



THE UNIVERSITY OF  
MELBOURNE

# INFO20003 Database Systems

Dr Renata Borovica-Gajic

Lecture 15  
Normalization

Week 8



MELBOURNE

- By the end of this lecture, you should be able to:
  - Define normalization
  - Explain and identify database anomalies
  - Define and identify functional dependencies
  - Normalize relations to:
    - 1st Normal Form (1NF)
    - 2nd Normal Form (2NF)
    - 3rd Normal Form (3NF)



- What happens if we don't normalize?



# What's wrong with the *organization* of data in this table?

Student ID#	Student Name	Campus Address	Degree	Phone	Subject ID	Subject Title	Lecturer Name	Lecturer Office	Lecturer Phone	Sem.	Grade
A121	Joy Egbert	166 Grattan Street	B.Com.	555-7771	ACC101	Accounting	Davern	T240C	8344-1846	1-11	H1
A121	Joy Egbert	166 Grattan Street	B.Com.	555-7771	ECO101	Economics	Smyth	T240F	8344-1868	1-11	H2B
A121	Joy Egbert	166 Grattan Street	B.Com.	555-7771	ECO104	Quant. M.	Collier	T240D	8344-5716	1-11	H2B
A121	Joy Egbert	166 Grattan Street	B.Com.	555-7771	FIN101	Finance.	James	T240D	8344-5275	1-11	H2A
A121	Joy Egbert	166 Grattan Street	B.Com.	555-7771	ACC103	Processes	Wise	T240E	8344-5309	1-11	H3
A123	Larry Mueller	302 Royal Parade	B.Com.	555-1235	ACC101	Accounting	Davern	T240C	8344-1846	1-11	H1
A123	Larry Mueller	302 Royal Parade	B.Com.	555-1235	ECO101	Economics	Smyth	T240F	8344-1868	1-11	H2B
A123	Larry Mueller	302 Royal Parade	B.Com.	555-1235	ECO104	Quant. M.	Collier	T240D	8344-5716	1-11	H2A
A123	Larry Mueller	302 Royal Parade	B.Com.	555-1235	FIN101	Finance.	James	T240D	8344-5275	1-11	H3
A124	Mike Guon	224 Swanston St.	B.Eco.	555-2214	ACC101	Accounting	Davern	T240C	8344-1846	1-11	H2A
A124	Mike Guon	224 Swanston St.	B.Eco.	555-2214	ECO101	Economics	Smyth	T240F	8344-1868	1-11	H2A
A124	Mike Guon	224 Swanston St.	B.Eco.	555-2214	ECO104	Quant. M.	Collier	T240D	8344-5716	1-11	H2B
A124	Mike Guon	224 Swanston St.	B.Eco.	555-2214	ACC103	Processes	Wise	T240E	8344-5309	1-11	H2B
A126	Jackie Judson	85 Barry Street	B.Eco.	555-1245	ACC101	Accounting	Davern	T240C	8344-1846	1-11	H1
A126	Jackie Judson	85 Barry Street	B.Eco.	555-1245	ECO101	Economics	Smyth	T240F	8344-1868	1-11	H2B
A126	Jackie Judson	85 Barry Street	B.Eco.	555-1245	ECO104	Quant. M.	Collier	T240D	8344-5716	1-11	H2B
A126	Jackie Judson	85 Barry Street	B.Eco.	555-1245	ACC103	Processes	Wise	T240E	8344-5309	1-11	H2A
...	...	...	...	...	...	...	...	...	...	...	...

problem ① duplication ② if we want to insert new subjects, will have problems unless a person enrol

③ if the last student enrolled was deleted, loss the information of subject

MELBOURNE

- Consider the following denormalized table (relation) :

Student-ID	Course-ID	Fee
130	C200	75
200	C300	100
250	C200	75
425	C400	150
500	C300	100
575	C500	50
...	...	...

