DATE

Normalization

Reason : Why do we need normalization?

Stud_info table :-

(PK) (id)	sid	name	age	branch_code	Branch_name	HOD_name
	1	A	18	101	IT	XYZ
	2	B	19	101	IT	XYZ
	3	C	18	101	IT	XYZ
	4	D	21	102	EC	PQR
	5	E	20	102	EC	PQR
	6	F	19	103	MECH	ABC

Redundancy of data.

Idea :- In the table student info we have tried to store entire data about student.

Result :- Entire branch data of a branch must be repeated for every student of the branch.

Redundancy :- When the same data is stored multiple times, unnecessarily in the database.

Disadvantage :- ① Insertion, deletion and modification anomalies (problems).
② Inconsistency (data)
③ Increase in database size and increase in time (to search the required data).

Insertion anomalies :- When certain data (attributes) cannot be inserted into db, without the presence of other data.
Ex :- If we want to insert civil department details

PAGE

classmate

DATE

and still none of the students have joined, then we will not be able to insert the department details itself.

Deletion anamoly :- if we want to delete the unwanted data, and if that deletion causes deletion of necessary data.

Eg :- Student id 6 left the college and was the only student who joined mech. By deleting student id 6, we also loose mech dept data which is necessary.

Update / modification anamoly :- When we want to update a single piece of data, but it must be done at all the places where it is repeated.

Eg :- changing the FC head name from 'PSR' to 'MNO', it should be done at all places, if missed even at single occurance leads to data inconsistency.

How do we normalize

a) 1 paragraph contains 1 idea, Similarly 1 table should contain info about 1 entity.

b) **Normalization** (decomposition of table) of table is done on the basis of Functional dependencies.

Stud

Sid	
1	
2	
3	
4	
5	
6	

1
2
3
4

DATE

Stud-info

Sid	Sname	age	Brcode	Brname	Brod.name
1	A	19	101	IT	XYZ
2	B	18	101	IT	XYZ
3	C	18	101	IT	XYZ
4	D	20	102	EC	PQR
5	E	19	102	EC	PQR
6	F	20	103	MECH	KLM

Stud-info

Sid	Sname	age	Brcode
1	A	18	101
2	B	19	101
3	C	18	101
4	D	21	102
5	E	20	102
6	F	19	103

Branch-info

Brcode	Brname	Brod.name
101	IT	XYZ
102	EC	PQR
103	MECH	KLM

Notations used in normalization.

1. Prime attributes and non-prime attributes
2. Partial functional dipendency
3. Fully Functional dipendency
4. Transitive functional dipendency.

Consider the given FD:-

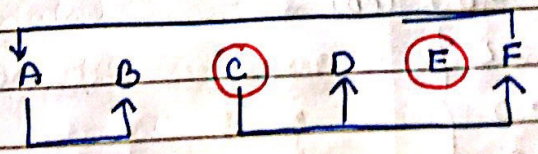
R(A, B, C, D, E, F)

$C \rightarrow F$

$F \rightarrow A$

$EC \rightarrow D$

$A \rightarrow B$



$(EC)^+ = \{E, C, D, F, A, B\}$ – CK.

PAGE

classmate

DATE

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

E.C	A, B, D, F
candidate key & Prime attribute	non-prime attribute

② Fully Functional dependency
Partial Functional dependency
Transitive Functional dependency.

② Fully Functional dependency :

X → y , 'y' is said to be fully FD on x if it cannot be determined by any subset of X.

ABC → D

③ Partial Functional dependency :-

X → y, 'y' is said to be partially FD on x if its subset can determine the value of Y.

ABC → D
AB → D } (PD)
AC → D } (PD) (proper subset of ABC)
BC → D } (PD)
A → D
B → D
C → D

DATE

Eg:- $AC \rightarrow P$ 'P' is not Fully FD on AC
$A \rightarrow D$ 'P' is Partially FD on AC.
$D \rightarrow P$

$(AC)^+ = (A, C, P, D)$
↳ subset.
$(A)^+ = \{A, D, P\}$ ✓ $A \rightarrow P$ (PD)
$(C)^+ = \{C\}$

④ Transitive Functional dependency :-

if x, y, z some set of attributes

$x \rightarrow y$
$y \rightarrow z$
$x \rightarrow z$ (transitive dependency)

Definition of Normalization and Normal Forms.

