**SQL FUNDAMENTALS**

**Getting Started with Databases**

**What is a database?**

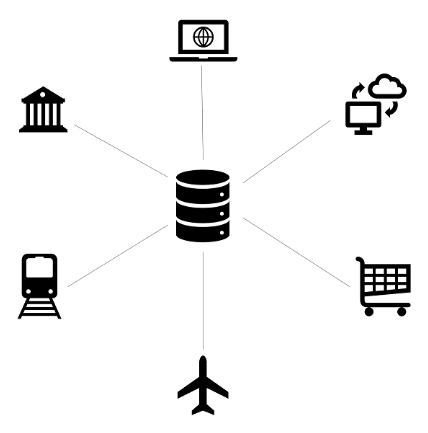
In simple terms, a database is a collection of data stored in a computer system. Here are other definitions of a database:  
   
According to **Wikipedia**:

A database is an organized collection of data, generally stored and accessed electronically from a computer system.

According to **Oracle**:

A database is an organized collection of structured information, or data, typically stored electronically in a computer system.

People use data and databases with or without awareness in their daily life activities.  
  
**How are databases used in the real-world?**



Databases are almost everywhere. Your bank, your grocery store, an app on your cellphone, websites all use databases to keep track of your data. When you access a website, the website starts to collect your data (e.g. accessing date and time, your location, your browser info) and store it in its database.

Let's take another example. When you order a product on a commercial website, your order is stored in a database. You withdrew money from your bank account. Your bank stores this transaction in the database. Social media platforms such as Facebook, Instagram, Twitter use databases to store data like members, their friends, member activities, messages, advertisements, etc.

(Note: In diagrams, databases are represented as a cylinder shape.)

| **Applications of the databases** | |
| --- | --- |
| **Category** | **Usage** |
| Banking&Finance | Customer information, accounts, transactions |
| Education | Student information, course registrations, grades |
| Telecommunication | Internet&phone usage, subscriber information |
| Human resources | Employees, managers, salaries, hire&termination dates |
| Websites | Products, visitors, website traffic statistics |
| Transportation | Passenger, reservation and schedule information |

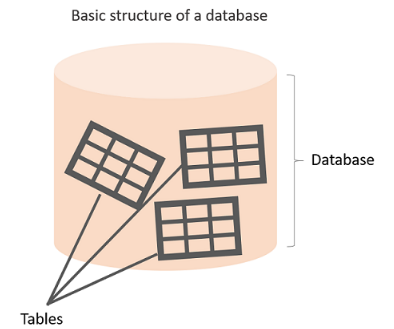
**☍ Discussion:** Think about other applications of databases. Try to give some examples.

A database is typically controlled by a **database management system (DBMS)**. Data and DBMS along with the applications that are associated with them are called a database system, often shortened to just database.  
  
Now that we've learned what the database is, we can move on to what is inside the database, how the data is stored in the database.

Q: What is a Database?  
A: A database is an organized collection of data, generally stored and accessed electronically from a computer system. In simple terms, a database is a collection of data stored in a computer system. When you order a product on a commercial website, your order is stored in a database. You withdrew money from your bank account. Your bank stores this transaction in the database. Social media platforms such as Facebook, Instagram, Twitter use databases to store data like members, their friends, member activities, messages, advertisements, etc.

### What is in a database?

The information inside the database is grouped into tables. A table in a database is called a *database table*. Tables are the basic unit of data storage in databases. We talked about the definition of the database in the previous lesson. We used the term **structured data.**Structured data heremeans table. A table consists of columns and rows. You may think of it as an Excel or Google spreadsheet. Although there are similarities between the database table and Excel/Google spreadsheet, they are different things. We won't get into details here.



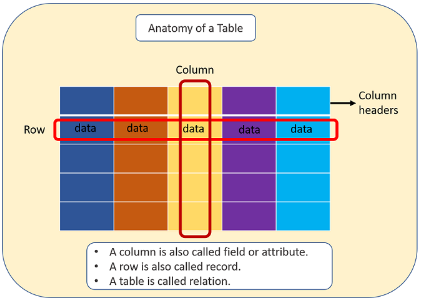
**Anatomy of a Table**

A table is made up of columns and rows. A column is a piece of data stored by the table. A row is a single set of columns that describe the attributes of a single thing. Columns should have a unique name. Columns and rows together make up a table.

**💡Tips:**

* In database world;
* A column is also called a field or attribute,
* A row is also called a record or a tuple,
* A table is also called a relation.

Column header names are written in lowercase, and there shouldn't be any space in a single name. Put an underscore between them. For instance, don't write "*last name"* as a header name, instead, write "*last\_name"*. Table names are also written in lowercase and without space (*employees*, *customers*, etc.)

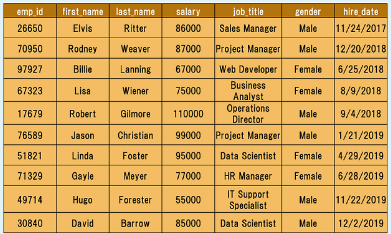


A database can consist of one or more tables. In most cases, more than one table. Each table has a unique name, such as employees, departments, or customers, etc.

##### Example

Let's take the case of a company database. Suppose that it has two tables. One is employees, and other is departments. Take a closer look at the employees table below. Here is the breakdown of the table.

* Table's name is *employees.*
* The table has seven columns (aka, fields or attributes)
* The table has ten rows (aka, records or tuples)
* Table's column header names are:  emp\_id, first\_name, last\_name, salary, job\_title, gender, hire\_date
* Inside the employees' table, there is data about each employee in the company



Of course, a real-world company would have many more employees. We use a small table to illustrate concepts.

