# Lesson 8: Functional Dependencies & Normalization

CSC430/530 – DATABASE MANAGEMENT SYSTEMS

# OUTLINE

- Introduction.
- Informal design guidelines.
- Functional dependencies.
- Normalization.
  - First normal form (1NF).
  - Second normal form (2NF).
  - Third normal form (3NF).
  - Boyce-Codd normal form (BCNF).

# INTRODUCTION

- **Relational database design**.
  - Process of **grouping** of **attributes** to form *"good"* **relation schemas**.
- Two **levels** of relational schemas:
  - **Logical level** – how user views the relations.
  - **Physical level** – how tuples are stored and updated.
- Relational design is mainly focused on **physical level** (*base relations*).
  - The **product** of relational database **design** is a set of **base relations**.
- Relational design has two implicit **goals**:
  - **Information preservation**.
    - Capturing all the concepts from EER design.
  - **Minimizing redundancy**.
    - Reducing redundant storage of the same information and the need for multiple updates.

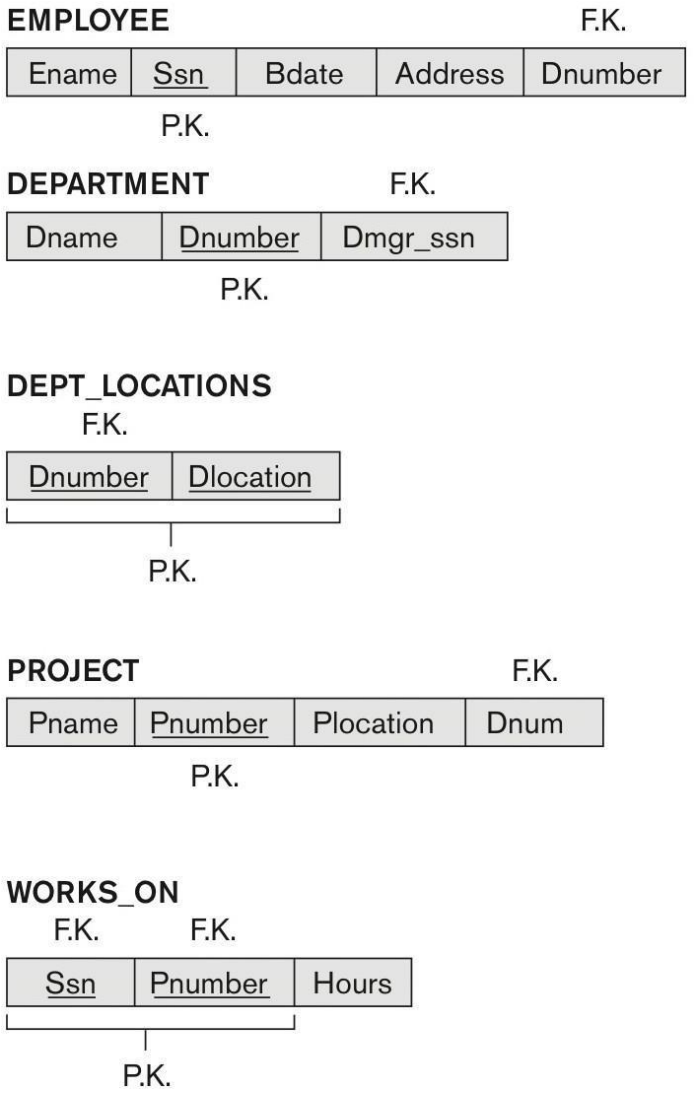
# INFORMAL DESIGN GUIDELINES: OVERVIEW

- **Informal design guidelines** that determine the quality of the relation schema design:
  - Making sure that the **semantics** of the attributes is **clear** in the schema;
  - **Reducing** the **redundant** information in tuples;
  - **Reducing** the **NULL values** in tuples;
  - **Disallowing** the possibility of generating **spurious tuples**.

# INFORMAL DESIGN GUIDELINES (1)

- **GUIDELINE 1:** Each tuple in a relation should represent **one** entity or relationship **instance**.
  - Attributes of different entities should **not be mixed** in the same relation.
  - Only **foreign keys** should be used to **refer** to other entities.
- Design schema that can be **explained easily** relation by relation.
  - The semantics of attributes should be **easy** to **interpret**.

# INFORMAL DESIGN GUIDELINES (1)




Simplified COMPANY relational schema

**(a)**

**EMP_DEPT**

| Ename | Ssn | Bdate | Address | Dnumber | Dname | Dmgr_ssn |
|---|---|---|---|---|---|---|

**(b)**

**EMP_PROJ**

| Ssn | Pnumber | Hours | Ename | Pname | Plocation |
|---|---|---|---|---|---|

Modified COMPANY relational schema

# INFORMAL DESIGN GUIDELINES (2)

- **GUIDELINE 2:** Schema should **not contain** *redundant information* and must **not suffer** from the *insertion*, *deletion* and *update* anomalies.
- If information stored **redundantly**:
  - **Wastes** storage space.
  - **Causes** update anomalies.
    - Insertion anomalies.
    - Deletion anomalies.
    - Modification anomalies.

**(a)**

**EMP_DEPT**

| Ename | Ssn | Bdate | Address | Dnumber | Dname | Dmgr_ssn |
|---|---|---|---|---|---|---|

**(b)**

**EMP_PROJ**

| Ssn | Pnumber | Hours | Ename | Pname | Plocation |
|---|---|---|---|---|---|

EMP_DEPT and EMP_PROJ relations

# INFORMAL DESIGN GUIDELINES (2)

Redundancy

**EMP_DEPT**

| Ename | Ssn | Bdate | Address | Dnumber | Dname | Dmgr_ssn |
|---|---|---|---|---|---|---|
| Smith, John B. | 123456789 | 1965-01-09 | 731 Fondren, Houston, TX | 5 | Research | 333445555 |
| Wong, Franklin T. | 333445555 | 1955-12-08 | 638 Voss, Houston, TX | 5 | Research | 333445555 |
| Zelaya, Alicia J. | 999887777 | 1968-07-19 | 3321 Castle, Spring, TX | 4 | Administration | 987654321 |
| Wallace, Jennifer S. | 987654321 | 1941-06-20 | 291 Berry, Bellaire, TX | 4 | Administration | 987654321 |
| Narayan, Ramesh K. | 666884444 | 1962-09-15 | 975 FireOak, Humble, TX | 5 | Research | 333445555 |
| English, Joyce A. | 453453453 | 1972-07-31 | 5631 Rice, Houston, TX | 5 | Research | 333445555 |
| Jabbar, Ahmad V. | 987987987 | 1969-03-29 | 980 Dallas, Houston, TX | 4 | Administration | 987654321 |
| Borg, James E. | 888665555 | 1937-11-10 | 450 Stone, Houston, TX | 1 | Headquarters | 888665555 |

Example of a redundancy in EMP_DEPT relation

# INFORMAL DESIGN GUIDELINES (2)

Redundancy Redundancy

**EMP_PROJ**

| Ssn | Pnumber | Hours | Ename | Pname | Plocation |
|---|---|---|---|---|---|
| 123456789 | 1 | 32.5 | Smith, John B. | ProductX | Bellaire |
| 123456789 | 2 | 7.5 | Smith, John B. | ProductY | Sugarland |
| 666884444 | 3 | 40.0 | Narayan, Ramesh K. | ProductZ | Houston |
| 453453453 | 1 | 20.0 | English, Joyce A. | ProductX | Bellaire |
| 453453453 | 2 | 20.0 | English, Joyce A. | ProductY | Sugarland |
| 333445555 | 2 | 10.0 | Wong, Franklin T. | ProductY | Sugarland |
| 333445555 | 3 | 10.0 | Wong, Franklin T. | ProductZ | Houston |
| 333445555 | 10 | 10.0 | Wong, Franklin T. | Computerization | Stafford |
| 333445555 | 20 | 10.0 | Wong, Franklin T. | Reorganization | Houston |
| 999887777 | 30 | 30.0 | Zelaya, Alicia J. | Newbenefits | Stafford |
| 999887777 | 10 | 10.0 | Zelaya, Alicia J. | Computerization | Stafford |
| 987987987 | 10 | 35.0 | Jabbar, Ahmad V. | Computerization | Stafford |
| 987987987 | 30 | 5.0 | Jabbar, Ahmad V. | Newbenefits | Stafford |
| 987654321 | 30 | 20.0 | Wallace, Jennifer S. | Newbenefits | Stafford |
| 987654321 | 20 | 15.0 | Wallace, Jennifer S. | Reorganization | Houston |
| 888665555 | 20 | Null | Borg, James E. | Reorganization | Houston |

