

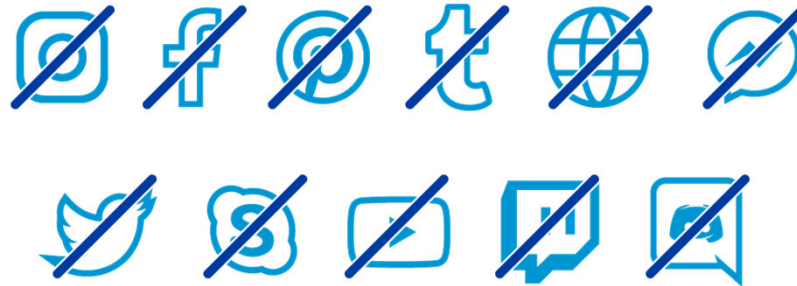
# DATABASES



Source: <https://en.itpedia.nl/2017/11/26/wat-is-een-database/>

Prof. Dr. Ulrike Herster  
Hamburg University of Applied Sciences

# COPYRIGHT



The publication and sharing of  
slides, images and sound recordings of this  
course is not permitted

© Professor Dr. Ulrike Herster

The slides and assignments are protected by copyright.

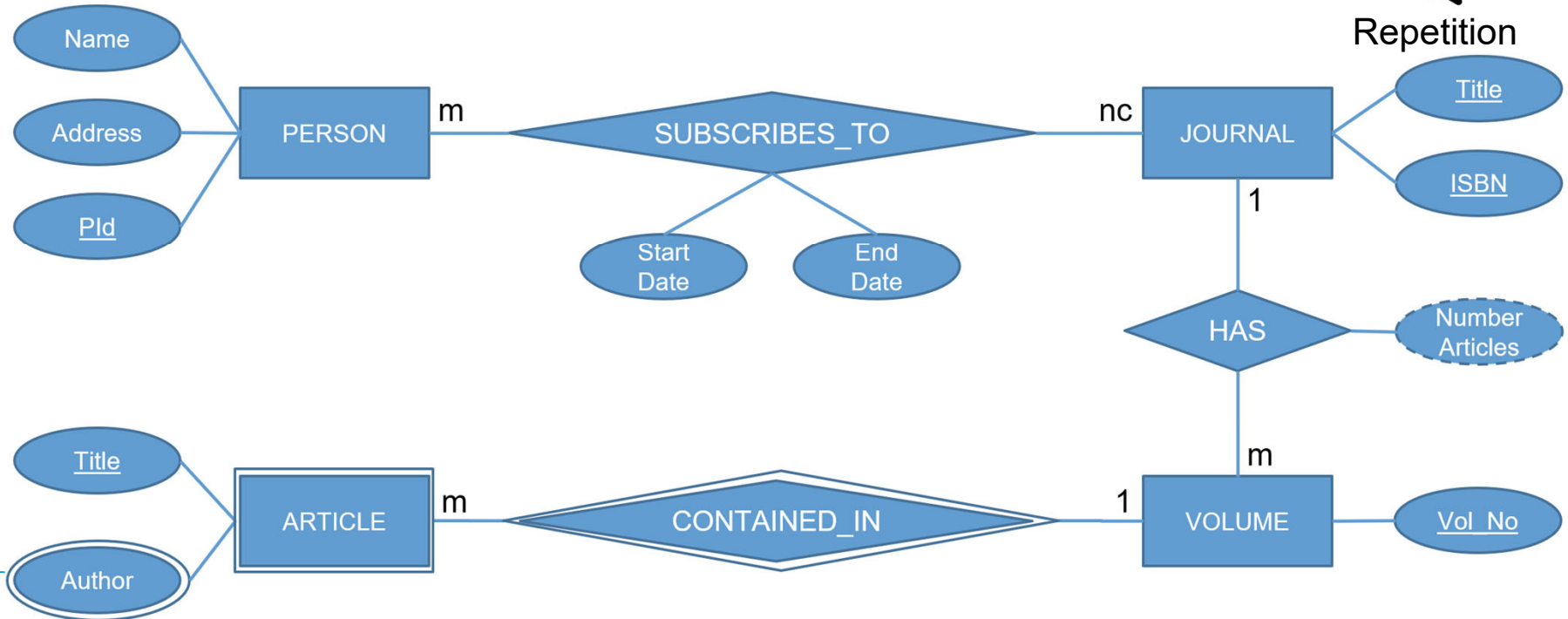
The use is only permitted in relation with the course of study.

It is not permitted to forward or republish it in other places (e.g., on the internet).

1

# RELATIONSHIPS

## ASSIGNMENT JOURNAL: CONVERT THE ERD TO A RM

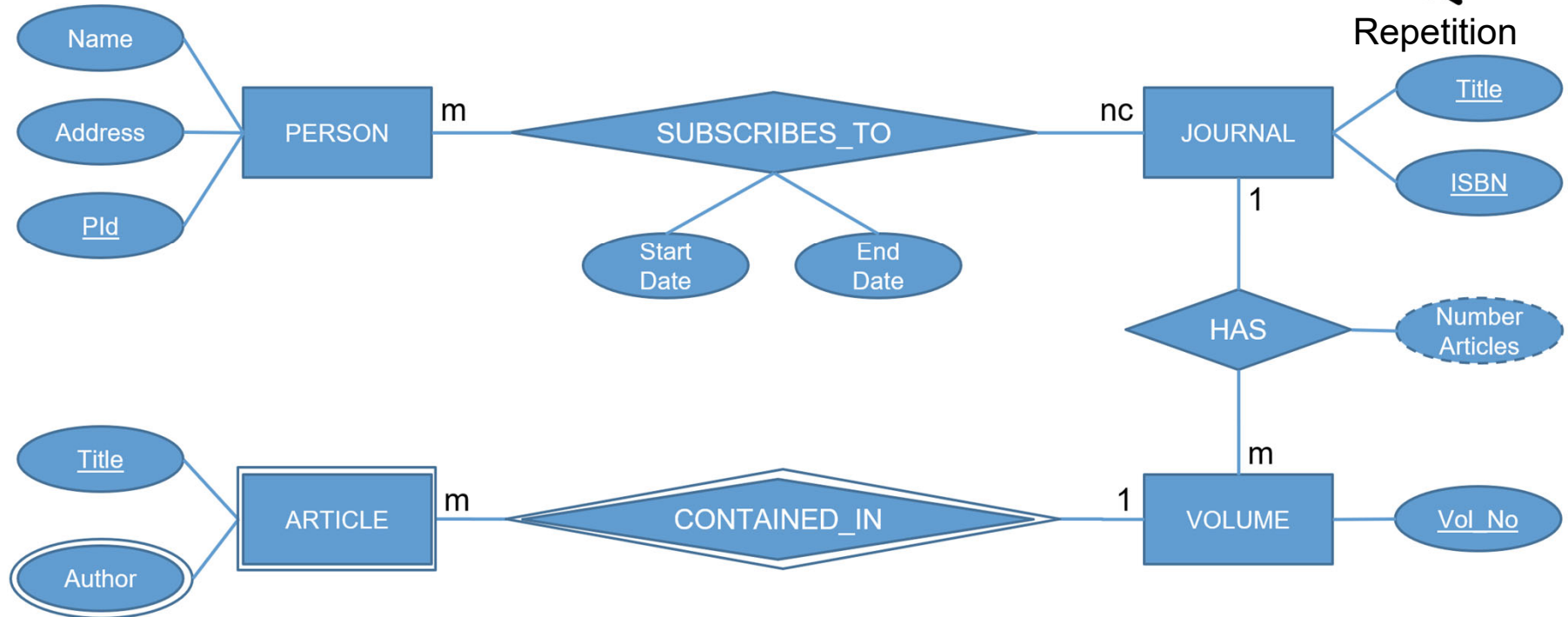


### Seven Steps

1. Mapping of regular entity types
2. Mapping of weak entity types
3. Mapping of binary 1:1 relationships
4. Mapping of binary 1:n relationships
5. Mapping of binary m:n relationships
6. Mapping of multivalued attributes
7. Mapping of n-ary relationships

# RELATIONSHIPS

## ASSIGNMENT JOURNAL: IMPLEMENT RM WITH SQL



```
[CONSTRAINT [symbol]] FOREIGN KEY
    [index_name] (col_name, ...)
    REFERENCES tbl_name (col_name,...)
    [ON DELETE reference_option]
    [ON UPDATE reference_option]
```

reference\_option:

RESTRICT | CASCADE | SET NULL | NO ACTION | SET DEFAULT

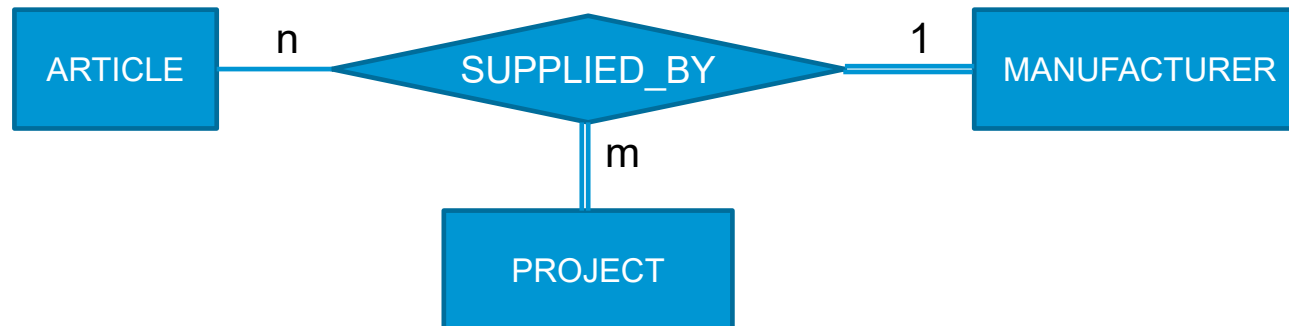
is of Database Systems – For personal use only

## RELATIONSHIPS

### ERM: TENARY RELATIONSHIP TYPES

Example:

- ❑ Manufacturers supply items for projects.
- ❑ A manufacturer must supply at least one item.
- ❑ An article from in-house production does not have to be supplied for a project but can be supplied for many projects.
- ❑ A project uses at least one item.
- ❑ An item is supplied by only one manufacturer for a project.

