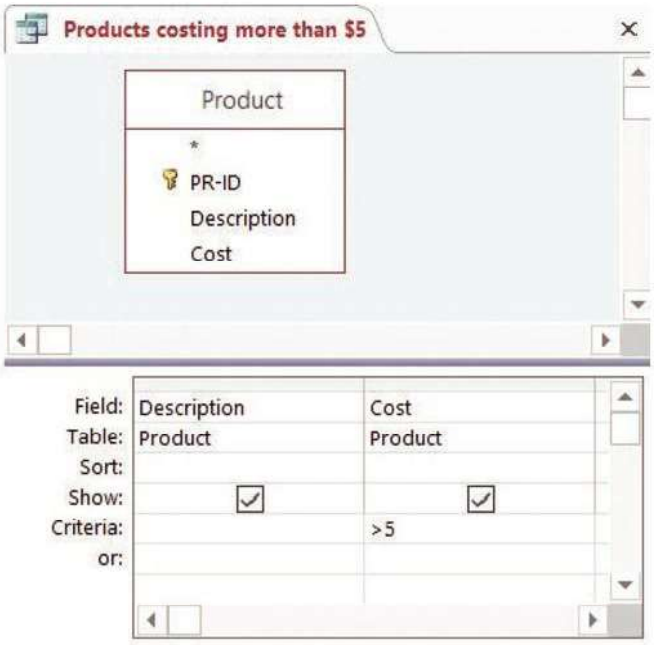


Fig 8.22 Searching for a description for product P3430



Description	Cost
Pocket Diary	$8.95
Receipt Book	$7.55

Fig 8.23 Query to find products that cost more than $5.00 and its result

Also notice that there is only one table selected. If the fields you need for the query are found in one table then the other tables should be deleted from the query. Otherwise you may find duplicate records in your result. To delete a table in query design, select the table, right-click and select remove table or select the table and press the *Delete* key on your keyboard.

Figures 8.23 and 8.24 show other examples of how to find products that cost more than $5.00 (Fig 8.23) or those that begin with 'R' (Fig 8.24).

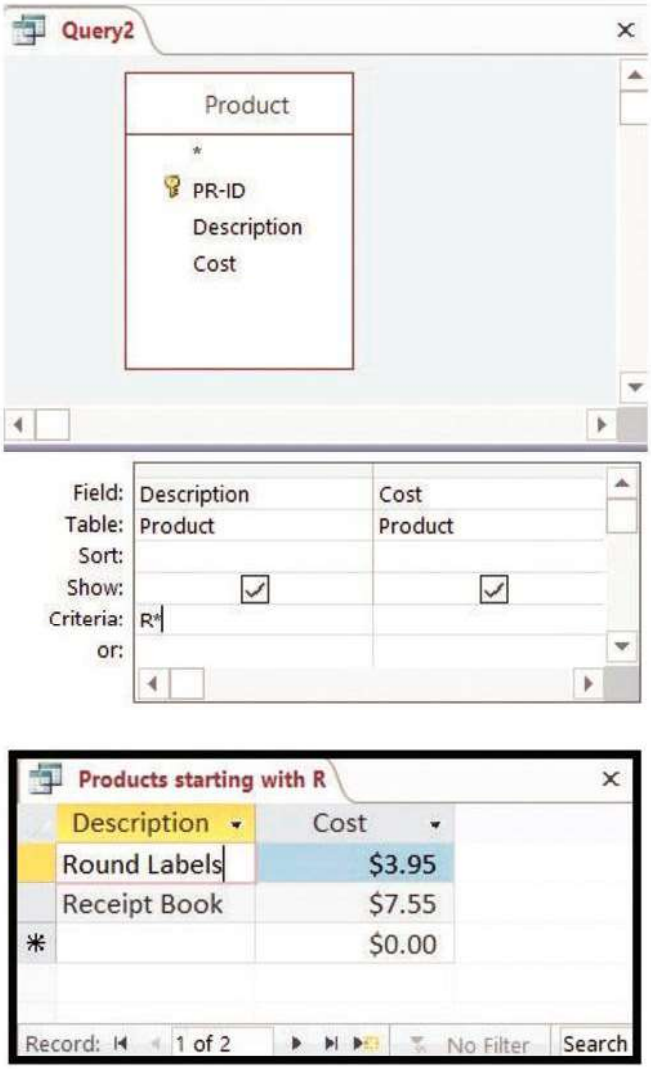


Fig 8.24 Query to find the names of products that begin with 'R' and its result

Queries using more than one field

You can create more complicated queries by linking together more than one search condition. For example, find the customers who ordered products costing less than $5 in quantities of more than 100. Figure 8.25 shows how in Microsoft Access <5 and >100 are placed on the same criteria row but under their field names. This joins the two queries with the AND operator.

Another query finds the customers who ordered products costing less than $5 OR products with quantities of more than 100. Figure 8.26 shows how in Microsoft Access <5 and >100 are placed on two lines labelled as 'criteria' and 'or'. It does not matter which one is entered in the criteria or the OR row once they are not in the same row. This joins the two queries with the OR operator. The result below can be saved as Less than $5 or More than 100.

First	Description	Cost	QTY
Lou	Small Clips	$2.75	10
Ann	Small Clips	$2.75	10
Lou	Round Labels	$3.95	500
Ann	Round Labels	$3.95	100
Lou	Black Pen	$1.25	50

