DBMS

Dr. Media A. Ibrahim
Assist.Lec: Areen Hamad
Software Engineering
ISE Department
Media.ibrahim@epu.edu.iq
Areen.hamad@epu.edu.iq

2024-2025

Outlines

- Database Design.
- Normalization.
- Functional Dependency.
- Types Of Normalization.

Database Design

- The ability to design databases and associated applications is critical to the success of the modern enterprise.
- The design of the application schema will determine the usability and query ability of the application. Done incorrectly, the application and users will suffer until someone else is forced to rewrite it.

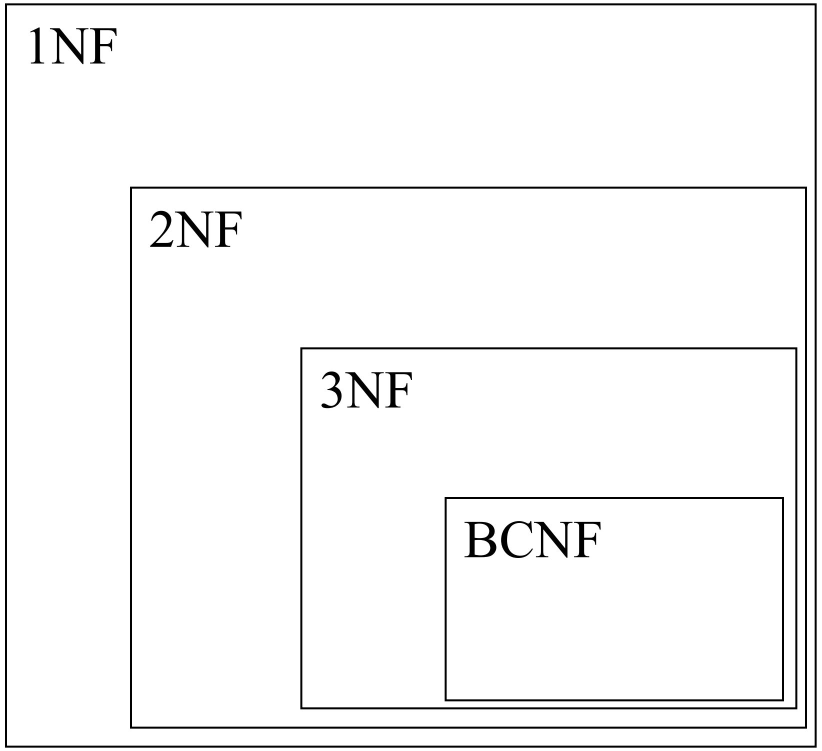
Normalization in DBMS

Normal Forms (NF): design standards based on database design theory
Normalization is the process of applying the NFs to table design to eliminate redundancy and create a more efficient organization of DB storage.
Each successive NF applies an increasingly stringent set of rules

The *objective* of normalization:
"*to create relations where every dependency is on the key, the whole key, and nothing but the key*".

First we introduce the concept of ***functional dependency***

Types Of Normalization



a relation in BCNF, is also in 3NF

a relation in 3NF is also in 2NF

a relation in 2NF is also in 1NF

Why Normalize Tables?

- Save typing of repetitive data
- Increase flexibility to query, sort, summarize, and group data (Simpler to manipulate data!)
- Avoid frequent restructuring of tables and other objects to accommodate new data
- Reduce disk space

Functional Dependencies

We say an attribute, B, has a *functional dependency* on another attribute, A, if for any two records, which have the same value for A, then the values for B in these two records must be the same. We illustrate this as:

A □ B

Example: Suppose we keep track of employee email addresses, and we only track one email address for each employee. Suppose each employee is identified by their unique employee number. We say there is a functional dependency of email address on employee number:

employee number □ email address

Functional Dependencies

<u>EmpNu</u>	**EmpEmail**	**EmpFname**	**EmpLname**
<u>m</u> 123	jdoe@abc.com	Joh	Doe
456	psmith@abc.com	Peter	Smith
555	alee1@abc.com	Alan	Lee
633	pdoe@abc.com	Peter	Doe
787	alee2@abc.com	Alan	Lee

