Normal Forms for Relational Databases

Normal Forms

Normal Forms:

- Criteria for a good database design (i.e., to resolve update anomalies)
- Formalized by functional (or other) dependencies

Types of Normal Forms:

- 1NF, 2NF, 3NF (Codd 1972)
- Boyce-Codd NF (1974)

First Normal Form (1NF)

The first normal form (1NF) simply asserts that attribute values be atomic.

(This is part of the definition of the relational model)

Consider the table to the right, adapted from Desai.

All the relations we've seen satisfy the requirements for 1NF.

CRS_PREF			
Prof	Course	Fac_Dept	Crs_Dept
Smith	353	Comp Sci	Comp Sci
Smith	379	Comp Sci	Comp Sci
Smith	221	Comp Sci	Decision Sci
Clark	353	Comp Sci	Comp Sci
Clark	351	Comp Sci	Comp Sci
Clark	379	Comp Sci	Comp Sci
Clark	456	Comp Sci	Mathematics
Turner	353	Chemistry	Comp Sci
Turner	456	Chemistry	Mathematics
Turner	272	Chemistry	Chemistry
Jamieson	353	Mathematics	Comp Sci
Jamieson	379	Mathematics	Comp Sci
Jamieson	221	Mathematics	Decision Sci
Jamieson	456	Mathematics	Mathematics
Jamieson	469	Mathematics	Mathematics

First Normal Form (1NF)

The relation has the following **drawbacks**: (part 1)

(1) Repetition of data:

- The fact that (a given professor is in a given department) may be repeated,
- The fact that (a given course is offered by a given department) may be repeated.

Can result in insertion and update anomalies

CRS_PREF			
Prof	Course	Fac_Dept	Crs_Dept
Smith	353	Comp Sci	Comp Sci
Smith	221	Comp Sci	Decision Sci
Turner	353	Chemistry	Comp Sci
Turner	456	Chemistry	Mathematics

First Normal Form (1NF)

The relation has the following **drawbacks**: (part 2)

(2) Some associations aren't stored explicitly:

- The association between professor and department will not be recorded unless the professor has some course references,
- The association between course and department is not recorded unless someone prefers the course.

Can lead to deletion anomalies

CRS_PREF			
Prof	Course	Fac_Dept	Crs_Dept
Smith	353	Comp Sci	Comp Sci
Smith	379	Comp Sci	Comp Sci
Turner	353	Chemistry	Comp Sci

Unspecified dependencies between Attr.

There is too much information in a table.

- *CRS_PREF(Prof, Course, Fac_Dept, Crs_Dept)*
- The FD's for these attributes are *F = {Prof → Fac_Dept, Course → Crs_Dept}.*
- From F, the only candidate is here *{Prof, Course}.*

From the FDs, we see

- Fac_Dept depends only on Prof and not on Course,
- Crs_Dept depends only on Course and not on Prof.

CRS_PREF			
Prof	**Course**	**Fac_Dept**	**Crs_Dept**
Smith	353	Comp Sci	Comp Sci
Smith	379	Comp Sci	Comp Sci
Smith	221	Comp Sci	Decision Sci
Clark	353	Comp Sci	Comp Sci
Clark	351	Comp Sci	Comp Sci
Clark	379	Comp Sci	Comp Sci
Clark	456	Comp Sci	Mathematics
Turner	353	Chemistry	Comp Sci
Turner	456	Chemistry	Mathematics
Turner	272	Chemistry	Chemistry
Jamieson	353	Mathematics	Comp Sci
Jamieson	379	Mathematics	Comp Sci
Jamieson	221	Mathematics	Decision Sci
Jamieson	456	Mathematics	Mathematics
Jamieson	469	Mathematics	Mathematics

Second Normal Form (2NF) (1)

2NF Prerequisite (1)

Definition (Prime attribute): An attribute of relation schema R is called a prime attribute of R if it is a member of some candidate key of R.

