

Relational databases

Basic Terms

An understanding of relational databases requires an understanding of some of the basic terms.

- Data are the values stored in the database. On its own, data means very little. "43156" is an example.
- Information is data that is processed to have a meaning. For example, "43156" is the population of the town of Littlewood.
- A database is a collection of tables.
- Each table contains records, which are the horizontal rows in the table. These are also called tuples.
- Each record contains fields, which are the vertical columns of the table. These are also called attributes. An example would be a product record.
- Fields can be of many different types. There are many standard types, and each DBMS (database management system, such as Oracle or MySQL can also have their own specific types, but generally they fall into at least three kinds - character, numeric and date. For example, a product description would be a character field, a product release date would be a date field, and a product quantity in stock would be a numeric field.
- The domain refers to the possible values each field can contain (it's sometimes called a field specification). For example, a field entitled "marital_status" may be limited to the values "Married" and "Unmarried".
- A field is said to contain a null value when it contains nothing at all. Fields can create complexities in calculations and have consequences for data accuracy. For this reason, many fields are specifically set not to contain NULL values.
- A key is a logical way to access a record in a table. For example, in the product table, the product_id field could allow us to uniquely identify a record. A key that uniquely identifies a record is called a primary key.
- An index is a physical mechanism that improves the performance of a database. Indexes are often confused with keys. However, strictly speaking they are part of the physical structure, while keys are part of the logical structure.
- A view is a virtual table made up of a subset of the actual tables.

- A one-to-one (1:1) relationship occurs where, for each instance of table A, only one instance of table B exists, and vice-versa. For example, each vehicle registration is associated with only one engine number, and vice-versa.
- A one-to-many (1:m) relationship is where, for each instance of table A, many instances of the table B exist, but for each instance of table B, only once instance of table A exists. For example, for each artist, there are many paintings. Since it is a one-to-many relationship, and not many-to-many, in this case each painting can only have been painted by one artist.
- A many to many (m:n) relationship occurs where, for each instance of table A, there are many instances of table B, and for each instance of table B, there are many instances of the table A. For example, a poetry anthology can have many authors, and each author can appear in many poetry anthologies.
- A mandatory relationship exists where, for each instance of table A, one or more instances of table B must exist. For example, for a poetry anthology to exist, there must exist at least one poem in the anthology. The reverse is not necessarily true though, as for a poem to exist, there is no need for it to appear in a poetry anthology.
- An optional relationship is where, for each instance of table A, there may exist instances of table B. For example, a poet does not necessarily have to appear in a poetry anthology. The reverse isn't necessarily true though, for example for the anthology to be listed, it must have some poets.
- Data integrity describes the accuracy, validity and consistency of data. An example of poor integrity would be where a poet's name is stored differently in two different places.
- Database normalization is a technique that helps us to reduce the occurrence of data anomalies and poor data integrity.