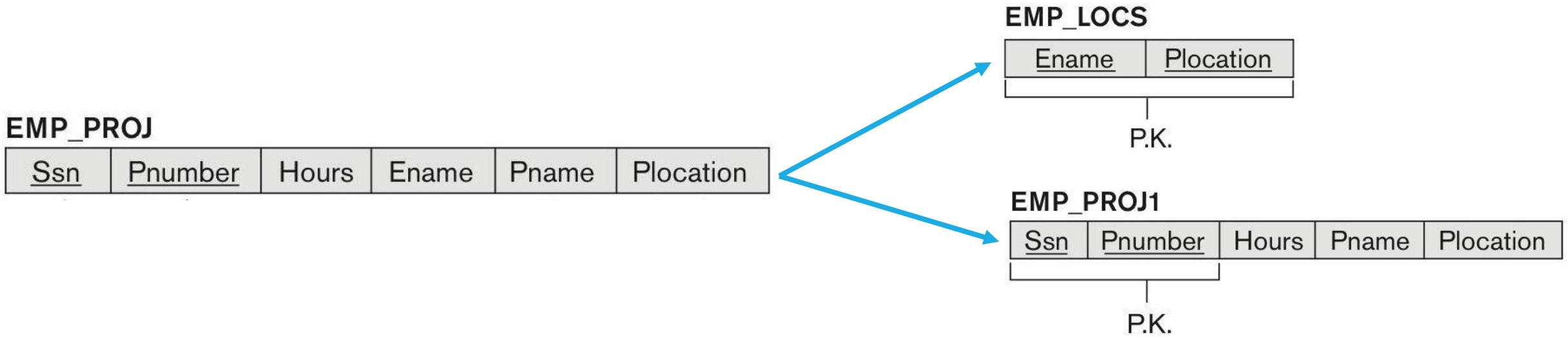
Example of a redundancy in EMP_PROJ relation

# INFORMAL DESIGN GUIDELINES (3)

- **GUIDELINE 3:** Relations should be designed such that their tuples have as **few NULL** values as possible.
  - If NULLs are **unavoidable**, make sure that they apply in **exceptional cases** only and do not apply to a **majority** of tuples in the relation.
- **Negative effects** of NULL values:
  - **Waste** of space.
  - **Harder** to **understand** the meaning of an attribute.
  - **Harder** to **apply** *JOINs* and aggregate functions (*COUNT*, *SUM*).

# INFORMAL DESIGN GUIDELINES (4)

- **GUIDELINE 4:** Relation schemas should only be able to be **joined** with equality conditions on **attributes** that are appropriately **related** pairs (*foreign key – primary key*).
  - This guarantees that **no spurious** tuples are generated.
- **Joining** relations that contain **matching attributes** that are not *"foreign key - primary key"* combinations may produce **spurious tuples**.




EMP_PROJ relation broken down into EMP_LOCS and EMP_PROJ1 relations

# INFORMAL DESIGN GUIDELINES (4)

**EMP_LOCS**

| Ename | Plocation |
|---|---|
| Smith, John B. | Bellaire |
| Smith, John B. | Sugarland |
| Narayan, Ramesh K. | Houston |
| English, Joyce A. | Bellaire |
| English, Joyce A. | Sugarland |
| Wong, Franklin T. | Sugarland |
| Wong, Franklin T. | Houston |
| Wong, Franklin T. | Stafford |
| Zelaya, Alicia J. | Stafford |
| Jabbar, Ahmad V. | Stafford |
| Wallace, Jennifer S. | Stafford |
| Wallace, Jennifer S. | Houston |
| Borg, James E. | Houston |

**EMP_PROJ1**

| Ssn | Pnumber | Hours | Pname | Plocation |
|---|---|---|---|---|
| 123456789 | 1 | 32.5 | ProductX | Bellaire |
| 123456789 | 2 | 7.5 | ProductY | Sugarland |
| 666884444 | 3 | 40.0 | ProductZ | Houston |
| 453453453 | 1 | 20.0 | ProductX | Bellaire |
| 453453453 | 2 | 20.0 | ProductY | Sugarland |
| 333445555 | 2 | 10.0 | ProductY | Sugarland |
| 333445555 | 3 | 10.0 | ProductZ | Houston |
| 333445555 | 10 | 10.0 | Computerization | Stafford |
| 333445555 | 20 | 10.0 | Reorganization | Houston |
| 999887777 | 30 | 30.0 | Newbenefits | Stafford |
| 999887777 | 10 | 10.0 | Computerization | Stafford |
| 987987987 | 10 | 35.0 | Computerization | Stafford |
| 987987987 | 30 | 5.0 | Newbenefits | Stafford |
| 987654321 | 30 | 20.0 | Newbenefits | Stafford |
| 987654321 | 20 | 15.0 | Reorganization | Houston |
| 888665555 | 20 | NULL | Reorganization | Houston |

State of EMP_LOCS and EMP_PROJ1 relations

# INFORMAL DESIGN GUIDELINES (4)

| | Ssn | Pnumber | Hours | Pname | Plocation | Ename |
|---|---|---|---|---|---|---|
| | 123456789 | 1 | 32.5 | ProductX | Bellaire | Smith, John B. |
| * | 123456789 | 1 | 32.5 | ProductX | Bellaire | English, Joyce A. |
| | 123456789 | 2 | 7.5 | ProductY | Sugarland | Smith, John B. |
| * | 123456789 | 2 | 7.5 | ProductY | Sugarland | English, Joyce A. |
| * | 123456789 | 2 | 7.5 | ProductY | Sugarland | Wong, Franklin T. |
| | 666884444 | 3 | 40.0 | ProductZ | Houston | Narayan, Ramesh K. |
| * | 666884444 | 3 | 40.0 | ProductZ | Houston | Wong, Franklin T. |
| * | 453453453 | 1 | 20.0 | ProductX | Bellaire | Smith, John B. |
| | 453453453 | 1 | 20.0 | ProductX | Bellaire | English, Joyce A. |
| * | 453453453 | 2 | 20.0 | ProductY | Sugarland | Smith, John B. |
| | 453453453 | 2 | 20.0 | ProductY | Sugarland | English, Joyce A. |
| * | 453453453 | 2 | 20.0 | ProductY | Sugarland | Wong, Franklin T. |
| * | 333445555 | 2 | 10.0 | ProductY | Sugarland | Smith, John B. |
| * | 333445555 | 2 | 10.0 | ProductY | Sugarland | English, Joyce A. |
| | 333445555 | 2 | 10.0 | ProductY | Sugarland | Wong, Franklin T. |
| * | 333445555 | 3 | 10.0 | ProductZ | Houston | Narayan, Ramesh K. |
| | 333445555 | 3 | 10.0 | ProductZ | Houston | Wong, Franklin T. |
| | 333445555 | 10 | 10.0 | Computerization | Stafford | Wong, Franklin T. |
| * | 333445555 | 20 | 10.0 | Reorganization | Houston | Narayan, Ramesh K. |
| | 333445555 | 20 | 10.0 | Reorganization | Houston | Wong, Franklin T. |

