

Beginning Relational Data Modeling



Joe Cline

SENIOR DATA ENGINEER & MODELER

@d8ajoe www.linkedin.com/in/josephcline



In This Module



Brief explanation of the relational model

- Short history
- Data domains
- Cardinality

Discovering Normalization

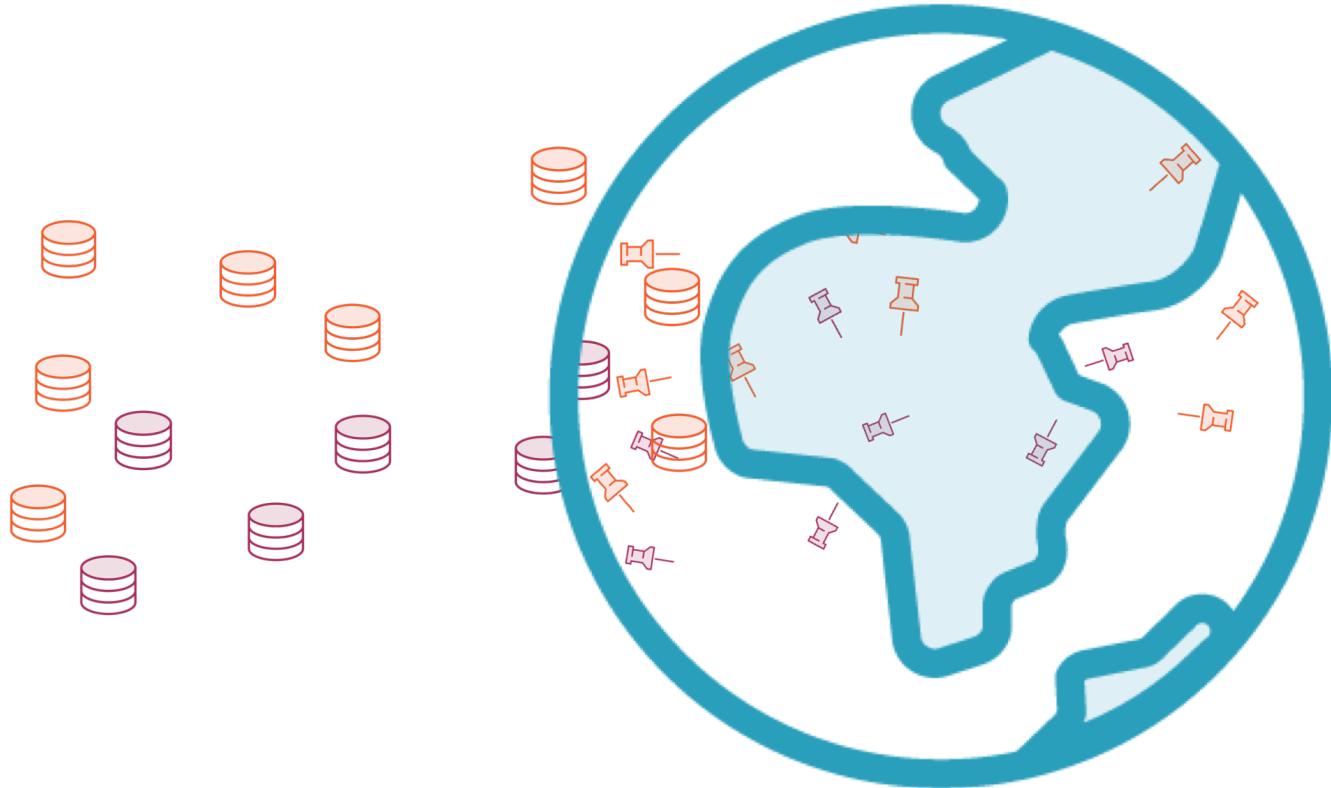
- Levels of normal form (NF)
- Describe 1NF, 2NF and 3NF

Demonstration

- Using a modeling tool
- Applying 1NF, 2NF and 3NF











Migrating to the Palm Breeze Chain



Analyze the schema from each database

Understand the underlying relationships

Translate naming conventions



<video 2 starts here - this slide will not be recorded>



The Relational Data Model



Databases prior to the Relational Model had few standards



First described by E.F. Codd around 1970



Implements the mathematical Set Logic

(a, b, c)