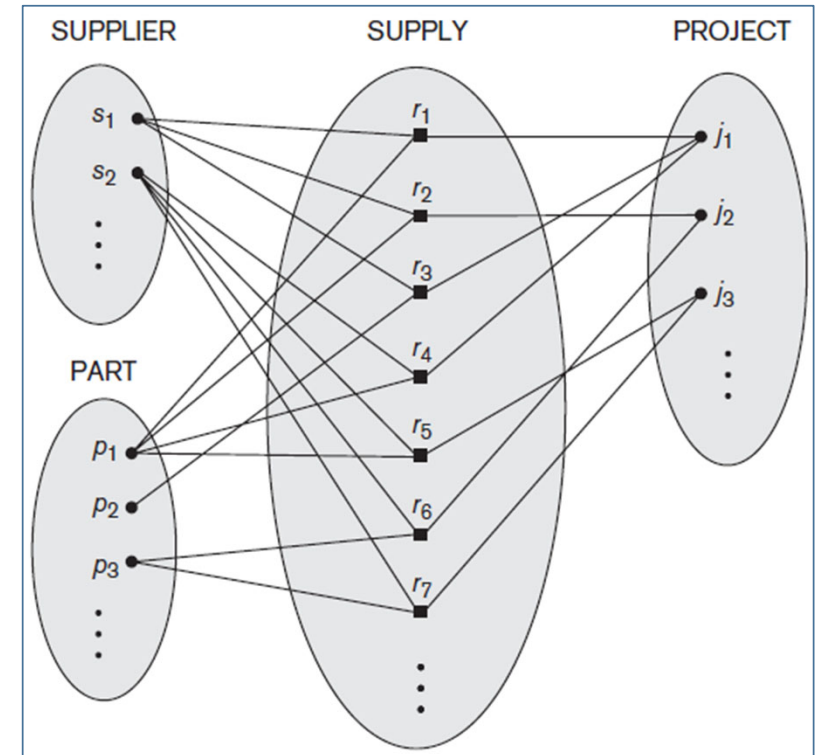
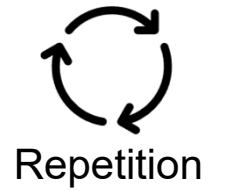
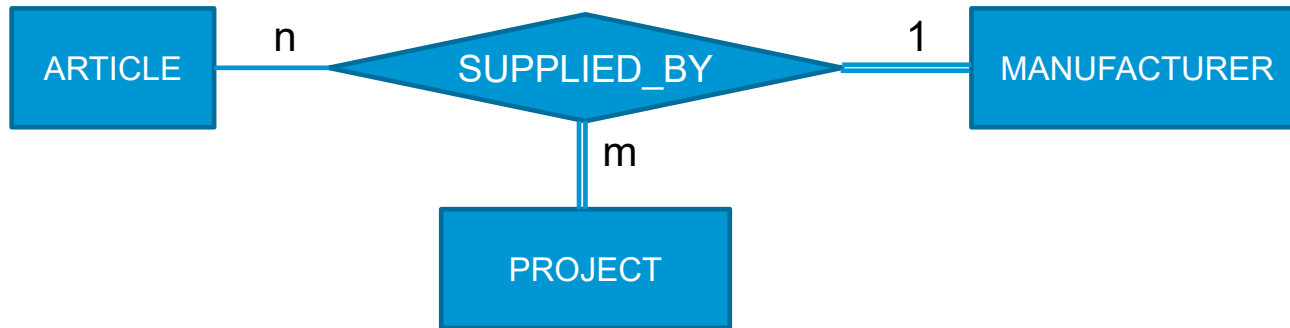


Question 1 (Cardinality): Can an entity of entity type A and an entity of entity type B be related to multiple entities of entity type C?

Question 2 (Participation): Must an entity type A be related to at least one entity type B and one entity type C?

# RELATIONSHIPS

## ERM: TENARY RELATIONSHIP TYPES

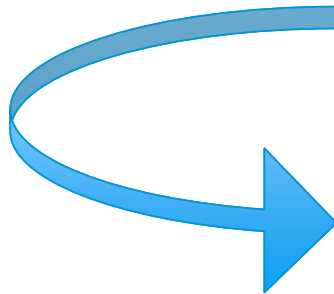
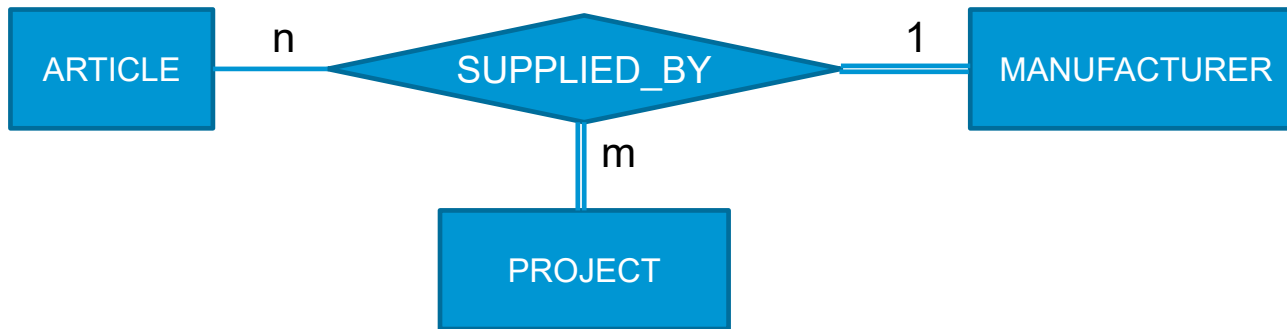
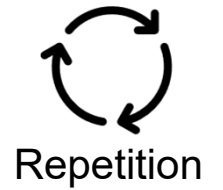