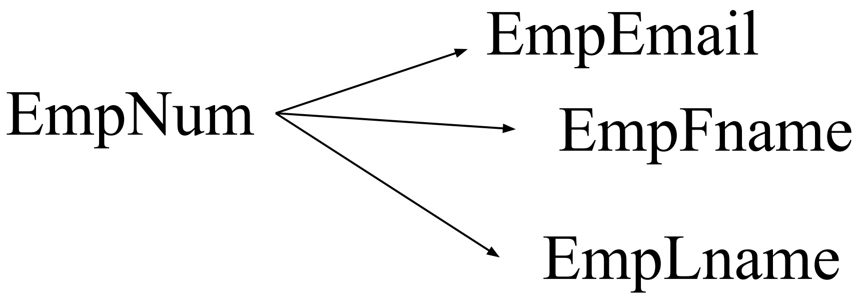
If EmpNum is the PK then the FDs:

EmpNum □ EmpEmail
EmpNum □ EmpFname
EmpNum □ EmpLname

Functional Dependencies

EmpNum □ EmpEmail
EmpNum □ EmpFname
EmpNum □ EmpLname

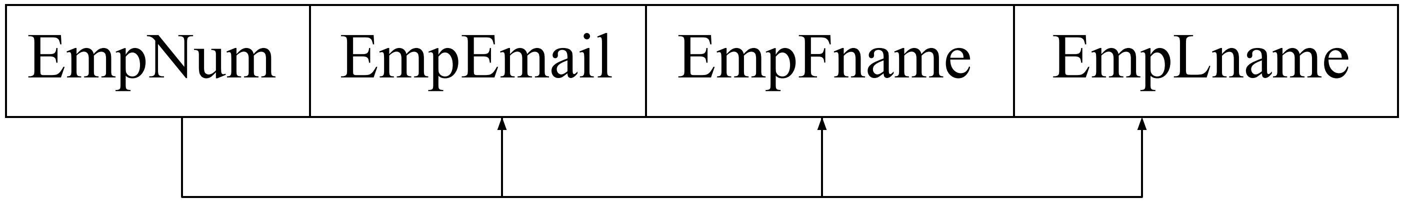
3 different ways you might see FDs depicted


EmpNum

EmpEmail

EmpFname

EmpLname


EmpNum	EmpEmail	EmpFname	EmpLname

Determinant

Functional Dependency

EmpNum □ EmpEmail

Attribute on the LHS is known as the ***determinant***

- EmpNum is a determinant of EmpEmail

Transitive dependency

Transitive dependency

Consider attributes A, B, and C, and where

A □ B and B □ C.