Definition (Nonprime attribute): An attribute is called nonprime if it is not a prime attribute—that is, if it is not a member of any candidate key.

Second Normal Form (2NF) (2)

2NF Prerequisite (2)

Definition (Full functional dependency): In an FD $X \rightarrow Y$, Y is fully functionally dependent on X if there is no $Z \subset X$ such that $Z \rightarrow Y$.

Definition (Partial functional dependency): In an FD $X \rightarrow Y$, Y is partially functionally dependent on X if there is any $Z \subset X$ such that $Z \rightarrow Y$.

Second Normal Form (2NF) (3)

Definition (Full functional dependency): In an FD X→ Y , Y is fully functionally dependent on X if there is no Z ⊂ X such that Z → Y.

Definition (Second Normal Form): A relation schema *R* is in second normal form (2NF) if every nonprime attribute *A* in *R* is not partially dependent on *any* key of *R*.

1. The 2nf is the first nf to consider restrictions on the normal form

2. Why specify nonprimes?

Second Normal Form (2NF) (4)

Recall (Second Normal Form): A relation schema *R* is in second normal form (2NF) if every nonprime attribute *A* in *R* is not partially dependent on *any* key of *R*.

Common confusion: ***Can we consider just the primary key instead?***
Answer: Only if the relation has one candidate key only

Common confusion: ***Why all keys rather than just one key?***
Answer: No special consideration will be given to the primary key over other candidate keys.

Second Normal Form (2NF) (5)

Recall (Second Normal Form): A relation schema *R* is in second normal form (2NF) if every nonprime attribute *A* in *R* is not partially dependent on *any* key of *R*.

Common confusion: ***Is it possible for a nonprime attribute to be partially dependent on the key?***
Answer: Yes

Second Normal Form (2NF)

Let's decompose this 1NF relation:

CRS_PREF			
Prof	**Course**	**Fac_Dept**	**Crs_Dept**
Smith	353	Comp Sci	Comp Sci
Smith	379	Comp Sci	Comp Sci
Smith	221	Comp Sci	Decision Sci
Clark	353	Comp Sci	Comp Sci
Clark	351	Comp Sci	Comp Sci
Clark	379	Comp Sci	Comp Sci
Clark	456	Comp Sci	Mathematics
Turner	353	Chemistry	Comp Sci
Turner	456	Chemistry	Mathematics
Turner	272	Chemistry	Chemistry
Jamieson	353	Mathematics	Comp Sci
Jamieson	379	Mathematics	Comp Sci
Jamieson	221	Mathematics	Decision Sci
Jamieson	456	Mathematics	Mathematics
Jamieson	469	Mathematics	Mathematics

Second Normal Form (2NF)

(You can decompose a 1NF into a 2NF)

COURSE_PREF	
Prof	Course
Smith	353
Smith	379
Smith	221
Clark	353
Clark	351
Clark	379
Clark	456
Turner	353
Turner	456
Turner	272
Jamieson	353
Jamieson	379
Jamieson	221
Jamieson	456
Jamieson	469