Organized data into tuples, also called rows



Data as Sets and Subsets

Enterprise data



Subject matter data domain



Hotel data

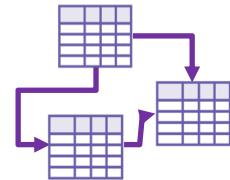
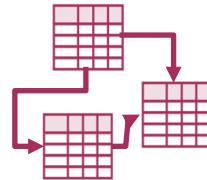
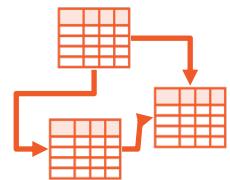


Franchisee data



Guest data

Entities



(“street_address”, “front_desk_phone_number”, “number_of_floors”, “number_of_rooms”)

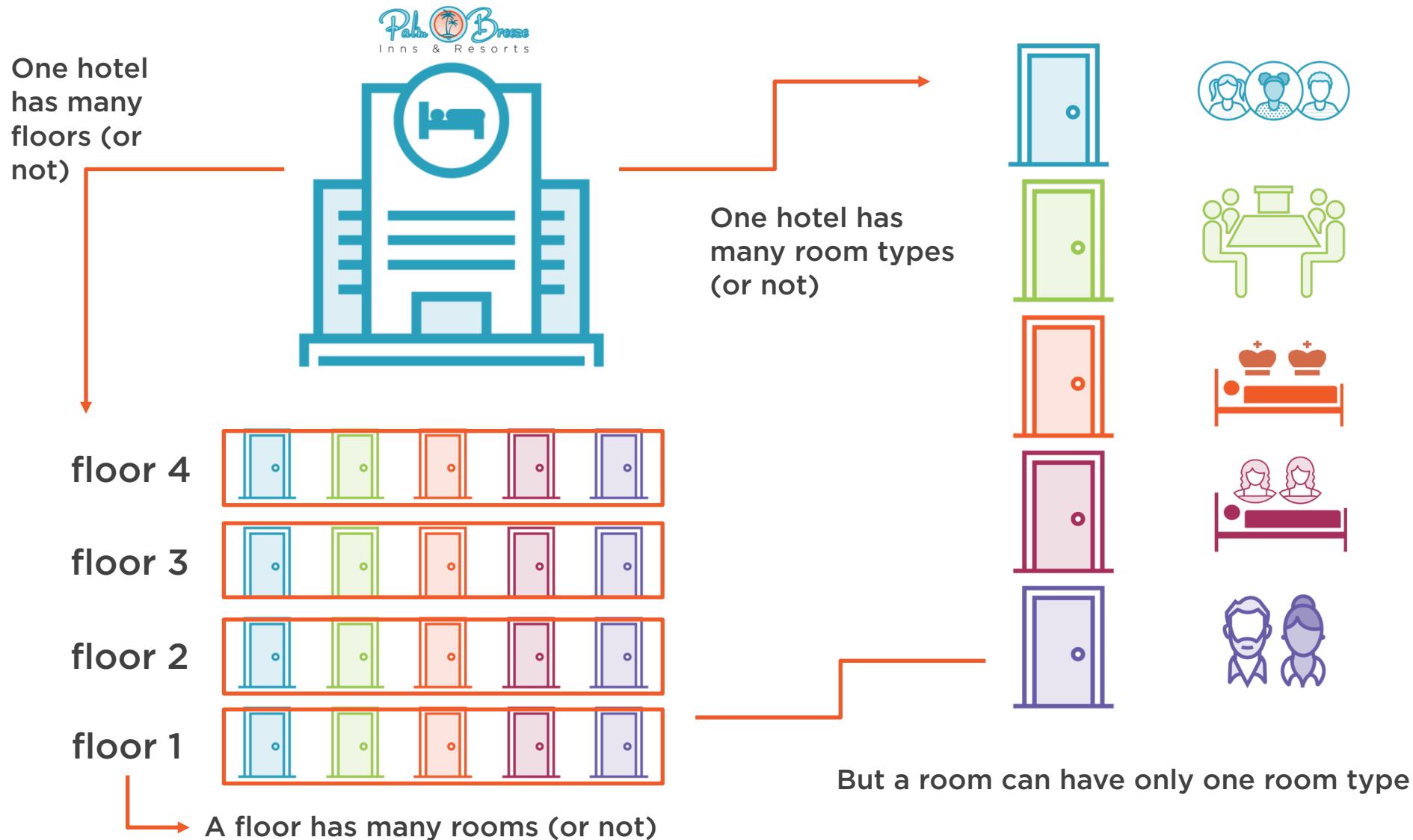
Tuples

(“street_address”, “phone_number”, “company_name”, “product_or_service”)

(“home_address”, “phone_number”, “first_name”, “last_name”)

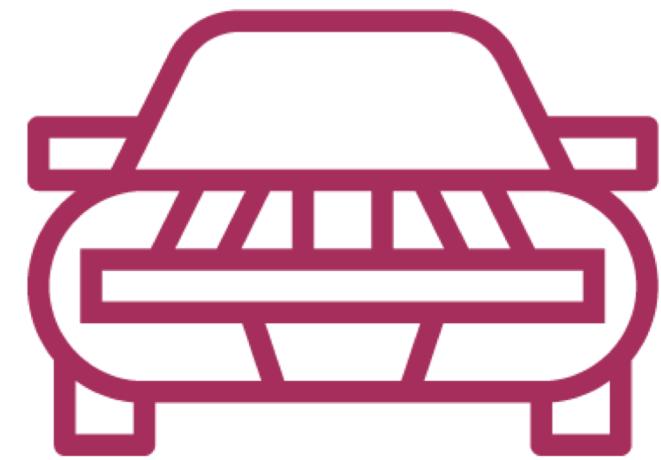
