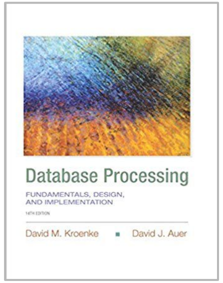
Fairfield University
DOLAN
School of Business



Database Processing

FUNDAMENTALS, DESIGN,
AND IMPLEMENTATION

14TH EDITION

David M. Kroenke ▪ David J. Auer

Databases for Analytics

Kroenke / Auer
Chapter 3

Learning Objectives

- **Skills:** You should know how to ...
  - Use normal forms to eliminate unnecessary redundancy and potential data anomalies
- **Theory:** You should be able to explain ...
  - The elements of the relational data model
  - How design can cause data anomalies
  - Why a database should be normalized and when it should not be normalized

Context

When might an analyst really need to know about the relational model and table design?

The Perpetual Dilemma

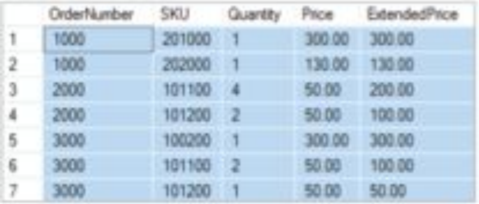
- We have received one or more tables of existing data with lots of redundancies
- The data is to be stored in a new database.
- QUESTION: Should the data be stored as received, or should it be transformed for storage?
- ANSWER: It depends, but we should at least know what can go wrong if we choose incorrectly

How Many Tables?

Should we store these two tables as they are, or should we combine them into one table in our new database?

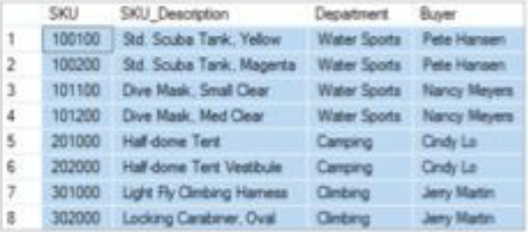
Can we somehow do both?

ORDER_ITEM



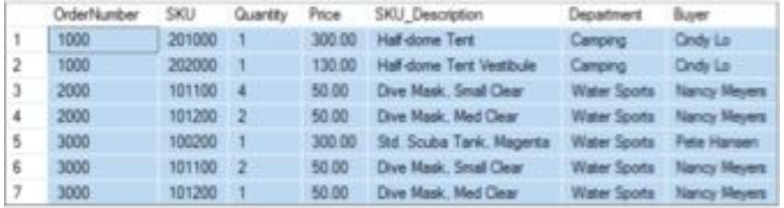
	OrderNumber	SKU	Quantity	Price	ExtendedPrice
1	1000	201000	1	300.00	300.00
2	1000	202000	1	130.00	130.00
3	2000	101100	4	50.00	200.00
4	2000	101200	2	50.00	100.00
5	3000	100200	1	300.00	300.00
6	3000	101100	2	50.00	100.00
7	3000	101200	1	50.00	50.00

SKU_DATA



	SKU	SKU_Description	Department	Buyer
1	100100	Std. Scuba Tank, Yellow	Water Sports	Pete Hansen
2	100200	Std. Scuba Tank, Magenta	Water Sports	Pete Hansen
3	101100	Dive Mask, Small Clear	Water Sports	Nancy Meyers
4	101200	Dive Mask, Med Clear	Water Sports	Nancy Meyers
5	201000	Half-dome Tent	Camping	Cindy Lo
6	202000	Half-dome Tent Vestibule	Camping	Cindy Lo
7	301000	Light Fly Climbing Harness	Climbing	Jerry Martin
8	302000	Locking Carabiner, Oval	Climbing	Jerry Martin

SKU_ITEM

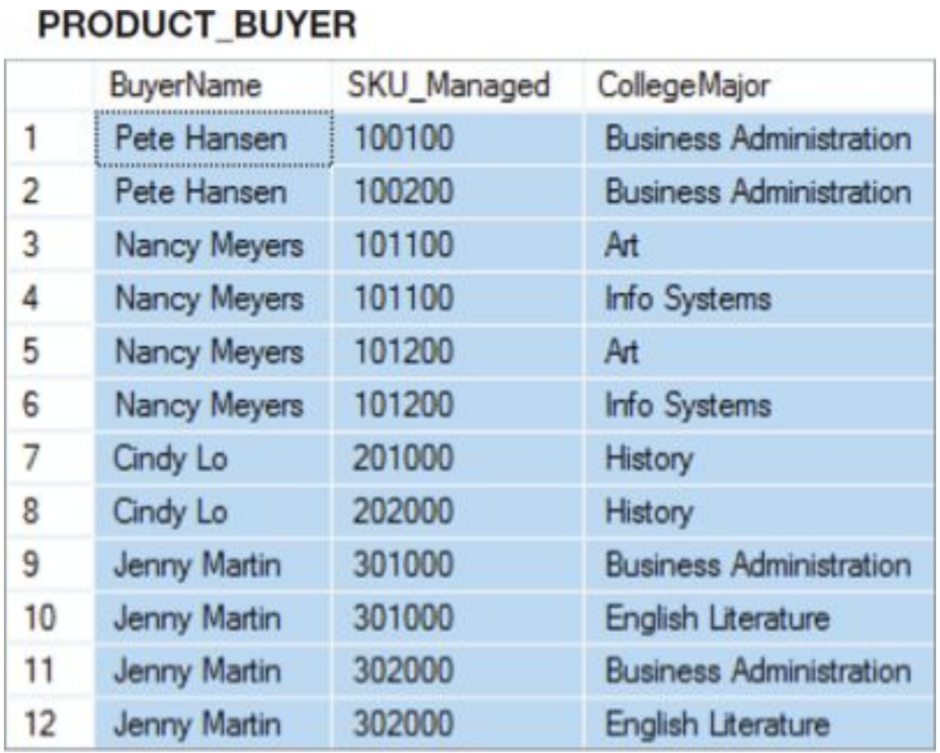


	OrderNumber	SKU	Quantity	Price	SKU_Description	Department	Buyer
1	1000	201000	1	300.00	Half-dome Tent	Camping	Cindy Lo
2	1000	202000	1	130.00	Half-dome Tent Vestibule	Camping	Cindy Lo
3	2000	101100	4	50.00	Dive Mask, Small Clear	Water Sports	Nancy Meyers
4	2000	101200	2	50.00	Dive Mask, Med Clear	Water Sports	Nancy Meyers
5	3000	100200	1	300.00	Std. Scuba Tank, Magenta	Water Sports	Pete Hansen
6	3000	101100	2	50.00	Dive Mask, Small Clear	Water Sports	Nancy Meyers
7	3000	101200	1	50.00	Dive Mask, Med Clear	Water Sports	Nancy Meyers

Coherency Problems ...

To understand why this is a very strange table, consider how you would add the fact that **Nancy Meyers** is now managing **SKU 101300!**

Do we have to provide a college major?



PRODUCT_BUYER

	BuyerName	SKU_Managed	CollegeMajor
1	Pete Hansen	100100	Business Administration
2	Pete Hansen	100200	Business Administration
3	Nancy Meyers	101100	Art
4	Nancy Meyers	101100	Info Systems
5	Nancy Meyers	101200	Art
6	Nancy Meyers	101200	Info Systems
7	Cindy Lo	201000	History
8	Cindy Lo	202000	History
9	Jenny Martin	301000	Business Administration
10	Jenny Martin	301000	English Literature
11	Jenny Martin	302000	Business Administration
12	Jenny Martin	302000	English Literature

What we need to know ...

