



Prifysgol Cymru
Y Drindod Dewi Sant
University of Wales
Trinity Saint David

200
1822 - 2022

Normalisation

Man Geni Addysg Uwch yng Nghymru
Where Higher Education Began in Wales

www.uwtsd.ac.uk

Week:

30-Sep-2019

Today's Lecture

- Functional dependency
- Normalisation

What is Normalisation?

- **Normalisation** is a technique for producing a set of *suitable relations that support the data requirements* of an enterprise.

Employee

Man#	Name	Birthdate	JobHistory				Children	
123	Michael Swart	Nov. 22	JobDate	Title	SalaryHistory		Childname	Birthyear
			2000	Lackey	SalaryDate	Salary	Mini-me 1	2000
					2000	\$1,000,000	Mini-me 2	2002
					2001	\$2,000,000		
			2002	Senior Lackey	SalaryDate	Salary		
					2002	\$3,000,000		
					2003	\$4,000,000		

What is Normalisation?

- **Normalisation** is a technique for producing a set of *suitable relations that support the data requirements* of an enterprise.

Employee'

Man#	Name	Birthdate
123	Michael Swart	Nov. 22

JobHistory'

<u>Man#</u>	<u>JobDate</u>	Title
123	2000	Lackey
123	2002	Senior Lackey

Children'

Man#	ChildName	BirthYear
123	Mini-me 1	2000
123	Mini-me 2	2002

SalaryHistory'

<u>Man#</u>	<u>JobDate</u>	<u>SalaryDate</u>	Salary
123	2000	2000	\$1,000,000
123	2000	2001	\$2,000,000
123	2002	2002	\$3,000,000
123	2003	2003	\$4,000,000

Unnormalized LibraryBooks Table

BookID	Title	Authors	Genre	PublishedYear	CopiesAvailable
1	"The Great Gatsby"	F. Scott Fitzgerald	Fiction	1925	3
2	"1984"	George Orwell	Fiction	1949	2
3	"A Brief History..."	Stephen Hawking	Non-Fiction	1998	5
4	"The Catcher in..."	J.D. Salinger, John Doe	Fiction	1951	1
5	"Sapiens"	Yuval Noah Harari	Non-Fiction	2011	4

Anomalies (another simple example to understand anomalies)

student

Student number	Game	Fee structure
1	Basketball	500
2	Basketball	500
3	Basketball	500
4	Cricket	600
5	Cricket	600
6	Cricket	600
7	Tennis	400

student table has from all three anomalies

- Possibility of :
 - Inserting new game without student enrolment?
- Losing information
Eg: deletion of Roll no 7- lost tennis.
- Changes made in more than one place.

Introduction to Functional Dependencies

- What are Functional Dependencies (FD)?

Example Dataset 1: Student Information

Student_ID	Name	Department	HOD
101	Alice	Computer Science	Dr. John
102	Bob	Mechanical Eng.	Dr. Smith
103	Charlie	Computer Science	Dr. John

- Identify the attribute(s) that uniquely determine student's name and department?
- Identify the attribute(s) that uniquely determine the head of department?

Introduction to Functional Dependencies

- What are Functional Dependencies (FD)?

Example Dataset 1: Student Information

Student_ID	Name	Department	HOD
101	Alice	Computer Science	Dr. John
102	Bob	Mechanical Eng.	Dr. Smith
103	Charlie	Computer Science	Dr. John

- Student ID uniquely determines the student's name and department
- Department determines the Head of Department

Normalization - First Normal Form (1NF)

Example Dataset 2: Non-1NF

Student_ID	Name	Courses
101	Alice	CSE101, MTH102
102	Bob	PHY101
103	Charlie	CSE101, ENG101, PHY101

Normalization - First Normal Form (1NF)

Example Dataset 2: Non-1NF

Student_ID	Name	Courses
101	Alice	CSE101, MTH102
102	Bob	PHY101
103	Charlie	CSE101, ENG101, PHY101

1NF requires that all entries in a column are atomic (indivisible), and each row must be unique.

Normalization - First Normal Form (1NF)

Example Dataset 2: Non-1NF

Student_ID	Name	Courses
101	Alice	CSE101, MTH102
102	Bob	PHY101
103	Charlie	CSE101, ENG101, PHY101

contains
atomic values

1NF requires that all entries in a column are atomic (indivisible), and each row must be unique.

