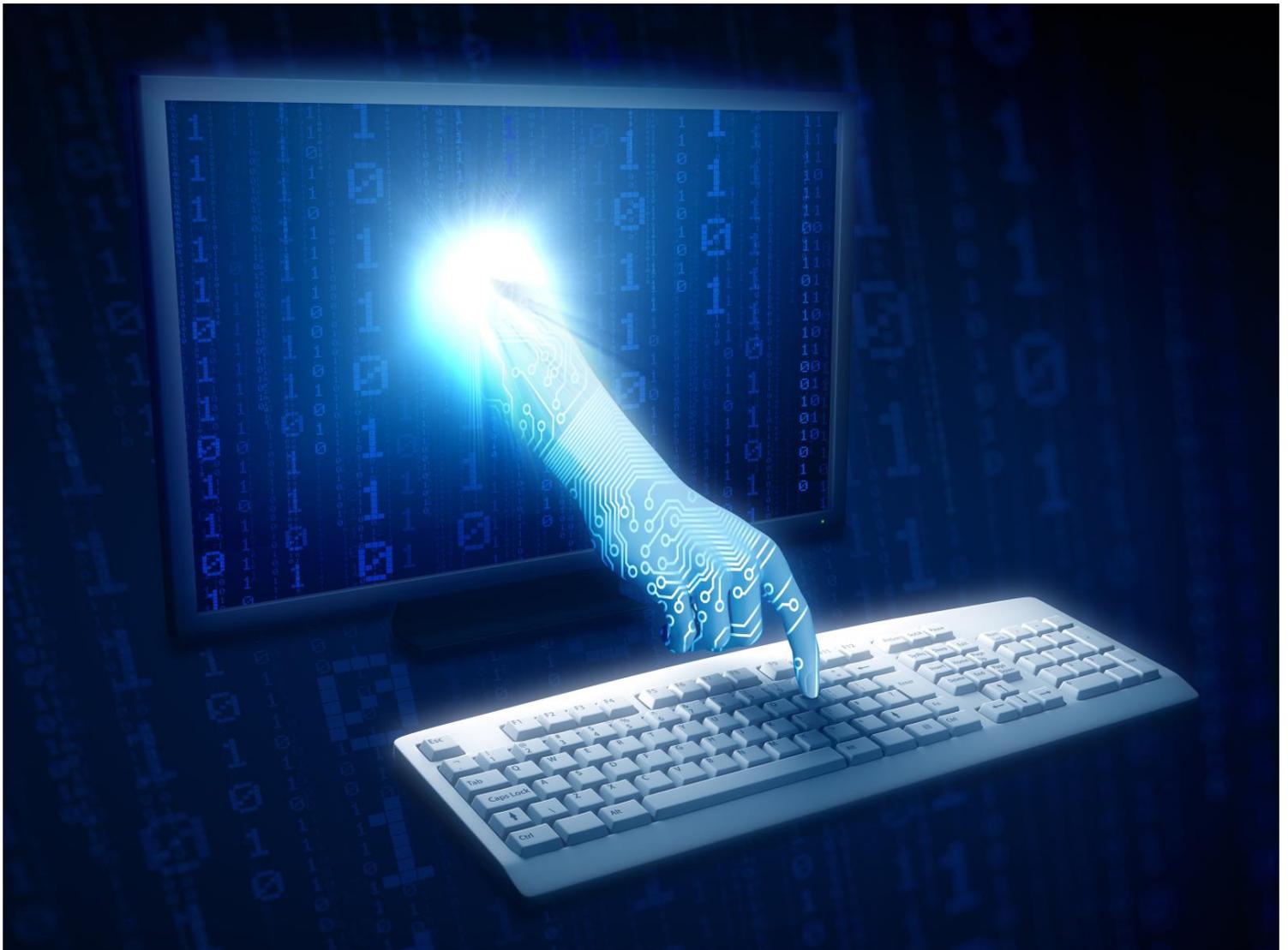


Database Theory



Information Rich & Poor

We live in an age where information and knowledge are so important that society can be divided up into two groups. These are “**information rich**” and “**information poor**”.

