DATABASE SYSTEMS

Dr. Noha Nagy

Lecture 17

Normalization

DB Lifecycle

- System Requirements
- DB Design
- Normalization "Schema Refinement”
- Relational Model

Problems without Normalization

normalization is

a technique of organizing the data into multiple related tables, to minimize DATA REDUNDANCY.

What is
Data Redundancy?

and why should we reduce it?

Data Redundancy

Repetition increases the size of the database

Other issues like:

- Insertion problems
- Deletion problems
- Update problems

Insertion Anomaly

- To insert redundant data for every new row (of Student data in our case) is a data insertion problem.

STUDENTS TABLE

rollno	name	branch	hod	office_tel
1	Akon	CSE	Mr. X	53337
2	Bkon	CSE	Mr. X	53337
3	Ckon	CSE	Mr. X	53337
4	Dkon	CSE	Mr. X	53337

Deletion Anomaly

STUDENTS TABLE

rollno	name	branch	hod	office_tel
1	Akon	CSE	Mr. X	53337
2	Bkon	CSE	Mr. X	53337
3	Ckon	CSE	Mr. X	53337
4	Dkon	CSE	Mr. X	53337

STUDENTS TABLE

rollno	name	branch	hod	office_tel
1	Akon	CSE	Mr. X	53337

- Delete some students
- We automatically delete the branch information
- We have not stored the branch information anywhere else

Update Anomaly

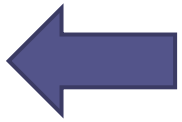
STUDENTS TABLE

rollno	name	branch	hod	office_tel
1	Akon	CSE	Mr. X	53337
2	Bkon	CSE	Mr. X	53337
3	Ckon	CSE	Mr. X	53337
4	Dkon	CSE	Mr. X	53337

