


Normalization Examples

The background of the slide features abstract, overlapping geometric shapes in various shades of green, ranging from light lime to dark forest green. These shapes are primarily located on the right side and bottom, creating a modern, layered effect. A thin, light gray line also extends diagonally across the lower right portion of the slide.

Dependencies: Definitions

- ***Multivalued Attributes*** (or *repeating groups*): non-key attributes or groups of non-key attributes the values of which are not uniquely identified by (directly or indirectly) (not functionally dependent on) the value of the Primary Key (or its part).

STUDENT

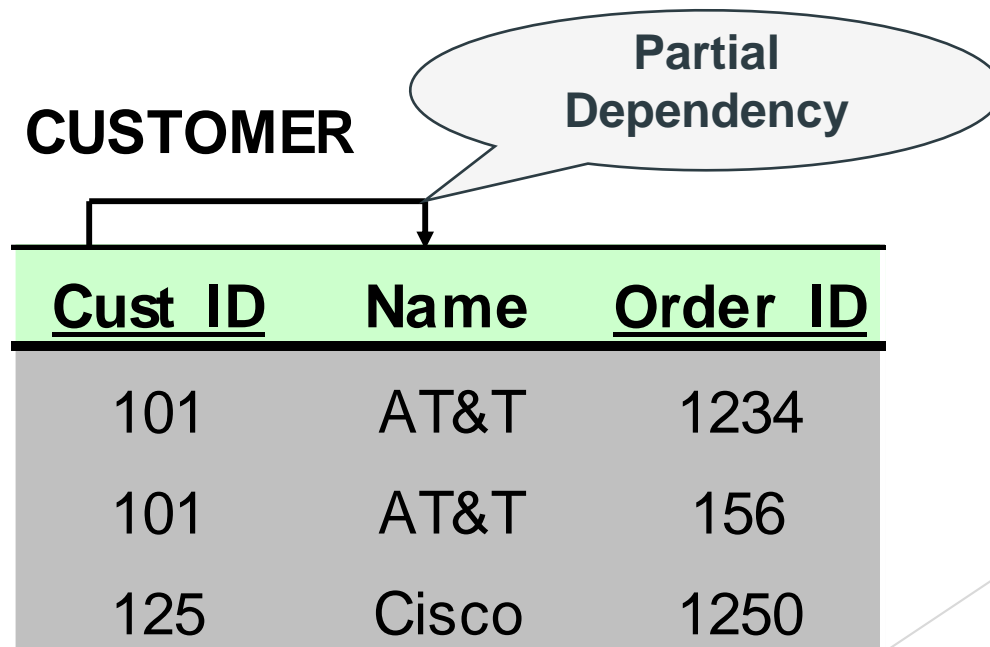


The diagram shows two horizontal arrows originating from the 'Stud ID' header. The first arrow points down to the 'Name' header, and the second arrow points down to the 'Course_ID' header, indicating functional dependencies.

<u>Stud ID</u>	Name	Course_ID	Units
101	Lennon	MSI 250	3.00
101	Lennon	MSI 415	3.00
125	Johnson	MSI 331	3.00