*
*
*

Spurious tuples when joining EMP_LOCS and EMP_PROJ1 relations

# FUNCTIONAL DEPENDENCIES (1)

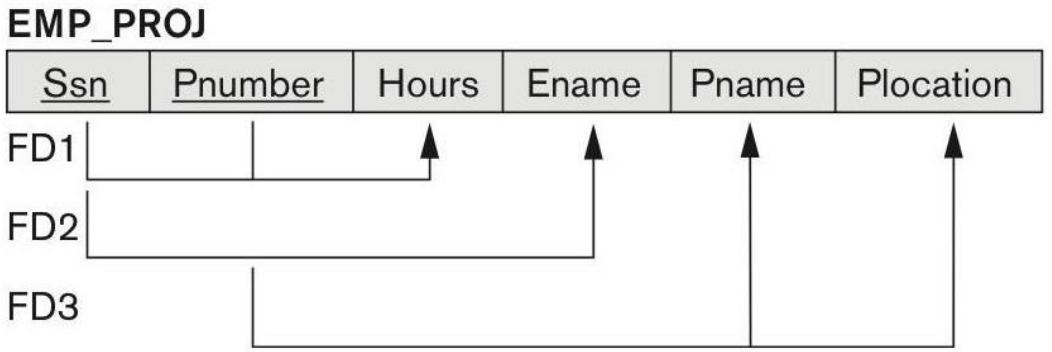
- **Functional dependencies (FDs)** are used to specify **formal** measures of "*goodness*" of the database design.
  - **FD** is a **constraint** between two sets of data **attributes**, derived from real-world constrains (*semantic*) on these attributes.
- **FD** is written as *X* -> *Y*.
  - *X* - left-hand-side (LHS).
  - *Y* - right-hand-side (RHS).
- A **set** of **attributes** *X* functionally **determines** a **set** of **attributes** *Y* if the value of *X* **determines** a unique value for *Y*.
  - *X* -> *Y* holds if whenever two tuples have the same value for X, they must have the same value for Y.
    - If $t_1[X] = t_2[X]$, then $t_1[Y] = t_2[Y]$.

# FUNCTIONAL DEPENDENCIES (2)

- FD constraint is a **property** of the **relation schema** and cannot be inferred automatically from a particular relation **state**.
  - FD **constraint** must hold on **every relation state**.
- If set of attribute *K* is a **key** of relation *R*, then *K* **functionally determines** all attributes in *R.*
  - Since there are **no two distinct** tuples with $t_1[K]=t_2[K]$.

# FUNCTIONAL DEPENDENCIES: EXAMPLES (1)

- Examples of **possible FDs** from EMP_PROJ relation:
- **Ssn -> Ename**
  - The value of an employee's social security number (Ssn) uniquely determines the employee name (Ename).
- **Pnumber -> {Pname, Plocation}**
  - The value of a project's number (Pnumber) uniquely determines the project name (Pname) and location (Plocation).
- **{SSN, Pnumber} -> Hours**
  - Combination of social security number (Ssn) and project number (Pnumber) values uniquely determines the number of hours the employee currently works on the project per week (Hours).
- FD is displayed as a horizontal line, LHS is a vertical line, RHS is a vertical line with arrow.



# FUNCTIONAL DEPENDENCIES: EXAMPLES (2)

- Specify **possible FD(s)** for following relation:

**TEACH**

| Teacher | Course | Text |
|---|---|---|
| Smith | Data Structures | Bartram |
| Smith | Data Management | Martin |
| Hall | Compilers | Hoffman |
| Brown | Data Structures | Horowitz |

| A | B | C | D |
|---|---|---|---|
| a1 | b1 | c1 | d1 |
| a1 | b2 | c2 | d2 |
| a2 | b2 | c2 | d3 |
| a3 | b3 | c4 | d3 |

- FDs that **may hold**:
  - Text -> Teacher
  - Text -> Course
- FDs that are **violated**:
  - Teacher -> Course
  - Teacher -> Text
  - Course -> Text
  - Course -> Teacher

# NORMALIZATION (1)

- **Normalization** (*of a relation*) – top-down **process** that **evaluates** relation against the criteria for **normal forms** and **decomposes** relation as necessary.
  - **Relation design** by *analysis*.
- Four main **normal forms**:
  - **First** normal form (*1NF*).
  - **Second** normal form (*2NF*).
  - **Third** normal form (*3NF*).
  - **Boyce-Codd** normal form (*BCNF*).
    - **Stricter** form of 3NF.
- Two additional NF (not used in practice):
  - Forth normal form (4NF).
  - Fifth normal form (5NF).
- All NFs are based on **functional dependencies** among the attributes of the relation & their **PKs**.

# NORMALIZATION (2)

- **Goals** of the **normalization** process:
  - **Minimize redundancy**.
  - **Minimize** the insertion, deletion, and update **anomalies.**
- **Normalization process** insures that the **relational schema** complies with two **properties**:
  - **Non-additive join** (*lossless* join) property.
    - No spurious tuples are generated after decomposition.
  - **Dependency preservation** property.
    - Each FD is represented in some relation after decomposition.

# DEFINITION OF KEYS (REVIEW)

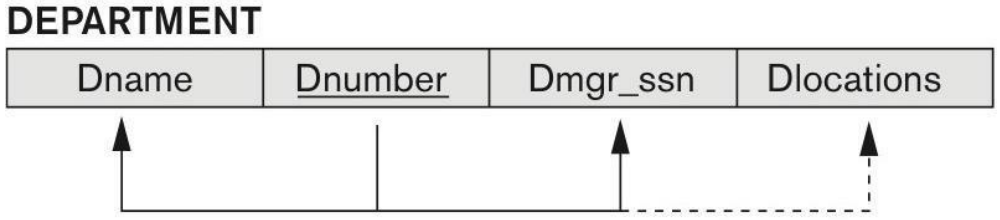
- **Superkey**.
  - Set of attributes with the property that no two tuples in any relation state have same values.
- **Key**.
  - Minimal superkey – If we remove an attribute from key it would no longer be a superkey.
- **Candidate key(s)**.
  - If more then one key in a relation, then each is a candidate key.
- **Primary key**.
  - One of the candidate keys that is arbitrarily designated to be primary.
- **Prime attribute.**
  - Attribute that is a member of some candidate key.
- **Non-prime attribute.**
  - Attribute that is not a member of any candidate key.

# FIRST NORMAL FORM (1NF)

- **First normal form (1NF) informal definition**.
  - No multivalued or composite attributes and their combinations are allowed.
- **First normal form (1NF) formal definition**.
  - **Domains** of the attributes must include only **atomic** (*simple, indivisible*) values and every value must be a **single** value.
    - No sets of values, tuples of values, or combinations.
    - No relations within relation or relations as attribute values within tuples.

# FIRST NORMAL FORM: EXAMPLE (1)

- DEPARTMENT relation is **not** in **1NF**, since *Dlocation* attributes is **not atomic** (*multivalued*).




**DEPARTMENT**

| Dname | Dnumber | Dmgr_ssn | Dlocations |
|---|---|---|---|
| Research | 5 | 333445555 | {Bellaire, Sugarland, Houston} |
| Administration | 4 | 987654321 | {Stafford} |
| Headquarters | 1 | 888665555 | {Houston} |

- Three ways to **normalize** DEPARTMENT into **1NF**:
  - **Remove** *Dlocation* **attribute** and place it in a **separate relation** DEPT_LOCATIONS.
    - Propagate PK from DEPARTMENT, thus PK of DEPT_LOCATIONS is {*Dnumber*, *Dlocation*}.
  - **Expand** the **PK** of DEPARTMENT to be {*Dnumber*, *Dlocation*}.
    - Disadvantage – redundancy.
  - Max # of locations known → **create separate attribute** for each location.
    - *Dlocation1*, *Dlocation2*, *Dlocation3* …
    - Disadvantages – NULL values & harder to write queries.

