Chapter 11

Database and charts

You already know how to

- define database record structures
- enter data into a database
- select subsets of data within a database
- sort data within a database
- produce and export reports for use within another package
- produce and label graphs and charts from a charting package
- create, rename and delete files using an operating system
- Access folders/directories within the operating system.

In this chapter you will learn how to:

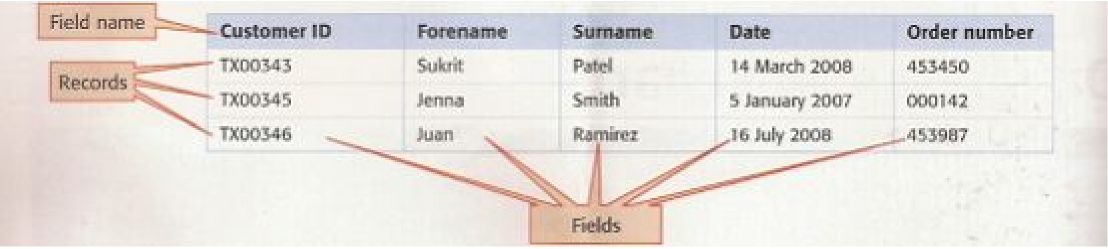
1. design and create the most efficient database structure to solve a problem
2. link data tables by creating and using relationships
3. understand the importance of accurate data entry
4. check data entry using validation and verification
5. select subsets, of data using multiple criteria
6. sort data in a database using multiple criteria
7. enter formulae to calculate results
8. use the display features of the database to produce a report
9. group data
10. summaries data
11. export data using different file formats
12. produce an appropriate type of graph or chart with suitable labels
13. add and edit data series and axes in a graph or chart.

11.1 Introduction and terminology

Data storage

A single **data item** is known as a field. Each **field** has a Specified type (e.g. it may contain numbers, text, a date, etc.). A **record** is a collection of fields (which may be of different data types) containing information about one person or one object. In terms of a database, a **file** is a logically organized collection of records.

MS Access stores its data in the form of tables, which are organized by rows and columns. Each row of the table contains a record. Each column in the table represents a field and each cell in that column has the same (pre-defined) field type. Within Access several tables can be stored in a single database. Here is an example of a small data table containing three records and five fields:



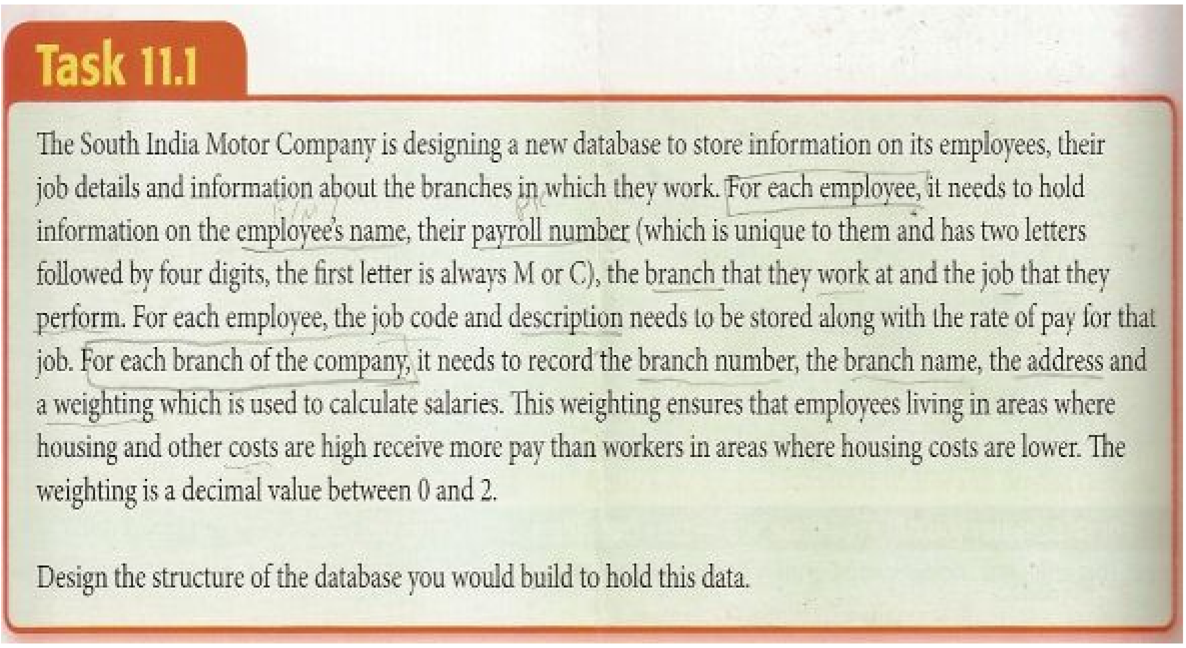
Field name
Records
Fields

Customer ID	Forename	Surname	Date	Order number
TX00343	Sukrit	Patel	14 March 2008	453450
TX00345	Jenna	Smith	5 January 2007	000142
TX00346	Juan	Ramirez	16 July 2008	453987

11.2 Designing a database structure:

Before you consider creating a database structure you the following questions:
- Which data items do you need to store?
- What field names should you use?
- Do you need one or more tables?
- Which field(s) should be primary key field(s)?
- What relationships will be needed?
- What data type should you use for each field?
- What field length should be used for each field?
- Which validation rules will help reduce data entry errors?

The answer to each question will help you to structure the database in the most efficient way



Task 11.1

The South India Motor Company is designing a new database to store information on its employees, their job details and information about the branches in which they work. For each employee, it needs to hold information on the employee's name, their payroll number (which is unique to them and has two letters followed by four digits, the first letter is always M or C), the branch that they work at and the job that they perform. For each employee, the job code and description needs to be stored along with the rate of pay for that job. For each branch of the company, it needs to record the branch number, the branch name, the address and a weighting which is used to calculate salaries. This weighting ensures that employees living in areas where housing and other costs are high receive more pay than workers in areas where housing costs are lower. The weighting is a decimal value between 0 and 2.

Design the structure of the database you would build to hold this data.

Which data items do you need to store?

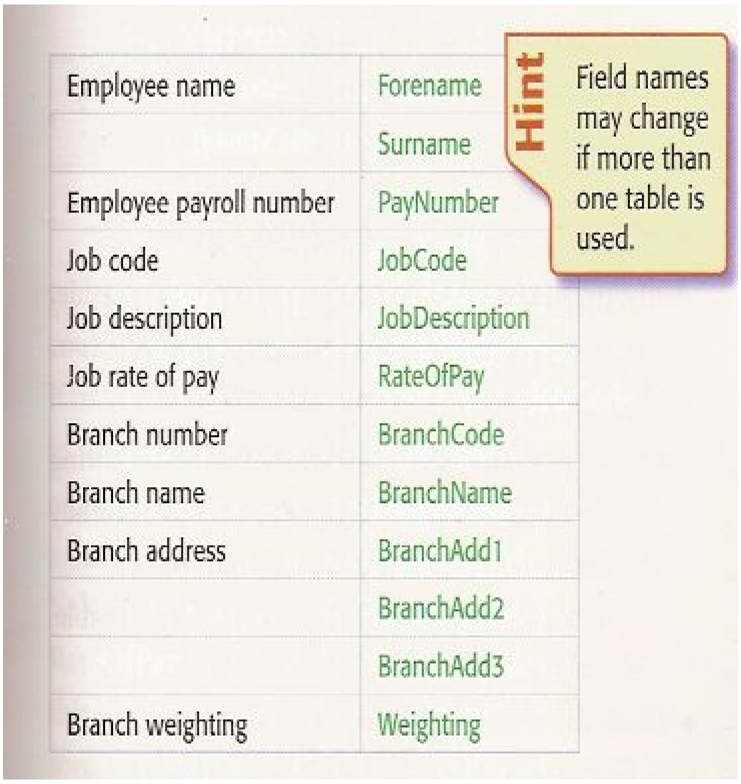
From the task you are told that the following data should be stored:

- Employee name
- Employee payroll number
- Job code
- Job description
- Job rate of pay

- Branch number
- Branch name
- Branch address
- Branch weighting.