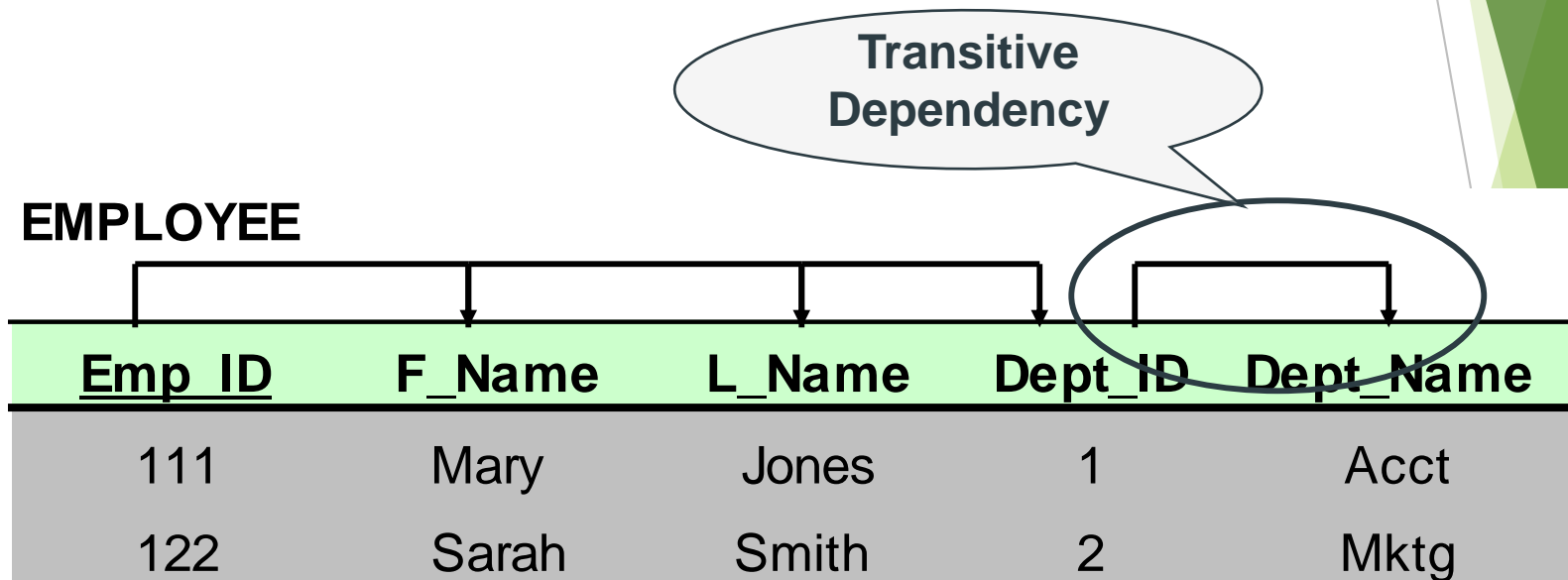
Dependencies: Definitions

- ***Partial Dependency*** - when a non-key attribute is determined by a part, but not the whole, of a **COMPOSITE** primary key.



Dependencies: Definitions

- **Transitive Dependency** - when a non-key attribute determines another non-key attribute.



Normal Forms: Review

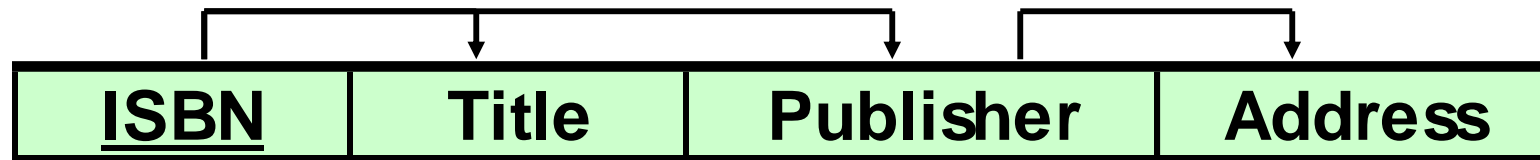
- ▶ Unnormalized - There are multivalued attributes or repeating groups
- ▶ 1 NF - No multivalued attributes or repeating groups.
- ▶ 2 NF - 1 NF plus no partial dependencies
- ▶ 3 NF - 2 NF plus no transitive dependencies

Example 1: Determine NF

- ▶ ISBN \rightarrow Title
- ▶ ISBN \rightarrow Publisher
- ▶ Publisher \rightarrow 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

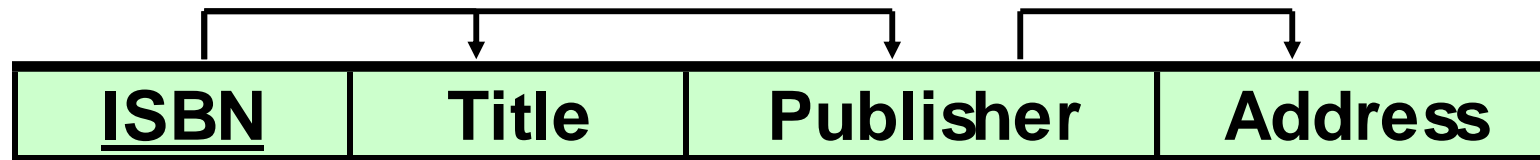
<u>ISBN</u>	Title	Publisher	Address
-------------	-------	-----------	---------

Example 1: Determine NF

- ▶ ISBN \rightarrow Title
- ▶ ISBN \rightarrow Publisher
- ▶ Publisher \rightarrow 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.

