Data Modeling × One Big Table

Movie								
Title	Language	ReleaseDate	RunningTime	Genre	FirstName	LastName	DateOfBirth	PlaceOfBirth
2001: A Space Odyssey	*English*	*1968*	*142*	*Sci-fi, Adventure*	*Stanley*	*Kubrick*	*Jul. 26, 1928*	*USA*
The Shining	*English*	*1980*	*144 (US), 119 (EU)*	*Drama, Horror,*	*Stanley*	*Kubrick*	*Jul. 26, 1928*	*USA*
A Clockwork Orange	*English*	*1971*	*136*	*Crime, Drama, Sci-Fi*	*Stanley*	*Kubrick*	*Jul. 26, 1928*	*USA*
The Birds	*English*	*1963*	*119*	*Drama, Horror*	*Alfred*	*Hitchcock*	*Aug. 13, 1899*	*England*
Psycho	*English*	*1960*	*109*	*Horror, Mystery, Thriller*	*Alfred*	*Hitchcock*	*Aug. 13, 1899*	*England*

Forget about data modeling in RDBMS
i.e., conceptual (ER), logical (relational) & physical (SQL) levels

Data Modeling × Anomaly

Anomaly | Inconsistency

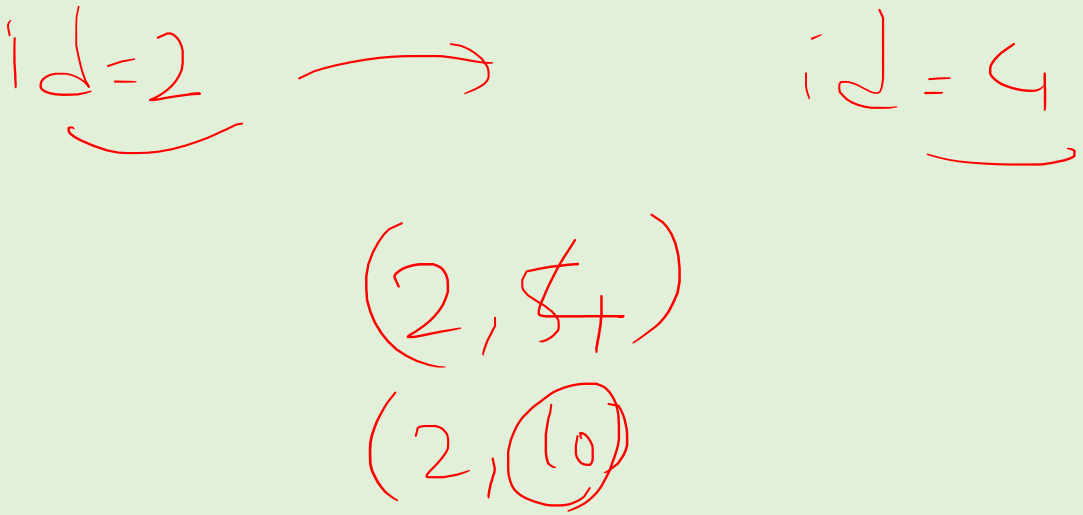
Something that deviates from our expectations

To avoid anomaly in RDBMS

Data Integrity | Integrity Constraints

SQL × DML × Data Integrity

I) Domain Integrity
II) Entity Integrity
III) Referential Integrity
IV) User-defined Integrity



Data Modeling × Normalization

To avoid anomaly in RDBMS

Table decomposition (normalization) to minimize redundancy and improve data integrity.

Data Modeling × Normalization

Movie								
Title	Language	ReleaseDate	RunningTime	Genre	FirstName	LastName	DateOfBirth	PlaceOfBirth
2001: A Space Odyssey	*English*	*1968*	*142*	*Sci-fi, Adventure*	*Stanley*	*Kubrick*	*Jul. 26, 1928*	*USA*
The Shining	*English*	*1980*	*144 (US), 119 (EU)*	*Drama, Horror,*	*Stanley*	*Kubrick*	*Jul. 26, 1928*	*USA*
A Clockwork Orange	*English*	*1971*	*136*	*Crime, Drama, Sci-Fi*	*Stanley*	*Kubrick*	*Jul. 26, 1928*	*USA*
The Birds	*English*	*1963*	*119*	*Drama, Horror*	*Alfred*	*Hitchcock*	*Aug. 13, 1899*	*England*
Psycho	*English*	*1960*	*109*	*Horror, Mystery, Thriller*	*Alfred*	*Hitchcock*	*Aug. 13, 1899*	*England*

Data Modeling × Normalization

Movie			
Id	Title	Language	ReleaseDate
1	2001: A Space Odyssey	1	1968
2	The Shining	1	1980
3	A Clockwork Orange	1	1971
4	The Birds	1	1963
5	Psycho	1	1960

Director				
Id	FirstName	LastName	DateOfBirth	PlaceOfBirth
1	Stanley	Kubrick	Jul. 26, 1928	USA
2	Alfred	Hitchcock	Aug. 13, 1899	England

Language	
Id	Title
1	English

Genre	
Id	Title
1	Sci-fi
2	Drama
3	Crime
4	Mystery
5	Thriller
6	Adventure
7	Horror

MovieGenre	
MovieId	GenreId
1	1
1	6
2	2
2	7
3	3
3	2
3	1
4	2
4	7
5	7
5	4
5	5

MovieDirector	
MovieId	DirectorId
1	1
2	1
3	1
4	2
5	2

MovieRunningTime		
MovieId	RunningTime	Scope
1	142	Globe
2	144	US
2	119	EU
3	136	Globe
3	119	Globe
3	109	Globe

Data Modeling × Normalization

Given a big table of all information, the process of decomposing it into tables in order to avoid redundancy and improve data integrity.

Machine-based!
Algorithm-based!

No conceptual level design. No E/R!
At conceptual or logical levels, we do not have actual data!

Data Modeling × Normalization

I) Functional Dependencies
II) Normal Forms

Data Modeling × Normalization

I) Functional Dependencies

II) Normal Forms

Functional Dependencies

A functional dependency occurs when the value of one (set of) attribute(s) determines the value of a second (set of) attribute(s)

Functional Dependencies

Movie								
Title	Language	ReleaseDate	RunningTime	Genre	FirstName	LastName	DateOfBirth	PlaceOfBirth
2001: A Space Odyssey	*English*	*1968*	*142*	*Sci-fi, Adventure*	*Stanley*	*Kubrick*	*Jul. 26, 1928*	*USA*
The Shining	*English*	*1980*	*144 (US), 119 (EU)*	*Drama, Horror,*	*Stanley*	*Kubrick*	*Jul. 26, 1928*	*USA*
A Clockwork Orange	*English*	*1971*	*136*	*Crime, Drama, Sci-Fi*	*Stanley*	*Kubrick*	*Jul. 26, 1928*	*USA*
The Birds	*English*	*1963*	*119*	*Drama, Horror*	*Alfred*	*Hitchcock*	*Aug. 13, 1899*	*England*
Psycho	*English*	*1960*	*109*	*Horror, Mystery, Thriller*	*Alfred*	*Hitchcock*	*Aug. 13, 1899*	*England*

