19/3/24

eid	ename	country	state

Key ⇒ eid

1NF ✓
2NF ✓
3NF ✗

eid → ename
eid → state
state → country ← Transitive dependencies.

eid	ename	State

State	Country

3NF ✓

Q ⇒ Pcode Ptitle Pmang budget. eid ename dno dname hourly rate

1NF

PCode	Ptitle	Pmang	Pbudget

Key {PCode}
2NF ✓

Pcode	eid	ename	dno	dname	hourly Rate

key {Pcode, eid}
2NF ✗

Pcode	eid	hourly rate

{Pcode, eid}

eid	ename	dno	dname

{eid} ✓

2NF

T1

P-code	Ptitle	Pmag.	Pbudget

Pcode → Ptitle
Pcode → Pmang
Pcode → budget

3NF ✓

T2

Pcode	eid	hourly rate

Pcode, eid → hourly rate

3NF ✓

T3

eid	ename	dno	dname

eid → ename eid → dname
eid → dno.
dno → dname

3NF ✗

Q ⇒ L.no Lect.name Lect grade dno dname subcode subname sublevel

Assume each lecturer may teach many subjects but may not belongs to more than 1 department.

2NF →

T1

lno.	Lname	lgrade	dno	dname

{Key = lno.}

T2

Scode	sname	slevel

{Key = Scode}

T3

lno	Scode

{Key ⇒ lno, scode}

{ lno → dno, dno → dname } 3NF ✗

T1 →

Lno	Lname	Lgrade	dcode

dcode	dname

BCNF

{Boyce Codd NF}

{A table with 2 attr with be always in BCNF}

⇒ A relation is in BCNF if and only if every non-trivial FD has a candidate key as its determinants.

OR

⇒ A relation is in BCNF if and only if only determinents are Candidate key.

Q)

empid	emp Country	emp dept	d-type	dno.

Key = {empid, empdept}

empid → empcountry

emp dept → d-type, dno.

empid	emp Country

{empid → empcountry}

empid	emp dept

{empid → empdept}

emp-dept	depttype	dno.

{empdept, depttype → dno}

20/03/2024

Decompositions

Lossless decomposition V/s Lossy decomposition.

Emp

eid	ename	did
1	A	101
2	B	101
3	A	102

Decomposition

eid	ename
1	A
2	B
3	A

ename	did
A	101
B	101
A	102

we will do natural join of the decomposed tables and if result is our original table then it will be a lossless decomposition.

First we will take Cartesian Product

eid	ename	ename	dno
1	A	A	101
1	A	B	101
1	A	A	102
2	B	A	101
2	B	B	101
2	B	A	102
3	A	A	101
3	A	B	101
3	A	A	102

eid	ename	did
1	A	101
1	A	102
2	B	101
3	A	101
3	A	102

{Lossy decomposition as we have gotten 2 extra rows, These are called spurious tuples}

R

R1   R2

$\{ R1 \bowtie R2 = R \}$

$\Downarrow$

Lossless else Lossy.

R (eid, ename, dno)

R1   R2

⇒ ① Attributes of $R1 \cup R2$ = Attributes of R

⇒ ② Attributes of $R1 \cap R2 \neq \emptyset$

⇒ ③ Attributes of $R1 \cap R2$ = (Superkey or Candidate key) of (R1 / R2 / R1, R2 Both)

{ In our example $R1 \cap R2$ was ename which was not SK. or CK so we had Lossy decomposition }

Emp

eid	ename	did
1	A	101
2	B	101
3	A	102

R1

eid	ename
1	A
2	B
3	A

R2

eid	did
1	101
2	101
3	102

(R1 × R2)

eid	ename	eid	did
1	A	1	101
1	A	2	101
1	A	3	102
2	B	1	101
2	B	2	101
2	B	3	102
3	A	1	101
3	A	2	101
3	A	3	102

$(R1 \bowtie R2)$

eid	ename	did
1	A	101
2	B	101
3	A	102

$R1 \bowtie R2$ = Emp ⇒ {Lossless decomposition}

① (eid, ename) ∪ (eid, did) ⇒ (eid, ename, did) = Emp

② (eid, ename) ∩ (eid, did) = eid   {≠ φ}

③ $R1 \cap R2$ = eid ⇒ (C.K.)