If you are **information rich** you have access to many TV and radio channels, books, newspapers, magazines, journals, and of course access to the World Wide Web.

Those who are **information poor** tend to not have access to the Web and probably find it difficult to access relevant books and journals. Even in general conversations a discussion about a TV programme shown on satellite TV will be lost on people who only have 4 or 5 free view channels.

To help us remember all the information that we need we use an **Information System**, which is a combination of hardware and software which is used to input, process and store data, and then output information.



What is an Information System? What is a Database?

An **information system** is a combination of computer hardware and software that is designed to **create, store, process** and **present** information. The amount of data stored within information systems can be vast and is continually increasing as the processing power and the storage capacity of computer systems increases.

e.g. : The Driver and Vehicle Licensing Agency (DVLA) **Information System** stores records on around 40 million drivers and 30 million vehicles in the UK.

A database is at the heart of all information systems and can be defined as an **organized collection of information or data**. Putting information into a computer database gives you far more flexibility in organising, displaying and printing that information.

Almost everyone who uses a computer will use a database system for something:

- Banks use a database package for storing customers' accounts information
- Your school will use a database package to store information on all the pupils in the school
- Police use databases to keep information on people with criminal records



Why do we use Databases?

Manual

To the right is a picture of a paper filing cabinet. Before computers were used to store data, it was stored on paper forms or cards in filing cabinets. These are still used today in offices but not as much as they used to be.



Disadvantages of manual databases:

- It takes a lot of room to store paper records.
- A record could be mis-filed or lost.
- Data may be duplicated in several records.
- It is time consuming to keep data up to date as records have to be changed manually.
- Data may be changed in one record but not in another, leading to inconsistency.
- It is time consuming to search through records to find what is required.
- Many people have to be employed to handle the data and keep it up to date.
- Only one person can use a record at any one time.



Advantages of manual databases:

- One does not have to purchase an expensive information system.
- If data is locked in a filing cabinet it is more secure than if it was stored on a computer.
- It is less likely that data will be altered by mistake.

Computerised

Computerised databases are now much more popular and widely used. There are many reasons for this. Paper filing cabinets were **big and bulky** and needed **a lot of organisation** to make sure everything was stored correctly. A computerised database doesn't take up lots of room and can be **easily edited** without having to re-print things.

A **Database Management System (DBMS)** is the name given to the program that is used to store the data.

Advantages of electronic databases:

- It is easier to search for data as the computer program does it for you.
- Data can be sorted into alphabetical or numerical order on any field or fields easily and quickly.
- More than one person can view the data at the same time.
- A lot of data can be stored in a very small space. One hard disc can hold as much data as several filing cabinets.
- It is easy to edit the data if it changes.
- Data can be extracted and made into a report. Reports are a useful way of displaying data in a form which is easy to read.



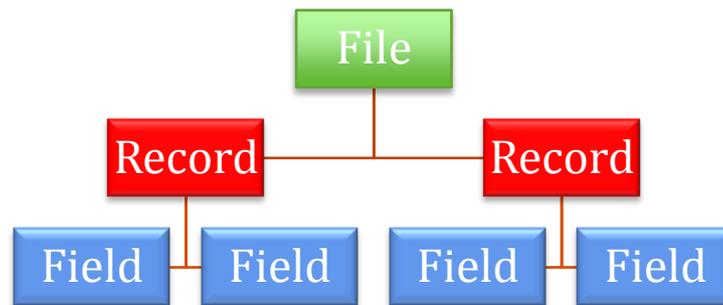
Disadvantages of electronic databases:

- Data is susceptible to unauthorised access and unauthorised modification.
- If there is a technical problem the data cannot be accessed.
- Data users have to be trained to use the programs.
- It can be expensive to buy the hardware and software to set up the database.
- People may require training in how to create and use a database.

How is a database organised?