What field names should you use?

The field names must be short enough to ensure that printouts fit on a page without wasted space and long enough to be meaningful In some database packages you cannot have a field name longer than **eight characters**. Check if this is the case with the software you are using. A good convention is to avoid **using spaces** within field names as some databases do not accept these (Access does accept them).
Looking at the data items which need to be stored, you will create a field name for each:



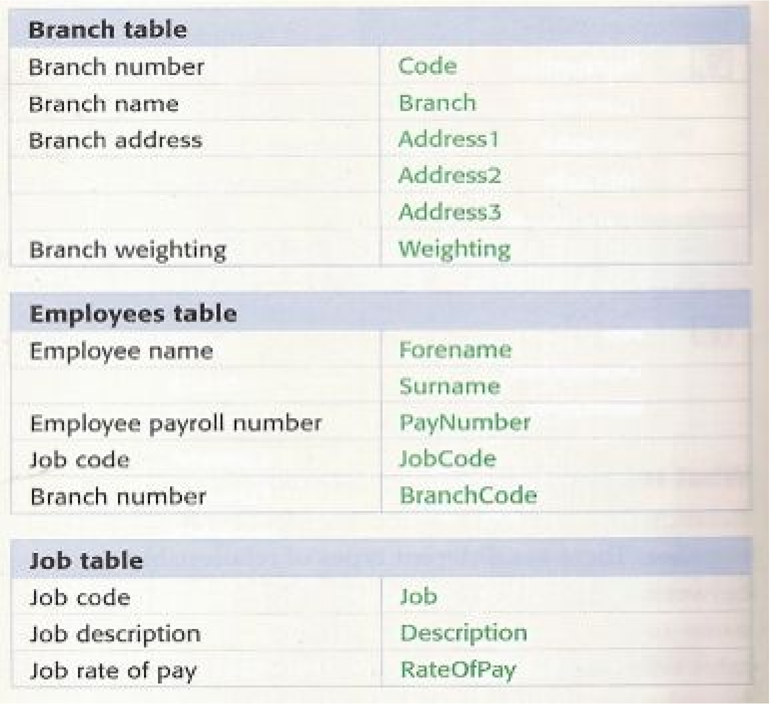
Employee name	Forename
	Surname
Employee payroll number	PayNumber
Job code	JobCode
Job description	JobDescription
Job rate of pay	RateOfPay
Branch number	BranchCode
Branch name	BranchName
Branch address	BranchAdd1
	BranchAdd2
	BranchAdd3
Branch weighting	Weighting

Hint

Field names may change if more than one table is used.

Do you need one or more tables?

Having selected the field names (these may change), you need to decide upon the number of tables to be used. This task could be completed using a single table. However, this will be very inefficient as some data items (e.g. the branch address, name, weighting, job description and rate of pay) will need to be stored for each employee, **leading to the repeated storage of some data**. There will also be other disadvantages to a single table. If a rate of pay for a particular job were changed it would need updating for every employee's record. The data that needs to be stored falls into three categories, data about the employee, the branch and the job. As each of these is related to the employee it is logical to store the data as three tables. This means that the field names will need to be set out like this:



Branch table	
Branch number	Code
Branch name	Branch
Branch address	Address1
	Address2
	Address3
Branch weighting	Weighting

Employees table	
Employee name	Forename
	Surname
Employee payroll number	PayNumber
Job code	JobCode
Branch number	BranchCode

Job table	
Job code	Job
Job description	Description
Job rate of pay	RateOfPay

Which fields should be primary key fields?

As you have chosen to use three tables for this task, you need three key fields. The primary key field for each table will be the field that <u>contains unique information</u> In the case of the Employees table, you were told in the task that the employee's **payroll number was unique to them'** so this will be an obvious choice for primary key field. For the job table, the **job code** field will contain unique values and will be suitable to use as a primary key for the branch table, the branch **code** will also contain unique values and should become the primary key field for this table.

What relationships will be needed?

Relationships are only needed if tables need to be linked together. There are different types of relationships between fields in different tables. The relationship can be:

a one-to-one (1—1) relationship, where each record in one table relates to only one record in another table;

or it can be a one-to-many (1—∞) relationship, where one record in one table can relate to many records in another table.

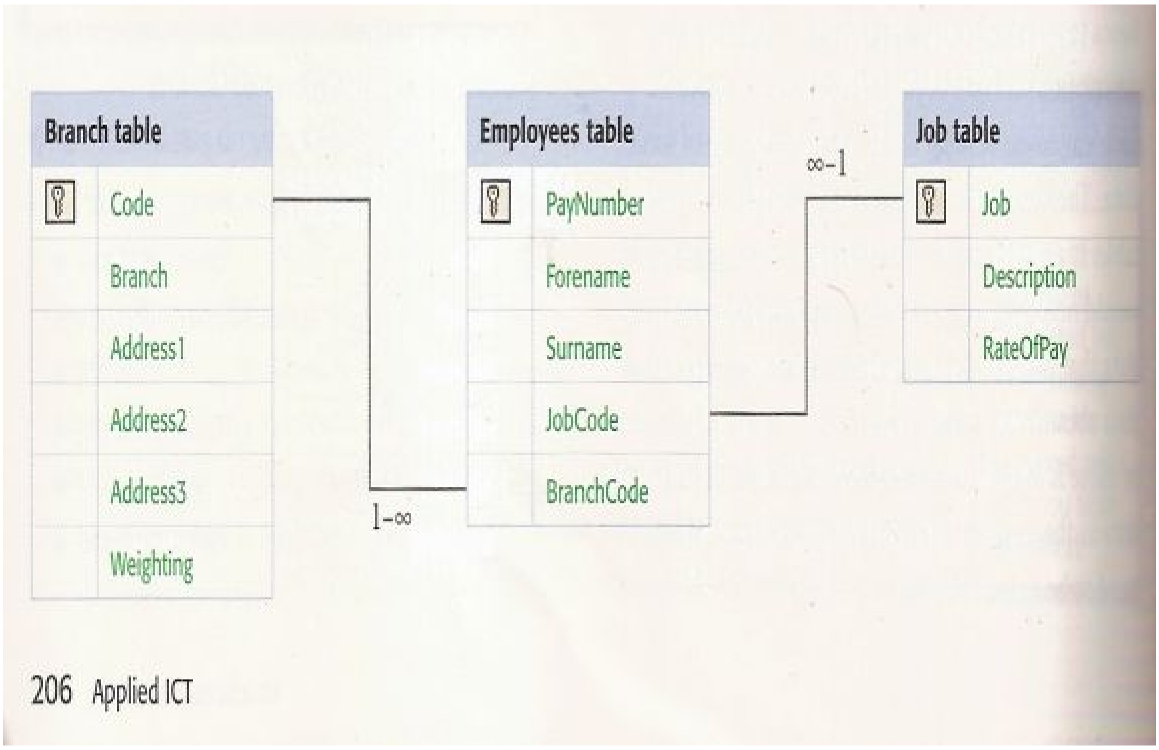
The relationship between the Employees table and the Job table will need to be created between the JobCode field in the Employees table and the Job field in the Job table.