Title → Title

(Title, ReleaseDate) → FirstName

(Title, ReleaseDate) → (FirstName, LastName)

Title → Genre

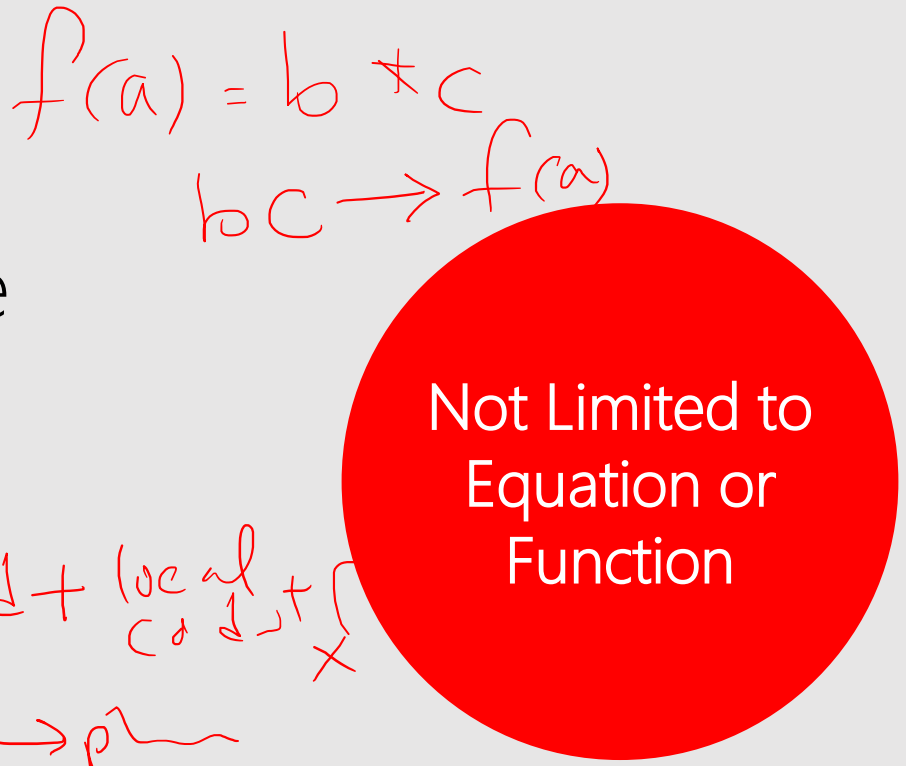
Genre ↛ Title

Functional Dependencies

Functional dependencies may be based on equations, e.g., in derived attributes:

TotalPrice = Quantity × UnitPrice
(Quantity, UnitPrice) → TotalPrice

But,



Functional Dependencies

Movie								
Title	Language	ReleaseDate	RunningTime	Genre	FirstName	LastName	DateOfBirth	PlaceOfBirth
2001: A Space Odyssey	*English*	*1968*	*142*	*Sci-fi, Adventure*	*Stanley*	*Kubrick*	*Jul. 26, 1928*	*USA*
The Shining	*English*	*1980*	*144 (US), 119 (EU)*	*Drama, Horror,*	*Stanley*	*Kubrick*	*Jul. 26, 1928*	*USA*
A Clockwork Orange	*English*	*1971*	*136*	*Crime, Drama, Sci-Fi*	*Stanley*	*Kubrick*	*Jul. 26, 1928*	*USA*
The Birds	*English*	*1963*	*119*	*Drama, Horror*	*Alfred*	*Hitchcock*	*Aug. 13, 1899*	*England*
Psycho	*English*	*1960*	*109*	*Horror, Mystery, Thriller*	*Alfred*	*Hitchcock*	*Aug. 13, 1899*	*England*
Psycho	*English*	*1998*	*104*	*Horror, Mystery, Thriller*	*Gus*	*Van Sant*	*July 24, 1952*	*USA*

Functional dependencies are based on the existing data:

- Title ↛ ReleaseDate
  Title → Genre
- Title ↛ FirstName, LastName

Functional Dependencies

Movie								
Title	Language	ReleaseDate	RunningTime	Genre	FirstName	LastName	DateOfBirth	PlaceOfBirth
2001: A Space Odyssey	*English*	*1968*	*142*	*Sci-fi, Adventure*	*Stanley*	*Kubrick*	*Jul. 26, 1928*	*USA*
The Shining	*English*	*1980*	*144 (US), 119 (EU)*	*Drama, Horror,*	*Stanley*	*Kubrick*	*Jul. 26, 1928*	*USA*
A Clockwork Orange	*English*	*1971*	*136*	*Crime, Drama, Sci-Fi*	*Stanley*	*Kubrick*	*Jul. 26, 1928*	*USA*
The Birds	*English*	*1963*	*119*	*Drama, Horror*	*Alfred*	*Hitchcock*	*Aug. 13, 1899*	*England*
Psycho	*English*	*1960*	*109*	*Horror, Mystery, Thriller*	*Alfred*	*Hitchcock*	*Aug. 13, 1899*	*England*
Psycho	*English*	*1998*	*104*	*Horror, Mystery, Thriller*	*Gus*	*Van Sant*	*July 24, 1952*	*USA*

Determinant might not be unique:

Title → Genre

FirstName → LastName

Functional Dependencies

Movie								
Title	Language	ReleaseDate	RunningTime	Genre	FirstName	LastName	DateOfBirth	PlaceOfBirth
2001: A Space Odyssey	*English*	*1968*	*142*	*Sci-fi, Adventure*	*Stanley*	*Kubrick*	*Jul. 26, 1928*	*USA*
The Shining	*English*	*1980*	*144 (US), 119 (EU)*	*Drama, Horror,*	*Stanley*	*Kubrick*	*Jul. 26, 1928*	*USA*
A Clockwork Orange	*English*	*1971*	*136*	*Crime, Drama, Sci-Fi*	*Stanley*	*Kubrick*	*Jul. 26, 1928*	*USA*
The Birds	*English*	*1963*	*119*	*Drama, Horror*	*Alfred*	*Hitchcock*	*Aug. 13, 1899*	*England*
Psycho	*English*	*1960*	*109*	*Horror, Mystery, Thriller*	*Alfred*	*Hitchcock*	*Aug. 13, 1899*	*England*
Psycho	*English*	*1998*	*104*	*Horror, Mystery, Thriller*	*Gus*	*Van Sant*	*July 24, 1952*	*USA*

But if a determinant is unique, then it is determinant of ALL other attributes.