Example R

(1) ⇒ eid, ename dno dname

R1: eid, ename, dno

R2: dno dname

Lossy or Lossless? ✓

Conditions

(1) R1 ∪ R2 ⇒ {eid, ename, dno, dname}

(2) R1 ∩ R2 ⇒ {dno} {≠ ϕ}

(3) R1 ∩ R2 ⇒ dno → ck of R2

(hence, Lossless)

---

(2) R1 = (tid, subject, age)

(tid, age) (tid, subject)

⇓

Lossless

21/3/24

Dependency Preservation

R (A, B, C, D)

A → B, B → C, C → D, D → B

T1 = (A, B)	T2 = (B, C)	T3 = (B, D)
A → B ✓	B → C ✓	B → D ✓
B → A ×	C → B ✓	D → B ✓
B^+ = {B, C, D}		

{ A → B ✓, B → C ✓, C → B, B → D ✓, D → B ✓ }

(Dependencies Preserved.)

R (A, B, C, D)

~~A → B, C → D~~

T1 (A, B), T2 (C, D)

Lossy decomposition

(A, B)	(C, D)
A → B ✓	C → D ✓
B → A ×	D → C ×

{ A → B, C → B } → Lossy (1), dependency Preserving (2)

Pcode eid ename dno dname hourly rate

Pcode, eid → hourly rate, eid → ename, eid → dno, eid → dname

dno → dname.

R1 { Pcode, eid, hourly rate }

R2 { eid, ename, dno, dname }

R1

Pcode → eid ×

Pcode → h-rate ×

eid, Pcode → h-rate ✓

R2

eid → ename ✓

eid → dno. ✓

eid → dname ✓

eid, ename → d-no ×

eid, ename → dname ×

dno → dname ✓

eid, Pcode → h-rate ✓
eid → ename ✓
eid → dno ✓
eid → dname ✓
dno → dname ✓

{ Dependency Preserving }

Q ⇒ R = (eid ename epin estate)

{ eid → ename , eid → epin , epin → estate }

R1 = { eid, ename }

eid → ename ✓

ename → eid ×

R2 { epin , estate }

epin → estate ✓

estate → epin ×

eid → ename
epin → estate

① Dependency preserving ×

② Lossy decomposition

Q → R (A, B, C)

A → B, B → C

R1 (A, B) R2 (A, C)

A → B ✓ A → C ×

B → A × (A → A ×

A → BC

A → B

A → C

{ A → B }

{ Lossless ✓
DP × }

Q ⇒ R (x, y, z, w, P, Q)

xy → w, xw → P, PQ → z , xy → Q

R1

(z P Q)

PQ → z ✓

~~PQ → z~~

~~z → PQ~~

z → PQ ×

R2

(xy w P Q)

x → y ×

x → w ×

x → P ×

x → Q ×

xy → w ✓

xy → P ✓

xy → Q ✓

xw → P ✓

xw → Q ×

{ Lossless,
DP ✓ }

22/3/2024

Minimal Cover / Canonical Cover

→ Irreducable set of F.D.

Conversion

① Make RHS of each FD as atomic.

② Remove ~~redundant~~ FDs {where LHS is atomic as well}

③ Remove extraneous attributes from Left hand side.

Example →

$A \to B$, $A \to C$, $B \to C$, $AB \to B$, $AB \to C$

$AC \to B$

find minimal set

Solution ⇒

Step 1 → All RHS are atomic, so no change.

Step 2 ⇒

$A \to B$, $B \to C$, $A \to C$, $AB \to B$ ✗, $AB \to C$,

$AC \to B$

if we remove $A \to B$ then can we still access same attributes

LHS atomic values

Let's first consider $A \to B$, $B \to C$, $A \to C$

we can remove $A \to C$ as ~~this~~ this can be achieved by ($A \to B$ & $B \to C$)

$A \to B$, $B \to C$, $AB \to B$, $AB \to C$, $AC \to B$

$AB \to B$ ↳ Trivial dependency

LHS non-atomic

③

$A \to B$, $B \to C$, $AB \to C$, $AC \to B$

$AB \to C$ ↓ $B \to C$

$AC \to B$ ↓ $A \to B$

⇓

Minimal Set

$\{A \to B, B \to C\}$