This will need to be a one-to-many relationship as one record in the Job table could match many employees within the company <u>who have the same job</u>.

The relationship between the Employees table and the Branch table will need to be created between the BranchCode field in the Employees table and the Code field in the Branch table.

This will also need to be a one- to-many relationship as one record in the Branch table (which gives the details of one branch of the company) could match many employees within the company who work in that branch.

The relationships should be mapped onto a relationship diagram like this:



What data type should you use for each field?
There are a variety of field types that could be selected. Depending upon the package used they can have different names. There are three main types of field:

- **Alphanumeric**

Alphanumeric data can store alpha characters (text) or numeric data that will not be used for calculations.

- **Numeric**

A numeric field type (as the name suggests) is used to store numeric values that may be used for calculations. This does not include numeric data like telephone numbers, which should be stored in an alphanumeric field type. There are different types of numeric field including:

1. Integer (and in some packages long integer) fields, which will store numeric data where only whole numbers are required.

2. Decimal formats, which will allow a large number of decimal places,
3. Currency values, which allow currency formatting to be added to the display.

- **Boolean**

A Boolean (sometimes called logical) field type used where a Yes/No (or True/False, 0/1) response required.

There are other field types as well, like memo, date and time. These may not be available in all packages.

In order to select the correct field type you will have to look carefully at the data which will be stored. For the Employees table, the payroll number contains ha characters as well as numeric characters so must be set as an alphanumeric field. In Access this has been called "text".

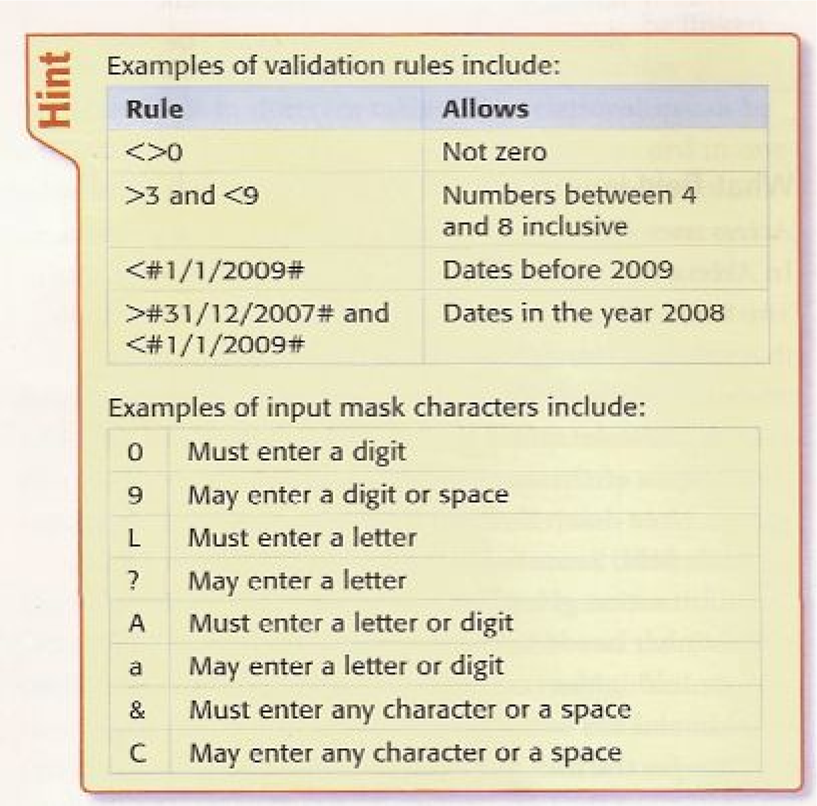
What field length should be used for each field?

Access uses a ‘text’ field type to hold alphanumeric data. In Access, only text fields require their lengths to be set. The field will be set with the shortest possible length, so that valuable storage capacity is not wasted. The most efficient method of calculating the length of any field is to look at the data, find the longest data item and count the number of characters. This count must include spaces.

Note down the field lengths that you will require for each field. Some field lengths can be obtained from the information given, for example in Task 11.1 the PayNumber needs to be set to six characters in length, as you are told in the task that the payroll number **‘is unique to them and has two letters followed by four digits**: If the data for the JobDescription field is analysed for this company and the longest data item is ‘Systems Analyst and Programmer the field length would need to be set to 30 characters. This was found by counting all the characters (including the spaces).

Which validation rules will help reduce data entry errors?

Look at each field and see if there are validation rules that could be applied to reduce data entry errors. These validation rules could include a range check, which checks that data fits within a specified range. A good example of this will be the weighting field in the branch table. The weighting must be between 0 and 2 (but cannot be equal to either of these values) so a rule could be applied to this field such that 0 <weighting < 2. A length check may be used to make sure that a field has the correct number of characters, for example the PayNumber field in the Employees table must be two letters, followed by four digits. This could be done by setting a validation rule to check that six characters only are entered and that the first character is M or C. The rule for this will be **‘Like “M?????”’or ‘Like “C?????”** In order to restrict the last four characters to numbers an input mask can be used; this will be **“LL0000’** See Section 11.3 for further details. Some fields like Forename and Surname cannot be validated as there are too many possible names.



Hint

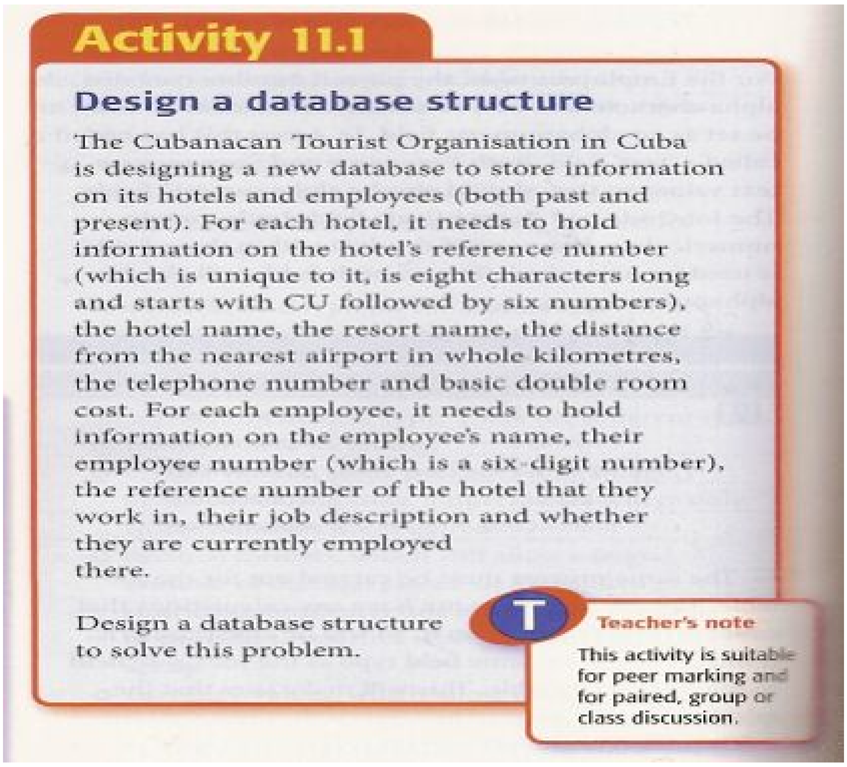
Examples of validation rules include:

Rule	Allows
<>0	Not zero
>3 and <9	Numbers between 4 and 8 inclusive
<#1/1/2009#	Dates before 2009
>#31/12/2007# and <#1/1/2009#	Dates in the year 2008

Examples of input mask characters include:

0	Must enter a digit
9	May enter a digit or space
L	Must enter a letter
?	May enter a letter
A	Must enter a letter or digit
a	May enter a letter or digit
&	Must enter any character or a space
C	May enter any character or a space

When you have the answers to these questions, you can start to create the database structure.`



Activity 11.1

Design a database structure

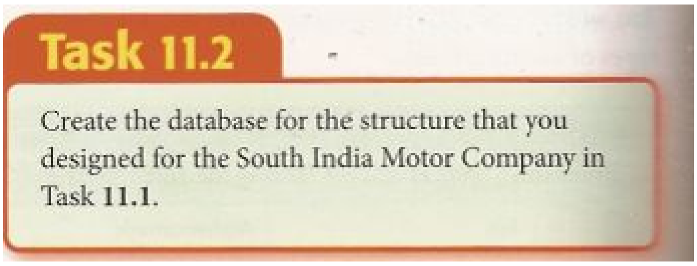
The Cubanacan Tourist Organisation in Cuba is designing a new database to store information on its hotels and employees (both past and present). For each hotel, it needs to hold information on the hotel's reference number (which is unique to it, is eight characters long and starts with CU followed by six numbers), the hotel name, the resort name, the distance from the nearest airport in whole kilometres, the telephone number and basic double room cost. For each employee, it needs to hold information on the employee's name, their employee number (which is a six-digit number), the reference number of the hotel that they work in, their job description and whether they are currently employed there.

Design a database structure to solve this problem.

Teacher's note

This activity is suitable for peer marking and for paired, group or class discussion.

11.3 Creating a database structure create a new database:



Task 11.2

Create the database for the structure that you designed for the South India Motor Company in Task 11.1.

1 Select the File menu, then New.
2 Select Blank Database from the New File window pane.
3 Enter a filename like SIMC (short for South India Motor Company), then click on [create]

Activities and tasks pages 212 to 224 from your book

/

11.9 Outputting data in Access:

You started to look at the structure of Access reports in Section 11.8 when you added a calculated control to a report. However, we did not cover the structures of these reports in depth. This section will give more detail to this area.

Access report structure:

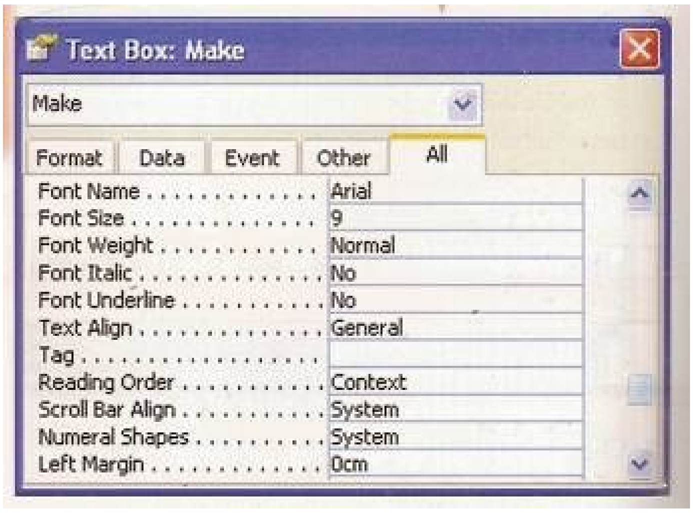
When you look at the design view of an Access report, you can see that it has five sections:
Report Header, Page Header, Detail, Page Footer and Report Footer.

Some information needs to appear on every page (e.g. field names) and other information is only required at the start or end of the entire document.

only once at the end of the document (like the calculated control in Section 11.8) are placed in the Report Footer.

The Detail section of the report is used for the individual records within the report. Controls can be moved from one section of the report to another by dragging and dropping them. It is sensible to do this by moving the control and its label at the same time. These are selected together by holding down the <Shift> key whilst selecting the second item.

The features of each calculated control can be changed using the properties window. Many of these features will be required for the practical examinations, including changing many font features like the font style, enhancements, size and alignment. These are amended here.



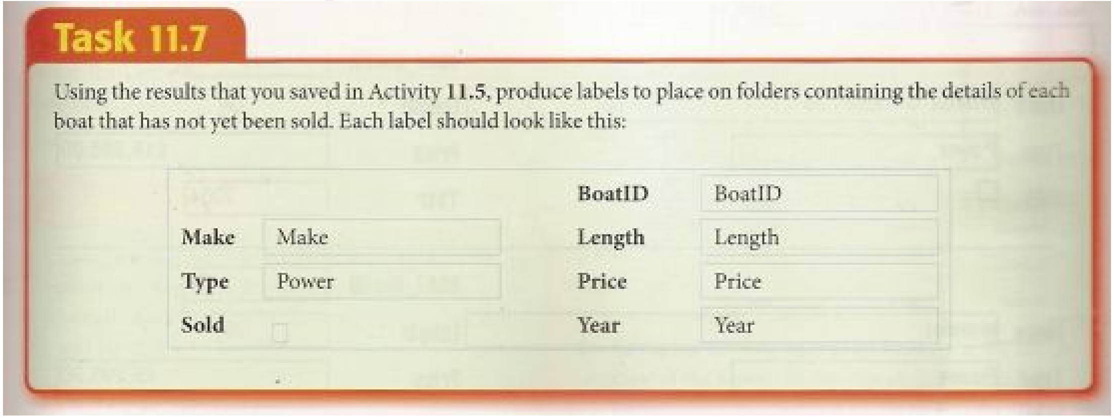
Make

Font Name	Arial
Font Size	9
Font Weight	Normal
Font Italic	No
Font Underline	No
Text Align	General
Tag	
Reading Order	Context
Scroll Bar Align	System
Numeral Shapes	System
Left Margin	0cm

The properties window can also be used to hide objects. Setting the Visible property to No will make the control invisible.

Another method of hiding objects is to change either the font color or the background color of the control so that they match each other.