Definition :- normalization of data can be looked upon as a process of analysing the given relation schema based on their FD and Primary key, to achieve the desired properties of minimizing redundancy and minimizing the insertion, deletion and update anomalies.

Several normal forms have been proposed, each normal form minimizes the redundancy upto some extent.

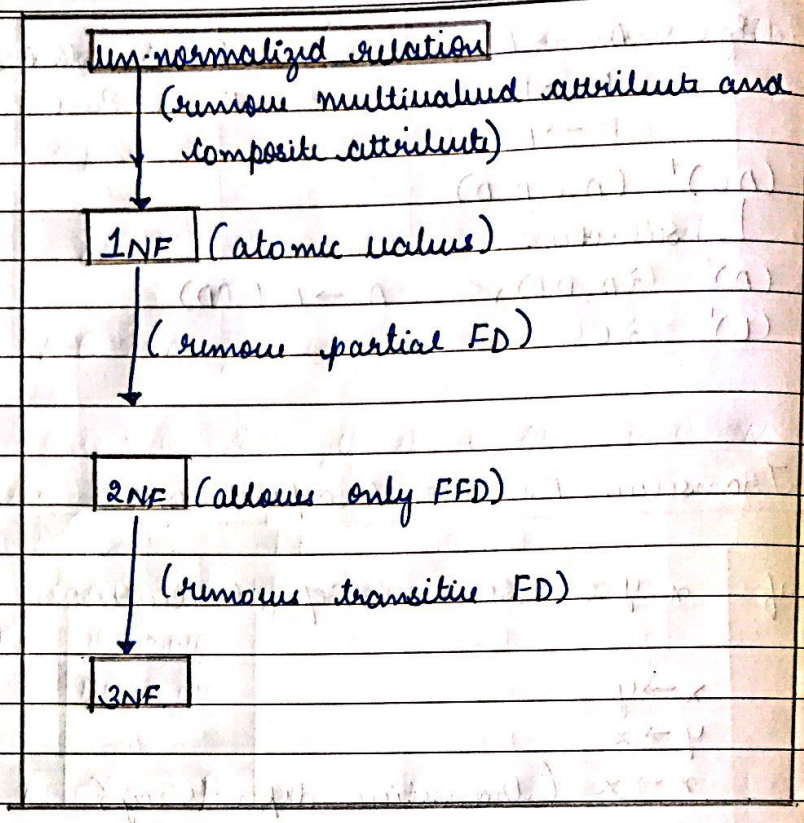
The higher normal forms are only of theoritical interest but not practically applicable.

Most DB Systems use normalization upto 3NF. (upto BCNF) is recommended.

PAGE

classmate

DATE



Normal forms

First normal form :- does not allow multi valued, complex and composite attribute.

※ Attribute should be atomic in nature.

Roll no	name	course opted
1	A	CO/CN
2	B	DBMS/OS

Roll no	name	course
1	A	CO
1	A	CN
2	B	DBMS
2	B	OS

PAGE

Second n

AB –
B –
$(AB)^+$ =

AB –
Prime a
non - pr

When
part of

$(AB)^+$ = {
$(B)^+$ = {

A B

total

decomp

class

DATE

Second normal form :- (Partial FD removal)

$AB \rightarrow D$

$B \rightarrow C$

$(AB)^+ = \{A, B, C, D\}$

$AB \rightarrow$ candidate key.

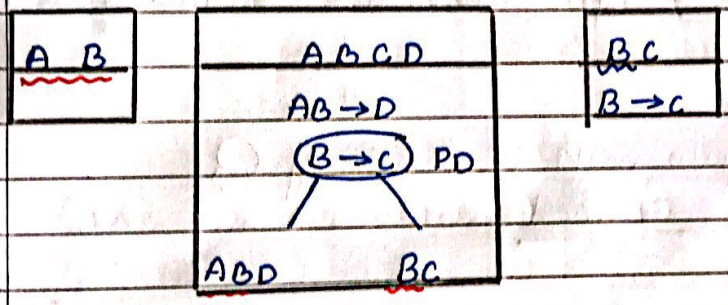
Prime attribute :- AB.

non-prime attribute :- CD.

When a non-prime attribute depends only on part of CK., then it is PFD.

$(AB)^+ = \{A, B, D, C\}$

$(B)^+ = \{B, C\}$.



total no of tables AB, ABD, BC (3 tables).

decomposition is lossless and dependency preserving.

AB ∩ ABD

AB → PK of 1st table & 2nd.

ABD ∩ BC

'B' → Key for 2nd table.