RunningTime → Title, Language, …, PlaceOfBirth

ReleaseDate → Title, Language, …, PlaceOfBirth

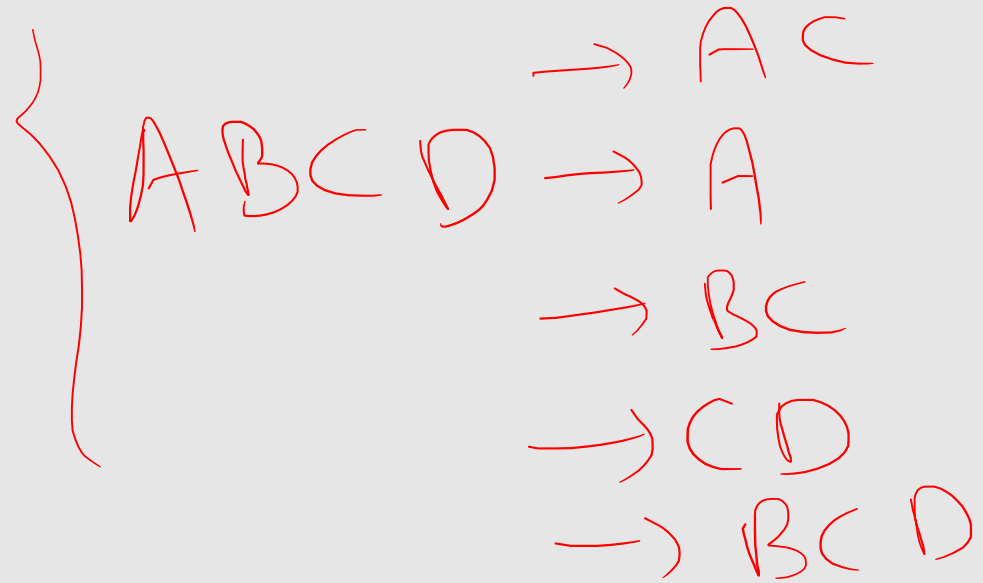
Functional Dependencies × Rules

IF	THEN	Not a complete list
A → BC	A → B A → C AB → C AC → B	
A → B B → C	A → C	Transitivity
AB → C	A ↛ C B ↛ C A ↛ BC	Be Careful!
A → B A → C	B ↛ C C ↛ B	Be Careful!

Functional Dependencies × Trivial

A functional dependency is trivial if it is satisfied by every tables

A → A
AB → A
AB → B



Generally, X → Y, where Y ⊆ X.

Trivial FD does not make a significant statement about real world constraints and we only interested in non-trivial FD's.

Functional Dependencies × Super Key

Super Key is a set of attributes that functionally determines ALL the attributes in a table.

Super Key is a set of attributes that identify an entity (row) uniquely.

The trivial Super Key is a set of all attributes of a table.

Functional Dependencies × Candidate Key

Candidate Key is a minimal Super Key.

Super Key is minimal if it is not possible to remove an attribute from it. Otherwise, it is not Super Key anymore.

Candidate Key is NOT a super key with smallest size!

Functional Dependencies × Candidate Key

Movie								
Title	Language	ReleaseDate	RunningTime	Genre	FirstName	LastName	DateOfBirth	PlaceOfBirth
2001: A Space Odyssey	*English*	*1968*	*142*	*Sci-fi, Adventure*	*Stanley*	*Kubrick*	*Jul. 26, 1928*	*USA*
The Shining	*English*	*1980*	*144 (US), 119 (EU)*	*Drama, Horror,*	*Stanley*	*Kubrick*	*Jul. 26, 1928*	*USA*
A Clockwork Orange	*English*	*1971*	*136*	*Crime, Drama, Sci-Fi*	*Stanley*	*Kubrick*	*Jul. 26, 1928*	*USA*
The Birds	*English*	*1963*	*119*	*Drama, Horror*	*Alfred*	*Hitchcock*	*Aug. 13, 1899*	*England*
Psycho	*English*	*1960*	*109*	*Horror, Mystery, Thriller*	*Alfred*	*Hitchcock*	*Aug. 13, 1899*	*England*
Psycho	*English*	*1998*	*104*	*Horror, Mystery, Thriller*	*Gus*	*Van Sant*	*July 24, 1952*	*USA*

Candidate Key (Minimal Super Key):

A) {Title, Language, …, DateOfBirth, PlaceOfBirth} ✗

B) {Title, FirstName} ✓

C) {RunningTime} ✓

D) {ReleaseDate} ✓

Functional Dependencies × Primary Key

Primary Key is a Candidate Key selected out of multiple Candidate Keys.

A) (Title, FirstName)
B) (RunningTime)
C) (ReleaseDate)

Best Practice: The best candidate for Primary Key:
I) Less #attributes AND
II) The attributes are mandatory (must have value)

Data Modeling × Normalization

I) Functional Dependencies

II) Normal Forms

Normalization × Normal Forms

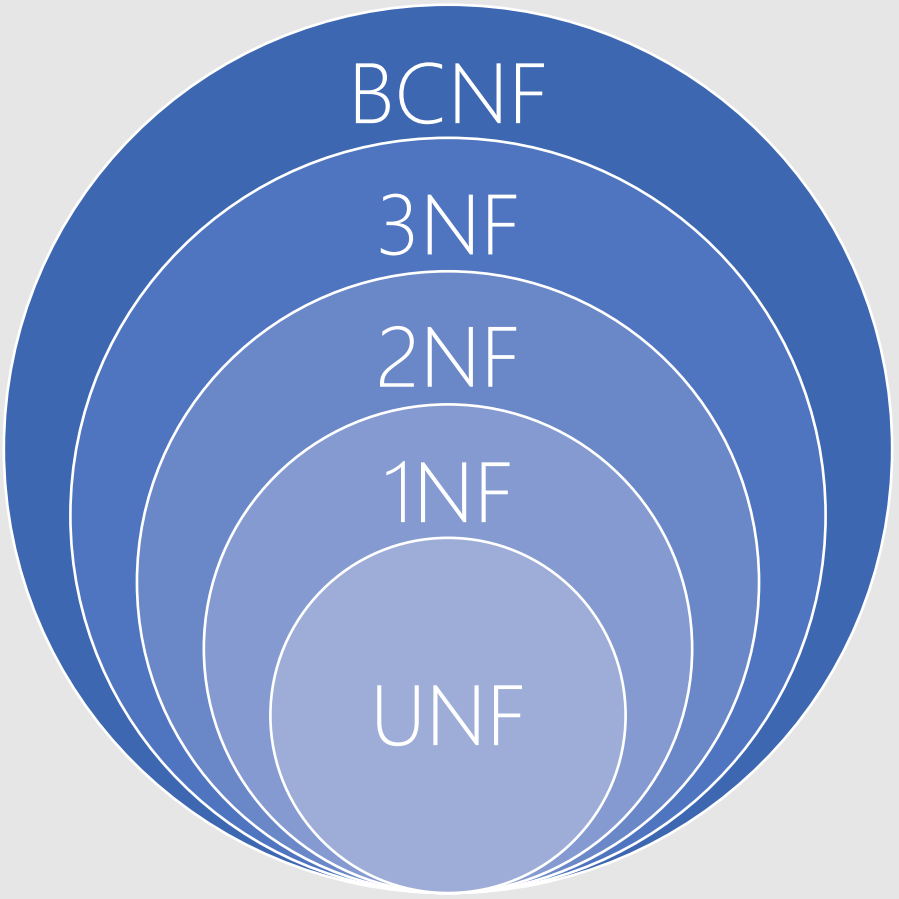
Normalization is done through decomposing tables based on series of normal forms.

There are 11 normal forms:

UNF (1970)	1NF (1971)	2NF (1971)	3NF (1971)	EKNF (1982)	BCNF (1974)	4NF (1977)	ETNF (2012)	5NF (1979)	DKNF (1981)	6NF (2003)

But we only consider UNF, 1NF, 2NF, 3NF and BCNF.