Produce alternative output formats, like labels and tickets:



Task 11.7

Using the results that you saved in Activity 11.5, produce labels to place on folders containing the details of each boat that has not yet been sold. Each label should look like this:

		BoatID	BoatID
Make	Make	Length	Length
Type	Power	Price	Price
Sold	□	Year	Year

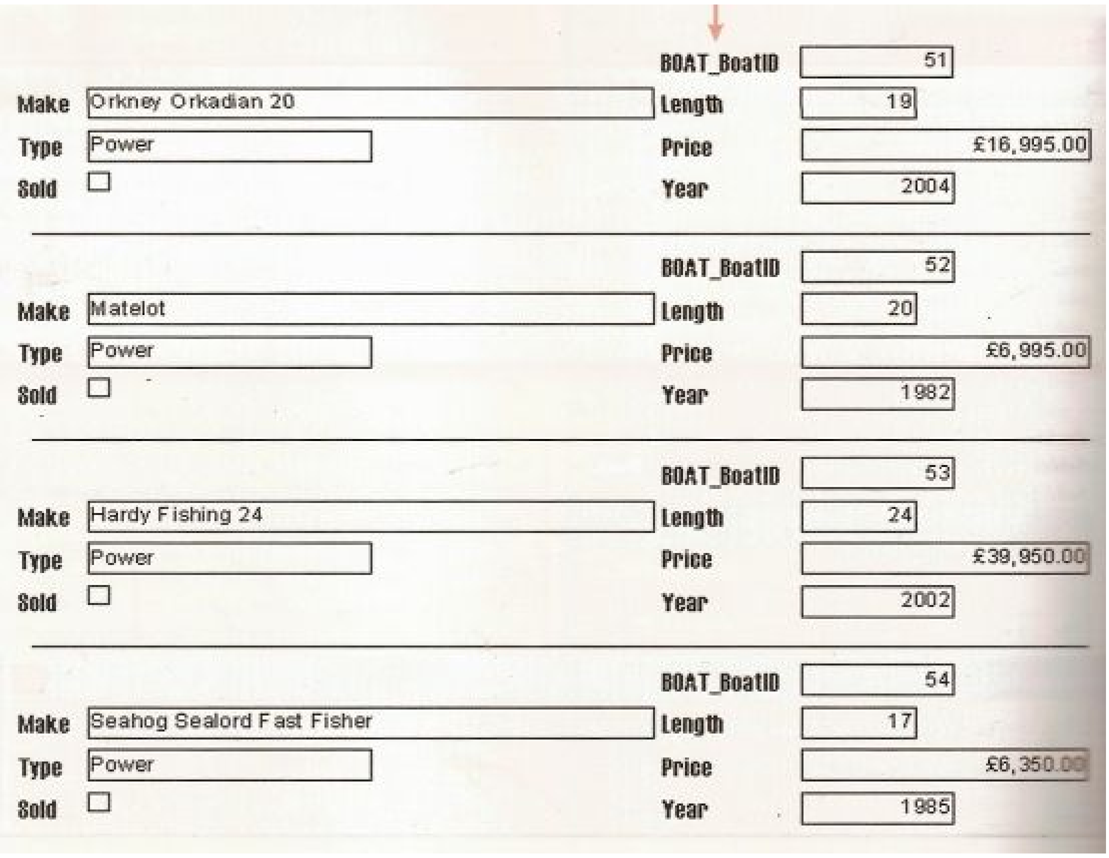
1. Perform a new query (as shown in Section 11.6).

2 Select all the fields from the Boat table. In the Boat.Sold field, set the search criteria to False.

3 Create a new report using the report wizard and select a Columnar Layout.

4 Enter the Design View of the report and delete the report header and footer and all of the controls within them. To do this, select the View menu, followed by Report Header/Footer.

5 This will require some resizing and movement of the controls as well as changing the size of the controls using their handles. The first few records of the finished result will look like this.

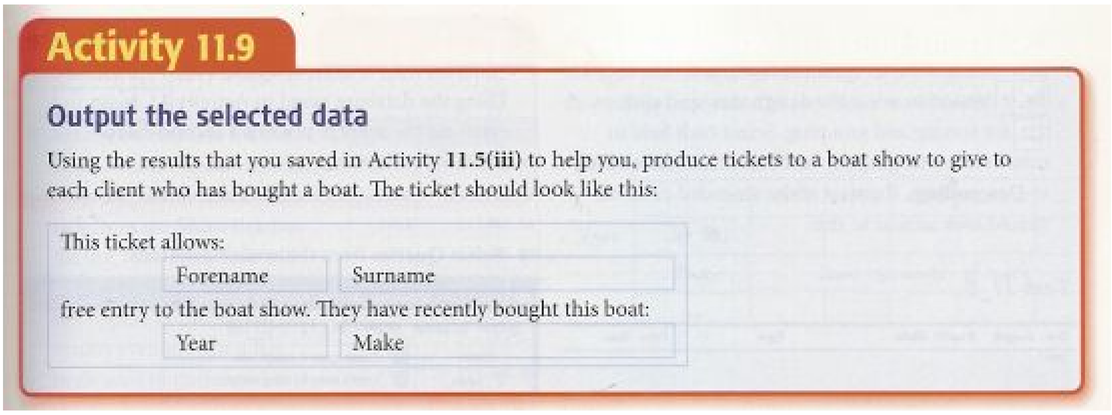


		BOAT_BoatID	51
Make	Orkney Orkadian 20	Length	19
Type	Power	Price	£16,995.00
Sold	□	Year	2004

		BOAT_BoatID	52
Make	Matelot	Length	20
Type	Power	Price	£6,995.00
Sold	□	Year	1982

		BOAT_BoatID	53
Make	Hardy Fishing 24	Length	24
Type	Power	Price	£39,950.00
Sold	□	Year	2002

		BOAT_BoatID	54
Make	Seahog Sealord Fast Fisher	Length	17
Type	Power	Price	£6,350.00
Sold	□	Year	1985



Activity 11.9

Output the selected data

Using the results that you saved in Activity **11.5(iii)** to help you, produce tickets to a boat show to give to each client who has bought a boat. The ticket should look like this:

This ticket allows:

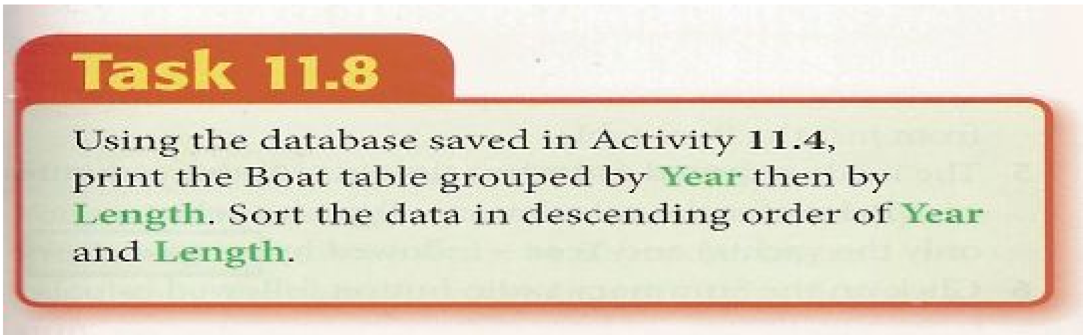
Forename	Surname

free entry to the boat show. They have recently bought this boat:

Year	Make

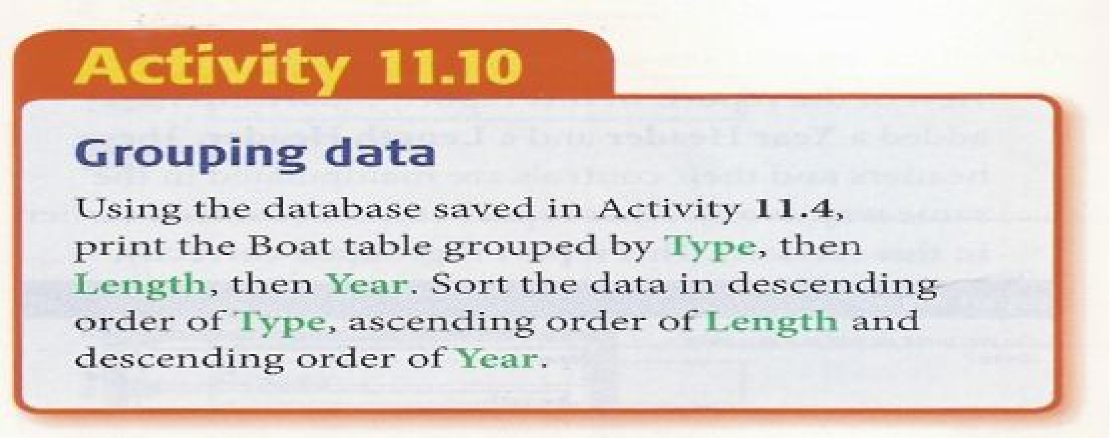
Produce a grouped report using the wizard

Access has the facility to group data. This is a process of collecting groups of similar data together for display purposes. Each group has the group header displayed only once, rather than viewing multiple occurrences of the same data item.