Mr. X leaves, and Mr. Y joins as the new HOD for CSE

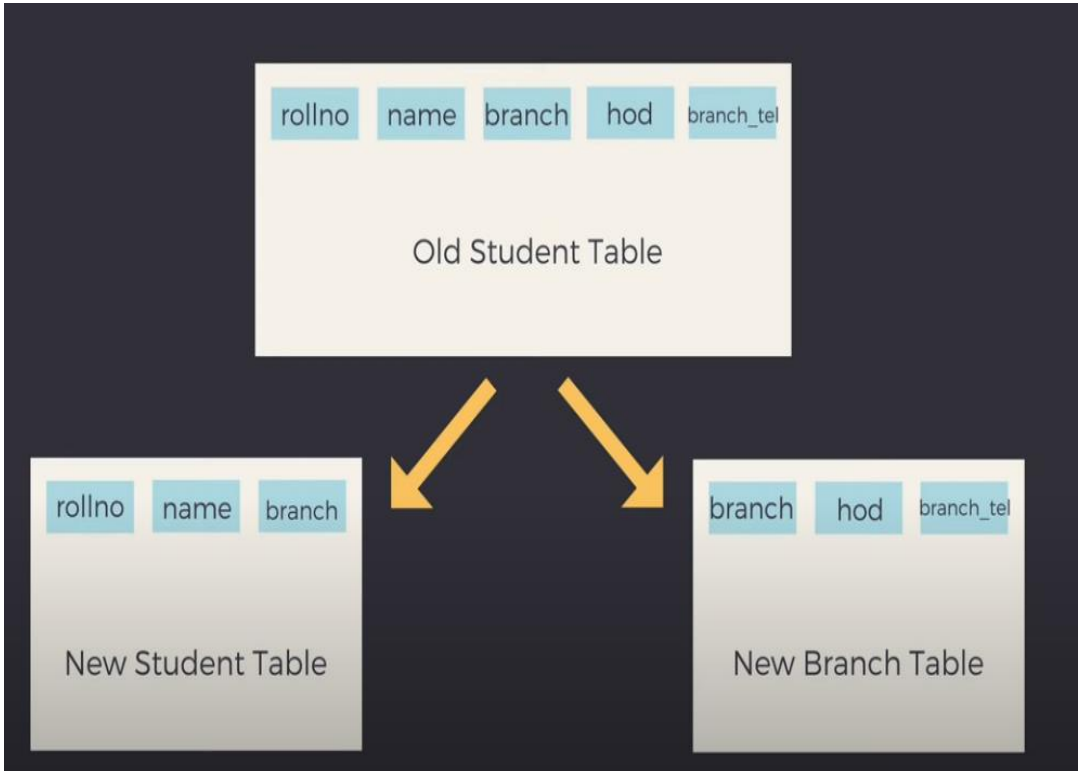
STUDENTS TABLE

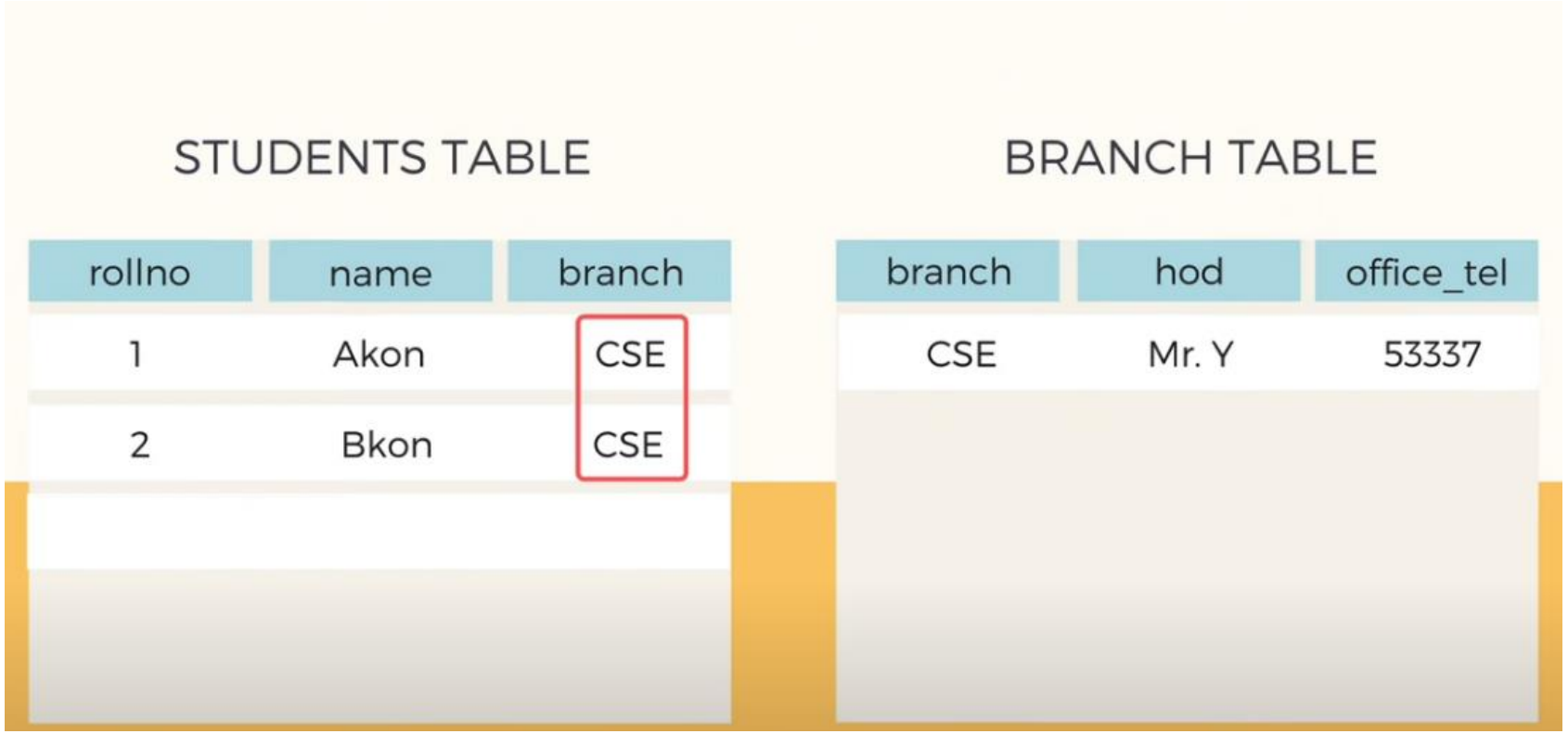
rollno	name	branch	hod	office_tel
1	Akon	CSE	~~Mr. X~~ Mr. Y	53337
2	Bkon	CSE	~~Mr. X~~ Mr. Y	53337
3	Ckon	CSE	Mr. X	53337
4	Dkon	CSE	~~Mr. X~~ Mr. Y	53337



How to solve These Problems?

- Use Normalization





- Normalization not eliminating data redundancy but minimizing it

Database Tables and Normalization

- **Normalization** is a process for assigning attributes to entities. It reduces data redundancies and helps eliminate the data anomalies.
- Normalization works through a series of stages called normal forms:
  - **First normal form (1NF)**
  - **Second normal form (2NF)**
  - **Third normal form (3NF)**
  - **Boyce Codd Normal Form (BCNF)**
  - **Fourth normal form (4NF)**
- The highest level of normalization is not always desirable.

First Normal Form

1. Each table cell contain atomic value(indivisible)
2. Each table has a primary key that uniquely identifies records

- Primary key
  - cannot be NULL
  - must be unique
  - The primary key values should rarely be changed

Functional dependency

Functional Dependencies

- A *Functional Dependency* describes a relationship between *attributes* in a single relation.
- An attribute is *functionally dependant* on another if we can use the value of one attribute to determine the value of another.

