Chapter 5
Relational-Database Design

Department: Computer
Course: DBMS
Faculty: Sana Shaikh

Functional Dependencies and Normalization for Relational Databases

Informal Design Guidelines for Relation Schemas

1.1 Semantics of the Relation Attributes

GUIDELINE 1: Informally, each tuple in a relation should represent one entity or relationship instance. (Applies to individual relations and their attributes).

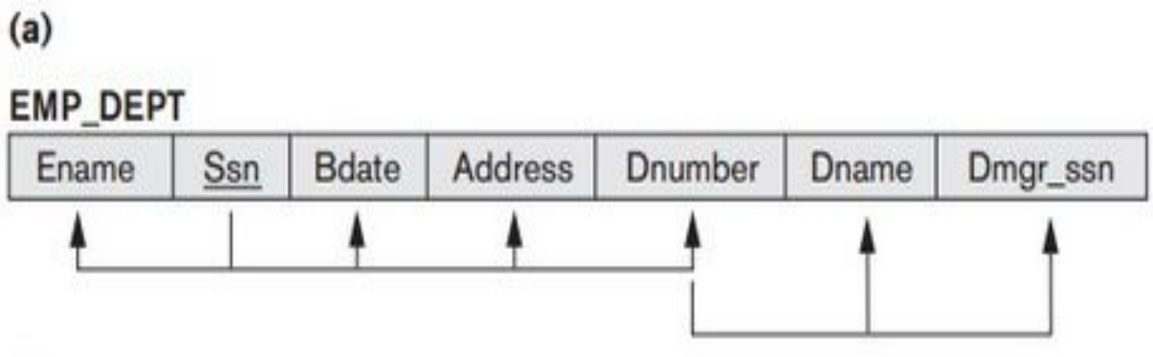
Informal Design Guidelines for Relation Schemas

1.1 Semantics of the Relation Attributes

GUIDELINE 1: Informally, each tuple in a relation should represent one entity or relationship instance. (Applies to individual relations and their attributes).

- Attributes of different entities (EMPLOYEEs, DEPARTMENTs, PROJECTs) should not be mixed in the same relation
- Only foreign keys should be used to refer to other entities
- Entity and relationship attributes should be kept apart as much as possible.

Explain this relation schema



A tuple in the EMP_DEPT relation schema in Figure 15.3(a) represents a single employee but includes additional information—namely, the name (Dname) of the department for which the employee works and the Social Security number (Dmgr_ssn) of the department manager.

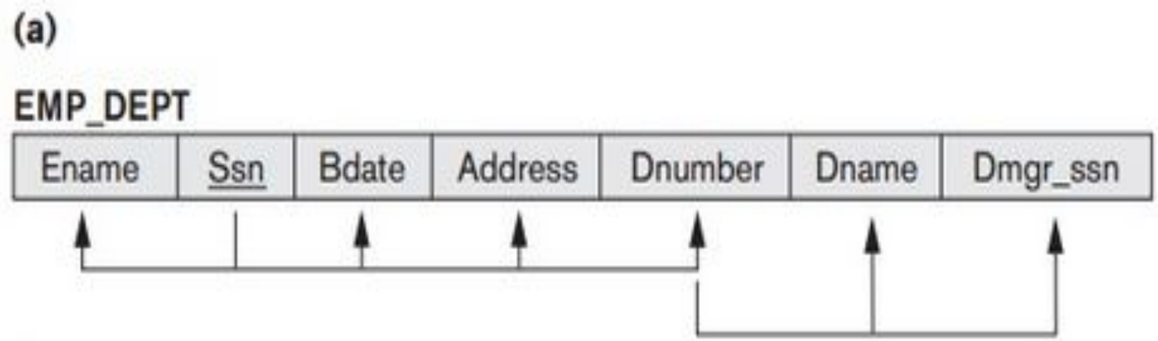
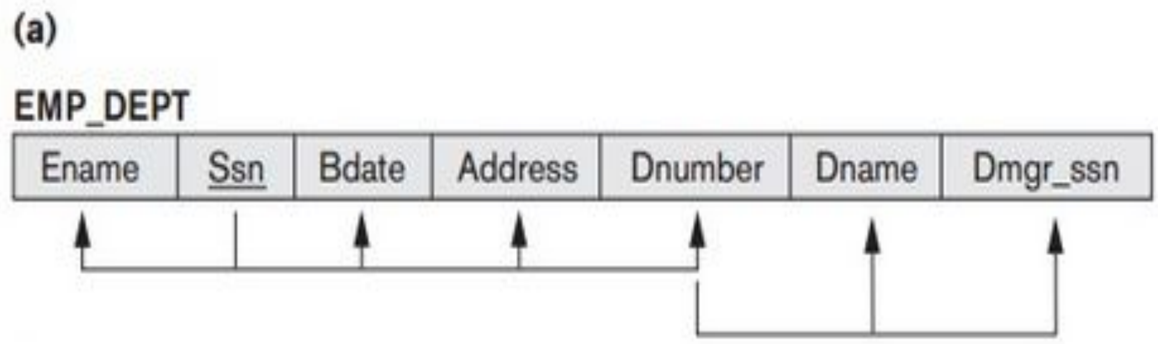


Figure 15.3
Two relation schemas suffering from update anomalies. (a) EMP_DEPT and (b) EMP_PROJ.

A tuple in the EMP_DEPT relation schema in Figure 15.3(a) represents a single employee but includes additional information—namely, the name (Dname) of the department for which the employee works and the Social Security number (Dmgr_ssn) of the department manager.

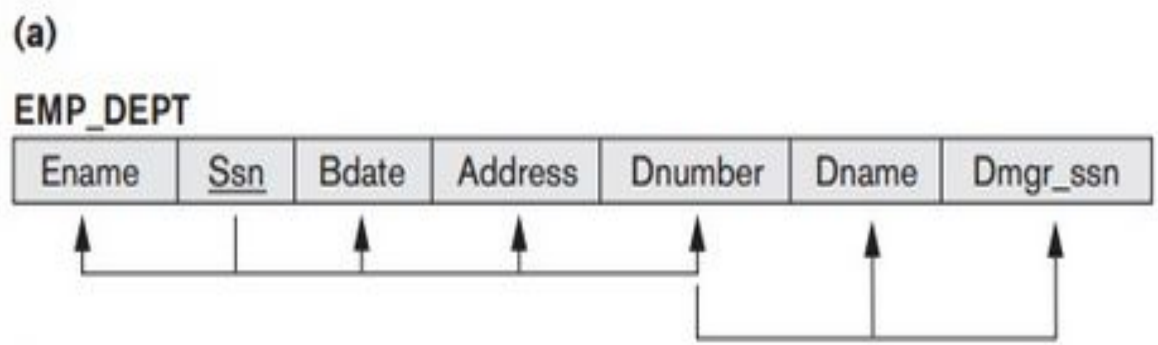
Figure 15.3
Two relation schemas suffering from update anomalies. (a) EMP_DEPT and (b) EMP_PROJ.



Although there is nothing wrong logically with these two relations, **they violate Guideline 1** by mixing attributes from distinct real-world entities: EMP_DEPT mixes attributes of employees and departments

A tuple in the EMP_DEPT relation schema in Figure 15.3(a) represents a single employee but includes additional information—namely, the name (Dname) of the department for which the employee works and the Social Security number (Dmgr_ssn) of the department manager.

Figure 15.3
Two relation schemas suffering from update anomalies. (a) EMP_DEPT and (b) EMP_PROJ.

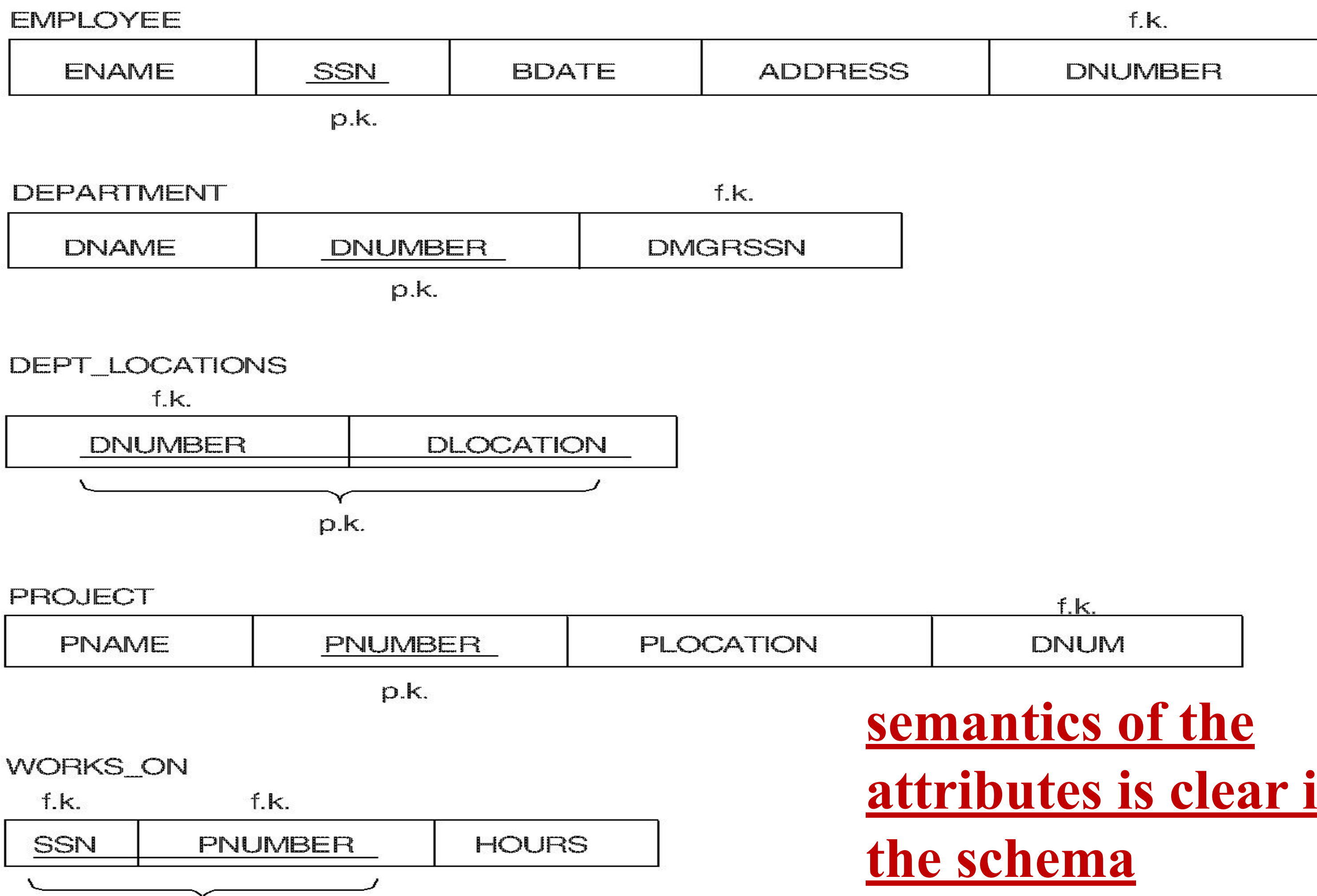


Although there is nothing wrong logically with these two relations, **they violate Guideline 1** by mixing attributes from distinct real-world entities: EMP_DEPT mixes attributes of employees and departments