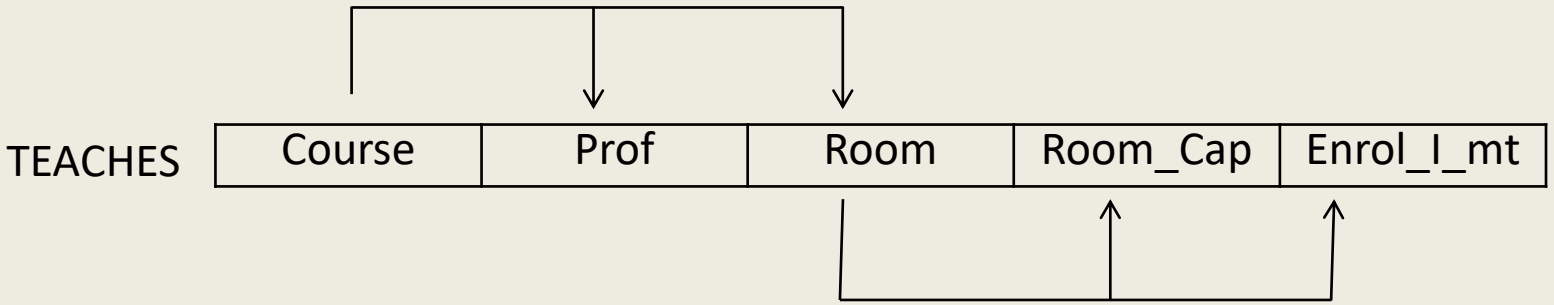
COURSE	
Course	Dept
353	Comp Sci
379	Comp Sci
221	Decision Sci
351	Comp Sci
456	Mathematics
272	Chemistry
469	Mathematics

FACULTY	
Prof	Dept
Smith	Comp Sci
Clark	Comp Sci
Turner	Chemistry
Jamieson	Mathematics

Second Normal Form (2NF)

Recall (Second Normal Form): A relation schema *R* is in second normal form (2NF) if every nonprime attribute *A* in *R* is not partially dependent on *any* key of *R*.

E.g., Is this table 2NF?



The relation TEACHES is in 2NF

Redundancy/Issue in 2NF

Do we resolve all issues? Not for all relations in 2NF.

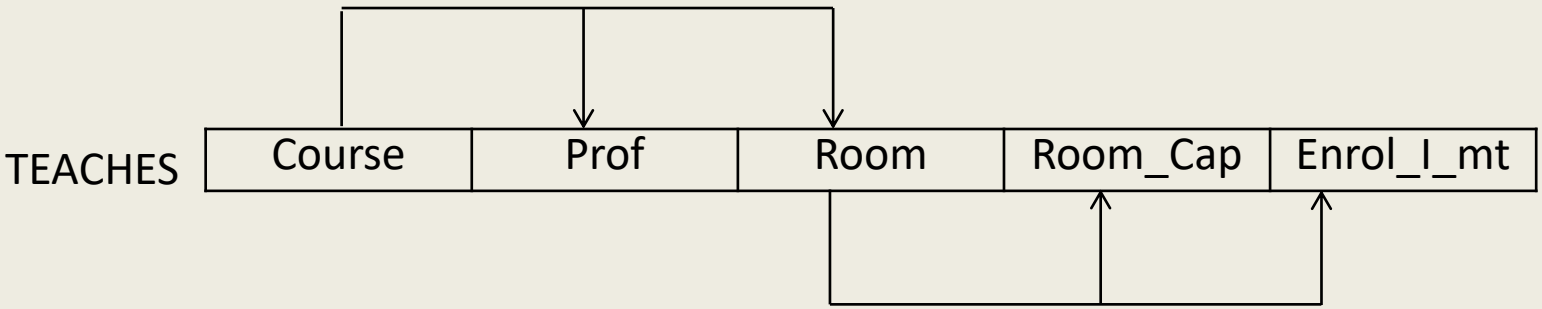
1. The fact that A532 has Room_Cap of 45 and Enrol_Lmt of 40 can still be stored twice.
2. If course 355 is deleted, then the fact that H940 has Room_Cap of 400 and Enrol_Lmt of 300 will be lost.

TEACHES				
Course	Prof	Room	Room_Cap	Enrol_I_mt
353	Smith	*A532*	*45*	*40*
351	Smith	C320	100	60
355	*Clark*	*H940*	*400*	*300*
456	Turner	B278	50	45
459	Jamieson	D110	50	45
500	*Bob*	*A532*	*45*	*40*

Transitive Functional dependency

Definition (Transitive dependency):

A FD X→Y is a transitive dependency if there is a Z that is not a subset of any key, such that X → Z and Z → Y. The attributes of Y are transitively dependent on X.