- Insertion Anomaly:** A new course cannot be added until at least one student has enrolled (which comes first student or course?)
- Deletion Anomaly:** If student 425 withdraws, we lose all record of course C400 and its fee!
- Update Anomaly:** If the fee for course C200 changes, we have to change it in multiple records (rows), else the data will be (inconsistent.)

- A technique used to remove undesired redundancy from databases (Break one large table into several smaller tables).

A relation is normalized if  
all determinants are  
candidate keys

**How do we normalise?**



## Bill To

John  
Synex Inc  
128 AA Juanita Ave  
Glendora  
CA 91740 US

## Ship To

John  
Synex Inc  
128 AA Juanita Ave  
Glendora  
CA 91740 US

Date	14-Aug-2009	Order No		Sales Person	Charles Wooten
Shipping Date	13-Aug-2009	Shipping Terms		Terms	COD
ID	SKU / Description	Unit Price (USD)	Qty	Amount (USD)	
PS.V880.005	AMD Athlon X2DC-7450, 2.4GHz/1GB/160GB/SMP-DVD/VB	580.00	6.00	3,480.00	
PS.V880.037	PDC-E5300 - 2.6GHz/1GB/320GB/SMP-DVD/FDD/VB	645.00	4.00	2,580.00	
LC.V890.002	LG 18.5" WLCD	230.00	10.00	2,300.00	
HP.Q754.071	HP LaserJet 5200	1,103.00	1.00	1,103.00	

# Invoice example – Spreadsheet Format

MELBOURNE

Invoice Number	Date	Customer Name	Customer Address	Sales Person	Terms	Product ID	Product Name	Unit Price	Quantity	Amount	Sub Total
INV0012	14-Aug-09	John / Synex	128 Juanita Ave...	Charles Wooten	COD	PSV880.006	AMD Athlon X2DC	580	6	3480	9463
						PSV880.037	PDC E5300	645	4	2580	
						LC.V890.002	LG 8.5" LCD	230	10	2300	
						HPQ754.071	HP LaserJet 5200	1103	1	1103	
INV0013	15-Aug-09	Mary / ThisCo	123 Smith Street...	Charles Wooten	COD	HP Q754.071	HP LaserJet 5200	1103	2	2206	3356
						LCV890.002	LG 8.5" LCD	230	5	1150	

information  
about invoice

information about product

This is not relational model





Invoice Number	Date	Customer Name	Customer Address	Sales Person	Terms	Sub Total	Discount	Sales Tax	Shipping
INV0012	14-Aug-09	John / Synex	128 Juanita Ave...	Charles Wooten	COD	9463	0	780.70	0
INV0013	15-Aug-09	Mary / ThisCo	123 Smith Street...	Charles Wooten	COD	3356	0	100	0

Product ID	Product Name	Unit Price	Quantity	Amount
PSV880.006	AMD Athlon X2DC	580	6	3480
PSV880.037	PDC E5300	645	4	2580
LC.V890.002	LG 8.5" LCD	230	10	2300
HPQ754.071	HP LaserJet 5200	1103	1	1103
HPQ754.071	HP LaserJet 5200	1103	2	2206
LCV890.002	LG 8.5" LCD	230	5	1150

Break into two  
But...  
How do we connect?

*one invoice → many products*



# Invoice example – Spreadsheet Format

Invoice Number	Date	Customer Name	Customer Address	Sales Person	Terms	Sub Total	Discount	Sales Tax	Shipping
INV0012	14-Aug-09	John / Synex	128 Juanita Ave...	Charles Wooten	COD	9463	0	780.70	0
INV0013	15-Aug-09	Mary / ThisCo	123 Smith Street...	Charles Wooten	COD	3356	0	100	0

Product ID	Product Name	Unit Price	Quantity	Amount	Invoice Number
PSV880.006	AMD Athlon X2DC	580	6	3480	INV0012
PSV880.037	PDC E5300	645	4	2580	INV0012
LC.V890.002	LG 8.5" LCD	230	10	2300	INV0012
HPQ754.071	HP LaserJet 5200	1103	1	1103	INV0012
HPQ754.071	HP LaserJet 5200	1103	2	2206	INV0013
LCV890.002	LG 8.5" LCD	230	5	1150	INV0013

