

Introduction to normalization

TEAM INFDEV

Hogeschool Rotterdam
Rotterdam, Netherlands

Introduction
to
normalization

TEAM
INFDEV

Lesson
overview

Introduction

Normal forms

General
normal forms

Summary

- Problems caused by bad database implementations.
- Reasons behind Normalization.
- Functional dependencies.
- Normal forms.

Introduction

Reasons behind normalization

Introduction
to
normalization

TEAM
INFDEV

Lesson
overview

Introduction

Normal forms

General
normal forms

Summary

- It is a process to improve the design of the database.
- Objective 1: minimize the storage space used by the relations.
- Objective 2: eliminate anomalies in the database.
- Objective 3: eliminate spurious tuples.

Create tables with a lot of information is prone to waste of storage space.

| EMPLOYEE | | | | | DEPARTMENT | | |
|----------------------|------------|------------|--------------------------|---------|----------------|----------------|-----------|
| ENAME | <u>SSN</u> | BDATE | ADDRESS | DNUMBER | DNAME | <u>DNUMBER</u> | DMGRSSN |
| Smith, John B. | 123456789 | 1965-01-09 | 731 Fondren, Houston, TX | 5 | Research | 5 | 333445555 |
| Wong, Franklin T. | 333445555 | 1955-12-08 | 638 Voss, Houston, TX | 5 | Administration | 4 | 987654321 |
| Zelaya, Alicia J. | 999887777 | 1968-07-19 | 3321 Castle, Spring, TX | 4 | Headquarters | 1 | 888665555 |
| Wallace, Jennifer S. | 987654321 | 1941-06-20 | 291 Berry, Bellaire, TX | 4 | | | |
| Narayan, Ramesh K. | 666884444 | 1962-09-15 | 975 Fire Oak, Humble, TX | 5 | | | |
| English, Joyce A. | 453453453 | 1972-07-31 | 5631 Rice, Houston, TX | 5 | | | |
| Jabbar, Ahmad V. | 987987987 | 1969-03-29 | 980 Dallas, Houston, TX | 4 | | | |
| Borg, James E. | 888665555 | 1937-11-10 | 450 Stone, Houston, TX | 1 | | | |

VS

| ENAME | <u>SSN</u> | 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 Fire Oak, 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 |

Introduction

Waste of storage space

Introduction
to
normalization

TEAM
INFDEV

Lesson
overview

Introduction

Normal forms

General
normal forms

Summary

- Using two separate tables and joining them uses less storage space than putting everything in the same table.
- The values of the attributes of the department are repeated for every employee who works in that department.
- It means that, if a department has 300 employees, we waste 300 times the memory space than the approach with two tables.

Introduction

Update Anomalies

Introduction
to
normalization

TEAM
INFDEV

Lesson
overview

Introduction

Normal forms

General
normal forms

Summary

| EMP_DEPT | | | | | redundancy | |
|----------------------|-----------|------------|--------------------------|---------|----------------|-----------|
| 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 |
| Narsyen, Ramesh K. | 666894444 | 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 |

- The values contained in DNAME are always coupled with the same values in DMGRSSN. For example 'Research' is always coupled with 333445555. This is called redundancy.
- Inserting a new employee requires to repeat this coupling manually (manual consistency).
- If you add a new department, then we have to enter a null value for each attribute of the employee.

Introduction

Update Anomalies (deletion)

Introduction
to
normalization

TEAM
INFDEV

Lesson
overview

Introduction

Normal forms

General
normal forms

Summary

- Deleting the tuple of the last employee of the department also removes all the information about that department.
- We must manually apply null values to the attributes of the employee in order to remove the last employee from the department.
- Example: Deleting Borg, James E. also removes all the information about the 'Headquarters' department.

Introduction

Update Anomalies (modification)

Introduction
to
normalization

TEAM
INFDEV

Lesson
overview

Introduction

Normal forms

General
normal forms

Summary

- Modifying the information of a department in one tuple is not reflected in the other tuples.
- If we want to update the information about a department we must repeat the update for all the tuples containing the employees of that department
- **Example:** If we change the name of the department 'Research' into 'Research and development' we have to repeat this change for 4 tuples.