# FIRST NORMAL FORM: EXAMPLE (1)

- **Solution 1** (*recommended*):

**DEPT_LOCATIONS**

| Dnumber | Dlocation |
|---|---|
| 1 | Houston |
| 4 | Stafford |
| 5 | Bellaire |
| 5 | Sugarland |
| 5 | Houston |

- **Solution 2** (*redundant*):

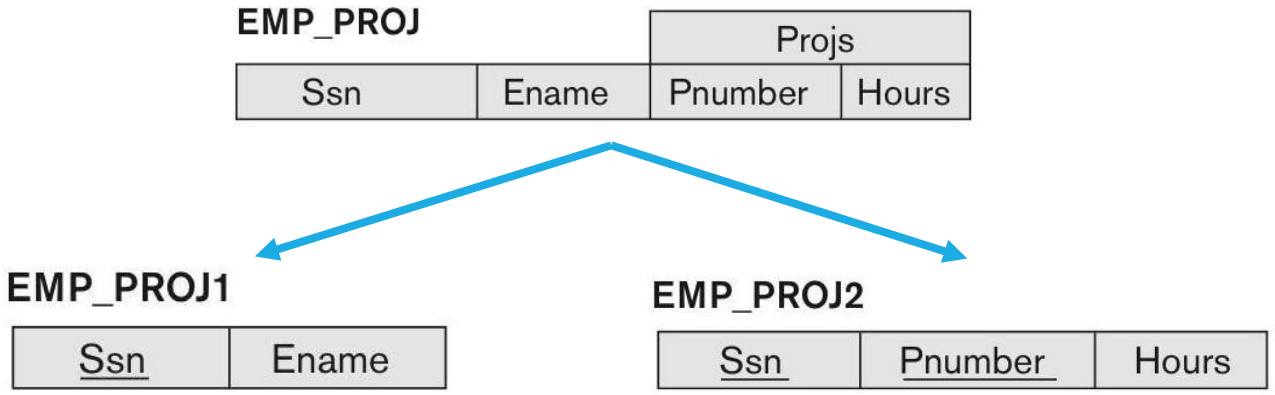
**DEPARTMENT**

| Dname | Dnumber | Dmgr_ssn | Dlocation |
|---|---|---|---|
| Research | 5 | 333445555 | Bellaire |
| Research | 5 | 333445555 | Sugarland |
| Research | 5 | 333445555 | Houston |
| Administration | 4 | 987654321 | Stafford |
| Headquarters | 1 | 888665555 | Houston |

# FIRST NORMAL FORM: EXAMPLE (2)

- EMP_PROJ relation is **not** in **1NF**.
  - EMP_PROJ is a **nested relation**.
    - Each tuple is an employee entity with a relation *PROJS(Pnumber, Hours)* as an attribute.

- **Normalized** into 1NF by **removing** nested relation **attribute** into **separate relation** and **propagating** PK of EMP_PROJ relation.




**EMP_PROJ**

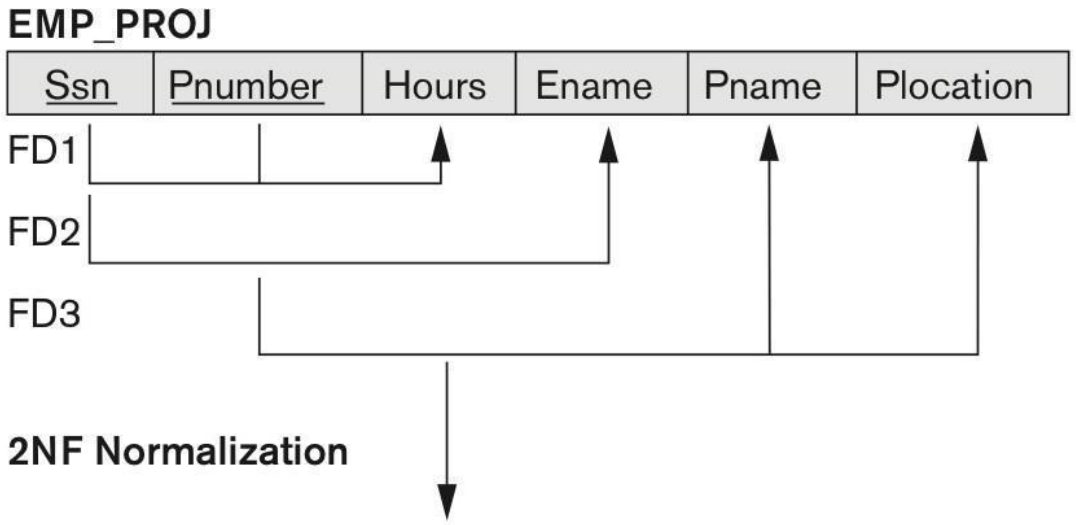
| Ssn | Ename | Pnumber | Hours |
|---|---|---|---|
| 123456789 | Smith, John B. | 1 | 32.5 |
| | | 2 | 7.5 |
| 666884444 | Narayan, Ramesh K. | 3 | 40.0 |
| 453453453 | English, Joyce A. | 1 | 20.0 |
| | | 2 | 20.0 |
| 333445555 | Wong, Franklin T. | 2 | 10.0 |
| | | 3 | 10.0 |
| | | 10 | 10.0 |
| | | 20 | 10.0 |
| 999887777 | Zelaya, Alicia J. | 30 | 30.0 |
| | | 10 | 10.0 |
| 987987987 | Jabbar, Ahmad V. | 10 | 35.0 |
| | | 30 | 5.0 |
| 987654321 | Wallace, Jennifer S. | 30 | 20.0 |
| | | 20 | 15.0 |
| 888665555 | Borg, James E. | 20 | NULL |

# SECOND NORMAL FORM (2NF)

- **Second normal form (2NF) formal definition**.
  - Relation is in **2NF** if every **non-prime attribute** is **fully functionally dependent** on the **PK**.
- **Full functional dependency.**
  - FD X -> Y is **full functional dependency** if **removing** any attributes from X means dependency **does not hold** anymore.
    - {*Ssn*, *Pnumber*} -> *Hours*
      - Neither *Ssn* -> *Hours*, nor *Pnumber* -> *Hours* holds.
- **Partial function dependency.**
  - FD X -> Y is **partial functional dependency** if some attribute(s) can be **removed** from X and the dependency **still holds**.
    - {*Ssn*, *Pnumber*} -> *Ename*
      - *Ssn* -> *Ename* holds.

# SECOND NORMAL FORM: EXAMPLE

- EMP_PROJ relation is **1NF**, but **not** in **2NF**.

