What is Database?

Database is a collection of information organized for easy access, management and maintenance.

Examples:

- ✓ Telephone directory
- ✓ Customer data
- ✓ Product inventory
- ✓ Visitor's register
- ✓ Weather records

What is Database Management System?

Database Management System (DBMS) is a software for storing and retrieving users' data while considering appropriate security measures. DBMS provides an interface between the data and the software application, which allows users to create their own databases as per their requirement.

Characteristics of Database Management System:

- Provides security and removes redundancy
- Self-describing nature of a database system
- Insulation between programs and data abstraction
- Support of multiple views of the data
- Sharing of data and multiuser transaction processing
- DBMS allows entities and relations among them to form tables.
- It follows the ACID concept (Atomicity, Consistency, Isolation, and Durability).
- DBMS supports multi-user environment that allows users to access and manipulate data in parallel.

Popular DBMS Software

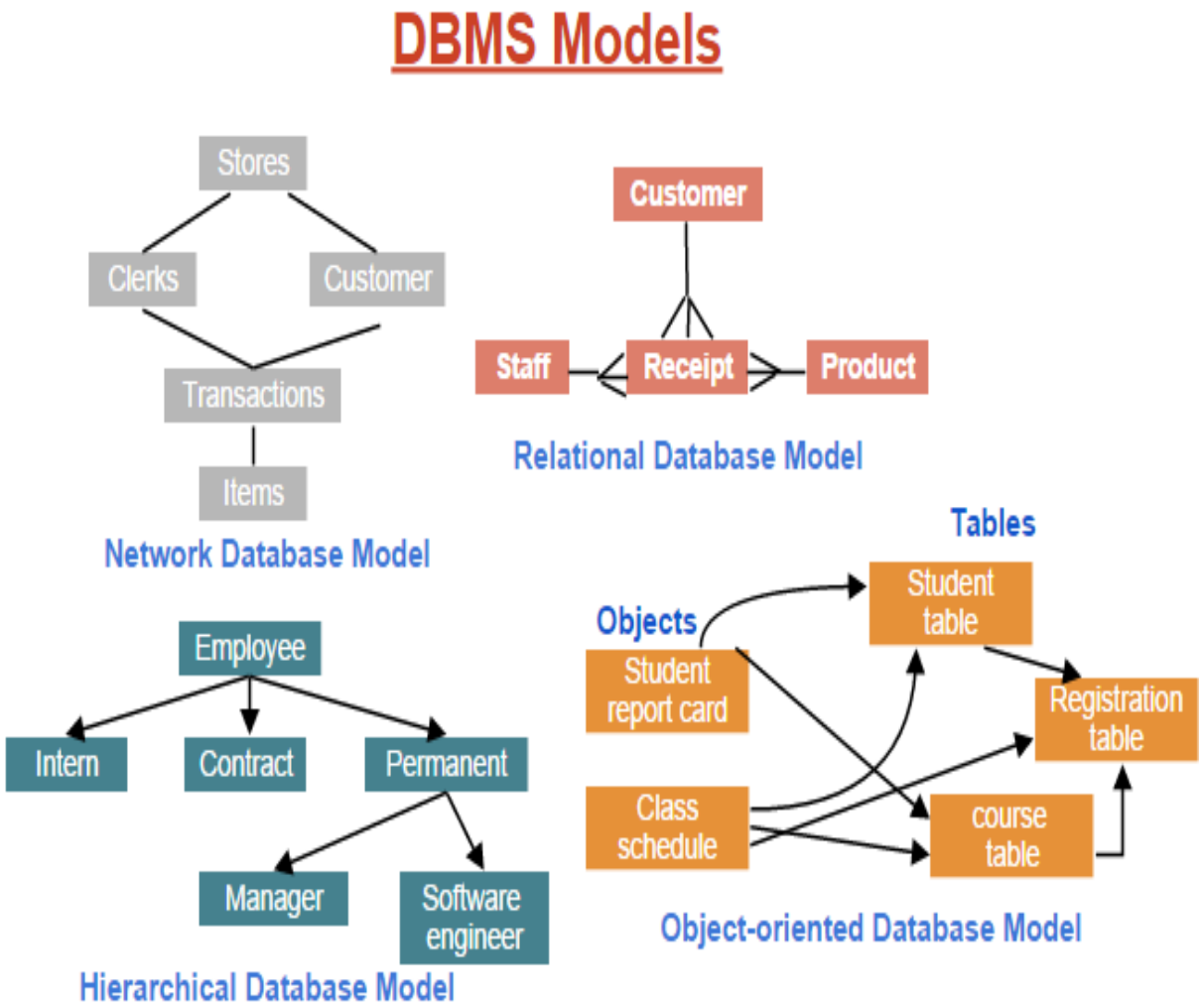
- MySQL
- Oracle
- PostgreSQL
- Microsoft SQL Server etc.

