Database Theory and Applications for Biomedical Research and Practice

BMIN 502 / EPID 635
Week 2: Data normalization

John H. Holmes, PhD

Institute for Biomedical Informatics

Objectives for today

- You will learn:
  - Basic normalization theory
  - When and why normalization is desirable
  - What happens when data are not normalized
  - How to normalize a database

The Schema: Another approach to logical data modeling

- Describes the contents of a table
  - Name of table
  - List of fields (entity attributes)
  - Key fields
- Each schema ultimately becomes a table in the database
- Syntax
  - Table(FIELD LIST)
    - Primary keys are underlined and listed first
    - Foreign keys can be underlined with a dashed line

So what? Haven't we already created a logical model in the E-R?

- Yes, but the schema provides a way to double check the robustness of the model
  - An ERD models entities and relationships between them
  - A schema models entity attributes and relationships between *them*
- The schema provides a way to normalize the relations and the entire database

Normalization

- A process in which an "unsatisfactory" relational schema is decomposed into smaller, more " desirable" schemata
- Performed to eliminate redundancy and update anomalies in database tables
- Levels of normalization are hierarchical
  - 0NF
  - 1NF
  - 2NF
  - 3NF

First Normal Form

A table is in 1NF only if no composite attributes and no repeating groups exist in the table

What is a composite attribute?

- An attribute that represents more than one concept
- Examples
  - Study IDs such as “001-01” where “001” is the subject’s study serial number and “01” is the study site
  - Addresses such as “123 Main St., Philadelphia, PA”
- Why do we care?
  - You can’t get at the component concepts without parsing
- Solution
  - Make all attributes atomic: one concept and only one concept

What is a repeating group?

- One or more concepts represented many times in the same table
- Two examples
  - HB1 HB2 HB3
    - HB represents a single concept, but is repeated
  - HB1 HCT1 HB2 HCT2 HB3 HCT3
    - HB+HCT represent two concepts, are repeated together

What if a database is not normalized?
Case 1: Repeating Groups

- Subject(SUBJ_ID DOB SEX DX1 DX2 DX3)
- DX1…DX3 represent a *repeating group*
  - What happens if you need DX4?
  - If you want to look for all MI patients, you need to go through three *separate* fields

How to get a table into First Normal Form

Subject(SUBJ_ID DOB SEX DX1 DX2 DX3)

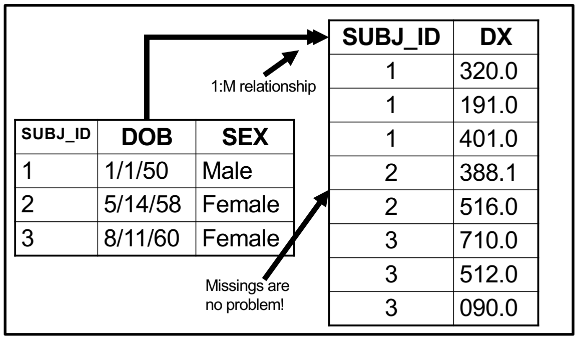
- **Is not in 1NF**
- (DX1, DX2, and DX3 repeat the same concept, DX)
- Create a new table for each non-similar repeating group, adding the primary key to the new table(s):

Subject(SUBJ_ID DOB SEX)
Diagnosis(SUBJ_ID DX)

Thus…

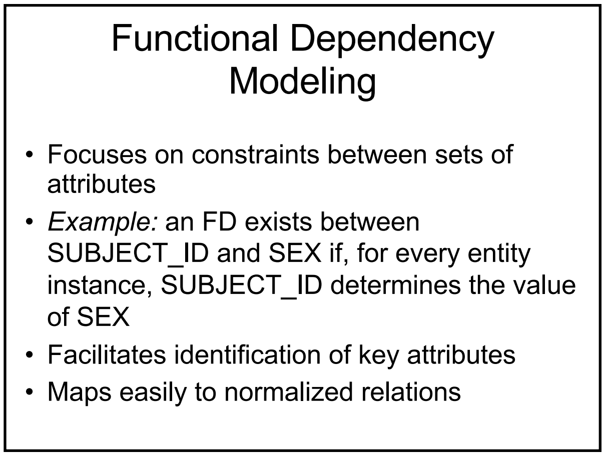
SUBJ_ID	DOB	SEX	DX1	DX2	DX3
1	1/1/50	Male	320.0	191.0	401.0
2	5/14/58	Female	388.1	XXXX	516.0
3	8/11/60	Female	710.0	512.0	090.0