In EMP_DEPT, the attribute values pertaining to a particular department (Dnumber, Dname, Dmgr_ssn) are repeated for *every employee who works for that department*

A simplified COMPANY relational database schema

Figure 14.1 Simplified version of the COMPANY relational database schema.



semantics of the attributes is clear in the schema

Informal Design Guidelines for Relation Schemas

1.2 Reducing the Redundant Information in Tuples and Update Anomalies

- Mixing attributes of multiple entities may cause problems
- Information is stored redundantly wasting storage
- Problems with various anomalies
  - Insertion anomalies
  - Deletion anomalies
  - Modification anomalies

Example States for EMP_DEPT and EMP_PROJ

Figure 14.4 Example relations for the schemas in Figure 14.3 that result from applying NATURAL JOIN to the relations in Figure 14.2. These may be stored as base relations for performance reasons.

EMP_DEPT

ENAME	SSN	BDATE	ADDRESS	DNUMBER	DNAME	DMGRSSN
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

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

Redundancy

Example States for EMP_DEPT and EMP_PROJ

Figure 14.4 Example relations for the schemas in Figure 14.3 that result from applying NATURAL JOIN to the relations in Figure 14.2. These may be stored as base relations for performance reasons.

EMP_DEPT

ENAME	SSN	BDATE	ADDRESS	DNUMBER	DNAME	DMGRSSN
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

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	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

Redundancy

EXAMPLE OF AN UPDATE ANOMALY (1)

- **Update Anomaly:** Changing the name of project number P10 from “Computerization” to “Comps-section”

EXAMPLE OF AN UPDATE ANOMALY (1)

- **Update Anomaly:** Changing the name of project number P10 from "Computerization" to "Comps-section"

may cause this update to be made for all employees working on project P10.

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 AN UPDATE ANOMALY (2)

- **Insert Anomaly:**

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 AN UPDATE ANOMALY (2)

- **Insert Anomaly:**

Cannot insert a project unless an employee is assigned to .

Inversely **- Cannot insert an employee unless an he/she is assigned to a project.**

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 AN UPDATE ANOMALY (2)

- **Delete Anomaly:**

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 AN UPDATE ANOMALY (2)

- **Delete Anomaly:**

When a project is deleted, it will result in deleting all the employees who work on that project. Alternately, if an employee is the sole employee on a project, deleting that employee would result in deleting the corresponding project.

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

Guideline to Redundant Information in Tuples and Update Anomalies

- **GUIDELINE 2:**
  - **Design a schema that does not suffer from the insertion, deletion and update anomalies. If there are any present, then note them so that applications can be made to take them into account**
  - **Reducing the Redundant Information in Tuples**

1.3 Null Values in Tuples

GUIDELINE 3: Relations should be designed such that their tuples will have as few NULL values as possible

- Reasons for nulls:
  - attribute not applicable or invalid
  - attribute value unknown (may exist)
  - value known to exist, but unavailable

1.3 Null Values in Tuples

GUIDELINE 3: Relations should be designed such that their tuples will have as few NULL values as possible

- Reasons for nulls:
  - attribute not applicable or invalid
  - attribute value unknown (may exist)
  - value known to exist, but unavailable
- Attributes that are NULL frequently could be placed in separate relations (with the primary key)

1.3 Null Values in Tuples

GUIDELINE 3: Relations should be designed such that their tuples will have as few NULL values as possible

- Reasons for nulls:
  - attribute not applicable or invalid
  - attribute value unknown (may exist)
  - value known to exist, but unavailable
- Attributes that are NULL frequently could be placed in separate relations (with the primary key)

Example: 15% Employees have individual Office

1.3 Null Values in Tuples

GUIDELINE 3: Relations should be designed such that their tuples will have as few NULL values as possible

- Reasons for nulls:
  - attribute not applicable or invalid
  - attribute value unknown (may exist)
  - value known to exist, but unavailable
- Attributes that are NULL frequently could be placed in separate relations (with the primary key)

Example: 15% Employees have individual Office

Solution : Instead of adding one attribute Office_number in Employee relation, create new relation as follows:

EMP_OFFICES(Essn, Office_number)

1.4 Spurious Tuples

- Bad designs for a relational database may result in erroneous results for certain JOIN operations
- The "lossless join" property is used to guarantee meaningful results for join operations

GUIDELINE 4: The relations should be designed to satisfy the lossless join condition. No spurious tuples should be generated by doing a natural-join of any relations.

1.4 Spurious Tuples Cont..

(a)

EMP_LOCS

Ename	Plocation

P.K.

EMP_PROJ1

Ssn	Pnumber	Hours	Pname	Plocation

P.K.

Figure 15.5
Particularly poor design for the EMP_PROJ relation in Figure 15.3(b). (a) The two relation schemas EMP_LOCS and EMP_PROJ1. (b) The result of projecting the extension of EMP_PROJ from Figure 15.4 onto the relations EMP_LOCS and EMP_PROJ1.

(b)

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

1.4 Spurious Tuples Cont..

	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.

Figure 15.6
Result of applying NATURAL JOIN to the tuples above the dashed lines in EMP_PROJ1 and EMP_LOCS of Figure 15.5. Generated spurious tuples are marked by asterisks.

Guideline 4

- Design relation schemas so that they can be joined with equality conditions on attributes that are appropriately related (primary key, foreign key) pairs in a way that guarantees that no spurious tuples are generated.

- Avoid relations that contain matching attributes that are not (foreign key, primary key) combinations because joining on such attributes may produce spurious tuples.

2.1 Functional Dependencies

- **Functional Dependency (FD)** is a constraint that determines the relation of one attribute to another attribute in a Database Management System (DBMS).
- Functional Dependency helps to maintain the quality of data in the database.
- It plays a vital role to find the difference between good and bad database design.
- A functional dependency is denoted by an arrow "→". The functional dependency of X on Y is represented by X → Y.

Functional Dependencies

Employee number	Employee Name	Salary	City
1	Dana	50000	San Francisco
2	Francis	38000	London
3	Andrew	25000	Tokyo

In this example, if we know the value of Employee number, we can obtain Employee Name, city, salary, etc.

By this, we can say that the city, Employee Name, and salary are functionally dependent on Employee number.

Employee number → {Employee Name, Salary, City}

Examples of FD constraints

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

Examples of FD constraints

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

- social security number determines employee name

  SSN -> ENAME
- project number determines project name and location

  PNUMBER -> {PNAME, PLOCATION}
- employee ssn and project number determines the hours per week that the employee works on the project

  {SSN, PNUMBER} -> HOURS

Examples of FD constraints

- An FD is a property of the attributes in the schema R
- The constraint must hold on *every relation instance* r(R)
- If K is a key of R, then K functionally determines all attributes in R (since we never have two distinct tuples with t1[K]=t2[K])

Examples of FD constraints

TEACH

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

Examples of FD constraints

TEACH

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

A relation state of TEACH with a *possible* functional dependency TEXT → COURSE. However, TEACHER → COURSE is ruled out.

Examples of FD constraints

Figure 15.8
A relation *R*(A, B, C, D) with its extension.

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

Following FD may hold

Following FD DO NOT hold

Examples of FD constraints

Figure 15.8
A relation *R*(A, B, C, D) with its extension.