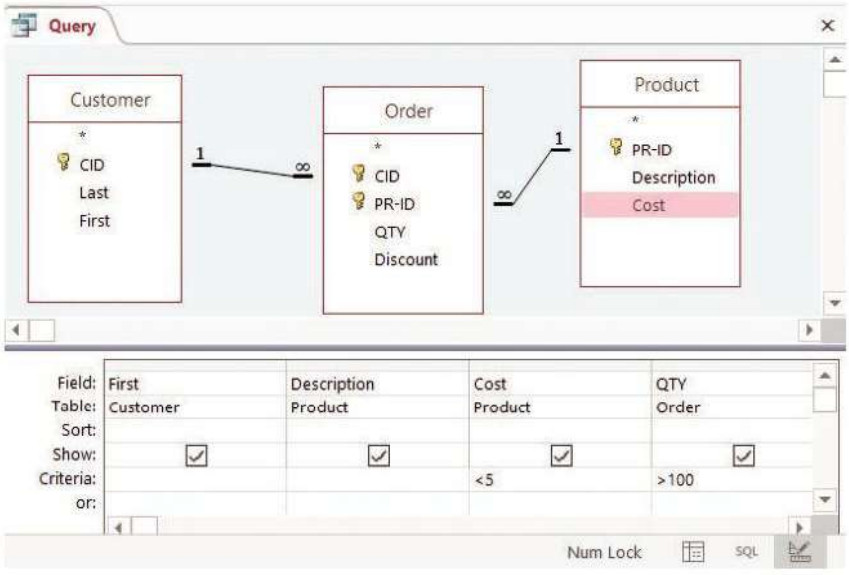


Fig 8.25a Query to find the customers who ordered products ***costing less than*** *$5 in quantities of more than 100*

Fig 8.25b results from query to find the customers who ordered products ***costing less than*** *$5 in quantities of more than 100*

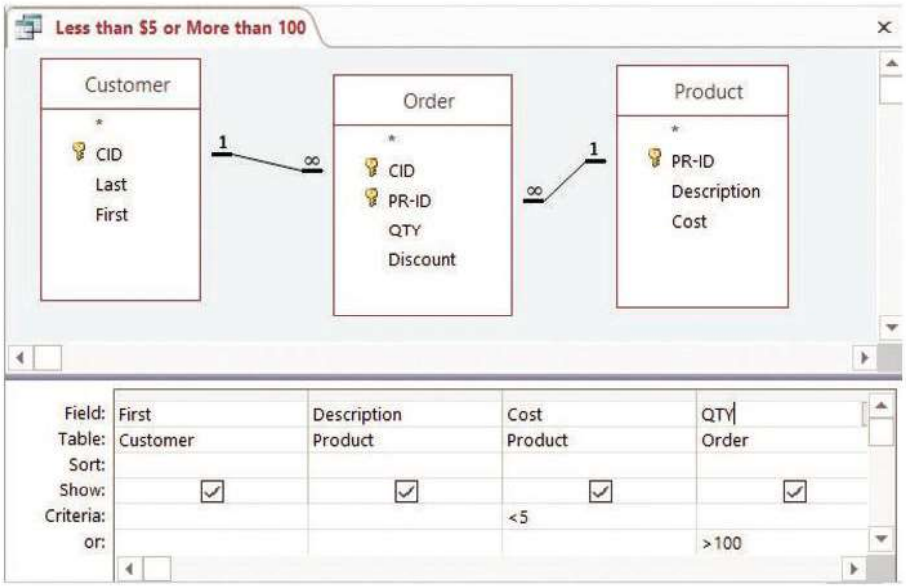


Fig 8.26 Query to find the customers who ordered products costing less than $5 OR quantities of more than 100

Notice that in these queries all three tables were used since fields were needed from each table. However, if fields from the Customer and Product tables were needed, the Order table should not be deleted since it is linking these two tables in the database. For example, suppose the Bookshop needed to contact all customers who ordered Black Pens. The First and Last field names would be selected from the Customer table, while the Product ID (PR-ID) or Description for the Black Pens could be selected from the Product table and not the Order table.

The words AND and OR are the only ones that can be used to link two simple conditions together to make a more complicated query. The difference between using AND or OR to join two conditions is:

- AND: Find only records which match *both* of the conditions.
- OR: Find records that match *either* or *both* of the conditions.

Reversing queries

Sometimes you may want to search for records which do not match a particular criteria. For example, you may want to find all customer orders except those for customer Lou. The keyword NOT is used to do this (Fig 8.27a).

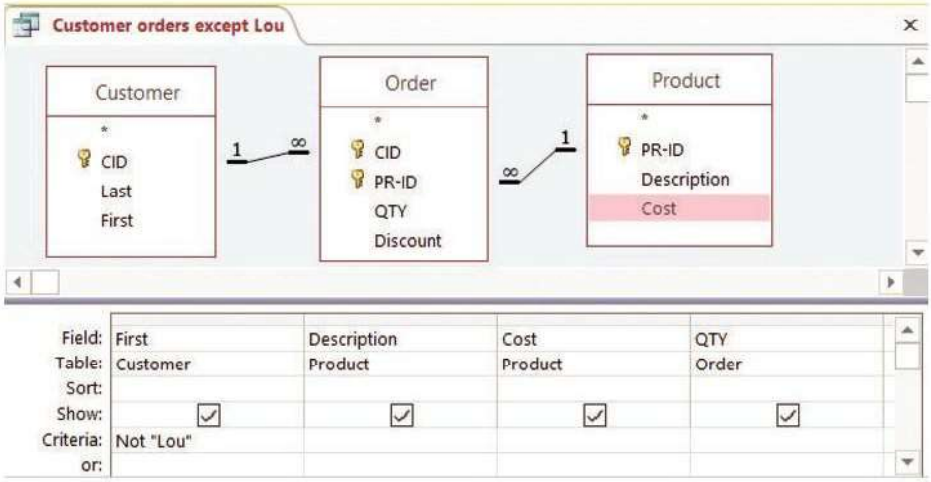


Fig 8.27a Query to find all orders NOT placed by Customer Lou

Fig 8.27b Result of query to find all orders NOT placed by Customer Lou

Sorting

Most databases will let you sort data so that it is displayed in a specified order. To sort a database into an order you must specify:

- which field in the database you wish to use to order the records
- whether you want the records in ascending (A to Z) or descending (Z to A) order.

Sorting records temporarily reorders the database file. Sorting allows you to browse, update, export or print records in a particular order. You can also sort numbers, dates and times in ascending and descending order. Being able to sort records quickly is one of the advantages of using a database. Depending on your version of Microsoft Access, you may have one or more methods of sorting your data. However, the result remains the same. The following three examples show how your results can be sorted by the last name field in ascending order.

Figure 8.28 shows how the sort option in Design view is used to order the data.