There are three key terms you must know:

- The whole set of data in a database is called the **file**
- The set of information on one single subject (for example, one pupil in your school) is called a **record**
- Each separate heading of information inside a record (for example, your date of birth) is called a **field**



To create a database requires the user to define the file structure. This involves **creating fields** by enter each field name and its appropriate field type.

Data Modelling

Types of Data

In designing a database it is important to decide on the types of data that each field will contain. The types of data are:-

- text –consists of letters of the alphabet, mixture of letters and numbers or a number, e.g. phone number, which is not required for a calculation.
- integer –a whole number, i.e. it does not contain a decimal part.
- real number – a number which contains a decimal part.
- object –a picture, e.g. a photo or a fingerprint.
- date –various formats include 01/01/2005 or 1st Jan, 2005.
- time –a time e.g. a train departure time.
- currency –contains money.
- Boolean – yes or no. (Named after George Boole, a mathematician and logician who created binary logic, e.g. AND, OR, NOT)

Field size

It is important to calculate the size of the field. The field size is the number of characters that the field can contain and is calculated by counting the number of characters, including spaces, in the longest entry which the field will hold.

e.g. if a surname field can contain the entries Smith, MacDonald, Ray or Waterston the field length would be 11 as Waterston, the longest name has 11 characters.

It is important to specify the field length when designing the database for 2 reasons:

- In Access the default size for a text field is 50 characters. If you wish to enter more than 50 characters it will not be accepted.
- It saves on disc storage space if field sizes are kept to a minimum. For example, it is a waste of storage space to have a field size of 50 for a "title" field, if the longest entry is going to be "Miss".

In Access it is not necessary to calculate lengths for the other field types as they are set automatically

Database Programs



There are 2 types of database program:-

- Flat File Database
- Relational Database

Flat File Database

This database you created with the Smartphone information is a flat file database as it only has one table and each row of the table is unique.

A flat file system is one that contains all of the data in one file. Microsoft Works is an example of a flat file system.

This type of program is useful if you do not want to store a lot of data. You could use this for storing your friends' names and addresses for Christmas cards. The disadvantages of flat file systems are:

- There may be duplication of data, e.g. a person's name may be entered more than once.
- If data is duplicated it could lead to data inconsistency. For example, a typing error could lead to a person's name being entered as John in one record and Jon in another record.
- Insertion problem means that sometimes data has to be included more often than is necessary. For example, in a library system every time a member borrowed a book the librarian may have to enter their name and address.
- Deletion problem means that if a single record is deleted it may remove important data which is still required. For example, in the library if a member returned a book and the record was deleted it may remove all trace of the member's name and address.

These problems can be overcome by storing the data in a relational database. In a relational database data is only entered and stored once.

Relational Database

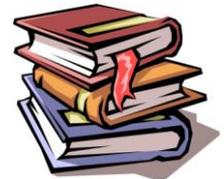
A relational database package allows you to create several files or tables and have them linked together. A table is another name for a file, i.e. a collection of records.

Microsoft Access is an example of a relational database.

Before 2 or more tables can be related they need to have a field in common. In the example shown each table has a field called "AccCode".

In a school database there could be a table for the pupils' personal information (name, address etc) and a separate one for the subjects they are studying

If the data is stored in separate tables it makes it easier to manage. For example, if a pupil changes their address only the personal details table will need to be altered. If a subject choice changes then only the subject table needs to be altered.



Example – consider a library database system

The data to be stored will be member surname, forename, address, town, postcode, subject, ISBN, date borrowed, date returned.