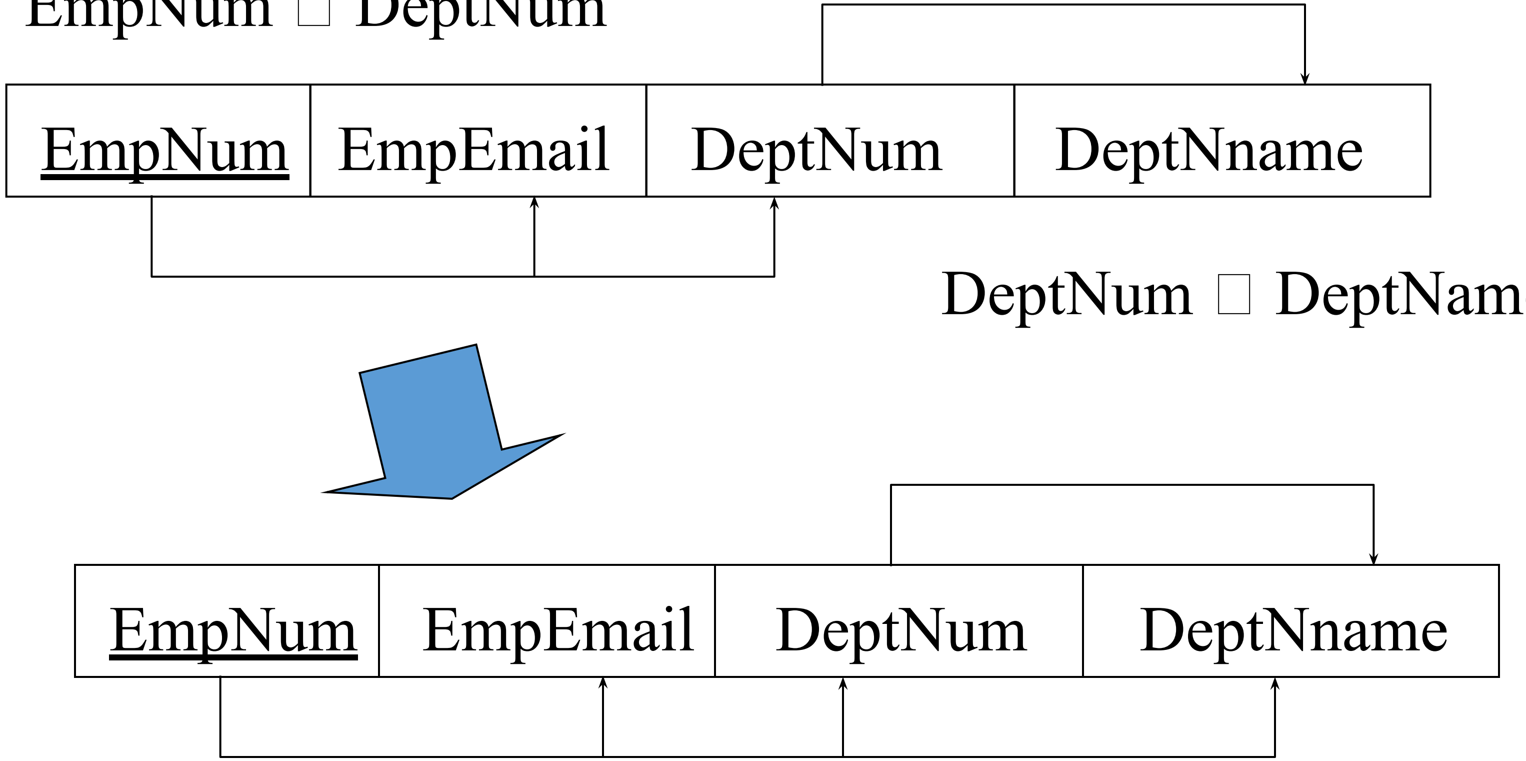
Functional dependencies are transitive, which means that

we also have the functional dependency A □ C

We say that C is transitively dependent on A through B.

Transitive dependency

EmpNum □ DeptNum



<u>EmpNum</u>	EmpEmail	DeptNum	DeptNname

DeptNum □ DeptName

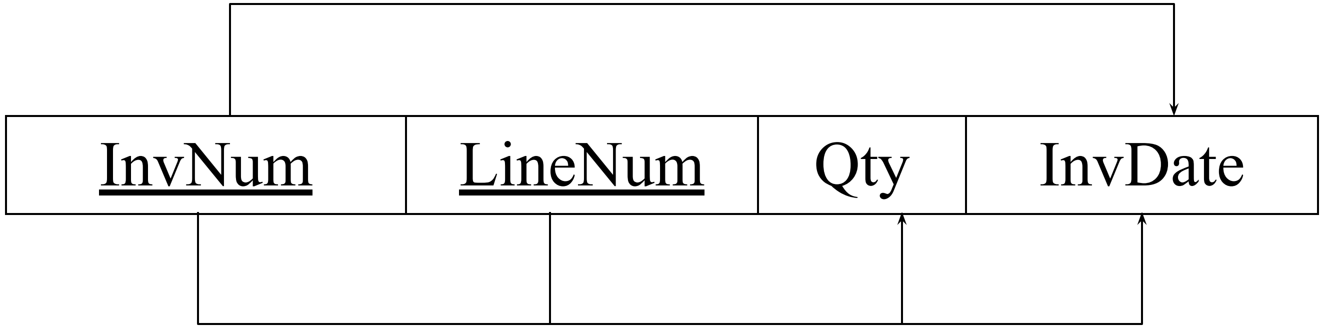
<u>EmpNum</u>	EmpEmail	DeptNum	DeptNname

DeptName is *transitively dependent* on EmpNum via DeptNum

EmpNum □ DeptName

Partial dependency

A **partial dependency** exists when an attribute B is functionally dependent on an attribute A, and A is a component of a multipart candidate key.



