Start @ 9:10

→ Reducing redundancy, imp for efficient storage, &
preserve data integrity (prevent Anomalies)

ToC

* Anomalies
* Functional Dependencies
* Normalization → 1NF, 2NF ...
* DCC → what is sharding

Anomalies

Eg:

→ Non Null			
Emp Id	Name	Dept	Subjects
→ 123	Devesh	(IT)	DSA
124	Akhil	CS̶ IT	DSA
124	Akhil	CS̶ IT	CN
250	Tanya	Marketing	Psycho
250	Tanya	Market	Sales
null	null	ManPro	null

Insertion Anomaly

new dept → ManPro

→ Data addition is contraint by other columns

Deletion Anomaly

Removal of data causes unintentional loss of data

Update Anomaly

① Data might be lost unintentionally

② Data consistency might be compromised

Functional Dependency

$$A \rightarrow B \quad \checkmark$$

SId	SName	SAge	AG

"If I know A, can I know B?"

$sId \rightarrow sName$? ✓

$sName \rightarrow sAge$? α

$\langle sId, sName \rangle \rightarrow \langle sAge \rangle$ ✓

$sid \rightarrow sid$ ✓

$\langle sid, sname \rangle \rightarrow sname$

$sid \rightarrow sName$ ✓
$sid \rightarrow sAge$ ✓

✓ $sid \rightarrow \langle sName, sAge \rangle$

$S \rightarrow A$, A is a subset of S ✓

Trivial : $A \rightarrow B$, where B is subset of A

Non Trivial : $A \rightarrow B$, where B is not subset of A

Properties

① Reflexive : If y is a subset of X, then X→y

② Transitive : If X⟶y, and y⟶z, then X⟶z

③ Union : If X→y, and X→z, then X⟶yz

④ Decomposition : If X⟶yz, then X→y and X⟶z

⑤ Pseudo Transitivity : If X⟶y

sid → sName

and   yw → z

sName, sAge → sAccess

then   xw → z

sid, sAge → sAccess

Normalization

Unoptimized dB

1NF → 2NF → 3NF → BCNF → ..
                                              4NF
                                              5NF

Anomalies

1 table

Redundancy

↓ Anomalies

↓ Redundancy

↑ tables

Query Eff ↓

1 NF
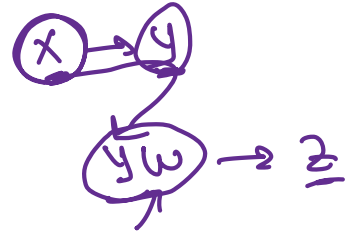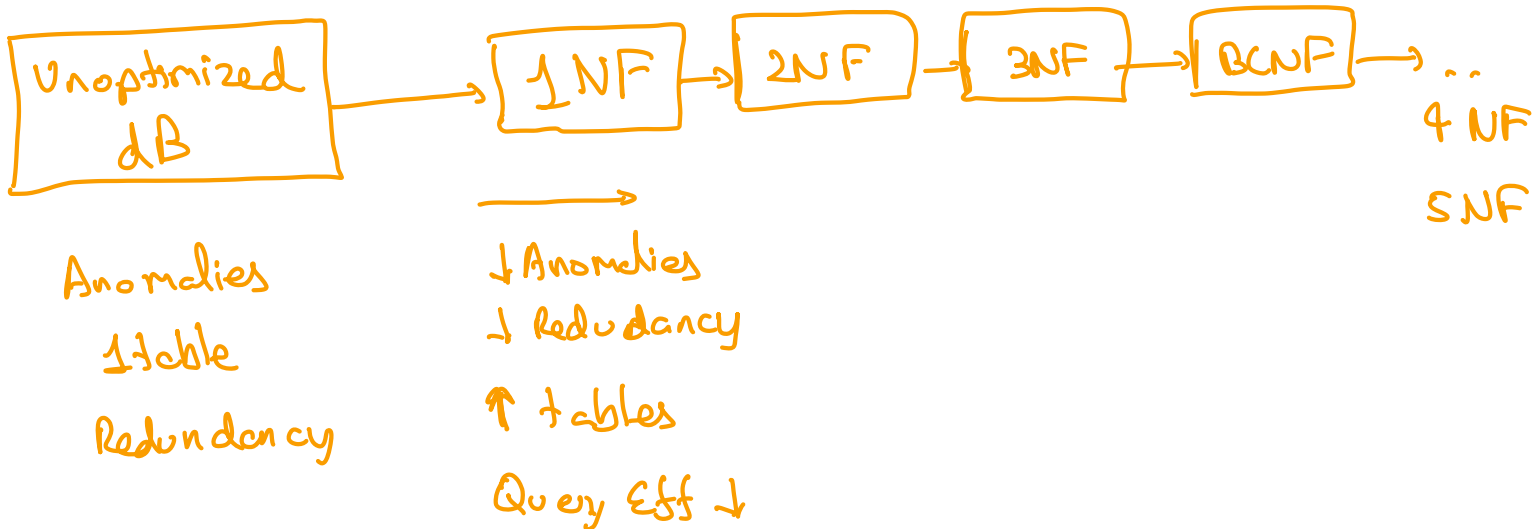
① Any feild should have atomic values. (no multi value attribute)