Normalization × Normal Forms

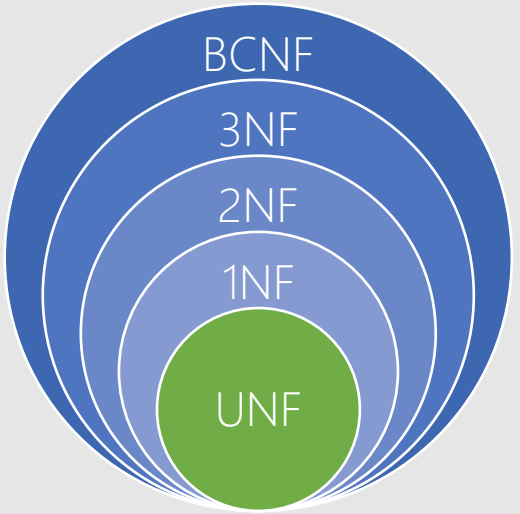


Normalization × UNF

Unnormalized Form is the initial normal form where there is no duplicate tuple (row) in a table.

Any table MUST have a Primary Key (Entity Integrity).

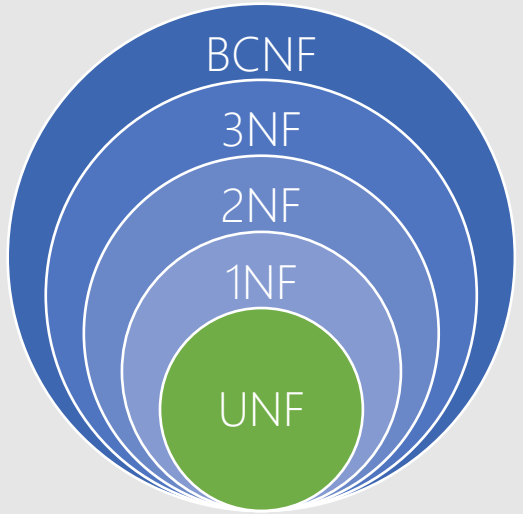
This normal form is usually taken for granted.



Normalization × UNF

Movie								
Title	Language	ReleaseDate	RunningTime	Genre	FirstName	LastName	DateOfBirth	PlaceOfBirth
2001: A Space Odyssey	*English*	*1968*	*142*	*Sci-fi, Adventure*	*Stanley*	*Kubrick*	*Jul. 26, 1928*	*USA*
The Shining	*English*	*1980*	*144 (US), 119 (EU)*	*Drama, Horror,*	*Stanley*	*Kubrick*	*Jul. 26, 1928*	*USA*
A Clockwork Orange	*English*	*1971*	*136*	*Crime, Drama, Sci-Fi*	*Stanley*	*Kubrick*	*Jul. 26, 1928*	*USA*
The Birds	*English*	*1963*	*119*	*Drama, Horror*	*Alfred*	*Hitchcock*	*Aug. 13, 1899*	*England*
Psycho	*English*	*1960*	*109*	*Horror, Mystery, Thriller*	*Alfred*	*Hitchcock*	*Aug. 13, 1899*	*England*
Psycho	*English*	*1998*	*104*	*Horror, Mystery, Thriller*	*Gus*	*Van Sant*	*July 24, 1952*	*USA*

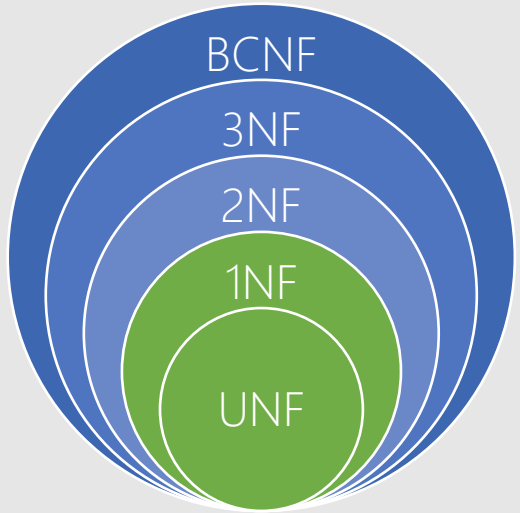
UNF ✓



Normalization × 1NF

1st Normal Form requires that the domain of each attribute contains only atomic (indivisible) values & UNF.

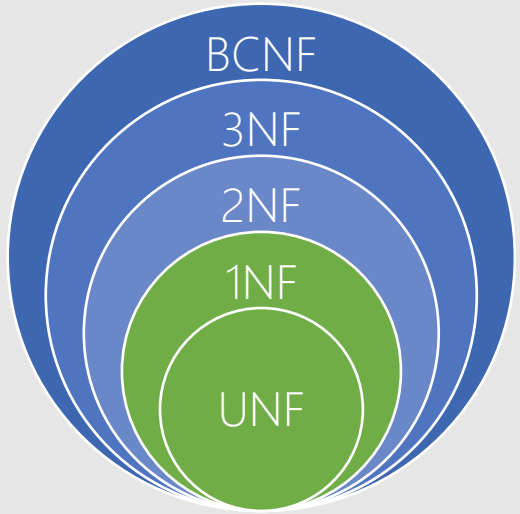
No composite attribute
No multivalued attribute



Normalization × 1NF

Movie								
Title	Language	ReleaseDate	RunningTime	Genre	FirstName	LastName	DateOfBirth	PlaceOfBirth
2001: A Space Odyssey	*English*	*1968*	*142*	*Sci-fi, Adventure*	*Stanley*	*Kubrick*	*Jul. 26, 1928*	*USA*
The Shining	*English*	*1980*	*144 (US), 119 (EU)*	*Drama, Horror,*	*Stanley*	*Kubrick*	*Jul. 26, 1928*	*USA*
A Clockwork Orange	*English*	*1971*	*136*	*Crime, Drama, Sci-Fi*	*Stanley*	*Kubrick*	*Jul. 26, 1928*	*USA*
The Birds	*English*	*1963*	*119*	*Drama, Horror*	*Alfred*	*Hitchcock*	*Aug. 13, 1899*	*England*
Psycho	*English*	*1960*	*109*	*Horror, Mystery, Thriller*	*Alfred*	*Hitchcock*	*Aug. 13, 1899*	*England*
Psycho	*English*	*1998*	*104*	*Horror, Mystery, Thriller*	*Gus*	*Van Sant*	*July 24, 1952*	*USA*