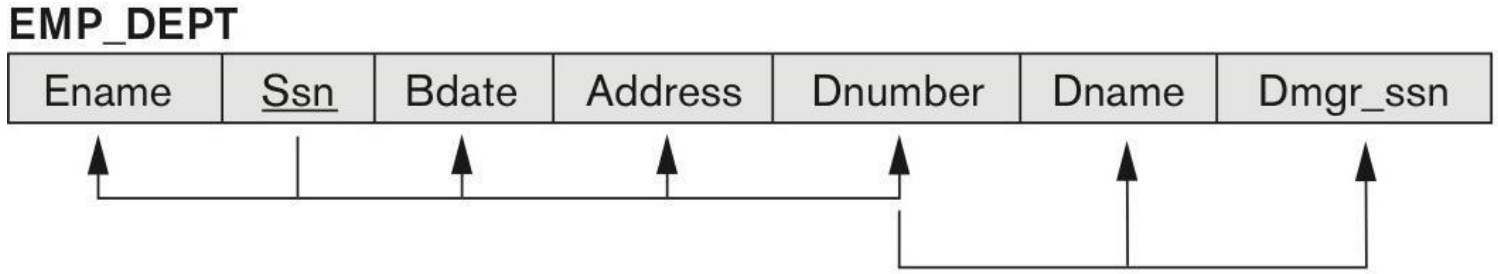



- **Violations** of 2NF:
  - **FD2**: *Ename* is functionally determined **only** by *Ssn*.
  - **FD3**: *Pname* & *Plocation* is functionally determined **only** by *Pnumber*.
- *Ename*, *Pname* & *Plocation* (*non-prime attributes*) **partially functionally dependent** on {*Ssn*, *Pnumber*} **PK**.
  - Both FD2 & FD3 **violate** 2NF.

- **Normalizing** EMP_PROJ into **2NF**:
  - **Decompose** relation into **2NF relations** where **non-prime attributes** are associated only with the **part** of the **PK** on which they are **fully functionally dependent**.

# THIRD NORMAL FORM (3NF)

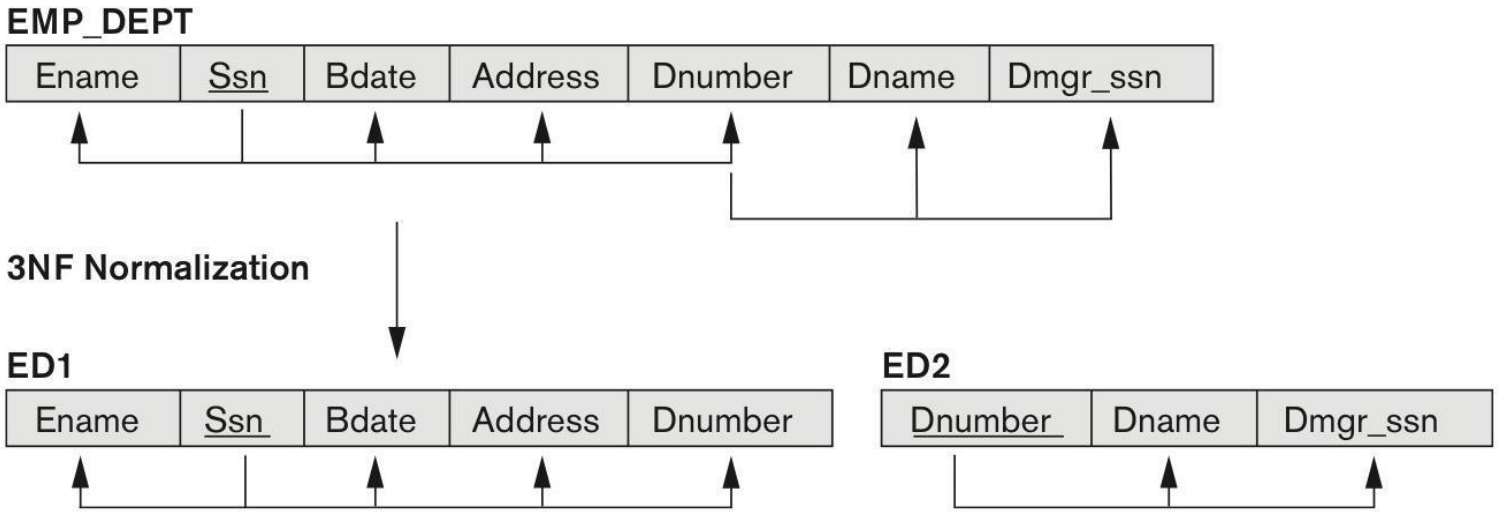
- **Third normal form (3NF) formal definition.**
  - Relation is in **3NF** if it is in **2NF** and no **non-prime attribute(s)** is **transitively dependent** on the **PK**.
- **Transitive dependency.**
  - FD X -> Y is **transitive dependency** if there exists set of attributes Z that is neither a **candidate key** nor a **subset** of any **keys**, and both X -> Z and Z -> Y **hold**.
  - Example:




  - *Ssn -> Dmgr_ssn* and *Ssn -> Dname* are **transitive through** *Dnumber*.
    - *Ssn -> Dnumber* and *Dnumber -> Dmgr_ssn* hold.
    - *Ssn -> Dnumber* and *Dnumber -> Dname* hold.
      - *Dnumber* is not a **key** and not a **subset** of a **key**.

# THIRD NORMAL FORM: EXAMPLE

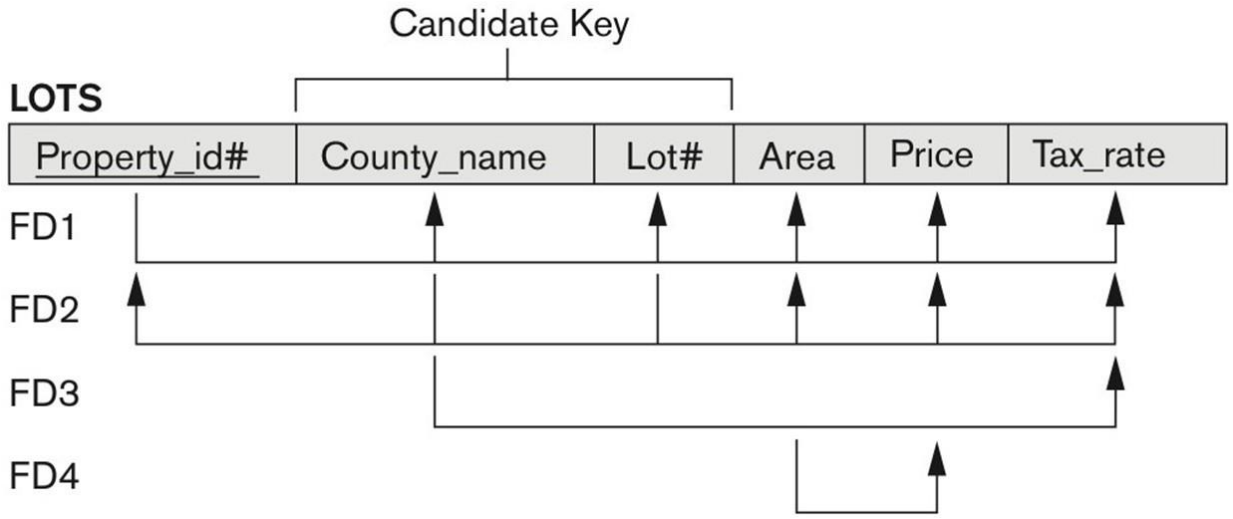
- EMP_DEPT is in **2NF**, but not in **3NF**.
  - 3NF is **violated** by **transitive dependencies**.
    - *Dmgr_ssn* & *Dname* are **transitively dependent** on *Ssn* through *Dnumber*.




- **Normalizing** EMP_DEPT into **3NF**:
  - **Decompose** relation into **3NF relations** where no **transitive dependencies** exist.
    - ED1 and ED2 represent independent facts about employees & departments, both of which are entities on their own.
    - By applying natural join on ED1 and ED2 we can recover original EMP_DEPT with no spurious tuples.