- The Relational Database Model
  - What makes a good table design for a given application?
  - How is the data organized into tables?
- The Purpose and Process of Table Normalization
  - What kinds of bugs might corrupt our data?
  - How does table design affect data integrity

Relational Model

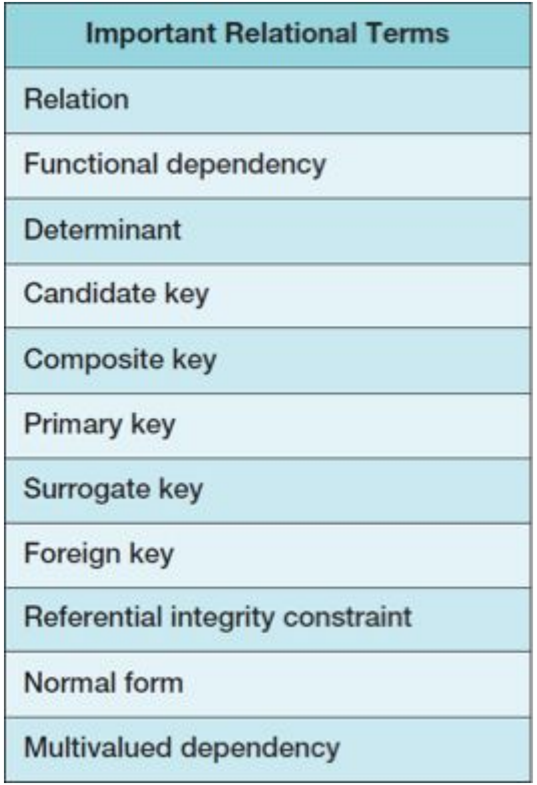
The Relational part of the RDBMS

Origins

- Introduced in a paper published in 1970.
- Created by E.F. Codd
    - He was an IBM engineer
    - The model used mathematics known as “relational algebra”
- Now the standard model for commercial DBMS products.
- SQL was designed to implement the relational model on 1970s-era mainframe hardware

Relational Terminology

We've already seen most of these terms, but now we will get a bit more precise as what they mean.



Important Relational Terms
Relation
Functional dependency
Determinant
Candidate key
Composite key
Primary key
Surrogate key
Foreign key
Referential integrity constraint
Normal form
Multivalued dependency

Entities

An **entity** is some identifiable **thing** (not data) that users want to track:

- Customers
- Computers
- Sales

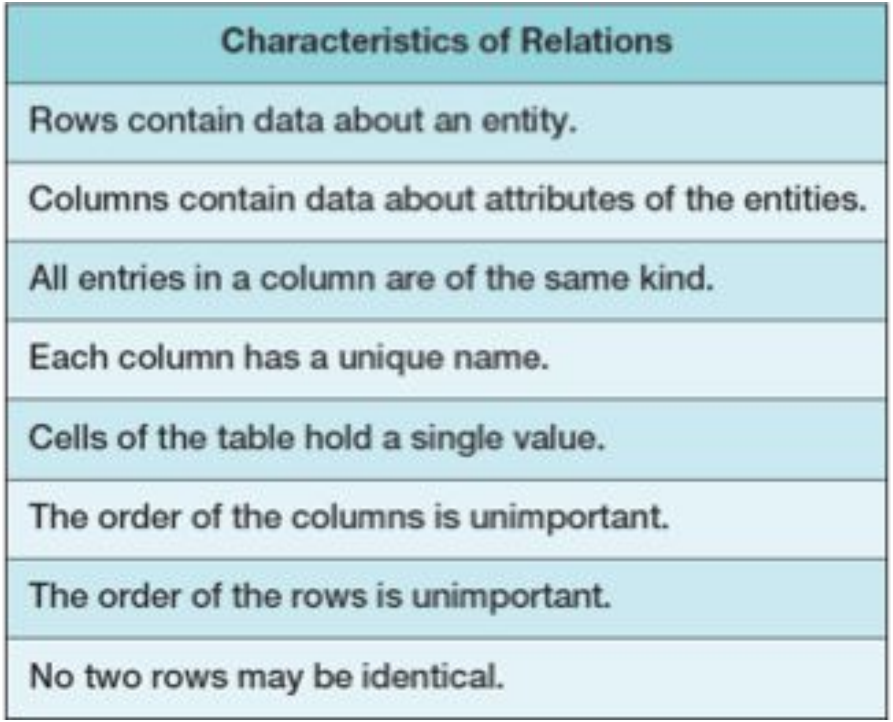
Is a dollar bill an entity? It depends!

Note: by *identifiable*, we mean that each entity has a unique identity.

Relations

Relational DBMS products store data about entities in **relations**, which are a special type of two dimensional table.

The term *relation* comes from math. You likely learned about relations in 8-th grade.



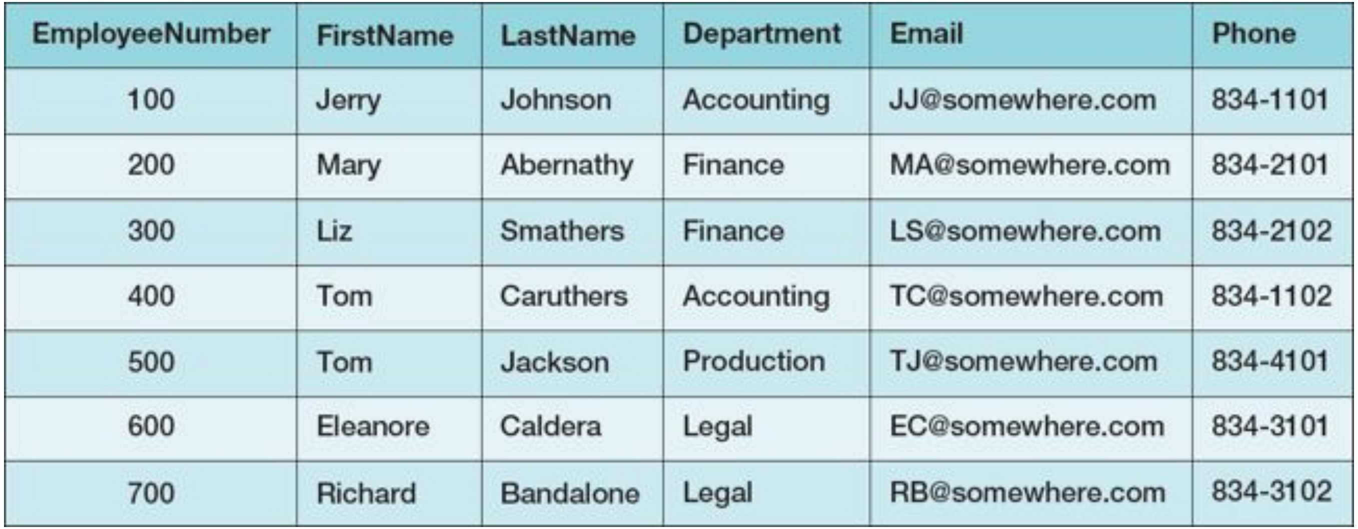
Characteristics of Relations
Rows contain data about an entity.
Columns contain data about attributes of the entities.
All entries in a column are of the same kind.
Each column has a unique name.
Cells of the table hold a single value.
The order of the columns is unimportant.
The order of the rows is unimportant.
No two rows may be identical.

A Coherent Relation

One entity per row

One attribute (and datatype) per column

No duplicate/repeated rows or columns



EmployeeNumber	FirstName	LastName	Department	Email	Phone
100	Jerry	Johnson	Accounting	JJ@somewhere.com	834-1101
200	Mary	Abernathy	Finance	MA@somewhere.com	834-2101
300	Liz	Smathers	Finance	LS@somewhere.com	834-2102
400	Tom	Caruthers	Accounting	TC@somewhere.com	834-1102
500	Tom	Jackson	Production	TJ@somewhere.com	834-4101
600	Eleanore	Caldera	Legal	EC@somewhere.com	834-3101
700	Richard	Bandalone	Legal	RB@somewhere.com	834-3102

