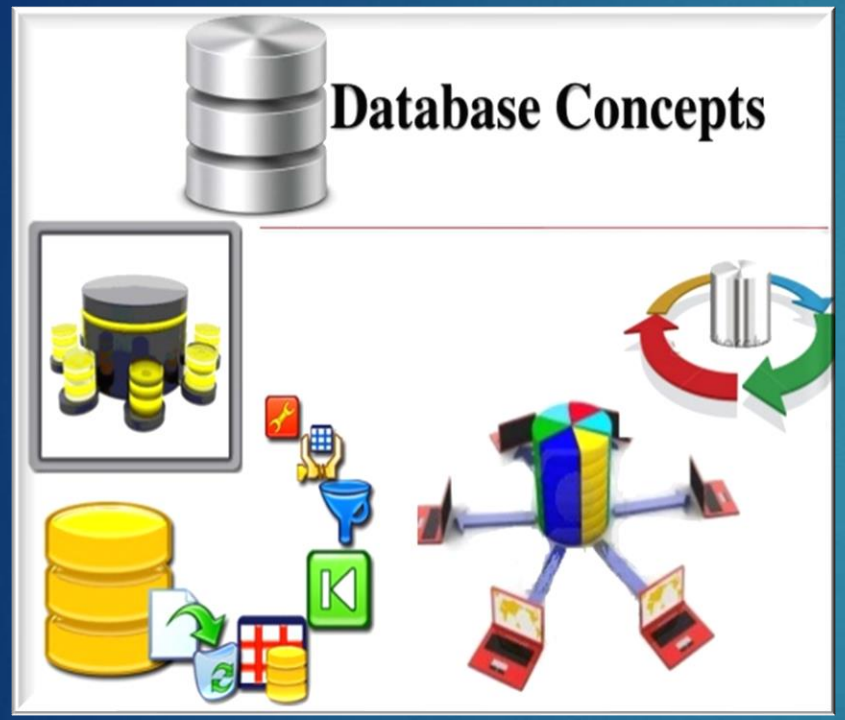
Database Normalization

Prof. K ADISESHA (Ph. D)



Database Normalization

- Introduction
- Normalization
- Database Anomaly
- Functional Dependency
- Normal Forms

Prof. K. Adisesha

Introduction

Database Design Strategies:

Database design can be defined as a collection of tasks or processes that enhance the designing, development, implementation, and maintenance of enterprise data management system.

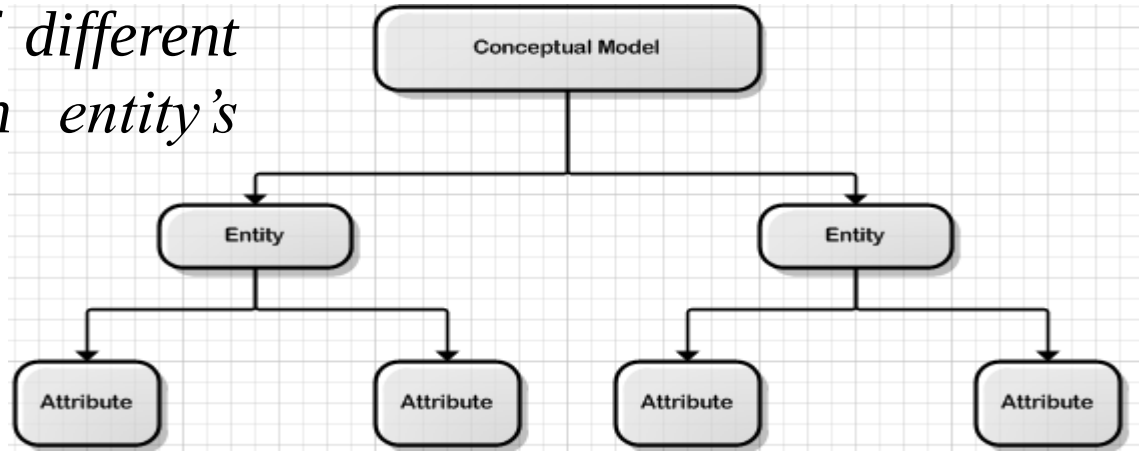
- *Designing a proper database reduces the maintenance cost thereby improving data consistency and the cost-effective measures are greatly influenced in terms of disk storage space.*
- *There are two approaches for developing any database:*
  - *The top-down method*
  - *The bottom-up method*

Introduction

Top – down design method:

The top-down design method starts from the general and moves to the specific.