Functional dependency

- We use the symbol '→' to indicate a functional dependency:

  X → Y

- is read "X functionally determines Y" or "Y functionally depends on X"
- more simply "X determines Y"
- or "Y depends on X"

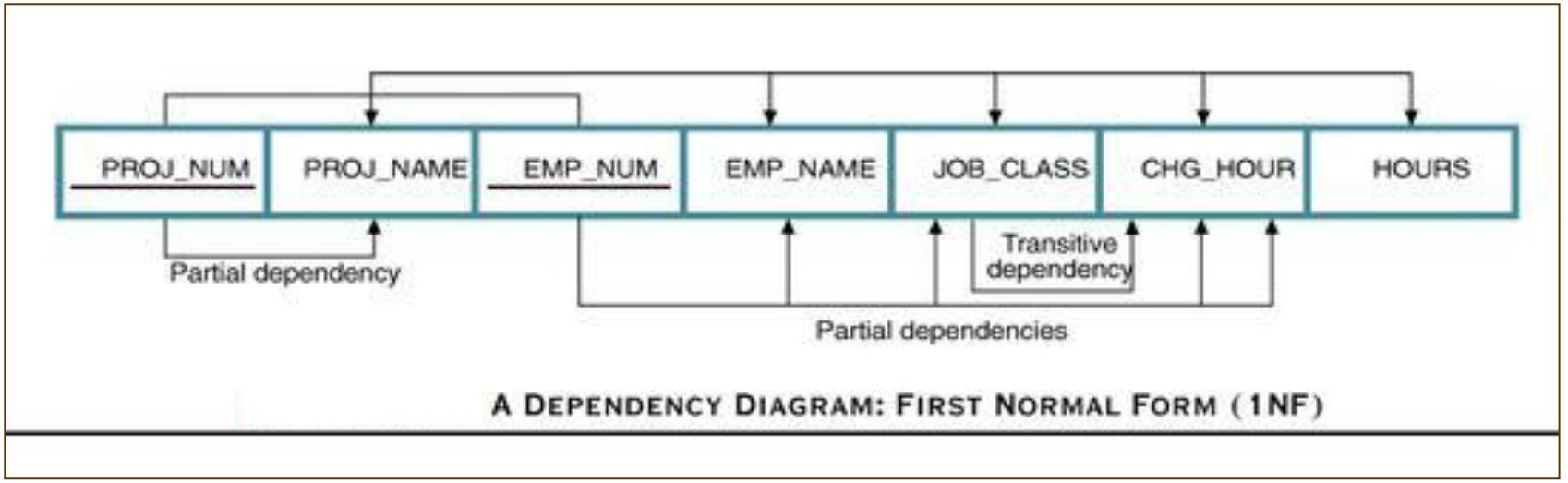
Functional dependency

Some Examples

- Student_ID → Saddress, SDoB
- Student_ID, Course# → Grade
- Model, Year → CarPrice
- Course_No, Section -> Professor, Classroom, Number of Student

Dependency Diagram

- Dependency Diagram
  - **The primary key components are bold, underlined, and shaded in a different color.**
  - **The arrows above entities indicate all desirable dependencies, i.e., dependencies that are based on PK.**
  - **The arrows below the dependency diagram indicate less desirable dependencies – partial dependencies and transitive dependencies.**



A DEPENDENCY DIAGRAM: FIRST NORMAL FORM (1NF)