BOOK



Example 1: Determine NF

- ▶ ISBN \rightarrow Title
- ▶ ISBN \rightarrow Publisher
- ▶ Publisher \rightarrow Address

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

BOOK

<u>ISBN</u>	Title	Publisher	Address
-------------	-------	-----------	---------

Example 1: Determine NF

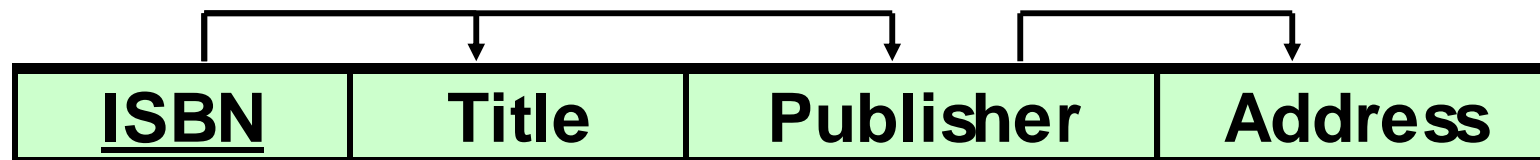
- ▶ ISBN \rightarrow Title
- ▶ ISBN \rightarrow Publisher
- ▶ Publisher \rightarrow Address

In your solution you will write the following justification:

- 1) No M/V attributes, therefore at least 1NF
- 2) No partial dependencies, therefore at least 2NF
- 3) There is a transitive dependency (Publisher \rightarrow Address), therefore, not 3NF

Conclusion: The relation is in 2NF

BOOK



Example 2: Determine NF

► Product_ID → Description

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

ORDER

<u>Order No</u>	<u>Product ID</u>	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

<u>Order No</u>	<u>Product ID</u>	Description
-----------------	-------------------	-------------

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

<u>Order No</u>	<u>Product ID</u>	Description
-----------------	-------------------	-------------

Example 2: Determine NF

► Product_ID → Description

In your solution you will write the following justification:

- 1) No M/V attributes, therefore at least 1NF
- 2) There is a partial dependency (Product_ID → Description), therefore not in 2NF

Conclusion: The relation is in 1NF

ORDER

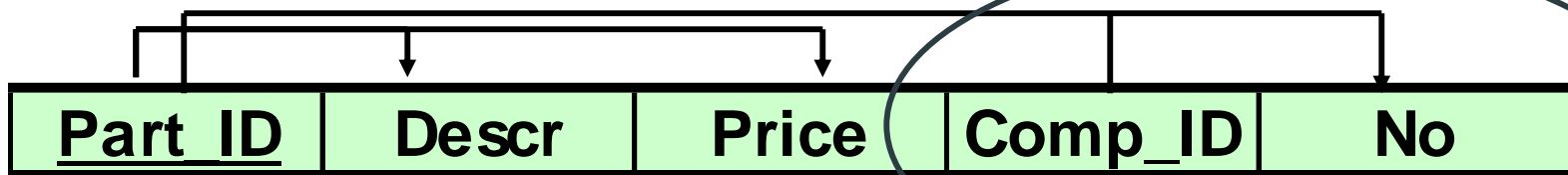
<u>Order No</u>	<u>Product ID</u>	Description
-----------------	-------------------	-------------

Example 3: Determine NF

- ▶ Part_ID → Description
- ▶ Part_ID → Price
- ▶ Part_ID, Comp_ID → No

Comp_ID and No are not determined by the primary key; therefore, the relation is NOT in 1 NF. No sense in looking at partial or transitive dependencies.