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

<u>Following FD may hold</u>

B → C;

C → B;

{A,B} → D;

{C, D} → B;

<u>Following FD DO NOT hold</u>

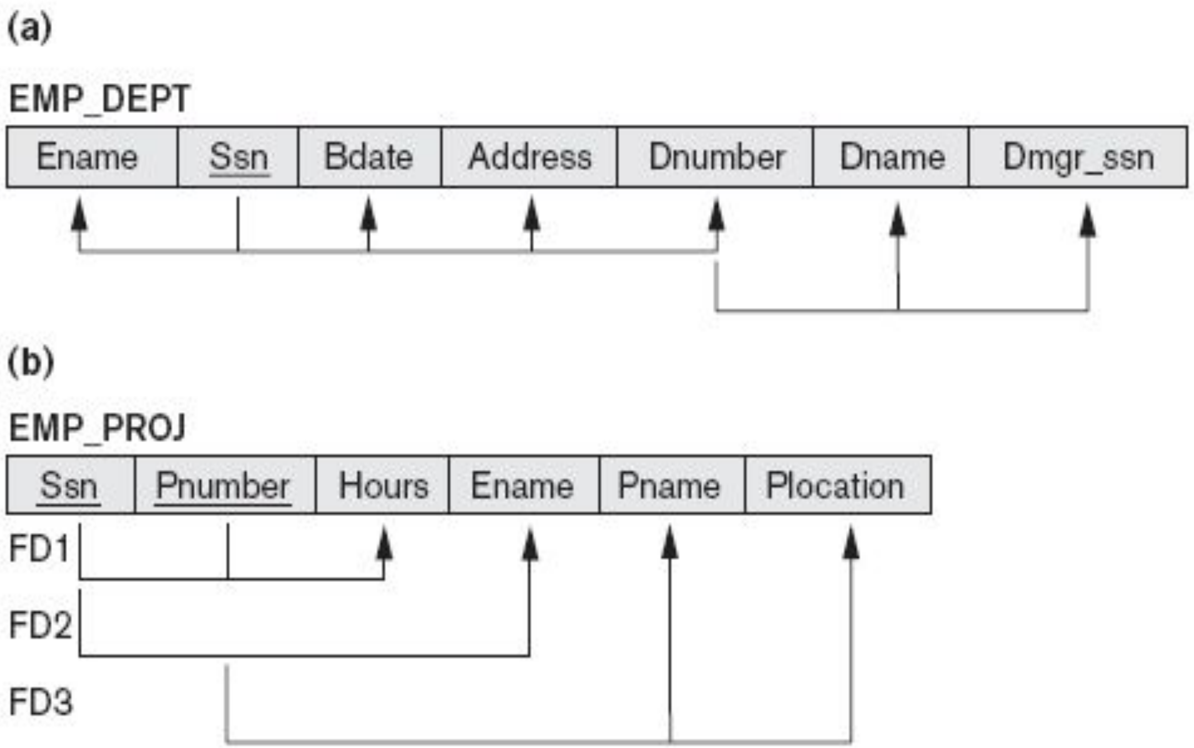
A → B;

B → A;

D → C;

•Diagrammatic notion of displaying FD

Figure 15.3
Two relation schemas suffering from update anomalies. (a) EMP_DEPT and (b) EMP_PROJ.



Test Yourself

Q.1 Which functional dependency holds in given relation R (A, B, C) and why?

1. AB $\rightarrow$ C && C $\rightarrow$ B
2. BC $\rightarrow$ A && B $\rightarrow$ C
3. BC $\rightarrow$ A && A $\rightarrow$ C
4. AC $\rightarrow$ B && B $\rightarrow$ C

A	B	C
7	1	8
7	2	5
7	3	5
5	8	8

Test Yourself

Q.1 Which functional dependency holds in given relation R (A, B, C) and why?

1. $AB \rightarrow C$ && $C \rightarrow B$
2. $BC \rightarrow A$ && $B \rightarrow C$
3. $BC \rightarrow A$ && $A \rightarrow C$
4. $AC \rightarrow B$ && $B \rightarrow C$

A	B	C
7	1	8
7	2	5
7	3	5
5	8	8

Solution:

$AB \rightarrow C$ satisfies condition1 (all the values of AB are unique), but $C \rightarrow B$ does not satisfy any of the three conditions (C has not unique values, B has not all same values and the value of C is repeated two times (8, 5) then the value of B is not repeated), so **option 1 is incorrect**.

$BC \rightarrow A$ satisfies condition1, and $B \rightarrow C$ also satisfies condition1, so **option 2 is correct**.

$BC \rightarrow A$ satisfies condition1, but $A \rightarrow C$ does not satisfy any of the three conditions, so **option 3 is incorrect**.

$AC \rightarrow B$ does not satisfy any of the three conditions, and $B \rightarrow C$ satisfies condition1, so **option 4 is incorrect**.

Since $BC \rightarrow A$ && $B \rightarrow C$ holds in above relation, **option 2 will be the answer.**

Armstrong's Axiom: Inference Rule for FD

- IR1 (reflexive rule)1: If $Y \subset X$, then $X \rightarrow Y$.
- IR2 (augmentation rule)2: $\{XZ \rightarrow YZ\} \models X \rightarrow Y$.
- IR3 (transitive rule): $\{X \rightarrow Y, Y \rightarrow Z\} \models X \rightarrow Z$.
- IR4 (decomposition, or projective, rule): $\{X \rightarrow YZ\} \models X \rightarrow Y$ & $X \rightarrow Z$
- IR5 (union, or additive, rule): $\{X \rightarrow Y, X \rightarrow Z\} \models X \rightarrow YZ$.
- IR6 (pseudotransitive rule): $\{X \rightarrow Y, \mathrm{W}Y \rightarrow Z\} \models X\mathrm{W} \rightarrow Z$.

Example : Consider relation R = (A,B,C,D,E,F) having set of FD's

A → B　　A→ C　　BC →D

B → E　　BC → F　　AC → F

Calculate some members of Axioms as given below:

1. A→ E　　2. BC → DF

3. AC → D　　4. AC → DF

Solution :

Example : Consider relation R = (A,B,C,D,E,F) having set of FD's

A → B   A→ C   BC →D

B → E   BC → F   AC → F

Calculate some members of Axioms as given below:

1. A→ E   2. BC → DF

3. AC → D   4. AC → DF

Solution :

1. A → E

As A→ B and B → E

So , using Transitive rule,

∴ A → E

Example : **Consider relation R = (A,B,C,D,E,F) having set of FD's**

A → B   A→ C   BC →D

B → E   BC → F   AC → F

Calculate some members of Axioms as given below:

1. A→ E   2. BC → DF

3. AC → D   4. AC → DF

Solution :

1. A → E

As A→ B and B → E

So , using Transitive rule,

∴ A → E

2. BC → DF

AS BC → D and BC → F

So , using union rule,

∴ BC → DF

Example : **Consider relation R = (A,B,C,D,E,F) having set of FD's**

$A \rightarrow B$ $A \rightarrow C$ $BC \rightarrow D$

$B \rightarrow E$ $BC \rightarrow F$ $AC \rightarrow F$

Calculate some members of Axioms as given below:

1. $A \rightarrow E$ 2. $BC \rightarrow DF$

3. $AC \rightarrow D$ 4. $AC \rightarrow DF$

Solution :

1. $A \rightarrow E$

As $A \rightarrow B$ and $B \rightarrow E$

So , using Transitive rule,

$\therefore A \rightarrow E$

2. $BC \rightarrow DF$

AS $BC \rightarrow D$ and $BC \rightarrow F$

So , using union rule,

$\therefore BC \rightarrow DF$

3. $AC \rightarrow D$

As $A \rightarrow B$ and $BC \rightarrow D$

So , using pseudo transitive rule,

$\therefore AC \rightarrow D$

Example : **Consider relation R = (A,B,C,D,E,F) having set of FD's**

A → B A→ C BC →D

B → E BC → F AC → F

Calculate some members of Axioms as given below:

1. A→ E 2. BC → DF

3. AC → D 4. AC → DF

Solution :

1. A → E

As A→ B and B → E

So , using Transitive rule,

∴ A → E

2. BC → DF

AS BC → D and BC → F

So , using union rule,

∴ BC → DF

3. AC → D

As A→ B and BC → D

So , using pseudo transitive rule,

∴ AC→ D

4. AC → DF

From Solution 3, AC → D

and AC → F

So , using union rule,

∴ AC → DF

Decomposition

Introduction:

- If a relation is not in the normal form and we wish the relation to be normalized so that some of the anomalies can be eliminated, it is necessary to decompose the relation in two or more relations.

- **Decomposition** of relation R into a set of relations R1, R2….Rn,
  Such that : **$R = R1 \cup R2 \cup R3 \cup \ldots . RN$**

- ***This is the process of dividing one table into multiple tables using projection operator.***

Vertical Decomposition:

- We may decompose tables into vertical segments.
- Vertical segmentation is done with the help of projection operator.
- By taking projection of original table we can create multiple vertically fragmented tables.

Vertical Decomposition:

- We may decompose tables into vertical segments.
- Vertical segmentation is done with the help of projection operator.
- By taking projection of original table we can create multiple vertically fragmented tables.

Original table

ID	Ename	Class
1	Sana	SE
2	Savita	BE
3	Mahesh	BE
4	Amiya	TE

Decomposed tables

Vertical Decomposition:

- We may decompose tables into vertical segments.
- Vertical segmentation is done with the help of projection operator.
- By taking projection of original table we can create multiple vertically fragmented tables.

Original table

ID	Ename	Class
1	Sana	SE
2	Savita	BE
3	Mahesh	BE
4	Amiya	TE

Decomposed tables

ID	Ename
1	Sana
2	Savita
3	Mahesh
4	Amiya

ID	Class
1	SE
2	BE
3	BE
4	TE

Vertical Decomposition (Example):

Dept_Student (DID, Dname, SID, Sname, Location)

From the above table, we can create new **Dept and Student** tables by decomposing the **Dept_Student**.

 Apply projection operator on the Dept_Student .

Dept $\pi_{\text{DID, Dname, Location}}$ (Dept_Student)

Student $\pi_{\text{SID, Sname, DID}}$ (Dept_Student)

As DID is common between above two tables so we can combine tables to regain Dept_Student schema.

This is example of Lossless-join decomposition

What is Lossy-join decomposition ???

Dept_Student (DID, Dname, SID, Sname, Location)

Dept $\pi_{\text{DID, Dname, Location}}$ (Dept_Student)

Student $\pi_{\text{SID, Sname}}$ (Dept_Student)

As Dept ∩ Student = φ

This is example of Lossy-join decomposition

Desirable Properties of Decomposition:

1.Lossless join decomposition

- decomposition must be lossless so that we do not lose any information from the relation that is decomposed.

- For join to be lossless we need to go for following steps:
  a. Let R1 and R2 form decomposition of relation R as R1 and R2 are both set of attributes from R.
  b. Decompose the relation Dept_Student (DID, Dname, SID, Sname, Location) into :

  Dept $\pi_{\text{DID, Dname, Location}}$ (Dept_Student)
  Student $\pi_{\text{SID, Sname, DID}}$ (Dept_Student)

  c. The common attribute must be a key for one of the relation for decomposition to be lossless. $R1 \cap R2 \neq \varphi$ there must not be null.