Application of DBMS

Sector	Use of DBMS
Banking	For customer information, account activities, payments, deposits, loans, etc.
Airlines	For reservations and schedule information.
Universities	For student information, course registrations, colleges and grades.
Telecommunication	It helps to keep call records, monthly bills, maintaining balances, etc.
Finance	For storing information about stock, sales, and purchases of financial instruments like stocks and bonds.
Sales	Use for storing customer, product & sales information.
Manufacturing	It is used for the management of supply chain and for tracking production of items. Inventories status in warehouses.
HR Management	For information about employees, salaries, payroll, deduction, generation of paychecks, etc.

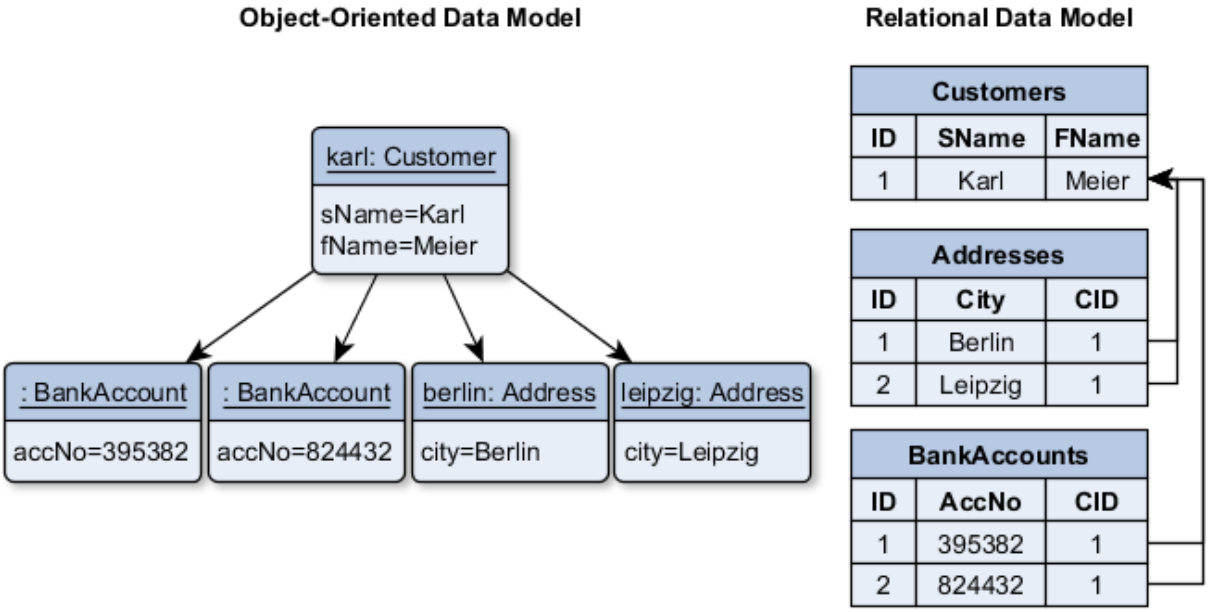
Types of Data Models:

- Object based logical model (Its act blue print of database consists of entities.)
  - Entity relationship model
- Record based logical model (Implementation on Object based logical model)
  - Hierarchical data model (Tree based model)
  - Network data model (Graph based model)
  - Relational data model (Tabular based model)



Object Based Logical Model

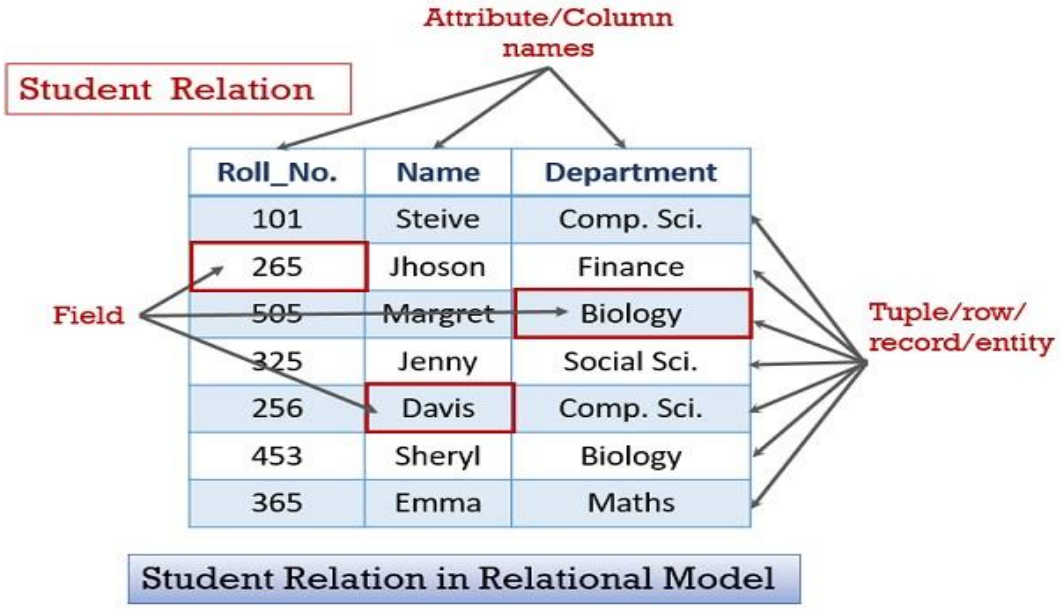
The **object-based models** use the concepts of entities or objects and relationships among them rather than the implementation-based concepts, such as records, used in the record-based models.



Record based logical model

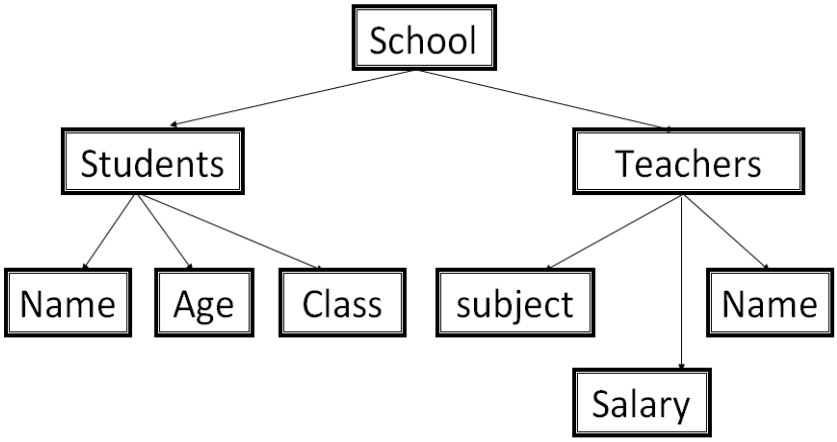
The **Record based model** describe data at conceptual and views levels. These model specify logical structure of database with records, fields and attributes.