1NF ×



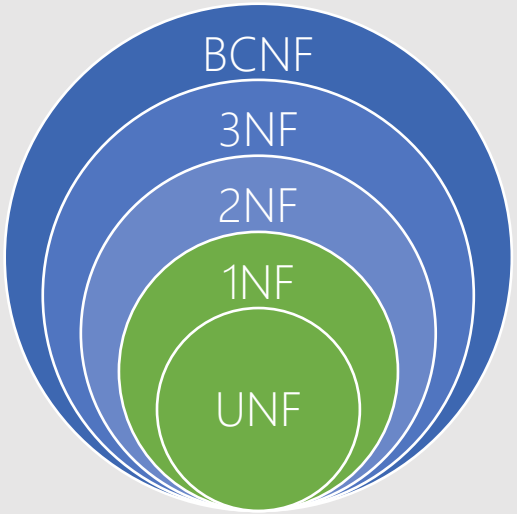
Normalization × 1NF

Movie											
Title	Genre1	Genre2	Genre3	RunningTime1	Scope1	RunningTime2	Scope2	MonthOfBirth	DayOfBirth	YearOfBirth	...
2001: A Space Odyssey	*Sci-fi*	*Adventure*		*142*	*Globe*			*Jul.*	*26*	*1928*	
The Shining	*Drama*	*Horror*		*144*	*US*	*119*	*EU*	*Jul.*	*26*	*1928*	
A Clockwork Orange	*Crime*	*Drama*	*Sci-Fi*	*136*	*Globe*			*Jul.*	*26*	*1928*	
The Birds	*Drama*	*Horror*		*119*	*Globe*			*Aug.*	*13*	*1899*	
Psycho	*Horror*	*Mystery*	*Thriller*	*109*	*Globe*			*Aug.*	*13*	*1899*	
Psycho	*Horror*	*Mystery*	*Thriller*	*104*	*Globe*			*July*	*24*	*1952*	

Normalization: Each Part, One Column

1NF ✓

What's the problem with this way of normalization?



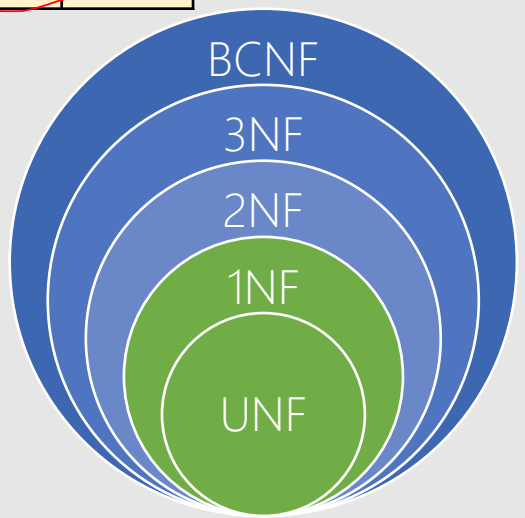
Normalization × 1NF

Movie						
Id	Title	Language	ReleaseDate	MonthOfBirth	DayOfBirth	YearOfBirth
1	2001: A Space Odyssey	English	1968	Jul.	26	1928
2	The Shining	English	1980	Jul.	26	1928
3	A Clockwork Orange	English	1971	Jul.	26	1928
4	The Birds	English	1963	Aug.	13	1899
5	Psycho	English	1960	Aug.	13	1899
6	Psycho	English	1998	104	24	1952

MovieGenre		
Id	MovieId	GenreId
1	1	1
2	1	6
3	2	2
4	2	7
5	3	3
6	3	2
7	3	1
8	4	2
9	4	7
10	5	7
11	5	4
12	5	5

Genre	
Id	Title
1	Sci-fi
2	Drama
3	Crime
4	Mystery
5	Thriller
6	Adventure
7	Horror

MovieRunningTime			
Id	MovieId	RunningTime	Scope
1	1	142	Globe
2	2	144	US
3	2	119	EU
4	3	136	Globe
5	3	119	Globe
6	3	109	Globe



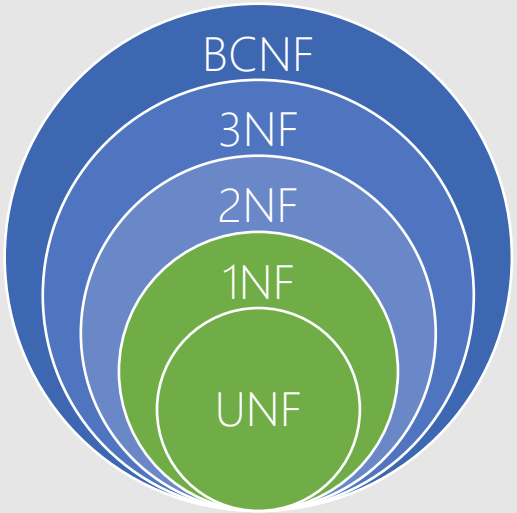
Normalization:

- Composite: Add Column
- Multivalued: One-2-Many or Many-2-Many

1NF ✓

Normalization × U,1 NF

U-1NF: Requiring existence of "the key" in the table



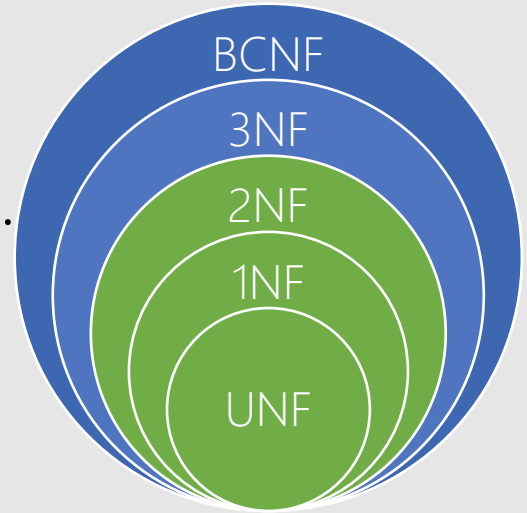
Normalization × 2NF

2nd Normal Form requires that a table:

I) Be in 1NF

II) Does NOT have any non-key attribute that is dependent on any proper subset of any candidate key of the table.

A non-key attribute of a table is an attribute that is not a part of any candidate key of the relation.



Normalization × 2NF

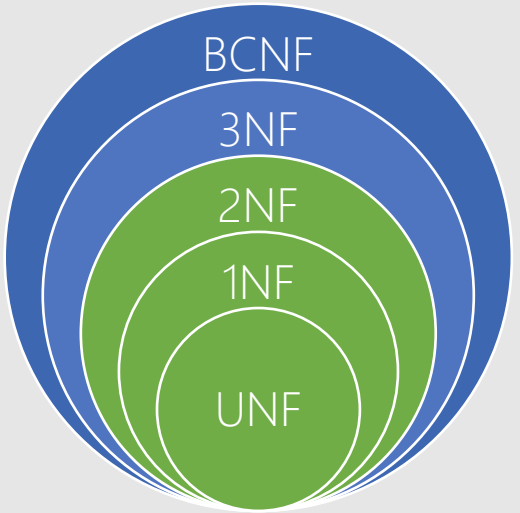
E.g., T(A, B, C, D, E, F)
CK1 = {A, B} i.e. AB→CDEF
CK2 = {C, D, E} i.e. CDE→ABF

AB → F
CDE → F

F is a non-key attribute since F∉CK1 and F∉CK2
If T is in 2NF, there must be none of these functional dependencies