E.g., Room_Cap is transitively dependent on {Course}, since {Course} → {Room} and {Room} → {Room_Cap} hold, and {Room} is not a subset of any key.

Superkey

Recall (superkey): A superkey of a relation schema $R = \{A1, A2, \ldots, An\}$ is a set of attributes $S \rightarrow R$ with the property that no two tuples $t1$ and $t2$ in any legal relation state r of R will have $t1[S] = t2[S]$.

Recall (key): A key K is a superkey with the additional property that removal of any attribute from K will cause K not to be a superkey anymore.

Third Normal Form (3NF)

Recall (Second Normal Form): A relation schema *R* is in second normal form (2NF) if every nonprime attribute *A* in *R* is not partially dependent on *any* key of *R*.

Definition (Third Normal Form):
A relation scheme is in third normal form (3NF) if for all non-trivial FD's of the form

$$X \rightarrow A$$

- Either
  X is a superkey
- or
  A is a prime attribute.

The 3NF disallows transitive dependencies

Third Normal Form (3NF)

Recall (Third Normal Form):
A relation scheme is in third normal form (3NF) if for all non-trivial FD's of the form X → A. Either X is a superkey or A is a prime attribute.

Let us enumerate all options:

X	A
Superkey	Prime
Superkey	Nonprime
Not Superkey	Prime
Not Superkey	Nonprime

The 3NF disallows FDs of the form "Not superkey -> Nonprime"

Note: not Superkey = either (1) *prime attributes that are proper subset of a key* or (2) *nonprime*

Third Normal Form (3NF)

Lets compare the 2nf and the 3nf

<table>
<tr><td colspan="2">X</td><td>A</td></tr>
<tr><td colspan="2">Superkey</td><td>Prime</td></tr>
<tr><td colspan="2">Superkey</td><td>Nonprime</td></tr>
<tr><td colspan="2">Not Superkey</td><td>Prime</td></tr>
<tr><td rowspan="2">Not Super Key</td><td>Prime attributes (proper subsets)</td><td rowspan="2">Nonprime</td></tr>
<tr><td>Non prime</td></tr>
</table>

2NF allows FDs of the type nonprime -> nonprime

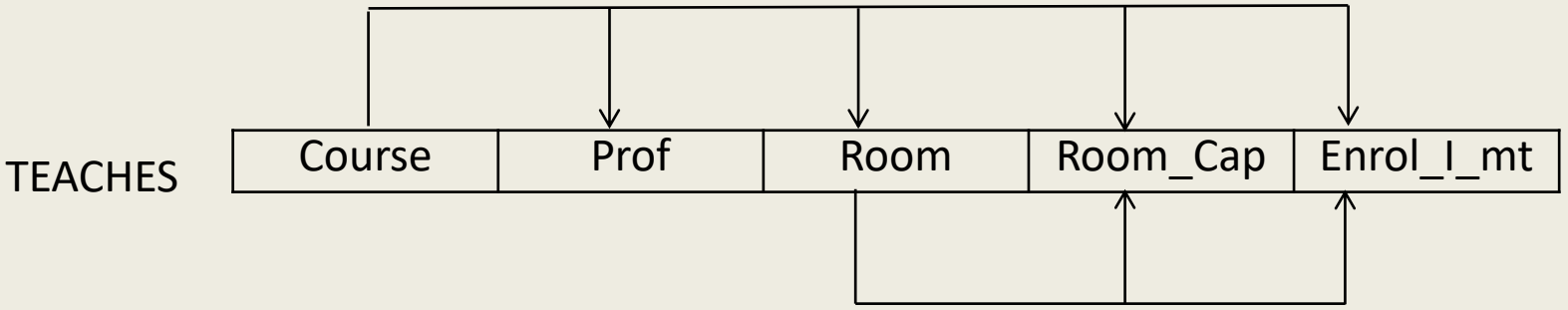
3NF disallows nonprime -> nonprime in addition

Third Normal Form (3NF)

Definition (Transitive dependency):

A FD X→Y is a transitive dependency if there is a Z that is not a subset of any key, such that X → Z and Z → Y. The attributes of Y are transitively dependent on X.