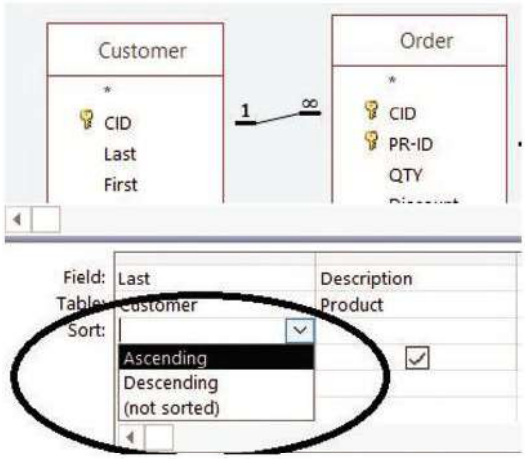


Fig 8.28 Design view showing how the last names of the customers can be sorted in ascending order

Figure 8.29 shows how the Sort and Filter menu option in Datasheet view is used to sort the data.

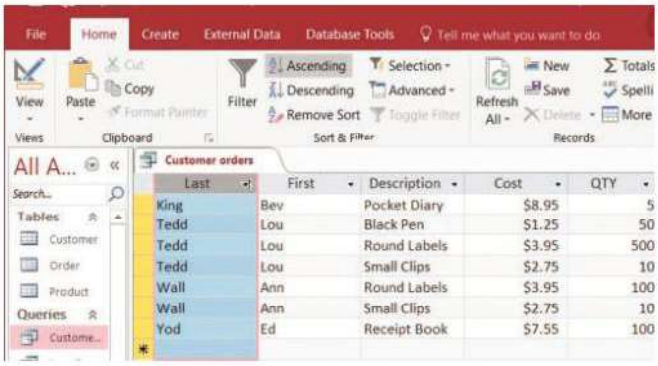


Fig 8.29 Using the Sort and Filter menu option in Datasheet view to sort the data

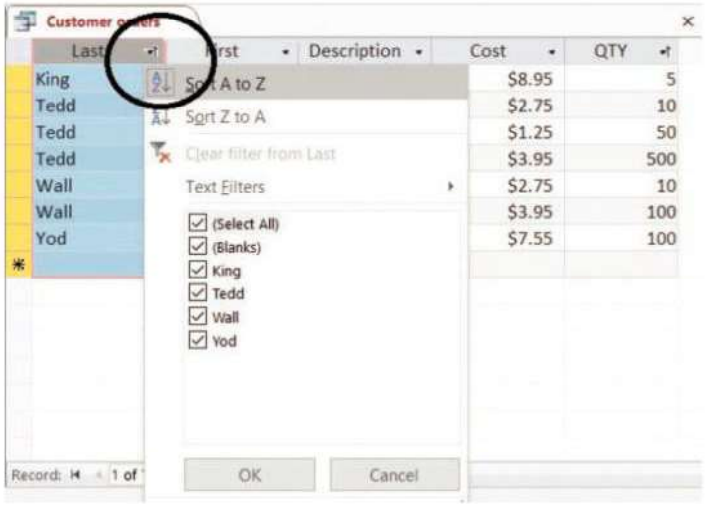


Fig 8.30 Using the sort option to the right of the field name to order the data

Figure 8.30 uses the Sort option in the field names to order the data.

All these methods will give the following results.

Last	First	Description	Cost	QTY
King	Bev	Pocket Diary	$8.95	5
Tedd	Lou	Small Clips	$2.75	10
Tedd	Lou	Round Labels	$3.95	500
Tedd	Lou	Black Pen	$1.25	50
Wall	Ann	Small Clips	$2.75	10
Wall	Ann	Round Labels	$3.95	100
Yod	Ed	Receipt Book	$7.55	100

Questions

1 Select the most suitable options that describe the sorting of data as A to Z:

- **i** ascending
- **ii** descending
- **iii** top to bottom
- **iv** bottom to top.

Use the query template shown in Figure 8.31 to answer questions 2 and 3:

2 You wish to list the names of the products that cost under $3.00.

- **a** Write one or more tables that would be used in the query.
- **b** List the field names that would be used in the query.
- **c** Write the field name and the criteria for the query.
- **d** How would the criteria change if the query was changed to list the products that cost $3.00 or less?

3 You wish to list the names of the products that are on discount.

- **a** Write one or more tables that would be used in the query.
- **b** List the field names that would be used in the query.
- **c** Write the field name and the criteria for the query.

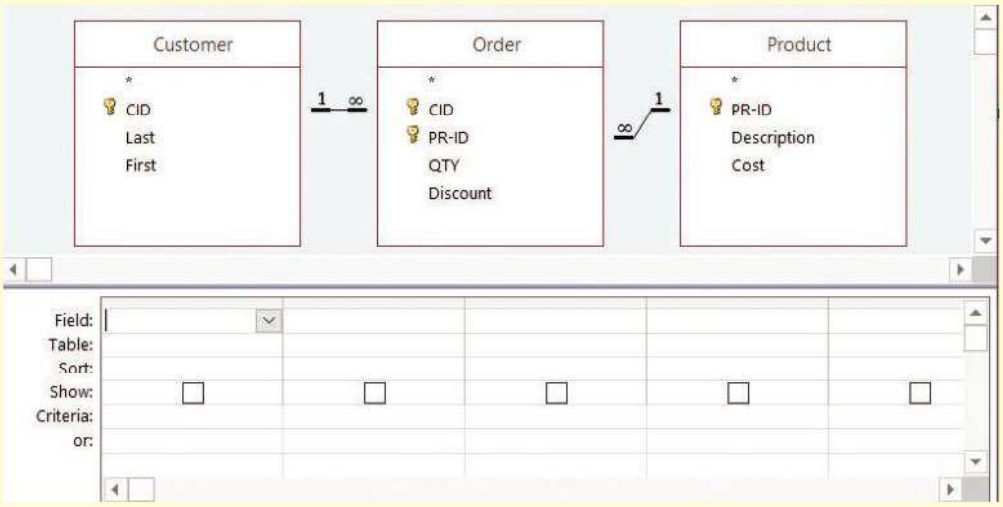


Fig 8.31 Query template

Practical exercises using Microsoft Access

Exercise 11: Searching and sorting

1 Use the Bookshop database to create queries to list:

- **a** the names of the products that cost under $3.00
- **b** the products that cost more than $5.00
- **c** the names of the products that are on discount; sort by the name of the product
- **d** all customers who ordered Black Pens; sort by the customer's last name
- **e** the customers who ordered products costing less than $5 in quantities of more than 100
- **f** the customers who ordered products costing less than $5 OR quantities of more than 100
- **g** all orders except those placed by Customer Lou
- **h** the names of products that begin with 'R'
- **i** the customers whose last name begins with 'Y'.