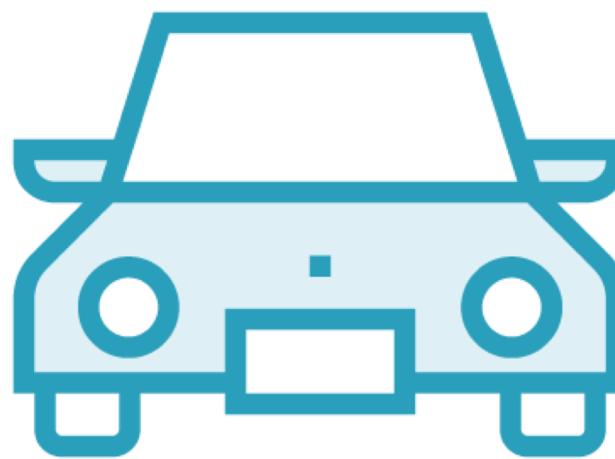


Cardinality Review





Parking at the Beach



Parking Problems - One-To-One

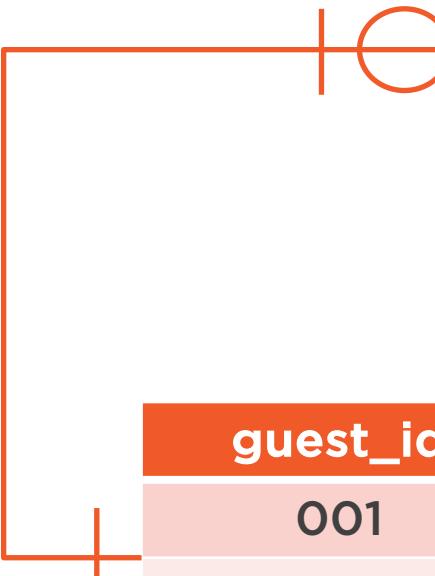
guest_id	first_name	...
001	Bob	...
002	Indera	...
003	Maria	...

parking_id	lot
12	A
13	A
14	A

guest_id	license	parking_id	make	...
001	XNE119	14	Chev	...
002	FSO934	12	Ford	...
003	PPS536	13	Merc	...



Parking Problems - Zero/One-To-One



guest_id	first_name	...
001	Bob	...
002	Indera	...
003	Maria	...
004	Raj	...

guest_id	license	parking_id	make	...
001	XNE119	14	Chev	...
002	FSO934	12	Ford	...
003	PPS536	13	Merc	...