Q: What is a table, column and row?  
A: A table is an organized collection of data stored in the form of columns and rows. Columns can be categorized as vertical and rows as horizontal. The columns in a table are called fields while the rows can be referred to as records.

**Type of Databases**

Typically, there are two main database storage types:

* **Relational Database - SQL**
* **Non-Relational Database - NoSQL**

A *relational database* is a type of database that stores and provides access to data points that are related to one another. Relational databases are based on the relational model, an intuitive, straightforward way of representing data in tables. In a relational database, each row in the table is a record with a unique ID called the *key*. The columns of the table hold attributes of the data, and each record usually has a value for each attribute, making it easy to establish the relationships among data points.

Actually, the term "relational database" was invented by E. F. Codd at IBM in 1970. Codd introduced the term in his research paper "A Relational Model of Data for Large Shared Data Banks". In this paper and later papers, he defined what he meant by "relational". One well-known definition of what constitutes a relational database system is composed of Codd's 12 rules. However, no commercial implementations of the relational model conform to all of Codd's rules, so the term has gradually come to describe a broader class of database systems, which at a minimum:

* Present the data to the user as relations (a presentation in tabular form, i.e. as a *collection* of tables with each table consisting of a set of rows and columns);
* Provide relational operators to manipulate the data in tabular form.

A software system used to maintain relational databases is called a *Relational Database Management System (RDBMS).*  Here are some examples of RDBMS:

* Amazon Aurora
* Amazon RDS
* Microsoft SQL Server
* Oracle Database
* MySQL
* IBM DB2
* Maria DB
* PostgreSQL
* SQLite

SQL (stands for Structured Query Language) is accepted as the standard Relational Database Management System (RDBMS) language. So we usually prefer to call Relational Database as SQL and Non-Relational database as NoSQL.

Until now in the course, what we've shown you some relational database features (structured data, tables, rows, columns, etc). Of course, there are more than those features coming in the upcoming parts of the course. During the pre-class phase of the SQL course, we will use SQLite as an RDBMS tool.

**💡Tips:**

* SQLite is a relational database management system contained in a C library. In contrast to many other database management systems, SQLite is not a client–server database engine. Rather, it is embedded into the end program. (Wikipedia)

Now, it's time to communicate with our database with SQL.

**Structured Query Language (SQL)**

SQL stands for Structured Query Language [and](http://ec2-35-173-203-107.compute-1.amazonaws.com/lms/mod/lesson/view.php?id=295) used to communicate with relational databases. SQL is a declarative language, not a procedural language. You write a single SQL declaration and hand it to the DBMS. The DBMS then executes internal code, which is hidden from us.

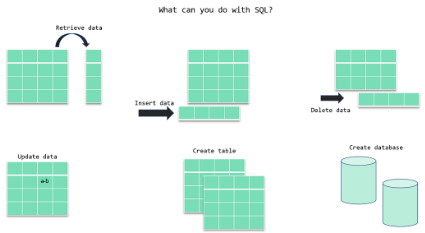
**💡Tips:**

* Declarative paradigm is where you say what you want without having to say how to do it. With procedural paradigm (used in JAVA, C), you have to specify exact steps to get the result. SQL is declarative than procedural since the queries don't specify steps to produce the result.
* SQL in it's purest form is not a programming language, but a query language. Because, it needs to be able to perform loops and control structures. However, with some extensions, SQL can have looping and control structures but they exist outside or rather as an appendage to the original SQL spec. In this manner, some argue that SQL is a programming language.

Most commercial database systems employ the SQL language.With SQL, you can access or manipulate data stored in the database. There are different types of access. These are:

* Retrieval of data from the database
* Insertion of new data into the database
* Updating the data in the database
* Deletion of data from the database

Besides, you can create new databases and tables using SQL.



In the next section, we will focus on the retrieval portion of the SQL. There is a particular word that is called *query* for retrieval information from the database. You will often encounter this word during your SQL course journey.

ⓘ A **query** is a statement asking for the retrieval of information from the database.

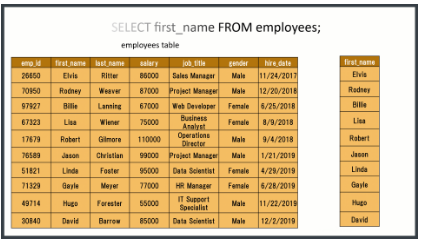
Since SQL is a language, it has grammar. Now, let's learn how to write in that language.

Q: What is SQL?  
A: SQL stands for Structured Query Language and used to communicate with a database. With SQL, you can access or manipulate data stored in the database.

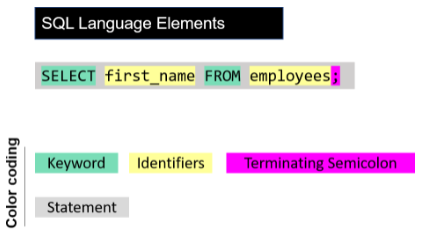
### SQL Language Elements

SQL language structure is easy to understand. It looks like plain English. Here is an example SQL statement:

SELECT first\_name FROM employees;

If we apply this SQL statement to our *employees*table, we get a single column that is *first\_name. See the command in action below: (Note: if the below gif doesn't play fast, please refresh the page)*  
  


The whole command SELECT first\_name FROM employees; is called **statement.**Let's breakdown of this statement. Herein SELECT, FROM words are keywords. They are special commands for SQL. first\_name, employees are identifiers. SQL statements end with a semicolon (;).  SQL Language Elements is also called SQL Syntax. 



There are other SQL elements which we will cover later in the course. In the next part, we will introduce you SELECT statement in detail. We will learn the syntax and start to write our commands.