A calculated field is a 'virtual field' in a query or report. The value in this calculated field is a function of one or more fields from a table or query.

The syntax of a calculated field is always the same:

New field name: [Expression]

Calculated fields with numbers

For example, the syntax for the calculated field called Total Cost is:

Total Cost: [Cost] * [QTY]

where Cost is the cost of the item and QTY is the quantity ordered (Fig 8.32).

Note that:

- There must be a colon after the new field name.
- Only the field names are enclosed in square brackets.
- The field names must be written exactly as they are in the table, otherwise the system may not recognise them.
- The calculated field name (such as Total Cost) should be a meaningful name and can contain spaces and underscores or dashes.

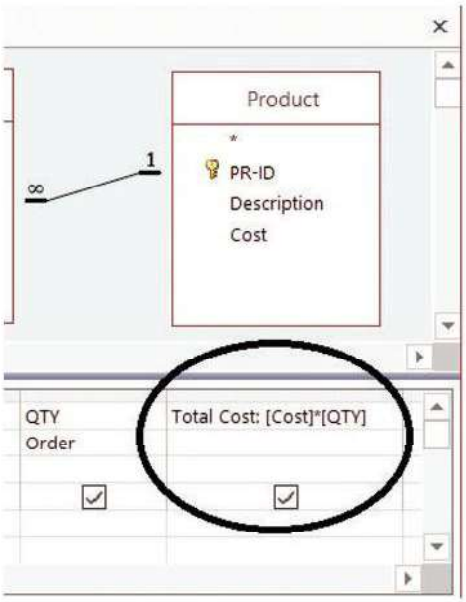
In this case, the expression for this calculated field involves two fields from the Bookshop table (Cost and QTY), and the multiplication operator. The Total Cost field may contain $27.50 as one of the calculated field results.

Mathematical functions

An aggregate query is a special type of query within Access that allows you to group your numerical data to perform specific mathematical functions. Some of these functions are shown in Table 8.6.

Table 8.6 *Aggregate functions*

Group By	matching values in a field are grouped together
Sum	will give a total of all the records in this field for the groupings indicated in the Group By field
Avg	will give the average of all records in the specified fields within each grouping
Min	will return the single lowest value from the group of records
Max	will return the single highest value from all records within the grouping
Count	counts the number of entries within the designated field



QTY	Total Cost
10	$27.50
500	$1,975.00
50	$62.50
100	$755.00
5	$44.75
10	$27.50
100	$395.00

Fig 8.32 Creating a calculated field for the total cost of a product

To use an aggregate function, you must first create a query either by activating the wizard or using Design view. Add fields which can be used in a calculation. Next click on the Totals icon Σ. This causes an extra row in the criteria area to be added. A suitable aggregate function from the Group By list can be selected and then the results are viewed.

For example, to find the total cost for each customer, after adding the cost field to the query, click on the Totals icon (Fig 8.33). Then in the Cost field select the SUM function from the Group By list (Fig 8.34). The total cost of each customer's order is shown in the result. Save the query as Costs of Orders.

You can choose to sum the items, giving an overall total of the items sold (Fig 8.34). Save the query as 'Sales' and click on the Datasheet view icon to see the results. You can click on the Design view icon again to perform other functions shown in Table 8.6.

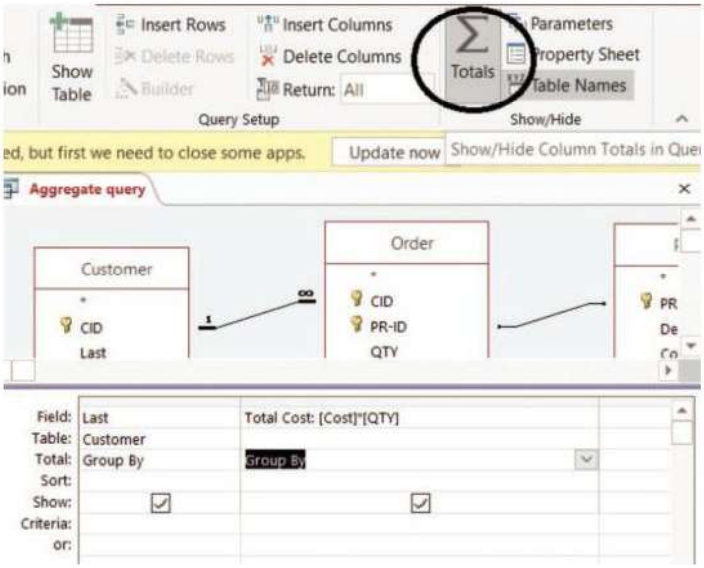


Fig 8.33 The Totals icon is used to create aggregate queries. The icon is found in the Design view of the query menu

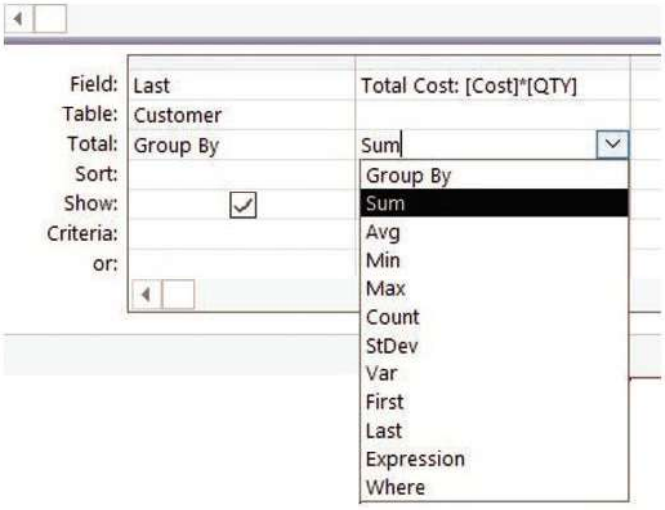


Fig 8.34 Clicking on the Totals icon in the query produces an extra line 'Total' in the design grid, where the list of aggregate functions is shown