Reservations- One-To-Many

guest_id	first_name	...
001	Bob	...
002	Indera	...
003	Maria	...
004	Raj	...

confirmation_id	guest_id	arrival	departure	...
10001	002	12/22/2017	12/27/2017	...
10002	001	12/24/2017	1/2/2018	...
...
10099	002	3/08/2018	3/22/2018	



Easy peasy!



Supertypes and Subtypes

confirmation_id	guest_id	arrival	departure	status
10001	002	12/22/2017	12/27/2017	R
10002	001	12/24/2017	1/2/2018	C
...	R
10099	002	3/08/2018	3/22/2018	R



confirmation_id	folio_id	payment_on_file	status
10001	00012345	Visa	I
10002	00012893	Mastercard	I
...	O
10099	00120304	Visa	I



Normalization Notation

1NF

First normal form

2NF

Second normal form

3NF

Third normal form, etc.



Normalization Notation

EKNF

Elementary key normal form

BCNF

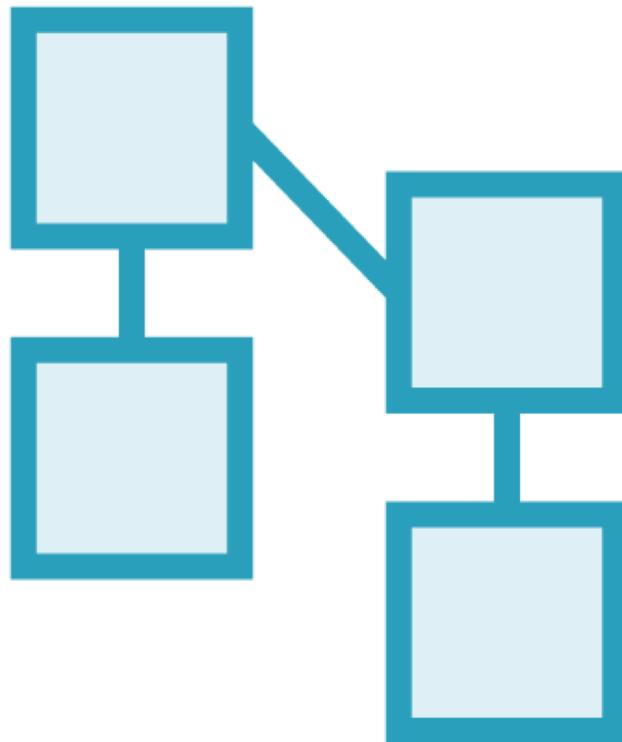
Boyce-Codd normal form

DKNF

Domain key normal form



Levels of Normal Form



UNF - Unnormalized

1NF - First

2NF - Second

3NF - Third

EKNF - Elementary-key

BCNF - Boyce-Codd

4NF - Fourth

ETNF - Essential tuple

5NF - Fifth

DKNF - Domain-key

6NF - Sixth



<video 3 starts here - this slide will not be recorded >



Why Normalize Data?

The Relational Data Model



Eliminate redundancies
Eliminate insert anomalies

Staff Name	Staff Hire Date	Dept Number
Hector	14-Feb-2011	GK-001
Mark	04-Apr-2013	GK-002

Marcus	16-Dec-2014
--------	-------------

?

Why Normalize Data?

The Relational Data Model



Eliminate redundancies

Eliminate insert and update anomalies

Staff Name	Staff Hire Date	Dept Number
Hector	14-Feb-2011	GK-001
Mark	04-Apr-2013	GK-002
Marcus	16-Dec-2014	FB-023

Why Normalize Data?

The Relational Data Model



Eliminate redundancies

Eliminate insert, update and delete anomalies

Staff ID	Staff Name	Staff Hire Date	Dept Number
002	Mark	04-Apr-2013	GK-002
002	Mark	16-Dec-2014	FB-023

Why Normalize Data?

The Relational Data Model



- Eliminate redundancies
- Eliminate insert, update and delete anomalies
- Several levels normal form (1NF, 2NF, 3NF, ...)