Test for 3NF (for simple one key): There should be no transitive dependency of a nonkey attribute on the primary key.



Third Normal Form (3NF)

You can decompose TEACHES from 2NF into 3NF:

TEACHES				
Course	Prof	Room	Room_Cap	Enrol_l_mt
353	Smith	*A532*	*45*	*40*
351	Smith	C320	100	60
355	*Clark*	*H940*	*400*	*300*
456	Turner	B278	50	45
459	Jamieson	D110	50	45
500	*Bob*	*A532*	*45*	*40*

2NF
(but not 3NF)

COURSE_DETAILS		
Course	Prof	Room
353	Smith	A532
351	Smith	C320
456	Turner	B278
459	Jamieson	D110
355	Clark	H940

ROOM_DETAILS		
Room	Room_Cap	Enrol_l_mt
A532	45	40
C320	100	60
B278	50	45
D110	50	45
H940	400	300

3NF

Summary

We have covered 1NF, 2NF, 3NF (Codd 1972)

Details:

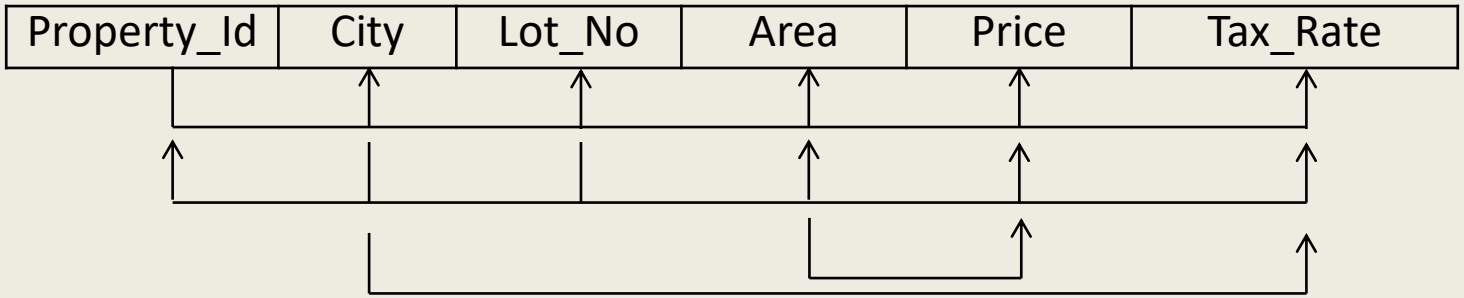
- Each normal form satisfies the requirements of the lower normal forms
- The normal form of a relation is the highest normal form it satisfies.
- A relation is “normalized“ if it meets the 3NF

"The key, the whole key, and nothing but the key, so help me Codd”

Review

What normal form is this?

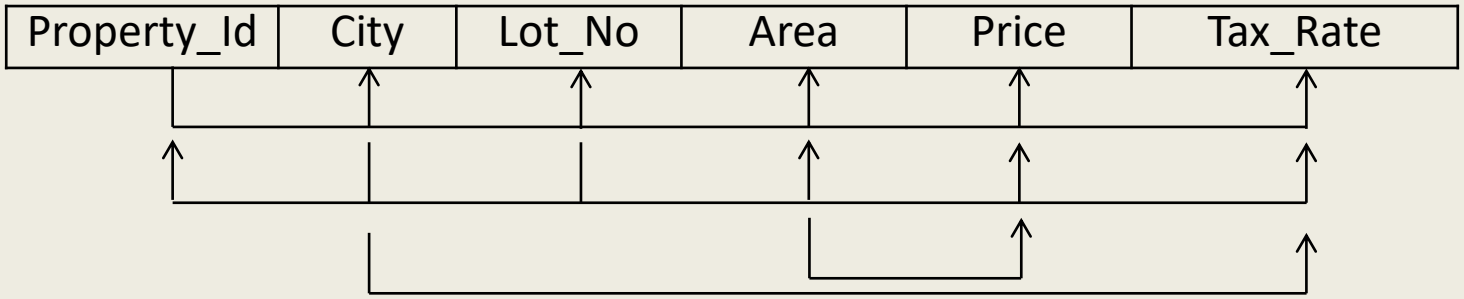
(Assume given that this is 1NF)