Add FK



# Invoice example – Spreadsheet Format

Invoice Number	Date	Customer Name	Customer Address	Sales Person	Terms	Sub Total	Discount	Sales Tax	Shipping
INV0012	14-Aug-09	John / Synex	128 Juanita Ave...	Charles Wooten	COD	9463	0	780.70	0
INV0013	15-Aug-09	Mary / ThisCo	123 Smith Street...	Charles Wooten	COD	3356	0	100	0

Product ID	Product Name	Unit Price	Quantity	Amount	Invoice Number
PSV880.006	AMD Athlon X2DC	580	6	3480	INV0012
PSV880.037	PDC E5300	645	4	2580	INV0012
LC.V890.002	LG 8.5" LCD	230	10	2300	INV0012
HPQ754.071	HP LaserJet 5200	1103	1	1103	INV0012
HPQ754.071	HP LaserJet 5200	1103	2	2206	INV0013
LCV890.002	LG 8.5" LCD	230	5	1150	INV0013

This is about product

This is about Order (invoice)



# Invoice example – Spreadsheet Format

Invoice Number	Date	Customer Name	Customer Address	Sales Person	Terms	Sub Total	Discount	Sales Tax	Shipping
INV0012	14-Aug-09	John / Synex	128 Juanita Ave...	Charles Wooten	COD	9463	0	780.70	0
INV0013	15-Aug-09	Mary / ThisCo	123 Smith Street...	Charles Wooten	COD	3356	0	100	0

Product ID	Quantity	Amount	Invoice Number
PSV880.006	6	3480	INV0012
PSV880.037	4	2580	INV0012
LC.V890.00 2	10	2300	INV0012
HPQ754.07 1	1	1103	INV0012
HPQ754.07 1	2	2206	INV0013
LCV890.00	5	1150	INV0013

Break into two

Product ID	Product Name	Unit Price
PSV880.006	AMD Athlon X2DC	580
PSV880.037	PDC E5300	645
LC.V890.00 2	LG 8.5" LCD	230
HPQ754.07 1	HP LaserJet 5200	1103

# Invoice example – Spreadsheet Format

Invoice Number	Date	Customer Name	Customer Address	Sales Person	Terms	Sub Total	Discount	Sales Tax	Shipping
INV0012	14-Aug-09	John / Synex	128 Juanita Ave...	Charles Wooten	COD	9463	0	780.70	0
INV0013	15-Aug-09	Mary / ThisCo	123 Smith Street...	Charles Wooten	COD	3356	0	100	0

"derived attribute" → redundant

Product ID	Quantity	Amount	Invoice Number
PSV880.006	6	3480	INV0012
PSV880.037	4	2580	INV0012
LC.V890.002	10	2300	INV0012
HPQ754.071	1	1103	INV0012
HPQ754.071	2	2206	INV0013
LCV890.00	5	1150	INV0013

What about amount?

Product ID	Product Name	Unit Price
PSV880.006	AMD Athlon X2DC	580
PSV880.037	PDC E5300	645
LC.V890.002	LG 8.5" LCD	230
HPQ754.071	HP LaserJet 5200	1103

# Invoice example – Spreadsheet Format

Invoice Number	Date	Customer Name	Customer Address	Sales Person	Terms	Sub Total	Discount	Sales Tax	Shipping
INV0012	14-Aug-09	John / Synex	128 Juanita Ave...	Charles Wooten	COD	9463	0	780.70	0
INV0013	15-Aug-09	Mary / ThisCo	123 Smith Street...	Charles Wooten	COD	3356	0	100	0

Product ID	Quantity	Invoice Number
PSV880.006	6	INV0012
PSV880.037	4	INV0012
LC.V890.002	10	INV0012
HPQ754.071	1	INV0012
HPQ754.071	2	INV0013
LCV890.00	5	INV0013

What about sales person?

