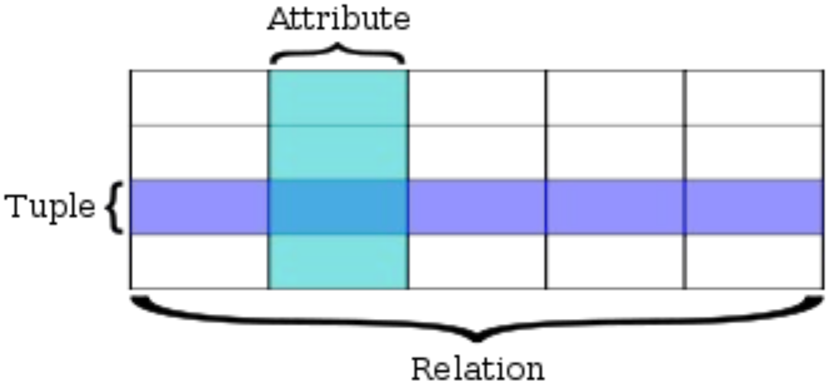
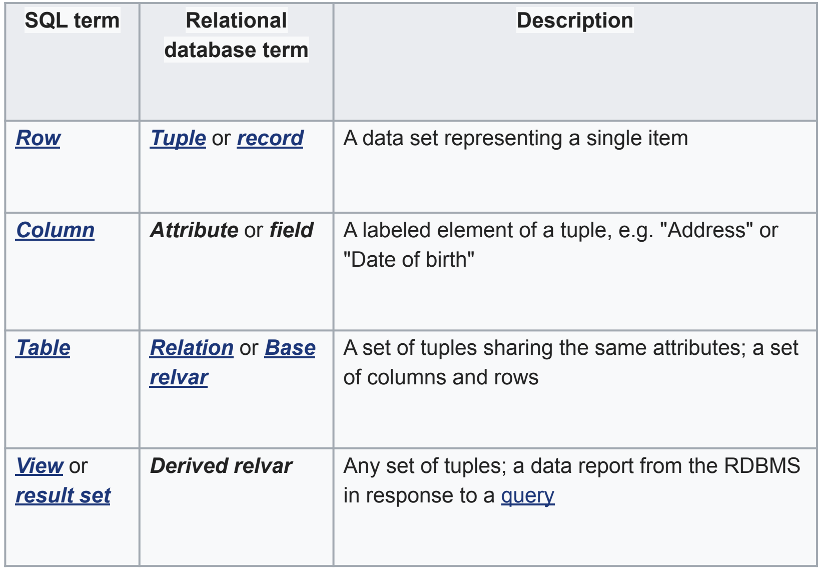
Database 5.1

A database is a collection of information that is organized so that it can be easily accessed, managed and updated.

Data is organized into rows, columns and tables, and it is indexed to make it easier to find relevant information. Data gets updated, expanded and deleted as new information is added. Databases process workloads to create and update themselves, querying the data they contain and running applications against it.



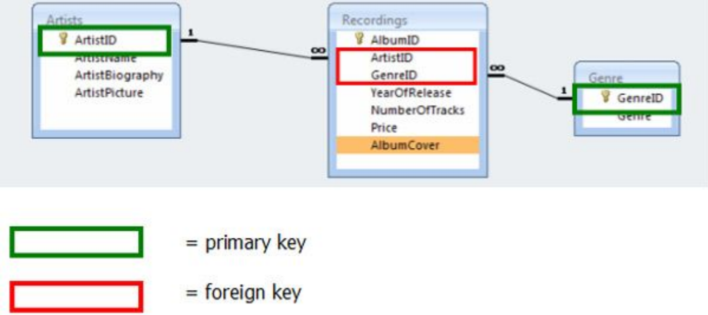


SQL term	Relational database term	Description
Row	***Tuple*** or ***record***	A data set representing a single item
Column	***Attribute*** or ***field***	A labeled element of a tuple, e.g. "Address" or "Date of birth"
Table	***Relation*** or ***Base relvar***	A set of tuples sharing the same attributes; a set of columns and rows
View or ***result set***	***Derived relvar***	Any set of tuples; a data report from the RDBMS in response to a query

A **primary key** is a special relational database table column (or combination of columns) designated to uniquely identify all table records.

A **foreign key** is a column or group of columns in a relational database table that provides a link between data in two tables. It acts as a cross-reference between tables because it references the primary key of another table, thereby establishing a link between them.

A **database index** is a data structure that improves the speed of data retrieval operations on a database table at the cost of additional writes and storage space to maintain the index data structure. Indexes are used to quickly locate data without having to search every row in a database table every time a database table is accessed. Indexes can be created using one or more columns of a database table, providing the basis for both rapid random lookups and efficient access of ordered records.



ERD

An entity-relationship diagram (ERD) is a data modeling technique that graphically illustrates an information system's entities and the relationships between those entities. An ERD is a conceptual and representational model of data used to represent the entity framework infrastructure.