Task 11.8

Using the database saved in Activity **11.4**, print the Boat table grouped by Year then by Length. Sort the data in descending order of Year and Length.



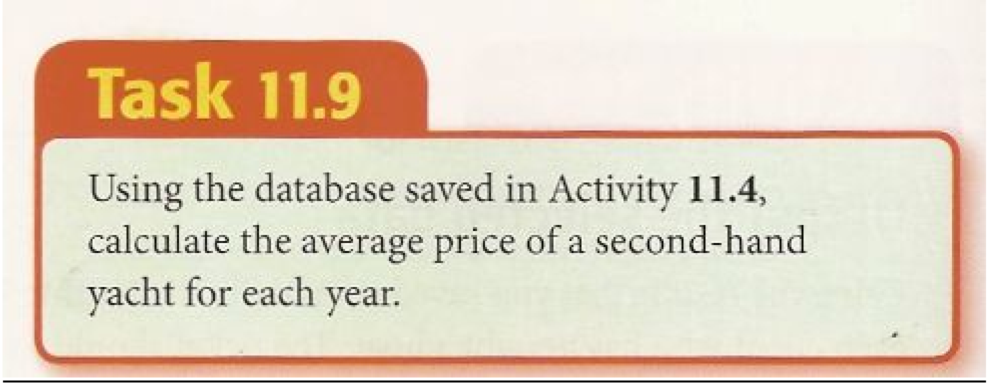
Activity 11.10

Grouping data

Using the database saved in Activity **11.4**, print the Boat table grouped by Type, then Length, then Year. Sort the data in descending order of Type, ascending order of Length and descending order of Year.

Summarize data using the simple query wizard

You can use a simple query to make it easier to analyze the data. This is achieved by using simple functions to perform calculations on the data. <u>A simple query can calculate a sum, average, minimum or maximum and count the number of records for a selected field or fields within a table or query.</u>



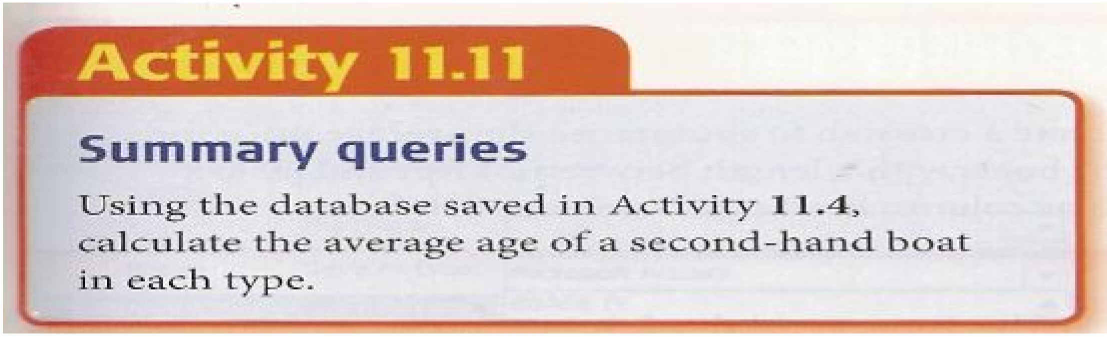
Task 11.9

Using the database saved in Activity **11.4**, calculate the average price of a second-hand yacht for each year.

1 Select Queries from the available options. (Database objects).
2. Select create query by using wizard
3 Select the table (or other query) that you wish to use for the search. In this example, the data will come from just the Boat table.
4 The fields required must be selected — in this case Price (for the calculation), Type (to select only the yachts) and Year — followed by Next.
5 Click on the Summary radio button followed by Summary Option.

6 The Summary Options window is where the calculations are performed. Click on the tick box for the function that you require for each field. In this case you want the average price
.
7 When you have selected all of the required options. Click on OK, then Next.
8 Name the query, ensure that the radio button for Modify the query design is chosen, then click on finish.

9 Set the criteria line for the Type field to "Yacht" to select only the yachts. Remove the tick from the Show tick box for the Type field.



Activity 11.11

Summary queries

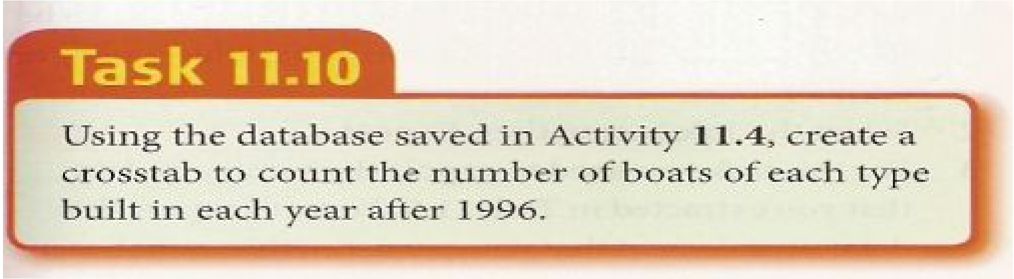
Using the database saved in Activity 11.4, calculate the average age of a second-hand boat in each type.

Summarise data using crosstab queries

You use a crosstab query to make it easier to analyse the data in two dimensions, by restructuring and calculating the data. A crosstab query can calculate a sum, average, count or other type of total for data that is grouped by two types of information.

This produces a two-dimensional grid which has one type of information down the left-hand side of a datasheet and another across the top, with summary data between.

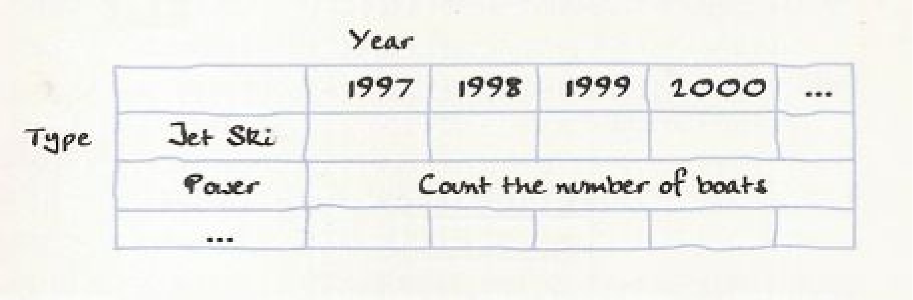
Before starting to attempt any question using a crosstab query it is important to identify the row headings, column headings and data to be summarized.



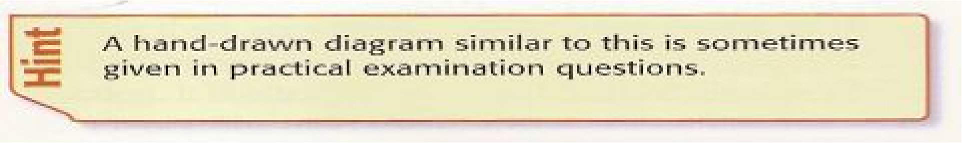
Task 11.10

Using the database saved in Activity 11.4, create a crosstab to count the number of boats of each type built in each year after 1996.