- *It start with a general idea of what is needed for the system and then work your way down to the more specific details of how the system will interact.*
- *This process involves the identification of different entity types and the definition of each entity's attributes.*

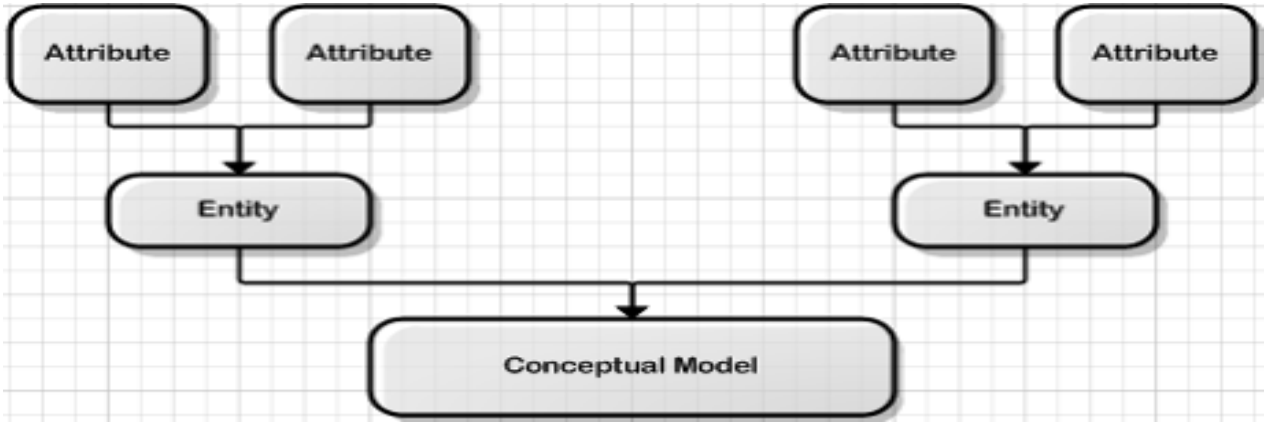


Introduction

Bottom – up design method:

The bottom-up approach begins with the specific details and moves up to the general.

- *This is done by first identifying the data elements (items) and then grouping them together in data sets.*
- *In other words, this method first identifies the attributes, and then groups them to form entities.*



Introduction

Normalization is the process of organizing the data and the attributes of a database.

