(T45)討論 DatabaseNormalization
CourseGUID: e48417fc-9db5-4e99-822c-706c5ccef6cc
==========
(T45)討論 DatabaseNormalization
==========
1. Normalization
1NF Eliminate Repeating Groups
2NF Eliminate Redundant Data
3NF Eliminate Non-Dependent Columns
4NF No Independent Multiple Relationships
5NF Isolate Semantically Related Relationships
==========

1. Normalization

Normalization can educe data storage and data redundancy by making sure any given piece of data is stored only once.
There are normally 5 or 6 techniques (5NF or 6NF) to do the fully normalizing a database design.
The 5 normal forms are shortened as 1NF, 2NF, 3NF, 4NF, and 5NF, many people consider 3NF is sufficiently practical.

- 1NF: Eliminate Repeating Groups
- 2NF: Eliminate Redundant Data
- 3NF: Eliminate Non-Dependent Columns
- 4NF: No Independent Multiple Relationships
- 5NF: Isolate Semantically Related Relationships

1NF Eliminate Repeating Groups

Reference:
http://www.gitta.info/LogicModelin/en/html/DataConsiten_Norm1NF.html

1 NF must eliminate repeating groups which means no duplicate records in the table, no multi-valued column, and the same column always contain the same data type.

Gamer Table

Name	Magic
Name01	FireMagic, WaterMagic, WoodMagic
Name02	EarthMagic, MetalMagic
Name03	FireMagic

Magic column is currently a multi-valued column, thus, it need to be separated.

Gamer Table

ID	Name
1	Name01
2	Name02
3	Name03

GamerMagic Table

GamerID	Magic
1	FireMagic
1	WaterMagic
1	WoodMagic
2	EarthMagic
2	MetalMagic
3	FireMagic

2NF Eliminate Redundant Data

2NF has to firstly meet all conditions of 1NF.

2NF also has to move Redundant Data to a **separate table**
,then create **relationship between** these **tables** using **foreign keys**.
In our example, moving Magic to its own table also achieved 2NF.

Gamer Table

Name	Magic
Name01	FireMagic, WaterMagic, WoodMagic
Name02	EarthMagic, MetalMagic
Name03	FireMagic

2NF also has to move Redundant Data, Magic, to a separate table
,then create relationship between these tables using **foreign keys**.

Gamer Table

ID	Name
1	Name01
2	Name02
3	Name03

GamerMagic Table

GamerID	Magic
1	FireMagic
1	WaterMagic
1	WoodMagic
2	EarthMagic
2	MetalMagic
3	FireMagic

3NF Eliminate Non-Dependent Columns

Reference:
https://www.tutorialcup.com/dbms/third-normal-form.htm

3NF has to firstly meet all conditions of 2NF.
3NF also has to remove Non-Dependent Columns
that means table should not contain columns that are not fully dependent upon the primary key.

GamerTeam Table

GamerID	GamerName	TeamID	TeamName
1	Name01	1	Team1
2	Name02	1	Team1
3	Name03	2	Team2
4	Name04	2	Team2

GamerID is Primary key, but TeamName column are not fully dependent upon the primary key.
Thus, 3NF has to move TeamName to other table.
The TeamName is actually depending on TeamID.

Gamer Table

GamerID	GamerName	TeamID
1	Name01	1
2	Name02	1
3	Name03	2
4	Name04	2

Team Table

TeamID	TeamName
1	Team1
2	Team2

4NF No Independent Multiple Relationships

Reference:
https://www.tutorialcup.com/dbms/fourth-normal-form.htm

4NF has to firstly meet all conditions of 3NF.
4NF also has to remove multi-valued dependencies.

Magic Table

MagicID	MagicName	MagicType	MagicSpell
1	FireMagic	Fire	Fire Go Go
2	WaterMagic	Water	Water Go Go
3	WoodMagic	Wood	Wood Go Go

This satisfies 3NF,
but the MagicType and the MagicSpell are really 2 independent entities here.
There is no relationship between MagicType and MagicSpell.
4NF also has to remove multi-valued dependencies which means we have to separate MagicType and MagicSpell.

MagicType Table

MagicID	MagicName	MagicType
1	FireMagic	Fire
2	WaterMagic	Water
3	WoodMagic	Wood

MagicSpell Table

MagicID	MagicName	MagicSpell
1	FireMagic	Fire Go Go
2	WaterMagic	Water Go Go
3	WoodMagic	Wood Go Go

Too much normalization might reduce performance, as SQL must repeatedly join a lot of tables.
Many people think 3NF is good enough.

5NF Isolate Semantically Related Relationships

Reference:
https://www.tutorialcup.com/dbms/fifth-normal-form.htm

5NF has to firstly meet all conditions of 4NF.
5NF also has to isolate Semantically Related Relationships.

Magic Table

MagicID	MagicName	MagicType	MagicSpell
1	FireMagic	Fire	Fire Go Go
2	WaterMagic	Water	Water Go Go
3	WoodMagic	Wood	Wood Go Go

This satisfies 3NF,
but the MagicType and the MagicSpell are really 2 independent entities here.
There is no relationship between MagicType and MagicSpell.
4NF also has to remove multi-valued dependencies which means we have to separate MagicType and MagicSpell.

MagicType Table

MagicID	MagicName	MagicType
1	FireMagic	Fire
2	WaterMagic	Water
3	WoodMagic	Wood

MagicSpell Table

MagicID	MagicName	MagicSpell
1	FireMagic	Fire Go Go
2	WaterMagic	Water Go Go
3	WoodMagic	Wood Go Go

This satisfies 4NF,

but we still can see the magic Id is appeared in both tables.

5NF also has to isolate Semantically Related Relationships.

Magic Table

MagicID	MagicName
1	FireMagic
2	WaterMagic
3	WoodMagic

MagicType Table

MagicName	MagicType
FireMagic	Fire
WaterMagic	Water
WoodMagic	Wood

MagicSpell Table

MagicName	MagicSpell
FireMagic	Fire Go Go
WaterMagic	Water Go Go
WoodMagic	Wood Go Go

Too much normalization might reduce performance, as SQL must repeatedly join a lot of tables.

Many people think 3NF is good enough.