2NF

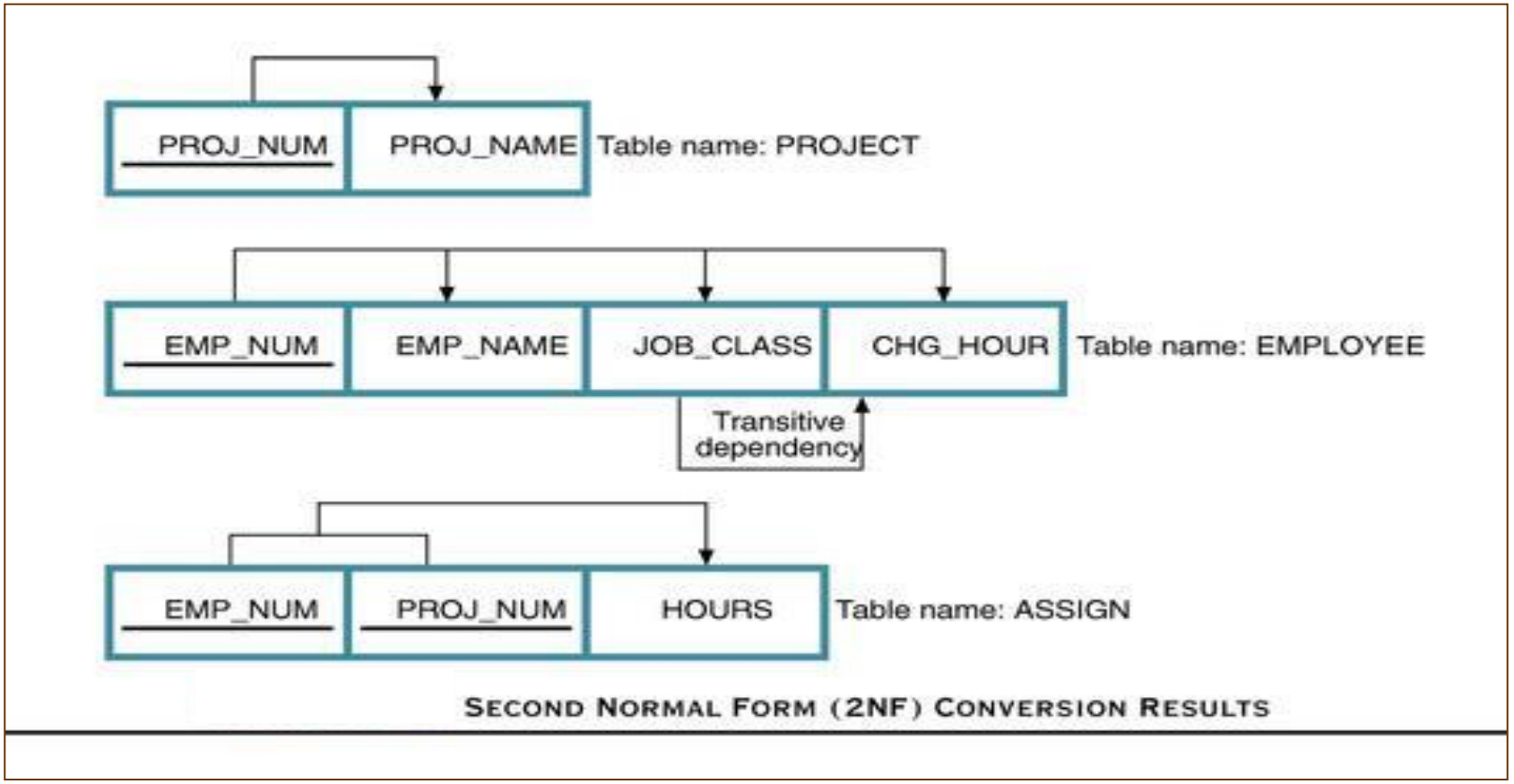
- 2NF Definition
  - **A table is in 2NF if:**
    - **It is in 1NF and**
    - **It includes no partial dependencies; that is, no attribute is dependent on only a portion of the primary key.**

      (It is still possible for a table in 2NF to exhibit transitive dependency; that is, one or more attributes may be functionally dependent on nonkey attributes.)

2NF

- **Conversion to Second Normal Form**
  - **Starting with the 1NF format, the database can be converted into the 2NF format by**
    - **Writing each key component on a separate line, and then writing the original key on the last line and**
    - **Writing the dependent attributes after each new key.**

Second Normal Form (2NF) Conversion Results



3NF

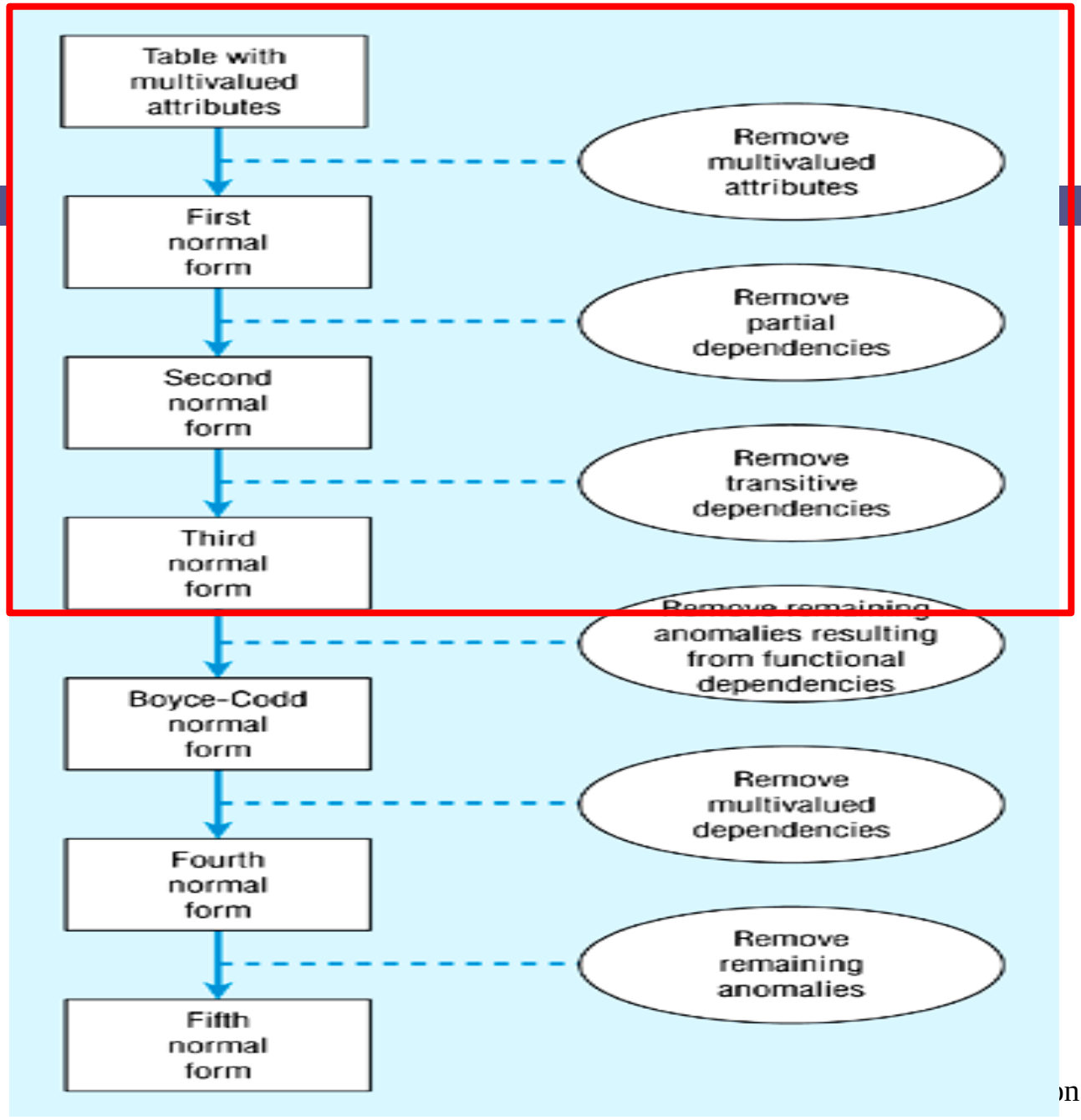
- 3NF Definition
  - **A table is in 3NF if:**
    - **It is in 2NF and**
    - **It contains no transitive dependencies.**