  (NOTE: You are joining a Primary Key and Foreign Key of Table.)

Desirable Properties of Decomposition:

2. Dependency Preservation

- If $X \rightarrow Y$ holds that we know that the two (sets) attributes are closely related to functionally dependent and it would be useful if both attributes in the same relation so that the dependency can be checked easily.
- We have to maintained Functional Dependency.
- Consider relation R(X,Y,Z,W) that has the following dependencies:

$$X \rightarrow Y \quad \text{and} \quad X \rightarrow Z$$

If we decompose the above relation into R1(X, Y) and R2 (Y,Z,W) the dependency $X \rightarrow Z$ is not preserved.

Desirable Properties of Decomposition:

3. Repetition of information

- Decomposition that we have done should not suffer from any repetition of information problem.
- It is desirable not to have any redundancy in the database.
- **This property may be achieved by Normalization Process.**

Normalization

While designing a database out of an entity–relationship model, the main problem existing in that “raw” database is redundancy.

A redundancy creates several problems like the following:

Normalization

While designing a database out of an entity–relationship model, the main problem existing in that "raw" database is redundancy.

A redundancy creates several problems like the following:

- Extra storage space
- large amount of disk space.
- Entering same data more than once during data insertion.
- Deleting data from more than one place during deletion.
- Modifying data in more than one place.
- Anomalies may occur in the database if insertion, deletion, modification etc are not done properly.

"It creates inconsistency and unreliability in the database."

To solve this problem, the "raw" database needs to be normalized.

Normalization of Relations

- **Normalization**: The process of decomposing unsatisfactory "bad" relations by breaking up their attributes into smaller relations

- This is a step by step process of removing different kinds of redundancy and anomaly at each step.

- At each step a specific rule is followed to remove specific kind of impurity in order to give the database a slim and clean look.

- **Normal form**: Condition using keys and FDs of a relation to certify whether a relation schema is in a particular normal form

Practical Use of Normal Forms

- **Normalization** is carried out in practice so that the resulting designs are of high quality and meet the desirable properties
  - Normalization is the process of organizing the data in the database. Normalization is used to minimize the redundancy from a relation or set of relations.
  - It is also used to eliminate the undesirable characteristics like Insertion, Update and Deletion Anomalies.
  - Normalization divides the larger table into the smaller table and links them using relationship.
  - The normal form is used to reduce redundancy from the database table.
- The database designers ***need not*** normalize to the highest possible normal form. (usually up to 3NF, BCNF or 4NF)

Un-Normalized Form (UNF)

If a table contains non-atomic values at each row, it is said to be in UNF. An **atomic value** is something that can not be further decomposed. A **non-atomic value**, as the name suggests, can be further decomposed and simplified. Consider the following table:

Un-Normalized Form (UNF)

If a table contains non-atomic values at each row, it is said to be in UNF. An **atomic value** is something that can not be further decomposed. A **non-atomic value**, as the name suggests, can be further decomposed and simplified. Consider the following table:

FULL NAMES	PHYSICAL ADDRESS	MOVIES RENTED	SALUTATION
Janet Jones	First Street Plot No 4	Pirates of the Caribbean, Clash of the Titans	Ms.
Robert Phil	3rd Street 34	Forgetting Sarah Marshal, Daddy's Little Girls	Mr.
Robert Phil	5th Avenue	Clash of the Titans	Mr.

First Normal Form

- Disallows composite attributes, multivalued attributes, and **nested relations**; attributes whose values *for an individual tuple* are non-atomic

Normalization of Multi-valued attribute into 1NF

Figure 14.8 Normalization into 1NF. (a) Relation schema that is not in 1NF. (b) Example relation instance. (c) 1NF relation with redundancy.

(a)

DEPARTMENT

DNAME	<u>DNUMBER</u>	DMGRSSN	DLOCATIONS

(b)

DEPARTMENT

DNAME	<u>DNUMBER</u>	DMGRSSN	DLOCATIONS
Research	5	333445555	{Bellaire, Sugarland, Houston}
Administration	4	987654321	{Stafford}
Headquarters	1	888665555	{Houston}

(

Normalization of Multi-valued attribute into 1NF

Figure 14.8 Normalization into 1NF. (a) Relation schema that is not in 1NF. (b) Example relation instance. (c) 1NF relation with redundancy.

(a)

DEPARTMENT

DNAME	<u>DNUMBER</u>	DMGRSSN	DLOCATIONS

(b)

DEPARTMENT

DNAME	<u>DNUMBER</u>	DMGRSSN	DLOCATIONS
Research	5	333445555	{Bellaire, Sugarland, Houston}
Administration	4	987654321	{Stafford}
Headquarters	1	888665555	{Houston}

Problem ???

(c)

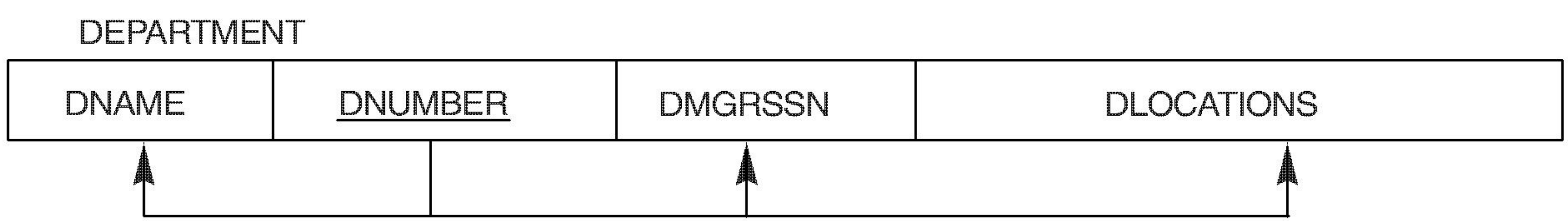
DEPARTMENT

DNAME	DNUMBER	DMGRSSN	DLOCATION
Research	5	333445555	Bellaire
Research	5	333445555	Sugarland
Research	5	333445555	Houston
Administration	4	987654321	Stafford
Headquarters	1	888665555	Houston

Normalization of Multi-valued attribute into 1NF

Figure 14.8 Normalization into 1NF. (a) Relation schema that is not in 1NF. (b) Example relation instance. (c) 1NF relation with redundancy.

(a)



(b)

DEPARTMENT

DNAME	DNUMBER	DMGRSSN	DLOCATIONS
Research	5	333445555	{Bellaire, Sugarland, Houston}
Administration	4	987654321	{Stafford}
Headquarters	1	888665555	{Houston}

Problem

(c)

DEPARTMENT

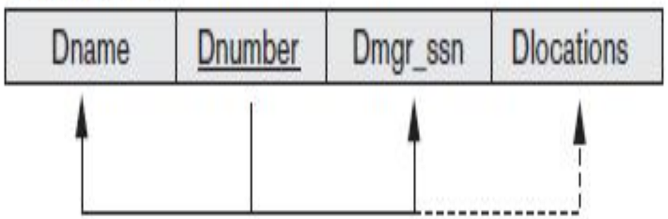
DNAME	DNUMBER	DMGRSSN	DLOCATION
Research	5	333445555	Bellaire
Research	5	333445555	Sugarland
Research	5	333445555	Houston
Administration	4	987654321	Stafford
Headquarters	1	888665555	Houston

Figure 15.9
Normalization into 1NF. (a) A relation schema that is not in 1NF. (b) Sample state of relation DEPARTMENT.

There are three main techniques to achieve first normal form for given relation:

(a)

DEPARTMENT



(b)

DEPARTMENT

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

Figure 15.9
Normalization into 1NF. (a) A relation schema that is not in 1NF. (b) Sample state of relation DEPARTMENT.

There are three main techniques to achieve first normal form for given relation:

DEPARTMENT

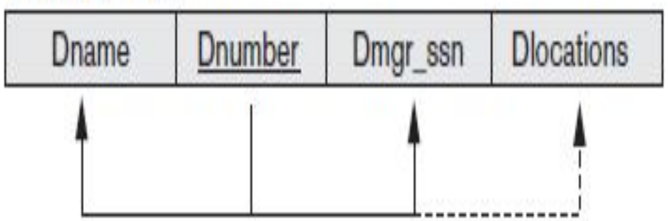
Dname	Dnumber	Dmgr_ssn
Research	5	333445555
Administration	4	987654321
Headquarters	1	888665555

DEPT_LOCATIONS

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

(a)

DEPARTMENT



(b)

DEPARTMENT

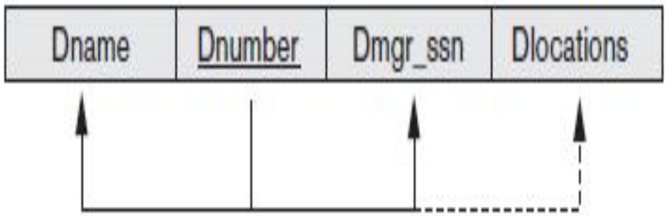
Dname	Dnumber	Dmgr_ssn	Dlocations
Research	5	333445555	{Bellaire, Sugarland, Houston}
Administration	4	987654321	{Stafford}
Headquarters	1	888665555	{Houston}

There are three main techniques to achieve first normal form for given relation:

Figure 15.9
Normalization into 1NF. (a) A relation schema that is not in 1NF. (b) Sample state of relation DEPARTMENT.