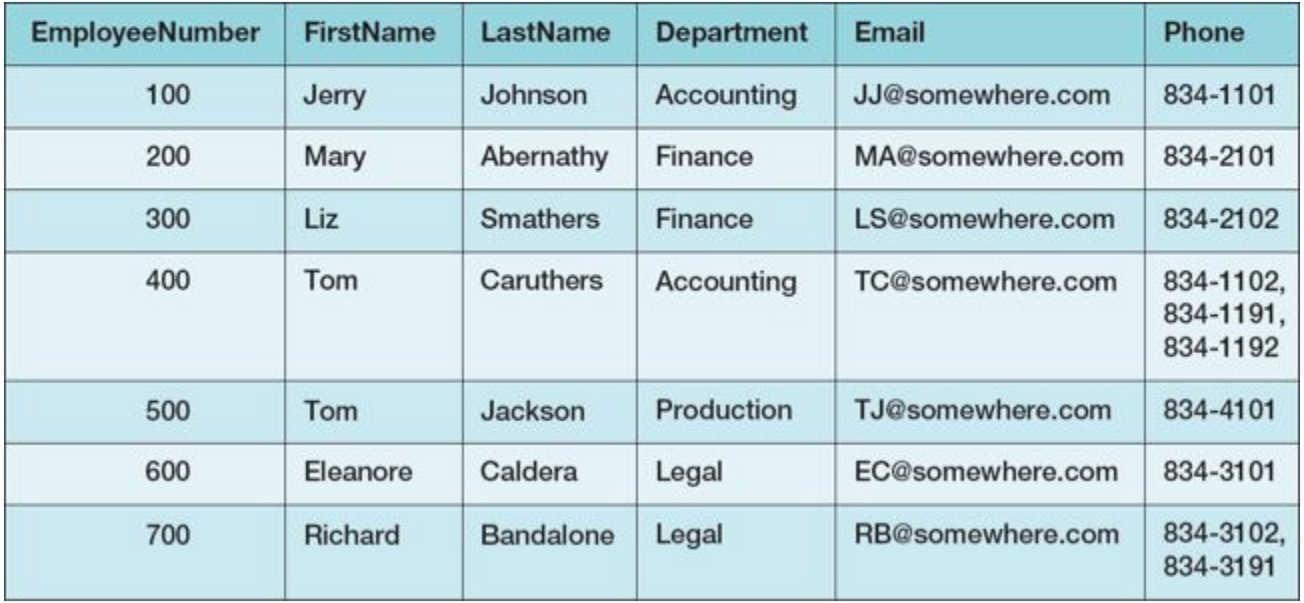
Each column name is unique

One fact per cell (row and column)

Don't need to know row or column order to interpret the data

Invalid Relation w/ Overloaded Cells

This has multiple entries per cell in the Phone column.

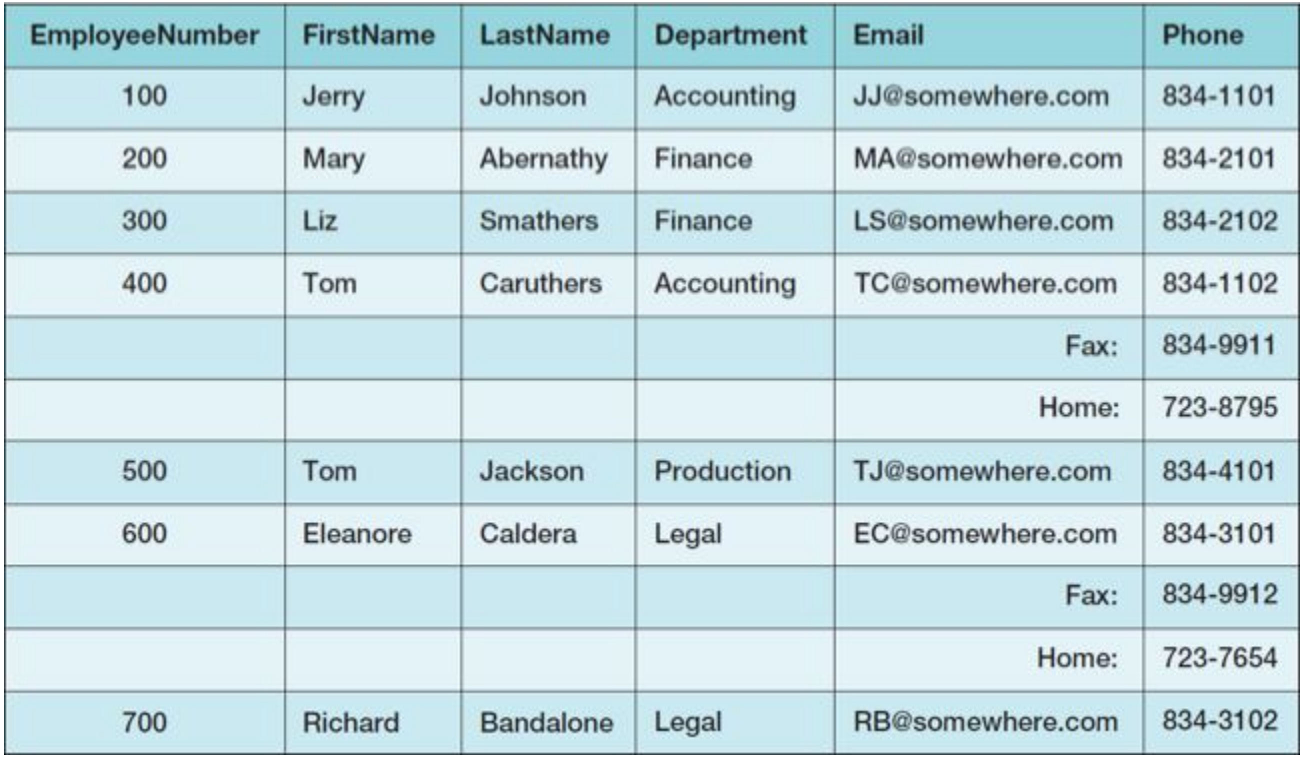


EmployeeNumber	FirstName	LastName	Department	Email	Phone
100	Jerry	Johnson	Accounting	JJ@somewhere.com	834-1101
200	Mary	Abernathy	Finance	MA@somewhere.com	834-2101
300	Liz	Smathers	Finance	LS@somewhere.com	834-2102
400	Tom	Caruthers	Accounting	TC@somewhere.com	834-1102, 834-1191, 834-1192
500	Tom	Jackson	Production	TJ@somewhere.com	834-4101
600	Eleanore	Caldera	Legal	EC@somewhere.com	834-3101
700	Richard	Bandalone	Legal	RB@somewhere.com	834-3102, 834-3191

Invalid Relation w/ Req Row Order

Fixes the overloading problem, but ...

What happens if we shuffle the rows?



EmployeeNumber	FirstName	LastName	Department	Email	Phone
100	Jerry	Johnson	Accounting	JJ@somewhere.com	834-1101
200	Mary	Abernathy	Finance	MA@somewhere.com	834-2101
300	Liz	Smathers	Finance	LS@somewhere.com	834-2102
400	Tom	Caruthers	Accounting	TC@somewhere.com	834-1102
				Fax:	834-9911
				Home:	723-8795
500	Tom	Jackson	Production	TJ@somewhere.com	834-4101
600	Eleanore	Caldera	Legal	EC@somewhere.com	834-3101
				Fax:	834-9912
				Home:	723-7654
700	Richard	Bandalone	Legal	RB@somewhere.com	834-3102

Domain Integrity Constraint

You learned this in 8-th grade math too. A relation is a ***mapping*** from a ***domain*** to a ***range***.

- The requirement that all values in a column are of the same kind is the **domain integrity constraint**.
- The domain is the set of all possible values a column can store:
  - **FirstName** could have a domain of names such as Albert, Bruce, Cathy, David, Edith, and so forth.
  - All values of FirstName must come from the names in that domain.
- Columns in different relations may have the same name or even the same domain.

Alternative Terminology

Although not all tables are relations, **the terms *table* and *relation* are normally used interchangeably**.

The following sets of terms are equivalent:

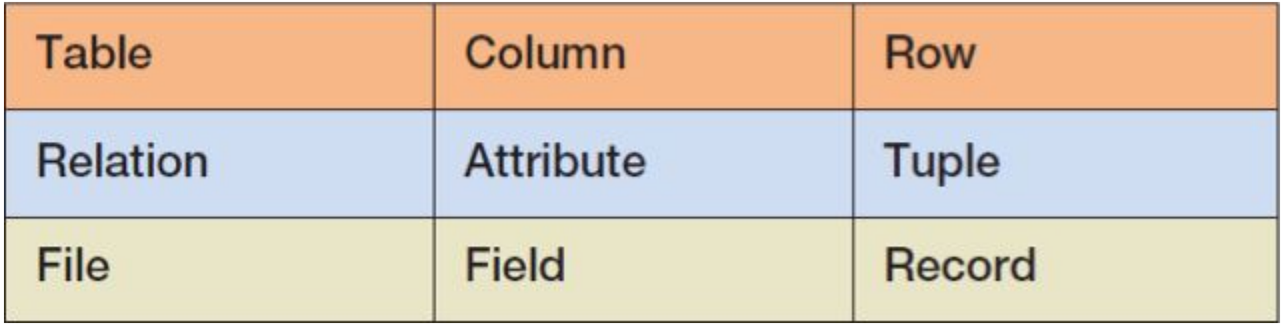


Table	Column	Row
Relation	Attribute	Tuple
File	Field	Record

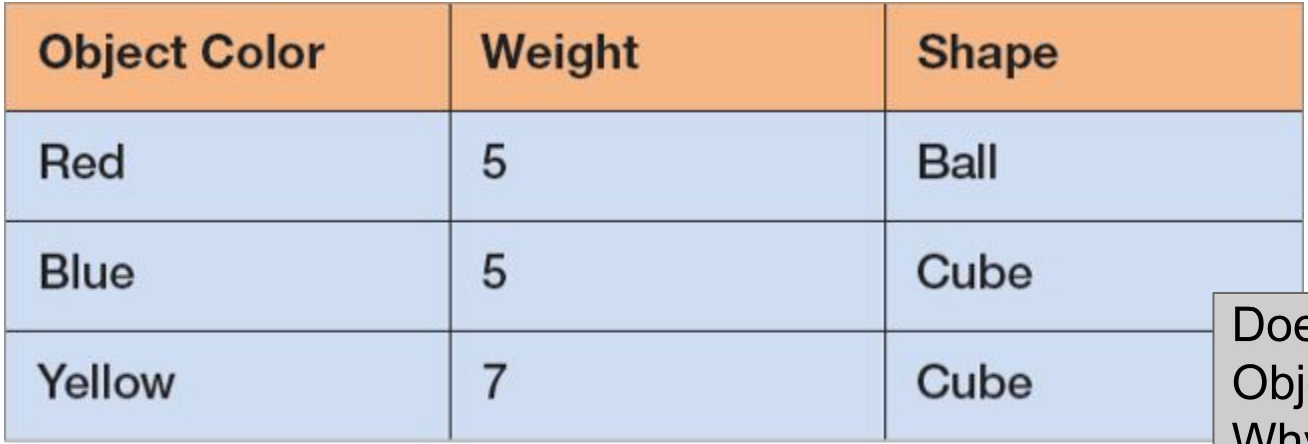
Primary Keys

- In a relation as originally defined by Codd:
  - The rows of a relation must be **unique**
  - These is **no requirement** for a **designated primary key**
- However …
  - The requirement for unique rows implies that a primary key can be designated.
  - In the “real world”, every relation has a primary key.
- When do we designate a primary key?

Functional Dependencies

- A **functional dependency** occurs when the value of one (set of) attribute(s) determines the value of a second (set of) attribute(s):
  - **StudentID → StudentName**
  - **StudentID → (DormName, DormRoom, Fee)**
- The attribute on the **left side** is called the **determinant**.
- Functional dependencies may be *based* on equations:
  - **ExtendedPrice = Quantity X UnitPrice**
  - **(Quantity, UnitPrice) → ExtendedPrice**
- Function dependencies are *not* equations!

Functional Dependency Example



Object Color	Weight	Shape
Red	5	Ball
Blue	5	Cube
Yellow	7	Cube

Does Shape → ObjectColor? Why not?

ObjectColor → Weight

ObjectColor → Shape

ObjectColor → (Weight, Shape)

Composite Determinant

Composite determinant = a determinant of a functional dependency that consists of more than one attribute

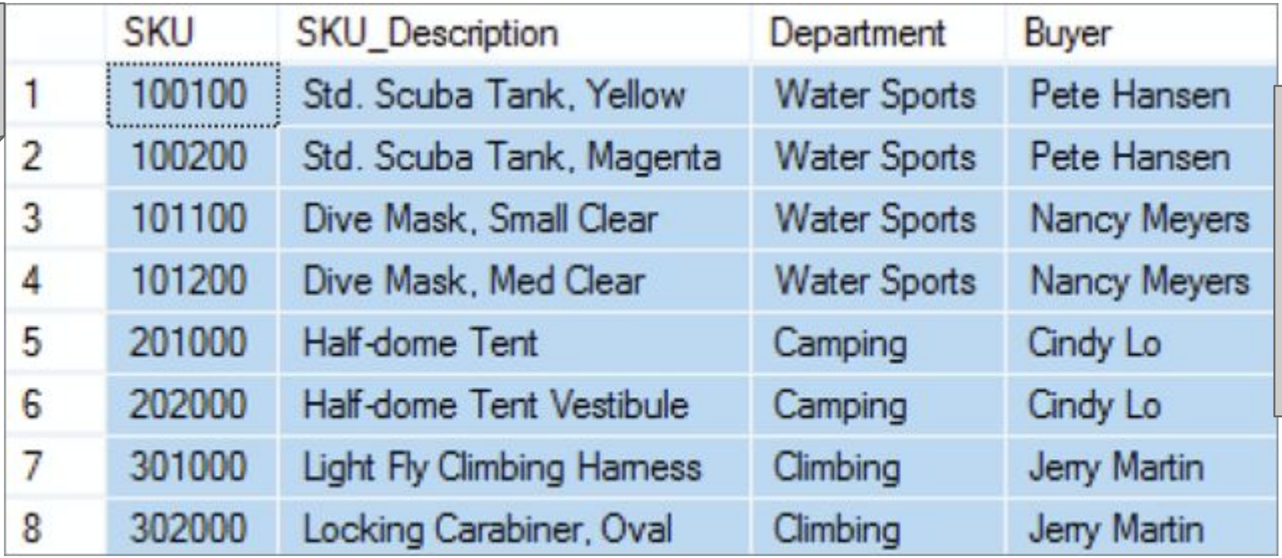
(StudentName, ClassName) → (Grade)

Some Useful Rules

- If A → (B, C), then A → B and A → C.
  - This is the **decomposition rule**.
- If A → B and A → C, then A → (B, C).
  - This is the **union rule**.
- However, if (A,B) → C, then *neither A nor B determines C by itself*.

Identify the Potential Dependencies

The SKU_DATA table



	SKU	SKU_Description	Department	Buyer
1	100100	Std. Scuba Tank, Yellow	Water Sports	Pete Hansen
2	100200	Std. Scuba Tank, Magenta	Water Sports	Pete Hansen
3	101100	Dive Mask, Small Clear	Water Sports	Nancy Meyers
4	101200	Dive Mask, Med Clear	Water Sports	Nancy Meyers
5	201000	Half-dome Tent	Camping	Cindy Lo
6	202000	Half-dome Tent Vestibule	Camping	Cindy Lo
7	301000	Light Fly Climbing Harness	Climbing	Jerry Martin
8	302000	Locking Carabiner, Oval	Climbing	Jerry Martin

There is always a risk in basing dependencies from data, which might not cover all possibilities.

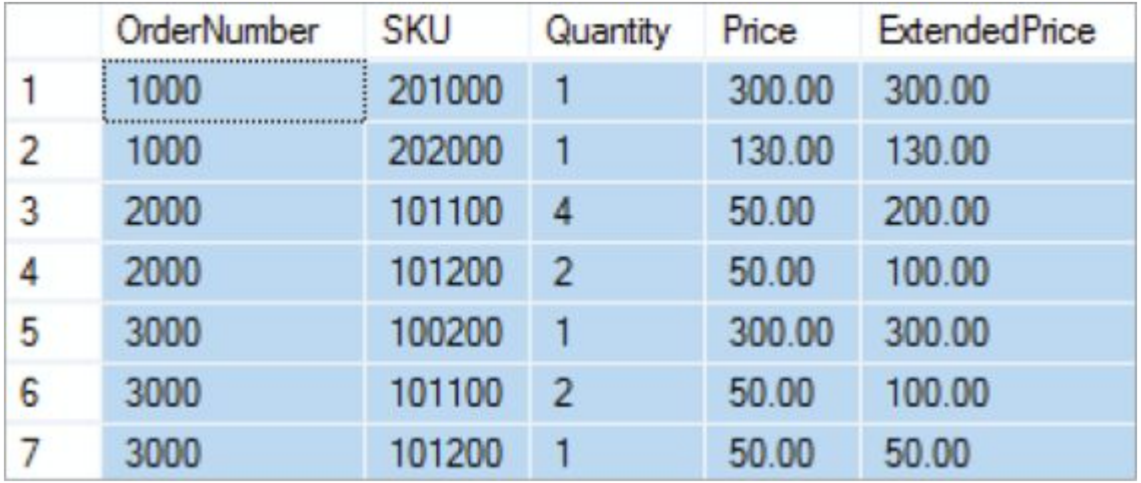
SKU → (SKU_Description, Department, Buyer)

SKU_Description → (SKU, Department, Buyer)

Buyer → Department

Again ...

