# Database

Friday, January 3, 2025 9:21 AM

What is normalization?
It is a technique to reduce redundancy from the table.

| Normalization Form | Definition | Key Requirements | Purpose | Example |
|---|---|---|---|---|
| **1NF** | A relation is in 1NF if it has only atomic (indivisible) values and each column contains unique values for that attribute. | - Each column has atomic values.- Rows are uniquely identifiable (with a primary key). | Eliminates duplicate rows and ensures data is organized in tabular format. | A table storing customer data where each cell contains a single value (e.g., one phone number per customer). |
| **2NF** | A relation is in 2NF if it is in 1NF and all non-prime attributes are fully functionally dependent on the entire primary key. | - Must be in 1NF.- Non-prime attributes must depend on the entire primary key (eliminates partial dependency). | Removes partial dependency. | Splitting a table into two to ensure that attributes related to only part of a composite key are stored in a separate table. |
| **3NF** | A relation is in 3NF if it is in 2NF and there are no transitive dependencies (non-prime attributes depend only on the primary key). | - Must be in 2NF.- No transitive dependency (no non-prime attribute depends on another non-prime attribute). | Removes transitive dependency. | Creating a separate table for department names if they indirectly determine an employee's address via a department ID. |
| **BCNF** (Boyce-Codd) | A stricter version of 3NF where every determinant is a candidate key. | - Must be in 3NF.- Every determinant must be a candidate key (eliminates dependency anomalies). | Ensures minimal redundancy and dependency. | A table that contains attributes for professors and courses split into two to prevent a non-candidate key determining another attribute. |
| **4NF** | A relation is in 4NF if it is in BCNF and has no multi-valued dependencies (MVDs). | - Must be in BCNF.- No multi-valued dependencies (MVDs). | Removes multi-valued dependency. | Splitting a table with attributes for students and hobbies into separate tables, one for hobbies and another for skills, to eliminate MVDs. |
| **5NF** | A relation is in 5NF if it is in 4NF and cannot be further decomposed into smaller tables without losing data (removes join dependency). | - Must be in 4NF.- No join dependency (JD). | Prevents loss of data during recombination. | Decomposing a table to separate supplier, part, and project relationships into individual tables to preserve information during recombination operations. |

Two types of redundency:

1) row level
2) column level

* row level redundency can resolve using primary key

* Column level
  - → update anomaly (Problem)
  - → insert anomaly
  - → deletion anomaly

Column level redundency remove using normaization

Normaiization:

1] 1NF 2] 2NF 3] 3NF 4] BCNF

① 1 NF: Table should not contain any multivalued attribute.

eg

| Rollno | name | Course |
|---|---|---|
| 1 | A | C/C++ |
| 2 | B | C |

Soln 1

| Rollno | name | Course |
|---|---|---|
| 1 | A | C |
| 1 | A | C++ |
| 2 | B | C |

Compo site

Primary key = Rollno + Course

Page No.
Roll No.

Date :

Soln 2

| Rollno | name | course1 | Course 2 | Problem |
|---|---|---|---|---|
| 1 | A | C | C++ | ↓ many null value |
| 2 | B | C | — | |

Primary key = Rollno

Soln 3

t1

| Rollno | name |
|---|---|
| 1 | A |
| 2 | B |

PK

Pk = Rollno

Fkey

t2

| Rollno | Course |
|---|---|
| 1 | C |
| 1 | C++ |
| 2 | C |

Pk = Rollno + course

* Closure : Find all candidate key

Relation attribute

R (A, B, C, D)

FD {A → B, B → C, C → D}

$A^+$ → B, C, D, A  Candidate key ✓

$B^+$ → B, C, D ×

$C^+$ → C, D ×, $D^+$ → D ×

~~AB+ also candidate key but Cant~~

AB+ Can be a Closure but Candidate key always minimal

$C_k$ = {A}

where : R → Relation

(A, B, C, D) → Attribute

FD → Functional dependency

A+ → Closure

Ck → Candidate key

Date :

Roll No.

(2) R (A, B, C, D)

FD → {A → B, B → C, C → D, D → A}

$A^+$ = A, B, C, D.
$B^+$ = B, C, D, A
$C^+$ = C, D, A, B
$D^+$ = D, A, B, C

all C.key

Prime attribute : A, B, C, D
non-Prime attribute : null

(3) R(A, B, C, D, E)

FD = {A → B, BC → D, E → C, D → A}

$A^+$ = {A, B}
$B^+$ = {B}
$C^+$ = {C}
$D^+$ = {D, A, B}
$E^+$ = {E, C}

no do like this

① First Check which attribute no on right side of FD.

e.g R(A, B, C, D, E)