Introduction

Spurious tuples

Introduction
to
normalization

TEAM
INFDEV

Lesson
overview

Introduction

Normal forms

General
normal forms

Summary

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. | Houston |
| Borg, James E. | Houston |

EMP_PROJ1

| SSN | PNUMBER | HOURS | PNAME | PLOCATION |
|-----------|---------|-------|-----------------|-----------|
| 123456789 | 1 | 32.5 | Product X | Bellaire |
| 123456789 | 2 | 7.5 | Product Y | Sugarland |
| 666884444 | 3 | 40.0 | Product Z | Houston |
| 453453453 | 1 | 20.0 | Product X | Bellaire |
| 453453453 | 2 | 20.0 | Product Y | Sugarland |
| 333445555 | 2 | 10.0 | Product Y | Sugarland |
| 333445555 | 3 | 10.0 | Product Z | 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 |

| 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 |

- Normalization uses decomposition to eliminate anomalies and remove redundancy.
- The two tables above in the previous slide are a decomposition of the bottom one.
- If we join the two tables, we get more tuples than those that were in the original table
- Those below the dotted line are extra tuples called spurious tuples.
- Normalization must ensure that the decomposition does not create spurious tuples after a join.

- Normalization is the process of converting the relations (tables) of a database into other relations in some normal form.
- Normal forms are properties that a relation must satisfy.
- There are two different kinds of definitions of normal forms:
 - Historical definitions: The original definitions defined by Codd, taking into account only the primary key.
 - Refined definitions: A refinement of the normal forms that takes into account also the candidate keys.

Introduction
to
normalization

TEAM
INFDEV

Lesson
overview

Introduction

Normal forms

General
normal forms

Summary

BRACE YOURSELVES

MATH IS COMING

Normal forms

Functional dependencies

Introduction
to
normalization

TEAM
INFDEV

Lesson
overview

Introduction

Normal forms

General
normal forms

Summary

- Both kinds of definitions of normal forms rely on the concept of functional dependency.
- In what follows we write $t_i[X]$ to indicate the values of the row i of a table for a set of attributes X

- Both kinds of definitions of normal forms rely on the concept of functional dependency.
- In what follows we write $t_i[X]$ to indicate the values of the row i of a table for a set of attributes X
- Example:** $t_3[\text{SSN}, \text{HOURS}, \text{NAME}]$ in the table below is the combination of values: $t_3[\text{SSN}, \text{HOURS}, \text{NAME}] = \{666884444, 40.0, \text{Narayan, Ramesh K.}\}$

| EMP_PROJ | | redundancy | | redundancy | |
|-----------|---------|------------|----------------------|-----------------|-----------|
| 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 |
| 997967967 | 10 | 35.0 | Jaibbar, Ahmad V. | Computerization | Stafford |
| 997967967 | 30 | 5.0 | Jaibbar, Ahmad V. | Newbenefits | Stafford |
| 997654321 | 30 | 20.0 | Wallace, Jennifer S. | Newbenefits | Stafford |
| 997654321 | 20 | 15.0 | Wallace, Jennifer S. | Reorganization | Houston |
| 888665555 | 20 | null | Borg, James E. | Reorganization | Houston |

Definition:

Given a subset of attributes X and Y in a relation, we say that Y depends on X , and we write $X \rightarrow Y$, if for any pair of rows where we have $t_i[X] = t_j[X]$ we also have $t_i[Y] = t_j[Y]$.

Normal forms

Functional dependencies

Introduction
to
normalization

TEAM
INFDEV

Lesson
overview

Introduction

Normal forms

General
normal forms

Summary

Definition:

Given a subset of attributes X and Y in a relation, we say that Y depends on X , and we write $X \rightarrow Y$, if for any pair of rows where we have $t_i[X] = t_j[X]$ we also have $t_i[Y] = t_j[Y]$.