Results of using the SUM aggregate function:

Last	Total Cost
King	$44.75
Tedd	$2,065.00
Wall	$422.50
Yod	$755.00

Calculated fields with text

Suppose you wish to produce a field containing the product ID and its description. The syntax for the calculated field called Desc is:

Desc: [PR-ID] & [Description]

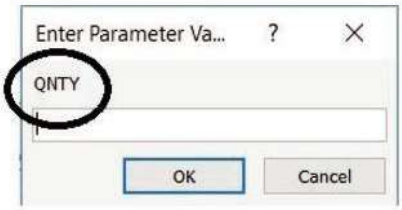
In this case, the expression involves two fields from the Product table (PR-ID and Description) and the ampersand (&) operator. So, the Desc field may contain 'P3639Round Labels' as one of the calculated field results.

Instead of having PR-ID and Description run together in the new Desc field, you may prefer to have a space separating the two parts. The syntax can therefore be modified to:

Desc: [PR-ID] & " " & [Description] to produce 'P3639 Round Labels'.

Errors in queries

Sometimes, after you have created a calculated field and run the query, you see a dialogue box asking you to 'Enter parameter value'. This occurs when you spell a field name incorrectly and your database program cannot find the misspelled field name (Fig 8.35). To solve the problem, simply correct the spelling mistake in the field name.



Access could not find the field name QNTY. The correct field name is QTY

Fig 8.35 A Spelling error in a calculated field

Questions

1 Given the field names Firstname, Lastname, Description and Cost, create calculated fields to produce the following:
   - **a** Join the Firstname and Lastname fields so that the result is similar to 'Jane Smith'.
   - **b** Join the Firstname and Lastname fields so that the result is similar to 'Smith, Jane'.
   - **c** Join the Description and Cost fields to produce output similar to 'Black pens cost $1.25'.

2 What is the typical cause of an error in a query?

3 Using the table called Bookshop, write calculations to:
   - **a** find the total cost of each product ordered where Total cost = Quantity × Cost
   - **b** calculate how much discount is deducted if there is 10% discount on the cost of the product
   - **c** find the mark-up on each product where New Cost = cost × 1.10.

4 State the function that will produce:
   - **a** the total cost of all orders
   - **b** the product that is the least expensive
   - **c** the product that is the most expensive
   - **d** the total number of products.

Practical exercises using Microsoft Access

Exercise 12: Calculated fields

1 Use the Bookshop database to create the queries to:
   - **a** calculate how much discount is deducted if there is 10% discount on the cost of the product
   - **b** find the mark-up on each product where New Cost = cost × 1.10
   - **c** find the name of the product that is the least expensive
   - **d** identify the most expensive product
   - **e** calculate the total number of products.

Exercise 13: Creating an aggregate query

1 Open the Bookshop database.

2 To produce a calculated query, create a query, either by activating the wizard or using Design view.

3 Add the Last name field in the Customer table, Cost field in the Product table and QTY field in the Order table to the query.

4 Click on the Datasheet view icon to see the results. Then in Design view, in a new field type Total Cost:[Cost]*[QTY]

5 Remove the checkmarks in the rows for Cost and QTY that show those columns of data, so that only Last and the Total Cost are shown.

6 Click on the Datasheet view icon to see the results.

7 Select the Totals icon on the Menu bar. Your query now has an extra line 'Total' in the design grid, while the Group By option is now shown under each field in the query.

8 Change the Group By in the Cost and QTY columns to Expression, and change the Group By in the Total Cost column to SUM.

9 Click on the Datasheet view icon to see the results.

10 Save the query as 'Total Orders'.

A query simply selects particular records from the database. Often you may want to display only some fields from the records that are found or display the records in a particular order. A report is an effective way to present your data in a printed format, because you have control over the size and appearance of the data and headings. Fields in the report can be grouped with different kinds of subtotals including sum, minimum, maximum and average.

Typically, a report will let you specify these things:

- which fields to display
- where to display the fields
- the order in which records should be displayed
- how records should be grouped together
- what statistics you want the database to calculate from the records (for example, number of records, average values of fields).

Standard report formats

Standard reports can be either:

- tabular (set out as a table) (Fig 8.36) or
- columnar (set out as a form) (Fig 8.37).

When a *tabular report* is printed, multiple records are printed on each page. Each record takes up one line in a table. The names of the fields are printed at the top of the table. Options provided in the report will let you determine in what order the names are printed on the report. Grouping could be used to divide the field names on the report into groups. For example, the types of items could be grouped by the supplier. Most databases will print each group on a separate page. *Columnar reports* print each record on a separate form. Usually each form is printed on a different page. Columnar reports are often used for printing items such as tickets.


Order Report

Order Report

Last	First	Description	Cost	Quantity	Discount?
King	Bev	Pocket Diary	$8.95	5	☐
Tedd	Lou	Black Pen	$1.25	50	☐
Tedd	Lou	Round Labels	$3.95	500	☑
Tedd	Lou	Small Clips	$2.75	10	☐
Wall	Ann	Round Labels	$3.95	100	☑
Wall	Ann	Small Clips	$2.75	10	☐
Yod	Ed	Receipt Book	$7.55	100	☐

Friday, February 23, 2018 Page 1 of 1

Fig 8.36 Example of a tabular report

Fig 8.37 Example of a columnar report

Creating a report

The fastest and easiest way to create a simple report using a database such as Microsoft Access is with

a wizard which automatically creates a report by arranging all of the fields from a table or query into a neatly formatted report.

To create a report:

1 Activate the Report Wizard on the Access menu.
2 Select the table or query from the Tables/Queries drop-down menu (Fig 8.38).

