Normalization

Mattew Pike & Yuan Yao

Univeristy of Nottingham Ningbo China (UNNC)

Warm Up

Recall: Relational Model

ID	Name	Salary	Department
M139	John Smith	18000	Marketing
M140	Mary Jones	22000	Marketing
A368	Jane Brown	22000	Accouts
P222	Mark Brown	24000	Personals
A367	David Jones	2000	Accounts

- Attributes, Schema, Tuples.
- Degree of the relation.
- Cardinality of the relation.

Warm up: An example database

sID	Name	Department	Building	mCode	Module
1	John Smith	Computer Science	B1	DBI, FAI	Database and Interfaces, Foundation of Artificial Intelligence
2	Mark Brown	Computer Science	B1	FAI	Foundation of Artificial Intelligence
3	Mary Jones	Computer Science	B1	PGA, DBI	Programming and Algorithms, Database and Interfaces
4	David Jones	Mathematics	A1	MCS	Mathematics for Computer Scientists

- Each staff belongs to one department.
- Each department has its own building.

Warm up: Non-atomic values

sID	Name	Department	Building	mCode	Module
1	John Smith	Computer Science	B1	DBI, FAI	Database and Interfaces, Foundation of Artificial Intelligence
2	Mark Brown	Computer Science	B1	FAI	Foundation of Artificial Intelligence
3	Mary Jones	Computer Science	B1	PGA, DBI	Programming and Algorithms, Database and Interfaces
4	David Jones	Mathematics	A1	MCS	Mathematics for Computer Scientists

- How to solve this problem?

Warm up: A table with atomic values

sID	Name	Department	Building	mCode	Module
1	John Smith	Computer Science	B1	DBI	Database and Interfaces
1	John Smith	Computer Science	B1	FAI	Foundation of Artificial Intelligence
2	Mark Brown	Computer Science	B1	FAI	Foundation of Artificial Intelligence
3	Mary Jones	Computer Science	B1	PGA	Programming and Algorithms
3	Mary Jones	Computer Science	B1	DBI	Database and Interfaces
4	David Jones	Mathematics	A1	MCS	Mathematics for Computer Scientists

Warm up: Data Redundancy

sID	Name	Department	Building	mCode	Module
1	John Smith	Computer Science	B1	DBI	Database and Interfaces
1	John Smith	Computer Science	B1	FAI	Foundation of Artificial Intelligence
2	Mark Brown	Computer Science	B1	FAI	Foundation of Artificial Intelligence
3	Mary Jones	Computer Science	B1	PGA	Programming and Algorithms
3	Mary Jones	Computer Science	B1	DBI	Database and Interfaces
4	David Jones	Mathematics	A1	MCS	Mathematics for Computer Scientists

Learning Outcomes

Learning Outcomes

By the end of this lecture, you should be able to

- Understand the basic idea of **normalization**.
- Understand the definition of **functional dependencies**.
- Know what are **Normal Forms**.
- Translate given relations into **1NF**.
- Understand the problems of **1NF**.

Normalization

What is normalization?

- Normalization: A technique for producing a set of relations with desirable properties, given the data requirement.
- Desirable properties/requirements:
  - Minimal number of attributes to support the data requirement.
  - Attributes with a close logical relationship should be defined in the same relation.
  - Minimal redundancy.
- Why we need it?
  - Easier to access and maintain data.
  - Minimal storage space.
- Normalization steps: 1NF, 2NF and 3NF

1NF

Normalization to 1NF

- 1NF: The relation contains only atomic values.

sID	Name	Department	Building	mCode	Module
1	John Smith	Computer Science	B1	DBI, FAI	Database and Interfaces, Foundation of Artificial Intelligence
2	Mark Brown	Computer Science	B1	FAI	Foundation of Artificial Intelligence
3	Mary Jones	Computer Science	B1	PGA, DBI	Programming and Algorithms, Database and Interfaces
4	David Jones	Mathematics	A1	MCS	Mathematics for Computer Scientists

1NF

- To convert a relation into 1NF, we need to split all non-atomic values

sID	Name	Department	Building	mCode	Module
1	John Smith	Computer Science	B1	DBI	Database and Interfaces
1	John Smith	Computer Science	B1	FAI	Foundation of Artificial Intelligence
2	Mark Brown	Computer Science	B1	FAI	Foundation of Artificial Intelligence
3	Mary Jones	Computer Science	B1	PGA	Programming and Algorithms
3	Mary Jones	Computer Science	B1	DBI	Database and Interfaces
4	David Jones	Mathematics	A1	MCS	Mathematics for Computer Scientists

INSERT Anomalies

- If we want to add a new staff David Ford from Computer Science with ID = 5.

sID	Name	Department	Building	mCode	Module
1	John Smith	Computer Science	B1	DBI	Database and Interfaces
...	...	...	...	...	...
4	David Jones	Mathematics	A1	MCS	Mathematics for Computer Scientists
5	David Ford	Computer Science	Null	Null	Null

DELETION Anomalies

- If we want to delete module MCS

sID	Name	Department	Building	mCode	Module
1	John Smith	Computer Science	B1	DBI	Database and Interfaces
1	John Smith	Computer Science	B1	FAI	Foundation of Artificial Intelligence
2	Mark Brown	Computer Science	B1	FAI	Foundation of Artificial Intelligence
3	Mary Jones	Computer Science	B1	PGA	Programming and Algorithms
3	Mary Jones	Computer Science	B1	DBI	Database and Interfaces
4	David Jones	Mathematics	A1	MCS	Mathematics for Computer Scientists