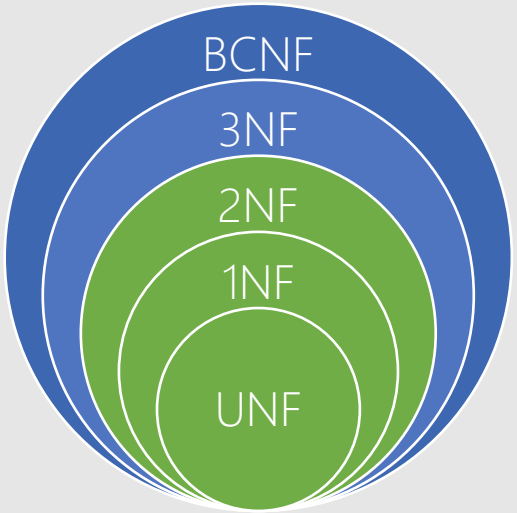
Subsets of CK1: {A}, {B}
A→F | B→F

Subsets of CK2: {C}, {D}, {E}, {CD}, {DE}, {CE}
C→F | D→F | E→F | CD→F | DE→F | CE→F



Normalization × U,1,2,3NF

U-1NF: Requiring existence of "the key" in the table
2NF: Requiring that non-key attributes be dependent on "the whole key"



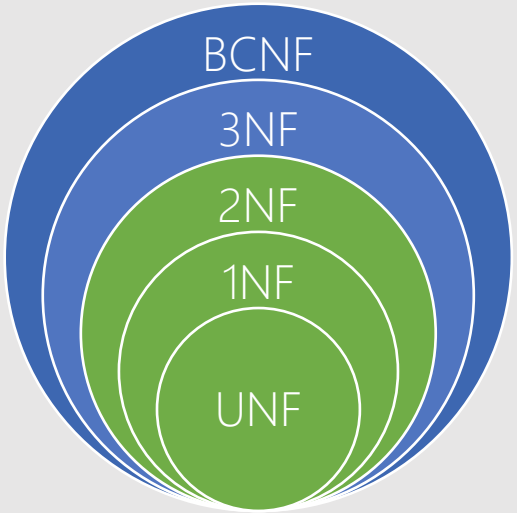
Normalization × 2NF

Movie					
Title	Language	ReleaseDate	FirstName	LastName	PlaceOfBirth
2001: A Space Odyssey	*English*	*1968*	*Stanley*	*Kubrick*	*USA*
The Shining	*English*	*1980*	*Stanley*	*Kubrick*	*USA*
A Clockwork Orange	*English*	*1971*	*Stanley*	*Kubrick*	*USA*
The Birds	*English*	*1963*	*Alfred*	*Hitchcock*	*England*
Psycho	*English*	*1960*	*Alfred*	*Hitchcock*	*England*
Psycho	*English*	*1998*	*Gus*	*Van Sant*	*USA*

Candidate Keys:
CK1={Title, FirstName}
CK2={RunningTime}
CK3={ReleaseDate}

Non-keys:
- Language
- LastName
- PlaceOfBirth

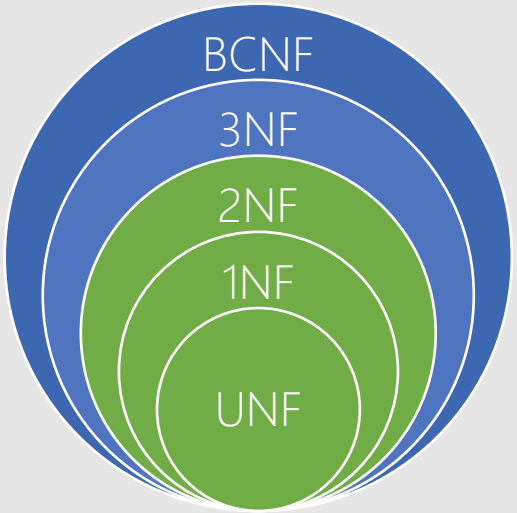
2NF violations:
Title→Language
FirstName→LastName



Normalization × 2NF

ALGORITHM: Normalize a table in 2NF

Move out data into new tables for functional dependencies that violate 2NF



Normalization × 2NF

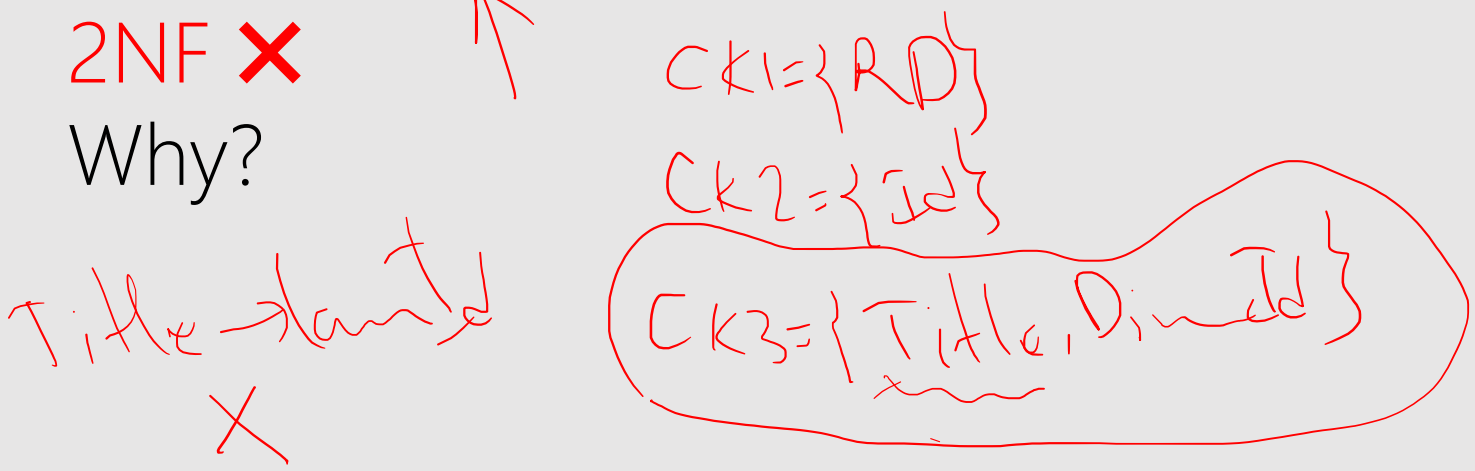
Movie				
Id	Title	LanguageId	ReleaseDate	DirectorId
1	*2001: A Space Odyssey*	*1*	*1968*	*1*
2	*The Shining*	*1*	*1980*	*1*
3	*A Clockwork Orange*	*1*	*1971*	*1*
4	*The Birds*	*1*	*1963*	*2*
5	*Psycho*	*1*	*1960*	*2*
6	*Psycho*	*1*	*1998*	*3*

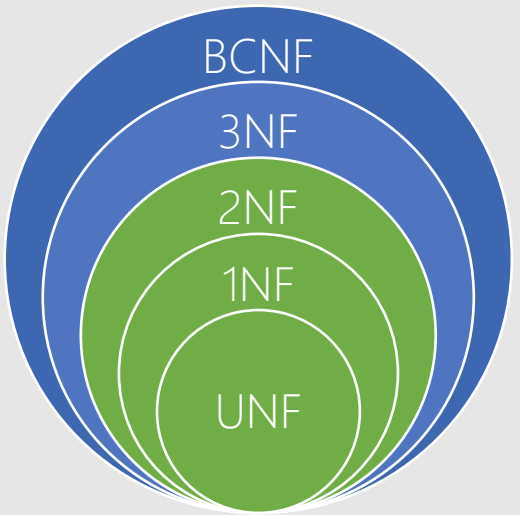
Language	
Id	Title
1	*English*

Director			
Id	FirstName	LastName	PlaceOfBirth
1	*Stanley*	*Kubrick*	*USA*
2	*Alfred*	*Hitchcock*	*England*
3	*Gus*	*Van Sant*	*USA*

2NF ✗

Why?