3NF

- Conversion to Third Normal Form
  - **Create a separate table with attributes in a transitive functional dependence relationship.**

PROJECT (PROJ_NUM, PROJ_NAME)
ASSIGN (PROJ_NUM, EMP_NUM, HOURS)
EMPLOYEE (EMP_NUM, EMP_NAME, JOB_CLASS)
JOB (JOB_CLASS, CHG_HOUR)

Normalization
Normal Forms



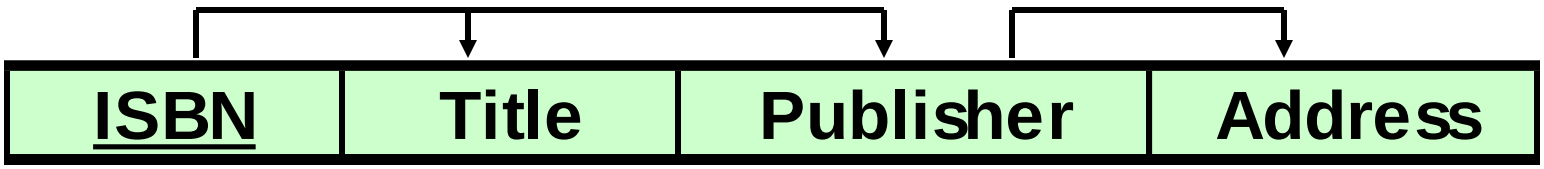
Normalization Examples

Example 1: Determine NF

- ISBN → Title
- ISBN → Publisher
- Publisher → Address

All attributes are directly or indirectly determined by the primary key; therefore, the relation is at least in 1 NF

BOOK

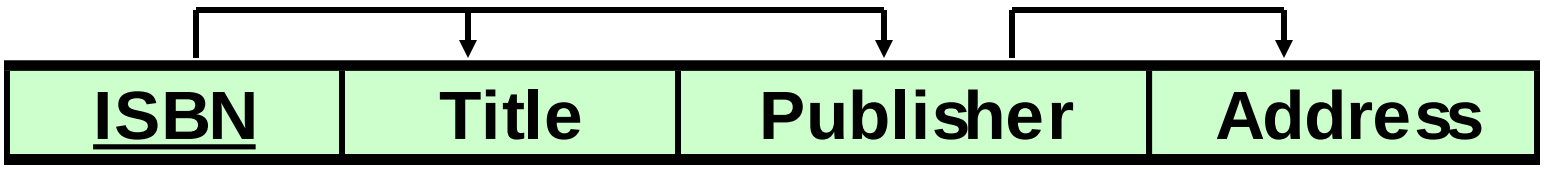


Example 1: Determine NF

- ISBN → Title
- ISBN → Publisher
- Publisher → Address

The relation is at least in 1NF. There is no COMPOSITE primary key, therefore there can't be partial dependencies. Therefore, the relation is at least in 2NF