Fig 8.38 Select the table or query from the Tables/Queries drop-down menu

3 Select the fields that should be displayed in the report by transferring them from the Available Fields window in the left pane to the Selected Fields window. Use the single right arrow button > to move fields one at a time or the double arrow button >> to move all the fields at once.
4 Click the Next button to move to the next screen.
5 Depending on the number of tables in your database, you may have options on how to view your data. You may not always see this option.
6 Decide if you would like to group your fields (Fig 8.39). Click the right arrow button > to add those fields to be grouped. Use the Priority buttons to change the order of the grouped fields if more than one field is selected.
7 Click Next to continue.

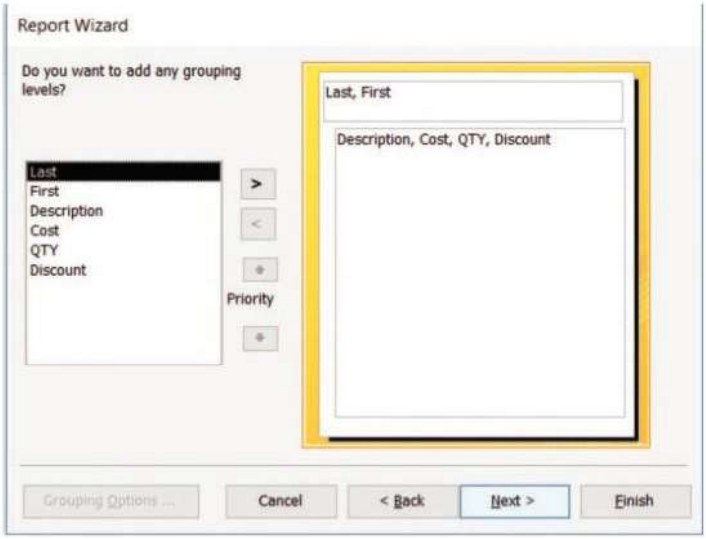


Fig 8.39 Decide if you wish to group any fields

8 Decide if you wish to sort any fields (Fig 8.40). If the records should be sorted, identify a sort order here. Select the Last field to be sorted by and click to choose from ascending or descending order. Click Next to continue.

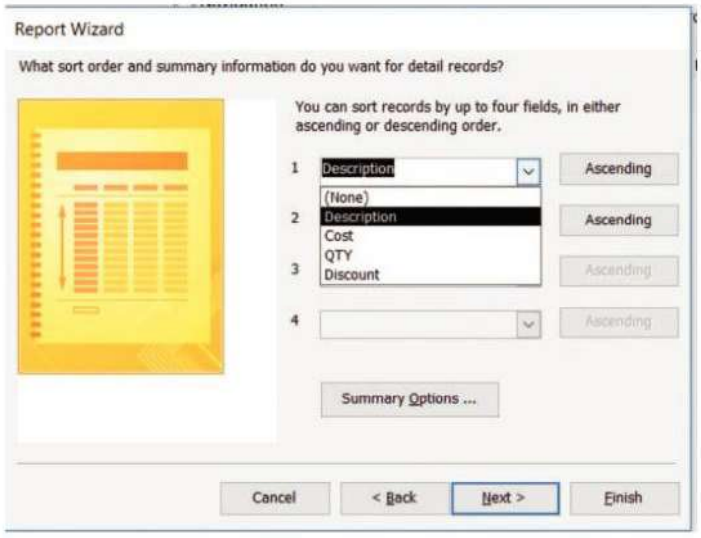


Fig 8.40 Decide if you wish to sort any fields

9 If there are fields in the report that contain numeric values, you can choose whether to apply certain summary functions such as sum, average, max or min to the report. For example, you may want to sum the quantities of items ordered (Fig 8.41). Click Next to continue.

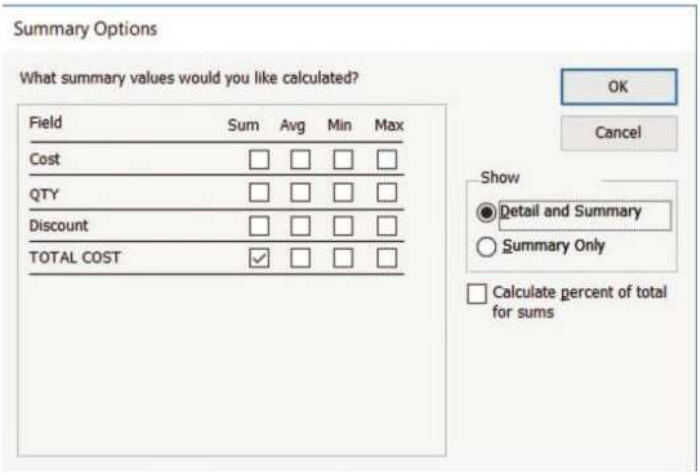


Fig 8.41 Summary options can be added to the report

10 Select the layout and the paper orientation you want and click Next if there are any other options regarding the orientation or the style you desire.

11 On the final screen, name the report (Fig 8.42) and select to open it typically in Print Preview mode. Click the Finish button to create the report. You may wish to preview the report before you print it.

Fig 8.42 Finally, give your report a title

12 If the fields are too wide you can adjust the final look of the report (Fig 8.43). Close Print preview mode and choose Design view or Layout view to adjust the width of the fields.

Fig 8.43 You can adjust the report if the fields are too wide

Fig 8.44 Order report after fields are adjusted

Fig 8.45 Order report with summary fields included

Printing the report

Once your report is ready to be distributed, it can be printed or exported to other applications. You can activate the print option to print your report to a printer or export it to a format that is compatible with a word processor. In recent versions of Microsoft Access, you are able to export your report to formats such as Excel, PDF, rich text, HTML or as an email attachment via Microsoft Outlook. Rich text is a common format for exporting a report to a word processing document such as Microsoft Word (Fig 8.46).

To export a report to rich text format:

1. Select the report that you wish to export.
2. Double-click to open the report.
3. Use the View icon and select Print Preview.
4. Select rich text option to export the report.
5. You will be prompted to confirm the name and location of the exported report.
6. Select the option to open the destination file after the export operation has been completed. This is useful to view the exported report.