FD ⇒ {A → B, BC → D, E → C, D → A}

$E^+$ = B D C A

E ~~only~~ Prime attribute

② check, using Combining more attribute

for e.g

$AE^+$ = {A, E, B, C, D} Ckey

$BE^+$ = {B, E, ...}

$CE^+$ = {C, E, ...}

Ck = {AE, DE, ~~BCE~~, BE, CE} ckey key

Prime = {A, B, C, D, E}

non-prime = {—}

Date :

Reflexivity : if y is subset of x then $x \to y$

Argumentaiton : if $x \to y$, they $xz \to yz$

Transitive : if $x \to y$ & $y \to z$ then $x \to z$

union : if $x \to y$ & $x \to z$ then $x \to yz$

decomposition : if $x \to yz$ then $x \to y$ & $x \to z$

Composition : if $x \to y$ & $z \to w$ then $xz \to yw$

2NF :

① relation or table must be in 1st normal form

② All non-Prime attribute should be fully functional dependent on Candidate key

or there should partial dependency in rel

no

# e.g C.key AB Proper subset → A, B

But A or B single determine non Prime attribute then their is partial dependency

A → C or B → D

ate :

R (A B C D E F)

FD = { C → F , E → A, EC → D, A → B }

C.k: EC = F A D B

Ect = E, C, A, F, D, B

① Ck : {EC}

② Prime attribute : E, C

③ non-Prime attribue : A, B, D, F

④ Proper subset of EC : E, C

FD = { C → F , E → A, EC → D, A → B }

PD PD

3NF: ① relation or table must be 1NF & 2NF

② there should be no transitive dependency in table.

R (A B C D)

FD = {AB → CD, D → A}

LHS → RHS (OR)

C.k or S.k — P.Attr valid

C.key : $AB^+$, $DB^+$

Prime Attr : A, B, D

non-prime Attr : C

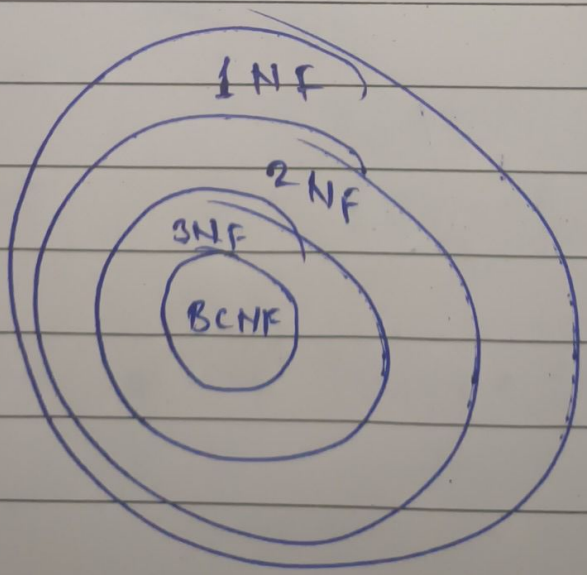
FD : {AB → CD, D → A}

✓ ✓

Relation Already in 3NF

BCNF: ① relation in 3NF

② X should be a super key for every FD (FD) X → Y in a given relation.



3NF always ensures dependency preserving decomposition but not in BCNF.

Both third & BCNF ensure lossless decomposition.

e.g. R(ABCD)

FD = {AB → CD, D → A}

AB+ = {A, B, C, D} Ckey

CB+ = {C, A} ×

DB+ = {D, B, A, C} Ckey

Ckey = {AB, DB}

Prime att$^r$ = {A, B, D}

non-Prime = {C}

FD = {AB → CD, D → A}

✓ ✓ 3NF

FD = {AB → CD, D → A}

✓ × Not in BCNF

R(ABCD)

$R_1$(D A) {D → A}

$R_2$(A B C D) {BD → C}

B+ = B BD+ = B, D, A, C

C+ = C

D+ = D

∴ BCNF not always Preserve dependency.

ate :

R (A B C D E F)

FD = { AB → C, C → D, C → E, E → F, F → A }

CK = { AB, FB, EB, CB }

PA = { A, B, C, F, E }

NPA = { D }

① Already in 1 NF

② FD = { AB → C, C → D, C → E, E → F, F → A }

F ↛ F  T ↛ T  T ↛ F  T ↛ F  T ↛ F

'FD'  (PD)  'FD'  'FD'  'FD'

not in 2NF

R(A B C D E F)

$R_1$(A B E F C)

$R_2$(C D)  C ↛ C, D

$R_2$ { C → D }

C.k = C

$R_1$ { AB → C, E → F, C → E, F → A }

Ck : { AB, EB, FB, CB }

③ Already in 3NF

④ BCNF:

$R_1$ { AB → C, E → F, C → E, F → A }

∝  ∝  ∝

$R_2$ { C → D }  BCNF  O/. Redundency

$R_1$(A B C E F)

$R_2$ (C → E)  (C E)  {C}  C.key

$R_3$ (E → F)  (E F)  {E}

$R_4$ (F → A)  (F A)  {F}

$R_5$ AB → C)  (A B C)  {AB}

O/. Redundency