Could be derived

Product ID	Product Name	Unit Price
PSV880.006	AMD Athlon X2DC	580
PSV880.037	PDC E5300	645
LC.V890.002	LG 8.5" LCD	230
HPQ754.071	HP LaserJet 5200	1103

# Invoice example – Spreadsheet Format

Invoice Number	Date	Customer Name	Customer Address	Sales Person ID	Terms	Sub Total	Discount	Sales Tax	Shipping
INV0012	14-Aug-09	John / Synex	128 Juanita Ave...	1	COD	9463	0	780.70	0
INV0013	15-Aug-09	Mary / ThisCo	123 Smith Street...	1	COD	3356	0	100	0

What about customer?

Product ID	Quantity	Invoice Number
PSV880.006	6	INV0012
PSV880.037	4	INV0012
LC.V890.002	10	INV0012
HPQ754.071	1	INV0012
HPQ754.071	2	INV0013
LCV890.00	5	INV0013

Sales Person ID	Sales Person
1	Charles Wooten

Product ID	Product Name	Unit Price
PSV880.006	AMD Athlon X2DC	580
PSV880.037	PDC E5300	645
LC.V890.002	LG 8.5" LCD	230
HPQ754.071	HP LaserJet 5200	1103

# Invoice example – Spreadsheet Format

Invoice Number	Date	Customer ID	Sales Person ID	Terms	Sub Total	Discount	Sales Tax	Shipping
INV0012	14-Aug-09	1	1	COD	9463	0	780.70	0
INV0013	15-Aug-09	2	1	COD	3356	0	100	0

Product ID	Quantity	Invoice Number
PSV880.006	6	INV0012
PSV880.037	4	INV0012
LC.V890.002	10	INV0012
HPQ754.071	1	INV0012
HPQ754.071	2	INV0013
LCV890.00	5	INV0013

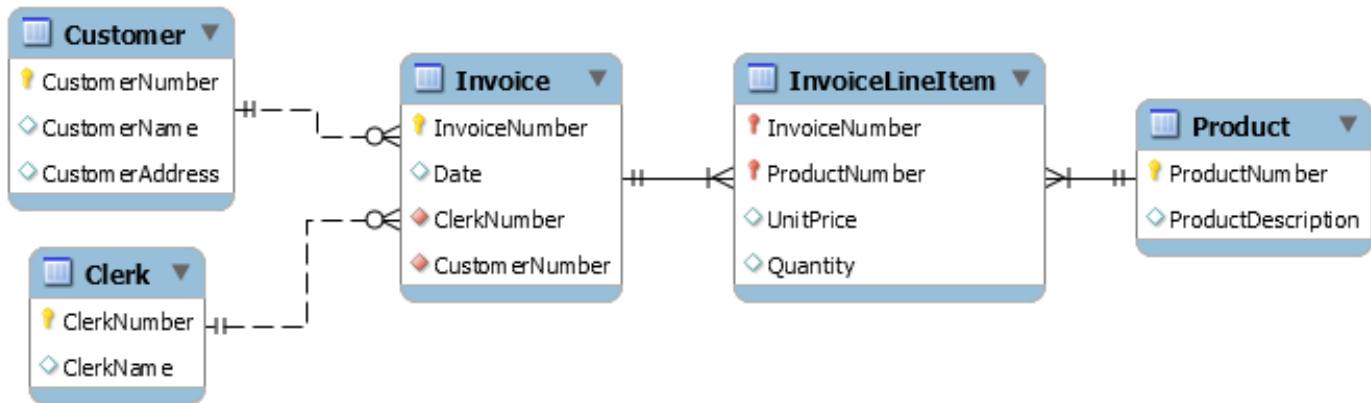
Sales Person ID	Sales Person
1	Charles Wooten

Customer ID	Customer Name	Customer Address
1	John / Synex	128 Juanita Ave...
2	Mary / ThisCo	123 Smith Street...