# NORMALIZATION EXAMPLE (1)

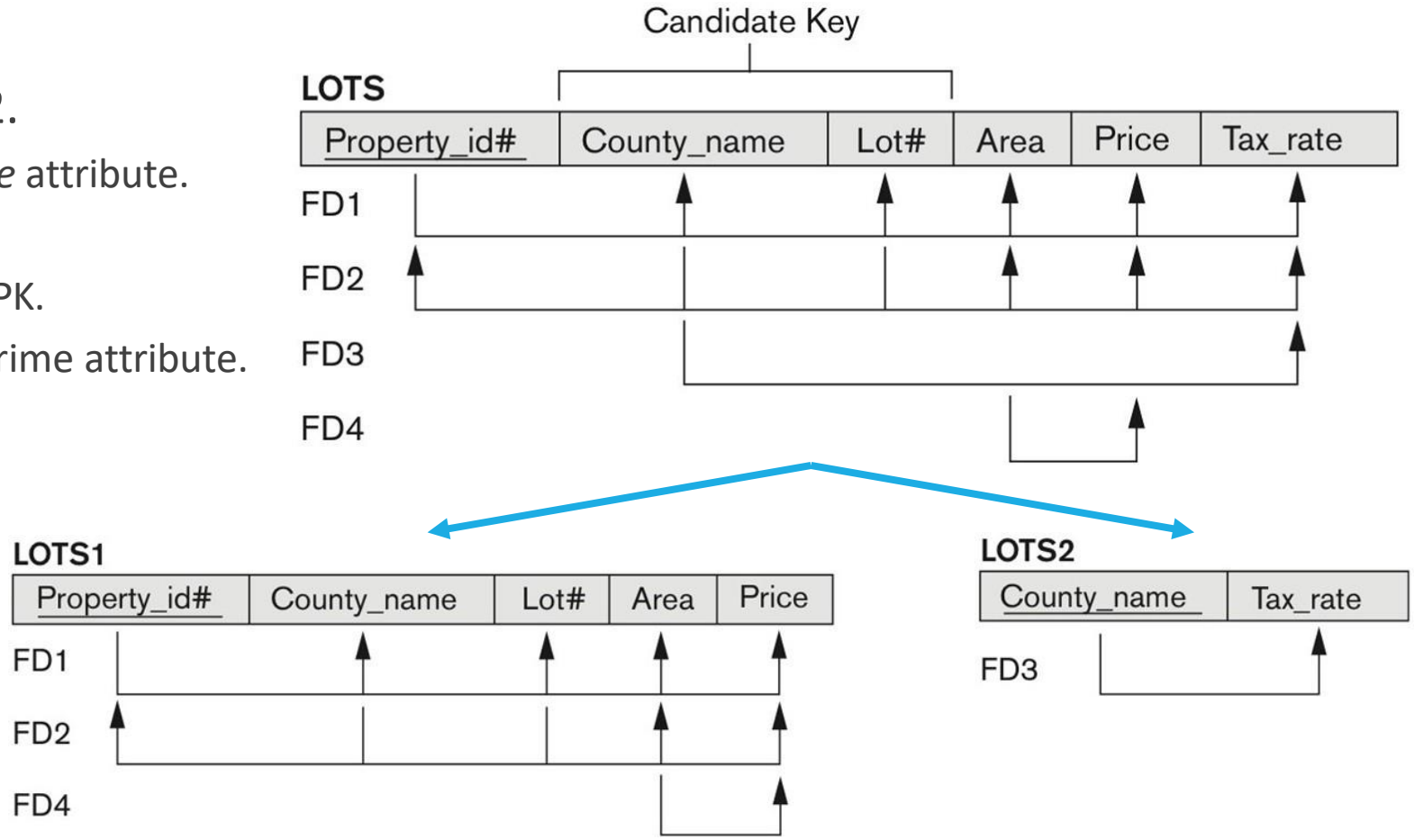
- LOTS relation describes land for sale in various counties of a state.
  - **Candidate keys**:
    - *Property_id#* (chosen as primary).
      - Property id numbers are unique across counties.
    - {*County_name*, *Lot#*}
      - Lot number is unique only within county.
  - **FDs**:
    - FD1 and FD2 are based on candidate keys.
    - FD3: *County_name -> Tax_rate*
      - Tax rate is fixed for a given county.
    - FD4: *Area -> Price*
      - Price of a lot is determined by the area regardless of which county it is.




- What is the highest NF of this relation?

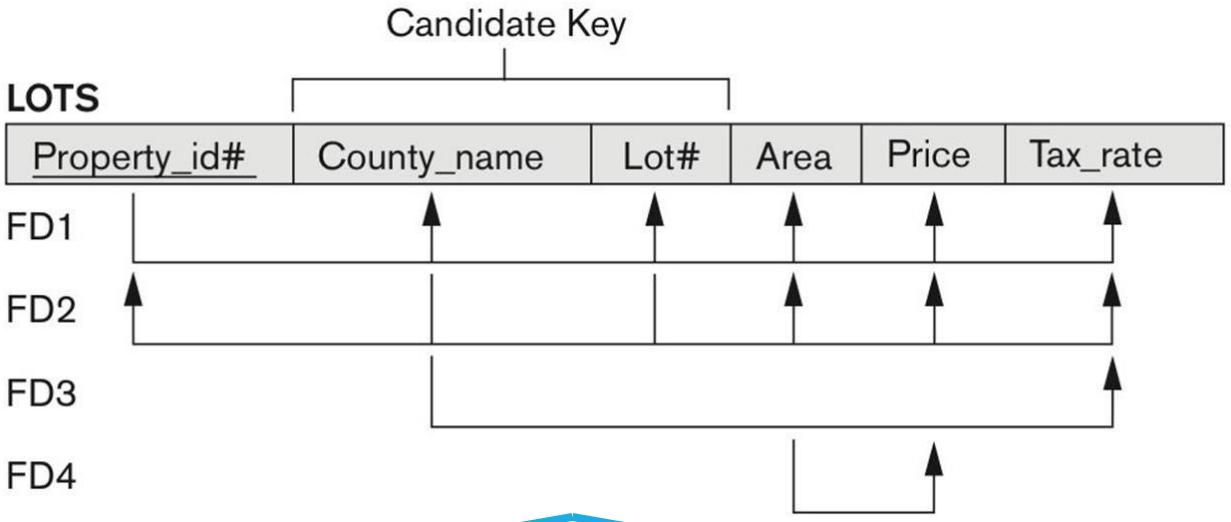
# NORMALIZATION EXAMPLE (2)

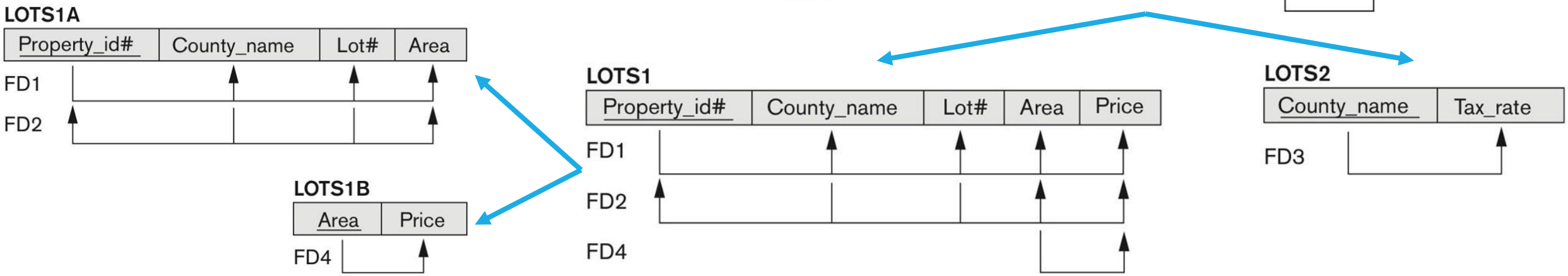
- FD3: *County_name* -> *Tax_rate* **violates 2NF**.
  - *Tax_rate* is **partially dependent** on the **candidate key** {*County_name*, *Lot#*}.
- To **normalize** LOTS into **2NF**:
  - **Decompose** it into LOTS1 and LOTS2.
    - LOTS1 is constructed by removing *Tax_rate* attribute.
    - LOTS2 is constructed as a FD3.
      - LHS of FD3 (*Country_name*) becomes a PK.
      - RHS of FD3 (*Tax_rate*) becomes a non-prime attribute.



