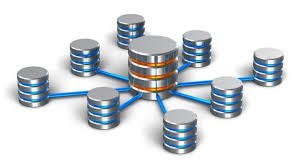
**Database Introduction**

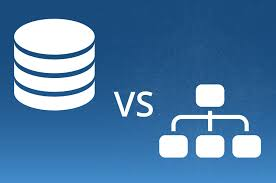
At the end of this Chapter, the learner will be able to:

* Define a database
* Illustrate the advantages of using databases
* List the main steps to design a database



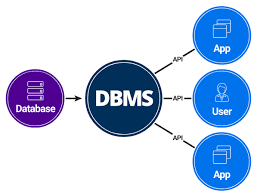
**File System Drawbacks**

* **Data Redundancy:** Data redundancy often requires higher storage costs and poor access time.
* **Data Inconsistency:** Data Inconsistency is caused by data Redundancy. It happens when similar data is stored in different formats in more than one file.
* **Data Isolation:** Data isolation occurs in writing new applications when retrieving data which can be stored in different format and files



**DataBase definition**

* Structured set of information designed and produced to be easily accessed, managed and modified by several users.
* A DataBase Management System (DBMS) is used to store information in a database. Using a DBMS, we can find, sort, transform and select information stored in database.



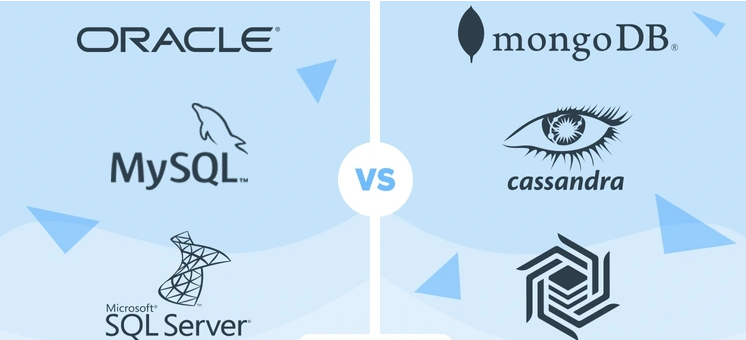
**SQL vs NoSQL: 5 Critical Differences**

When choosing a [modern database](https://www.integrate.io/blog/which-database/), one of the biggest decisions is picking a relational (SQL) or non-relational (NoSQL) data structure. While both are viable options, there are key differences between the two that users must keep in mind when making a decision.

Here, we break down the most important distinctions and discuss the best SQL and NoSQL database systems available.

The five critical differences between SQL vs NoSQL are:

**1.** SQL databases are relational, NoSQL databases are non-relational.  
**2.** SQL databases use structured query language and have a predefined schema. NoSQL databases have dynamic schemas for unstructured data.  
**3.** SQL databases are vertically scalable, while NoSQL databases are horizontally scalable.  
**4.** SQL databases are table-based, while NoSQL databases are document, key-value, graph, or wide-column stores.  
**5.** SQL databases are better for multi-row transactions, while NoSQL is better for unstructured data like documents or JSON.



**These resources can help you**

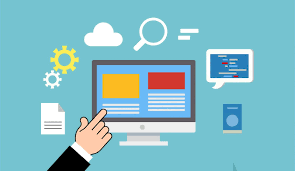
**dbms vs traditional file system**

[https://www.includehelp.com/dbms/dbms-vs-traditional-file-system.aspx](https://www.includehelp.com/dbms/dbms-vs-traditional-file-system.aspx" \t "_blank)

**Relational DataBase Management Systems (RDBMS)**

Using RDBMS allows us to:

* No Data redundancy
* Data Consistency and Integrity
* Easy access to data
* More flexibility than files
* Recovery process



**Examples of RDBMS’s**

There is so many Relational Database management systems. For this, it is important to find a way for them to communicate



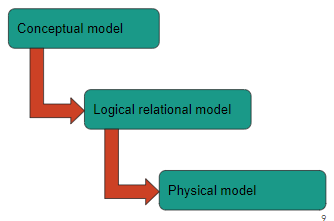
**These resources can help you**

**What is RDBMS**

[https://www.codecademy.com/articles/what-is-rdbms-sql](https://www.codecademy.com/articles/what-is-rdbms-sql" \t "_blank)

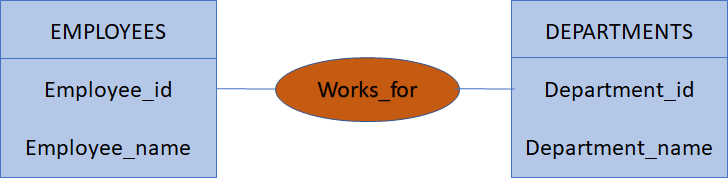
**Database design**

To design a database, it is important to follow 3 main steps:



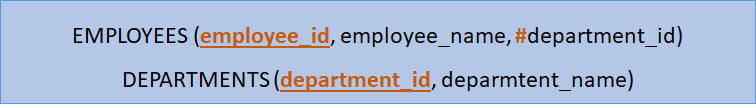
**Conceptual Model**

* A conceptual model is a graphic representation that serves to describe the functioning of a Database.
* It represents the main objects contained in the database, their characteristics and the relationships established between these different objects.
* This representation is normalized according to a well-defined modeling.
* The conceptual model is independent from DBMS.



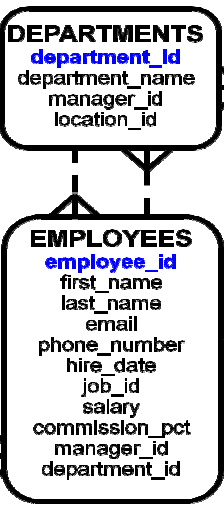
**Logical Relational model**

* A logical model or relational schema is a data representation model that adds more information to the conceptual model based on several rules.
* The logical model illustrates the data elements structure and set the relationships between these different elements.
* The logical model is independent from the DBMS



**Physical model**

* The physical data model or physical schema is an implementation of the logical data model in a specific RDBMS.
* It represents the structure and content of the database (tables, columns, data types, constraints, etc.).
* The physical model is dependent from the DBMS



**These resources can help you**

**Data Modeling**

[http://www.agiledata.org/essays/dataModeling101.html](http://www.agiledata.org/essays/dataModeling101.html" \t "_blank)

**Recap**

* A database is an organized set of data stored and managed using a DBMS.
* To design a database, we must follow 3 main steps:
  + Create the conceptual data model
  + Create the logical data model
  + Create the physical model based on a DBMS