As there may be more than one member with the same name or living at the same address neither of these fields can be used as a key field. It is necessary to introduce a new field to hold the member ID number. (Surrogate key)

In a manual, paper-based, system the structure may look as follows:

Table 1

ID	surname	forename	address	town	post code	subject	ISBN	date borrowed	date returned
001	Bell	John	23 High Street	Arbroath	DD7 5FT	Fiction	00918 70364	12/4/05	19/4/05
						Fishing	00716 75439	19/4/05	
013	Smith	Emma	17 Main Street	Carnoustie	DD7 5VF	Fiction	00918 70364	20/4/05	
067	Roberts	Susan	The Elms	Arbroath	DD7 8HG	Autobiography	00634 89654	18/4/05	24/4/05

This system is not very efficient for the following reasons:

- One member has 2 books out on loan, recorded in the same record – this is called having multi-valued fields.
- Every time a member borrows a book the whole file will have to be loaded in and changed.

- If a member changes their address the whole file will require to be altered.

Computer databases do not allow more than one value to be stored in a field.

If the data was being stored in a flat-file database the structure would now look like this:

Table 2

ID	surname	forename	address	town	post code	subject	ISBN	date borrowed	date returned
001	Bell	John	23 High Street	Arbroath	DD7 5FT	Fiction	00918 70364	12/4/05	19/4/05
001	Beel	John	23 High Street	Arbroath	DD7 5FT	Fishing	00716 75439	19/4/05	
013	Smith	Emma	17 Main Street	Carnoustie	DD7 5VF	Fiction	00918 70364	20/4/05	
067	Roberts	Susan	The Elms	Arbroath	DD7 8HG	Autobiography	00634 89654	18/4/05	24/4/05

There are a number of problems associated with storing the data in this form as a single table.

- There is duplication of data. John Bell's details are entered twice.
- If data is duplicated it could lead to data inconsistency. For example, human error could result in Beel being entered instead of Bell in one of the records.
- It is not possible to store information about a member without storing details of a book - insertion problem.
- Removing a book from the database may result in removing the only record that stores information about a member - deletion problem.

These problems can be overcome by storing the data in a relational database. In a relational database data is only entered and stored once, so removing the possibility of data duplication and inconsistency.

Data



What do the following items mean to you?

- 12 34 24 33 19 11
- 21-11-1998
- AAAAA

Probably nothing - this is because they have no specific meaning or application. They are examples of Data. Data consists of unprocessed, raw facts and figures – they do not have any meaning until they are processed and turned into something useful. Data comes in many forms, the main ones being letters, numbers and symbols.

Information



When data is processed, (known as **data processing**), and given meaning and context it becomes information which is suitable for someone to understand and make use of. For example:

- Lottery Draw for 12/01/2013: 12 34 24 33 19 11
- Date of Birth: 21-11-1998
- Predicted Higher Grades: AAAAA

Therefore, information is processed data that has been given meaning and context. In order for information to be useful it must be interpreted by someone with knowledge.

Summary

Data are raw, unprocessed facts, which do not mean anything by themselves.

Information is data which has been organised or structured to give it meaning.

Examples of Data:

76
Jones
brown
01382563728
DD65RT
129

Examples of Information:

the man is 76 years old
the surname is Jones
the girl has brown eyes
telephone number
a post code
a house number



Knowledge

Knowledge is our understanding of the information which occurs from having the ability to apply information, i.e. it is an understanding of rules. For example:

- The lottery draw numbers 12 34 24 33 19 11 match the numbers on your ticket. - **You have won the lottery.**
- Your friend's date of birth is 21-11-2000 - **She is 13 and too young to vote**
- Predicted Higher Grades: AAAAAA - **You should be able to go to a good university**



Using Information – Searching

One of the most powerful tools in a database is the ability the program has to search through vast quantities of information.

There are various types of searches. You can do a simple search e.g. the school database could be searched for all pupils who live in Laurencekirk.

A complex search would allow you to search on more than one criteria. For example, you could search for all the pupils who live in Laurencekirk and who are in S1. You could also search for pupils living in Laurencekirk or Edzell. In this way the Logical Operators, AND OR NOT, can be used to search the database for specific information.

Sometimes you may want to search for something less than, greater than, contains etc. In Access the following operators can be used:

> more than	< less than
= equal to	<> not equal to
>= more than or equal to	<= less than or equal to

A wildcard, using the symbol *, can be used to find data which is included in a field. For example, in a music database if you wanted to find all song titles containing the word “baby” you would enter the criteria *baby* in the query.