Staff Name	Staff Hire Date	Dept Number
Hector	14-Feb-2011	00-000
Mark	04-Apr-2013	GK-002
Marcus	16-Dec-2014	FB-023

Default Value

The more normalized your data, the more entities and relationships you will have



UNF

Unnormalized Form

Data has no specified shape

Data is not in tuples (rows)

Data doesn't have the same attributes

Data doesn't have the same number of attributes

There might be tables or lists within tables or lists of data



First Normal Form

1NF

Data should be organized into rows



1NF: Organized into Rows

confermation_id	guest_id	arrival	departure	status
10001	002	12/22/2017	12/27/2017	R
10002	001	12/24/2017	1/2/2018	C
...	R
10099	002	3/08/2018	3/22/2018	R



First Normal Form

1NF

Data should be organized into rows

Column values must be of the same kind



1NF: Column Values Must Be Same Kind

confermation_id	guest_id	arrival	departure	status
10001	002	12/22/2017	12/27/2017	R
10002	001	12/24/2017	1/2/2018	C
...	R
10099	002	3/08/2018	3/22/2018	R

Big Integer Big Integer Date Date Char



1NF

First Normal Form

Data should be organized into rows

Column values must be of the same kind

Each row should have a primary key



1NF: Each Row Should Have a Primary Key

confermation_id	guest_id	arrival	departure	status
10001	002	12/22/2017	12/27/2017	R
10002	001	12/24/2017	1/2/2018	C
...	R
10099	002	3/08/2018	3/22/2018	R



1NF: Each Row Should Have a Primary Key

confermation_id	guest_id	arrival	departure	status
10001	002	12/22/2017	12/27/2017	R
10002	001	12/24/2017	1/2/2018	C
...	R
10099	002	3/08/2018	3/22/2018	R



1NF

First Normal Form

Data should be organized into rows

Column values must be of the same kind

Each row should have a primary key

No repeating groups



1NF: No Repeating Groups

confermation_id	guest_id	arrival	departure	status
10001	002	12/22/2017	12/27/2017	R, I, O
10002	001	12/24/2017	1/2/2018	R, C
...	R, I, O
10099	002	3/08/2018	3/22/2018	R



1NF: No Repeating Groups

confermation_id	guest_id	arrival	departure	status
10001	002	12/22/2017	12/27/2017	P, I, O
10002	001	12/24/2017	1/2/2018	R, C
...	R, I, O
10099	002	3/08/2018	3/22/2018	R



Just because you can
doesn't mean you should



1NF: No Repeating Groups

confermation_id	guest_id	arrival	departure	status
10001	002	12/22/2017	12/27/2017	"R I, O"
10002	001	12/24/2017	1/2/2018	"R, C"
...	"R, I, O"
10099	002	3/08/2018	3/22/2018	"R"



1NF: No Repeating Groups

confermation_id	guest_id	arrival	departure	status
10001	002	12/22/2017	12/27/2017	"P"
10002	001	12/24/2017	1/2/2018	"R"
...
10099	002	3/08/2018	3/22/2018	"R"

Ugh!



Don't, just...don't



1NF: No Repeating Groups

confermation_id	guest_id	arrival	departure	status_1	status_2	status_3
10001	002	12/22/2017	12/27/2017	R	I	O
10002	001	12/24/2017	1/2/2018	R	C	
...	R	I	O
10099	002	3/08/2018	3/22/2018	R		



1NF: No Repeating Groups

confermation_id	guest_id	arrival	departure	status_1	status_2	status_3
10001	002	12/22/2017	12/27/2017	R	I	O
10002	001	12/24/2017	1/2/2018	R	C	
...	R	I	O
10099	002	3/08/2018	3/22/2018	R		



Second Normal Form

2NF

Be in 1NF

No partial dependency on the key



“The key, whole key and nothing
but the key, so help me Codd”