PART



Example 3: Determine NF

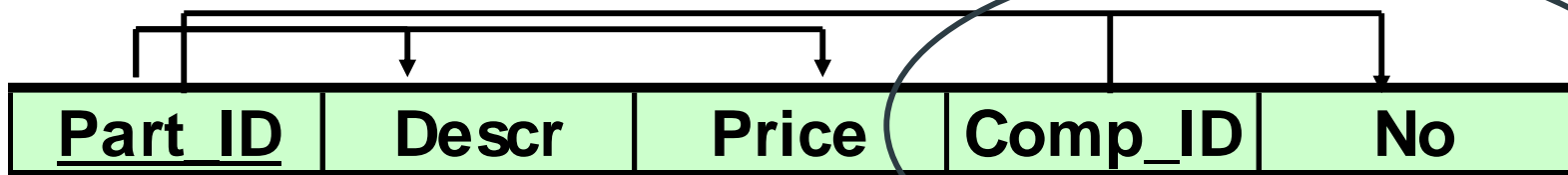
- ▶ Part_ID → Description
- ▶ Part_ID → Price
- ▶ Part_ID, Comp_ID → No

In your solution you will write the following justification:

- 1) There are M/V attributes; therefore, not 1NF

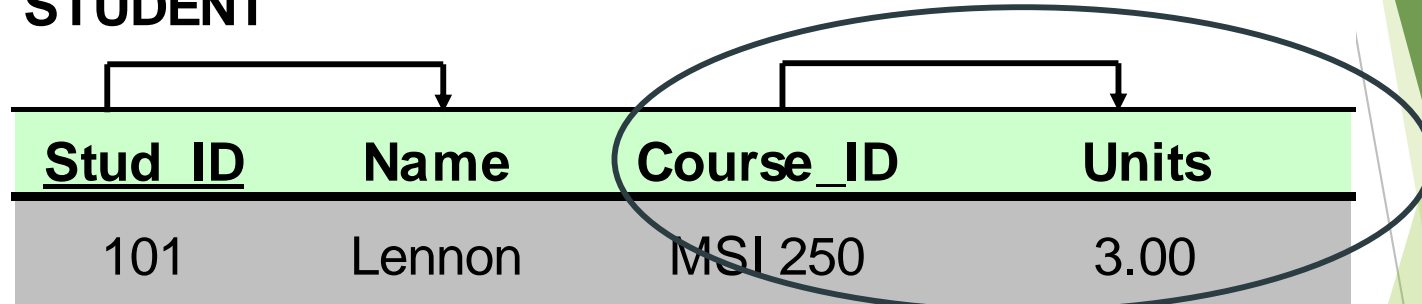
Conclusion: The relation is not normalized.