Product ID	Product Name	Unit Price
PSV880.006	AMD Athlon X2DC	580
PSV880.037	PDC E5300	645
LC.V890.002	LG 8.5" LCD	230
HPQ754.071	HP LaserJet 5200	1103



- We can name the relations now
  - Customer (CustomerNumber, CustomerName, CustomerAddress)
  - Clerk (ClerkNumber, ClerkName)
  - Product (ProductNumber, ProductDescription)
  - Invoice (InvoiceNumber, Date, *CustomerNumber*, *ClerkNumber*)
  - InvoiceLineItem (InvoiceNumber, ProductNumber, UnitPrice, Quantity)



*from normalisation*

*start from final schema (denormalised & not of issue)*

*→ fixed by breaking tables and connection between table*

- Now let's go back to theoretical concepts...

MELBOURNE

- A functional dependency concerns values of attributes in a relation
  - A set of attributes  $X$  **determines** another set of attributes  $Y$  if each value of  $X$  is associated with only one value of  $Y$ 
    - Written  $X \rightarrow Y$
    - $X$  determines  $Y$  (If I know  $X$  then I also know  $Y$ )
- 
- Emp#  $\longrightarrow$  Emp-name
  - Emp#  $\longrightarrow$  Salary

- **Determinants**  $(\boxed{X, Y} \rightarrow Z)$   
– the attribute(s) on the left hand side of the arrow  
*X, Y key attribute*  
 $A(\underline{X}, \underline{Y}, Z, D)$   
*X and Y are primary composite key*
- **Key and Non-Key attributes**  
– each attribute is either part of the primary key or it is not
- **Partial functional dependency**  $(Y \rightarrow Z)$   
– a functional dependency of one or more non-key attributes upon part (but not all) of the primary key
- **Transitive dependency**  $(Z \rightarrow D)$   
– a functional dependency between 2 (or more) non-key attributes

# Functional dependencies can be identified using Armstrong's Axioms

$A = (X_1, X_2, \dots, X_n)$  and  $B = (Y_1, Y_2, \dots, Y_n)$

1. Reflexivity:  $B \subseteq A \Rightarrow A \rightarrow B$

*if I know the student\_ID and name,  
obviously I know their name*

**Example:** Student\_ID, name  $\rightarrow$  name

2. Augmentation:  $A \rightarrow B \Rightarrow AC \rightarrow BC$

**Example:** Student\_ID  $\rightarrow$  name  $\Rightarrow$  Student\_ID, surname  $\rightarrow$  name, surname

3. Transitivity:  $A \rightarrow B$  and  $B \rightarrow C \Rightarrow A \rightarrow C$

*" $B \subseteq A, C \subseteq B \Rightarrow C \subseteq A$ "*

**Example:** ID  $\rightarrow$  birthdate and birthdate  $\rightarrow$  age then ID  $\rightarrow$  age

First Normal Form



Second Normal Form



Third Normal Form



★ for the relation  
to be in the third normal form  
the first two need to be satisfied

Keep atomic data/Remove repeating groups

*relational model is flat → each attribute store a single value  
⇒ whenever have multiple value to store in attribute,*

Remove partial dependencies

*need to be a separate table*

*non-key attribute depend on part of primary key*

Remove transitive dependencies

*non-key depend on another non-key attribute*

Remove Repeating Groups

- repeating groups of attributes cannot be represented in a flat, two dimensional table
- removing cells with multiple values (keep atomic data)

Set of values

Example: Order-Item (Order#, Customer#, (Item#, Desc, Qty))

- Order-Item (Order#, Customer#, (Item#, Desc, Qty))