Before you start using the computer, it is often easier to sketch a planning diagram for the layout of the crosstab. In this task you are asked to produce a result similar to this:



		Year				
		1997	1998	1999	2000	...
Type	Jet Ski					
	Power	Count the number of boats				
	...					



Hint

A hand-drawn diagram similar to this is sometimes given in practical examination questions.

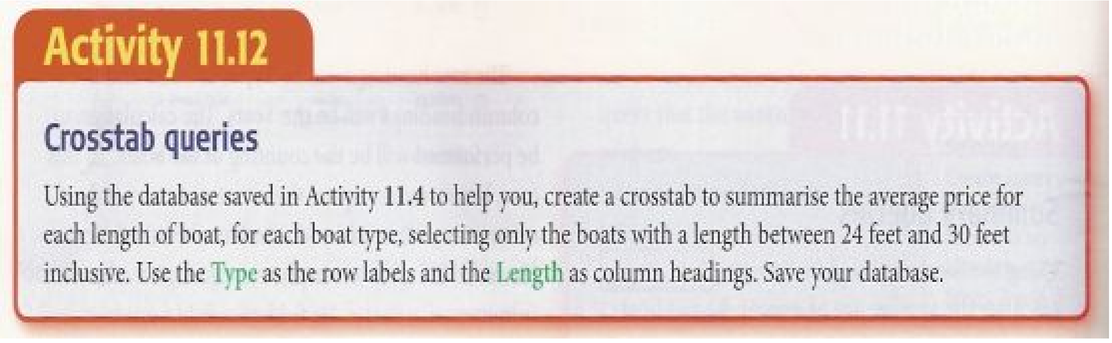
The row headings will be Types of boats
And the column headings will be the Years. The calculation to be performed will be the counting of the boats.

Summarise data using the crosstab query wizard

1 Select Queries from the available options.
2 Click on New to allow you to select the type of query that the wizard will perform.
3 Select the Crosstab Query Wizard then OK
4 Select the source from which to extract the fields for the crosstab query. In this case all of the data will be extracted from the Boat table, so this table should be selected followed by Next>
5 Select the field to be used for the row headings (use the planning diagram that you sketched to help you). In this case it is Type. Then click on Next>
6 Using the planning diagram select the field for the column headings, in this case Year, then click on Next>

7 Select the field to be used for the calculation. In this case it does not matter which field is selected as you are going to count the number of items.

8 Select Count from the list of available functions. In this case you do not need a subtotal for each row, so the tick box yes, include row sums can be left blank. Then click on [Next]



Activity 11.12

Crosstab queries

Using the database saved in Activity 11.4 to help you, create a crosstab to summarise the average price for each length of boat, for each boat type, selecting only the boats with a length between 24 feet and 30 feet inclusive. Use the Type as the row labels and the Length as column headings. Save your database.

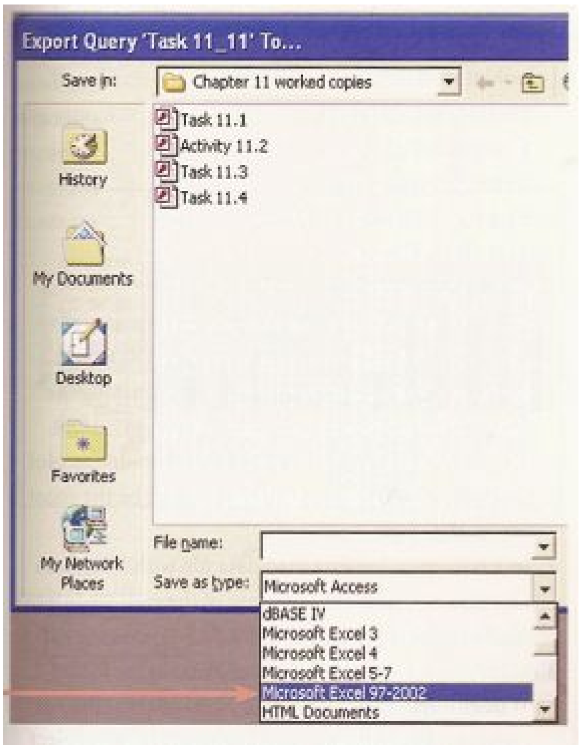
11.10 Exporting data into different packages

Before you start the process you must decide which format you want to use to export the data. This will depend on the software packages that you have. Many people who have a standard installation of the Microsoft Office Professional package do not have the facilities to produce graphs and charts within Access. The solution to this is to export the data in a format that can be read by Microsoft Excel.

The exact choice of file types for the export will depend on the export filters that have been installed on your machine when the software was installed. Changes to these settings are beyond the scope of this course, but you must be flexible enough to be able to export into a range of formats that could be used by the charting package. If you have selected Excel as your charting package, a suitable export format for the data would be an Excel file. There may be a number of these to choose from. Always select the latest version where possible, but if these are not installed, then saving the data as a text file would provide you with an alternative that can be opened in most software packages.

Export data into Excel

1 From the main database window select the data to be exported. This maybe in the form of a table, query even a report.
2 Select the File menu, then Export....



Export Query 'Task 11_11' To...

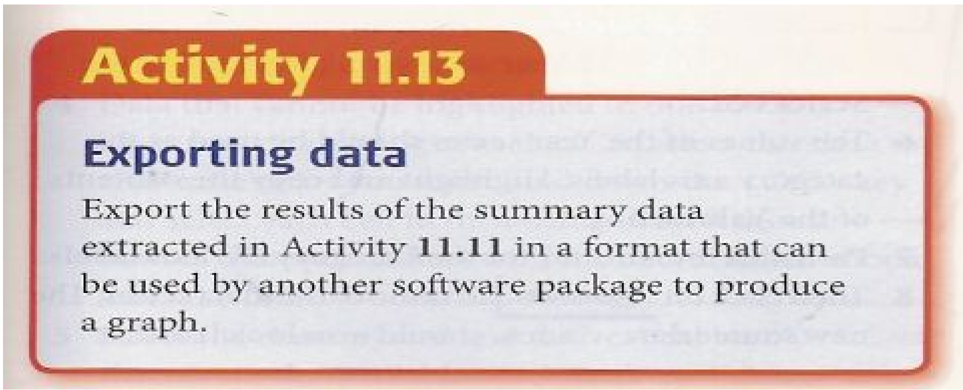
Save in: Chapter 11 worked copies

Task 11.1
Activity 11.2
Task 11.3
Task 11.4

File name:

Save as type: Microsoft Access

dBASE IV
Microsoft Excel 3
Microsoft Excel 4
Microsoft Excel 5-7
Microsoft Excel 97-2002
HTML Documents



Activity 11.13

Exporting data

Export the results of the summary data extracted in Activity 11.11 in a format that can be used by another software package to produce a graph.

Publish a report from Access in Microsoft Word:

Reports in Access can be exported using the Publish to Word option from the Database toolbar. This exports the file to your storage area in rich text format (.rtf), then opens Word. It automatically imports the file into Word ready for you to use.

