Normalization

Anomaly

- Anomaly is mismatch / inconsistencies in the data

- Types of Anomaly

    Insertion Anomaly

    Updation Anomaly

    Deletion Anomaly

Functional Dependency

All the non-key Columns of a table are said to be dependent on Primary key, which uniquely identifies the rows in a table.

For example:

Consider a CUSTOMER table with attributes:

Field	Key
Cust_id	PRI
Name	
Address	
State_code	
Telephone	

Cust_Id, Name, Address, State, Telephone.

Here **Cust_Id** is a primary Key, which uniquely identifies the remaining columns.

Hence, the non-key columns are said to be functionally dependent on Primary Key.

Normalization

- Normalization is the process of organizing the data attributes with their relationships.
- Normalization minimizes the redundancy of data rows.
- Normalization minimizes the dependency of columns.
- Anomalies (flaws) such as inserting, updating and deleting are eliminated.
- Normalization divides the larger table into smaller ones and establishes entity relationships among the smaller ones.

Normalization

- Normalization is evolved into several stages over a period of time.

Normal Form	Description
1NF	A relation is in 1NF if it contains an atomic value.
2NF	A relation will be in 2NF if it is in 1NF and all non-key attributes are fully ***functional*** dependent on the primary key.
3NF	A relation will be in 3NF if it is in 2NF and no ***transition*** dependency exists.

First Normal Form (1NF)

- A table is said to be in *First Normal Form*, when it follows the following 4 rules:

- A column/attribute in a table should consists of a scalar atomic value.
- All the values stored in a column should have same business term.
- All the columns in a table should be represented with unique names.
- The order of rows and columns doesn't matter.

First Normal Form (1NF) - Example

Raw Data:

Student_Details	Course_details		Pre-requisite	Result_details	
0101 Tim 11/4/1985	M1 Advance maths	7	Basic Math	02/11/2015 82	A
0102 Rob 10/04/1986	P4 Advance Physics	8	Basic Physics	21/11/2015 89	A
0103 Mary 11/07/1985	B3 Advance Biology	10	Basic Biology	12/11/2015 62	B
0104 Rob 10/04/1986	H6 Advance History	9	Basic History	21/11/2015 89	A
0105 Tom 03/08/1988	C3 Advance Chemistry	11	Basic Biology	12/11/2015 50	C

Normalization before and after 1 - NF

Student#	Student_Name	DOB	Course#	CourseName	Pre Requisite	Duration in days	Date of Exam	Marks	Grade
0101	Tim	11/4/1985	M1	Advance Math	Basic Math	7	02/11/2015	82	A
0102	Rob	10/04/1986	P4	Advance Physics	Basic Physics	8	21/11/2015	89	A
0103	Mary	11/07/1985	B3	Advance Biology	Basic Biology	10	12/11/2015	62	B

Second Normal Form (2NF)

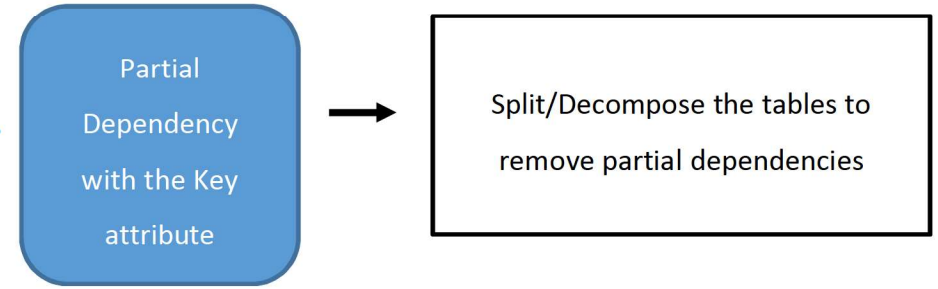
- Each and every non-key & independent attribute has a functional dependency on the primary key.
- 2NF is evolutional after 1NF , and handled the redundancy of data effectively.
- Below rules are followed to achieve 2NF.
- In 2NF, the data must be in 1NF.

Tables after 2-NF

- Student# ,Course# → Marks
- Student#, Course# → Grade
- Marks → Grade
- Student# → StudentName, DOB
- Course# → CourseName, Pre-Requisite,
- DurationDays, Date of exam

Second Normal Form – (2NF)

- Student# ,Course# → Marks
- Student#, Course# → Grade
- Marks → Grade
- Student# → StudentName, DOB
- Course# → CourseName, Pre-Requisite,
- DurationDays, Date of exam



Second Normal Form – (2NF)

Student Table

Student#	Student_Name	Date Of Birth
0101	Tim	11/4/1985
0102	Rob	10/04/1986
0103	Mary	11/07/1985

Result Table

Student#	Course#	Marks	Grade
0101	M1	82	A
0102	P4	89	A
0103	B3	62	B

Course Table

Course#	CourseName	Prerequisite	Durationindays	Date Of Exam
M1	Advance Math	Basic Math	7	02/11/2015
P4	Advance Physics	Basic Physics	8	21/11/2015
B3	Advance Biology	Basic Biology	10	12/11/2015

Third Normal Form (3-NF):

To achieve 3-NF,

- The attributes of a table should already be in 2-NF, so full functional dependency was achieved already in 2- NF.
- If any non-prime have transitive dependency, this is resolved with 3 - NF.
- Transitive dependency is like attribute - A depends on attribute - B, then attribute - B depends on attribute - C and so on

Third Normalization – (3NF)

Result_table

Student#	Course#	Marks	Grade
0101	M1	82	A
0102	P4	89	A
0103	B3	62	B

Student# ,Course# → Marks

Student#, Course# → Grade

Marks → Grade

Student#,Course#→ Marks→ Grade:TD

Summary

This slide discuss the anomalies and its types in the SQL.

When a correlation is generated directly from a user view, anomalies arise. Anomalies may appear as updates, deletions, or insertions. It is a data inconsistency that arises from partial updating and redundancy.

Thank You