Unknown

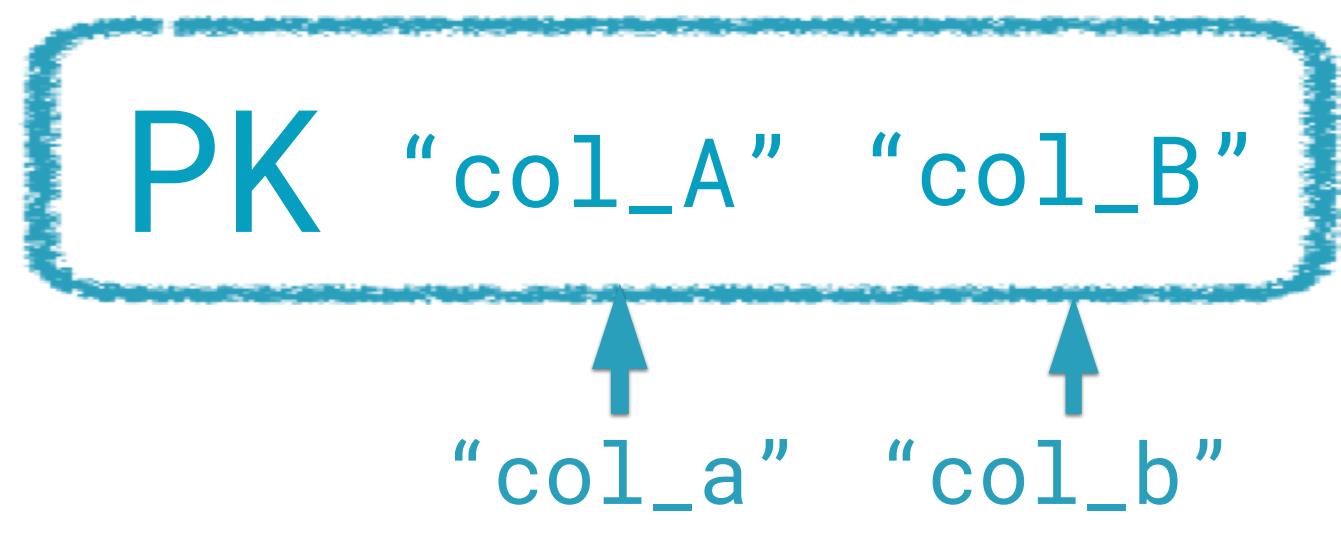


Second Normal Form

2NF

Be in 1NF

No partial dependency on the key

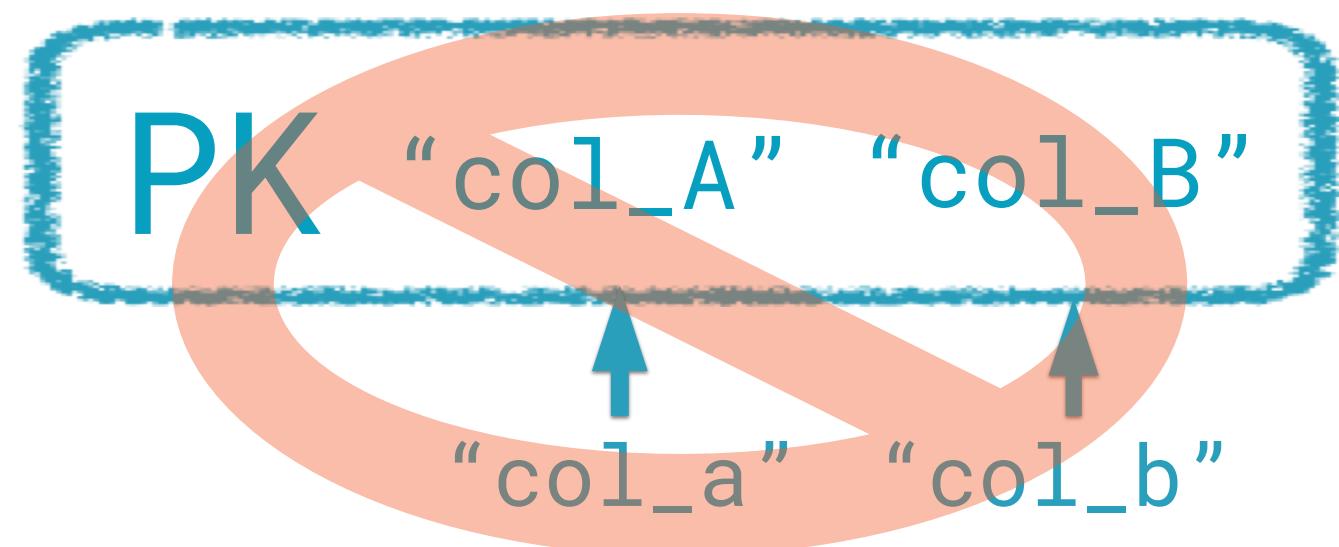


Second Normal Form

2NF

Be in 1NF

No partial dependency on the key



3NF

Third Normal Form

Must be in 2NF

Have no transitive dependencies

PK ← “col_a” ← “col_b”

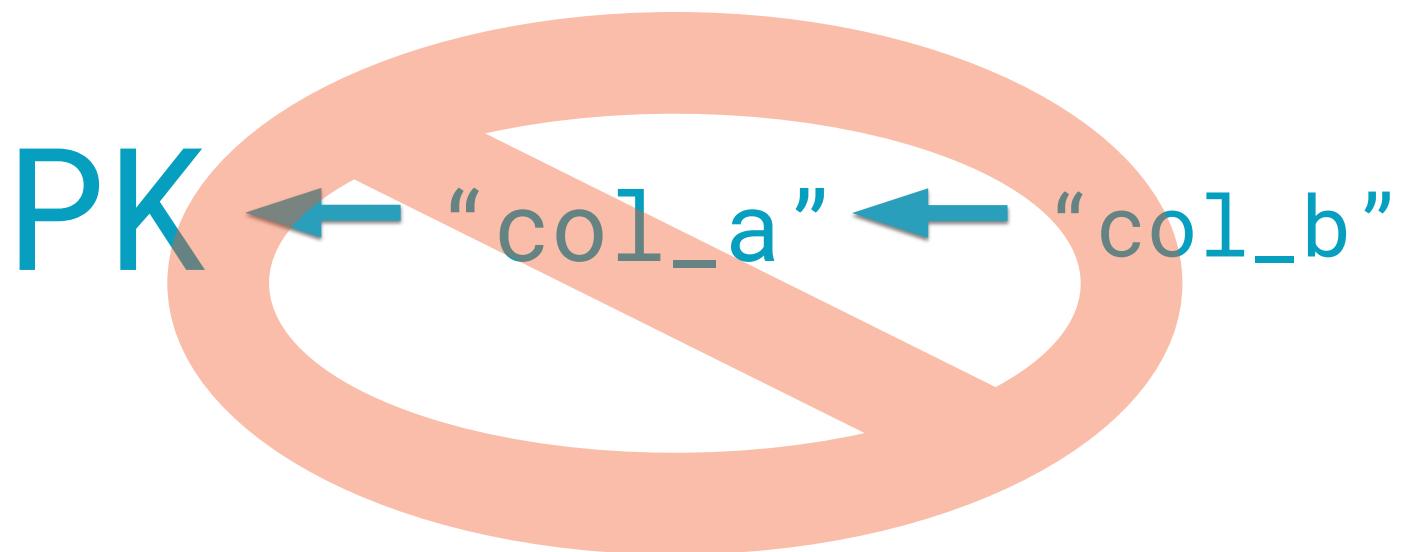


3NF

Third Normal Form

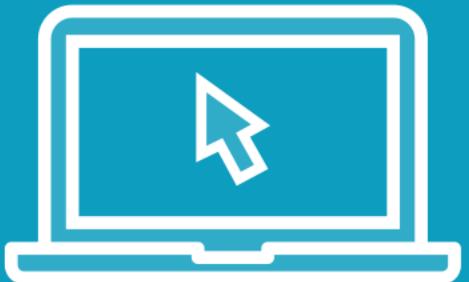
Must be in 2NF

Have no transitive dependencies



Why only the first three
levels?

Demo



Let's create a relational logical data model and normalize it to 3NF



Summary



Short history

Data domains

Cardinality

Levels of normal form (NF)

Describe 1NF, 2NF and 3NF

Demo using a modeling tool to apply normalization techniques to unstructured data



Next: Beginning NoSQL Modeling