BOOK



Example 1: Determine NF

- ISBN → Title
- ISBN → Publisher
- Publisher → Address

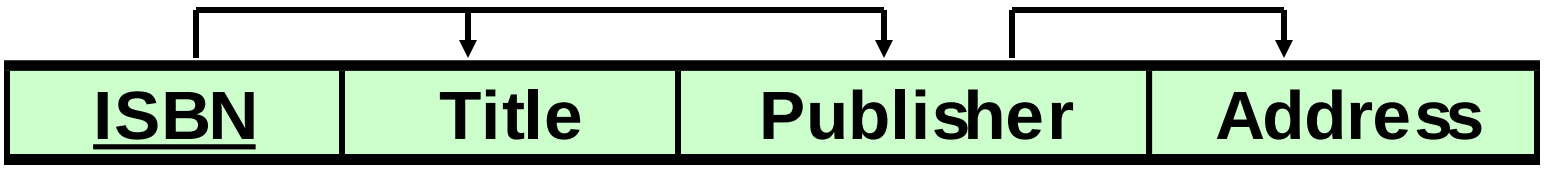
Publisher is a non-key attribute, and it determines Address, another non-key attribute. Therefore, there is a transitive dependency, which means that the relation is NOT in 3 NF.

Example 1: Determine NF

- ISBN → Title
- ISBN → Publisher
- Publisher → Address

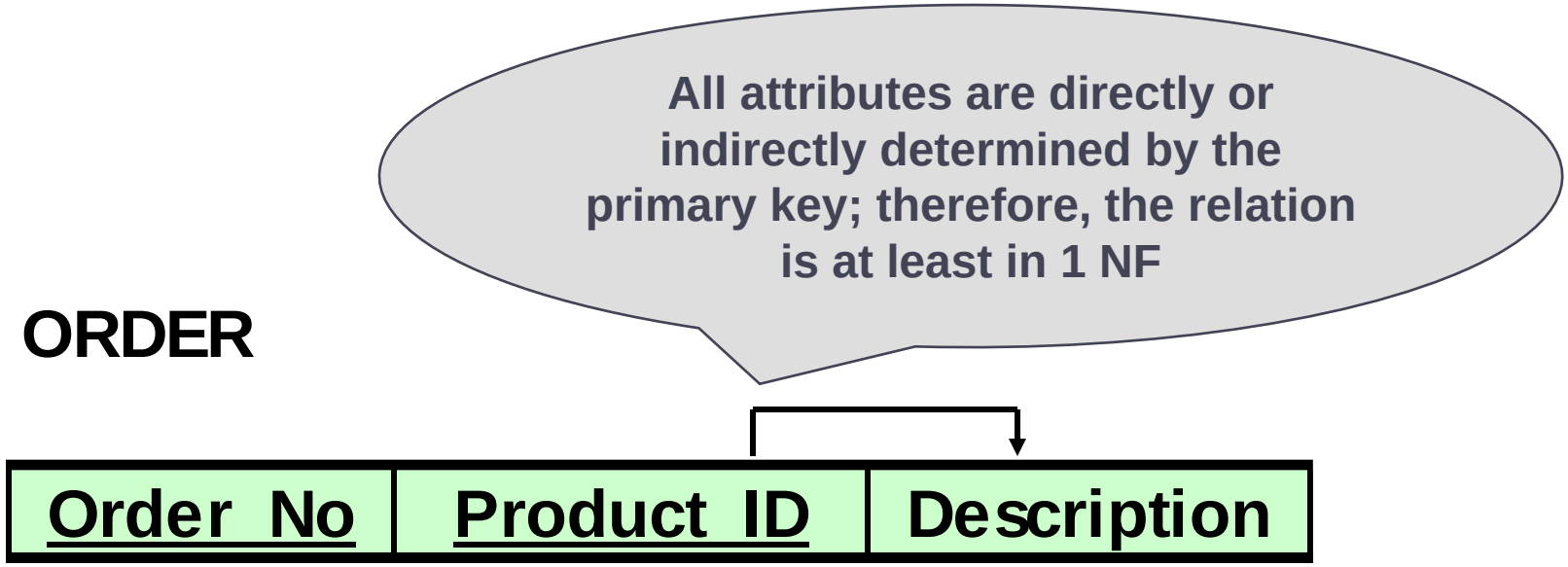
We know that the relation is at least in 2NF, and it is not in 3 NF. Therefore, we conclude that the relation is in 2NF.

BOOK



Example 2: Determine NF

- Product_ID → Description



Example 2: Determine NF

- Product_ID → Description

The relation is at least in 1NF.
There is a COMPOSITE Primary Key (PK) (Order_No, Product_ID), therefore there can be partial dependencies. Product_ID, which is a part of PK, determines Description; hence, there is a partial dependency. Therefore, the relation is not 2NF. No sense to check for transitive dependencies!