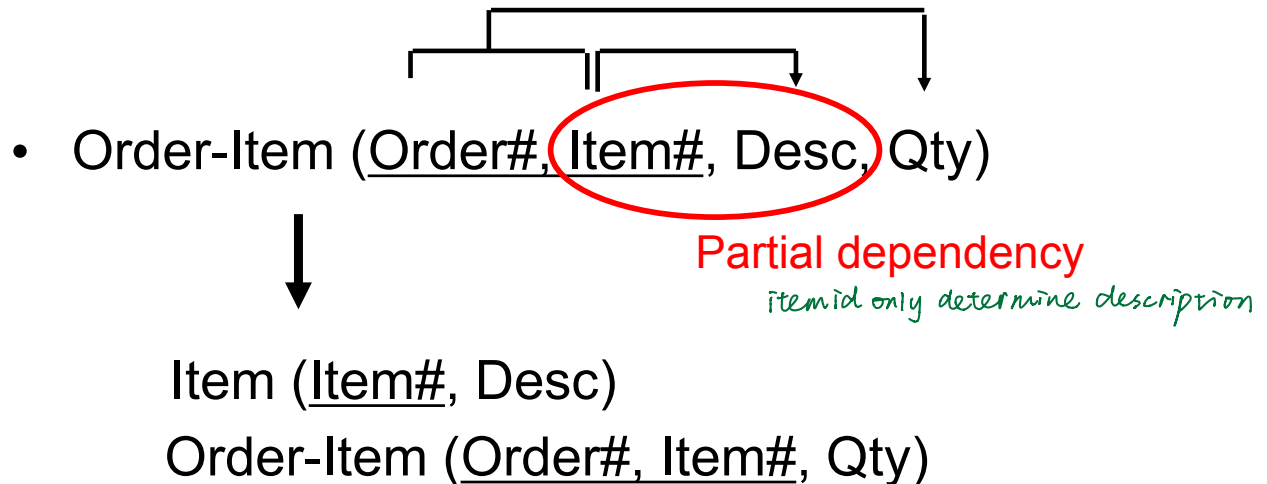
- Order-Item (Order#, Item#, Desc, Qty)
- Order (Order#, Customer#)

Break them into two  
Use PK/FK to connect

## Remove Partial Dependencies

a non-key attribute cannot be identified  
by part of a composite key

Example: Order-Item (Order#, Item#, Desc, Qty)





Order-Item (Order#, Item#, Desc, Qty)

Order#	Item#	Desc	Qty
27	873	nut	2
28	402	bolt	1
28	873	nut	10
30	495	washer	50

- *UPDATE*      change item desc in many places
- *DELETE*      data for last item lost when last order for that item is deleted
- *INSERT*      cannot add new item until it is ordered