Source: Elmasri, Fundamentals of Database Systems, Page 214

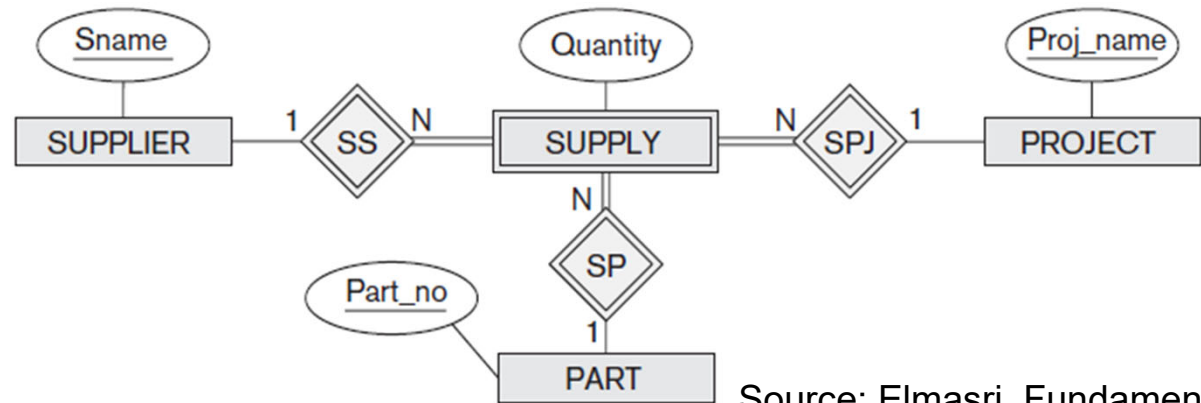
5

# RELATIONSHIPS

## ERM: TENARY RELATIONSHIP TYPES



If you do only have binary relationship types



Source: Elmasri, Fundamentals of Database Systems, Page 229

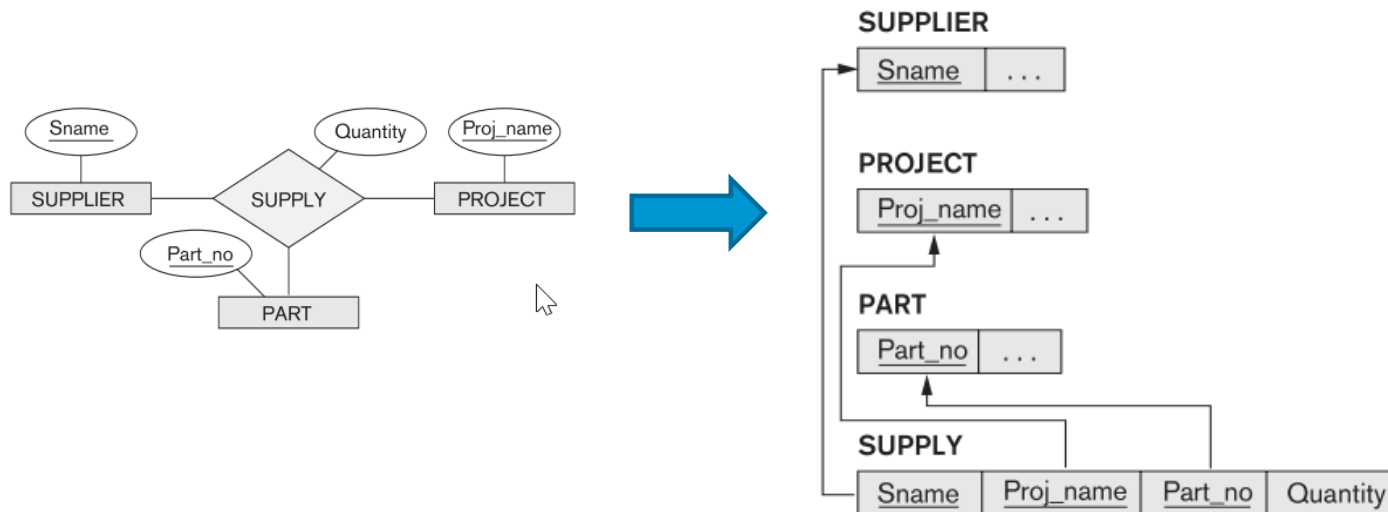
6

# RELATIONSHIPS

## RM: 7. MAPPING OF N-ARY RELATIONSHIP TYPES



Repetition



Source: Elmasri, Fundamentals of Database Systems, Page 286ff <sup>7</sup>



# RELATIONSHIPS

## MAPPING OF ERM TO RELATIONAL MODEL - SUMMARY



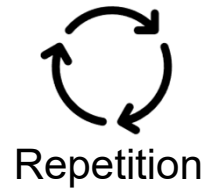
Repetition

ER Model	Relational Model
Entity type	Relation
1:1 or 1:N relationship type	Foreign key (or relationship relation)
M:N relationship type	Relationship relation and two foreign keys
N-ary relationship type	Relationship relation and n foreign keys
Simple attribute	Attribute
Composite attribute	Set of simple attributes
Multivalued attribute	Relation and foreign key
Value set	Domain
Key attribute	Primary key

Source: Elmasri, Fundamentals of Database Systems, Page 286ff

8

# CONSTRAINTS



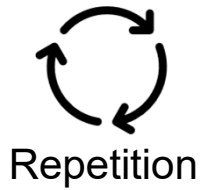
## Three categories

1. Constraints that are inherent in the data model  
→ *inherent model-based constraints* or *implicit constraints*  
Example: no duplicate tuples in a relation
2. Constraints that can be directly expressed in schemas of the data model  
→ *schema-based constraints* or *explicit constraints*  
Example: Domain constraints, key constraints, constraints on NULL, entity integrity constraints and referential integrity constraints
3. Constraints that cannot be directly expressed in the schemas of the data model, and hence must be expressed and enforced by the application programs  
→ *application-based* or *semantic constraints* or *business rules*

Source: Elmasri, Fundamentals of Database Systems, Page 59ff

9

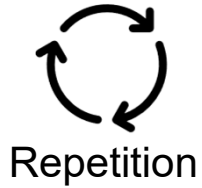
# CONSTRAINTS OVERVIEW



Constraint	Number of affected Relations
Domain constraints	1
Constraints on NULL	1
Entity integrity constraints (primary key)	1
Referential integrity constraints	$\geq 1$
Semantic integrity constraints	$\geq 1$
Functional dependencies constraint	$\geq 1$

# ORGANIZATION

## OUR JOURNEY IN THIS SEMESTER

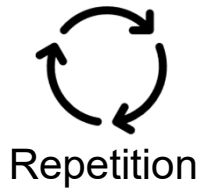


- Integrity, Trigger & Security
- Database Applications
- Transactions
- Subqueries & Views
- More SQL
- **Notations & Guidelines**
- Constraints
- Relationships
- Simple Entities and Attributes
- Basics

Source: Foto von Justin Kauffman auf Unsplash <sup>11</sup>

# NOTATIONS AND GUIDELINES

## ERM: MC NOTATION (MODIFIED CHEN NOTATION)



- Participation constraints
- Relationships can be mandatory or optional
- Types
  - ▣ Exactly one element: 1
  - ▣ One or no element: c (or 1c)
  - ▣ No or many elements: mc (or nc)
  - ▣ One or many elements: m (or n)

Also called *must-can* notation

# ORGANIZATION

## LABORATORY – 1. LAB ON 29.04.2024

- Joint lab with all three lab groups
  - **Attendance is mandatory!**
  - Division of the teams of two to work on the lab assignments
  - **Bring your own device with a working mySQL database!**
  - You do not have to submit solutions in moodle or implement the assignments in advance
- 
- Start time: 8:30 am
  - I strongly recommend that you look at and solve the assignments in advance! This will make the laboratory much more effective for you.

# ORGANIZATION

## OUR JOURNEY IN THIS SEMESTER



- Integrity, Trigger & Security
- Database Applications
- Transactions
- Subqueries & Views
- More SQL
- **Notations & Guidelines**
- Constraints
- Relationships
- Simple Entities and Attributes
- Basics

Source: Foto von Justin Kauffman auf Unsplash <sup>360</sup>



## NOTATIONS AND GUIDELINES

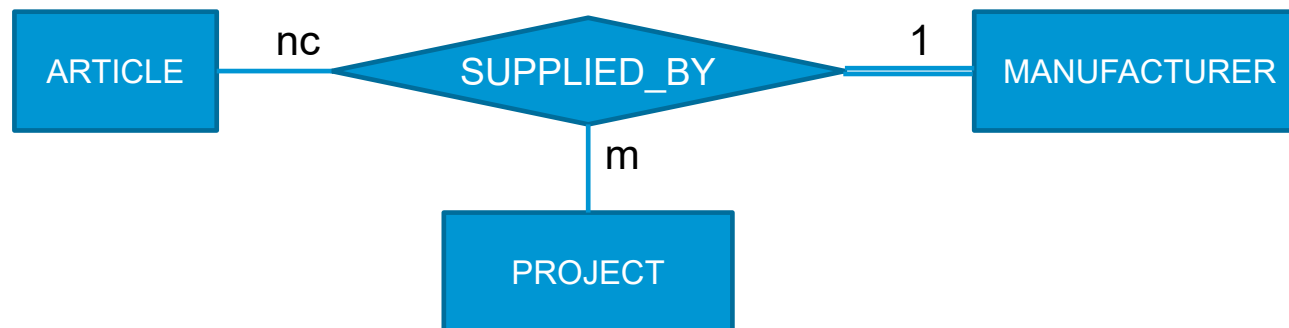
### ERM: MC NOTATION - TENARY RELATIONSHIP

Question 1 (Cardinality): Can an entity of entity type A and an entity of entity type B be related to multiple entities of entity type C?

Question 2 (Participation): Must an entity type A be related to at least one entity type B and one entity type C?

Example:

- ❑ Manufacturers supply items for projects.
- ❑ A manufacturer must supply at least one item.
- ❑ An article from in-house production does not have to be supplied for a project but can be supplied for many projects.
- ❑ A project uses at least one item.
- ❑ An item is supplied by only one manufacturer for a project.



375



## NOTATIONS AND GUIDELINES

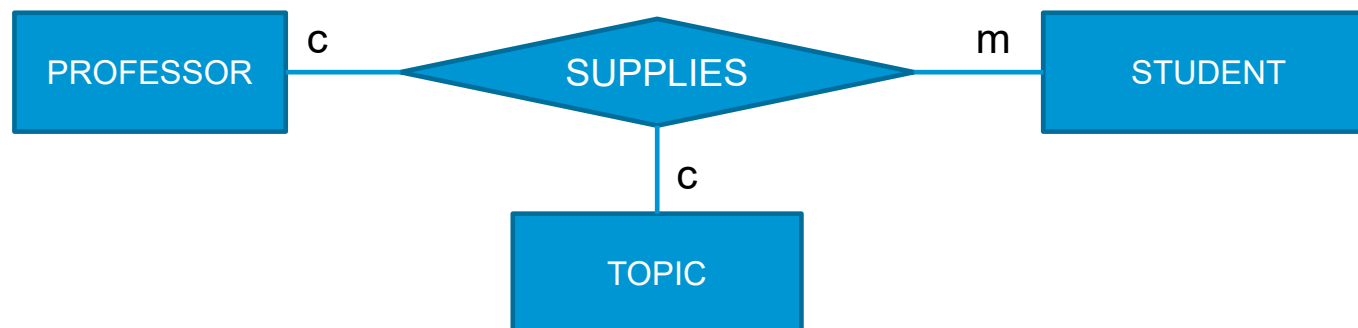
### ERM: MC NOTATION - TENARY RELATIONSHIPS

Question 1 (Cardinality): Can an entity of entity type A and an entity of entity type B be related to multiple entities of entity type C?

Question 2 (Participation): Must an entity type A be related to at least one entity type B and one entity type C?

Example:

- To prevent students from concentrating on one professor, they may only work with one professor on one seminar topic.
- In addition, a student can only work on a seminar topic with one professor.
- However, a professor may assign a seminar topic more than once.
- Students must attend seminars, but seminar topics do not have to be chosen.