ORDER

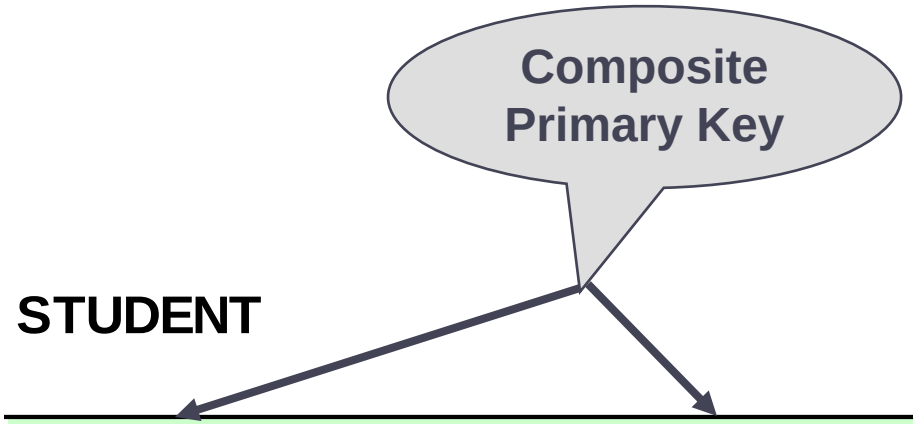
Example 2: Determine NF

- Product_ID → Description

We know that the relation is at least in 1NF, and it is not in 2 NF. Therefore, we conclude that the relation is in 1 NF.

ORDER

Bringing a Relation to 2NF

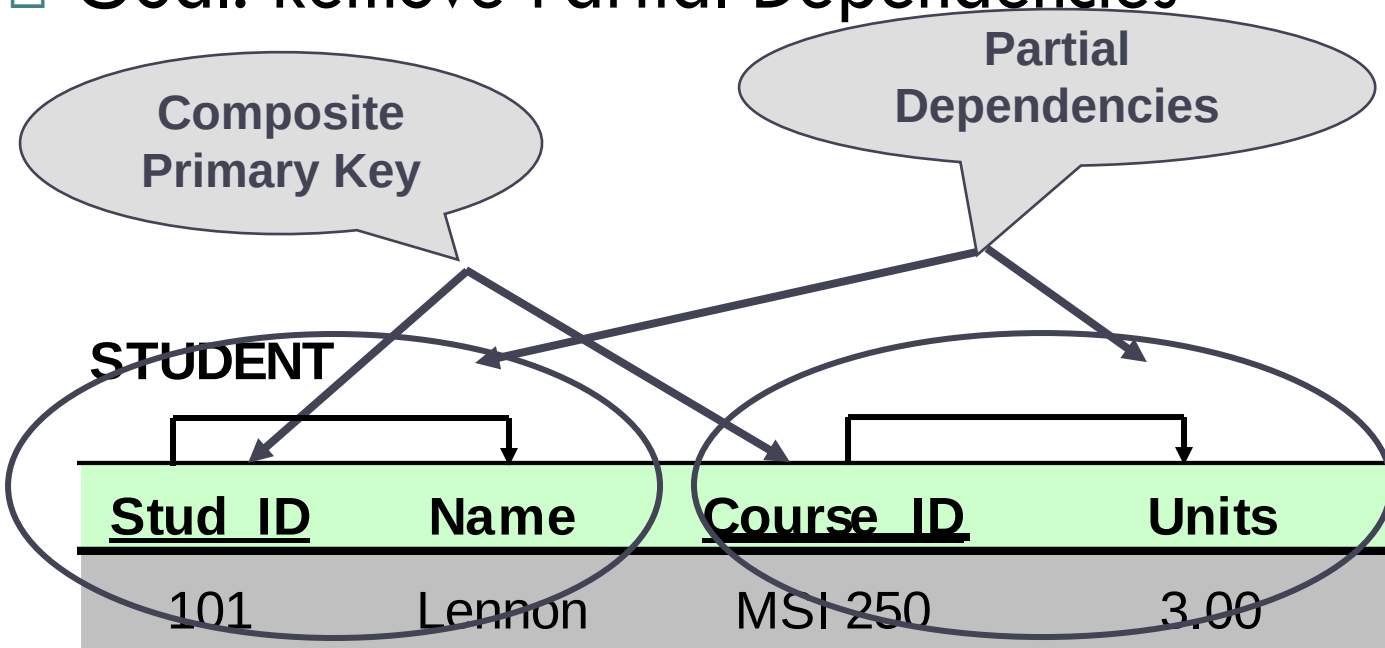


STUDENT

Stud_ID	Name	Course_ID	Units
101	Lennon	MSI 250	3.00
101	Lennon	MSI 415	3.00
125	Johnson	MSI 331	3.00

Bringing a Relation to 2NF

- Goal: Remove Partial Dependencies



Stud_ID	Name	Course_ID	Units
101	Lennon	MSI 250	3.00
101	Lennon	MSI 415	3.00
125	Johnson	MSI 331	3.00

Bringing a Relation to 2NF

- Remove attributes that are dependent from the part but not the whole of the primary key from the original relation. For each partial dependency, create a new relation, with the corresponding part of the primary key from the original as the primary key.