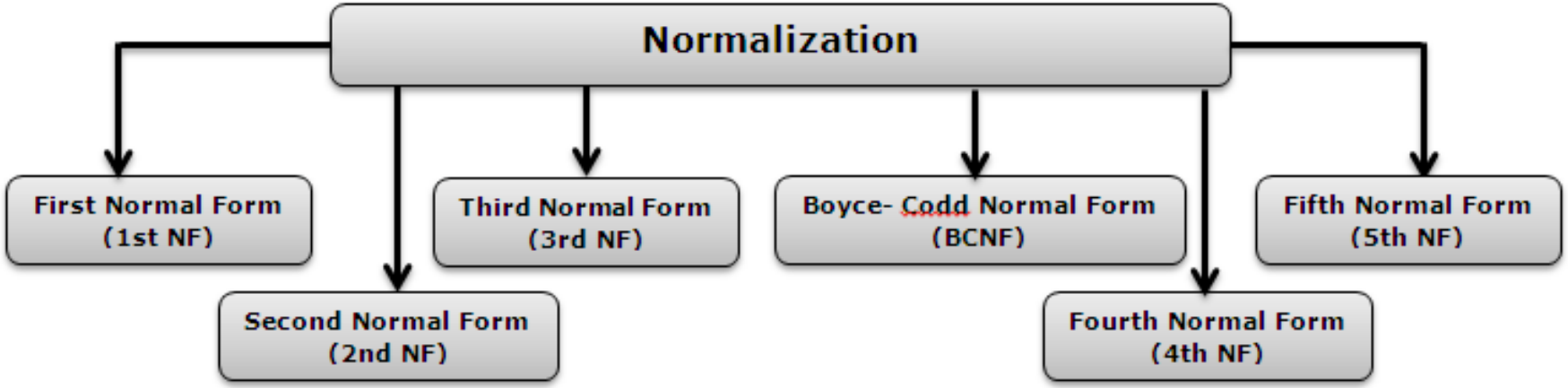
- *It is performed to reduce the data redundancy in a database and to ensure that data is stored logically.*
- *Data redundancy means having the same data but at multiple places.*
- *It is necessary to remove data redundancy because it causes anomalies in a database which makes it very hard for a database administrator to maintain it.*

Introduction

Normalization:

Normalization is a step by step process of removing the different kinds of redundancy and anomaly one step at a time from the database.

- E.F Codd developed for the relation data model in 1970.
- Normalization rules are divided into following normal form:



Normalization

Normalization is the process of organizing the data and the attributes of a database. The main reason for normalizing the relations is removing these anomalies.

- *Normalization is the process of organizing the data in the database.*
- *Normalization is used to minimize the redundancy from a relation or set of relations. It is also used to eliminate undesirable characteristics like Insertion, Update, and Deletion Anomalies.*
- *Normalization divides the larger table into smaller and links them using relationships.*
- *The normal form is used to reduce redundancy from the database table.*

Normalization

The main reason for normalizing the relations is removing these anomalies.

- ***Advantages of Normalization***
  - *Normalization helps to minimize data redundancy.*
  - *Greater overall database organization.*
  - *Data consistency within the database.*
  - *Much more flexible database design.*
  - *Enforces the concept of relational integrity.*
- ***Disadvantages of Normalization***
  - *It is very time-consuming and difficult to normalize relations of a higher degree.*
  - *Careless decomposition may lead to a bad database design, leading to serious problems.*

Normalization

Need for Normalization:

Normalization is used to reduce data redundancy. It provides a method to remove the following anomalies from the database and bring it to a more consistent state:

- *A **database anomaly** is a flaw in the database that occurs because of poor planning and redundancy.*
  - ***Insertion anomalies:** This occurs when we are not able to insert data into a database because some attributes may be missing at the time of insertion.*
  - ***Updation anomalies:** This occurs when the same data items are repeated with the same values and are not linked to each other.*
  - ***Deletion anomalies:** This occurs when deleting one part of the data deletes the other necessary information from the database.*

Normalization

Database Normal Forms.

The Theory of Data Normalization in MySQL server is still being developed further, there are discussions even on 6th Normal Form.

- *However, in most practical applications, normalization achieves its best in 3rd Normal Form. The evolution of Normalization in SQL theories is illustrated below-*
  - *1NF (First Normal Form)*
  - *2NF (Second Normal Form)*
  - *3NF (Third Normal Form)*
  - *BCNF (Boyce-Codd Normal Form)*
  - *4NF (Fourth Normal Form)*
  - *5NF (Fifth Normal Form)*
  - *6NF (Sixth Normal Form)*

1NF	2NF	3NF	BCNF
Atomic	Functional Dependency not Partial Dependency	Not Transitive Dependency	X → Y X is Super Key 0% redundancy (due to Functional Dependency)

Normalization

Database Normal Forms.

To understand the above-mentioned normal forms, we first need to have an understanding of the functional dependencies.