The ORDER_ITEM table



	OrderNumber	SKU	Quantity	Price	ExtendedPrice
1	1000	201000	1	300.00	300.00
2	1000	202000	1	130.00	130.00
3	2000	101100	4	50.00	200.00
4	2000	101200	2	50.00	100.00
5	3000	100200	1	300.00	300.00
6	3000	101100	2	50.00	100.00
7	3000	101200	1	50.00	50.00

(OrderNumber, SKU) → (Quantity, Price, ExtendedPrice)

(Quantity, Price) → (ExtendedPrice)

PK Determinants Must be *Unique*

- A determinant is **unique** in a relation if and only if, it determines every other column in the relation.
- You cannot find the determinants of all functional dependencies simply by looking for unique values in one column:
    - Data set limitations (not enough data to *discover* ambiguities)
    - Must be logically a determinant (have a process that *enforces* uniqueness)

Primary and Candidate Keys

- A **candidate key** is a combination of one or more columns that is used to identify rows in a relation.
  - The PK is one of the candidate keys
- A key can be **composite**, consisting of two or more columns.

Primary Key Considerations

A primary key is a candidate key selected as the primary means of identifying rows in a relation.

- There is only one primary key per relation.
- The primary key may be a composite key.
- The ideal primary key is short, numeric, and never changes.

Entity Integrity Constraint

- In order to function properly, the primary key **must have unique data values for every row in the table**
- The phrase **unique data values** implies that this column is NOT NULL, and does not allow a NULL value in any row.

Surrogate Keys

A **surrogate key** is an **artificial** column added to a relation to serve as a primary key.

- DBMS supplied
- Short, numeric, and never changes (an ideal primary key)
- Has artificial values that are meaningless to users
- Normally hidden in forms and reports

Need for Surrogate Keys

Without surrogate key:

RENTAL_PROPERTY (Street, City, State, Zip, RentalRate)

With surrogate key:

RENTAL_PROPERTY (PropertyID, Street, City, State, Zip, RentalRate)

Foreign Keys

A **foreign key** is the primary key of one relation that is placed in another relation to form a link between the relations.

- Can be a single column or a composite key.
- The term refers to the fact that key values are foreign to the relation in which they appear as foreign key values.
- Technically, the foreign key shares the domain of the primary key it references.

Referential Integrity Constraint

A **referential integrity** constraint is a statement that limits the values of the foreign key to those already existing as primary key values in the corresponding relation.

For example ...

Each SKU in ORDER_ITEM **must exist** in SKU in SKU_DATA

Database Integrity

A database has integrity if it satisfies ...

- The **Domain Integrity** Constraint
  - so that we always know how to interpret the facts
- The **Entity Integrity** Constraint
  - so that we know exactly which entity a fact applies to
- The **Referential Integrity** Constraint
  - so that we never refer to a thing that does not exist

Goal: Useful, meaningful, coherent data

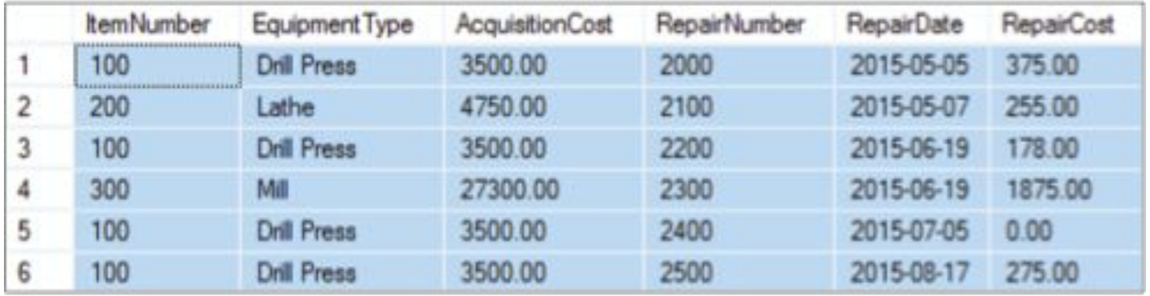
Normalization

How to ensure Database Integrity and avoid Data Anomalies

Data Anomaly Example

Assume that EquipmentType → AcquistionCost

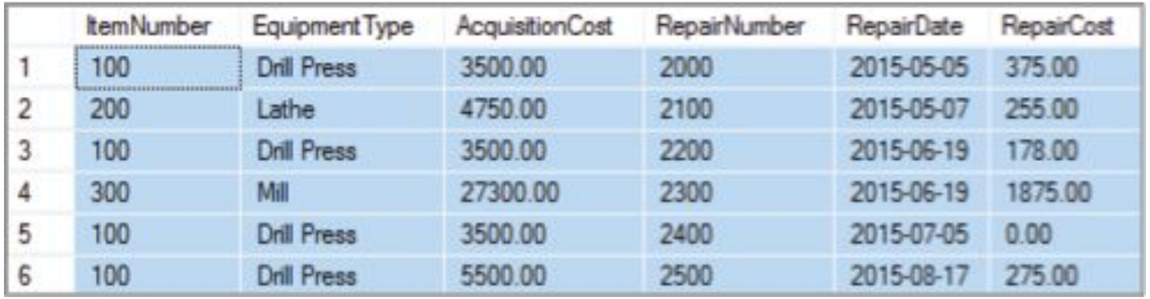
Before data edit



	ItemNumber	EquipmentType	AcquisitionCost	RepairNumber	RepairDate	RepairCost
1	100	Drill Press	3500.00	2000	2015-05-05	375.00
2	200	Lathe	4750.00	2100	2015-05-07	255.00
3	100	Drill Press	3500.00	2200	2015-06-19	178.00
4	300	Mill	27300.00	2300	2015-06-19	1875.00
5	100	Drill Press	3500.00	2400	2015-07-05	0.00
6	100	Drill Press	3500.00	2500	2015-08-17	275.00

Can you spot the anomaly?

After data edit



	ItemNumber	EquipmentType	AcquisitionCost	RepairNumber	RepairDate	RepairCost
1	100	Drill Press	3500.00	2000	2015-05-05	375.00
2	200	Lathe	4750.00	2100	2015-05-07	255.00
3	100	Drill Press	3500.00	2200	2015-06-19	178.00
4	300	Mill	27300.00	2300	2015-06-19	1875.00
5	100	Drill Press	3500.00	2400	2015-07-05	0.00
6	100	Drill Press	5500.00	2500	2015-08-17	275.00

How Anomalies Happen

Deletion Anomaly

- Deleting one or more records causes a violation of the referential integrity constraint

Insertion/Update Anomaly

- Inserting or modifying a record causes a functional dependency violation (inconsistency)

Table Normalization

Normalization is the design of tables to eliminate potential anomalies.

- Fun fact: the name is inspired "normalization" of relations between the US and China in the 1970s

Normalization is on a spectrum, with some designs having higher levels of normalization than others.

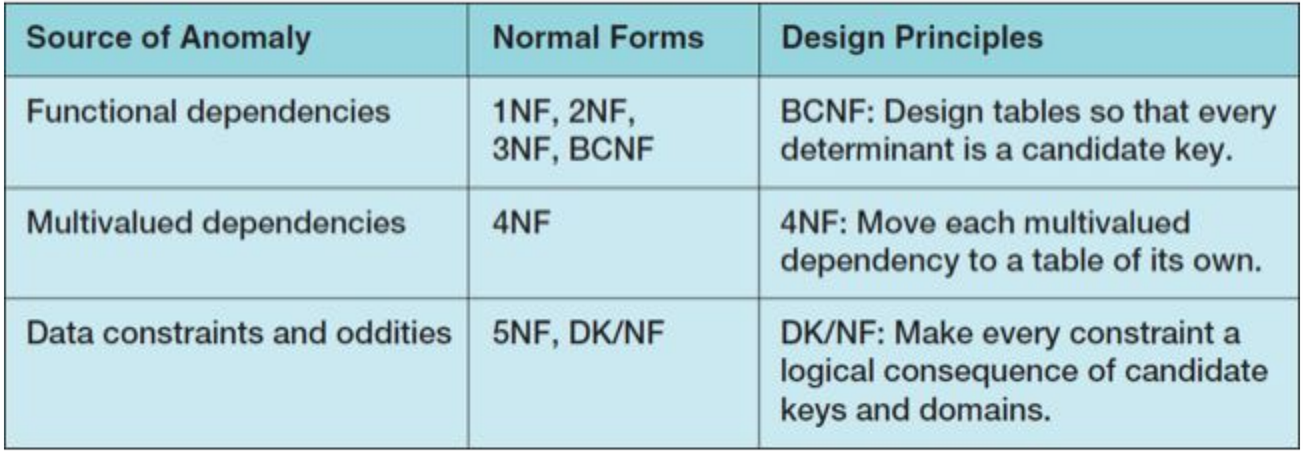
- How normalized you want the database to be is actually a common design decision.