STUDENT

Stud_ID	Name	Course_ID	Units
101	Lennon	MSI 250	3.00
101	Lennon	MSI 415	3.00
125	Johnson	MSI 331	3.00

Bringing a Relation to 2NF

CUSTOMER

Stud_ID	Name	Course_ID	Units
101	Lennon	MSI 250	3.00
101	Lennon	MSI 415	3.00
125	Johnson	MSI 331	3.00

STUDENT_COURSE

Stud_ID	Course_ID
101	MSI 250
101	MSI 415
125	MSI 331

STUDENT

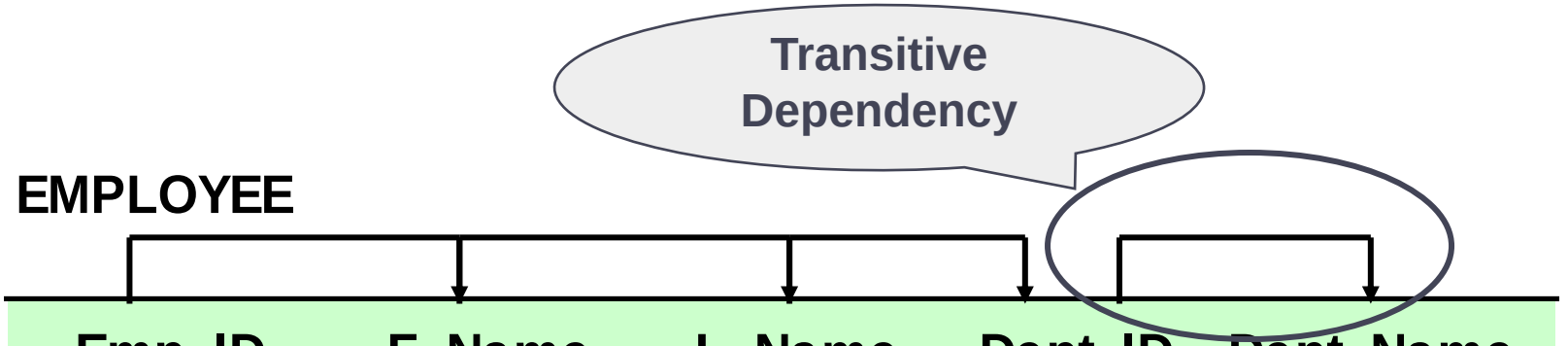
Stud_ID	Name
101	Lennon
101	Lennon
125	Johnson

COURSE

Course_ID	Units
MSI 250	3.00
MSI 415	3.00
MSI 331	3.00

Bringing a Relation to 3NF

- Goal: Get rid of transitive dependencies.

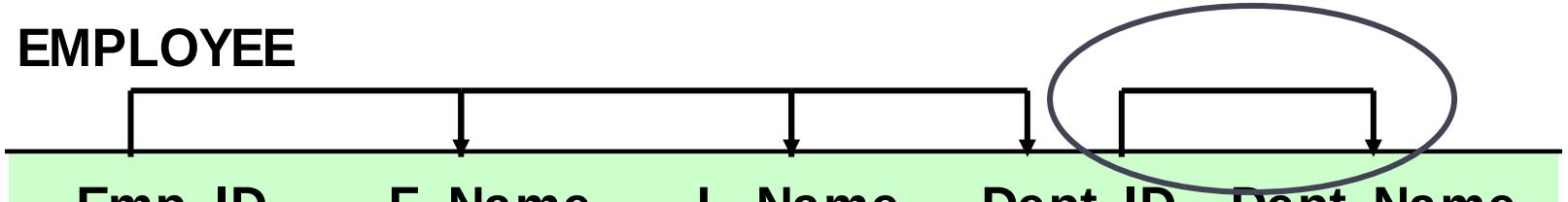


Emp_ID	F_Name	L_Name	Dept_ID	Dept_Name
111	Mary	Jones	1	Acct
122	Sarah	Smith	2	Mktg

Bringing a Relation to 3NF

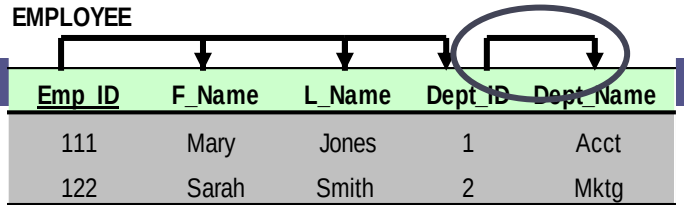
- Remove the attributes, which are dependent on a non-key attribute, from the original relation. For each transitive dependency, create a new relation with the non-key attribute which is a determinant in the transitive dependency as a primary key, and the dependent non-key attribute as a dependent.

EMPLOYEE



Emp_ID	F_Name	L_Name	Dept_ID	Dept_Name
111	Mary	Jones	1	Acct
122	Sarah	Smith	2	Mktg

Bringing a Relation to 3NF



EMPLOYEE

Emp_ID	F_Name	L_Name	Dept_ID
111	Mary	Jones	1
122	Sarah	Smith	2

DEPARTMENT

Dept_ID	Dept_Name
1	Acct
2	Mktg