Topic: Databases

A database is a way of storing *information* in a structured, logical way. They are used to collect and organize information such as customer details for a business, medical records at a surgery, or stock items at a warehouse.

Database content is easy to manage and information can be accessed and updated quickly and efficiently. A database can store and handle vast amounts of data. A user can sort and search a database to find any desired data.

Most databases have the following properties:

1. Tables

Data is stored in rows and columns (similar to a spreadsheet – the main difference is **HOW** the data is organized). Each row in a table is called a record which is made up of a number of fields (columns in the table). The data type in the fields is usually either text, numeric or date/time. Most databases contain a number of tables which are usually linked together in some way.

2. Records

A record is a collection of fields that contains data about a single object – it is a row within a table.

3. Fields

A field is a single category of data within a database, which appears in all the records of a table – it is a column within a table.

Key fields

A key field is used to identify the records within a database. There are two types of keys:

- Primary key;
- Secondary key.

Primary key

The Primary key is a unique field that identifies a single record. This prevents any records from having the same value.

Some 'natural' primary keys are:

- CarRegistrationNumber;
- ISBN – a 10-digit code that uniquely identifies a book;
- MAC number – a 6-part number that uniquely identifies a network card
- National Insurance Number – can uniquely identify employees of a company (not usable for under 16s or for non-British nationalists!)

Secondary key

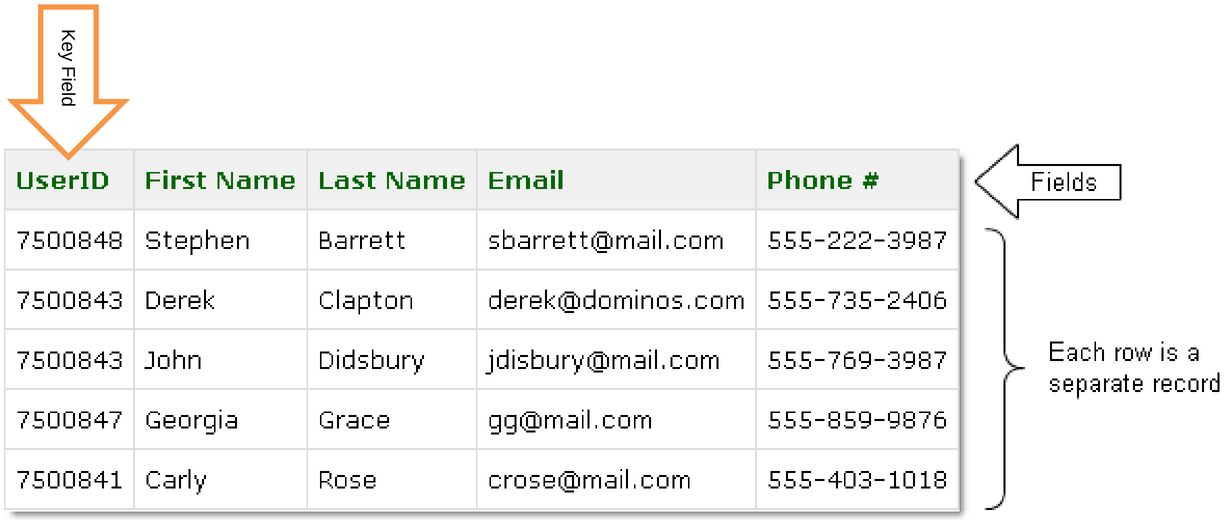
A Secondary key is a non-unique field, used in a search that does not always produce only one matching record.

