5.4 Databases

A **database** is a collection of information which is structured in some way to permit manipulation and searching of data.

Why are databases used?

- They promote data consistency. When data is updated on a database it is up to date for *any* application which uses the database.
- Data duplication is reduced to a minimum since only one copy of each data item needs to be kept.
- It is relatively easy to expand the database if some new application is being considered.

- Security of data is easier to monitor and maintain. Data access can be controlled by database **front ends**; the actual database will be 'invisible' to all users except the database administrator (see Figure 5.3).

Early databases were examples of flat file structures, as described in Section 5.3 and illustrated by the COMPANY EMPLOYEE FILE. The way the data is organised makes it difficult to search for a specific piece of information or to create **reports** which only contain certain information (fields) from each of the records.

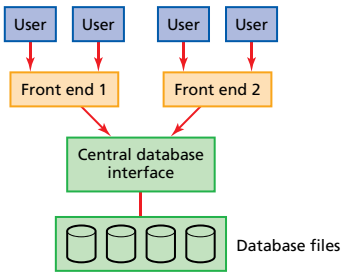

Figure 5.3 User access to data contained in a database

Relational databases

Relational databases were first introduced in 1970 following the work of F. F. Codd, a researcher at IBM.

Relational databases consist of a number of separate **tables** which are related (a table is made up of rows and columns in much the same way as a spreadsheet is structured). Each table contains a primary (key) field that is also a field in at least one other table. It is possible to combine data from different tables to produce a report which only contains the required information.

Relational databases do not need to repeat data, which is one of the problems of flat file structure (in the example that follows, three flat files would be needed containing repeated fields of key data since there would be no links connecting each file). Information is stored in separate tables only connected by the primary (key) field. Other advantages of relational databases include:

- faster data retrieval (because of links between tables)
- easy expansion of the database by adding extra data or new tables
- the need to change data in only one table – all other references to this data will then also be up to date, resulting in what is known as **data integrity**.

Tables can also contain **foreign keys** that relate tables in the database to one another. A foreign key in one table is a primary key in another.

We will now look at an example that shows the structure of tables in a relational database. In this example, there are three connected tables. It is important to note that in commercial databases there will be several tables connected together. The examples shown here contain only two or three and are being used to show the principle of relational databases.

Example

A garage sells cars and keeps a database of sales, customers and servicing, as shown in Figure 5.4 overleaf.

The primary (key) field is **Car number** and the column is shaded red).

The foreign key fields are **Invoice number** and **Engine ID** and these are shaded green.

Information from all three tables is linked together so, for example, if a car service was due then by typing in **Car number** the customer's details and servicing details are brought up on the screen. This means information can be sent to the customer as a reminder. Once the service is carried out, the servicing table will be updated which means all other references to it will also be up to date.

To help you understand these complex structures, go through the following exercise, which contains a flat data structure and a relational database.

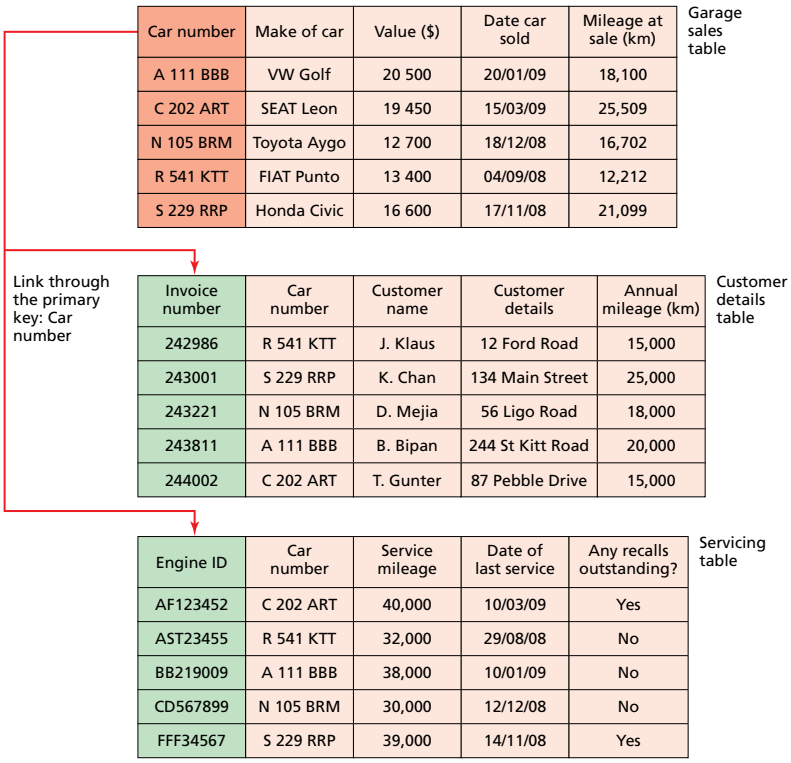

Figure 5.4 Three related tables in a relational database for a garage

Exercise 5a

Example 1
Table 5.6 shows a flat data structure.

Employee number	Name of employee	Date joined	Salary ($)	Department	Telephone number
A6121	Mr J. Bloggs	30/01/2000	18,000	Sales	151 216 009
B4142	Ms N. Kahn	19/02/2001	25,000	Accounts	153 423 111
B5041	Ms R. Spacek	04/11/2001	19,000	Sales	155 119 110
A3046	Mr K. Silva	15/12/2003	40,000	Legal	148 222 333
A5211	Mr N. Choudry	01/07/2004	25,000	Accounts	130 115 100

Table 5.6 Flat file structure for data on employees in a company

a How many records are there in this section of the database?
b How many fields are there in each record?
c Which field contains:
  i numeric data only
  ii text data only
  iii alphanumeric data only?
d What field is the primary key?
e In which field has the database been sorted?
f If the database was sorted in descending order on salary, using the Employee Number only, what would be the new order of data in the sorted database?

Example 2
The database shown in Table 5.7 contains two linked tables. The database is being used to keep a record of which customers have borrowed CDs from the music lending library.

CD stock table

Barcode	CD title	CD Artist	Year released	Number of tracks
77779287727	Seasons End	Marillion	1989	10
99969313424	Kingdom of Rust	Doves	2009	9
24354273506	Let It Go	Nada Surf	2002	11
94639624829	Our Love to Admire	Interpol	2007	9
02498669105	The Invitation	Thirteen Senses	2004	12
45099625627	Seal	Seal	1994	10

Table 5.7 Tables of data relating to CDs borrowed from a library

Customer borrowing table

Customer number	Customer name	Telephone number	CD borrowed	Date due back
M10411	Mr K. Sahz	415 003 455	02498669105	15/10/2009
M21516	Mr D. Silva	841 133 222	77779287727	14/10/2009
F18113	Mr A. Adak	614 555 211	45099625627	14/10/2009
M20004	Mr R. Choudhury	416 888 210	24354273506	12/10/2009
F16117	Ms L. Smith	416 219 000	94639624829	11/10/2009
F50316	Mr M. Egodi	841 567 228	99969313424	10/10/2009

a How many records are there in the CD stock table?
b How many fields are there in each record in the Customer borrowing table?
c What type of database is being used here?
d What is the primary key in the CD stock table?
e Which field is a foreign key?
f What data type would you use in the Date due back field?
g What data types have been used in all the other fields?