This is better…



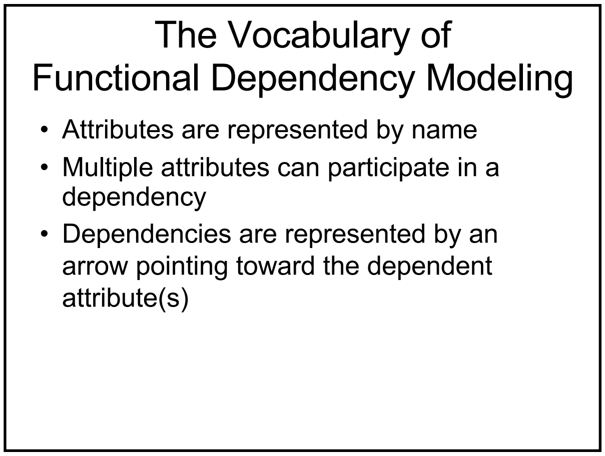
SUBJ_ID	DOB	SEX
1	1/1/50	Male
2	5/14/58	Female
3	8/11/60	Female

SUBJ_ID	DX
1	320.0
1	191.0
1	401.0
2	388.1
2	516.0
3	710.0
3	512.0
3	090.0



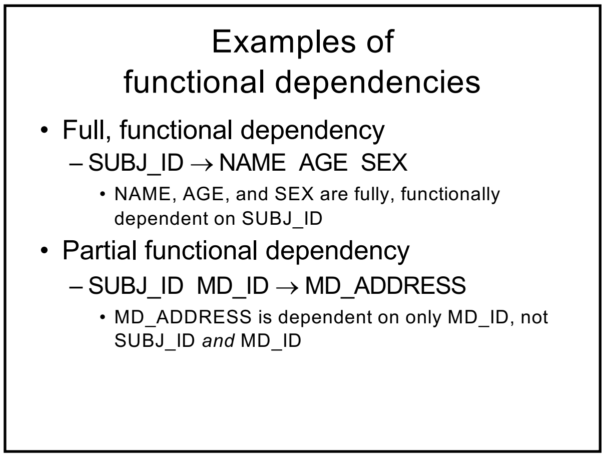
Functional Dependency Modeling

- Focuses on constraints between sets of attributes
- *Example:* an FD exists between SUBJECT_ID and SEX if, for every entity instance, SUBJECT_ID determines the value of SEX
- Facilitates identification of key attributes
- Maps easily to normalized relations



The Vocabulary of Functional Dependency Modeling

- Attributes are represented by name
- Multiple attributes can participate in a dependency
- Dependencies are represented by an arrow pointing toward the dependent attribute(s)



Examples of functional dependencies

- Full, functional dependency
  - SUBJ_ID → NAME AGE SEX
    - NAME, AGE, and SEX are fully, functionally dependent on SUBJ_ID
- Partial functional dependency
  - SUBJ_ID MD_ID → MD_ADDRESS
    - MD_ADDRESS is dependent on only MD_ID, not SUBJ_ID *and* MD_ID

What if a database is not normalized?

Case 2: Partial Functional Dependency

- Situation:
  - Multi-site clinical trial, where subjects at each site are given a unique, site-specific ID number (ID=1-100, X5)
- Subject(SUBJ_ID SITE_ID DOB SITE_ADDRESS)
- SITE_ADDRESS is *partially dependent* on the primary key (only the SITE_ID, not SUBJ_ID *and* SITE_ID)
  - What happens if the address of the site changes?