11.11 Producing graphs and charts in Excel

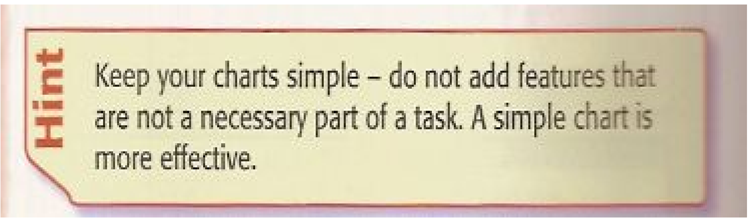
Selecting the most appropriate type of chart for any application is perhaps the most difficult area for many students. When you open any computer based charting package there are a number of different options, many of which appear different but perform the same basic function. It is often better to use a limited number of chart types. The most useful are:

- **Pie charts**: used to compare parts of a whole; these are often used to show comparisons using percentage values; pie charts are most effective if there are a small number of segments

- **bar charts**: used to emphasise the contrast between different quantities; these can be vertical (called 'column charts' in Excel) or horizontal; vertical bar charts should be used where time is to be plotted (this includes plotting months or comparing years)

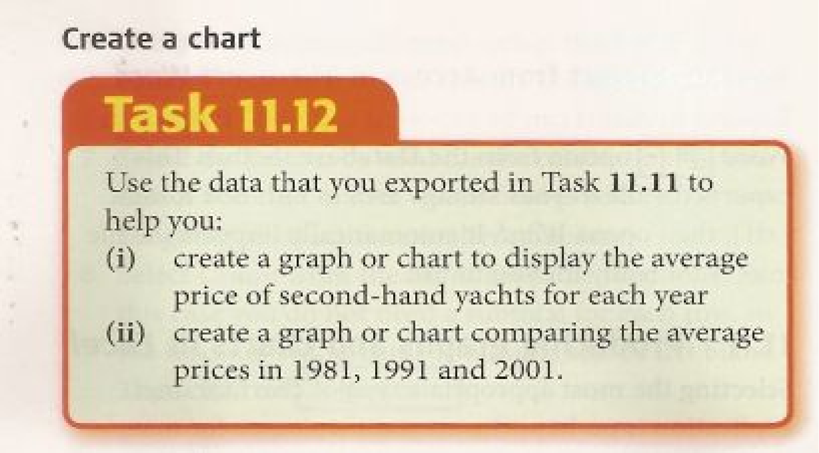
- **Line graphs:** used to show the relationship between one variable and another; they can be created using straight lines or curves (often to plot trends between the variables).

Other types of chart are used to compare two or more sets of data; these include comparative bar charts and comparative graphs with more than one type (e.g. a combination of bar chart and line graph).



Hint

Keep your charts simple – do not add features that are not a necessary part of a task. A simple chart is more effective.



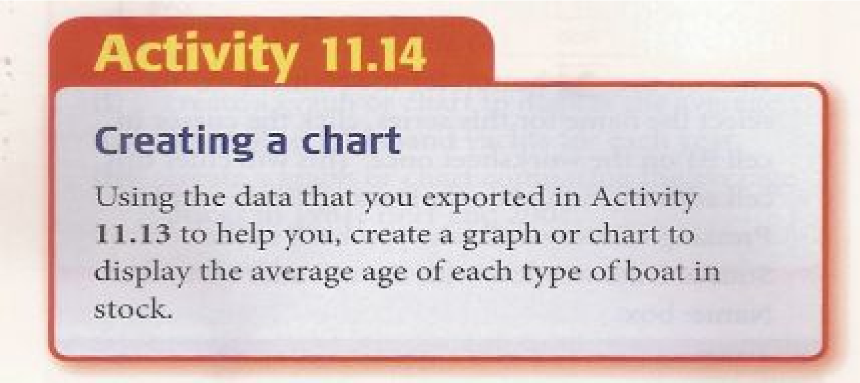
Create a chart

Task 11.12

Use the data that you exported in Task **11.11** to help you:

(i) create a graph or chart to display the average price of second-hand yachts for each year

(ii) create a graph or chart comparing the average prices in 1981, 1991 and 2001.

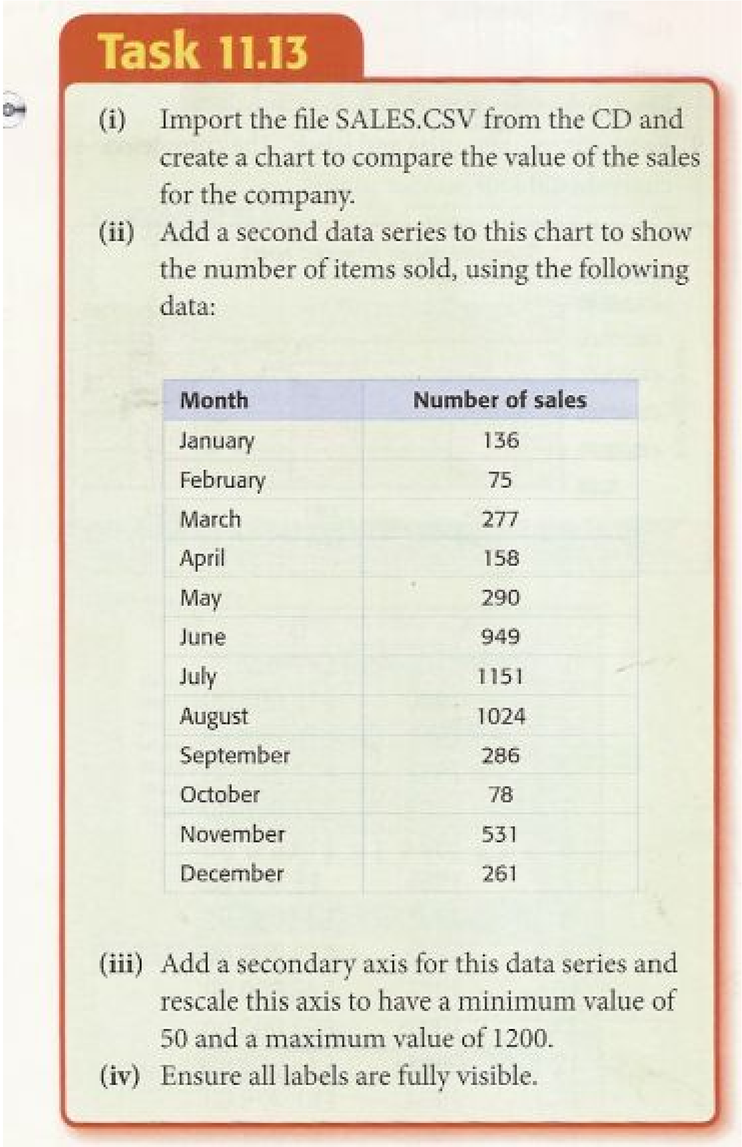


Activity 11.14

Creating a chart

Using the data that you exported in Activity **11.13** to help you, create a graph or chart to display the average age of each type of boat in stock.

Add data to an existing graph or chart:



Task 11.13

(i) Import the file SALES.CSV from the CD and create a chart to compare the value of the sales for the company.

(ii) Add a second data series to this chart to show the number of items sold, using the following data:

Month	Number of sales
January	136
February	75
March	277
April	158
May	290
June	949
July	1151
August	1024
September	286
October	78
November	531
December	261

(iii) Add a secondary axis for this data series and rescale this axis to have a minimum value of 50 and a maximum value of 1200.

(iv) Ensure all labels are fully visible.

The steps from your book

DONE BY:
IT INSTRUCTOR
AHMAD HIRZALLH
ABDUL HAMID SHARAF SCHOOL