PART



Bringing a Relation to 1NF

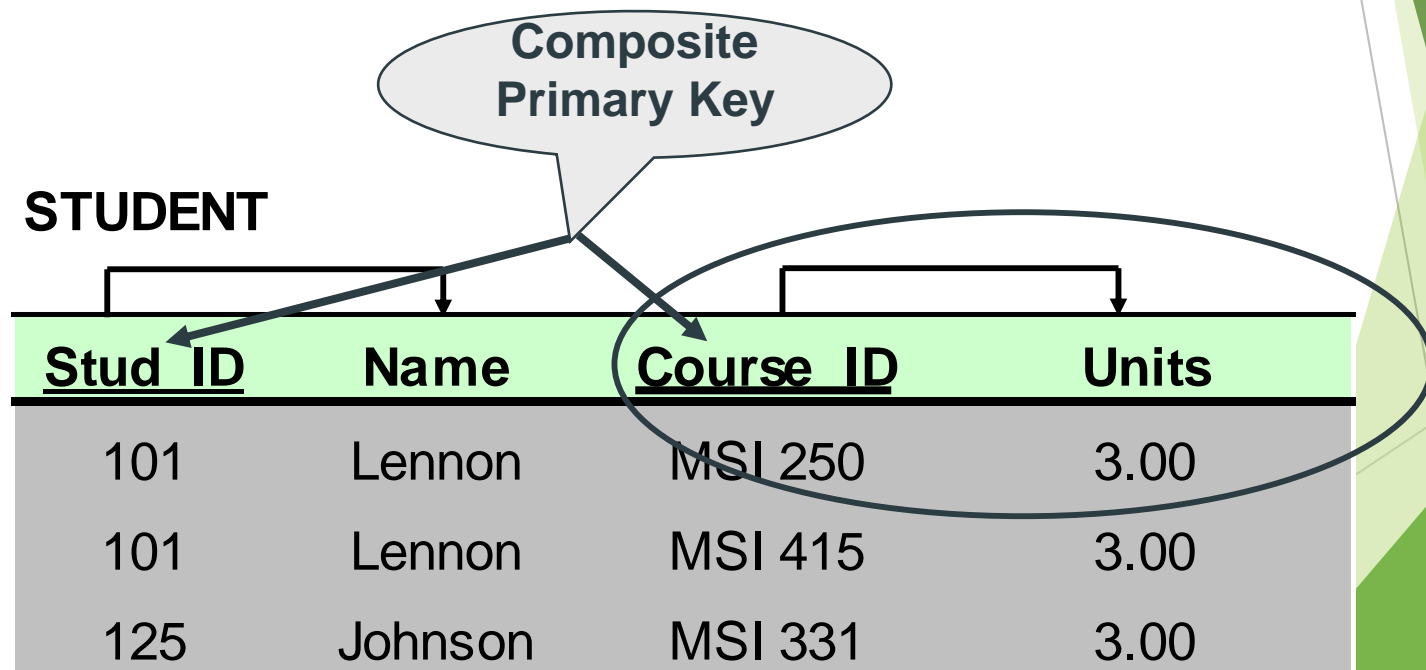
STUDENT



<u>Stud_ID</u>	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 1NF

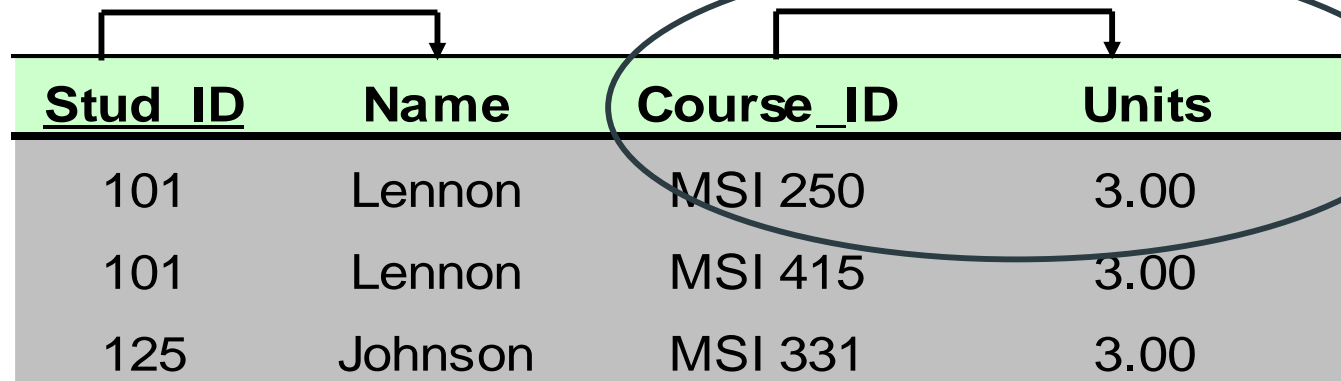
- Option 1: Make a determinant of the repeating group (or the multivalued attribute) a part of the primary key.



Bringing a Relation to 1NF

- Option 2: Remove the entire repeating group from the relation. Create another relation which would contain all the attributes of the repeating group, plus the primary key from the first relation. In this new relation, the primary key from the original relation and the determinant of the repeating group will comprise a primary key.

STUDENT




The diagram shows a table with four columns: Stud_ID, Name, Course_ID, and Units. The first two columns are grouped by a bracket above them. The last two columns are grouped by a bracket above them and are circled with an oval. An arrow points from the Stud_ID column to the Course_ID column, indicating a transformation where the primary key is moved to the new relation.