Using Information – Sorting

Data can be sorted in a specific order on a particular field or fields.

Ascending Order: A- Z or 1-10

Descending Order: Z- A or 10 – 1

To sort data in order of surname alphabetically you would sort the surname field in ascending order.

Activity 3: Data, Information & Knowledge

1. Open a blank Word document and carry out the following:
 - a. Summarise the explanation of Data into your own words.
 - b. Summarise the explanation of Information into your own words.
 - c. Summarise the explanation of Knowledge into your own words.

Key Points to Remember

- An **Information System** is a combination of hardware and software which is used to input, process and store data, and then output information.
- **Data** consists of unprocessed, raw facts and figures.
- When data is processed and given meaning and context it becomes **Information**
- **Knowledge** is our understanding and interpretation of the information

User Interfaces: Forms



This form on the left is from Thomas Cook, the travel agents, website. In previous activities you have designed a database. This was you designing the “back end” of the system.

As a user of a database we tend to use an interface called a form to manipulate the system. This can be referred to as the “front end”.

A screenshot of the Thomas Cook website's holiday search form. The form is titled "Find My Perfect Holiday" and includes several sections: "Package Holidays" with radio buttons for "Flight + Hotel", "Hotels", "Flights", "City Breaks", and "Cruise"; "Provide Party Details" with dropdown menus for "Room 1" (set to 2) and "Children" (set to 0), and a link to "Add another room"; "Departure Date" set to "31 Aug 2013" with an "Amend" icon; a checked box for "I am flexible +/- 3 days"; "Holiday Duration" set to "14 Nights"; "Departure Airport" set to "Glasgow"; "Destination" set to "USA - Any (2)"; "Resorts" set to "Universal Orlando Resort Hotels (2)"; and "Board Basis" set to "I Don't Mind". A green "SEARCH >" button is at the bottom.

In this example you can see I have input information into the form. When I click on search this will search the database and then display my results.

Can you think of any other forms you have used whilst being online?

Key Fields

In database management systems, a key is a field that you use to sort data. It can also be called a key field or index. There are three types of database key:-

- Primary Key
- Foreign Key
- Surrogate Key



Primary key

Imagine what would happen if Mrs Smith telephoned the school to ask the secretary to pass on a message to her daughter Charlotte.

If there were nothing to uniquely identify the students' records, the secretary would have to search for all of the 'Charlotte Smiths' in school and then say, 'sorry to keep you waiting Mrs Smith, could you just tell me if your daughter is in S1, S2, S3, S4, S5 or S6? And if she is in S2, which one of these five girls is she?'

How about a patient going to have an operation. 'Oh hello Mr Jones, I have 4 Mr Joneses having an operation today, can you tell me if yours is to remove your tonsils, your arm, put your leg in a cast or give you a hair transplant?' I think that would be one very worried patient!

Therefore, it is important that **every single record** in a database has something to **uniquely identify** it and this is called the 'Primary Key' or sometimes the 'Key Field'.