Is it in 2NF?

Review

LOTS

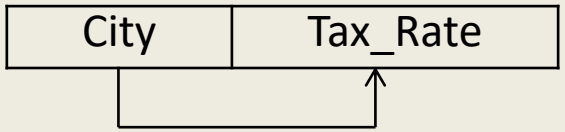


LOTS is **not in 2NF**:
Since City→Tax_Rate, Tax_Rate is not prime, and *{City,Lot_No}* is a key, making Tax_Rate partially dependent on a key.

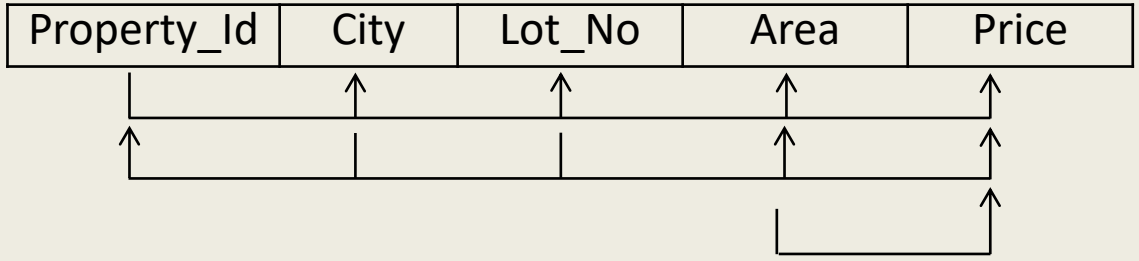
Review

Now we have 2NF

LOTS1



LOTS2

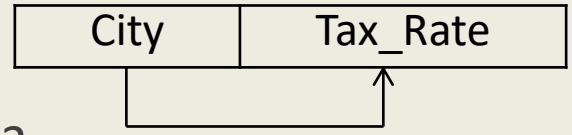


Is this in 3NF?