Second Normal Form

A relation is in 2NF if it is in 1NF *and* every non-key attribute is *fully* dependent on the *entire* primary key

How to get a table into Second Normal Form

- Solution: create a new table that contains as its primary key the attributes involved in the partial dependency

Subject(SUBJ_ID SITE_ID DOB SITE_ADDRESS)

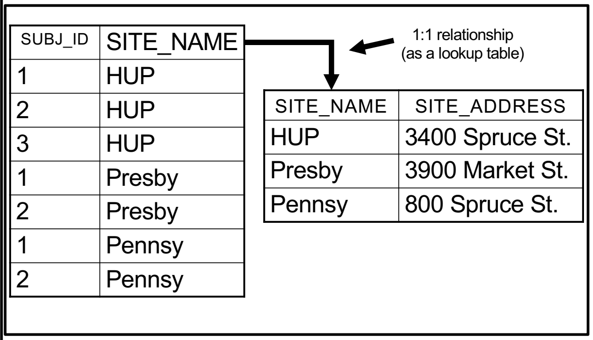
decomposes to:

Subject(SUBJ_ID SITE_ID DOB)
Site(SITE_ID SITE_ADDRESS)

Thus…

~~SUBJ_ID~~	~~SITE_ID~~	~~SITE_ADDRESS~~
~~1~~	~~HUP~~	~~3400 Spruce St.~~
~~2~~	~~HUP~~	~~3400 Spruce St.~~
~~3~~	~~HUP~~	~~3400 Spruce St.~~
~~1~~	~~Presby~~	~~3900 Market St.~~
~~2~~	~~Presby~~	~~3900 Market St.~~
~~1~~	~~Pennsy~~	~~800 Spruce St.~~
~~2~~	~~Pennsy~~	~~800 Spruce St.~~

This is better…



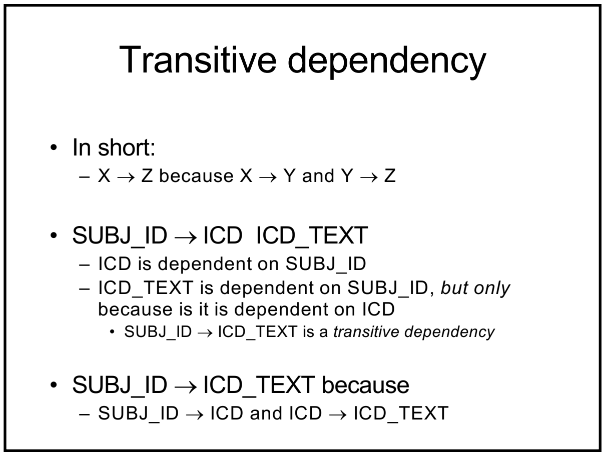
SUBJ_ID	SITE_NAME
1	HUP
2	HUP
3	HUP
1	Presby
2	Presby
1	Pennsy
2	Pennsy

1:1 relationship
(as a lookup table)

SITE_NAME	SITE_ADDRESS
HUP	3400 Spruce St.
Presby	3900 Market St.
Pennsy	800 Spruce St.

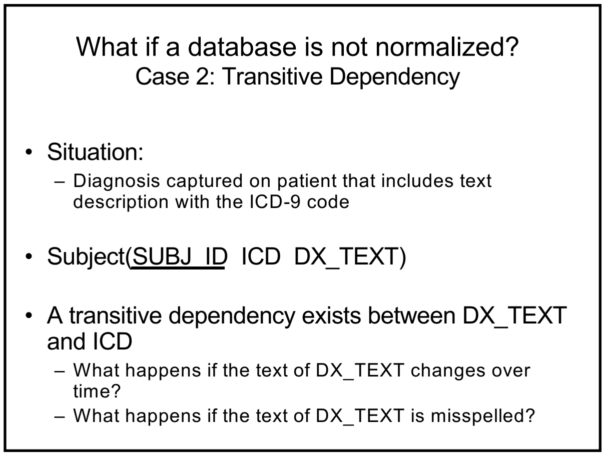
Second Normal Form: Shortcuts

- A table is in 2NF automatically, if it is in 1NF *and* the primary key contains one and only one attribute
  - No possibility of a partial dependency!
- A table is in 2NF automatically, if it is in 1NF *and* there are no non-key attributes
  - No dependency at all!