1. Relational data model
   A relational data model involves the use of data tables that collect groups of elements into relations. These models work based on the idea that each table setup will include a primary key or identifier. Other tables use that identifier to provide "relational" data links and results.



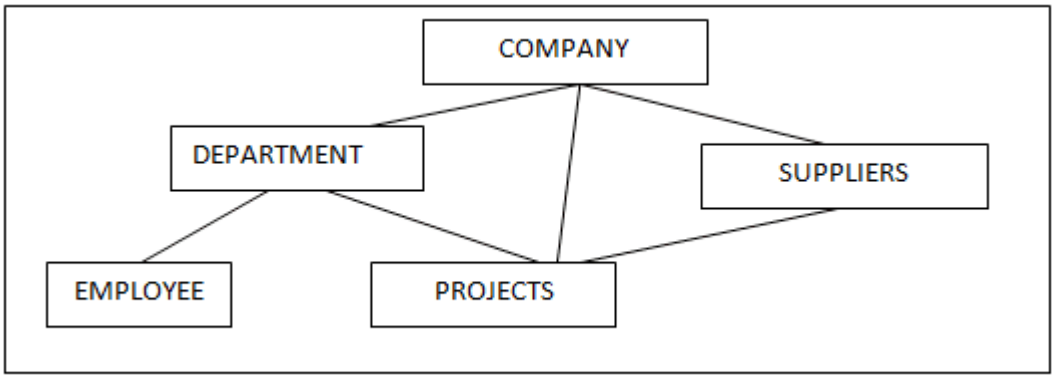
2. Hierarchical data model

In hierarchical model, data is organized into a tree like structure with each record is having one parent record and many children. The main drawback of this model is that, it can have only one to many relationships between nodes.



3. Network data model

A network database model is a database model that allows multiple records to be linked to the same owner file. The relationship that the information has in the network database model is defined as many-to-many relationship because one owner file can be linked to many member files and vice versa.

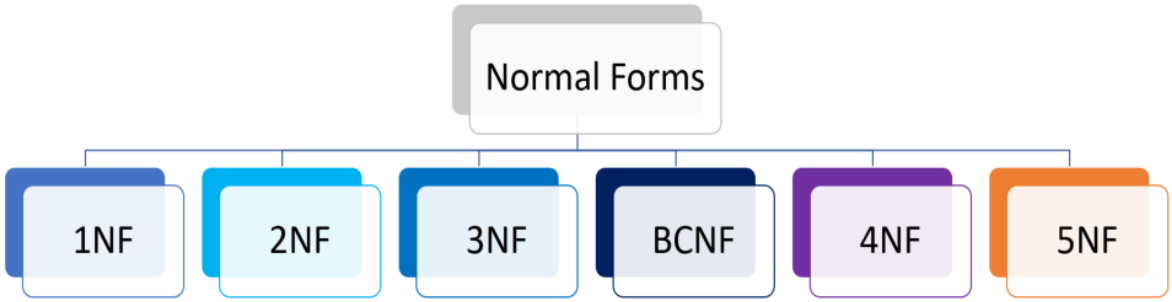


Network Database vs. Hierarchical Database Model

Network Database Model	Hierarchical Database Model
Many-to-many relationship	One-to-many relationship
Easily accessed because of the linkage between the information	Difficult to navigate because of its strict owner to member connection
Great flexibility among the information files because the multiple relationships among the files	Less flexibility with the collection of information because of the hierarchical position of the files

Normalization

Database normalization is the process of structuring a relational database in accordance with a series of so-called normal forms in order to reduce data redundancy, insertion anomaly, update anomaly & deletion anomaly and improve data integrity.