Surrogate Key

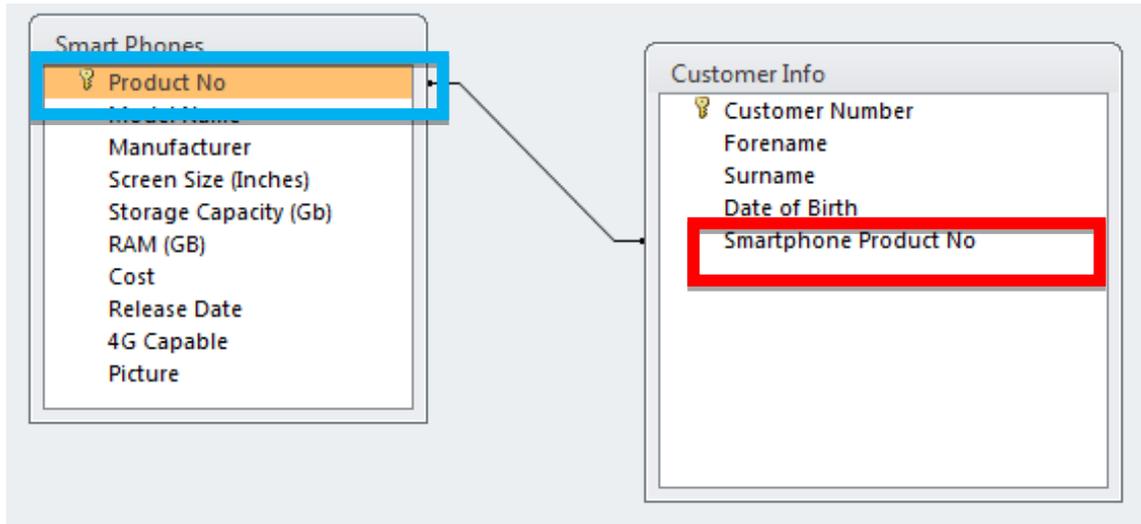
This is when a field does not exist in the table that would uniquely identify each record and therefore we have to add a sensible primary key.

So for example if we had a list of the names all the pupils who attend Strathaven Academy it is likely we would have some duplicated data. If we add a pupil ID field, each pupil can now be uniquely identified. Pupil ID would be a surrogate key as it was not part of the table to begin with and has been added to ensure each pupil can be uniquely identified.

Foreign Key

This is a field which is not the primary key in one table but is the primary key in a linked table. A foreign key is a field in a relational table that matches the primary key column of another table. The foreign key can be used to cross-reference between these tables. A foreign key is used to link tables together and create a relationship. It is a field in one table that is linked to the primary key in another table.

The **foreign key** is the key which allows a table to be linked to another table, in this table it is TeacherID and is identified with an *.



= Primary Key



= Foreign Key

When designing a database from scratch the first thing to do is to decide on the data to be stored, the fields and data types of each table. A foreign key is depicted as having an asterisk after it. (See the example below.)

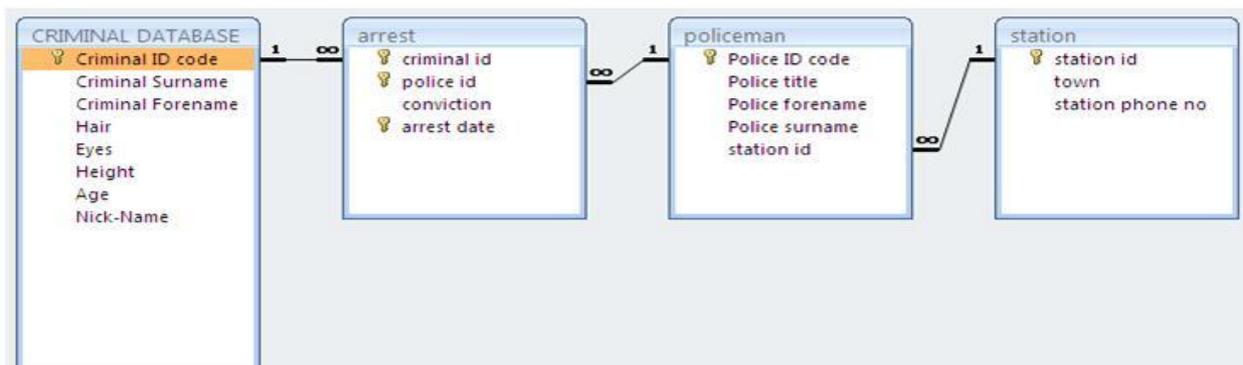
E.g. PUPIL(IDnumber
 Firstname
 Surname
 Dateofbirth
 Class
 TeacherID*).

IDnumber would be a **primary key** in the Pupil table, as primary keys are identified by being underlined.