376

## NOTATIONS AND GUIDELINES

### ERM: MC NOTATION - TERNARY RELATIONSHIP TYPES

Example:

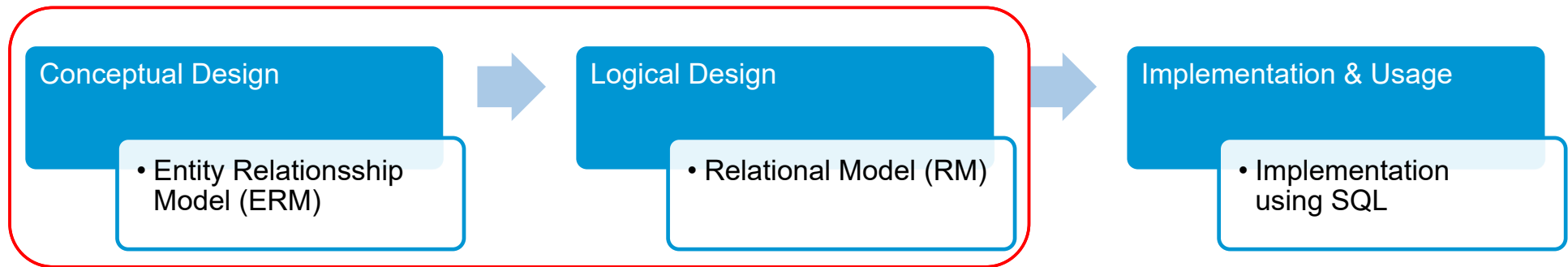


Source: <https://stackoverflow.com/questions/34506945/erm-cardinality-in-ternary-relationships>

377

# NOTATIONS AND GUIDELINES

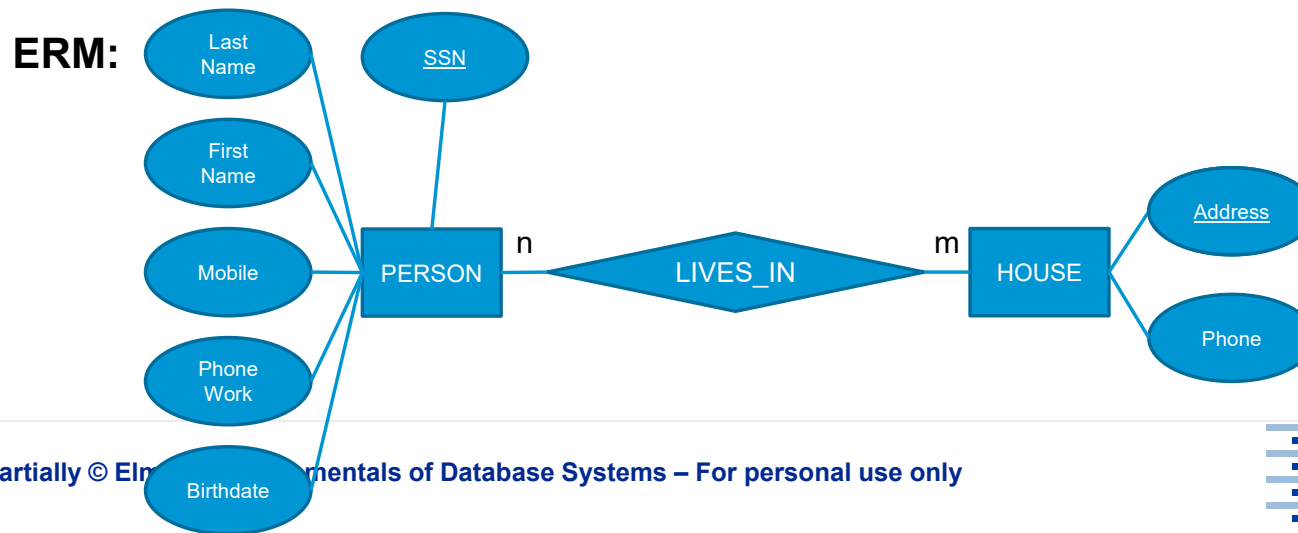
## DATABASE DESIGN



# NOTATIONS AND GUIDELINES

## COMPARISON ERM AND RELATIONAL MODEL

- ERM:
  - ▣ Conceptual Database Design
  - ▣ Describes a collection of **entities**, also called as real-world **objects** and **relations** between those entities
  - ▣ Basic elements: **entity type**, **relationship type** and **attributes**
  - ▣ **Constraints** like **Cardinality**, **Participation ratio** and **Keys**

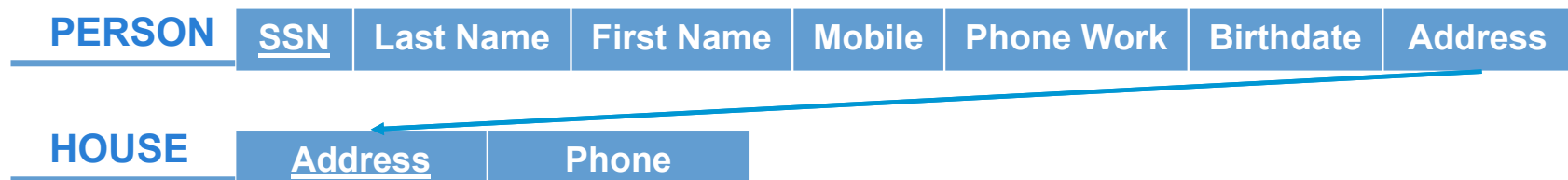


# NOTATIONS AND GUIDELINES

## COMPARISON ERM AND RELATIONAL MODEL

- Relational Model:
  - ▣ Logical Database Design
  - ▣ Describes data and relation among those data by tables
  - ▣ Basic elements: Relations and Attributes
  - ▣ Constraints: Domain constraints, key constraints, constraints on NULL, entity integrity constraints and referential integrity constraints

### Relational Model

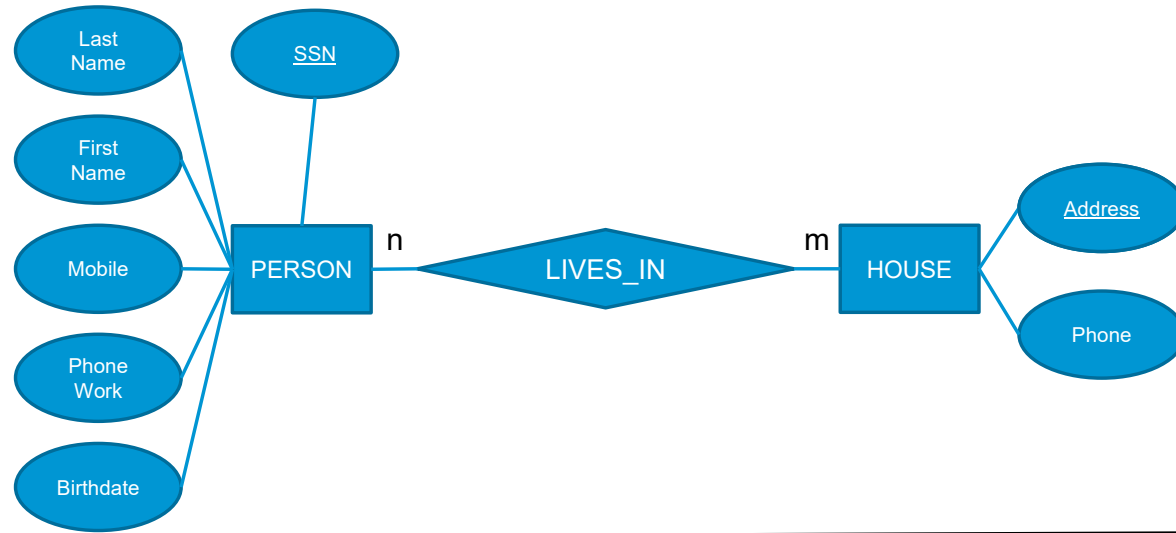


380

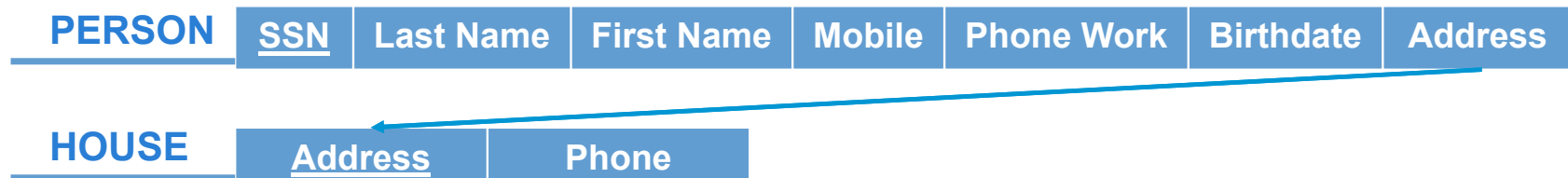
# NOTATIONS AND GUIDELINES

## COMPARISON ERM AND RELATIONAL MODEL

ERM



Relational Model



381

## NOTATIONS AND GUIDELINES

### COMPARISON ERM AND RELATIONAL MODEL

Aspect	ERM	RM
Basic	It represents the collection of objects called entities and relation between those entities	It represents the collection of tables and the relation between those tables
Describe	ERMs describe data as entity set, relationship set and attributes	Relational model describes data in a table as domains, attributes, tuples
Relationship	In an ERM, it is easier to understand the relationships between entities	Comparatively, it is less easy to derive a relation between tables in relational model
Mapping	ERM describes mapping cardinalities	Relational model does not describe mapping cardinalities