<u>Stud_ID</u>	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 1NF

STUDENT



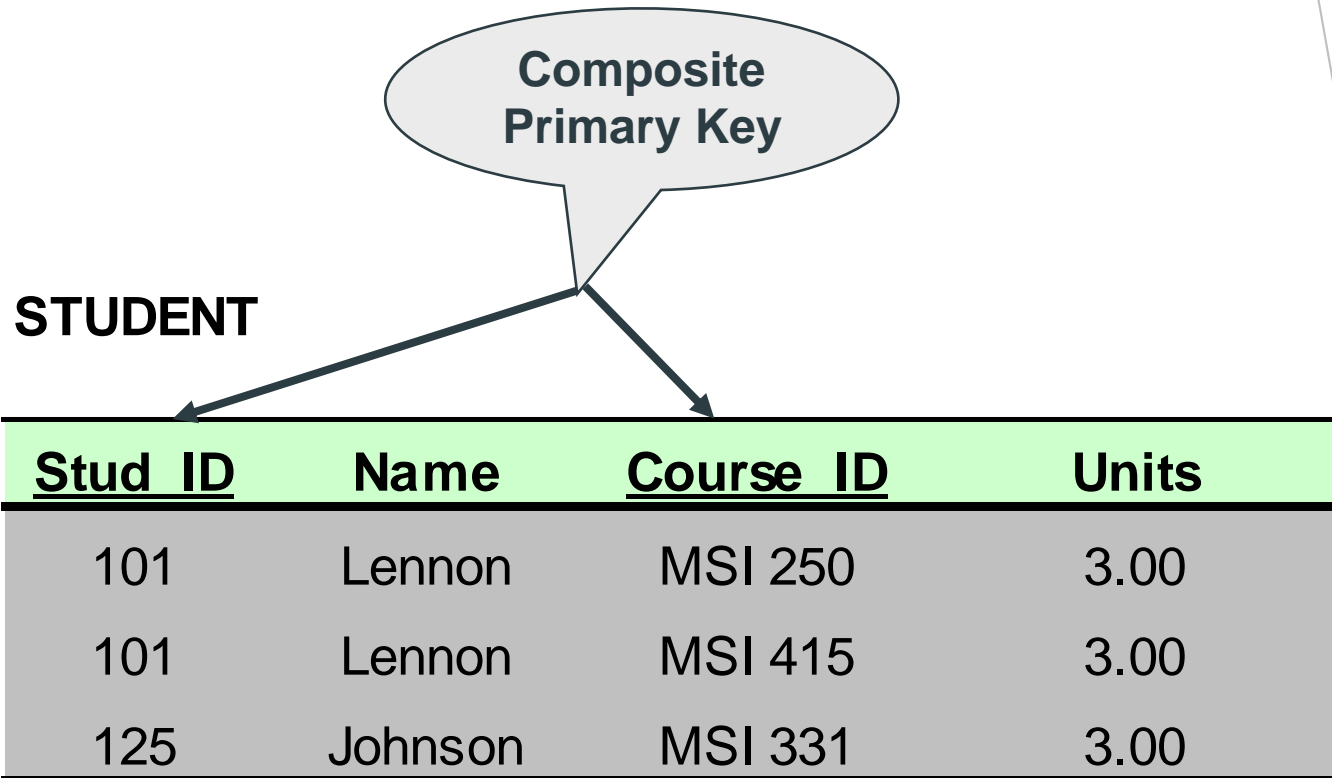
<u>Stud ID</u>	Name
101	Lennon
125	Jonson