- *There are four types of normal forms that are usually used in relational databases*
  - ***1NF:*** *A relation is in 1NF if all its attributes have an atomic value.*
  - ***2NF:*** *A relation is in 2NF if it is in 1NF and all non-key attributes are fully functional dependent on the candidate key.*
  - ***3NF:*** *A relation is in 3NF if it is in 2NF and there is no transitive dependency.*
  - ***BCNF:*** *A relation is in BCNF if it is in 3NF and for every Functional Dependency, LHS is the super key.*

Functional dependency

Functional dependency .

The functional dependency is a relationship that exists between two attributes. It typically exists between the primary key and non-key attribute within a table.

- *The left side of FD is known as a determinant, the right side of the production is known as a dependent.*
- *It is denoted by X -> Y, where X is called a determinant and Y is called dependent.*
- ***Example:***
  - *Assume we have an employee table with attributes: Emp_Id, Emp_Name, Emp_Address.*
  - *Functional dependency can be written as: Emp_Id → Emp_Name*
  - *We can say that Emp_Name is functionally dependent on Emp_Id.*
  - *Here Emp_Id attribute can uniquely identify the Emp_Name attribute of employee table because if we know the Emp_Id, we can tell that employee name associated with it.*

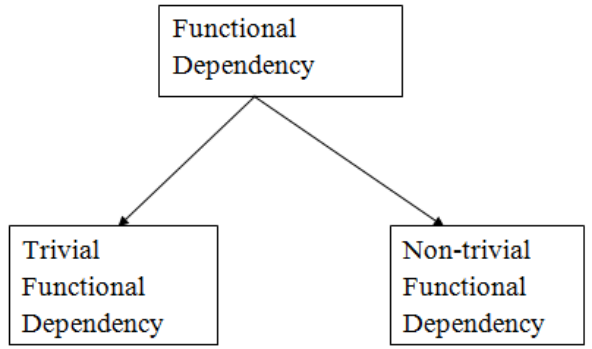
Functional dependency

Functional dependency .

The functional dependency is a relationship that exists between two attributes. It typically exists between the primary key and non-key attribute within a table.

- *Types of Functional dependency*
  - *Trivial functional dependency*
  - *Non-Trivial functional dependency*
  - *Multivalued functional dependency*
  - *Transitive functional dependency.*

roll_no	name	age
42	abc	17
43	pqr	18
44	xyz	18



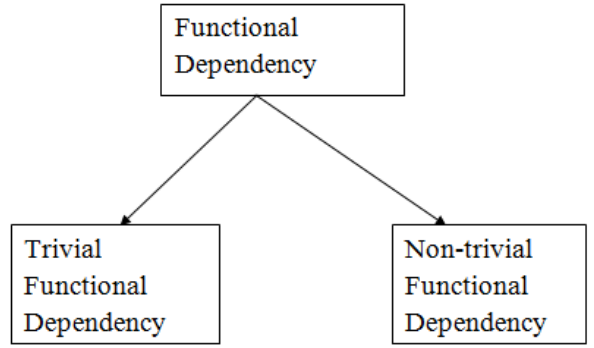
- *Here, {roll_no, name} → name is a trivial functional dependency, since the dependent name is a subset of determinant set {roll_no, name}*
- *roll_no → name is a non-trivial functional dependency, since the dependent name is not a subset of determinant roll_no.*
- *If a → {b, c} and there exists no functional dependency between b and c, then it is called a multivalued functional dependency.*
- *If a → b & b → c, then according to axiom of transitivity, a → c. This is a transitive*

Functional dependency

Functional dependency .

The functional dependency is a relationship that exists between two attributes. It typically exists between the primary key and non-key attribute within a table.