Transitive dependency

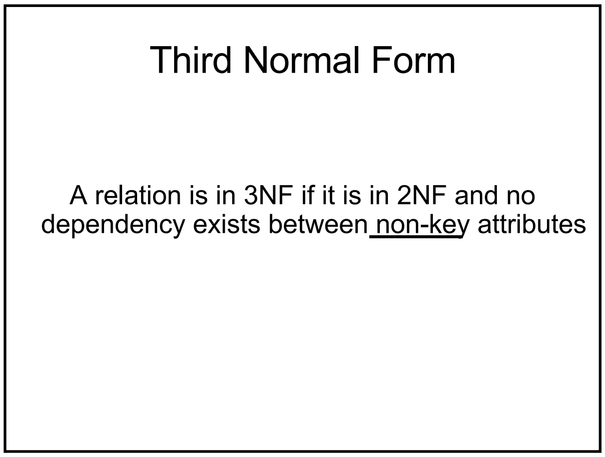
- In short:
  - X → Z because X → Y and Y → Z
- SUBJ_ID → ICD ICD_TEXT
  - ICD is dependent on SUBJ_ID
  - ICD_TEXT is dependent on SUBJ_ID, *but only* because is it is dependent on ICD
    - SUBJ_ID → ICD_TEXT is a *transitive dependency*
- SUBJ_ID → ICD_TEXT because
  - SUBJ_ID → ICD and ICD → ICD_TEXT



What if a database is not normalized?

Case 2: Transitive Dependency

- Situation:
  - Diagnosis captured on patient that includes text description with the ICD-9 code
- Subject(SUBJ_ID ICD DX_TEXT)
- A transitive dependency exists between DX_TEXT and ICD
  - What happens if the text of DX_TEXT changes over time?
  - What happens if the text of DX_TEXT is misspelled?



Third Normal Form

A relation is in 3NF if it is in 2NF and no dependency exists between non-key attributes

How to get a table into Third Normal Form

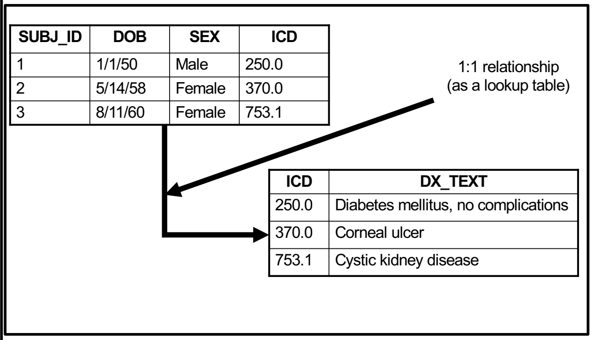
- Create a new table which contains the transitive dependency
- Identify the attribute(s) in the dependency as a primary key
- Make sure the key attribute is kept in both tables
  - The primary key in the new table will be a foreign key in the old
- Subject(SUBJ_ID DOB SEX ICD DX_TEXT) *decomposes to*:

  Subject(SUBJ_ID DOB SEX ICD)
  ICD(ICD DX_TEXT)

Thus…

SUBJ_ID	DOB	SEX	ICD	DX_TEXT
1	1/1/50	Male	250.0	Diabetes mellitus, no complications
2	5/14/58	Female	370.0	Corneal ulcer
3	8/11/60	Female	753.1	Cystic kidney disease

This is better…



SUBJ_ID	DOB	SEX	ICD
1	1/1/50	Male	250.0
2	5/14/58	Female	370.0
3	8/11/60	Female	753.1

1:1 relationship (as a lookup table)

ICD	DX_TEXT
250.0	Diabetes mellitus, no complications
370.0	Corneal ulcer
753.1	Cystic kidney disease

Third Normal Form: Shortcuts

- A table is in 3NF automatically, if it is in 2NF *and* there are no (or only one) non-key attributes
  - No possibility of a transitive dependency!

Assignment 2: Create a 3NF normalized schema for the ABIC database

Submit as a Word document to Canvas by 9am 2/5