(a)

DEPARTMENT



(b)

DEPARTMENT

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

DEPARTMENT

Dname	Dnumber	Dmgr_ssn
Research	5	333445555
Administration	4	987654321
Headquarters	1	888665555

DEPT_LOCATIONS

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

DEPARTMENT

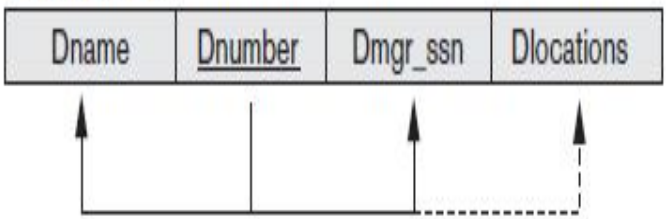
Dname	Dnumber	Dmgr_ssn	Dlocation
Research	5	333445555	Bellaire
Research	5	333445555	Sugarland
Research	5	333445555	Houston
Administration	4	987654321	Stafford
Headquarters	1	888665555	Houston

Figure 15.9
Normalization into 1NF. (a) A relation schema that is not in 1NF. (b) Sample state of relation DEPARTMENT.

There are three main techniques to achieve first normal form for given relation:

(a)

DEPARTMENT



(b)

DEPARTMENT

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

DEPARTMENT

Dname	Dnumber	Dmgr_ssn
Research	5	333445555
Administration	4	987654321
Headquarters	1	888665555

DEPT_LOCATIONS

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

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

DEPARTMENT

Dname	Dnumber	Dmgr_ssn	DLoc1	DLoc2	DLoc3
Research	5	333445555	Bellaire	Sugarland	Houston
Administration	4	987654321	Stafford	Null	Null
Headquarters	1	888665555	Houston	Null	Null

1NF - Example

(a) **EMP_PROJ**

SSN	ENAME	PROJS	
		PNUMBER	HOURS

(b) **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

Figure 14.9 Normalizing nested relations into 1NF. (a) Schema of the EMP_PROJ relation with a "nested relation" PROJS. (b) Example extension of the EMP_PROJ relation showing nested relations within each tuple. (c) Decomposing EMP_PROJ into 1NF relations EMP_PROJ1 and EMP_PROJ2 by propagating the primary key.

1NF - Example

(a) **EMP_PROJ**

SSN	ENAME	PROJS	
		PNUMBER	HOURS

(b) **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

(c) **EMP_PROJ1**

SSN	ENAME

EMP_PROJ2

SSN	PNUMBER	HOURS

Figure 14.9 Normalizing nested relations into 1NF. (a) Schema of the EMP_PROJ relation with a "nested relation" PROJS. (b) Example extension of the EMP_PROJ relation showing nested relations within each tuple. (c) Decomposing EMP_PROJ into 1NF relations EMP_PROJ1 and EMP_PROJ2 by propagating the primary key.

1NF - Example

Course	Content
Programming	Java, c++
Web	HTML, PHP, ASP

1NF - Example

Course	Content
Programming	Java, c++
Web	HTML, PHP, ASP

We re-arrange the relation (table) as below, to convert it to First Normal Form.

Course	Content
Programming	Java
Programming	c++
Web	HTML
Web	PHP
Web	ASP

Each attribute must contain only a single value from its pre-defined domain.

Second Normal Form

- Uses the concepts of **FDs**, **primary key**

Definitions:

- **Prime attribute** - attribute that is member of the primary key K
- **Full functional dependency** - a FD Y **->** Z where removal of any attribute from Y means the FD does not hold any more

Example

(b)

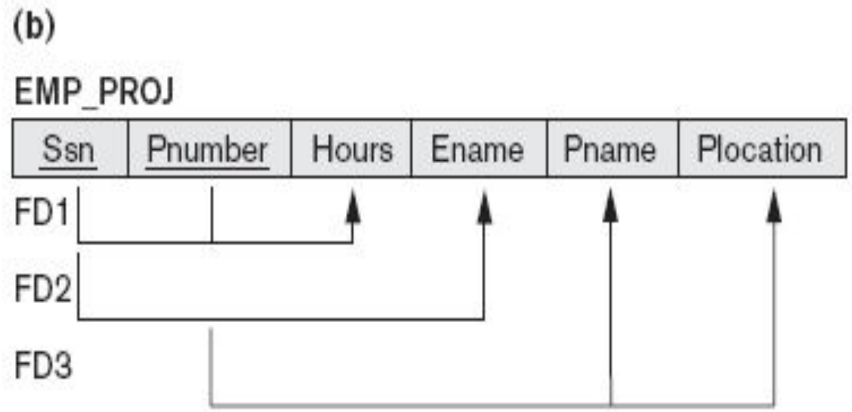
EMP_PROJ

Ssn	Pnumber	Hours	Ename	Pname	Plocation

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



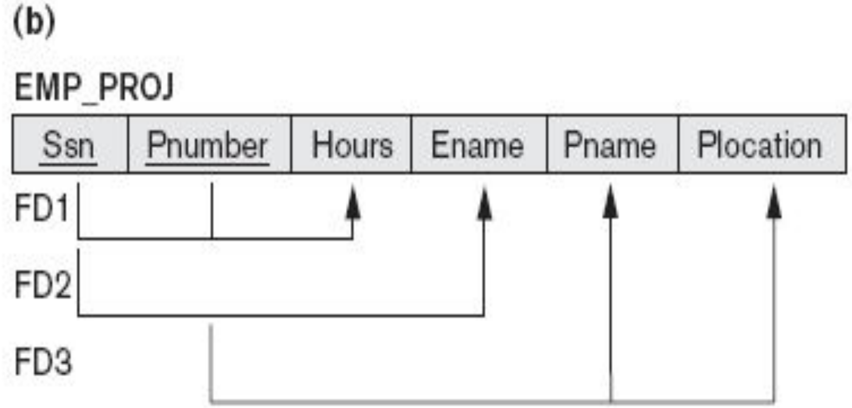
EMP_PROJ

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

Examples:

- {SSN, PNUMBER} -> HOURS is a full FD since neither SSN -> HOURS nor PNUMBER -> HOURS hold



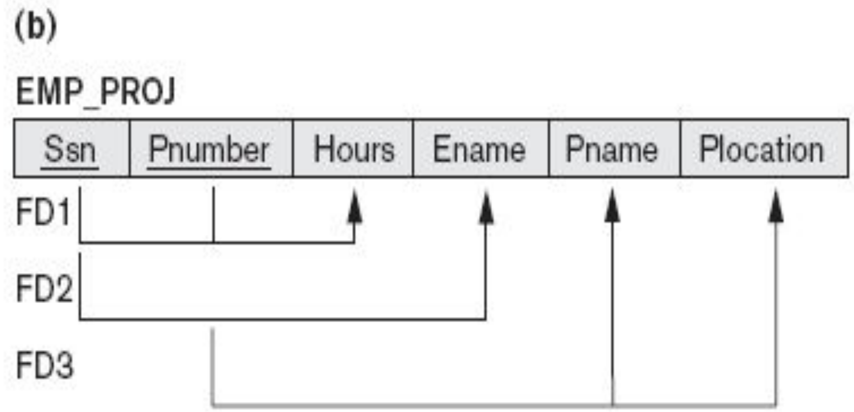
EMP_PROJ

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

Examples:

- {SSN, PNUMBER} -> HOURS is a full FD since neither SSN -> HOURS nor PNUMBER -> HOURS hold



- {SSN, PNUMBER}-> ENAME is *not* a full FD (it is called a *partial dependency*) since SSN -> ENAME also holds

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

Second Normal Form (2)

- Definition

  "A relation schema R is in **second normal form (2NF)** if relation is in 1NF and **if every non-prime attribute A in R is fully functionally dependent on the primary key**"

- R can be decomposed into 2NF relations via the process of 2NF normalization

Figure 10.10 Normalizing into 2NF

Figure 14.10 The normalization process. (a) Normalizing EMP_PROJ into 2NF relations.