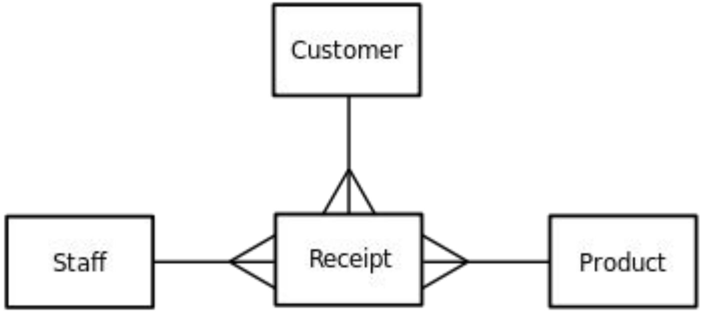
The elements of an ERD are:

- Entities
- Relationships
- Attributes

Steps involved in creating an ERD include:

1. Identifying and defining the entities
2. Determining all interactions between the entities
3. Analyzing the nature of interactions/determining the cardinality of the relationships
4. Creating the ERD

A receipt has one customer
A customer can have many receipts
A receipt has one product
A product can be part of many receipts
A receipt was created by one staff member
A staff member can create many receipts



To link tables together and allow for us to query a database we use relationships. There are three types of relationship that you need to know. Each is shown below using 'crows foot notation' which is one of many ways to describe these relationships:

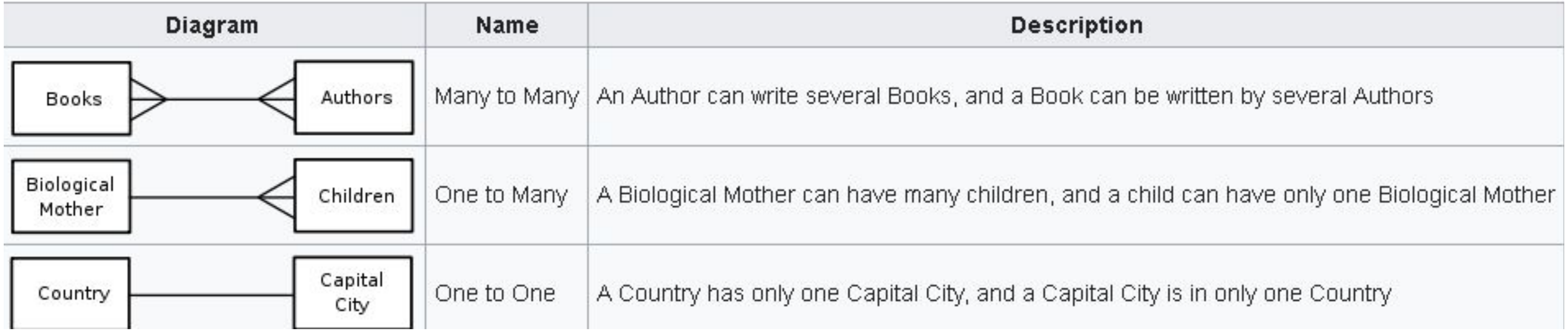


Diagram	Name	Description
Books Authors	Many to Many	An Author can write several Books, and a Book can be written by several Authors
Biological Mother Children	One to Many	A Biological Mother can have many children, and a child can have only one Biological Mother
Country Capital City	One to One	A Country has only one Capital City, and a Capital City is in only one Country

A **data definition language** or **data description language** (**DDL**) is a syntax similar to a computer programming language for defining data structures, especially database schemas.

A **data manipulation language** (**DML**) is a computer programming language used for adding (inserting), deleting, and modifying (updating) data in a database A DML is often a sublanguage of a broader database language such as SQL, with the DML comprising some of the operators in the language.

1. **DDL vs. DML**: DDL statements are used for creating and defining the Database structure. DML statements are used for managing data within Database.

2. **Sample Statements**: DDL statements are CREATE, ALTER, DROP, TRUNCATE, RENAME etc. DML statements are SELECT, INSERT, DELETE, UPDATE, MERGE, CALL etc.

3. **Number of Rows**: DDL statements work on whole table. CREATE will a create a new table. DROP will remove the whole table. TRUNCATE will delete all records in a table. DML statements can work on one or more rows. INSERT can insert one or more rows. DELETE can remove one or more rows.

4. **WHERE clause**: DDL statements do not have a WHERE clause to filter the data. Most of DML statements support filtering the data by WHERE clause.

5. **Commit**: Changes done by a DDL statement can not be rolled back. So there is no need to issue a COMMIT or ROLLBACK command after DDL statement. We need to run COMMIT or ROLLBACK to confirm our changed after running a DML statement.

6. **Transaction**: Since each DDL statement is permanent, we can not run multiple DDL statements in a group like Transaction. DML statements can be run in a Transaction. Then we can COMMIT or ROLLBACK this group as a transaction. Eg. We can insert data in two tables and commit it together in a transaction.

7. **Triggers**: After DDL statements no triggers are fired. But after DML statements relevant triggers can be fired.

Database Manipulation Language (DML)

DML statements are used to work with data in an existing database. The most common DML statements are:

- `SELECT`
- `INSERT`
- `UPDATE`
- `DELETE`

Database Definition Language (DDL)

DDL statements are used to structure objects in a database. The most common DDL statements are:

- `CREATE`
- `ALTER`
- `DROP`

SQL

The SQL **SELECT** statement returns a result set of records from one or more tables.

An SQL **INSERT** statement adds one or more records to any single table in a relational database.

An SQL **UPDATE** statement changes the data of one or more records in a table. Either all the rows can be updated, or a subset may be chosen using a condition.

In the database structured query language (SQL), the **DELETE** statement removes one or more records from a table. A subset may be defined for deletion using a condition, otherwise all records are removed. Some DBMSs, like MySQL, allow deletion of rows from multiple tables with one DELETE statement (this is sometimes called multi-table DELETE).

The CREATE TABLE statement is used to create a new table in a database.

The ALTER TABLE statement is used to add, delete, or modify columns in an existing table.

The ALTER TABLE statement is also used to add and drop various constraints on an existing table.

The DROP TABLE statement is used to drop an existing table in a database.