Within a database, tables are linked to prevent duplication of data within a row of a table.

Linked Tables

Looking at a basic 2 table database storing information about criminals – one criminal may be arrested a number of times. The first table would store information about the criminal: name, DOB, height, address etc. The second table would store information about the conviction: date, arresting officer, crime, location etc. Taking it a step further you may have the following 4 table structure:



This is a process used to design a database in the most efficient way so that data is not repeated, the data is easy to search and edit and related tables are linked together.

Table 1 on page 7 shows how the data would look before it has been normalised. It has not been normalised as it contains multi-valued fields. The field's subject, ISBN, date borrowed and date returned contain more than one value for the member John Bell.

In normalisation one of the processes is to remove the repeating values and place them in a table of their own. If a database is in **First Normal form (1NF)**, it does not have any multi-valued fields. To put the library database into 1NF it is necessary to create separate tables for member and for loan:

MEMBER(MemberID, Surname, Forename, Address, Town, Postcode)

LOAN(ISBN, DateBorrowed, DateReturned, *MemberID)

MemberID is the primary key in the MEMBER table.

ISBN is the primary key in the LOAN table. MemberID is a foreign key in the LOAN table.

Table Relationship Diagram

This is used to show the relationship between different tables in a database.

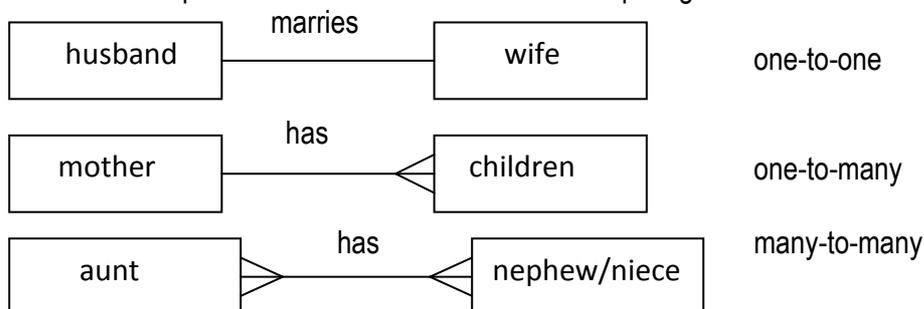
There are 3 possible types of relationship:

One-to-one(1:1) e.g. a husband has only one wife and a wife has only one husband

One-to-many(1:M) e.g. a mother can have lots of children but the children only have one mother

Many-to-many (M:M) e.g. A child can have many aunts and an aunt can have many nephews/nieces.

These relationships can be shown in a table relationship diagram as shown:



Referential Integrity means that data that exists in your foreign key field in one table must be present in the primary key field in the other table.

Validation



A validation check is a check done by the software to make sure the data is sensible, or valid.

The types of validation checks are as follows:

- ▢ **Restricted choice** – this means that only certain data will be accepted. For example, a check could be made in the Title field so that the only data permitted is “Mr”, “Ms”, “Dr” or “Mrs”. If a user tries to enter anything else an error message will be shown and the user will have to re-enter the data.
- ▢ **Range check** – only data within a certain range is acceptable.

An example of this could be checking that secondary school pupils are not older than 19 or younger or 10.

▮ Presence check – there has to be an entry in the field.

For example, in your school record the date of birth field has to be completed.

▮ Length check – there should be a specified number of characters in the field entry.

For example, a telephone number always has 12 characters, e.g 01382 123456 (the space is included as a character).

Verification

This is a check to make sure the data has been entered correctly. A common error in data entry is mis-spelling or mis-typing. Data can be verified by asking the user to enter it twice and if the 2 entries do not match the user will be asked to re-enter.

One of the best ways of ensuring that data is correct is to use machine-readable data entry, such as bar codes or OCR (Optical Character Recognition)



Registration

To register for access, please enter your desired username and password below. You will need to enter your password twice to ensure it's correct.

Preferred Username:

Password:

Password (again):