Normalization Goals

- Each table represents a single subject
- No data item will be unnecessarily stored in more than one table
- All nonprime (not PK) attributes in a table are dependent on the PK
- Each table is void of insertion, update, and deletion anomalies

Normal Forms and Anomalies

A normal form is a set of rules that eliminate a kind of anomaly. We will study of the the more common ones.



Source of Anomaly	Normal Forms	Design Principles
Functional dependencies	1NF, 2NF, 3NF, BCNF	BCNF: Design tables so that every determinant is a candidate key.
Multivalued dependencies	4NF	4NF: Move each multivalued dependency to a table of its own.
Data constraints and oddities	5NF, DK/NF	DK/NF: Make every constraint a logical consequence of candidate keys and domains.

The "Usual" Normal Forms

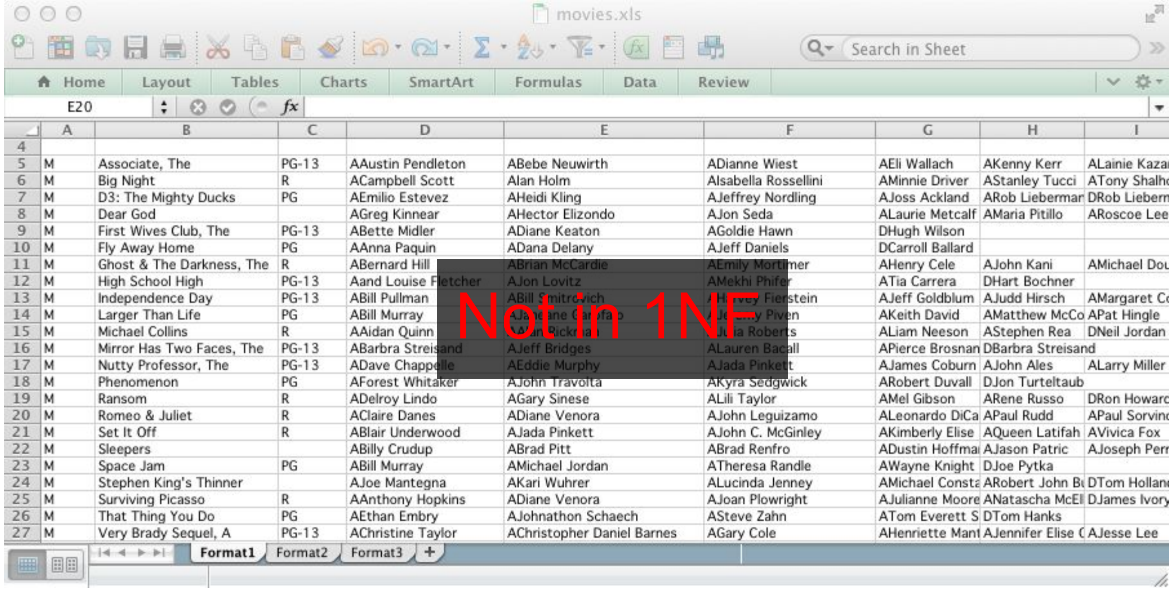
- **1NF** —a table that qualifies as a relation is in 1NF.
- **2NF** —a relation is in 2NF if all of its nonkey attributes are dependent on *all* of the primary keys.
- **3NF**—a relation is in 3NF if it is in 2NF and has no determinants except the primary key.
- **Boyce-Codd Normal Form (BCNF)**—a relation is in BCNF if every determinant is a candidate key.

The “Right” Level of Normalization

- 2NF is better than 1NF; 3NF is better than 2NF
- For most business database design purposes, 3NF is as much as we need
- Highest level of normalization is not always most desirable
    - ***Denormalization*** can sometimes improve performance, though with more redundancy

First Normal Form (1NF)

PK identified, each attribute contains a single value, and there are no repeating groups.