Source: <https://techdifferences.com/difference-between-e-r-and-relational-model-in-dbms.html>

# NOTATIONS AND GUIDELINES

## MAPPING OF ERM TO RELATIONAL MODEL

### Main rules

- Entity types
    - ▣ Mapped to relations
    - ▣ Relations contain the attributes
    - ▣ Composite attributes: set of simple attributes
  - Relationship Types
    - ▣ Foreign keys
- or
- ▣ Relations plus Foreign keys

Source: Elmasri, Fundamentals of  
Database Systems, Page 286ff <sup>383</sup>



# NOTATIONS AND GUIDELINES

## MAPPING OF ERM TO RELATIONAL MODEL

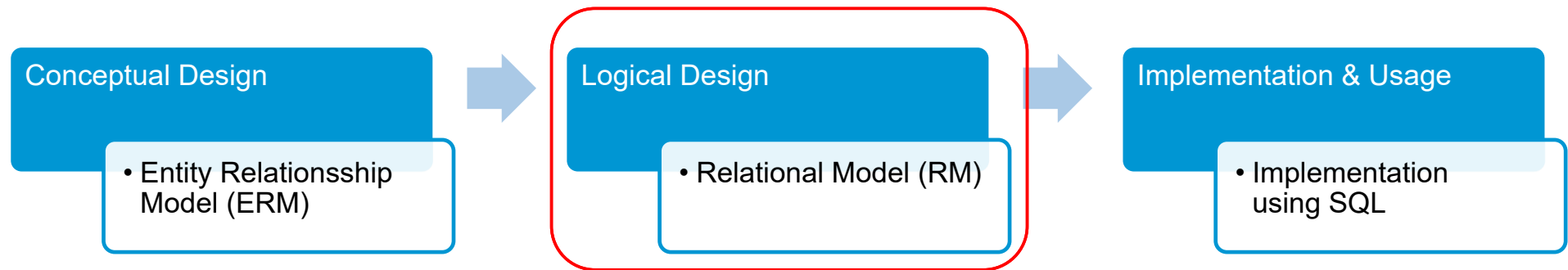
### Seven Steps

1. Mapping of regular entity types
2. Mapping of weak entity types
3. Mapping of binary 1:1 relationships
4. Mapping of binary 1:n relationships
5. Mapping of binary m:n relationships
6. Mapping of multivalued attributes
7. Mapping of n-ary relationships

Source: Elmasri, Fundamentals of  
Database Systems, Page 286ff <sup>384</sup>

# NOTATIONS AND GUIDELINES

## DATABASE DESIGN



# NOTATIONS AND GUIDELINES

## DESIGN GUIDELINES

### Guideline 1

- ❑ Clear semantics of relation attributes
- ❑ Meaning of attributes should be easy to explain
- ❑ Do not mix attributes from different entity types or relationship types into one relation

#### EMPLOYEE

Fname	Minit	Lname	<u>Ssn</u>	Bdate	Address	Sex	Salary	Super_ssn	Dno
-------	-------	-------	------------	-------	---------	-----	--------	-----------	-----

#### DEPARTMENT

Dname	<u>Dnumber</u>	Mgr_ssn	Mgr_start_date
-------	----------------	---------	----------------

#### DEPT\_LOCATIONS

<u>Dnumber</u>	<u>Dlocation</u>
----------------	------------------

Source: Elmasri, Fundamentals of Database Systems, Page 286ff <sup>386</sup>

# NOTATIONS AND GUIDELINES

## DESIGN GUIDELINES

### Guideline 2

- Avoid schemas that cause insertion, deletion, or modification anomalies
- If anomalies are present, note them clearly!
- Someone must take care of them (application, triggers)

#### EMPLOYEE

Fname	Minit	Lname	<u>Ssn</u>	Bdate	Address	Sex	Salary
-------	-------	-------	------------	-------	---------	-----	--------

#### DEPARTMENT

Dname	<u>Dnumber</u>
-------	----------------

#### PROJECT

Pname	<u>Pnumber</u>	Plocation
-------	----------------	-----------

#### DEPENDENT

<u>Essn</u>	<u>Dependent_name</u>	Sex	Bdate	Relationship
-------------	-----------------------	-----	-------	--------------



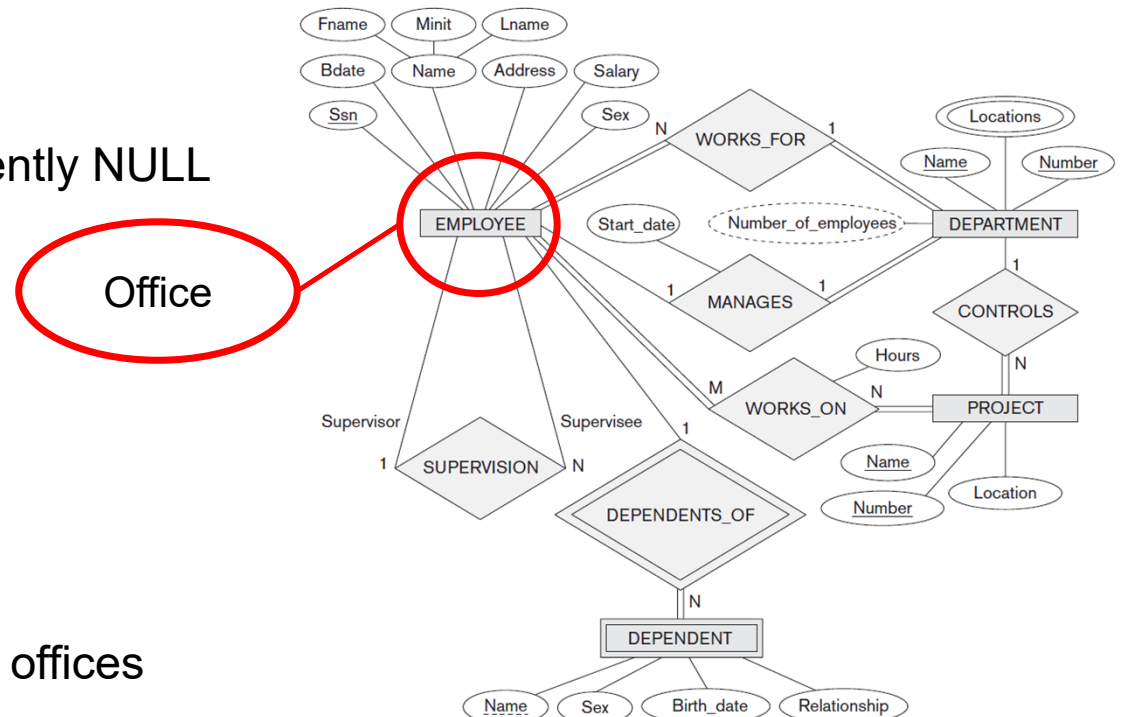
Source: Elmasri, Fundamentals of Database Systems, Page 286ff 387

# NOTATIONS AND GUIDELINES

## DESIGN GUIDELINES

### Guideline 3

- ❑ Avoid attributes whose values are frequently NULL
- ❑ NULL values should be the exception
- ❑ Attributes that are NULL frequently could be placed in separate relations (with the primary key): 1:c
- ❑ Example: Attribute *Office* if only 15 percent of employees have individual offices



Source: Elmasri, Fundamentals of Database Systems, Page 286ff 388

# NOTATIONS AND GUIDELINES

## DESIGN GUIDELINES

### Guideline 4

- Avoid relations that contain matching attributes that are not (foreign key, primary key) combinations
- Joining on such attributes may produce *spurious tuples*
- Matching attributes in relations should be (foreign key, primary key) combinations  
→ Do not forget the Foreign Keys!

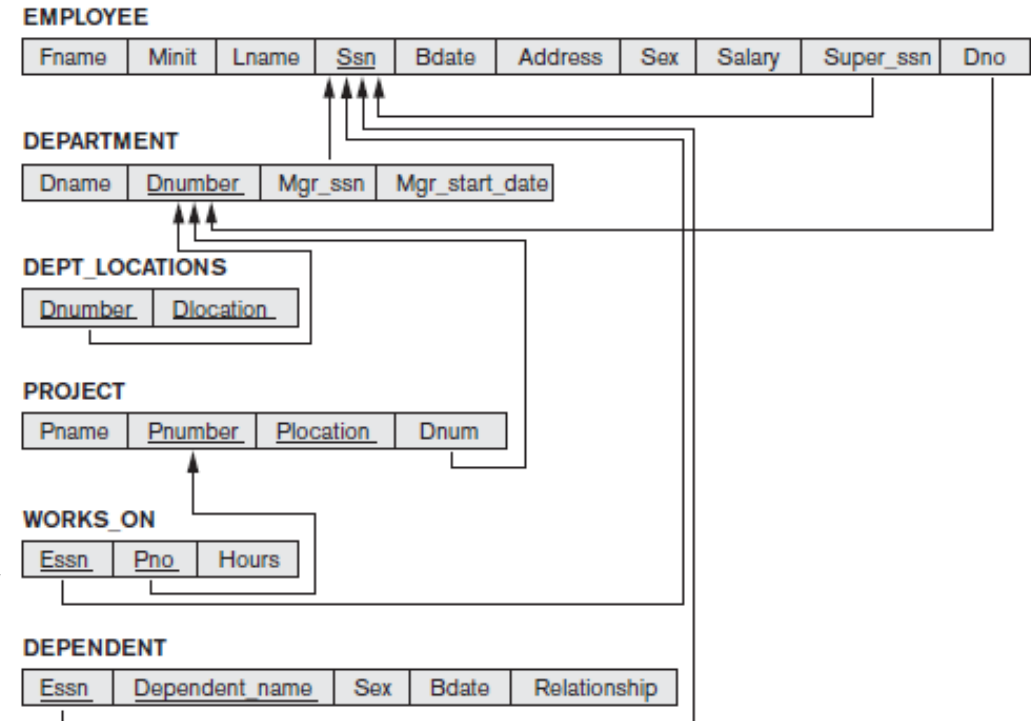
Source: Elmasri, Fundamentals of Database Systems, Page 286ff <sup>389</sup>