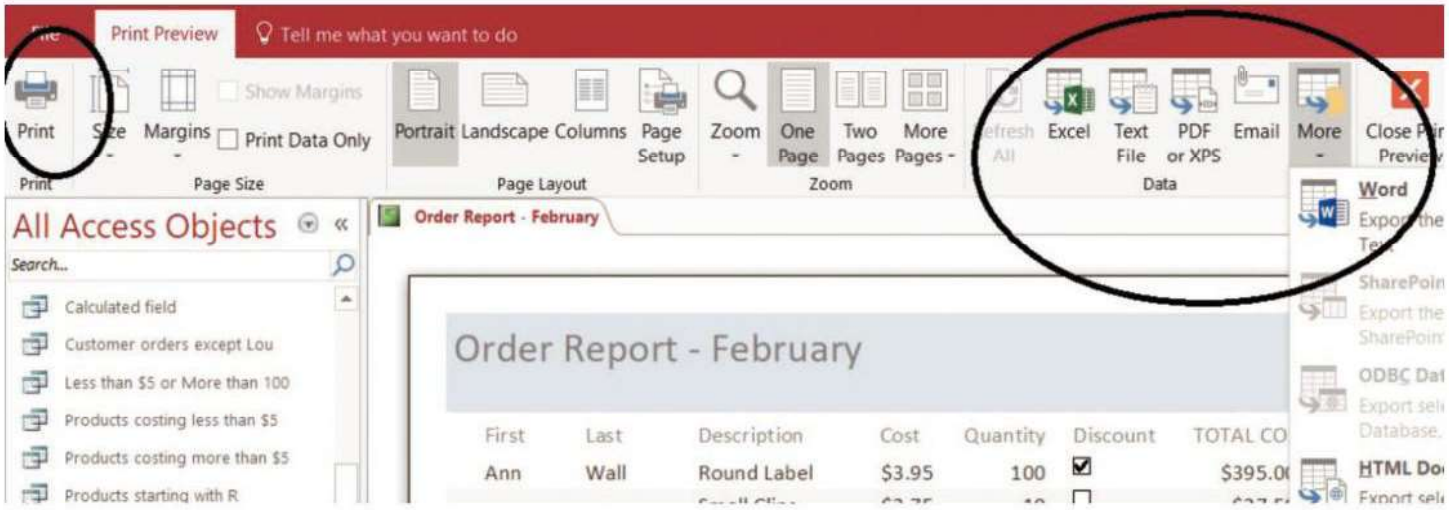


Fig 8.46 Reports can be printed or exported to other applications

Questions

Consider the report shown in Figure 8.44.

1. Explain whether the report is formatted as tabular or columnar. Identify:
   - **a** the data type of each field
   - **b** the grouping field
   - **c** the sort field
   - **d** the calculated field
   - **e** the title of the report
   - **f** how many products Lou Tedd ordered.

2 Consider the following report. Identify:

- **a** the data type of each field
- **b** the grouping field
- **c** the sort field
- **d** the calculated field
- **e** the title of the report
- **f** the summary function used in the report.

Payroll for September				
Department	**Last Name**	**First Name**	**Days Worked**	**Fees**
Marketing				
	Betham	Milo	28	$4,200
	Janis	Yannick	31	$3,720
	Pimmot	Ross	28	$3,360
	Rithmont	Cath	22	$2,640
Summary for 'Department" (4 detail records)				
			Sum	$13,924
Human Resources				
	Jomes	Mike	28	$3,360
	Steele	Jonat	29	$3,480
	Ummer	Rain	28	$3,360
Summary for Human Resources (3 detail records)				
			Sum	$10,203
			Grand Total	$24,127

Practical exercises using Microsoft Access

Exercise 14: Creating reports

1 Create a report using the Bookshop database.

- **a** Create the reports in Figure 8.36 and 8.37.
- **b** Create the report in Figure 8.43.
- **c** Group the fields by First and Last.
- **d** Sort by Description.
- **e** Use the block layout so that the first and last names are on the same row.
- **f** Give the report the name Order Report.
- **g** View the report.

2 Use the Sports database that you created in Exercise 10 to produce a report. (The data is shown again here.)

- **a** Use the fields Category, NameOfAthlete and Gender.
- **b** Group the report by Category.
- **c** Sort the data by Gender.
- **d** Name the report 'Report on Athletes by Category'.

3 Create another report using the Sports database.

- **a** Group by Gender.
- **b** Sort by Category.
- **c** Name the report 'Report on Athletes by gender'.

TEAM			
AthleteID	**NameOfAthlete**	**Code**	**Gender**
121	Jade Boyce	U13	F
231	Shade Skeete	U13	F
351	Neil Hall	U20	M
142	Figman McJig	O20	M

DIVISION	
Code	**Category**
U13	Under 13
U20	Under 20
O20	Seniors

You can import data from another database file, or from another application into an existing database. For example, schools could each send a database file to the national sports association for the upcoming sports meet. This data can then be imported into the association's database so that the categories of sports and age groups can be organised without having to re-enter the data. It is also possible to merge (combine) two or more database files.

Importing to a database

Here are two hints when importing data to Access:

- Name your fields using one-word names. Access allows you to use typical field names, but it may give you problems if you have field names with more than one word, especially if there are spaces between them. Therefore, it is best to use underscore (_) or simply keep the words in the field name together.
- If possible, export or import a small test amount of the data.

If the table has lots of records but not too many fields, try importing the first 10 and last 10 records, then write some queries. This will save time in the long run. If it has lots of fields, but not too many records, then go ahead and import the full table.

To import data to a table, select the table. Then right-click on the table and select Import. There are many options to choose which application you wish to use to import the data. The most common methods of importing data are from another Microsoft Access database, an Excel spreadsheet or a text file.

Exporting from a database

One way to export data is using the copy and paste method. Open the database containing a table or query that is to be copied (exported) to another database. Find the table or query in the Database window and highlight it. Then select all the data (Fig 8.47) and click on the Copy icon. Open an Excel sheet or Word document and paste the table or query in the required location.

Fig 8.47 Selecting all data in a table or query

Questions

1. What is the difference between exporting and importing data?
2. What is another method of exporting data from Access to a Word document?