Convert to 1NF:

Student_ID	Name	Course
101	Alice	CSE101
101	Alice	MTH102
102	Bob	PHY101
103	Charlie	CSE101
103	Charlie	ENG101
103	Charlie	PHY101

Activity - First Normal Form (1NF)

Normalize the table below to 1NF

Order_ID	Customer	Products
001	Jane	TV, Laptop
002	John	Smartphone
003	Alice	TV, Camera, Headphones

Activity - First Normal Form (1NF)

1NF requires that all entries in a column are atomic (indivisible), and each row must be unique.

Order_ID	Customer	Products
001	Jane	TV, Laptop
002	John	Smartphone
003	Alice	TV, Camera, Headphones

Expected Result (1NF):

Order_ID	Customer	Product
001	Jane	TV
001	Jane	Laptop
002	John	Smartphone
003	Alice	TV
003	Alice	Camera
003	Alice	Headphones

Normalization - Second Normal Form (2NF)

- The table must be in 1NF.
- There should be no **partial dependency**

StudentID	StudentName	CourseID	CourseName	InstructorName	Grade
101	Alice Johnson	C101	Database Systems	Dr. Smith	A
101	Alice Johnson	C102	Data Structures	Dr. Johnson	B
102	Bob Brown	C101	Database Systems	Dr. Smith	B

Identify the composite key from the above table?

Normalization - Second Normal Form (2NF)

- The table must be in 1NF.
- There should be no **partial dependency**

StudentID	StudentName	CourseID	CourseName	InstructorName	Grade
101	Alice Johnson	C101	Database Systems	Dr. Smith	A
101	Alice Johnson	C102	Data Structures	Dr. Johnson	B
102	Bob Brown	C101	Database Systems	Dr. Smith	B

Identify the composite key from the above table?

composite key is **{StudentID, CourseID}**

Which FD is enough to uniquely determine studentname?

a) {StudentID, CourseID} -> {StudentName}

b) {StudentID} -> {StudentName}

Normalization - Second Normal Form (2NF)

- The table must be in 1NF.
- There should be no **partial dependency**

StudentID	StudentName	CourseID	CourseName	InstructorName	Grade
101	Alice Johnson	C101	Database Systems	Dr. Smith	A
101	Alice Johnson	C102	Data Structures	Dr. Johnson	B
102	Bob Brown	C101	Database Systems	Dr. Smith	B

Identify the composite key from the above table?

composite key is {**StudentID, CourseID**}

Which FD is enough to uniquely determine studentname?

a) {StudentID, CourseID} -> {StudentName}

b) {StudentID} -> {StudentName}

Normalization - Second Normal Form (2NF)

- The table must be in 1NF.
- There should be no **partial dependency**

StudentID	StudentName	CourseID	CourseName	InstructorName	Grade
101	Alice Johnson	C101	Database Systems	Dr. Smith	A
101	Alice Johnson	C102	Data Structures	Dr. Johnson	B
102	Bob Brown	C101	Database Systems	Dr. Smith	B

Identify the composite key from the above table?

composite key is {**StudentID**, **CourseID**}

Which FD is enough to uniquely determine CourseName?

a) {StudentID, CourseID} -> {CourseName}

b) {CourseID} -> {CourseName}

c) {CourseID} -> {InstructorName}

Normalization - Second Normal Form (2NF)

- The table must be in 1NF.
- There should be no **partial dependency**

StudentID	StudentName	CourseID	CourseName	InstructorName	Grade
101	Alice Johnson	C101	Database Systems	Dr. Smith	A
101	Alice Johnson	C102	Data Structures	Dr. Johnson	B
102	Bob Brown	C101	Database Systems	Dr. Smith	B

Can you create two tables to eliminate partial dependencies?

Normalization - Second Normal Form (2NF)

- The table must be in 1NF.
- There should be no **partial dependency**

StudentID	StudentName	CourseID	CourseName	InstructorName	Grade
101	Alice Johnson	C101	Database Systems	Dr. Smith	A
101	Alice Johnson	C102	Data Structures	Dr. Johnson	B
102	Bob Brown	C101	Database Systems	Dr. Smith	B

Students Table (after removing partial dependencies):

StudentID	StudentName
101	Alice Johnson
102	Bob Brown

Courses Table (after removing partial dependencies):

CourseID	CourseName	InstructorName
C101	Database Systems	Dr. Smith
C102	Data Structures	Dr. Johnson

Where to
place
'Grade'?