Some typical secondary keys are:

- LastName;
- PostCode;
- DateOfBirth;

Topic: Databases

The following terms are used to describe parts of a database:


Key Field
Fields
Each row is a separate record

UserID	First Name	Last Name	Email	Phone #
7500848	Stephen	Barrett	sbarrett@mail.com	555-222-3987
7500843	Derek	Clapton	derek@dominos.com	555-735-2406
7500843	John	Didsbury	jdisbury@mail.com	555-769-3987
7500847	Georgia	Grace	gg@mail.com	555-859-9876
7500841	Carly	Rose	crose@mail.com	555-403-1018

Record structure

Before setting up a database, the record structure must be decided to make better use of the memory and backing store, and to make searching and report creation easier.
For example, a car showroom wants to record details of the cars it sells. Before setting up the database, the following attributes need to be decided:

- Field Name
- Field type
- Field size
- Format
- Input Mask
- Validation Rule

Topic: Databases

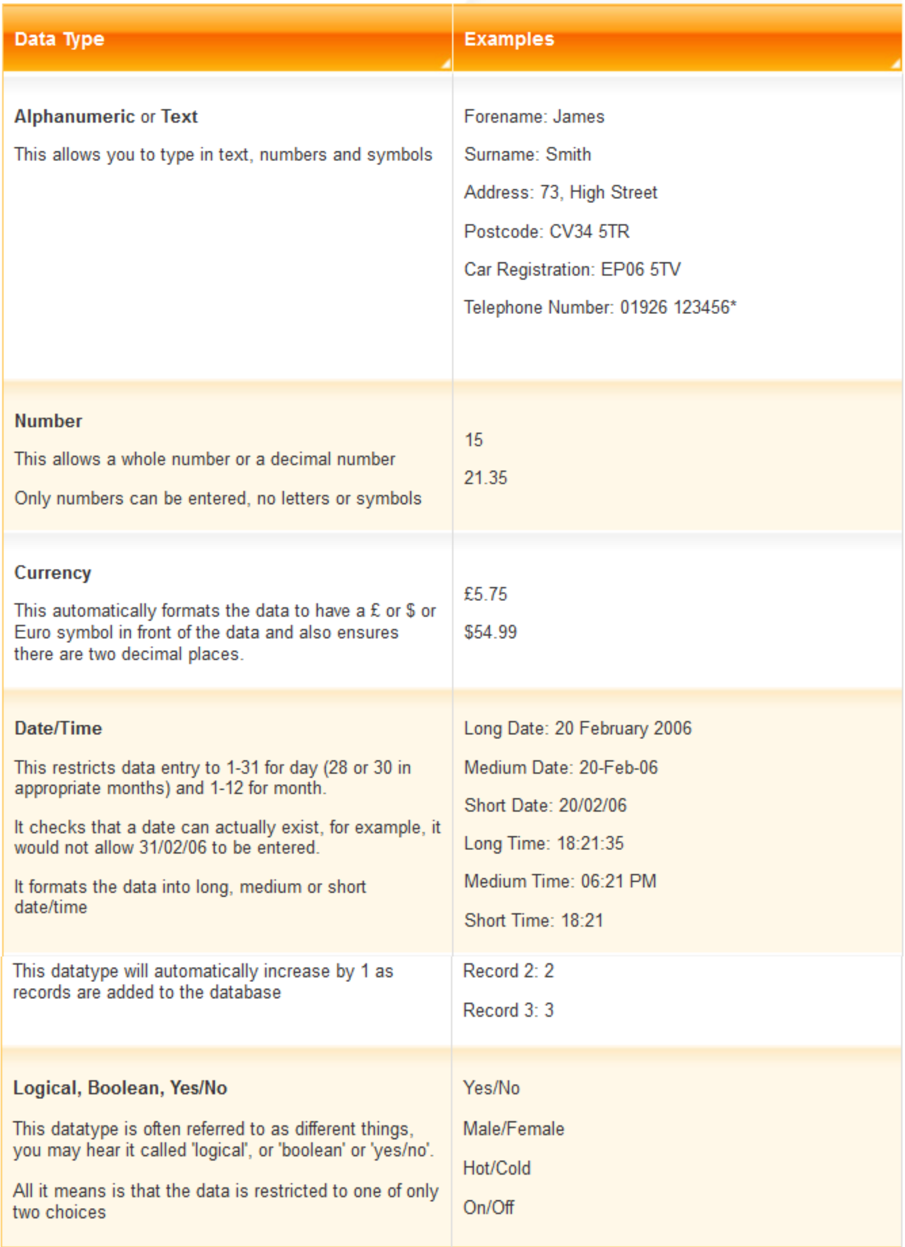
Record structure

Field name	Field type	Format
Registration number	Alphanumeric	Up to 7 characters - the key field
Make	Alphanumeric	Up to 15 characters
Model	Alphanumeric	Up to 15 characters
Date first registered	Date	DDMMYY
Price	Currency	Up to 5 numbers
Taxed	Yes/No (Boolean)	1 character Y/N
	Etc…	

When designing a database it is important to choose the correct field type. This ensures that the *data* stored is usable and it makes validation easier. For example, if the price paid for goods was stored in a text field, then the database wouldn't be able to add each individual figure to produce a total.