Candidate keys: {InvNum, LineNum} InvDate is *partially dependent* on {InvNum, LineNum} as InvNum is a determinant of InvDate and InvNum is part of a candidate key

First Normal Form (1NF)

A table is considered to be in 1NF if all the fields contain only atomic values (**no repeating group, no composite attributes**).

Example (Not 1NF)

ISBN	**Title**	**AuName**	**AuPhone**	**PubName**	**PubPhone**	**Price**
0-321-32132-1	Balloon	Sleepy, Snoopy, Grumpy	321-321-1111, 232-234-1234, 665-235-6532	Small House	714-000-0000	$34.00
0-55-123456-9	Main Street	Jones, Smith	123-333-3333, 654-223-3455	Small House	714-000-0000	$22.95
0-123-45678-0	Ulysses	Joyce	666-666-6666	Alpha Press	999-999-9999	$34.00
1-22-233700-0	Visual Basic	Roman	444-444-4444	Big House	123-456-7890	$25.00

1NF - Decomposition

1. Place all items that appear in the repeating group in a new table
2. Designate a primary key for each new table produced.
3. Duplicate in the new table the primary key of the table from which the repeating group was extracted or vice versa.

Example (1NF):

ISBN	Title	PubName	PubPhone	Price
0-321-32132-1	Balloon	Small House	714-000-0000	$34.00
0-55-123456-9	Main Street	Small House	714-000-0000	$22.95
0-123-45678-0	Ulysses	Alpha Press	999-999-9999	$34.00
1-22-233700-0	Visual Basic	Big House	123-456-7890	$25.00

ISBN	AuName	AuPhone
0-321-32132-1	Sleepy	321-321-1111
0-321-32132-1	Snoopy	232-234-1234
0-321-32132-1	Grumpy	665-235-6532
0-55-123456-9	Jones	123-333-3333
0-55-123456-9	Smith	654-223-3455
0-123-45678-0	Joyce	666-666-6666
1-22-233700-0	Roman	444-444-4444

Second Normal Form (2NF)

For a table to be in 2NF, there are two requirements

- The database is in first normal form
- All **nonkey** attributes in the table must be functionally dependent on the entire primary key

Note: *Remember that we are dealing with non-key attributes*

Example (Not 2NF)

PART (Primary Key)	WAREHOUSE (Primary Key)	QUANTITY	WAREHOUSE ADDRESS
P0010	Warehouse A	400	1608 New Field Road
P0010	Warehouse B	543	4141 Greenway Drive
P0010	Warehouse C	329	171 Pine Lane
P0020	Warehouse B	200	4141 Greenway Drive
P0020	Warehouse D	278	800 Massey Street

2NF - Decomposition

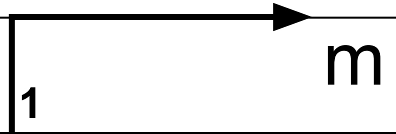
1. If a data item is fully functionally dependent on only a part of the primary key, move that data item and that part of the primary key to a new table.
2. If other data items are functionally dependent on the same part of the key, place them in the new table also

2NF – Decomposition(cont.)

Example (Convert to 2NF)

PART_STOCK TABLE

PART (Primary Key)	WAREHOUSE (Primary Key)	QUANTITY
P0010	Warehouse A	400
P0010	Warehouse B	543
P0010	Warehouse C	329
P0020	Warehouse B	200
P0020	Warehouse D	278



WAREHOUSE TABLE

WAREHOUSE (Primary Key)	WAREHOUSE_ADDRESS
Warehouse A	1608 New Field Road
Warehouse B	4141 Greenway Drive
Warehouse C	171 Pine Lane
Warehouse D	800 Massey Street

Third Normal Form (3NF)

For a table to be in 3NF, there are two requirements

- The table should be second normal form
- No attribute is transitively dependent on the primary key

Example (Not in 3NF)

EMPLOYEE_DEPARTMENT TABLE

EMPNO (Primary Key)	FIRSTNAME	LASTNAME	WORKDEPT	DEPTNAME
000290	John	Parker	E11	Operations
000320	Ramlal	Mehta	E21	Software Support
000310	Maude	Setright	E11	Operations

3NF - Decomposition

1. Move all items involved in transitive dependencies to a new entity.
2. Identify a primary key for the new entity.

Example (Convert to 3NF)

EMPLOYEE TABLE

EMPNO (Primary Key)	FIRSTNAME	LASTNAME	WORKDEPT
000290	John	Parker	E11
000320	Ramlal	Mehta	E21
000310	Maude	Setright	E11

DEPARTMENT TABLE

DEPTNO (Primary Key)	DEPTNAME
E11	Operations
E21	Software Support

Boyce-Codd Normal Form (BCNF)

- BCNF requires that the table is 3NF and only determinants are the candidate keys.

