

A database is a collection of organised data that can be efficiently stored, sorted, and searched.

How the data is organised will often determine the *type* of database used. There are many different types of database; some examples of the different types are relational, object-orientated, graphical, NoSQL, and distributed.

## ACID principle

To ensure the integrity of a database, each change or transaction must conform to a set of rules known as ACID:

- **atomicity**
  - when changing data within a database, if any part of the change fails, the whole change will fail and the data will remain as it was before the change was made; this prevents partial records being created. Basically a safeguard
- **consistency**
  - before data can be changed in a database, it must be validated against a set of rules
- **isolation**
  - databases allow multiple changes at the same time, but each change is isolated from others
- **durability**
  - once a change has been made, the data is safe, even in the event of system failure

Databases will have mechanisms for **backup**, **distribution**, and **redundancy**, to ensure data is not lost.

## Database management system

A DBMS is software that can retrieve, add, and alter existing data in a database. MySQL, PostgreSQL, MongoDB, MariaDB are all examples of DBMSs. You can work with them via programming languages like PHP or through graphical clients such as PHPMyAdmin, MicrosoftSQL, Adminer etc. There is also SQLite which runs on the client not the server, so useful for learning and local development. SQLite is also useful when you need a database specific to a single device without networked communication, such as on mobile.

There are also CLI tools for all the major databases.

While I will be working primarily through PHP, graphical database software is useful for visual grepping and checking that scripts are working as they should.

## Relational database architecture

Tables, fields and records are the basic building blocks of databases

FL-Databases-1.5\_\_terminology 1.gif

### **Table**

A group of similar data with rows for **records** and columns for each **field**.

### **Record**

Horizontal/“row”: a collection of items which may be of different data types all relating to the individual or object that the record describes

### **Field**

Vertical/ “column” : stores a single particular unit of data for each record. Each field must use the same data type.

Each individual field has **properties:** such as the data type, length or the total memory allocation.