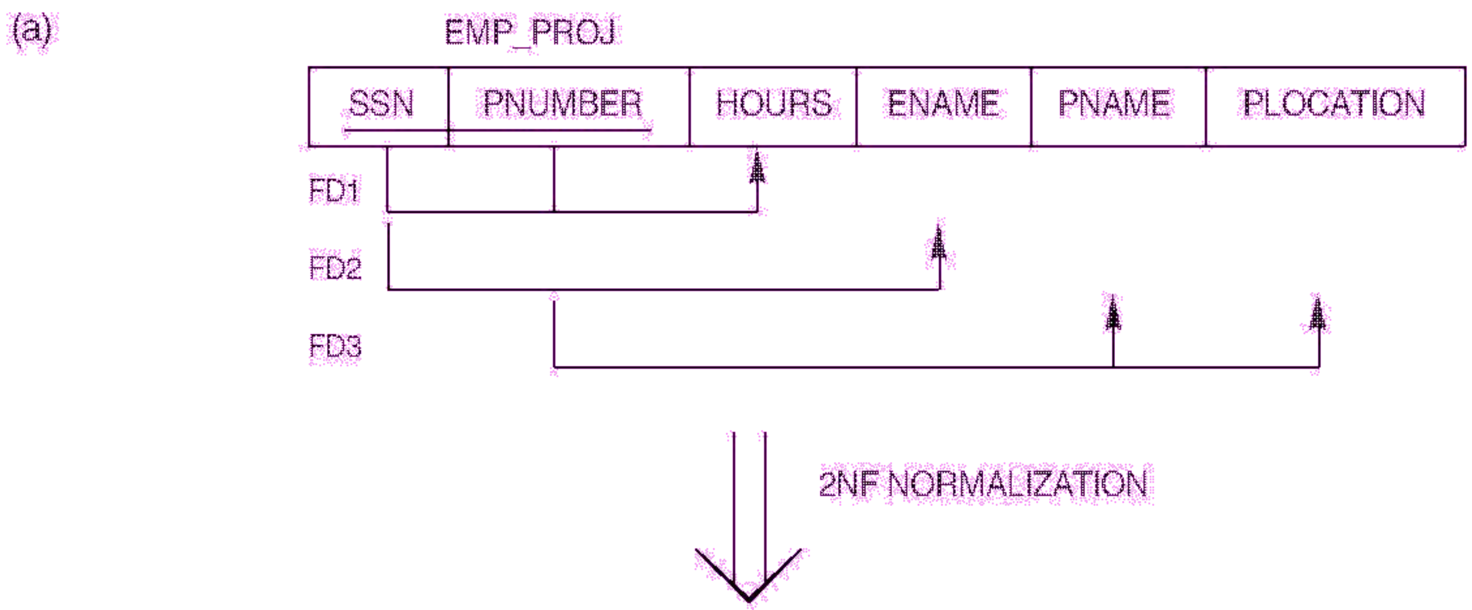
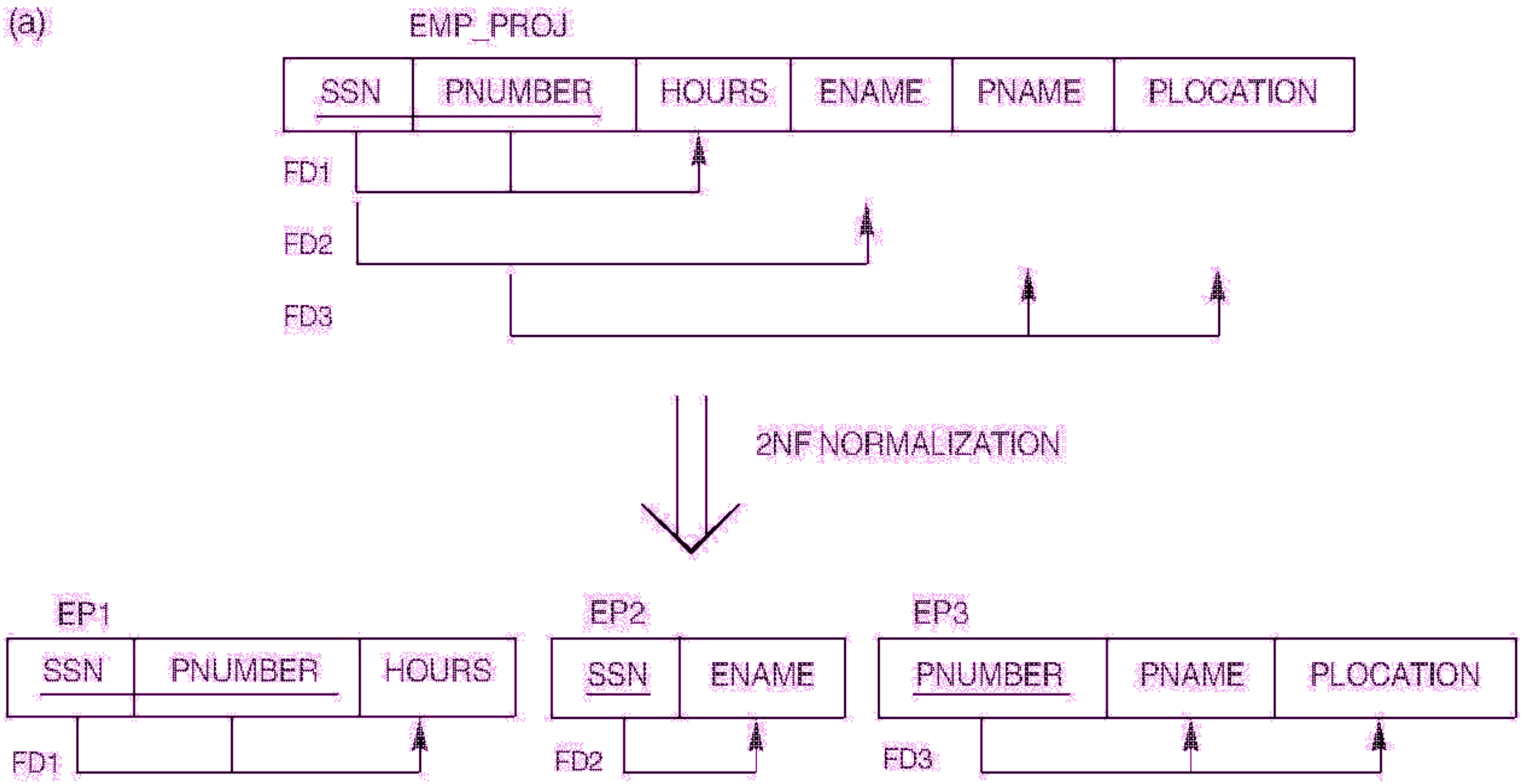


Figure 10.10 Normalizing into 2NF

Figure 14.10 The normalization process. (a) Normalizing EMP_PROJ into 2NF relations.

(a)



Another Example for 2NF

Employee

Emp-Id	Month	Emp-Name	Sales	Bank-Id
E01	Jan	AA	1000	B01
E01	Feb	AA	1100	B03
E01	March	AA	3000	B01
E02	Jan	BB	1200	B02
E02	Feb	BB	2200	B02
E03	Jan	CC	2200	B01
E03	Feb	CC	2200	B01
E04	Jan	DD	1500	B03

Another Example for 2NF

Employee

Emp-Id	Month	Emp-Name	Sales	Bank-Id
E01	Jan	AA	1000	B01
E01	Feb	AA	1100	B03
E01	March	AA	3000	B01
E02	Jan	BB	1200	B02
E02	Feb	BB	2200	B02
E03	Jan	CC	2200	B01
E03	Feb	CC	2200	B01
E04	Jan	DD	1500	B03

Emp-Id, Month $\rightarrow$ {Bank-Id, Sales}
Since Emp-Name is non-prime attribute which does not depends on Primary key. Remove it and put in another table

Solution :

Employee

Emp-Id	**Month**	**Sales**	**Bank-Id**
E01	Jan	1000	B01
E01	Feb	1100	B03
E01	March	3000	B01
E02	Jan	1200	B02
E02	Feb	2200	B02
E03	Jan	2200	B01
E03	Feb	2200	B01
E04	Jan	1500	B03

Emp_names

Emp-Id	**Emp-Name**
E01	AA
E01	AA
E01	AA
E02	BB
E02	BB
E03	CC
E03	CC
E04	DD

Third Normal Form (1)

<u>Definition:</u>

- **Transitive functional dependency** - a FD X -> Z that can be derived from two FDs X -> Y and Y -> Z

<u>Examples:</u>

SSN -> DMGRSSN is a *transitive* FD since

SSN -> DNUMBER and DNUMBER -> DMGRSSN hold

- SSN -> ENAME is *non-transitive* since there is no set of attributes X where SSN -> X and X -> ENAME

Third Normal Form (2)

- A relation schema R is in **third normal form** **(3NF)** if it is in 2NF *and* **no non-prime attribute A in R is transitively dependent on the primary key**
- R can be decomposed into 3NF relations via the process of 3NF normalization

NOTE:

In X **->** Y and Y **->** Z, with X as the primary key, we consider this a problem only if Y is <u>not</u> a candidate key. When Y is a candidate key, there is no problem with the transitive dependency .

E.g., Consider EMP (SSN, Emp#, Salary).

Here, SSN **->** Emp# **->** Salary and Emp# is a candidate key.

Figure 10.10 Normalizing into 3NF

Figure 14.10 The normalization process.

b) Normalizing EMP_DEPT into 3NF relations.

(b)

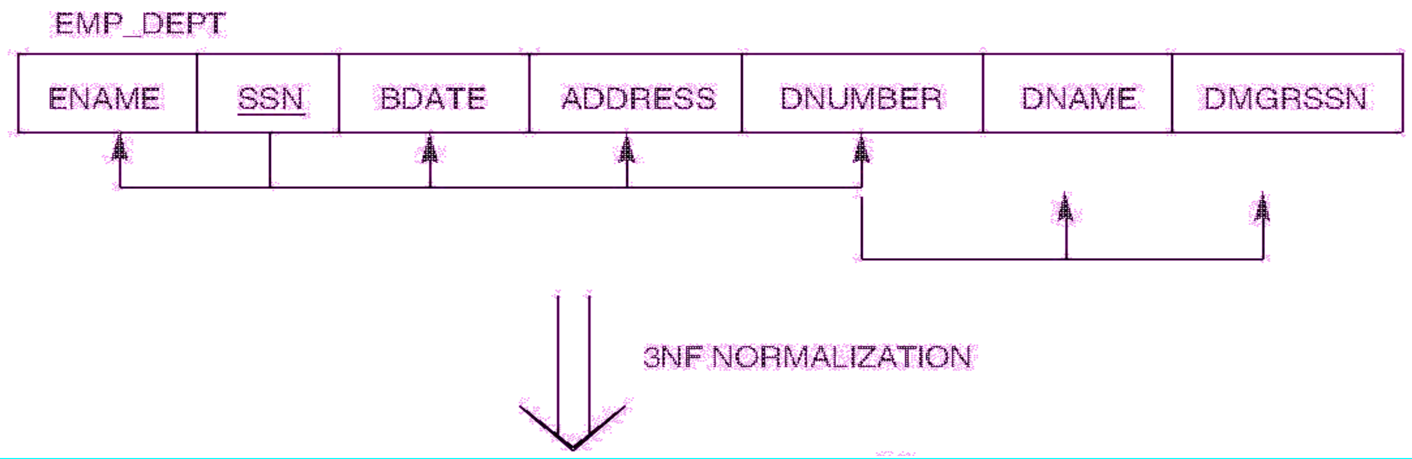
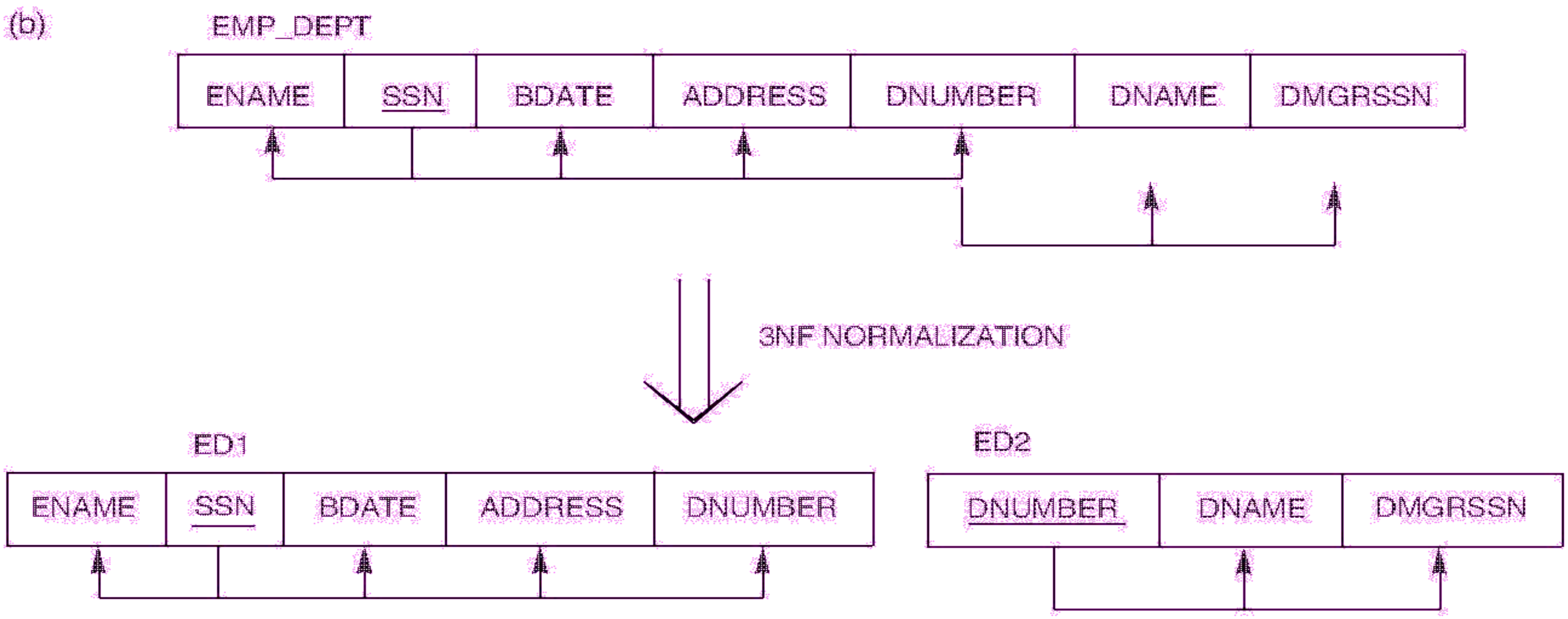