Example: In the table below we have, if we assume that there is a dependency $PNAME \rightarrow PLOCATION$, then when $t_i[PNAME] = t_j[PNAME]$ we also have $t_i[PLOCATION] = t_j[PLOCATION]$. For instance every time we have the value 'ProductY' in the column PNAME, we also have 'Sugarland' in the column PLOCATION. It is not possible that a row has 'ProductY' in the column PNAME and 'Belaire' in the column 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 |
| 987654321 | 3 | 40.0 | Narayan, Ramesh K. | ProductZ | Houston |
| 432109876 | 1 | 20.0 | English, Joyce A. | ProductX | Bellaire |
| 432109876 | 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 |
| 997957967 | 10 | 35.0 | Jabbar, Ahmad V. | Computerization | Stafford |
| 997957967 | 30 | 5.0 | Jabbar, Ahmad V. | Newbenefits | Stafford |
| 997654321 | 30 | 20.0 | Wallace, Jennifer S. | Newbenefits | Stafford |
| 997654321 | 20 | 15.0 | Wallace, Jennifer S. | Reorganization | Houston |
| 888665555 | 20 | null | Borg, James E. | Reorganization | Houston |

Functional dependencies

Important observations

Introduction
to
normalization

TEAM
INFDEV

Lesson
overview

Introduction

Normal forms

General
normal forms

Summary

- Functional dependencies are defined by the database designer.
- It is not possible to infer functional dependencies by reading the data from the table.
- Functional dependencies should be read in the documentation of the database or asked to the designer.
- If you need to normalize a database and there is no documentation or the designer is unavailable, **as last resort**, you can try to infer functional dependencies from the data.

Functional dependencies

Bad practice - An example

Introduction
to
normalization

TEAM
INFDEV

Lesson
overview

Introduction

Normal forms

General
normal forms

Summary

| <u>ssn</u> | name | salary |
|------------|---------------|--------|
| 299260 | John Smith | 2500 |
| 993693 | Walter White | 3000 |
| 388528 | John Smith | 2500 |
| 396926 | James Garnett | 1800 |

Functional dependencies

Bad practice - An example

| <u>ssn</u> | name | salary |
|------------|---------------|--------|
| 299260 | John Smith | 2500 |
| 993693 | Walter White | 3000 |
| 388528 | John Smith | 2500 |
| 396926 | James Garnett | 1800 |

By looking at the data in the table I could infer that name \rightarrow salary because two employees with the name John Smith earn 2500 euros per month. If later I add the tuple (395935, John Smith, 2000) (another employee with the same name but different salary) the dependency is gone! This because clearly the salary of an employee should not depend on a person's name, but this is what we could infer just by looking at the data.

Normal forms

1st Normal form

Introduction
to
normalization

TEAM
INFDEV

Lesson
overview

Introduction

Normal forms

General
normal forms

Summary

- The first normal form is automatically granted by the relational model.
- If you designed the conceptual model properly and translated correctly the ERD into the relational model, then all your relations should be in 1NF.

Definition:

A table is in 1st normal form (1NF) if no row has multiple values for an attribute.

Example:

The table to the right violates the 1NF since some rows have multiple values for PNUMBER and HOURS.

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 |

Normal forms

2nd Normal form

Introduction
to
normalization

TEAM
INFDEV

Lesson
overview

Introduction

Normal forms

General
normal forms

Summary

Definition: A table is in 2nd normal form (2NF) if it is in 1NF and every attribute that is not part of a key depends on the whole primary key.

Normal forms

2nd Normal form

Introduction
to
normalization

TEAM
INFDEV

Lesson
overview

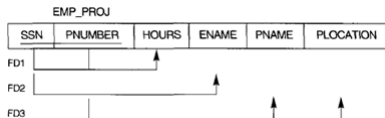
Introduction

Normal forms

General
normal forms

Summary

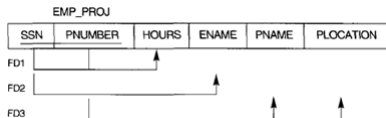
Definition: A table is in 2nd normal form (2NF) if it is in 1NF and every attribute that is not part of a key depends on the whole primary key.



Definition: A table is in 2nd normal form (2NF) if it is in 1NF and every attribute that is not part of a key depends on the whole primary key.

Example:

The table to the right is not in 2NF because PNAME depends only on PNUMBER and not on the whole key. The same for PLOCATION



Useful trick: If you have a table where the primary key is made of only one attribute, the table is always in 2NF. You can say this without any additional check

Why?

Useful trick: If you have a table where the primary key is made of only one attribute, the table is always in 2NF. You can say this without any additional check

Why? Because to break the 2NF there must exist a functional dependency where the left side is a part of the primary key, but this is impossible if you have only one attribute.