STUDENT_COURSE



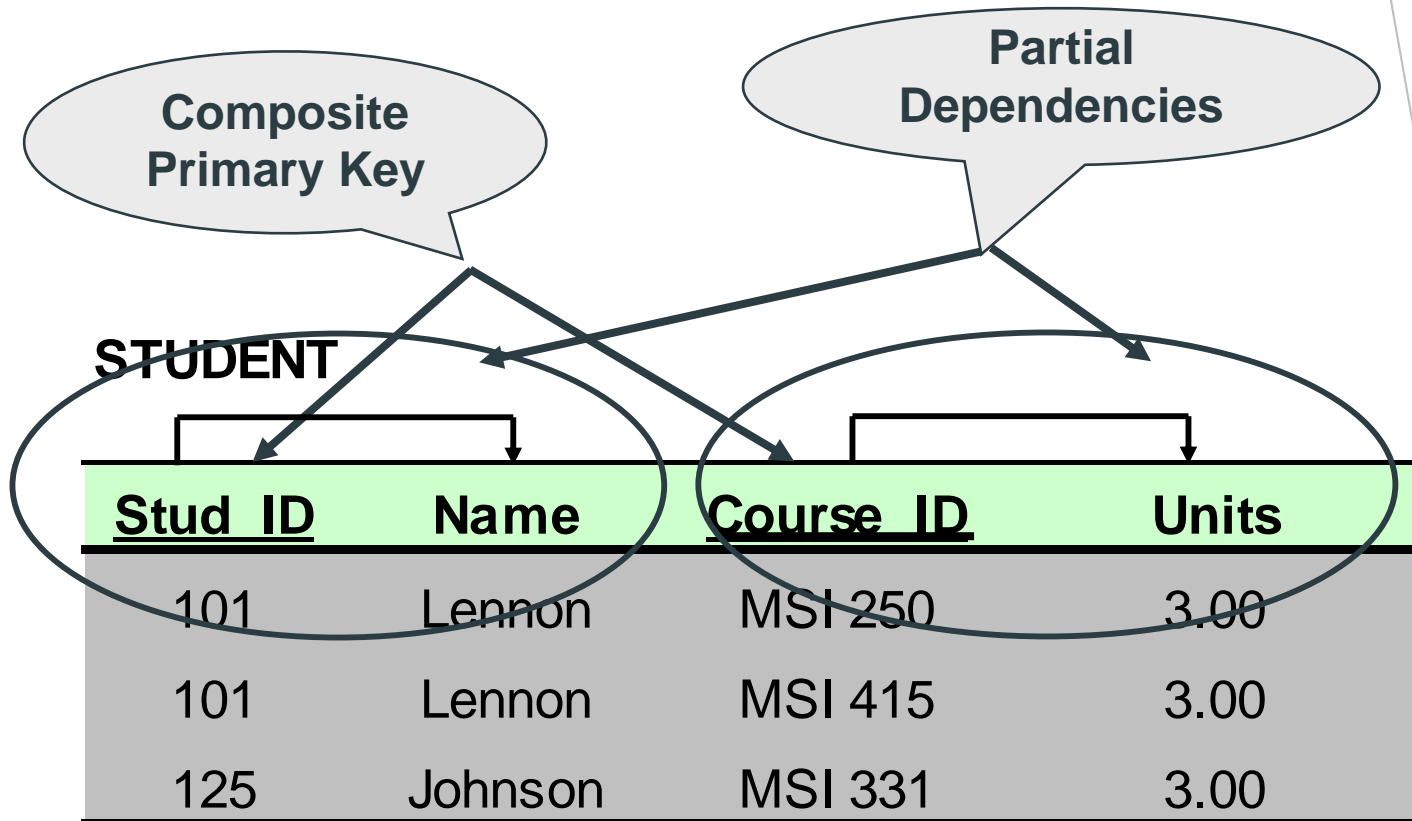
<u>Stud ID</u>	<u>Course</u>	Units
101	MSI 250	3
101	MSI 415	3
125	MSI 331	3

Bringing a Relation to 2NF



Bringing a Relation to 2NF

- Goal: Remove Partial Dependencies



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

<u>Stud ID</u>	Name	<u>Course ID</u>	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

<u>Stud ID</u>	Name	<u>Course ID</u>	Units
101	Lennon	MSI 250	3.00
101	Lennon	MSI 415	3.00
125	Johnson	MSI 331	3.00

Diagram illustrating the decomposition of the CUSTOMER relation. The relation is shown with columns Stud ID, Name, Course ID, and Units. Stud ID and Course ID are underlined, indicating they are primary keys. Arrows point from Stud ID to Name and from Course ID to Units, indicating functional dependencies.

STUDENT_COURSE

<u>Stud ID</u>	<u>Course ID</u>
101	MSI 250
101	MSI 415
125	MSI 331

STUDENT

<u>Stud ID</u>	Name
101	Lennon
101	Lennon
125	Johnson

Diagram illustrating the decomposition of the CUSTOMER relation into the STUDENT relation. The relation is shown with columns Stud ID and Name. Stud ID is underlined, indicating it is the primary key. An arrow points from Stud ID to Name, indicating a functional dependency.