Topic: Databases

Following is the list of common data types:



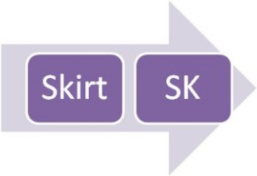
Data Type	Examples
Alphanumeric or **Text** This allows you to type in text, numbers and symbols	Forename: James Surname: Smith Address: 73, High Street Postcode: CV34 5TR Car Registration: EP06 5TV Telephone Number: 01926 123456*
Number This allows a whole number or a decimal number Only numbers can be entered, no letters or symbols	15 21.35
Currency This automatically formats the data to have a £ or $ or Euro symbol in front of the data and also ensures there are two decimal places.	£5.75 $54.99
Date/Time This restricts data entry to 1-31 for day (28 or 30 in appropriate months) and 1-12 for month. It checks that a date can actually exist, for example, it would not allow 31/02/06 to be entered. It formats the data into long, medium or short date/time	Long Date: 20 February 2006 Medium Date: 20-Feb-06 Short Date: 20/02/06 Long Time: 18:21:35 Medium Time: 06:21 PM Short Time: 18:21
This datatype will automatically increase by 1 as records are added to the database	Record 2: 2 Record 3: 3
Logical, Boolean, Yes/No This datatype is often referred to as different things, you may hear it called 'logical', or 'boolean' or 'yes/no'. All it means is that the data is restricted to one of only two choices	Yes/No Male/Female Hot/Cold On/Off

Topic: Databases

Coding of data:

Any system will need to have data collected, entered and stored.

One method of storing data is to **assign codes** to it. This usually means shortening the original data in an agreed manner. The agreement is between the users of the system. This coding scheme could be part of the training of how to use the system, and it could also be documented within the system for new users.



If the coding is completely obvious then there is no such need for formal documentation. For example if a field called 'Gender' has only two values 'M' and 'F'. It should be obvious from the field name that this refers to Male and Female.

Example 1
Original data: Monday; Tuesday; Wednesday; Thursday; Friday

Coded data: Mon; Tues; Wed; Thurs; Fri

Example 2
Original data: Extra Large; Large; Medium; Small

Coded data: XL; L; M; S

Advantages of coding	Disadvantages of coding
Data entry can be faster	Coarsening of data
Data entry can be more accurate	Meaning of data can be obscured
Validation can further improve accuracy	Value judgments are difficult to code
Less storage space required	If people don't know the code it can slow down data entry
Faster searching for data	If codes are complicated they might be entered incorrectly
Coded data can be more secure if people don't know what it means	Might run out of code combinations

Topic: Databases

Estimate the size of a file from its structure and the number of records

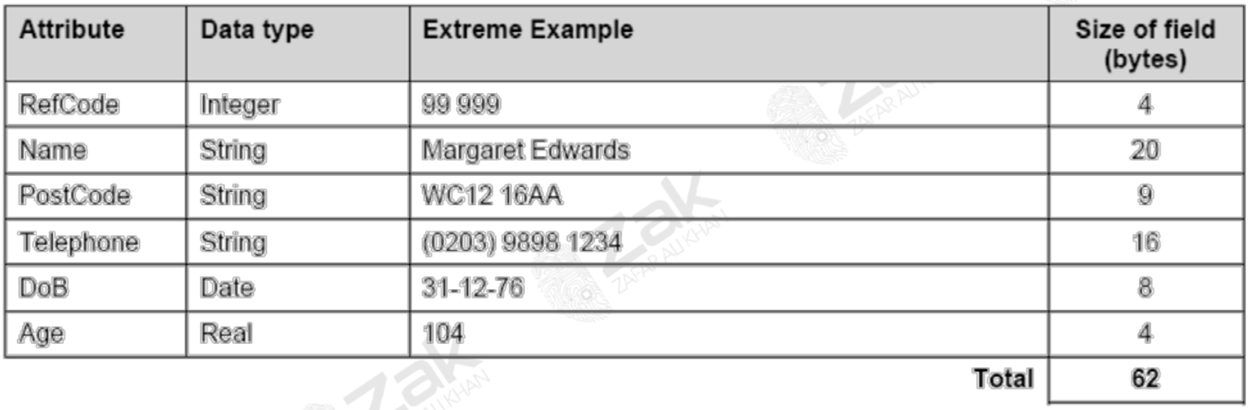
The basic formula for estimating the size of a file is:

$$\text{Size of file} = [\text{size of each record}] \times [\text{number of records}] + [\text{a little bit more!}]$$

If we consider a file with 200 records, which stores the details of an organisation's customers:

CUSTOMER(RefCode, Name, PostCode, Telephone, DoB, Age)

We can estimate the size of the record as follows:



Attribute	Data type	Extreme Example	Size of field (bytes)
RefCode	Integer	99 999	4
Name	String	Margaret Edwards	20
PostCode	String	WC12 16AA	9
Telephone	String	(0203) 9898 1234	16
DoB	Date	31-12-76	8
Age	Real	104	4
		Total	62

Thus 200 records would require:

$$\begin{aligned} 62 \times 200 &= 12400 \text{ bytes} \\ &= \frac{12400}{1024} \text{Kbytes} \\ &= 12.1 + 1.21\ (10\%) \\ &= \underline{\underline{13.3 \text{ Kbytes}}} \end{aligned}$$

Note that to determine the maximum field length, an extreme case was considered and several bytes added to play safe.

Topic: Databases

Database Management System:

We have discussed the structure of a database as consisting of one or more tables, each of which contains records and fields of various data types.

The next requirement is to have a system in place that can act upon that data as well as creating and maintaining the database itself.

This is the role of the 'database management system' usually referred to as a DBMS.

A DBMS is an application designed to control all aspects of a database.

The DBMS will have a command language. This includes command statements for:

- Creating an empty database
- Deleting an entire database
- Creating and deleting tables
- Inserting new records into tables
- Updating and deleting records
- Being able to extract data sets
- Being able to produce reports that summarizes the data
- Being able to process the data to create new data

There are many database management systems that are either commercial products or free open source applications.

Examples include

Name	Comment
MySQL	A very popular, free open source system, widely used on web sites
Access	Included in some versions of the Microsoft Office suite
Base	Part of the free Open Office suite
Oracle	A multi-user enterprise level database management system. Widely used in industry

Topic: Databases

Queries

Queries most commonly allow information to be retrieved from tables. Since the information is often spread across several tables, queries allow it to be viewed as one single *datasheet*. They also allow filtering so only the records required are seen. Queries can be either results seen directly on the screen or the output to another form or report. Example of a query: (house > 200 000) OR (no_of_rooms < 4).

Sample database/table from a database

Name of substance	Toxic	Smallest size (microns)	Largest size (microns)	Filtered out by 10 micron mesh?
cement dust	no	3	100	some
coal dust	no	1	100	some
common sand	no	10	2000	yes
paint pigments	yes	0.01	5	no
talcum powder	no	3	80	some
pollen	no	10	100	yes
smog	no	0.01	1	no
viruses	yes	0.002	0.04	no
bacteria	yes	0.2	20	some
human hair	no	20	300	yes

Points to note: (1) there are 10 records in this section of the database
(2) Each record has 4 fields
(3) Sample queries:
(Smallest size (microns) < 1) OR (Toxic = “yes”)
Would output records 4, 7, 8 and 9
(Largest size (microns) > 99) AND (Toxic = “no”)
Would output records 1, 2, 3, 6 and 10

The query should match up with the field titles to ensure a correct search is carried out. Search engines on the Internet work in a similar way; the clever part is how the information is stored on the databases so that the time to do the search (based on key words) and the effectiveness of the search itself results in a very powerful and very useful tool.