Normal forms

2nd Normal Form - Reson

Introduction
to
normalization

TEAM
INFDEV

Lesson
overview

Introduction

Normal forms

General
normal forms

Summary

| EMP_PROJ | | redundancy | | redundancy | |
|-----------|---------|------------|----------------------|-----------------|-----------|
| SSN | PNUMBER | HOURS | ENAME | PNAME | PLOCATION |
| 123456789 | 1 | 32.5 | Smith, John B. | ProductX | Bellaire |
| 123456789 | 2 | 7.5 | Smith, John B. | ProductY | Sugarland |
| 999999999 | 3 | 40.0 | Narayan, Ramesh K. | ProductZ | Houston |
| 456789453 | 1 | 20.0 | English, Joyce A. | ProductX | Bellaire |
| 456789453 | 2 | 20.0 | English, Joyce A. | ProductY | Sugarland |
| 339465555 | 2 | 10.0 | Wong, Franklin T. | ProductY | Sugarland |
| 339465555 | 3 | 10.0 | Wong, Franklin T. | ProductZ | Houston |
| 339465555 | 10 | 10.0 | Wong, Franklin T. | Computerization | Stafford |
| 339465555 | 20 | 10.0 | Wong, Franklin T. | Reorganization | Houston |
| 999997777 | 30 | 30.0 | Zelaya, Alicia J. | Newbenefits | Stafford |
| 999997777 | 10 | 10.0 | Zelaya, Alicia J. | Computerization | Stafford |
| 997957967 | 10 | 35.0 | Jabbar, Ahmad V. | Computerization | Stafford |
| 997957967 | 30 | 5.0 | Jabbar, Ahmad V. | Newbenefits | Stafford |
| 997654321 | 30 | 20.0 | Wallace, Jennifer S. | Newbenefits | Stafford |
| 997654321 | 20 | 15.0 | Wallace, Jennifer S. | Reorganization | Houston |
| 888885555 | 20 | null | Borg, James E. | Reorganization | Houston |

If I want to assign an employee to an existing project, I have to manually repeat the data in PNAME and PLOCATION for the new tuple even if they already exist in other tuples.

For example if I add the employee 394394343 to project 1, I have to repeat the values ProductX and Bellaire in the new tuple manually.

Normal forms

Transitive dependency

Introduction
to
normalization

TEAM
INFDEV

Lesson
overview

Introduction

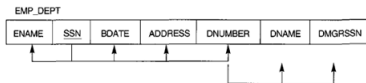
Normal forms

General
normal forms

Summary

The 3rd normal form requires to define a *transitive dependency*

Definition: A functional dependency $X \rightarrow Y$ is transitive if there is a set of attributes Z , which is neither a candidate key nor a subset of any key, and both $X \rightarrow Z$ and $Z \rightarrow Y$ are valid functional dependency



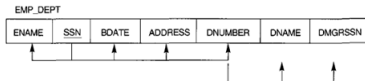
Example: In the table above the functional dependency $SSN \rightarrow \{ DNAME, DMGRSSN \}$ is transitive because also $SSN \rightarrow DNUMBER$ and $DNUMBER \rightarrow \{ DNAME, DMGRSSN \}$ are valid functional dependencies.

Another useful trick: We do not give the proof of this, but remember that you can always decompose a functional dependency, with a right side containing more than one attribute, into two smaller functional dependencies. This procedure is recursive, so you can apply it again to the two functional dependencies created at the previous step.

Example: $\text{DNUMBER} \rightarrow \{ \text{DNAME}, \text{DMGRSSN} \}$ can be decomposed into the two functional dependencies $\text{DNUMBER} \rightarrow \text{DNAME}$ and $\text{DNUMBER} \rightarrow \text{DMGRSSN}$

Definition (3NF): A table is in 3rd normal form (3NF) if it is in 2NF and no attribute that is not part of a key is transitively dependent on the primary key.

Example: The table below is in 2NF but not in 3NF because of the transitive dependency $SSN \rightarrow \{ DNAME, DMGRSSN \}$



Normal forms

3rd normal form - Reason

Introduction
to
normalization

TEAM
INFDEV

Lesson
overview

Introduction

Normal forms

General
normal forms

Summary

| 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 |

redundancy

If I add a new employee to a department I have to repeat DNAME and DMGRSSN for that department because of the transitive dependency.