Normalization × 2NF

Movie				
Id	Title	LanguageId	ReleaseDate	DirectorId
1	*2001: A Space Odyssey*	1	1968	1
2	*The Shining*	1	1980	1
3	*A Clockwork Orange*	1	1971	1
4	*The Birds*	1	1963	2
5	*Psycho*	1	1960	2
6	*Psycho*	1	1998	3

Language	
Id	Title
1	*English*

Director			
Id	FirstName	LastName	PlaceOfBirth
1	*Stanley*	*Kubrick*	*USA*
2	*Alfred*	*Hitchcock*	*England*
3	*Gus*	*Van Sant*	*USA*

2NF ✗

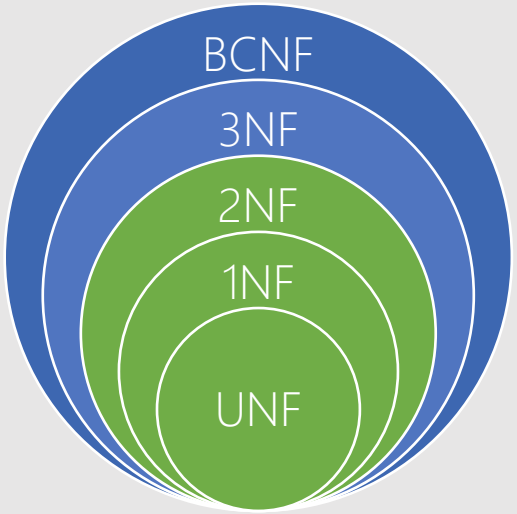
CK1: {Id}

CK2: {ReleaseDate}

CK3: {Title, DirectorId}

Non-Key: {LanguageId}

DirectorId → LanguageId ✗



Normalization × 2NF

Movie			
Id	Title	ReleaseDate	DirectorId
1	*2001: A Space Odyssey*	*1968*	*1*
2	*The Shining*	*1980*	*1*
3	*A Clockwork Orange*	*1971*	*1*
4	*The Birds*	*1963*	*2*
5	*Psycho*	*1960*	*2*
6	*Psycho*	*1998*	*3*

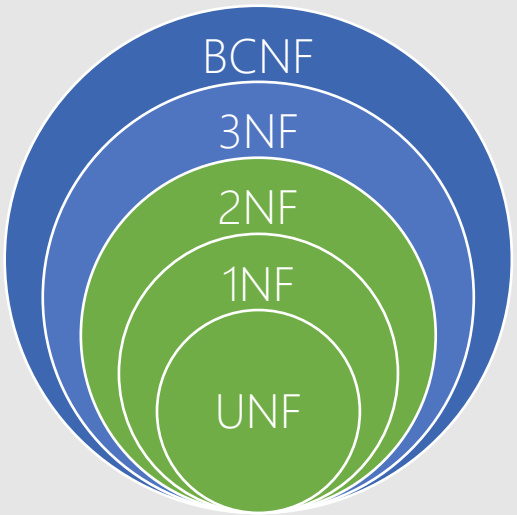
2NF ✔

MovieLanguage		
Id	MovieId	Language
1	1	*English*
2	2	*English*
3	3	*English*
4	4	*English*
5	5	*English*
6	6	*English*

Title→Language

Director			
Id	FirstName	LastName	PlaceOfBirth
1	*Stanley*	*Kubrick*	*USA*
2	*Alfred*	*Hitchcock*	*England*
3	*Gus*	*Van Sant*	*USA*

FirstName→LastName



Normalization × 3NF

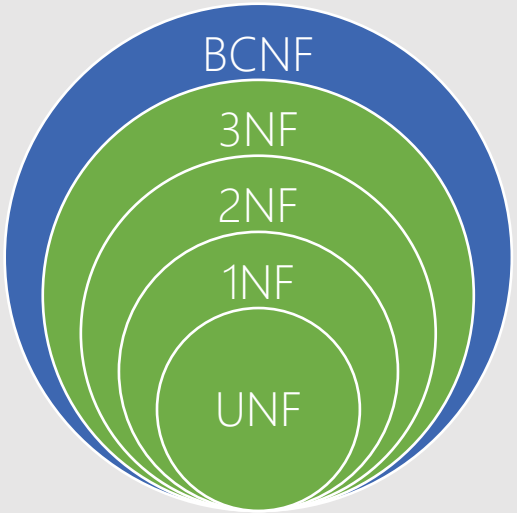
3rd Normal Form requires that a table:

I) Be in 2NF

II) All the attributes in a table are determined only by the candidate keys and not by any non-key attributes.

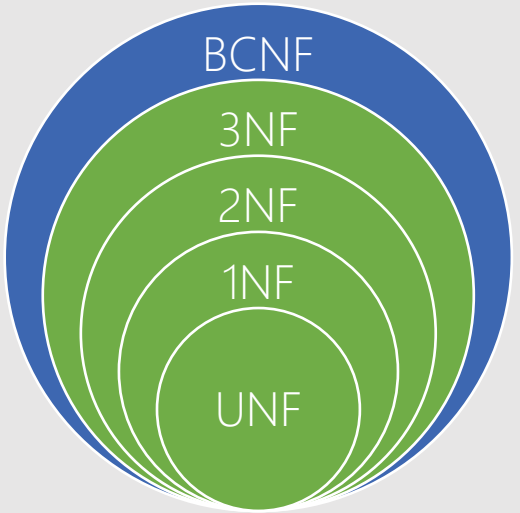
Every non-key attribute of the table is non-transitively dependent only on all candidate keys.

No functional dependencies between non-keys.