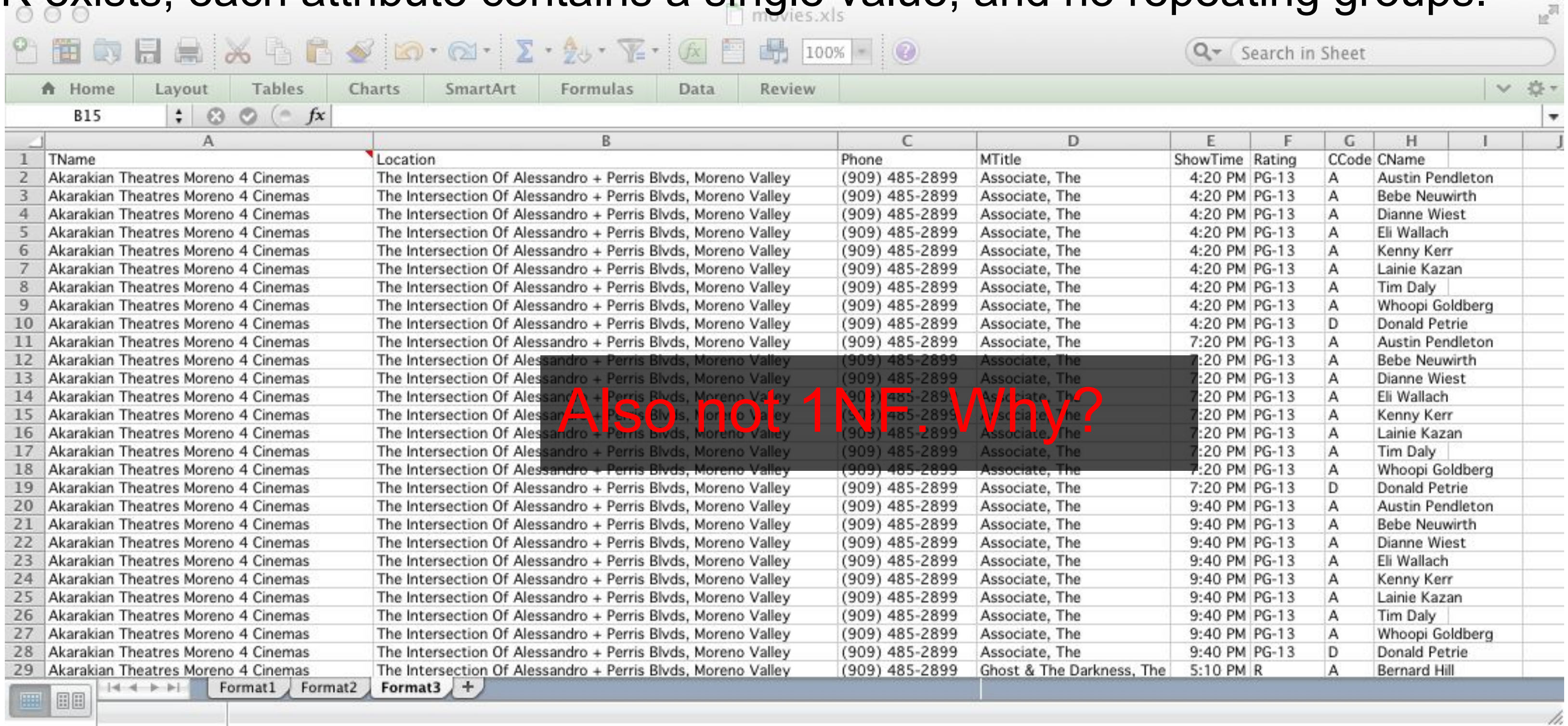


Not in 1NF

	A	B	C	D	E	F	G	H	I
4									
5	M	Associate, The	PG-13	AAustin Pendleton	ABebe Neuwirth	ADianne Wiest	AEli Wallach	AKenny Kerr	ALainie Kazan
6	M	Big Night	R	ACampbell Scott	Alan Holm	AIsabella Rossellini	AMinnie Driver	AStanley Tucci	ATony Shalho
7	M	D3: The Mighty Ducks	PG	AEmilio Estevez	AHeidi Kling	AJeffrey Nordling	AJoss Ackland	ARob Liebermar	DRob Liebern
8	M	Dear God		AGreg Kinnear	AHector Elizondo	AJon Seda	ALaurie Metcalf	AMaria Pitillo	ARoscoe Lee
9	M	First Wives Club, The	PG-13	ABette Midler	ADiane Keaton	AGoldie Hawn	DHugh Wilson		
10	M	Fly Away Home	PG	AAnna Paquin	ADana Delany	AJeff Daniels	DCarroll Ballard		
11	M	Ghost & The Darkness, The	R	ABernard Hill	ABrian McCardie	AEmily Mortimer	AHenry Cele	AJohn Kani	AMichael Dou
12	M	High School High	PG-13	Aand Louise Fletcher	AJon Lovitz	AMekhi Phifer	ATia Carrera	DHart Bochner	
13	M	Independence Day	PG-13	ABill Pullman	ABill Smitrovich	AHarvey Fierstein	AJeff Goldblum	AJudd Hirsch	AMargaret Co
14	M	Larger Than Life	PG	ABill Murray	AJaneane Garofalo	AJeremy Piven	AKeith David	AMatthew McCo	APat Hingle
15	M	Michael Collins	R	AAidan Quinn	AAlan Rickman	AJulia Roberts	ALiam Neeson	AStephen Rea	DNeil Jordan
16	M	Mirror Has Two Faces, The	PG-13	ABarbra Streisand	AJeff Bridges	ALauren Bacall	APierce Brosnan	DBarbra Streisand	
17	M	Nutty Professor, The	PG-13	ADave Chappelle	AEddie Murphy	AJada Pinkett	AJames Coburn	AJohn Ales	ALarry Miller
18	M	Phenomenon	PG	AForest Whitaker	AJohn Travolta	AKyra Sedgwick	ARobert Duvall	DJon Turteltaub	
19	M	Ransom	R	ADelroy Lindo	AGary Sinese	ALili Taylor	AMel Gibson	ARene Russo	DRon Howard
20	M	Romeo & Juliet	R	AClaire Danes	ADiane Venora	AJohn Leguizamo	ALeonardo DiCa	APaul Rudd	APaul Sorvino
21	M	Set It Off	R	ABlair Underwood	AJada Pinkett	AJohn C. McGinley	AKimberly Elise	AQueen Latifah	AVivica Fox
22	M	Sleepers		ABilly Crudup	ABrad Pitt	ABrad Renfro	ADustin Hoffma	AJason Patric	AJoseph Perr
23	M	Space Jam	PG	ABill Murray	AMichael Jordan	ATheresa Randle	AWayne Knight	DJoe Pytka	
24	M	Stephen King's Thinner		AJoe Mantegna	AKari Wuhrer	ALucinda Jenney	AMichael Consta	ARobert John Bu	DTom Holland
25	M	Surviving Picasso	R	AAnthony Hopkins	ADiane Venora	AJoan Plowright	AJulianne Moore	ANatascha McEl	DJames Ivory
26	M	That Thing You Do	PG	AEthan Embry	AJohnathon Schaech	ASteve Zahn	ATom Everett S	DTom Hanks	
27	M	Very Brady Sequel, A	PG-13	AChristine Taylor	AChristopher Daniel Barnes	AGary Cole	AHenriette Mant	AJennifer Elise (	AJesse Lee

1NF (Again)

PK exists, each attribute contains a single value, and no repeating groups.



	A	B	C	D	E	F	G	H
1	TName	Location	Phone	MTitle	ShowTime	Rating	CCode	CName
2	Akarakian Theatres Moreno 4 Cinemas	The Intersection Of Alessandro + Perris Blvds, Moreno Valley	(909) 485-2899	Associate, The	4:20 PM	PG-13	A	Austin Pendleton
3	Akarakian Theatres Moreno 4 Cinemas	The Intersection Of Alessandro + Perris Blvds, Moreno Valley	(909) 485-2899	Associate, The	4:20 PM	PG-13	A	Bebe Neuwirth
4	Akarakian Theatres Moreno 4 Cinemas	The Intersection Of Alessandro + Perris Blvds, Moreno Valley	(909) 485-2899	Associate, The	4:20 PM	PG-13	A	Dianne Wiest
5	Akarakian Theatres Moreno 4 Cinemas	The Intersection Of Alessandro + Perris Blvds, Moreno Valley	(909) 485-2899	Associate, The	4:20 PM	PG-13	A	Eli Wallach
6	Akarakian Theatres Moreno 4 Cinemas	The Intersection Of Alessandro + Perris Blvds, Moreno Valley	(909) 485-2899	Associate, The	4:20 PM	PG-13	A	Kenny Kerr
7	Akarakian Theatres Moreno 4 Cinemas	The Intersection Of Alessandro + Perris Blvds, Moreno Valley	(909) 485-2899	Associate, The	4:20 PM	PG-13	A	Lainie Kazan
8	Akarakian Theatres Moreno 4 Cinemas	The Intersection Of Alessandro + Perris Blvds, Moreno Valley	(909) 485-2899	Associate, The	4:20 PM	PG-13	A	Tim Daly
9	Akarakian Theatres Moreno 4 Cinemas	The Intersection Of Alessandro + Perris Blvds, Moreno Valley	(909) 485-2899	Associate, The	4:20 PM	PG-13	A	Whoopi Goldberg
10	Akarakian Theatres Moreno 4 Cinemas	The Intersection Of Alessandro + Perris Blvds, Moreno Valley	(909) 485-2899	Associate, The	4:20 PM	PG-13	D	Donald Petrie
11	Akarakian Theatres Moreno 4 Cinemas	The Intersection Of Alessandro + Perris Blvds, Moreno Valley	(909) 485-2899	Associate, The	7:20 PM	PG-13	A	Austin Pendleton
12	Akarakian Theatres Moreno 4 Cinemas	The Intersection Of Alessandro + Perris Blvds, Moreno Valley	(909) 485-2899	Associate, The	7:20 PM	PG-13	A	Bebe Neuwirth
13	Akarakian Theatres Moreno 4 Cinemas	The Intersection Of Alessandro + Perris Blvds, Moreno Valley	(909) 485-2899	Associate, The	7:20 PM	PG-13	A	Dianne Wiest
14	Akarakian Theatres Moreno 4 Cinemas	The Intersection Of Alessandro + Perris Blvds, Moreno Valley	(909) 485-2899	Associate, The	7:20 PM	PG-13	A	Eli Wallach
15	Akarakian Theatres Moreno 4 Cinemas	The Intersection Of Alessandro + Perris Blvds, Moreno Valley	(909) 485-2899	Associate, The	7:20 PM	PG-13	A	Kenny Kerr
16	Akarakian Theatres Moreno 4 Cinemas	The Intersection Of Alessandro + Perris Blvds, Moreno Valley	(909) 485-2899	Associate, The	7:20 PM	PG-13	A	Lainie Kazan
17	Akarakian Theatres Moreno 4 Cinemas	The Intersection Of Alessandro + Perris Blvds, Moreno Valley	(909) 485-2899	Associate, The	7:20 PM	PG-13	A	Tim Daly
18	Akarakian Theatres Moreno 4 Cinemas	The Intersection Of Alessandro + Perris Blvds, Moreno Valley	(909) 485-2899	Associate, The	7:20 PM	PG-13	A	Whoopi Goldberg
19	Akarakian Theatres Moreno 4 Cinemas	The Intersection Of Alessandro + Perris Blvds, Moreno Valley	(909) 485-2899	Associate, The	7:20 PM	PG-13	D	Donald Petrie
20	Akarakian Theatres Moreno 4 Cinemas	The Intersection Of Alessandro + Perris Blvds, Moreno Valley	(909) 485-2899	Associate, The	9:40 PM	PG-13	A	Austin Pendleton
21	Akarakian Theatres Moreno 4 Cinemas	The Intersection Of Alessandro + Perris Blvds, Moreno Valley	(909) 485-2899	Associate, The	9:40 PM	PG-13	A	Bebe Neuwirth
22	Akarakian Theatres Moreno 4 Cinemas	The Intersection Of Alessandro + Perris Blvds, Moreno Valley	(909) 485-2899	Associate, The	9:40 PM	PG-13	A	Dianne Wiest
23	Akarakian Theatres Moreno 4 Cinemas	The Intersection Of Alessandro + Perris Blvds, Moreno Valley	(909) 485-2899	Associate, The	9:40 PM	PG-13	A	Eli Wallach
24	Akarakian Theatres Moreno 4 Cinemas	The Intersection Of Alessandro + Perris Blvds, Moreno Valley	(909) 485-2899	Associate, The	9:40 PM	PG-13	A	Kenny Kerr
25	Akarakian Theatres Moreno 4 Cinemas	The Intersection Of Alessandro + Perris Blvds, Moreno Valley	(909) 485-2899	Associate, The	9:40 PM	PG-13	A	Lainie Kazan
26	Akarakian Theatres Moreno 4 Cinemas	The Intersection Of Alessandro + Perris Blvds, Moreno Valley	(909) 485-2899	Associate, The	9:40 PM	PG-13	A	Tim Daly
27	Akarakian Theatres Moreno 4 Cinemas	The Intersection Of Alessandro + Perris Blvds, Moreno Valley	(909) 485-2899	Associate, The	9:40 PM	PG-13	A	Whoopi Goldberg
28	Akarakian Theatres Moreno 4 Cinemas	The Intersection Of Alessandro + Perris Blvds, Moreno Valley	(909) 485-2899	Associate, The	9:40 PM	PG-13	D	Donald Petrie
29	Akarakian Theatres Moreno 4 Cinemas	The Intersection Of Alessandro + Perris Blvds, Moreno Valley	(909) 485-2899	Ghost & The Darkness, The	5:10 PM	R	A	Bernard Hill