Explained with example:

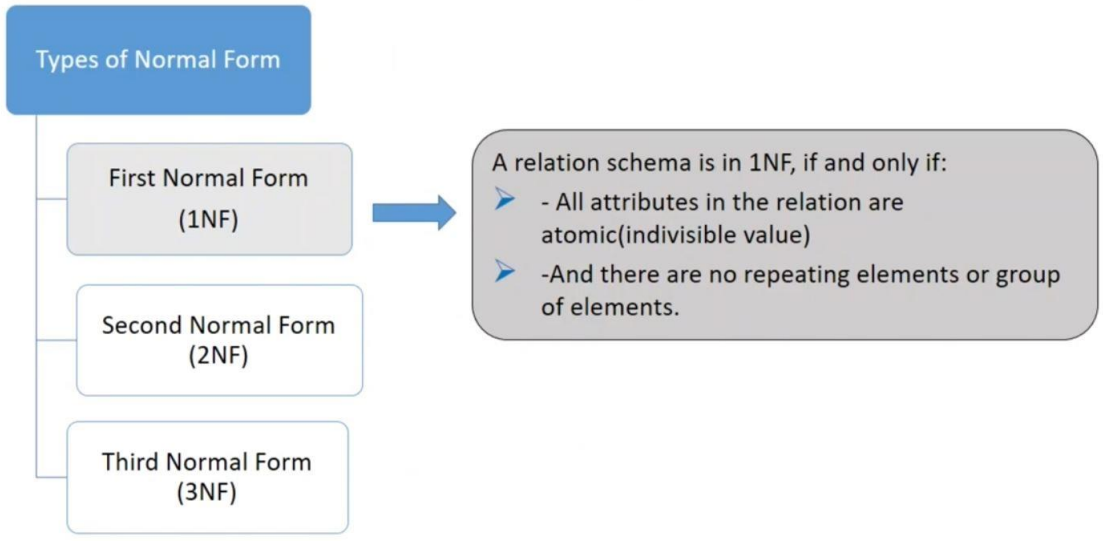
Need for Normalization

- **RAW DATABASE**

Student_Details	Course_details		Pre-requisite	Result_details	
101 Jack 11/4/1975	M1 Advance Math	17	Basic Math	03/11/2015 82	A
102 Rock 10/04/1976	P4 Advance Physics	18	Basic Physics	22/11/2015 83	A
103 Mary 11/07/1975	B3 Advance Biology	10	Basic Biology	14/11/2015 68	B
104 Roby 10/04/1976	H6 Advance History	19	Basic History	22/11/2015 83	A
105 Jim 03/08/1978	C3 Advance Chemistry	12	Basic Biology	15/11/2015 50	C

Step 1:

First Normal Form – (1NF)



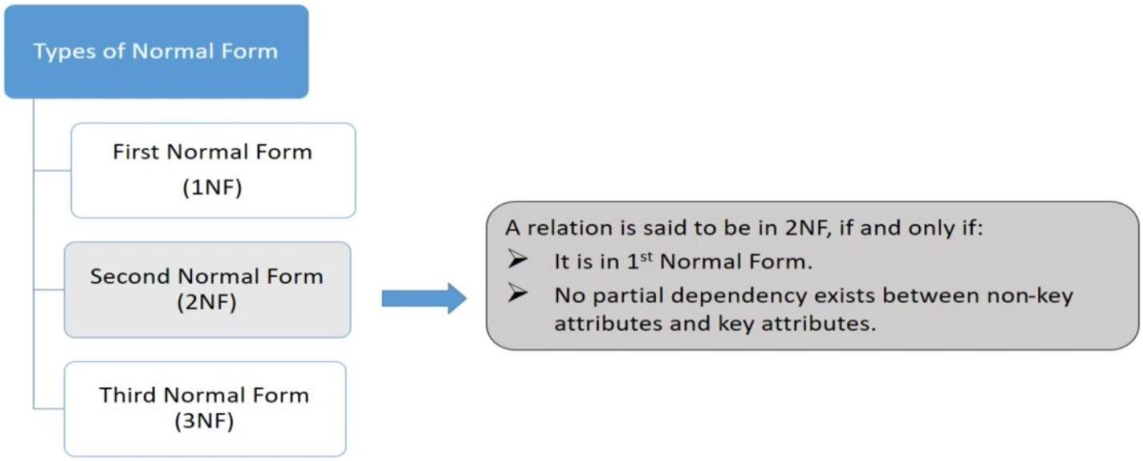
First Normal Form - (1NF)

