**Databases**

**Introduction**

Databases are collections of data, usually organized under a schema, and stored in a format that is efficient for storing and retrieving the data. When people talk about databases they tend to talk about the underlying **Database Management System (DBMS)**. These are programs like MySQL or PostgreSQL which are designed to complete the tasks of storing, retrieving, updating, caching, deleting, and other data manipulation.

Databases use tables for managing data. Using tables we can handle [big data](https://lms.clarusway.com/mod/lesson/view.php?id=996" \o "Big Data) storage, build a relationship between data and give priorities to some data on the table. Thus we can reach specific information asking questions to tables. These questions called **query**.

| **https://docs.google.com/uc?id=1mV-LTDPYEzYDrFqW9YLvOZolZAKG7sNZ** |
| --- |
| *Database and tables* |

Q: Explain Database and Database Management System.  
A: A database is an organized collection of structured information, or data, typically stored electronically in a computer system. A database is usually controlled by a database management system (DBMS). Some examples of popular database software or DBMSs include MySQL, MongoDB, PostgreSQL, Microsoft SQL Server

**Database Structure**

Imagine an education program that stores student information, lesson and class information, teacher information, a student enrolled course information etc.

1. All of these data relate to one another:
   * All students can enroll more than one courses (students, courses, etc)
   * All courses have to have a teacher (teacher, courses)
   * Courses have a status (close, open for enrolling, etc)
2. Imagine this data is stored in [files](https://lms.clarusway.com/mod/lesson/view.php?id=1052" \o "Files) on a hard drive.  
   Is it stored by the student's name? If so, *how can we select only students that passed "Computer Essentials and Network" Course*?

A database system’s fundamental goal is to provide consistent views of structured data, just like the relationships we’ve laid out between students and courses.

**Structure**

SQL databases are based on around relational algebra

* *Tables* are the way we look at our relevant data.
* *Columns* are fields in the table.
* *Rows* define a relation between fields.
* A *Primary key* is a set of columns that uniquely identify rows in a table.
* A *Foreign key* is a column that matches the primary key of another table.

**Table-1 Students**

| **Student-ID (Primary key)** | **Student Name (Column-2)** | **Student Mail Address (Column-3)** |
| --- | --- | --- |
| 0001 (Row-1) | Albert Einstein | einstein@clarusway.com |
| 0002 (Row-2) | Nikola Tesla | tesla@clarusway.com |
| ... | ... | ... |

**Table-2 Courses**

| **Course Code (Primary key)** | **Course Name** | **Enrolled Student (Foreign key to Table 1)** |
| --- | --- | --- |
| CESN501 | Computer Essentials and Networking | 0002 |
| ... | ... | ... |

**Types of Databases**

There are two broad types of databases. These are SQL and NoSQL.

1. SQL: Stores data in tables organized by column and field.
2. NoSQL: Stores data differently than an SQL database.

SQL databases are primarily called Relational Databases (RDBMS); whereas the NoSQL database is primarily called a non-relational or distributed database.

**SQL**

* SQL databases are classic databases and are what we default to talking about when we teach databases.
* SQL databases define and manipulate data based on structured query language (SQL)
* SQL databases are table-based
* Great support is available for all SQL databases from their vendors. Also, a lot of independent consultations are there who can help you with SQL database for a very large scale deployments

Examples:

* MySQL/MariaDB
* PostgreSQL
* SQLite

**NoSQL**

* A NoSQL database has a dynamic schema for unstructured data.
* NoSQL databases are either key-value pairs, document-based, graph databases or wide-column stores.
* NoSQL databases can also offer more flexibility in storage options, allowing one to spread data across many machines more easily than SQL databases tend to do.
* In NoSQL database you have to rely on community support and only limited outside experts are available for setting up and deploying your large scale NoSQL deployments.

Examples:

* MongoDB
* Apache Casandra
* Elasticsearch

Q: What is SQL and have you heard about NoSQL?  
A: SQL is a programming language used by nearly all **relational databases** to query, manipulate, define data and to provide access control. A NoSQL, or **nonrelational database**, allows unstructured and semistructured data to be stored and manipulated (in contrast to a relational database)

**Database and Query Example**

Assume that Apple's HR department wants to hire a person who knows Java programming language and age is under 25. Let's look and make a query to find where is the lucky guy from Clarusway Database?

**Students** (Table)

| **ID** | **Student Name** | **Enrolled Courses** | **Age** |
| --- | --- | --- | --- |
| 0001 | Albertoo Einstein | SQL | 24 |
| 0002 | Nikolass Tesla | Introduction to Testing | 32 |
| 0003 | Steve Jobies | Java | 28 |
| 0004 | Tim Cookie | Java | 21 |

Write Query will be like that:

SELECT Student Name FROM Students

WHERE Enrolled Courses= Java AND Age < 25 ;

#(Tim [Cookies](https://lms.clarusway.com/mod/lesson/view.php?id=1044))