Normalization × U,1,2,3NF

U-1NF: Requiring existence of "the key" in the table
2NF: Requiring that non-key attributes be dependent on "the whole key"
3NF: Requiring that non-key attributes be dependent on "nothing but the key"



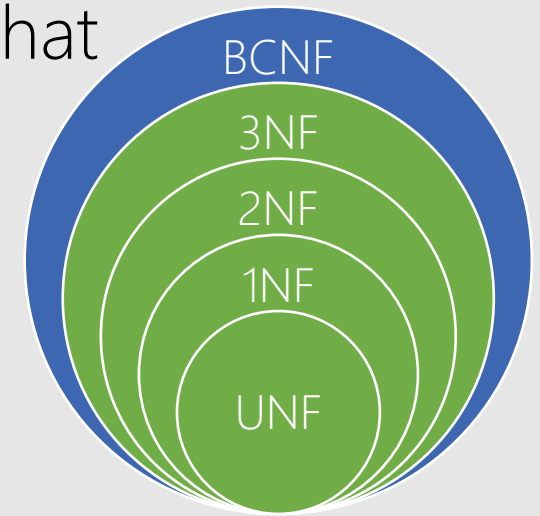
Normalization × 3NF

E.g., T(A, B, C, D, E, F)
CK1 = {A, B}, i.e., AB→CDEF
CK2 = {C, D}, i.e., CD→ABEF

E and F are non-key attributes since E,F∉CK1 and E,F∉CK2

If T is 3NF, there must be NO (E → F) | (F → E) such that

AB → E & E → F then AB→F (transitivity)
CD → E & E → F then CD → F (transitivity)
AB → F & F → E then AB→ E (transitivity)
CD → F & F → E then CD → E (transitivity)

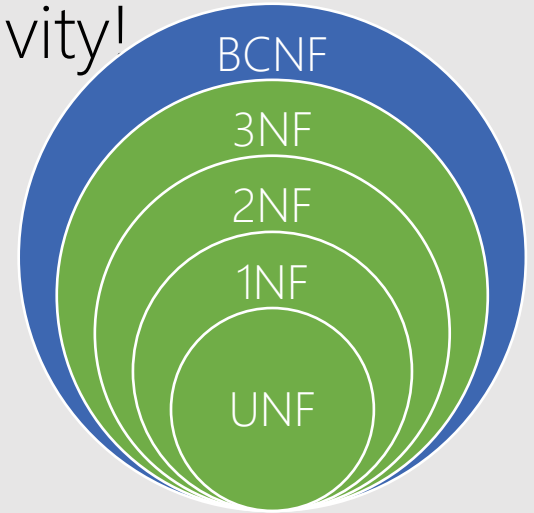


Normalization × 3NF

E.g., T(A, B, C, D, E, F)
CK1 = {A, B}, i.e., AB→CDEF
CK2 = {C, D}, i.e., CD→ABEF

E and F are non-key attributes since E,F∉CK1 and E,F∉CK2

CK1 or CK2 should give E or F directly, not via transitivity!



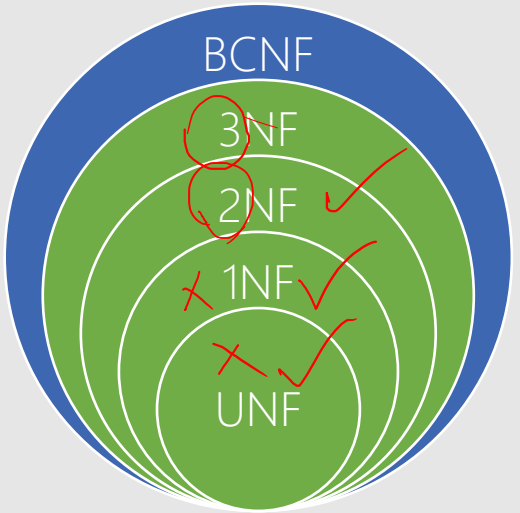
Normalization × 3NF× Example

Invoice(Id, CustomerId, ProductId, Quantity, Price)

PK={Id} i.e., Id → CustomerId, ProductId, Quantity, Price

ProductId → Price

Invoice is NOT in 3NF since …



Normalization × 3NF× Example

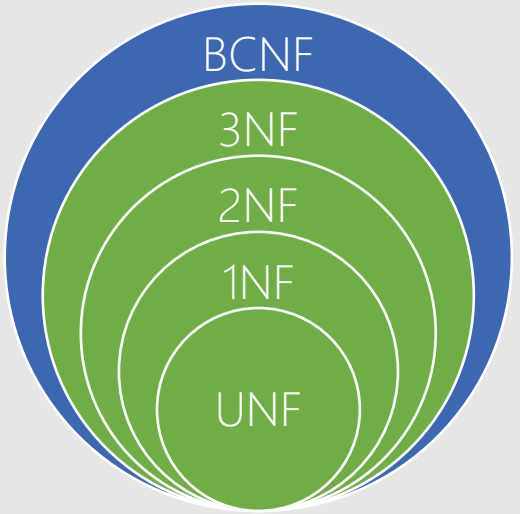
Invoice(Id, CustomerId, ProductId, Quantity, Price)

PK={Id} i.e., OrderId → CustomerId, ProductId, Quantity, Price

ProductId → Price

ProductId as a non-key is a determinant for Price as another non-key

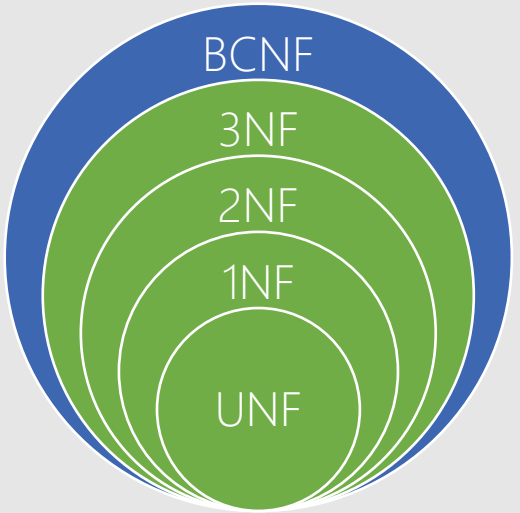
Id → ProductId & ProductId → Price THEN Id → Price (transitive)



Normalization × 3NF

ALGORITHM: Normalize a table in 3NF

Move out data into new tables for functional dependencies that violate 3NF



Normalization × 3NF× Example

Invoice(Id, CustomerId, ProductId, Quantity, Price)

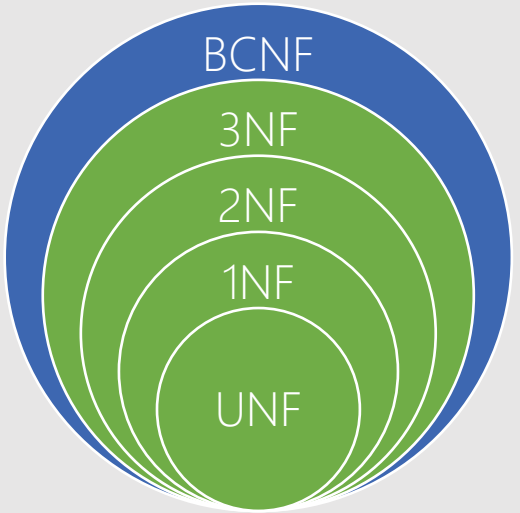
PK={Id} i.e., Id → CustomerId, ProductId, Quantity, Price

ProductId → Price

Invoice(Id, CustomerId, ProductId, Quantity)

ProductPrice(Id, ProductId, Price)

CK1 = {ProductId}



Normalization × U-1,2,3NF

U-1NF: Requiring existence of "the key" in the table
2NF: Requiring that non-key attributes be dependent on "the whole key"
3NF: Requiring that non-key attributes be dependent on "nothing but the key"

- Both 2NF and 3NF are concerned equally with ALL Candidate Keys of a table and not just any one key
- If there is no non-key, i.e., all attributes are part of at least a candidate key, then table is already in 2NF and 3NF.
- In 3NF, a non-key attribute is able to be a determinant of a key attribute!