Practical exercises using Microsoft Access

Exercise 15: Importing an Excel file

1 Open Excel and create the following worksheet starting at cell A1. Save it as DATA.

	A	B	C
1	**StudentID**	**Name**	**Age**
2	1123	Angelo	16
3	2212	Shanico	15
4	3346	Franco	17
5	3358	Vanessa	16

Generally, spreadsheet files should be formatted as follows:

- **a** The top row should contain only row headings or field names.
- **b** Either remove any cells that you do not wish to import or copy the ones that you do to another worksheet.
- **c** Save the file.
- **d** Remember the name of the worksheet and the file and its location.

2 Open Access.

- **a** Either create a new database, or open an existing one, depending on where you wish to place the Excel file. For this example, create a database called TEST.
- **b** On the menu, activate the tab with the External Data option.
- **c** Look for the icon to import data and select Excel or the Excel data type depending on your version of Microsoft Access.
- **d** If the name of the file you wish to import is not visible in the Import window, browse to the folder where it is located. In this exercise, browse for the spreadsheet file named DATA and select it.
- **e** You can choose to import it to a new table, or append it to an existing table
- **f** Select the worksheet from the pop-up window or, if you have named a range, select it.
- **g** Click Next.
- **h** If you get a message that the first row contains some data that cannot be used for valid field names, click OK to let the wizard assign valid field names. You can clean this up later.
- **i** If the box 'First Row Contains Headings' is not checked, tick it.
- **j** Click Next.
- **k** Select 'In a new table' and click Next.
- **l** You can change the name and data type of a field while importing:
  - **i** Select the field whose name you wish to change by clicking anywhere in that column
  - **ii** Click Next.
  - **iii** Choose the field name which will be the primary key and click Next. In this example choose StudentID.
  - **iv** Enter the name 'STUDENTS' for this table and click Finish.
- **m** You will receive a message that Access has finished importing the file.
- **n** Click OK.

3 If the Excel file was not formatted properly, and you get an error message, return to Excel and prepare the file as in step 1.

4 If there were blank lines in the Excel file you may wish to delete these 'records'.

Before you create a database you need to think about the most efficient and convenient way to store the information so that it can be retrieved in the format you want. Although a database can be modified after it has been created, it is much easier to get it right to begin with. Points to consider are:

- The names of your database, tables, forms, queries and reports. Like any other names, they should reflect the information they contain.
- The way in which you wish to store the information. For example, names of people should have at least three fields – title, initials and last name – to give maximum flexibility in using the data; similarly, addresses normally require at least six fields, for building/house name, street, town, county, postcode and country (if appropriate).
- The number of tables and fields you require and how the tables might be linked. Try to think of all the information you might want to get out of the database.
- The names of the fields. These may be up to 64 characters in length, but about 15 is a more normal size. You can use A–Z and 0–9 in your field names but are advised to avoid using spaces and symbols. Use underscores if you wish to separate field names. Try to choose sensible names which are self-explanatory.
- The size of the fields. For example, you might choose a field size of 8 for a Text field that stores eight-character codes. That means you cannot accidentally enter more than eight characters in the field.
- The type of data to be entered. You need to be aware of text or specific numbers that will not be used in calculations, as opposed to numbers that will be used in calculations, dates, currency, etc. This can help you to decide what data or field types to use for each field. You may want to consider primary key fields as Text fields if they do not involve calculations.
- The design of forms. If the information is being taken from, for example, a paper form, it is more efficient to design your online form so that the information is entered in the same order as it is read from the paper form.

When you have set up the structure of your database, you should:

- put some information into the database so that it can be tested
- run the queries and reports to test if the database is operating correctly.

Common design problems

There are several common pitfalls to keep in mind as you design your database. These problems can cause your data to be harder to use, maintain and retrieve. The following are signs that you should re-evaluate your database design:

- You have one table with a large number of fields that do not all relate to the same subject. For example, one table might contain fields with information about your customers as well as fields that contain sales information. Each table should contain data about only one subject.
- You have fields that are intentionally left blank in many records because they are not applicable to those records. This usually means that the fields belong in another table.
- You have a large number of tables, many of which contain the same fields. For example, you have separate tables for January sales and February sales, or for local customers and remote customers, in which you store the same type of information. Try consolidating all the information about a single subject in one table. You may also need to add an extra field, for example to identify the sales date.

Questions

1. Why it is advisable to design a database before you start to create it using the computer?
2. Give four design suggestions that relate to tables.
3. Why should you enter sample data into your database initially?
4. Explain one common design problem.

Multiple choice questions

1 When a customer makes an online hotel booking, the database is updated by using a:
- **a** table
- **b** form
- **c** query
- **d** report.

2 When making a payment online, the database is updated in:
- **a** batch mode
- **b** real-time
- **c** HTML
- **d** e-commerce.

3 The database view that presents data in a format that is similar to a spreadsheet is:
- **a** datasheet
- **b** design
- **c** print
- **d** report.

4 Which of the following defines what kind of data is used in a database?
- **a** type
- **b** length
- **c** name
- **d** description.

5 In databases and tables the rows refer to _____ and columns refer to _____.
- **a** data types, records
- **b** field names, records
- **c** records, data types
- **d** records, field names.

6 A database form can do all of the following ways, *except*:
- **a** add
- **b** modify
- **c** merge
- **d** delete.

7 Datasheet view allows you to enter _____ into your database table.
- **a** fields
- **b** raw data
- **c** data types
- **d** descriptions.

8 The correct database syntax for a field called Price which calculates 10% of the cost of an item in a field named Cost is:
- **a** Price: [Cost] × 10%
- **b** Price: [Cost]*.10
- **c** Price: [Cost] × [10%]
- **d** Price: Cost*.10

9 Data can be imported to a database using any of the following, *except*:
- **a** text delimited
- **b** tab-separated text
- **c** comma-separated values (CSV)
- **d** Portable Document Format (PDF).

10 A primary key field that appears in one table but is also located in another is called a(n):
- **a** foreign key
- **b** alternate key
- **c** candidate key
- **d** secondary key.

Short answer questions

11 BGI is a training company with branches in the Caribbean. The company needs to update its schedule of workshops and assign trainers.
- **a** For each of the following tasks, describe one application that would be most suitable to use:
  - **i** A list of employees who have completed recent workshops along with a schedule of upcoming workshops are sent to the Human Resource Director of each business.
  - **ii** Payments for travel and hotel accommodation are calculated for monthly salary payments.