Figure 10.10 Normalizing into 3NF

Figure 14.10 The normalization process.

b) Normalizing EMP_DEPT into 3NF relations.



Another Example for 3NF

Student

Is it in 3NF ???

<u>RollNo</u>	**Name**	**Tel_No**	**City**	**STD_Code**

Another Example for 3NF

Student

Not in 3NF

<u>RollNo</u>	**Name**	**Tel_No**	**City**	**STD_Code**

Solution: ???

Another Example for 3NF

Student

Not in 3NF

<u>**RollNo**</u>	**Name**	**Tel_No**	**City**	**STD_Code**

Solution:

Student

<u>**RollNo**</u>	**Name**	**Tel_No**	**City**

City_table

<u>**City**</u>	**STD_Code**

Solve this Example for 2NF + 3 NF

Employee

<u>**Emp-Id**</u>	<u>**Month**</u>	**Emp-Name**	**Contact**	**Sales**	**Bank-Id**	**Bank-Name**
E01	Jan	AA	123456	1000	B01	SBI
E01	Feb	AA	123456	1100	B03	ICICI
E01	March	AA	123456	3000	B01	SBI
E02	Jan	BB	111111	1200	B02	UTI
E02	Feb	BB	111111	2200	B02	UTI
E03	Jan	CC	222222	2200	B01	SBI
E03	Feb	CC	222222	2200	B01	SBI
E04	Jan	DD	333333	1500	B03	ICICI

BCNF (Boyce-Codd Normal Form)

- 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
- There exist relations that are in 3NF but not in BCNF
- The goal is to have each relation in BCNF (or 3NF)

BCNF (Boyce-Codd Normal Form)

- **Definition:**

“A relationship is said to be in BCNF if it is already in 3NF and and a relation schema R is in Boyce-Codd Normal Form (BCNF) if every determinant is a candidate key.”

OR

“ A relationship is said to be in BCNF if it is already in 3NF and the left hand side of every dependency is a candidate key.”

BCNF - Example

Professor Code	Department	Head of Dept.	Percent Time
P1	Physics	Ghosh	50
P1	Mathematics	Krishnan	50
P2	Chemistry	Rao	25
P2	Physics	Ghosh	75
P3	Mathematics	Krishnan	100

BCNF - Example

Professor Code	Department	Head of Dept.	Percent Time
P1	Physics	Ghosh	50
P1	Mathematics	Krishnan	50
P2	Chemistry	Rao	25
P2	Physics	Ghosh	75
P3	Mathematics	Krishnan	100

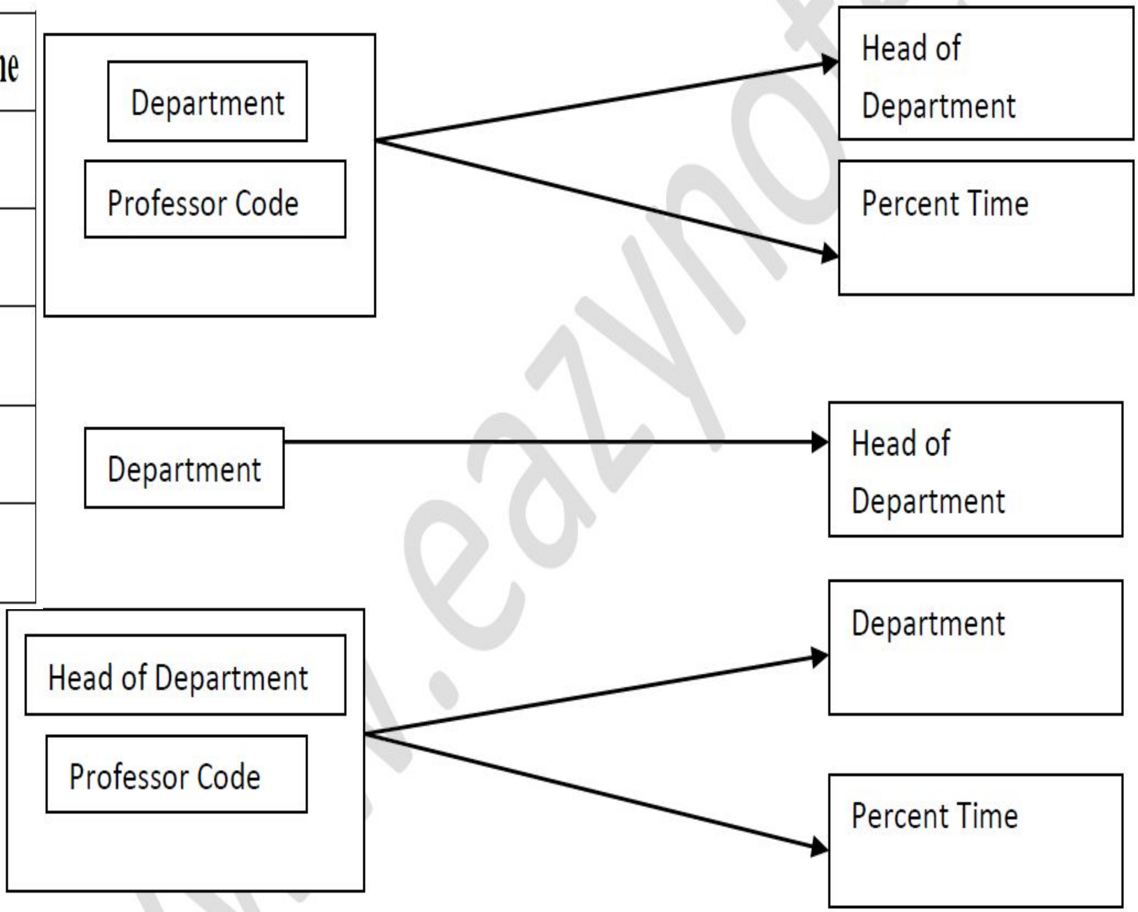
Consider, as an example, the above relation. It is assumed that:

1. A professor can work in more than one department
2. The percentage of the time he spends in each department is given.
3. Each department has only one Head of Department.

BCNF - Example

Professor Code	Department	Head of Dept.	Percent Time
P1	Physics	Ghosh	50
P1	Mathematics	Krishnan	50
P2	Chemistry	Rao	25
P2	Physics	Ghosh	75
P3	Mathematics	Krishnan	100

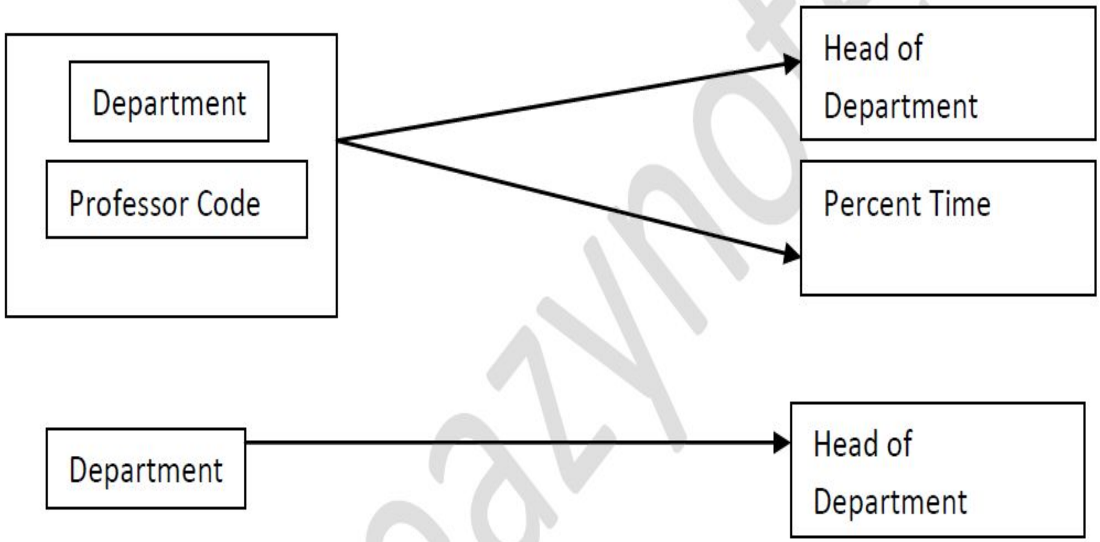
The relation diagram for the above relation is given as the following:



BCNF - Example

Professor Code	Department	Head of Dept.	Percent Time
P1	Physics	Ghosh	50
P1	Mathematics	Krishnan	50
P2	Chemistry	Rao	25
P2	Physics	Ghosh	75
P3	Mathematics	Krishnan	100

The relation diagram for the above relation is given as the following:



The given relation is in 3NF. Observe, however, that the names of Dept. and Head of Dept. are duplicated. Further, if Professor P2 resigns, rows 3 and 4 are deleted. We lose the information that Rao is the Head of Department of Chemistry.