## Order-Item

Order#	Item#	Qty
27	873	2
28	402	1
28	873	10
30	495	50

*delete last  
order for item,  
but item  
remains*

## Item

*add new item at any time*

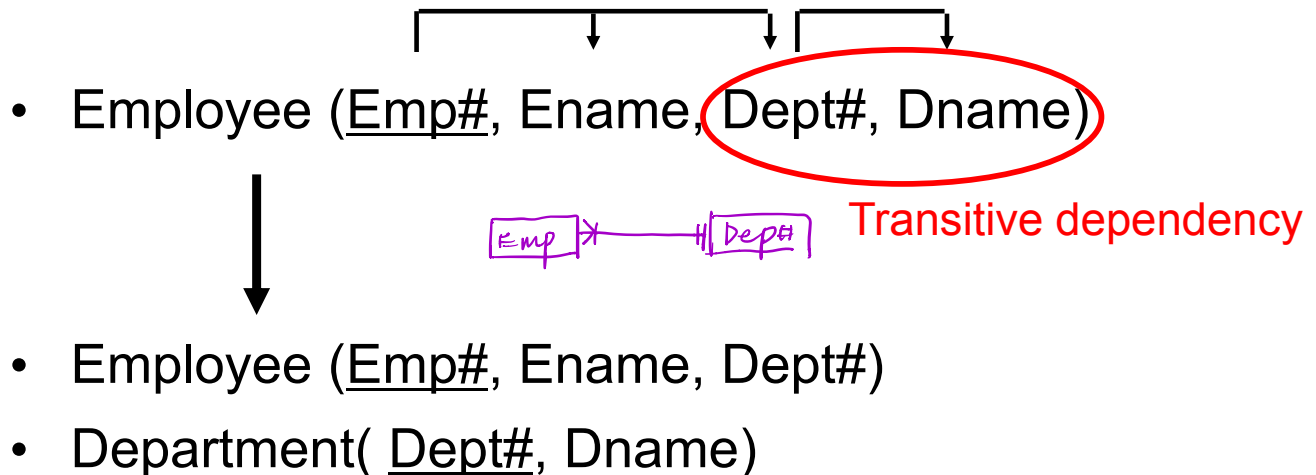
Item#	Desc
873	nut
402	bolt
495	washer

*change item  
description in one  
place*

## Remove Transitive Dependencies

a non-key attribute **cannot be identified**  
by another non-key attribute

Example: Employee (Emp#, Ename, Dept#, Dname)



Example: Employee (Emp#, Ename, Dept#, Dname)

Emp#	Ename	Dept#	Dname
10	Smith	D5	MIS
20	Jones	D7	Finance
25	Smith	D7	Finance
30	Black	D8	Sales

- *UPDATE*      change dept name in many places
- *DELETE*      data for dept lost when last employee for that dept is deleted
- *INSERT*      cannot add new dept until an employee is allocated to it

Employee

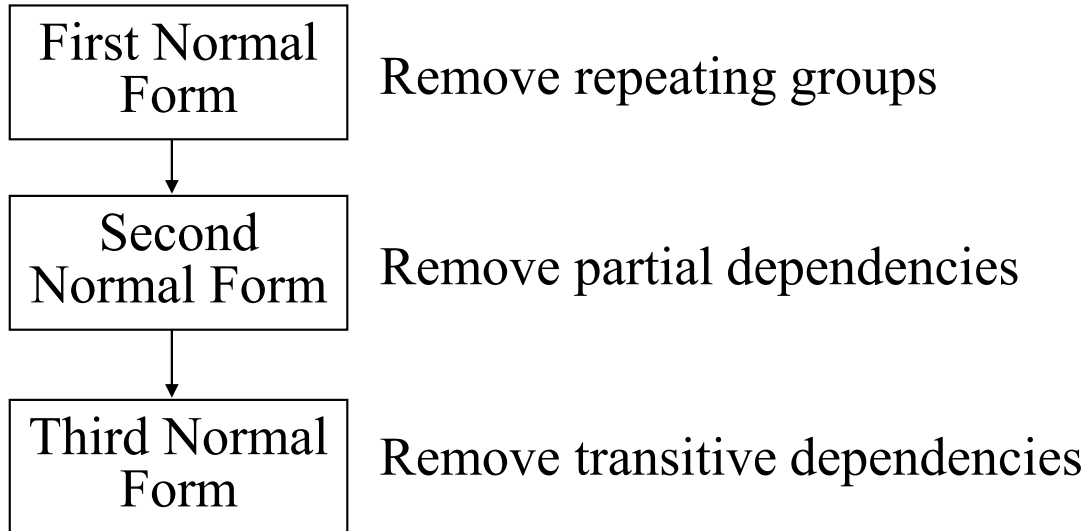
Emp#	Ename	Dept#
10	Smith	D5
20	Jones	D7
25	Smith	D7
30	Black	D8

*delete last emp in  
dept, but dept  
remains*

*add new dept at any time*

Dept#	Dname
D5	MIS
D7	Finance
D8	Sales

*change dept  
name in one  
place*



## Normalisation:

- Normalised relations contains a minimum amount of redundancy and allow users to **insert, modify, and delete** rows in tables **without errors or inconsistencies (anomalies)**

## Denormalization:

*data warehouse    only query data → denormalised table*

- **The pay-off: query speed.** *faster to read data from big table*
- The price: extra work on updates to keep redundant data consistent.
- Denormalization may be used to **improve performance of time-critical operations.**



MELBOURNE

- Normalisation Process
- Anomalies
- Functional dependencies





MELBOURNE

- Hands on Normalization