# NOTATIONS AND GUIDELINES

## DESIGN GUIDELINES

### Spurious Tuples

- Combining relations should produce facts only
  - Example:
    - Two relations
      - Employee
      - Project
    - When joining these relations, we want only tuples for real existing combinations of employees and projects
    - Other combinations: *Spurious tuples*
  - Spurious tuples are created when two tables are joined on attributes that are neither primary keys nor foreign keys



Source: Elmasri, Fundamentals of Database Systems, Page 286ff 390

## NOTATIONS AND GUIDELINES

### DESIGN GUIDELINES

Why don't we put everything in one big table?

ALL	Professor	Lecture	Student	LiteracyTitle	ISBN	Semester	...
	Smith	C++	John	Learn C++	12345	4	...
	Smith	C++	Julie	Learn C++	12345	4	...
	Collins	DBMS	Julie	Databases	23456	4	...
	Collins	DBMS	Anna	Databases	23456	4	...
	...	...	...	...	...	...	...



## NOTATIONS AND GUIDELINES

### DESIGN GUIDELINES

One problem is: Some data is the always the same (E.g., ISBN)

- This problem occurs in nearly all applications
- There are almost one or several attributes, which define the values of the remaining attributes
- This fact is called *functional dependency*
- Functional dependency is relevant in practical applications

ALL	Professor	Lecture	Student	LiteracyTitle	ISBN	Semester	...
	Smith	C++	John	Learn C++	12345	4	...
	Smith	C++	Julie	Learn C++	12345	4	...
	Collins	DBMS	Julie	Databases	23456	4	...
	Collins	DBMS	Anna	Databases	23456	4	...
	...	...	...	...	...	...	...

392

# NOTATIONS AND GUIDELINES

## DESIGN GUIDELINES

### Functional dependencies

- Definition: There is a relational schema called R, and X and Y are any attribute sets of the attributes of R. Then Y is functional dependent on X, noted as  $X \rightarrow Y$ , if and only if every value of X in R defines exactly one value of Y in R
- In other words: Whenever two tuples of relation R match in their X values, then the Y values of these tuples match as well

ALL	Professor	Lecture	Student	LiteracyTitle	ISBN	Semester	...
	Smith	C++	John	Learn C++	12345	4	...
	Smith	C++	Julie	Learn C++	12345	4	...
	Collins	DBMS	Julie	Databases	23456	4	...
	Collins	DBMS	Anna	Databases	23456	4	...
	...	...	...	...	...	...	...

Diagram illustrating functional dependencies: A red bracket labeled 'Y' groups the 'LiteracyTitle' and 'ISBN' columns. A red bracket labeled 'X' groups the 'ISBN' and 'Semester' columns. This indicates that 'LiteracyTitle' is functionally dependent on 'ISBN' ( $ISBN \rightarrow LiteracyTitle$ ).

# NOTATIONS AND GUIDELINES

## DESIGN GUIDELINES

### Functional dependencies

- Are used to specify formal measures of the "goodness" of relational designs
- Can help to identify redundancy and suggest refinements
- FD: If for 2 tuples X is the same, then Y must also be the same
- Notation:  $X \rightarrow Y$   
Read: "X determines Y"
- Generalization of keys
- A key determines all attributes:  $K = A_i$

# NOTATIONS AND GUIDELINES

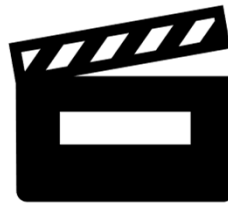
## DESIGN GUIDELINES

### Functional dependencies

- Actual data can help to identify FDs
- Note: FD is a statement about *all allowable* tuples!
- Based on semantics, NOT instances!
- Full functional dependency:
  - ▣  $X \rightarrow Y$  holds only for complete X, not for subset of X
  - ▣ You cannot remove an element of X without destroying the FD
  - ▣ Opposite: Partial Functional Dependency

# NOTATIONS AND GUIDELINES

## NORMALIZATION



Source: <https://www.youtube.com/watch?v=Nq91b4TQ29Q>

# NOTATIONS AND GUIDELINES

## NORMALIZATION

- Normal Forms provide quality statements on relations
- The process of decomposing unsatisfactory "bad" relations by breaking up their attributes into smaller relations
- There are different Normal Forms:  
1NF, 2NF, 3NF, BCNF, ...

# NOTATIONS AND GUIDELINES

## NORMALIZATION

### □ Example:

Title	Author	Author Nationality	Format	Price	Subject	Pages	Thickness	Publisher	Publisher Country	Publication Type	Genre ID	Genre Name
Beginning MySQL Database Design and Optimization	Chad Russell	American	Hardcover	49.99	MySQL, Database, Design	520	Thick	Apress	USA	E-book	1	Tutorial

Source: [https://en.wikipedia.org/wiki/Database\\_normalization](https://en.wikipedia.org/wiki/Database_normalization) <sup>398</sup>

# NOTATIONS AND GUIDELINES

## NORMALIZATION – 1NF

- The *First Normal Form* (1NF) defines that the values in each column of a table must be atomic
- Solution: Separate the duplicities into multiple columns using repeating groups 'subject'

Title	Author	Author Nationality	Format	Price	Subject	Pages	Thickness	Publisher	Publisher Country	Publication Type	Genre ID	Genre Name
Beginning MySQL Database Design and Optimization	Chad Russell	American	Hardcover	49.99	MySQL, Database, Design	520	Thick	Apress	USA	E-book	1	Tutorial



<u>Title</u>	<u>Format</u>	Author	Author Nationality	Price	Subject 1	Subject 2	Subject 3	Pages	Thickness	Publisher	Publisher country	Genre ID	Genre Name
Beginning MySQL Database Design and Optimization	Hardcover	Chad Russell	American	49.99	MySQL	Database	Design	520	Thick	Apress	USA	1	Tutorial

- Disadvantages?

Source: [https://en.wikipedia.org/wiki/Database\\_normalization](https://en.wikipedia.org/wiki/Database_normalization) 399



# NOTATIONS AND GUIDELINES

## NORMALIZATION – 1NF

### □ Alternative Solution:

Title	Author	Author Nationality	Format	Price	Subject	Pages	Thickness	Publisher	Publisher Country	Publication Type	Genre ID	Genre Name
Beginning MySQL Database Design and Optimization	Chad Russell	American	Hardcover	49.99	MySQL, Database, Design	520	Thick	Apress	USA	E-book	1	Tutorial



Book

<u>Title</u>	<u>Format</u>	Author	Author Nationality	Price	Pages	Thickness	Genre ID	Genre Name	<i>Publisher ID</i>
Beginning MySQL Database Design and Optimization	Hardcover	Chad Russell	American	49.99	520	Thick	1	Tutorial	1

Subject	
<u>Subject ID</u>	Subject name
1	MySQL
2	Database
3	Design

Title - Subject	
<u>Title</u>	<i>Subject ID</i>
Beginning MySQL Database Design and Optimization	1
Beginning MySQL Database Design and Optimization	2
Beginning MySQL Database Design and Optimization	3

Source: [https://en.wikipedia.org/wiki/Database\\_normalization](https://en.wikipedia.org/wiki/Database_normalization) 400

# NOTATIONS AND GUIDELINES

## NORMALIZATION – 1NF

Decomposition to 1NF:

- Composite Attributes
    - Split into single atomic attributes
  - Multi-valued Attributes
    - Decompose to new tuples
      - Results in redundancies
      - Further decomposition in following steps
- or
- Decompose to new relation with FK

## NOTATIONS AND GUIDELINES

### NORMALIZATION – 2NF

- A relation is in Second Normal Form (2NF), if
  - ▣ It is in 1NF and
  - ▣ Every attribute in R is fully functional dependent on every key in R (or is part of the key)
- In other words: If there is a key with different attributes (e.g., title and format) and an attribute depends on just a part of this key (e.g., title), then the second Normal Form is violated
- To create the Second Normal Form, you must decompose the relation:  
Depending attribute with part of key in new relation, delete depending attribute in old relation

# NOTATIONS AND GUIDELINES

## NORMALIZATION – 2NF

Book

<u>Title</u>	<u>Format</u>	Author	Author Nationality	Price	Pages	Thickness	Genre ID	Genre Name	Publisher ID
Beginning MySQL Database Design and Optimization	Hardcover	Chad Russell	American	49.99	520	Thick	1	Tutorial	1
Beginning MySQL Database Design and Optimization	E-book	Chad Russell	American	22.34	520	Thick	1	Tutorial	1
The Relational Model for Database Management: Version 2	E-book	E.F.Codd	British	13.88	538	Thick	2	Popular science	2
The Relational Model for Database Management: Version 2	Paperback	E.F.Codd	British	39.99	538	Thick	2	Popular science	2