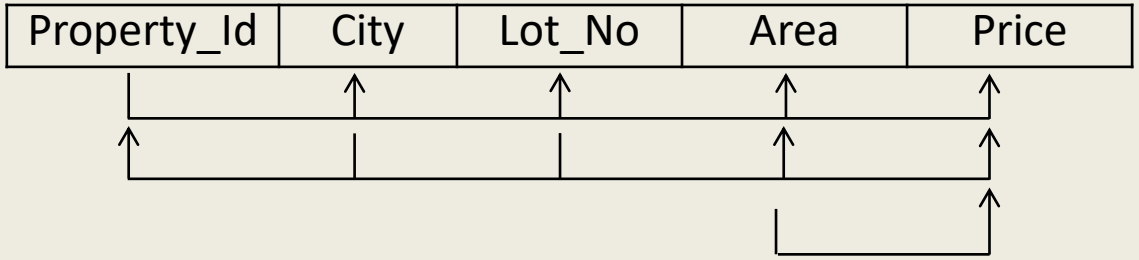
Review

Now we have 2NF

LOTS1



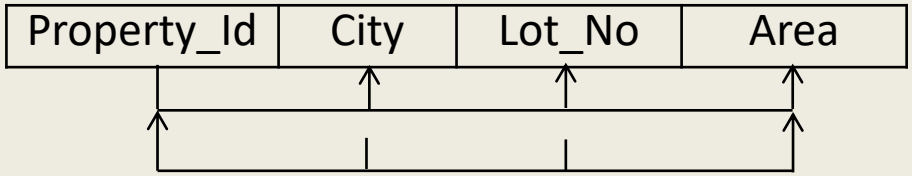
LOTS2



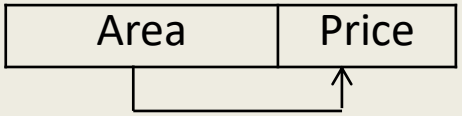
LOTS2 is **not in 3NF**:
Since Area → Price, {Area} is not a superkey and Price is not prime.

Third Normal Form (3NF) (cont)

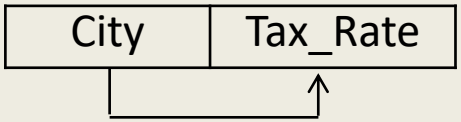
LOTS1A



LOTS1B

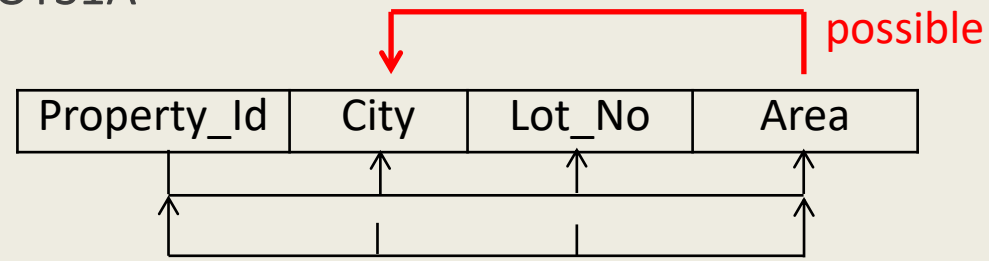


LOTS2

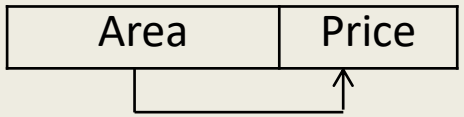


Third Normal Form (3NF) (cont)

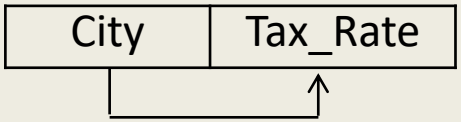
LOTS1A



LOTS1B



LOTS2



It is possible that the dependency *Area* → *City* exists

The relations schemes are still in 3NF since City is a prime attribute. However, there can be anomalies, just as before.

To illustrate the anomalies, we will use a simpler example in next slide.

3NF Example

Consider the booking relation *R*

- *R = (title, theater, city)*
  F = {theater → city, title city → theater}
- Two candidate keys: *{title, city}* and *{theater, title}*
- R is in 3NF
  - *title city → theater*
    {title, city} is a superkey
  - *theater → city*
    city is contained in a candidate key

Redundancy in 3NF

There is some **redundancy** in this schema

Example of problems due to redundancy in 3NF

- R = (title, theater, city)
  F = {theater $\rightarrow$ city, title city $\rightarrow$ theater }

title	*theater*	*city*
j_1	l_1	k_1
j_2	l_1	k_1
j_3	l_1	k_1

- repetition of information (e.g., the relationship l_1, k_1)

Boyce-Codd Normal Form (BCNF)

Definition (Third Normal Form):
A relation scheme is in third normal form (3NF) if for all non-trivial FD's of the form X → A

- Either X is a superkey
- or A is a prime attribute.

Definition (Boyce-Codd Normal Form): A relation scheme is in Boyce-Codd Normal Form (BCNF) if whenever X→A holds and X→A is non-trivial, X is a superkey.

BCNF is also known as 3.5NF

Boyce-Codd Normal Form (BCNF)

Definition (Boyce-Codd Normal Form): A relation scheme is in Boyce-Codd Normal Form (BCNF) if whenever X→A holds and X→A is non-trivial, X is a superkey.