Also not 1NF. Why?

Converting to 1NF

- Step 1: Split out Repeating Groups
  - Also eliminate nulls: each repeating group attribute contains an appropriate data value
- Step 2: Identify the Primary Key
  - Must uniquely identify attribute value
  - New key must be composed
- Step 3: Identify All Dependencies
  - Dependencies are depicted with a diagram

Second Normal Form (2NF)

2NF = 1NF and No Partial Dependencies

- 1NF means we can assume
    - All keys are defined
    - No repeating groups
    - All attributes dependent on the PK
- Must eliminate partial dependencies
    - Look for attributes that are dependent on only part of the PK
    - Only applies to composite keys

Converting to 2NF

- **Step 1: Make New Tables to Eliminate Partial Dependencies**
  - Write each key component on separate line, then write original (composite) key on last line
  - Each component will become key in new table
- **Step 2: Reassign Corresponding Dependent Attributes**
  - Determine attributes that are dependent on other attributes
  - At this point, most anomalies have been eliminated

Third Normal Form (3NF)

3NF = 2NF and No Transitive Dependencies

- 2NF allows us to assume …
  - 1NF table structure
  - Each attribute is dependent on the entire PK
- Must eliminate transitive dependencies
  - If A → B and B → C then B is a candidate key for C
  - Split the B → C relationship into a new table

Conversion to 3NF

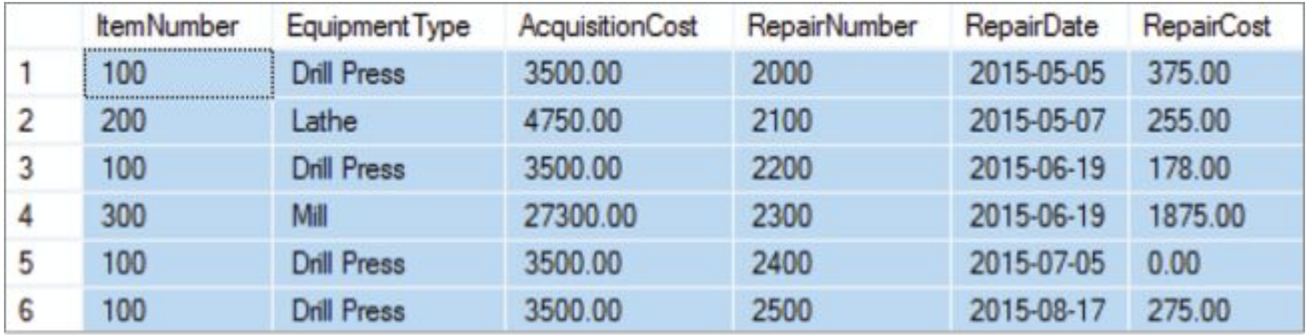
- **Step 1: Make New Tables to Eliminate Transitive Dependencies**
  - For every transitive dependency, write its determinant as PK for new table
  - Determinant: any attribute whose value determines other values within a row
- **Step 2: Reassign non-PK determinants and their dependants to the new tables**

Boyce-Codd Normal Form (BCNF)

BCNF = 3NF and Every Determinant is a Candidate Key

- Most of the time 3NF implies BCNF
- One can go straight from 1NF to BCNF by identifying every functional dependency and then breaking into one table per candidate key

BCNF Example



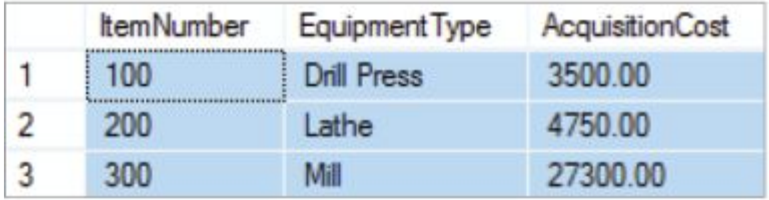
	ItemNumber	EquipmentType	AcquisitionCost	RepairNumber	RepairDate	RepairCost
1	100	Drill Press	3500.00	2000	2015-05-05	375.00
2	200	Lathe	4750.00	2100	2015-05-07	255.00
3	100	Drill Press	3500.00	2200	2015-06-19	178.00
4	300	Mill	27300.00	2300	2015-06-19	1875.00
5	100	Drill Press	3500.00	2400	2015-07-05	0.00
6	100	Drill Press	3500.00	2500	2015-08-17	275.00

ItemNumber → (Type, AcquisitionCost)

RepairNumber → (ItemNumber, Type, AcquisitionCost, RepairDate, RepairAmount)

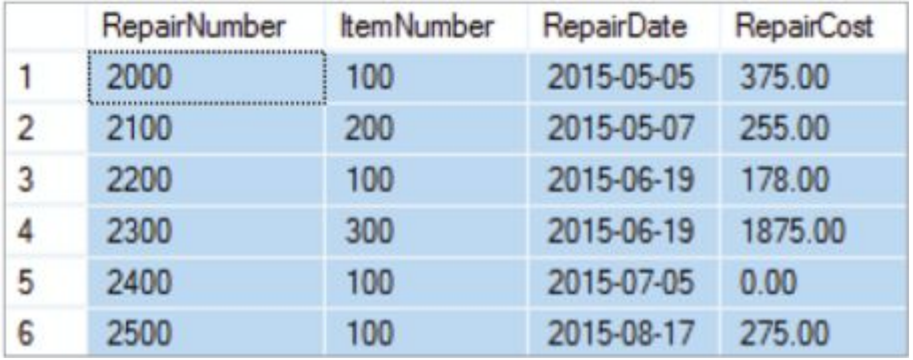
Two Derived BCNF Relations

EQUIPMENT_ITEM



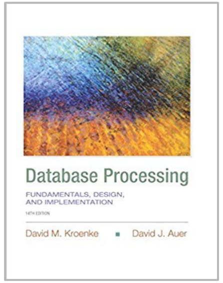
	ItemNumber	EquipmentType	AcquisitionCost
1	100	Drill Press	3500.00
2	200	Lathe	4750.00
3	300	Mill	27300.00

REPAIR



	RepairNumber	ItemNumber	RepairDate	RepairCost
1	2000	100	2015-05-05	375.00
2	2100	200	2015-05-07	255.00
3	2200	100	2015-06-19	178.00
4	2300	300	2015-06-19	1875.00
5	2400	100	2015-07-05	0.00
6	2500	100	2015-08-17	275.00

Fairfield University
DOLAN
School of Business



Database Processing

FUNDAMENTALS, DESIGN,
AND IMPLEMENTATION

14TH EDITION

David M. Kroenke ▪ David J. Auer

Databases for Analytics

Kroenke / Auer
Chapter 3