Normalization - Second Normal Form (2NF)

- The table must be in 1NF.
- There should be no **partial dependency**

Students Table (after removing partial dependencies):

StudentID	StudentName
101	Alice Johnson
102	Bob Brown

Courses Table (after removing partial dependencies):

CourseID	CourseName	InstructorName
C101	Database Systems	Dr. Smith
C102	Data Structures	Dr. Johnson

Enrollment Table (no partial dependencies, stores relationships and grades):

StudentID	CourseID	Grade
101	C101	A
101	C102	B
102	C101	B

Activity - Second Normal Form (2NF)

- The table must be in 1NF.
- There should be no **partial dependency**

EmployeeID	EmployeeName	DeptID	DeptName	ProjectID	ProjectName	HoursWorked
E001	Alice Johnson	D01	HR	P101	Payroll System	30
E001	Alice Johnson	D01	HR	P102	Recruitment App	25
E002	Bob Brown	D02	IT	P101	Payroll System	40
E003	Charlie Davis	D01	HR	P103	Employee Portal	20
E004	Daisy White	D03	Finance	P104	Budgeting App	35

Step 1: Identify the Composite Key

Step 2: Identify Partial Dependencies

Step 3: Break the Table into 2NF Tables

Activity - Second Normal Form (2NF)

- The table must be in 1NF.
- There should be no **partial dependency**

Employee Table (2NF)

EmployeeID	EmployeeName	DeptID
E001	Alice Johnson	D01
E002	Bob Brown	D02
E003	Charlie Davis	D01
E004	Daisy White	D03

Department Table (2NF)

DeptID	DeptName
D01	HR
D02	IT
D03	Finance

Activity - Second Normal Form (2NF)

- The table must be in 1NF.
- There should be no **partial dependency**

Project Table (2NF)

ProjectID	ProjectName
P101	Payroll System
P102	Recruitment App
P103	Employee Portal
P104	Budgeting App

Assignment Table (2NF)

This table contains the relationships between employees and the projects they are working on, including the HoursWorked.

EmployeeID	ProjectID	HoursWorked
E001	P101	30
E001	P102	25
E002	P101	40
E003	P103	20
E004	P104	35

Normalization - Third Normal Form (3NF)

- The table must be in 2NF.
- There should be no **transitive dependency**

StudentID	StudentName	CourseID	CourseName	InstructorName	InstructorPhone
101	Alice Johnson	C101	Math	Dr. Smith	123-456-7890
102	Bob Brown	C102	History	Dr. Adams	987-654-3210
101	Alice Johnson	C102	History	Dr. Adams	987-654-3210

StudentID → **StudentName**: The student's name depends on the student's ID.

CourseID → **CourseName**: The course name depends on the course's ID.

CourseID → **InstructorName**: The instructor's name depends on the course.

InstructorName → **InstructorPhone**: The instructor's phone number depends on the instructor.

Normalization - Third Normal Form (3NF)

- The table must be in 2NF.
- There should be no **transitive dependency**

StudentID → **StudentName**

CourseID → **CourseName**

CourseID → **InstructorName**

InstructorName → **InstructorPhone**

New Tables (in 3NF):

1. Student Table (No changes):

StudentID	StudentName
101	Alice Johnson
102	Bob Brown

2. Course Table (No transitive dependency):

CourseID	CourseName	InstructorName
C101	Math	Dr. Smith
C102	History	Dr. Adams

3. Instructor Table (Instructor phone moved to a separate table):

InstructorName	InstructorPhone
Dr. Smith	123-456-7890
Dr. Adams	987-654-3210

Database Normalization!

1NF made you clean up the mess,

2NF asked you to stop being so *partially dependent*, and

3NF taught you that transitive dependencies are more confusing than they sound!

Further Reading :

Chapter 11 Database Analysis and the DreamHome Case Study

Chapter 14: Normalisation

Database Systems: A Practical Approach to Design, Implementation, and Management Sixth edition by • Thomas Connolly • Carolyn Begg