- *Types of Functional dependency*
- *Trivial functional dependency*
  - *A → B has trivial functional dependency if B is a subset of A.*
  - *The following dependencies are also trivial like: A → A, B → B*
- *Non-trivial functional dependency*
  - *A → B has a non-trivial functional dependency if B is not a subset of A.*
  - *When A intersection B is NULL, then A → B is called as complete non-trivial.*
- *In Multivalued functional dependency: entities of the dependent set are not dependent on each other. i.e. If a → {b, c} and there exists no functional dependency between b and c, then it is called a multivalued functional dependency.*



Inference Rule (IR):

Inference Rule (IR):.

The Functional dependency has 6 types of inference rule:.

- ***Reflexive Rule (IR1):****In the reflexive rule, if Y is a subset of X, then X determines Y.*

  If X ⊇ Y then X → Y

- ***Augmentation Rule (IR2):****The augmentation is also called as a partial dependency. In augmentation, if X determines Y, then XZ determines YZ for any Z.* ***If X → Y then XZ → YZ***

- ***Transitive Rule (IR3):****In the transitive rule, if X determines Y and Y determine Z, then X must also determine Z.*

  If X → Y and Y → Z then X → Z

- ***Union Rule (IR4):****Union rule says, if X determines Y and X determines Z, then X must also determine Y and Z.*

  If X → Y and X → Z then X → YZ

- ***Decomposition Rule (IR5):****Decomposition rule is also known as project rule. This Rule says, if X determines Y and Z, then X determines Y and X determines Z separately.*

  If X → YZ then X → Y and X → Z

- ***Pseudo transitive Rule (IR6):****In Pseudo transitive Rule, if X determines Y and YZ determines W, then XZ determines W.* ***If X → Y and YZ → W then XZ → W***

Normal Form :

First Normal Form (1NF):

1NF: A relation is in 1NF if all its attributes have an atomic value.

- *A relation will be 1NF if it contains an atomic value.*
- *It states that an attribute of a table cannot hold multiple values. It must hold only single-valued attribute.*
- *First normal form disallows the multi-valued attribute, composite attribute, and their combinations.*

S_ID	S_NAME	S_PHONE	S_COURSE
101	SUNNY	72XXXX9064 738XXXX238	BCA
102	PRAJWAL	857XXXX832	BBA

S_ID	S_NAME	S_PHONE	S_COURSE
101	SUNNY	72XXXX9064	BCA
101	SUNNY	738XXXX238	BCA
102	PRAJWAL	857XXXX832	BBA

Normal Form :

Second Normal Form (2NF):

2NF: The first condition for the table to be in Second Normal Form is that the table has to be in First Normal Form. The table should not possess partial dependency.

- *In the 2NF, relational must be in 1NF.*
- *In the second normal form, all non-key attributes are fully functional dependent on the primary key*

S_ID	S_NAME	S_COURSE
101	SUNNY	BCA
102	SHAILU	BCA
103	PRAJWAL	BBA

Second Normal Form (2NF)