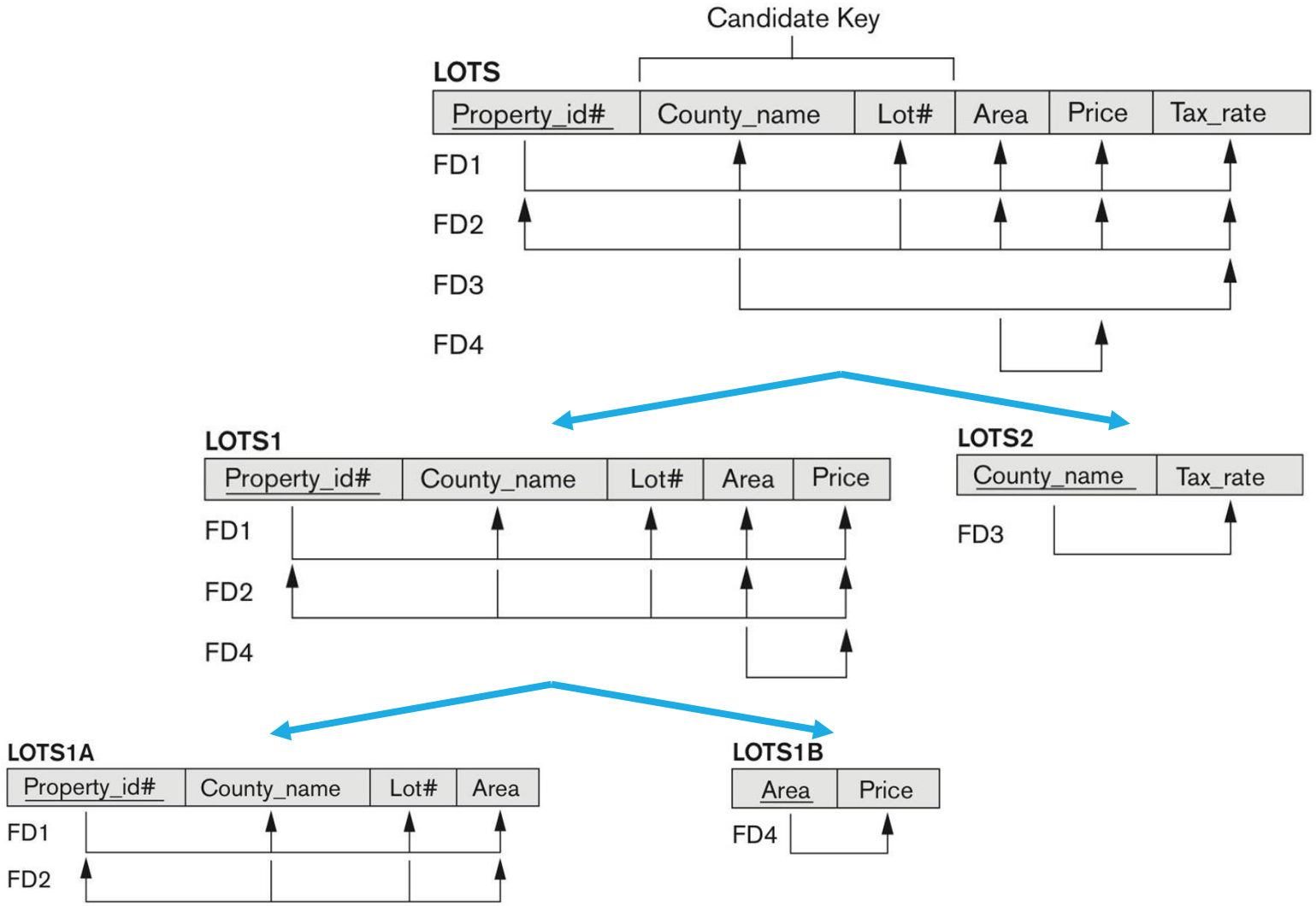
# NORMALIZATION EXAMPLE (3)

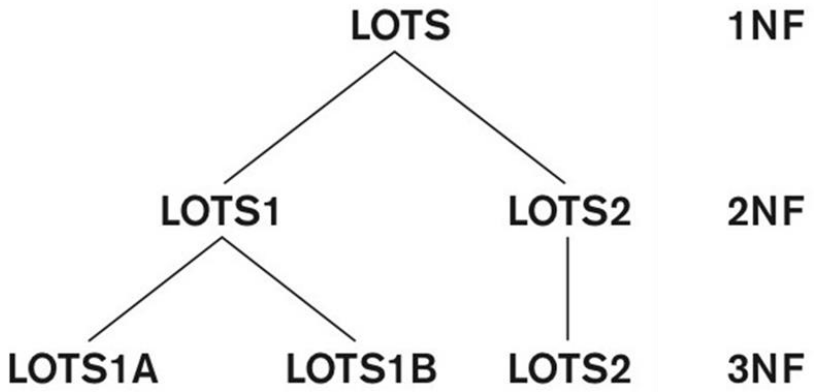
- FD4 in LOTS1: *Area* -> *Price* **violates 3NF**.
  - Price is **transitively dependent** on each of the **candidate keys** of LOTS1 through **non-prime attribute** *Area*.
- To **normalize** LOTS1 into **3NF**:
  - **Decompose** it into LOTS1A and LOTS1B.
    - LOTS1A is constructed by removing *Price* attribute.
    - LOTS1B is constructed as a FD4.
      - LHS of FD4 (*Area*) becomes a PK.
      - RHS of FD4 (*Price*) becomes a non-prime attribute.



# NORMALIZATION EXAMPLE (4)



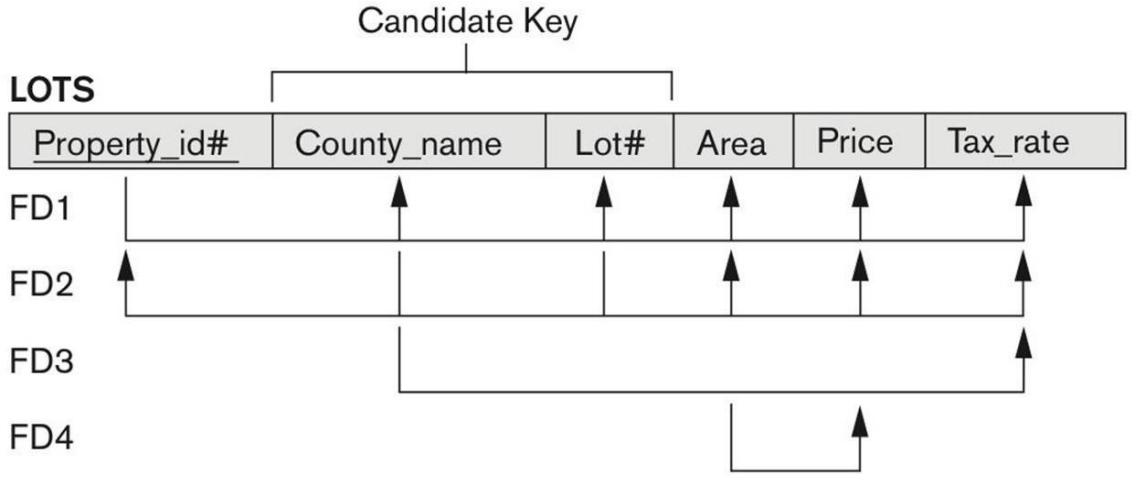
# NORMAL FORMS SUMMARY

Summary of Normal Forms Based on Primary Keys and Corresponding Normalization

| Normal Form | Test | Remedy (Normalization) |
|---|---|---|
| First (1NF) | Relation should have no multivalued attributes or nested relations. | Form new relations for each multivalued attribute or nested relation. |
| Second (2NF) | For relations where primary key contains multiple attributes, no nonkey attribute should be functionally dependent on a part of the primary key. | Decompose and set up a new relation for each partial key with its dependent attribute(s). Make sure to keep a relation with the original primary key and any attributes that are fully functionally dependent on it. |
| Third (3NF) | Relation should not have a nonkey attribute functionally determined by another nonkey attribute (or by a set of nonkey attributes). That is, there should be no transitive dependency of a nonkey attribute on the primary key. | Decompose and set up a relation that includes the nonkey attribute(s) that functionally determine(s) other nonkey attribute(s). |