□ Primary key: Title + Format

□ Functional dependencies:

Title → Author

Title → Author Nationality

Title, Format → Price

Title → Pages

Title → Thickness

Title → GenreID

Title → Genre Name

Title → Publisher ID

Source: [https://en.wikipedia.org/wiki/Database\\_normalization](https://en.wikipedia.org/wiki/Database_normalization)

403

# NOTATIONS AND GUIDELINES

## NORMALIZATION – 2NF

**Book**

<u>Title</u>	<u>Format</u>	Author	Author Nationality	Price	Pages	Thickness	Genre ID	Genre Name	Publisher ID
Beginning MySQL Database Design and Optimization	Hardcover	Chad Russell	American	49.99	520	Thick	1	Tutorial	1
Beginning MySQL Database Design and Optimization	E-book	Chad Russell	American	22.34	520	Thick	1	Tutorial	1
The Relational Model for Database Management: Version 2	E-book	E.F.Codd	British	13.88	538	Thick	2	Popular science	2
The Relational Model for Database Management: Version 2	Paperback	E.F.Codd	British	39.99	538	Thick	2	Popular science	2



**Book**

<u>Title</u>	Author	Author Nationality	Pages	Thickness	Genre ID	Genre Name	Publisher ID
Beginning MySQL Database Design and Optimization	Chad Russell	American	520	Thick	1	Tutorial	1
The Relational Model for Database Management: Version 2	E.F.Codd	British	538	Thick	2	Popular science	2

**Format - Prices**

<u>Title</u>	<u>Format</u>	Price
Beginning MySQL Database Design and Optimization	Hardcover	49.99
Beginning MySQL Database Design and Optimization	E-book	22.34
The Relational Model for Database Management: Version 2	E-book	13.88
The Relational Model for Database Management: Version 2	Paperback	39.99

Source: [https://en.wikipedia.org/wiki/Database\\_normalization](https://en.wikipedia.org/wiki/Database_normalization) 404

# NOTATIONS AND GUIDELINES

## NORMALIZATION – 2NF

### Decomposition to Second Normal Form

- Has the relation in 1NF following design  
(keyPart1, keyPart2, noKey1, noKey2)  
and there is a functional dependency  
FD:  $\text{keyPart2} \rightarrow \text{noKey2}$
- Then the decomposition respecting this FD results in following schema:  
(keyPart1, keyPart2, noKey1)  
(keyPart2, noKey2)

# NOTATIONS AND GUIDELINES

## NORMALIZATION – 2NF

### Decomposition to Second Normal Form

- Has the relation in 1NF following design (keyPart1, keyPart2, noKey1, noKey2) and there is a functional dependency  
FD: keyPart2 → noKey2

Book									
<u>Title</u>	<u>Format</u>	Author	Author Nationality	Price	Pages	Thickness	Genre ID	Genre Name	<i>Publisher ID</i>
Beginning MySQL Database Design and Optimization	Hardcover	Chad Russell	American	49.99	520	Thick	1	Tutorial	1
Beginning MySQL Database Design and Optimization	E-book	Chad Russell	American	22.34	520	Thick	1	Tutorial	1
The Relational Model for Database Management: Version 2	E-book	E.F.Codd	British	13.88	538	Thick	2	Popular science	2
The Relational Model for Database Management: Version 2	Paperback	E.F.Codd	British	39.99	538	Thick	2	Popular science	2

Source: [https://en.wikipedia.org/wiki/Database\\_normalization](https://en.wikipedia.org/wiki/Database_normalization)

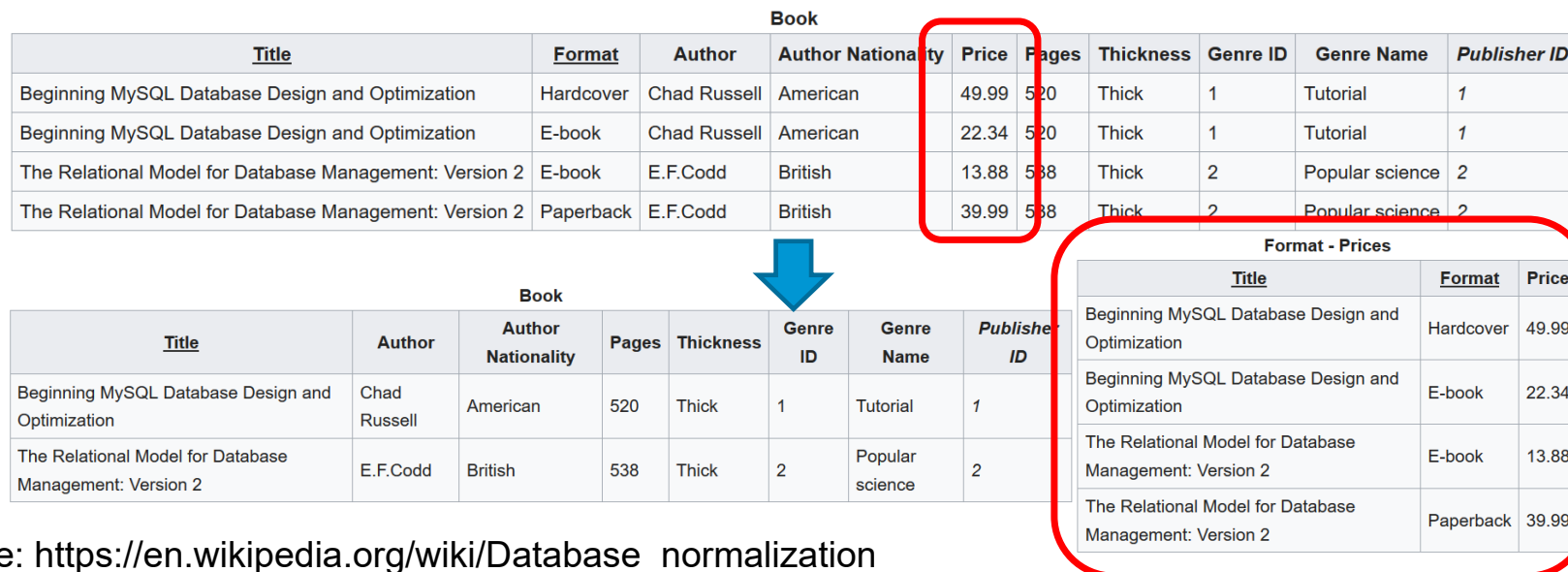
406

# NOTATIONS AND GUIDELINES

## NORMALIZATION – 2NF

### Decomposition to Second Normal Form

- Then the decomposition respecting this FD results in following schema:  
 (keyPart1, keyPart2, noKey1)  
 (keyPart2, noKey2)



Source: [https://en.wikipedia.org/wiki/Database\\_normalization](https://en.wikipedia.org/wiki/Database_normalization)

407



# NOTATIONS AND GUIDELINES

## NORMALIZATION – 3NF

The Third Normal Form (3NF) describes the problem

- Informal: A relation is in 3NF if every tuple consists of a primary key and a set of other attributes that are independent of each other
- Formal: A relation is in 3NF if
  - ▣ It is in 2NF and
  - ▣ Every non-primary-key attribute is directly dependent on the primary key (especially no transitive dependencies)
- In other words: A table in third normal form (3NF) is a table in 2NF that has no transitive dependencies

# NOTATIONS AND GUIDELINES

## NORMALIZATION – 3NF

### Transitive Dependency

- Z is transitive dependent on X, if
  - ▣  $X \rightarrow Y \rightarrow Z$
  - ▣ But not  $Y \rightarrow X$
- Second condition important: not a transitive dependency if X and Y are both keys!

# NOTATIONS AND GUIDELINES

## NORMALIZATION – 3NF

Book							
<u>Title</u>	Author	Author Nationality	Pages	Thickness	Genre ID	Genre Name	Publisher ID
Beginning MySQL Database Design and Optimization	Chad Russell	American	520	Thick	1	Tutorial	1
The Relational Model for Database Management: Version 2	E.F.Codd	British	538	Thick	2	Popular science	2
Learning SQL	Alan Beaulieu	American	338	Slim	1	Tutorial	3
SQL Cookbook	Anthony Molinaro	American	636	Thick	1	Tutorial	3

### □ Functional dependencies:

Title → Author

Title → Author Nationality

Title → Pages

Title → Thickness

Title → GenreID

Title → Genre Name

Title → Publisher ID

Author → Author Nationality

GenreID → Genre Name

Source: [https://en.wikipedia.org/wiki/Database\\_normalization](https://en.wikipedia.org/wiki/Database_normalization)

410

# NOTATIONS AND GUIDELINES

## NORMALIZATION – 3NF

Book							
<u>Title</u>	Author	Author Nationality	Pages	Thickness	Genre ID	Genre Name	Publisher ID
Beginning MySQL Database Design and Optimization	Chad Russell	American	520	Thick	1	Tutorial	1
The Relational Model for Database Management: Version 2	E.F.Codd	British	538	Thick	2	Popular science	2
Learning SQL	Alan Beaulieu	American	338	Slim	1	Tutorial	3
SQL Cookbook	Anthony Molinaro	American	636	Thick	1	Tutorial	3



