Normal Forms

The process of structuring tables and table relationships in a logical way that minimizes data redundancy.

3 rules or steps in normalization
- 1st normal form
- 2nd normal form
- 3rd normal form

How is first normal form violated?

Violates first normal form

Student

ID	Last	First	Grades	Classes
1	Wood	Bob	C, B	Geog357, Geog20
2	Kent	Chuck	B, D	Geog115, Geog356
3	Smith	Jane	A, B	Geog357, Geog20
4	Boone	Dan	B, A	Geog357, Geog455

Solution: Create new records

2nd Normal Form

2NF if: *it is in 1NF and if every non-key attribute is functionally dependent on the primary key*

What is functional dependency?
If you know an item (or items) for a row, then you automatically know a second set of items for the row

Notation:
A -> B (if we know A, then we know B)
e.g., if we know the person's name, then we know the address

What is the 2nd normal form violation?

Hint: start by listing the functional dependencies.

Student

primary key

Year	Last	First	Grade	Status
2	Wood	Bob	C	sophomore
4	Kent	Chuck	B	senior
3	Smith	Jane	A	junior
3	Boone	Dan	B	junior

Year/Last/First -> Grade

Year -> Status

Violates second normal form because *Status* is dependent only on *Year*, not on *Year/Last/First*

What's the solution?

Solution

Create separate tables:

Student

primary key

Last	First	Grade	Year
Wood	Bob	C	2
Kent	Chuck	B	4
Smith	Jane	A	3
Boone	Dan	B	3

Year

primary key

Year	Status
1	freshman
2	sophomore
3	junior
4	senior

3rd Normal Forms

Remove *transitive* functional dependencies

A transitive functional dependency is when
$A \to B$ (if we know A, then we know B)
and
$B \to C$ (if we know B, then we know C)
So
$A \to C$ (if we know A, then we know C).

To be in 3NF, we must identify all transitive functional dependencies and remove them, typically by splitting the table(s) that contain them
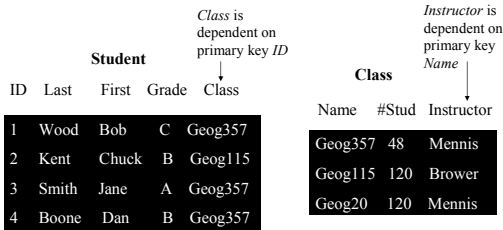
What is the 3rd normal form violation?

Student

ID	Last	First	Grade	Class	Instructor
1	Wood	Bob	C	Geog357	Mennis
2	Kent	Chuck	B	Geog115	Brower
3	Smith	Jane	A	Geog357	Mennis
4	Boone	Dan	B	Geog357	Mennis

ID -> Last, First,
    Grade, Class

Class -> Instructor

Solution

Create separate tables:

Class is dependent on primary key *ID*

Instructor is dependent on primary key *Name*

Student

ID	Last	First	Grade	Class
1	Wood	Bob	C	Geog357
2	Kent	Chuck	B	Geog115
3	Smith	Jane	A	Geog357
4	Boone	Dan	B	Geog357

Class

Name	#Stud	Instructor
Geog357	48	Mennis
Geog115	120	Brower
Geog20	120	Mennis

Normal Forms Summary

No repeat columns (create new records such that there are multiple records per entry).

All non-key attributes depend on a primary key (split the tables).

No transitive functional dependencies (split tables further).