S_ID	S_NAME	C_ID
101	SUNNY	C01
102	SHAILU	C01
103	PRAJWAL	CO2

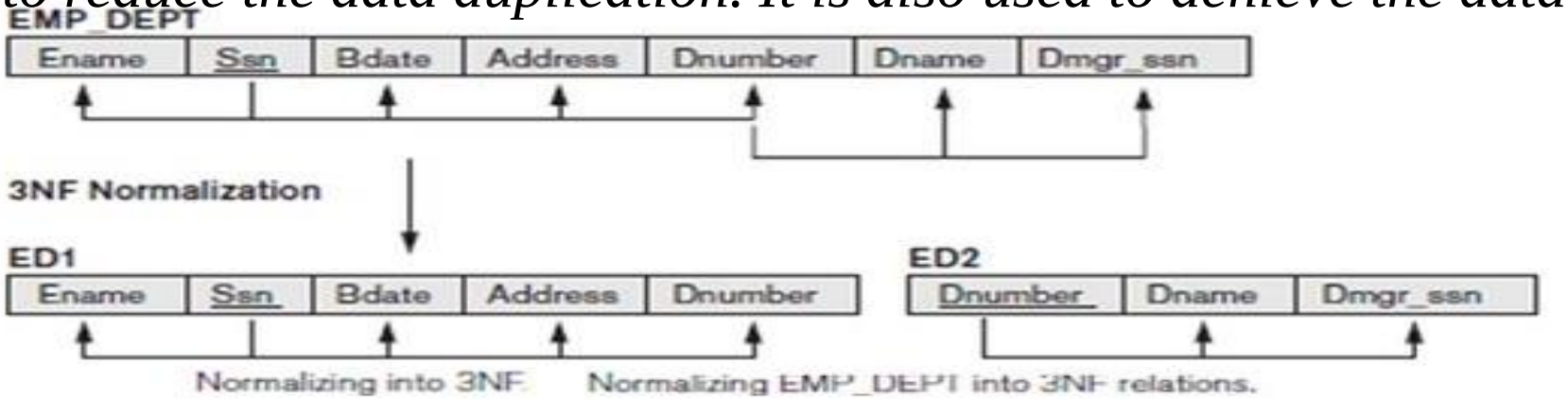
C_ID	S_COURSE
C01	BCA
C01	BCA
CO2	BBA

Normal Form :

Third Normal Form (3NF):

3NF: The third Normal Form ensures the reduction of data duplication. It is also used to achieve data integrity.

- *The third Normal Form ensures the reduction of data duplication. It is also used to achieve data integrity.*
- *A relation will be in 3NF if it is in 2NF and not contain any transitive partial dependency.*
- *3NF is used to reduce the data duplication. It is also used to achieve the data integrity.*



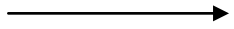
Normal Form :

Third Normal Form (3NF):

3NF: A relation is in 3NF if it is in 2NF and there is no transitive dependency.

- *The third Normal Form ensures the reduction of data duplication. It is also used to achieve data integrity.*
- *In the student table, stu_id determines subid, and subid determines sub. Therefore, stu_id determines sub via subid. This implies that the table possesses a transitive functional dependency, and it does not fulfill the third normal form criteria..*

stu_id	name	subid	sub	address
1	Arun	11	SQL	Delhi
2	Varun	12	Java	Bangalore
3	Harsh	13	C++	Delhi
4	Keshav	12	Java	Kochi



stu_id	name	subid	address
1	Arun	11	Delhi
2	Varun	12	Bangalore
3	Harsh	13	Delhi
4	Keshav	12	Kochi

subid	subject
11	SQL
12	java
13	C++
12	Java

Normal Form

Boyce Codd normal form (BCNF):

BCNF: A relation is in BCNF if it is in 3NF and for every Functional Dependency, LHS is the super key.

- *BCNF is the advance version of 3NF. It is stricter than 3NF.*
- *A table is in BCNF if every functional dependency X → Y, X is the super key of the table.*
- *For BCNF, the table should be in 3NF, and for every FD, LHS is super key.*
- *To transform the table into the BCNF, you will divide the table into two parts. One table will hold stuid which already exists and the second table will hold a newly created column profid.*

stuid	subject	professor
1	SQL	Prof. Mishra
2	Java	Prof. Anand
2	C++	Prof. Kanth
3	Java	Prof. James
4	DBMS	Prof. Lokesh

	stuid	profid
▶	1	101
	2	102
	2	103
	3	102
	4	104

	profid	subject	professor
▶	1	SQL	Prof. Mishra
	2	Java	Prof. Anand
	2	C++	Prof. Kanth
	3	Java	Prof. James
	4	DBMS	Prof. Lokesh

Normal Form

Fourth normal form (4NF):

4NF: A relation will be in 4NF if it is in Boyce Codd normal form and has no multi-valued dependency.