Student Marks Table in 1NF

Student #	Student_ Name	Date Of Birth	Cours e#	CourseName	Pre-Requisite	Duration in days	Date Of Exam	Marks	Grade
101	Jack	11/4/1975	M1	Advance Math	Basic Math	17	02/11/ 2015	82	A
102	Roby	10/04/1976	P4	Advance Physics	Basic Physics	18	21/11/ 2015	83	A
103	Mary	11/07/1975	B3	Advance Biology	Basic Biology	10	12/11/ 2015	68	B
...	...	...	...	...	...	...	...	...	...

Step 2:

Second Normal Form – (2NF)



Second Normal Form – (2NF)

Student Table

Student#	Student_Name	Date Of Birth
101	Jack	11/4/1975
102	Roby	10/04/1976
103	Mary	11/07/1975

Result Table

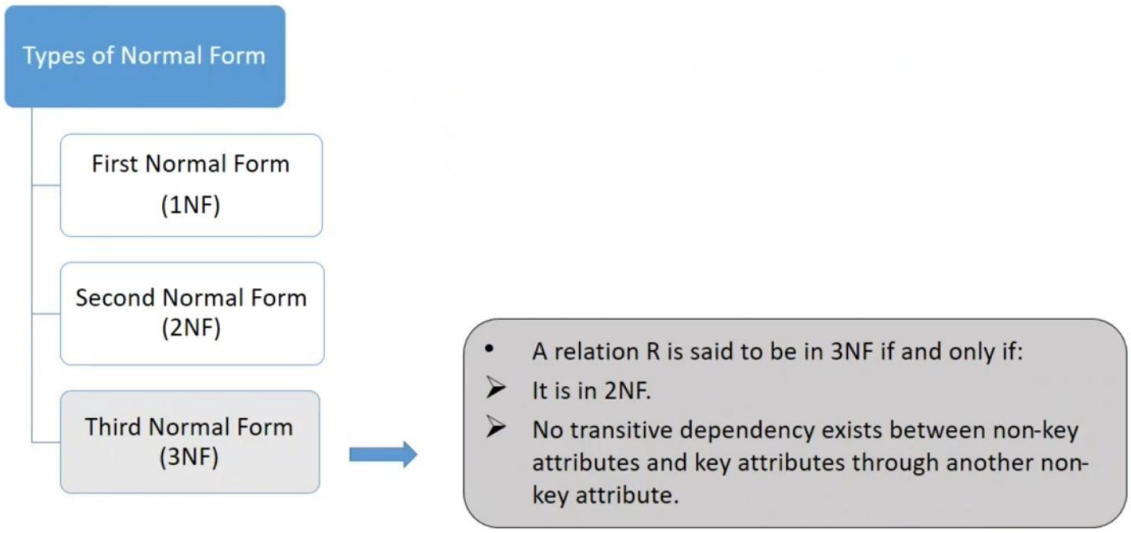
Student#	Course#	Marks	Grade
101	M1	82	A
102	P4	83	A
103	B3	68	B

Course Table

Course#	CourseName	Prerequisite	Durationindays	Date Of Exam
M1	Advance Maths	Basic Math	17	02/11/2015
P4	Advance Physics	Basic Physics	18	21/11/2015
B3	Advance Biology	Basic Biology	10	12/11/2015

Step 3:

Third Normal Form – (3NF)



Finally,

Third Normalization – (3NF)

Result Table

Student#	Course#	Marks
101	M1	82
102	P4	83
103	B3	68

Marks Grade Table

Marks	Grade
82	A
83	A
68	B