DELETION Anomalies

- If we want to delete module MCS

sID	Name	Department	Building	mCode	Module
1	John Smith	Computer Science	B1	DBI	Database and Interfaces
1	John Smith	Computer Science	B1	FAI	Foundation of Artificial Intelligence
2	Mark Brown	Computer Science	B1	FAI	Foundation of Artificial Intelligence
3	Mary Jones	Computer Science	B1	PGA	Programming and Algorithms
3	Mary Jones	Computer Science	B1	DBI	Database and Interfaces

UPDATE Anomalies

- Mary Jones is now transferred to the department of Mathematics

sID	Name	Department	Building	mCode	Module
1	John Smith	Computer Science	B1	DBI	Database and Interfaces
1	John Smith	Computer Science	B1	FAI	Foundation of Artificial Intelligence
2	Mark Brown	Computer Science	B1	FAI	Foundation of Artificial Intelligence
3	Mary Jones	Computer Science	B1	PGA	Programming and Algorithms
3	Mary Jones	Computer Science	B1	DBI	Database and Interfaces
4	David Jones	Mathematics	A1	MCS	Mathematics for Computer Scientists

UPDATE Anomalies

- Mary Jones is now transferred to the department of Mathematics

sID	Name	Department	Building	mCode	Module
1	John Smith	Computer Science	B1	DBI	Database and Interfaces
1	John Smith	Computer Science	B1	FAI	Foundation of Artificial Intelligence
2	Mark Brown	Computer Science	B1	FAI	Foundation of Artificial Intelligence
3	Mary Jones	Mathematics	B1	PGA	Programming and Algorithms
3	Mary Jones	Mathematics	B1	DBI	Database and Interfaces
4	David Jones	Mathematics	A1	MCS	Mathematics for Computer Scientists

Solution: Decomposition

- Decomposition:
  - Decompose a large relation into smaller relations.
- Properties:
  - Lossless-join: any instance of the original relation can be identified in the smaller relations
  - Dependency preservation: all constraints still remain.
- How to do it?

Functional Dependencies

- Functional Dependencies (FD) is a link between two sets of attributes in a relation.

ID	First	Last
...	...	...

- A set of attributes A functionally determines another set B, ($A \rightarrow B$):
  - If whenever two rows of the relation have the same value for all the attributes in A, then they also have the same values for all the attributes in B.

- $\{ID\} \rightarrow \{First\}$?
- $\{ID\} \rightarrow \{First, Last\}$?
- $\{First\} \rightarrow \{Last\}$?
- $\{ID, First\} \rightarrow \{Last\}$?
- $\{First, Last\} \rightarrow \{ID\}$?

Excercise 1: FDs

sID	Name	Department	Building	mCode	Module
1	John Smith	Computer Science	B1	DBI	Database and Interfaces
1	John Smith	Computer Science	B1	FAI	Foundation of Artificial Intelligence
2	Mark Brown	Computer Science	B1	FAI	Foundation of Artificial Intelligence
3	Mary Jones	Computer Science	B1	PGA	Programming and Algorithms
3	Mary Jones	Computer Science	B1	DBI	Database and Interfaces
4	David Jones	Mathematics	A1	MCS	Mathematics for Computer Scientists

Why we care about FDs?

sID	Name	Department	Building	mCode	Module
...	...	...	...	...	...

Redundancy is ofthen caused by a functional dependency:

- $\{sID\} \rightarrow \{Name, Department\}$
- $\{Department\} \rightarrow \{Building\}$
- $\{mCode\} \rightarrow \{Module\}$

Normal Forms (e.g., 1NF, 2NF, 3NF):

- Each Normal Form has fewer FDs. (What does it mean?)
- Not all FDs cause a problem.
- Each NF removes a type of FD that is a problem.
- Need a way to remove FDs.

Properties of FDs

In any relation:

- The candidate keys functionally determine any set of attributes in that relation.
  - $K \rightarrow X$, where K is a candidate key, and X is a subset of attributes.
- Any set of attributes is FD on itself
  - $X \rightarrow X$ for any sets of attributes X.

Rules:

- Reflexivity:
  - If $B \subseteq A$, then $A \rightarrow B$.
- Argmentation:
  - If $A \rightarrow B$, then $A \cup C \rightarrow B \cup C$.
- Transitivity:
  - If $A \rightarrow B$ and $B \rightarrow C$, then $A \rightarrow C$.

Examples

sID	Name	Department	Building	mCode	Module
...	...	...	...	...	...

- *sID* and *mCode* together is the Primary Key.
  - $\{sID, mCode\} \rightarrow \{Name, Department, Building, Module\}$, ...
- Reflexivity:
  - $\{Name, Department\} \rightarrow \{Name\}$, ...
- Augmentation:
  - $\{mCode, sID\} \rightarrow \{Module, sID\}$
- Transitivity:
  - $\{sID\} \rightarrow \{Department\}$, $\{Department\} \rightarrow \{Building\}$, $\{sID\} \rightarrow \{Building\}$

Full vs Partial functional dependency

sID	Name	Department	Building	mCode	Module
...	...	...	...	...	...

- Full FDs:
  - $A \rightarrow B$ is a full FD, if there is no such $C \subset A$, $C \rightarrow B$
  - E.g., $mCode \rightarrow Module$
- Partial FDs:
  - $A \rightarrow B$ is a partial FD, if there exists a $C \subset A$, such that $C \rightarrow B$
  - E.g., $\{sID, Name\} \rightarrow \{Department\}$

Summary

Summary

- Why we want to do normalization?
- What is normalization.
- What are normal forms.
- What is functional dependency.
- How to convert an unnormalised relation into 1NF.
- We will learn 2NF and 3NF and how to convert relations into them.