② Entries in the same column should be of the same type

③ Rows should not repeat multiple times.

Eg.

sid	sName	s Subjects
1	Devesh	DSA|CN|OS
2	Radhit	DBMS|Graphs

→ 1NF ? α

↓

P	NA	P
sid	sName	sSubjects
1	Devesh	DSA
1	"	CN
1	"	OS
2	Radhit	DBMS
2	"	Graphs
3	Devesh	DSA

CK =

α <sName, sSubject> α

SK : ⟨ sid, sName, sSubjects⟩

CK: ⟨sid, subject⟩ ✓

PK: CK

Extra

sid → sName ? ✓

⟨ sid, sSubject⟩ → ⟨sid, sname, subject⟩? ✓

2 NF

Attributes $\longrightarrow$ Prime $\longrightarrow$ Attributes which are included in any CK

$\longrightarrow$ Non Prime $\longrightarrow$ Attributes which are not included in any CK

$\longrightarrow$ 1 NF

$X \longrightarrow Y \longrightarrow$ Y is NPA
$X \subset CK$

$\longrightarrow$ No NPA should be dependent on any proper subset of any CK/SK.

sid	sName	sPin	sCity	Cid	CName	Cfees

CK $\longrightarrow$

sid, cname ✓   CK

sid ✗

\# sid, cid ✓   CK $\longrightarrow$ PK

sid, cid, spin ✗ $\longrightarrow$ SK , but not CK

CK $\longrightarrow$ { sid, cid }, { sid, cname }   PA $\longrightarrow$ sid, cid, cname

NPA $\longrightarrow$ sName, sPin, sCity, Cfees

PS of CK   NPA
sid $\longrightarrow$ sName) ✓

          NPA
sid $\longrightarrow$ sPin ✓

          NPA
sid $\longrightarrow$ sCity ✓

                NPA
sid $\longrightarrow$ < sName, sPin, sCity > ✓

sid	sName	sPin	sCity	Cid	CName	Cfees
1	DS	–	–	101		
1	DS	–	–	102		

sid → sName, sPin, sCity   cid → cname, c fees

CK = cid, cname
NPA = Cfees

3NF

PK

sid	sName	sPin	sCity
1	DS	11.~	Delhi

2NF ✓

→ CK → sid

Proper subsets of CK → φ

PK

cid	cName	Cfees
101	DSA	10
102	CN	20

2NF ✓

→ CK → cid   cid, cname CK

FK   FK

sid	cid
1	101
1	102

PK = < sid, cid >

sid	sName	sPin	sCity	Cid	CName	Cfees
1	DS	1100..	X	101	DSA	10
1	DS	1100..	X	102	CN	C0
2	AB	–	–	201	–	–
2	AB	–	–	202	–	–
2	AB	–	–	203	–	–