Book						
<u>Title</u>	Author	Author Nationality	Pages	Thickness	Genre ID	Publisher ID
Beginning MySQL Database Design and Optimization	Chad Russell	American	520	Thick	1	1
The Relational Model for Database Management: Version 2	E.F.Codd	British	538	Thick	2	2
Learning SQL	Alan Beaulieu	American	338	Slim	1	3
SQL Cookbook	Anthony Molinaro	American	636	Thick	1	3

Book Genres	
<u>Genre ID</u>	Genre Name
1	Tutorial
2	Popular science

Source: [https://en.wikipedia.org/wiki/Database\\_normalization](https://en.wikipedia.org/wiki/Database_normalization)

411

# NOTATIONS AND GUIDELINES

## NORMALIZATION – 3NF

### Decomposition to Third Normal Form

- The relation in 2NF has the schema (KeyPart1, NonKey1, NonKey2) and there is a functional dependency  
FD: NonKey1  $\rightarrow$  NonKey2
- Then the decomposition results in a new schema, which is in Third Normal Form  
(KeyPart1, NonKey1)  
(NonKey1, NonKey2)

# NOTATIONS AND GUIDELINES

## NORMALIZATION – 3NF

### Decomposition to Third Normal Form

- The relation in 2NF has the schema (KeyPart1, NonKey1, NonKey2) and there is a functional dependency  
FD: NonKey1  $\rightarrow$  NonKey2

Book							
<u>Title</u>	Author	Author Nationality	Pages	Thickness	Genre ID	Genre Name	Publisher ID
Beginning MySQL Database Design and Optimization	Chad Russell	American	520	Thick	1	Tutorial	1
The Relational Model for Database Management: Version 2	E.F.Codd	British	538	Thick	2	Popular science	2
Learning SQL	Alan Beaulieu	American	338	Slim	1	Tutorial	3
SQL Cookbook	Anthony Molinaro	American	636	Thick	1	Tutorial	3

Source: [https://en.wikipedia.org/wiki/Database\\_normalization](https://en.wikipedia.org/wiki/Database_normalization)

413

# NOTATIONS AND GUIDELINES

## NORMALIZATION – 3NF

### Decomposition to Third Normal Form

- Then the decomposition results in a new schema, which is in Third Normal Form  
(KeyPart1, NonKey1)  
(NonKey1, NonKey2)

**Book**

<u>Title</u>	Author	Author Nationality	Pages	Thickness	Genre ID	Genre Name	Publisher ID
Beginning MySQL Database Design and Optimization	Chad Russell	American	520	Thick	1	Tutorial	1
The Relational Model for Database Management: Version 2	E.F.Codd	British	538	Thick	2	Popular science	2
Learning SQL	Alan Beaulieu	American	338	Slim	1	Tutorial	3
SQL Cookbook	Anthony Molinaro	American	636	Thick	1	Tutorial	3



**Book**

<u>Title</u>	Author	Author Nationality	Pages	Thickness	Genre ID	Publisher ID
Beginning MySQL Database Design and Optimization	Chad Russell	American	520	Thick	1	1
The Relational Model for Database Management: Version 2	E.F.Codd	British	538	Thick	2	2
Learning SQL	Alan Beaulieu	American	338	Slim	1	3
SQL Cookbook	Anthony Molinaro	American	636	Thick	1	3

<u>Genre ID</u>	Genre Name
1	Tutorial
2	Popular science

Author	Author Nationality
Chad Russell	American
E.F.Codd	British

# NOTATIONS AND GUIDELINES

## NORMALIZATION

**For normalization,  
use functional dependencies!!!**

**And not the semantics  
of the attributes!!!**



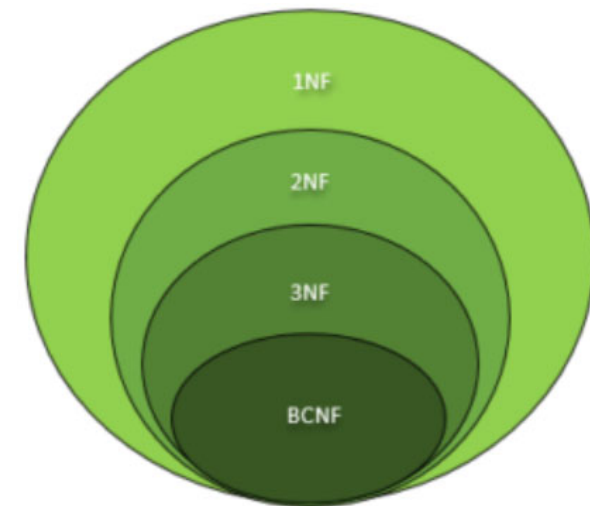


# NOTATIONS AND GUIDELINES

## NORMALIZATION – BCNF



- A relational schema  $R$  is in Boyce-Codd normal form (BCNF) if, for every one of its dependencies  $X \rightarrow Y$ , one of the following conditions hold true:
  - ▣  $X \rightarrow Y$  is a trivial functional dependency (i.e.,  $Y$  is a subset of  $X$ )
  - ▣  $X$  is a super key for schema  $R$
- Focusses on FDs within key attributes
- Every relation in BCNF is in 3NF, too  
... but not the other way round
- Informally: To test whether a relation is in BCNF, identify all the determinants and make sure that they are candidate keys



©RaunakMitra2

Source: <https://www.geeksforgeeks.org/boyce-codd-normal-form-bcnf/>

416

# NOTATIONS AND GUIDELINES

## NORMALIZATION

	UNF (1970)	1NF (1970)	2NF (1971)	3NF (1971)
Primary key (no duplicate tuples)	✓	✓	✓	✓
No repeating groups	✓	✓	✓	✓
Atomic columns (cells have single value) <sup>[8]</sup>	✗	✓	✓	✓
Every non-trivial functional dependency either does not begin with a proper subset of a candidate key or ends with a prime attribute (no partial functional dependencies of non-prime attributes on candidate keys) <sup>[8]</sup>	✗	✗	✓	✓
Every non-trivial functional dependency either begins with a superkey or ends with a prime attribute (no transitive functional dependencies of non-prime attributes on candidate keys) <sup>[8]</sup>	✗	✗	✗	✓
Every non-trivial functional dependency either begins with a superkey or ends with an elementary prime attribute <sup>[8]</sup>	✗	✗	✗	✗
Every non-trivial functional dependency begins with a superkey <sup>[8]</sup>	✗	✗	✗	✗
Every non-trivial multivalued dependency begins with a superkey <sup>[8]</sup>	✗	✗	✗	✗
Every join dependency has a superkey component <sup>[9]</sup>	✗	✗	✗	✗
Every join dependency has only superkey components <sup>[8]</sup>	✗	✗	✗	✗
Every constraint is a consequence of domain constraints and key constraints <sup>[8]</sup>	✗	✗	✗	✗
Every join dependency is trivial <sup>[8]</sup>	✗	✗	✗	✗

Source: [https://en.wikipedia.org/wiki/Database\\_normalization](https://en.wikipedia.org/wiki/Database_normalization) <sup>417</sup>

# NOTATIONS AND GUIDELINES

## NORMALIZATION – 1NF

- **There should be atomic attribute values only!**
- Disallows:
  - ▣ composite attributes
  - ▣ multivalued attributes
  - ▣ nested relations
- PK determines every atomic attribute value
- In SQL-92 it's not possible to have relations in Non First Normal Form (NFNF)

## NOTATIONS AND GUIDELINES

### NORMALIZATION – 2NF

- **Relation is in 1NF and every nonkey attribute is full functional dependent on the key!**
- No nonkey attribute should be functionally dependent on a part of the primary key
- Applies only to relations where the PK contains multiple attributes

## NOTATIONS AND GUIDELINES

### NORMALIZATION – 3NF

- **There should be no transitive dependency of a nonkey attribute on the primary key!**
- No nonkey attribute has a FD on another nonkey attribute
- Relation needs to be in 1NF and 2NF

## NOTATIONS AND GUIDELINES

### NORMALIZATION – RULES FOR DECOMPOSITION

- Should be lossless
- Tuples of the original relation can be restored when joining the decomposed relations
- Functional dependencies should be preserved in one of the decomposed relations
- ... so original FDs can be restored

## NOTATIONS AND GUIDELINES

### ASSIGNMENT: TRANSFORM THE TABLE INTO 1NF, 2NF, AND 3NF

Student No	Student Name	Major	Course No	Course Name	Instr No	Instr Name	Instr Loc	Grade
123	Smith	DB	1	DB	456	Jason	R04	1
			2	Math	567	Meyer	R01	1
			3	Physics	678	Fish	R02	2
234	Jones	Math	1	DB	456	Jason	R04	2
			2	Math	567	Meyer	R01	1
			4	OP	789	Dench	R07	3

## NOTATIONS AND GUIDELINES

### ASSIGNMENT: TRANSFORM THE TABLE INTO 1NF, 2NF, AND 3NF

Full_Name	Physical Address	Movies Rented	Salutation
Janet Jones	First Street Plot No 4	Pirates of the Caribbean, Clash of the Titans	Ms.
Robert Phil	3 <sup>rd</sup> Street No 34	Forgetting Sarah Marshal, Daddy's Little Girls	Mr.
Robert Phil	5 <sup>th</sup> Avenue No 4	Clash of the Titans	Mr.