If I add employee 394394343 to department number 5 I have to repeat the value Research and 333445555 for that department manually.

Normalization on candidate keys

Introduction
to
normalization

TEAM
INFDEV

Lesson
overview

Introduction
Normal forms

General
normal forms

Summary

- All the normal forms defined so far are defined for the primary key only.
- Anomalies might arise also with dependencies on the candidate key.
- We give more general definitions of normal forms that take into account also the candidate keys.
- The definition of 1NF is not affected since it only requires that each tuple has a single value for all the attributes (not based on a property on the primary key).

Normalization on candidate keys

General 2nd normal form

Introduction
to
normalization

TEAM
INFDEV

Lesson
overview

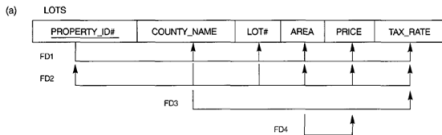
Introduction

Normal forms

General
normal forms

Summary

Definition: A table is in 2NF if every attribute that is not part of a key is not partially dependent on any key of the table (primary or candidate)



Normalization on candidate keys

General 2nd normal form

Introduction
to
normalization

TEAM
INFDEV

Lesson
overview

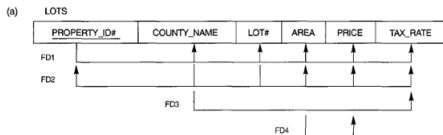
Introduction

Normal forms

General
normal forms

Summary

Definition: A table is in 2NF if every attribute that is not part of a key is not partially dependent on any key of the table (primary or candidate)



Example: The table is not in 2NF because a candidate key is

{ COUNTY_NAME, LOT# }

and the attribute TAX_RATE depends on COUNTY_NAME.

Normalization on candidate keys

General 2nd normal form - Reason

Introduction
to
normalization

TEAM
INFDEV

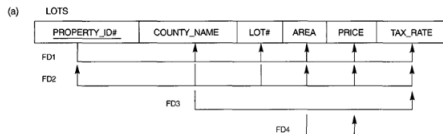
Lesson
overview

Introduction

Normal forms

General
normal forms

Summary



If I want to add a new property in an existing county, I have to manually repeat the data about AREA, PRICE, and TAX_RATE.

Trivial functional dependency: A functional dependency

$$X \rightarrow Y$$

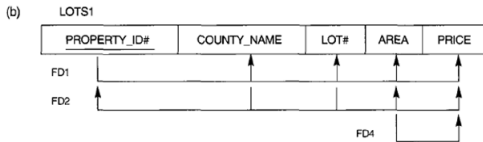
is trivial if Y is part of X .

Definition: A table is in 3rd normal form if it is in 2NF and for any non-trivial functional dependency $X \rightarrow A$ one of the two conditions is satisfied:

- 1 X is a superkey.
- 2 A is part of a key.

Normalization on candidate keys

General 3rd normal form - Reason



If I want to add a new property in a lot with the same area of another one, I have to manually repeat the data about the PRICE for that AREA.

It can be seen as a generalization of the previous 3NF where there must not be a transitive dependency also on a candidate key of the table

Normalization on candidate keys

Boyce-Codd normal form

Introduction
to
normalization

TEAM
INFDEV

Lesson
overview

Introduction
Normal forms

General
normal forms

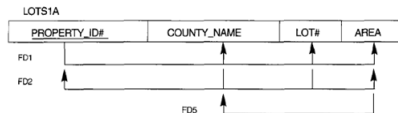
Summary

Definition: A table is in Boyce-Codd normal form (BCNF) if for all non-trivial dependencies $X \rightarrow A$ then X is a superkey of the table.

NOTE: If a table is in BCNF then it is also in 3NF. Indeed to ensure the 3NF it is enough that only one between the BCNF condition and the other one (see the previous slide) is met.

Example:

The table to the right is in 3NF but not in BCNF because `COUNTY_NAME` depends on `AREA`, which is not a superkey.



- Unnormalized tables might contain data redundancy and cause anomalies.
- Normal forms are a way to solve these problems.
- There are normal form definitions for tables with just one key (the primary key and no candidate keys).
- There are normal form definitions for tables with candidate keys.