# THIRD NORMAL FORM: GENERAL DEFINITION

- **Third normal** form **general definition** (based on *multiple* **candidate keys)**:
  - **Relation schema** R is in **3NF** if whenever a FD X -> A **holds** in R, then either:
    - X is a **superkey** of R, or
    - A is a **prime attribute** of R.
- **Third normal** form **alternative definition**:
  - **Relation schema** R is in **3NF** if every **nonprime attribute** of R meets following **conditions**:
    - It is **fully functionally dependent** on every **key** of R.
    - It is **non-transitively dependent** on every **key** of R.



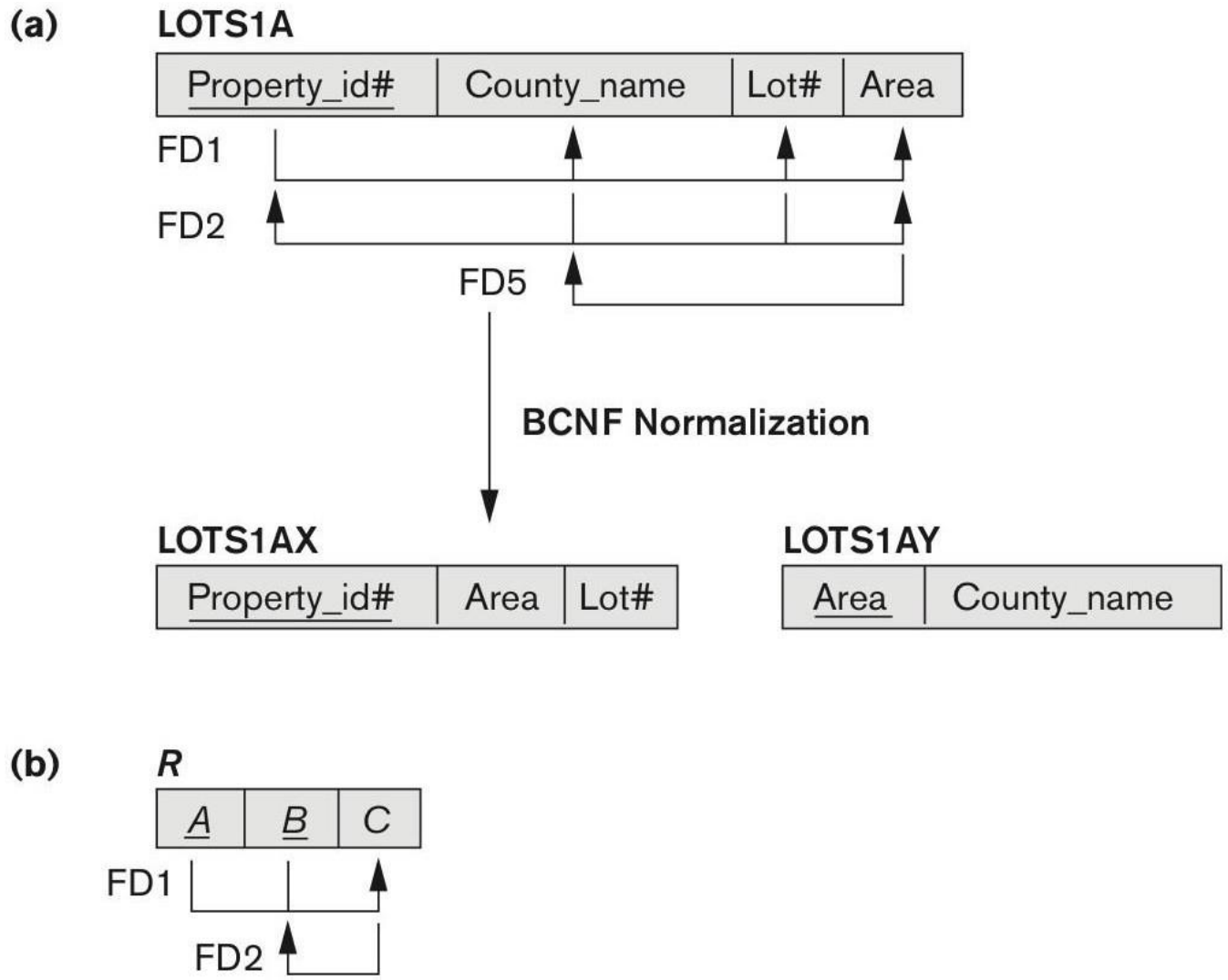

FD3 and FD4 violate 3NF general definition

# BOYCE-CODD NORMAL FORM (BCNF)

- **Boyce-Codd normal form** (**BCNF**) definition:
  - **Relation schema** R is in **BCNF** if whenever a FD X -> A **holds** in R, then X is a **superkey** of R.
- Each normal form is **strictly stronger** than the previous one:
  - Every **2NF** relation is in **1NF.**
  - Every **3NF** relation is in **2NF**.
  - Every **BCNF** relation is in **3NF**.
- The **goal** of **normalization** process is to **decompose** each relation into **BCNF** (or **3NF**).

# BOYCE-CODD NORMAL FORM: EXAMPLE (1)

- **Example (a):**
  - FD5 is added to LOTS1A relation.
    - FD5: Area -> County_name.
  - Relation LOTS1A is in 3NF, but not in BCNF.
    - Area is not superkey of LOTS1A.
  - Relation LOTS1A is decomposed into two:
    - LOTS1AX & LOTS1AY.
    - Both in BCNF.
- **Example (b):**
  - General case of relation in 3NF, but not in BCNF.




LOTS1A relation is in 3NF, but not in BCNF

# BOYCE-CODD NORMAL FORM: EXAMPLE (2)

- **Relation** TEACH has two FDs:
  - **FD1:** {Student, Course} -> Instructor
  - **FD2:** Instructor -> Course
- *{Student, Course}* is a **candidate key**.
- **Relation** TEACH is in **3NF**, but not **BCNF**. *Why?*
- Three **alternative decompositions** of TEACH:
  - **R1**(Student, Instructor) & **R2**(Student, Course).
  - **R1**(Course, Instructor) & **R2**(Course, Student).
  - **R1**(Instructor, Course) & **R2**(Instructor, Student).
- **Third alternative** is more **favorable**, since it preserves **non-additive join** property.
  - No **spurious** tuples are generated.

**TEACH**

| Student | Course | Instructor |
|---|---|---|
| Narayan | Database | Mark |
| Smith | Database | Navathe |
| Smith | Operating Systems | Ammar |
| Smith | Theory | Schulman |
| Wallace | Database | Mark |
| Wallace | Operating Systems | Ahamad |
| Wong | Database | Omiecinski |
| Zelaya | Database | Navathe |
| Narayan | Operating Systems | Ammar |

TEACH relation is in 3NF, but not in BCNF

# SUMMARY

- Informal relation database design guidelines.
- Functional dependencies.
- Normalization process.
- Normal forms and decomposition.
  - First normal from.
  - Second normal form
  - Third normal form.
  - Boyce-Codd normal form.