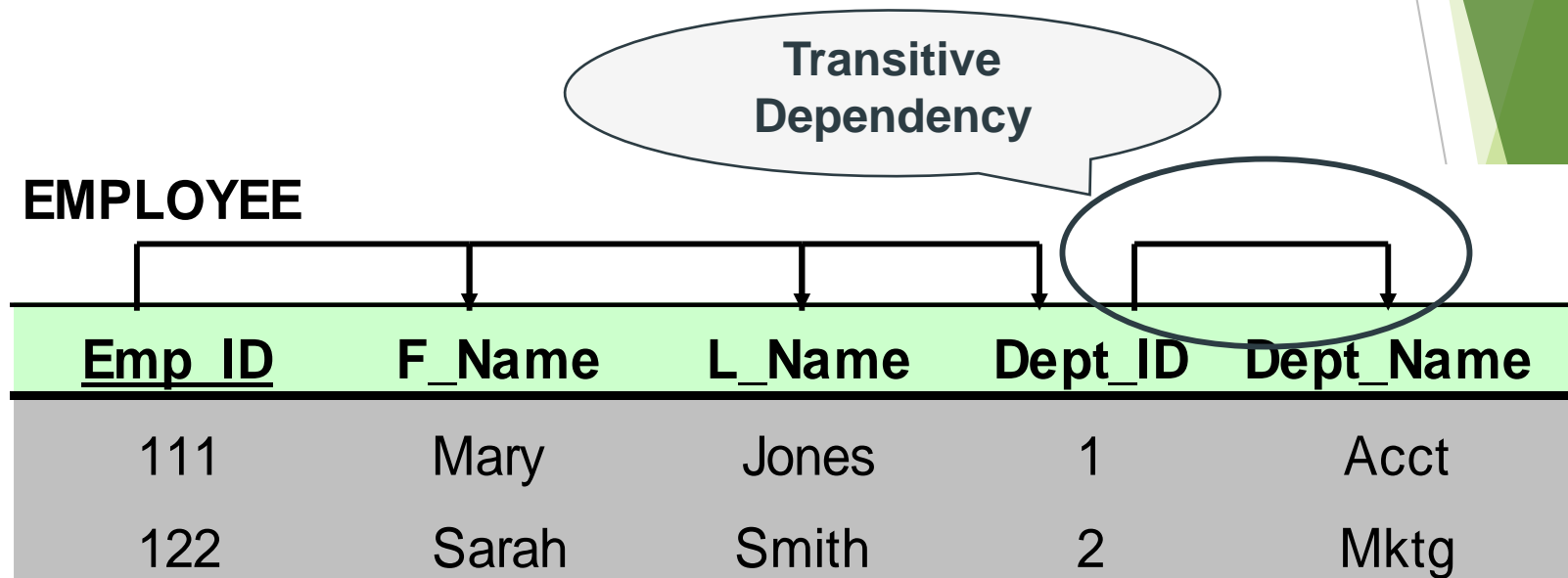
COURSE

<u>Course ID</u>	Units
MSI 250	3.00
MSI 415	3.00
MSI 331	3.00

Diagram illustrating the decomposition of the CUSTOMER relation into the COURSE relation. The relation is shown with columns Course ID and Units. Course ID is underlined, indicating it is the primary key. An arrow points from Course ID to Units, indicating a functional dependency.

Bringing a Relation to 3NF

- Goal: Get out of of transitive dependencies.



Bring Relation into 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

<u>Emp_ID</u>	F_Name	L_Name	Dept_ID	Dept_Name
111	Mary	Jones	1	Acct
122	Sarah	Smith	2	Mktg

Bring Relation into 3NF

EMPLOYEE

<u>Emp ID</u>	F_Name	L_Name	Dept_ID	Dept_Name
111	Mary	Jones	1	Acct
122	Sarah	Smith	2	Mktg

EMPLOYEE

<u>Emp ID</u>	F_Name	L_Name	Dept_ID
111	Mary	Jones	1
122	Sarah	Smith	2

DEPARTMENT

<u>Dept ID</u>	Dept_Name
1	Acct
2	Mktg