sid	sName	sPin	sCity	cid
1	DS	1100..	X	
2	AB	...	Y	

cid	cName	Cfees
101		
101		
102		

sid
1
2
1

↳ sid, cid

1	101
1	102
②	201
2	202
2	203

↑

CK → sid, cid

NPA = cName, cfees

cid → cName ✓

3NF

→ ②NF

→ No NPA should be transitively dependant on a CK/SK

↓

sid	sName	sPin	sCity
1	DS	1100..	X
2	AB	...	Y
3		1100..	X ✓

sid → sPin
sPin → sCity NPA

CK = sid

NPA = sName, sPin, sCity

∨

sid → sName
sid → sPin ✓
sid → sCity
sPin → sCity ✓

Link Table

sid	sName

CK → sid
NPA = sName

sPin	sCity

CK = sPin
NPA = sCity

sid	sPin

CK → <sid, sPin>
NPA = ∅

Valid
Approach

sid	sName	sPin

3NF

↗ CK = sid
NPA = sName, sPin

sPin	sCity
✓

3NF

Valid
Approach

sid → sName

sid → sFin

→ sName → sFin ? ✗

→ sFin → sName ? ✗

BCNF

→ For any A → B , A should be CK/SK

[sub — Prof 1:1]

sid	sub	Prof
1	Java	Mr J
1	DSA	Mr D
2	C++	Mr C
3	Go	Mr G
3	Java	Mr J

3NF → no NAA → NPA ✗

1NF → ✓

2NF → ✓

3NF → ✓

BCNF → ✗

CK = <sid, sub>

<sid, Prof>

NPA = ∅

<sid, sub> → Prof ✓

<sid, Prof> → sub ✓

sub → Prof ✗

Prof → sub ✗

sid,	sub

sub	Prof

BCNF ✓

DCC

no NPA ... transitively ... (CK)

$$X \to A$$
$$\alpha \quad X \to B \quad \alpha$$
$$A \to B$$

→ no NPA → NPA

CK → ~~key~~ X

$$X \to \langle A, B, C \rangle \quad \text{acid, sid} \to$$

NAA → A, B, C

$$X \to A$$
$$A \to B$$

NPA NPA

(A) → (B)

$$X^k \to A$$

A → B

$$A \to B \quad \nearrow$$
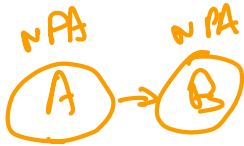
Each table
10 rows
<20 columns
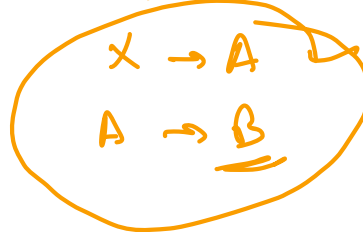
Sharding

Normalization → vertically split
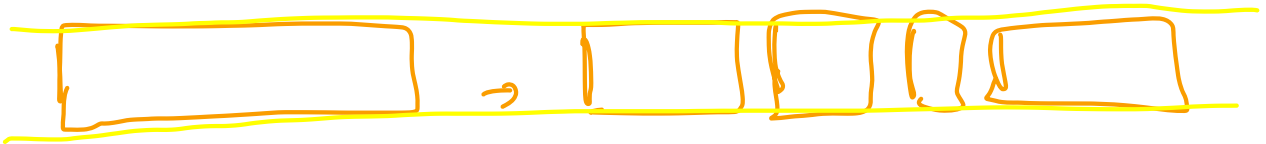
10 rows
20 columns ↝

Horiz Split →
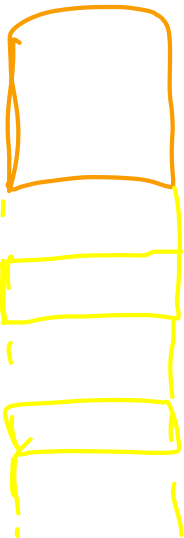
100 rows
10 cols ↝

<100 rows
10 columns