- *For a dependency A → B, if for a single value of A, multiple values of B exists, then the relation will be a multi-valued dependency.*
- *The given STUDENT table is in 3NF, but the COURSE and HOBBY are two independent entity. Hence, there is no relationship between COURSE and HOBBY.*
- *So to make the Student table into 4NF, we can decompose it into two tables Course & Hobby tables:*

STU_ID	COURSE	HOBBY
21	Computer	Dancing
21	Math	Singing
34	Chemistry	Dancing
74	Biology	Cricket

→

STU_ID	COURSE
21	Computer
21	Math
34	Chemistry
74	Biology

STU_ID	HOBBY
21	Dancing
21	Singing
34	Dancing
74	Cricket

Normal Form

Fifth normal form (5NF):

4NF: A relation is in 5NF if it is in 4NF and not contains any join dependency and joining should be lossless.

- *5NF is satisfied when all the tables are broken into as many tables as possible in order to avoid redundancy.*
- *5NF is also known as Project-join normal form (PJ/NF).*
- *So to make the SPC table into 4NF, we can decompose it into three tables SP, SC & PC tables:*

Table: SPC

Suplier	Product	Customer
Ali	ABC	Nauman
Ahmad	KLM	Qasim
Haider	XYZ	Saim

Table: SP

Suplier	Product
Ali	ABC
Ahmad	KLM
Haider	XYZ

Table: SC

Suplier	Customer
Ali	Nauman
Ahmad	Qasim
Haider	Saim

Table: PC

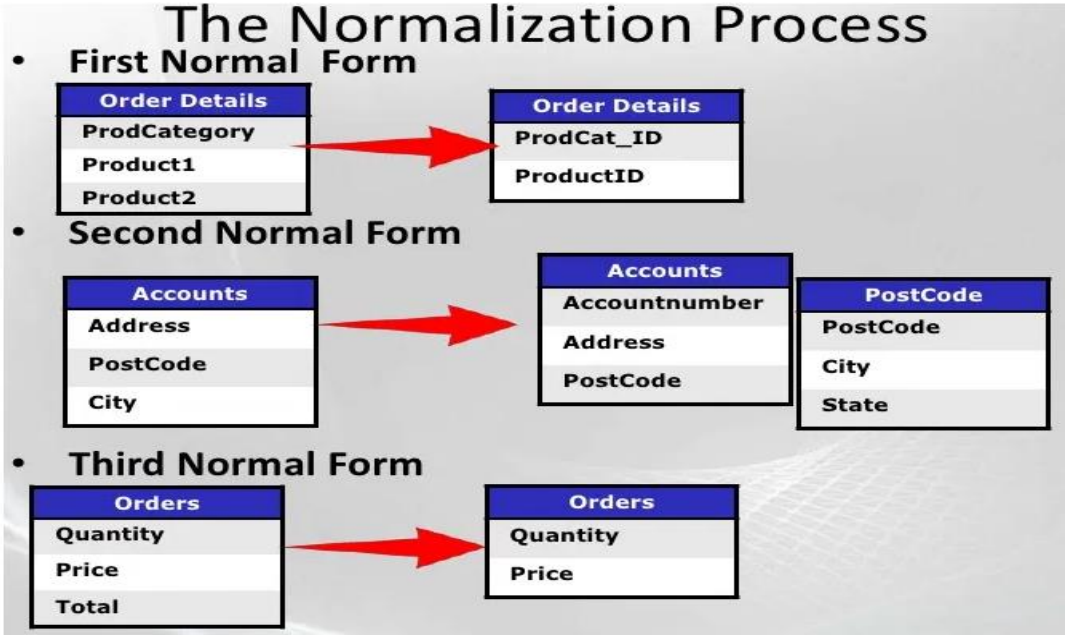
Product	Customer
ABC	Nauman
KLM	Qasim
XYZ	Saim

Normal Form

Normalization:

Normalization is a step by step process of removing the different kinds of redundancy and anomaly one step at a time from the database.

- 1st normal form
  - All attributes depend on **the key**
- 2nd normal form
  - All attributes depend on **the whole key**
- 3rd normal form
  - All attributes depend on **nothing but the key**



Discussion

Queries ?

Prof. K. Adisesha

9449081542

Thank you