- **Example (Not in BCNF)**

ClientInterview

ClientNo	interviewDate	interviewTime	staffNo	roomNo
CR76	13-May-02	10.30	SG5	G101
CR76	13-May-02	12.00	SG5	G101
CR74	13-May-02	12.00	SG37	G102
CR56	1-Jul-02	10.30	SG5	G102

BCNF - Decomposition

1. Place the two candidate primary keys in separate entities
2. Place each of the remaining data items in one of the resulting entities according to its dependency on the primary key.

Example (Convert to BCNF)

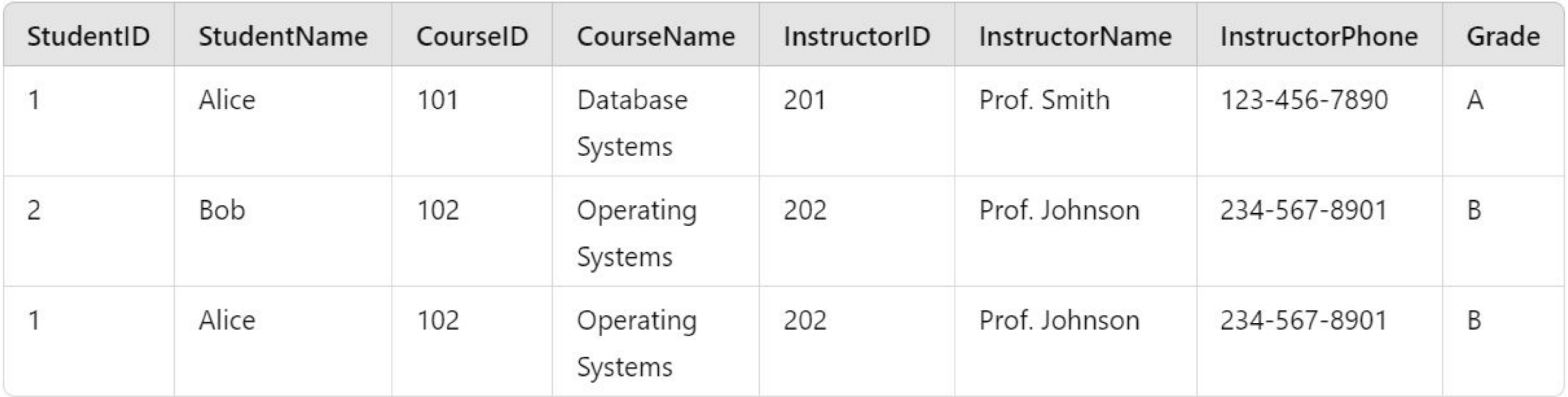
ClientNo	interviewDate	interviewTime	staffNo
CR76	13-May-02	10.30	SG5
CR76	13-May-02	12.00	SG5
CR74	13-May-02	12.00	SG37
CR56	1-Jul-02	10.30	SG5

staffNo	interviewDate	roomNo
SG5	13-May-02	G101
SG37	13-May-02	G102
SG5	1-Jul-02	G102

Conclusion

- Normalization is the process of efficiently organizing data in a database with two goals in mind
- First goal: eliminate redundant data
- Second Goal: ensure data dependencies make sense

Exercise: Normalize the Following Table TO 1,2,3NF



StudentID	StudentName	CourseID	CourseName	InstructorID	InstructorName	InstructorPhone	Grade
1	Alice	101	Database Systems	201	Prof. Smith	123-456-7890	A
2	Bob	102	Operating Systems	202	Prof. Johnson	234-567-8901	B
1	Alice	102	Operating Systems	202	Prof. Johnson	234-567-8901	B