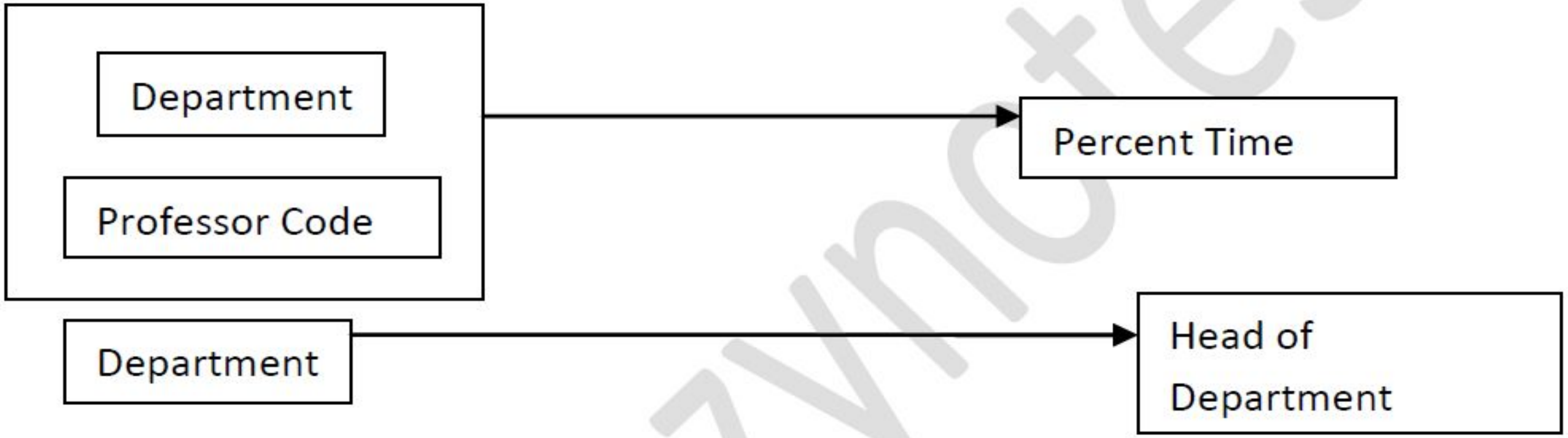
BCNF - Example

The normalization of the relation is done by creating a new relation for Dept. and Head of Dept. and deleting Head of Dept. form the given relation. The normalized relations are shown in the following.

Professor Code	Department	Percent Time
P1	Physics	50
P1	Mathematics	50
P2	Chemistry	25
P2	Physics	75
P3	Mathematics	100

Department	Head of Dept.
Physics	Ghosh
Mathematics	Krishnan
Chemistry	Rao

See the dependency diagrams for these new relations.



BCNF (Boyce-Codd Normal Form)

- Another Example

Table : R

ID	Ename	Qualification	Grade	Did	Dname
1	Sana	M.E.	B	40	Comps
2	Savita	M.E.	B	30	CE
3	Sohan	B.E.	C	20	EX
4	Mahesh	Ph.D.	A	10	IT
5	Hari	Ph.D.	A	40	Comps

BCNF (Boyce-Codd Normal Form)

Table : R1

ID	Ename	Qualification	Did
1	Sana	M.E.	40
2	Savita	M.E.	30
3	Sohan	B.E.	20
4	Mahesh	Ph.D.	10
5	Hari	Ph.D.	40

Table : R2

Qualification	Grade
B.E.	C
M.E.	B
Ph.D.	A

Table : R3

Did	Dname
20	EX
30	CE
10	IT
40	Comps

Discussion Questions

Exercise Questions

Dec 2011

1. (a) What are the five main functions of a database administrator ? 5

   (b) List all functional dependencies satisfied by the relation. 5

A	B	C
a1	b1	c1
a1	b1	c2
a2	b1	c1
a2	b1	c3

Q2. Relation R(A,B,C,D,E) showing following set of FDs.

$$A \rightarrow D$$
$$AB \rightarrow C$$
$$D \rightarrow E$$

If the above relation undergoes decomposition R1(A,B,C) and R2(A,D,E) weather above decomposition is lossless & dependency preserving.

3. The following relation is in which normal form ?

Employee

emp_no	name	dept_no	dept_name	skills
1	Kevin	201	R&D	C, Perl, Java
2	Barbara	224	IT	Linux, Mac
3	Jake	201	R&D	DB2, Oracle, Java

A. 1 NF **B. 2 NF** **C. Un-normalized** **D. 4 NF**

3. The following relation is in which normal form ?

Employee				
emp_no	**name**	**dept_no**	**dept_name**	**skills**
1	Kevin	201	R&D	C, Perl, Java
2	Barbara	224	IT	Linux, Mac
3	Jake	201	R&D	DB2, Oracle, Java

A. 1 NF **B. 2 NF** **C. Un-normalized** **D. 4 NF**

4. Convert above table into 1 NF.

3. The following relation is in which normal form ?

Employee

emp_no	name	dept_no	dept_name	skills
1	Kevin	201	R&D	C, Perl, Java
2	Barbara	224	IT	Linux, Mac
3	Jake	201	R&D	DB2, Oracle, Java

A. 1 NF **B. 2 NF** **C. Un-normalized** **D. 4 NF**

4. Convert above table into 1 NF.

Employee (1NF)

emp_no	name	dept_no	dept_name	skills
1	Kevin	201	R&D	C
1	Kevin	201	R&D	Perl
1	Kevin	201	R&D	Java
2	Barbara	224	IT	Linux
2	Barbara	224	IT	Mac
3	Jake	201	R&D	DB2
3	Jake	201	R&D	Oracle
3	Jake	201	R&D	Java

5. Convert the given table into 2 NF.

Employee (1NF)

emp_no	name	dept_no	dept_name	skills
1	Kevin Jacobs	201	R&D	C
1	Kevin Jacobs	201	R&D	Perl
1	Kevin Jacobs	201	R&D	Java
2	Barbara Jones	224	IT	Linux
2	Barbara Jones	224	IT	Mac
3	Jake Rivera	201	R&D	DB2
3	Jake Rivera	201	R&D	Oracle
3	Jake Rivera	201	R&D	Java

Hint:

Name, dept_no, and dept_name are functionally dependent on emp_no. **(emp_no -> name, dept_no, dept_name)**

Skills is not functionally dependent on emp_no since it is not unique to each emp_no.

5. Convert the given table into 2 NF.

Employee (1NF)

emp_no	name	dept_no	dept_name	skills
1	Kevin Jacobs	201	R&D	C
1	Kevin Jacobs	201	R&D	Perl
1	Kevin Jacobs	201	R&D	Java
2	Barbara Jones	224	IT	Linux
2	Barbara Jones	224	IT	Mac
3	Jake Rivera	201	R&D	DB2
3	Jake Rivera	201	R&D	Oracle
3	Jake Rivera	201	R&D	Java

Hint:

Name, dept_no, and dept_name are functionally dependent on emp_no. (emp_no -> name, dept_no, dept_name)

Skills is not functionally dependent on emp_no since it is not unique to each emp_no.

Employee (2NF)

emp_no	name	dept_no	dept_name
1	Kevin Jacobs	201	R&D
2	Barbara Jones	224	IT
3	Jake Rivera	201	R&D

Skills (2NF)

emp_no	skills
1	C
1	Perl
1	Java
2	Linux
2	Mac
3	DB2
3	Oracle
3	Java

6. Convert the given table into 3 NF.

Employee (2NF)			
emp_no	**name**	**dept_no**	**dept_name**
1	Kevin Jacobs	201	R&D
2	Barbara Jones	224	IT
3	Jake Rivera	201	R&D

Hint: Dept_no and dept_name are functionally dependent on emp_no however, department can be considered a separate entity.

6. Convert the given table into 3 NF.

Employee (2NF)

emp_no	name	dept_no	dept_name
1	Kevin Jacobs	201	R&D
2	Barbara Jones	224	IT
3	Jake Rivera	201	R&D

Hint: **Dept_no and dept_name are functionally dependent on emp_no however, department can be considered a separate entity.**

Employee (3NF)

emp_no	name	dept_no
1	Kevin Jacobs	201
2	Barbara Jones	224
3	Jake Rivera	201

Department (3NF)

dept_no	dept_name
201	R&D
224	IT

2. (a) Describe 2 NF, 3 NF and BCNF with examples. 8

Decompose 2 NF table into 3 NF tables with example

Decompose 3NF table into BCNF tables with example.

7. (a) What is normalization ? Explain 1NF, 2NF, 3NF and BCNF with suitable example. 10