Let us enumerate all options:

X	A
Superkey	Prime
Superkey	Nonprime
~~Not Superkey~~	~~Prime~~
~~Not Superkey~~	~~Nonprime~~

Note: not Superkey = either (1) *prime attributes that are proper subset of a key* or (2) *nonprime*

Comparisons

BCNF implies 3NF

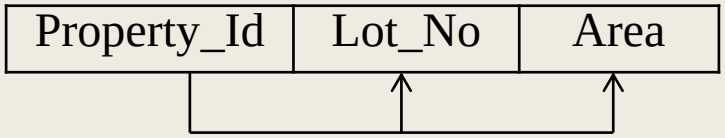
Property	3NF	BCNF
Elimination of redundancy due to functional dependency	Most	Yes
Lossless Join	Yes	Yes
Dependency preservation due to functional dependency	Yes	Maybe

It is not always possible to get a BCNF decomposition that is dependency preserving (dependency preservation discussed in future lectures)

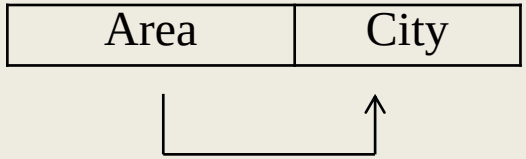
Boyce-Codd Normal Form (BCNF)(cont)

We can make our example into BCNF:

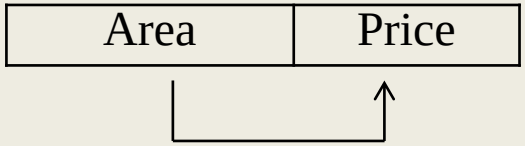
LOTS1AA



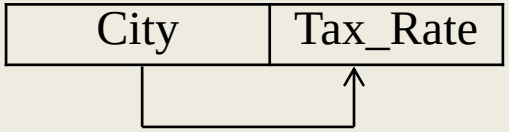
LOTS1AB



LOTS1B



LOTS2

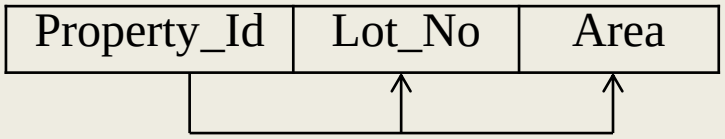


(One possible decomposition to satisfy BCNF)

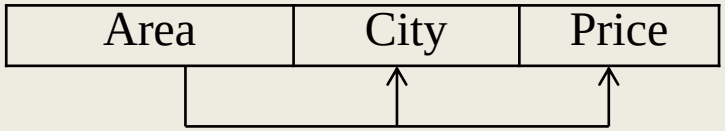
Boyce-Codd Normal Form (BCNF)(cont)

We can make our example into BCNF:

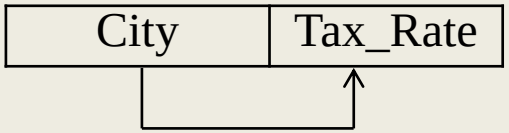
LOTS1AA



LOTS1C



LOTS2



(LOTS1AB and LOTS1B (from prev slide) into LOTS1C)

Discussion

We discussed that the normal form of a relation is the highest NF it satisfies. E.g., R is 2NF means that R is not 3NF or BCNF.

In terms of the database scheme as a whole…

- A database scheme is in 1NF if all its relations are in 1NF.
- A database scheme is in 2NF if all its relations are in 2NF.
- A database scheme is in 3NF if all its relations are in 3NF.
- A database scheme is in BCNF if all its relations are in BCNF.

Learning Outcomes

1. Definitions of 1NF, 2NF, 3NF and BCNF
2. (3NF and BCNF is the key NFs to understand)
3. How the presence of some types of functional dependencies can contribute to update anomalies
4. Determine the highest NF of a relation