PAGE

classmate

DATE

Third normal form :- (eliminate transitive FD)

Transitive dependency :- A FD from

$X \rightarrow Y$ is called transitive if

$X, Y \in$ non-prime.

① A relation is in 3NF if it in 2NF and no transitive dependency.

Note :- to easily identify 3NF

PD :- $P \rightarrow NP$

$NP \rightarrow NP$

$X \rightarrow B$

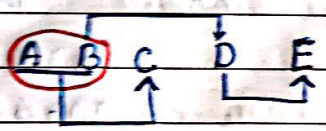
Example :-

① $R(A, B, C, D, E)$

$AB \rightarrow C$

$B \rightarrow D$

$D \rightarrow E$



1st step :-

$(AB)^+ = (A, B, C, D, E)$ ✓ candidate key – 1st table.

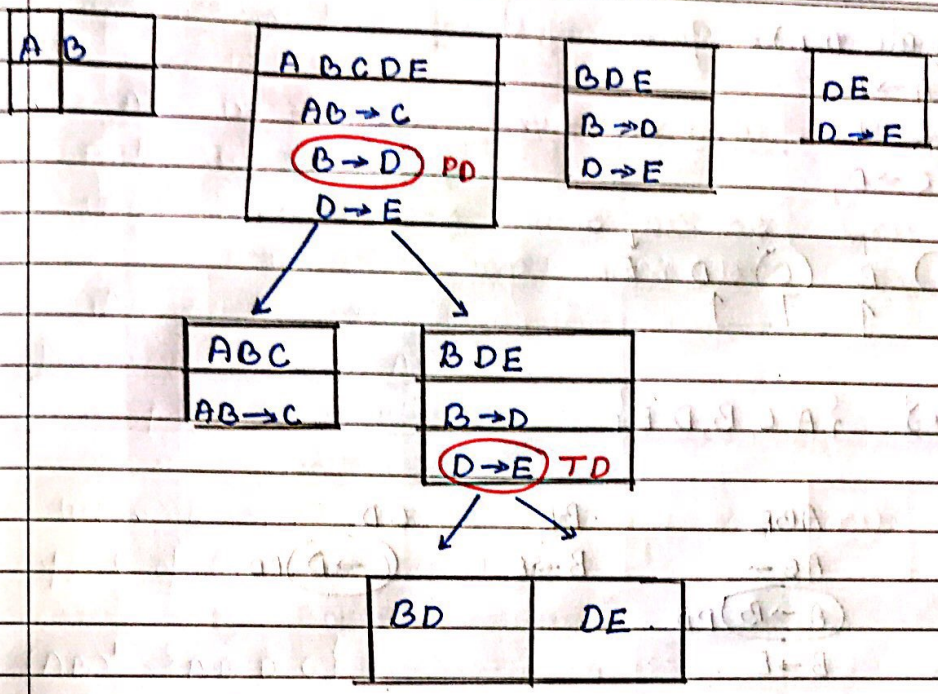
2nd step :-

$(AB)^+ = \{A, B, C, D, E\}$

$(B)^+ = \{B, D, E\}$

$(D)^+ = \{D, E\}$.

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



no of tables : AB ABC BD DE

given tables are in 3NF

② R(A,B,C,D,E)
A → B
B → E
C → D

③ R(A,B,C,D,E,F,G,H,I,J)
AB → C
A → DE
B → F
F → GH
D → IJ

④ R(A,B,C,D,E,F,G,H,I,J)
AB → C
A → DE
B → F
F → GH
D → IJ

PAGE

classmate

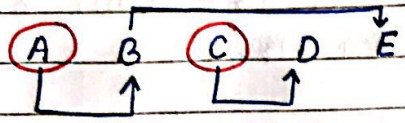
DATE

② R(A, B, C, D, E)

$A \to B$

$B \to E$

$C \to D$.



$(AC)^+ = \{A, C, B, D, E\}$.

AC	ABE	BE	CD
	~~AB~~	$B \to E$	$C \to D$ PD.
	$A \to B$ PD		
	$B \to E$		

AC	ABE	CD	BE (2nd normal form)
	$A \to B$		
	$B \to E$ TD		

AC AB BE CD. (4 tables are in 3NF)

AC ∩ AB

ABC ∩ BE

ABCE ∩ CD

ABCDE. lossless and dependency preserving.

③ R(A, B, C, D, E, F, G, H, I, J)

$AB \to C$

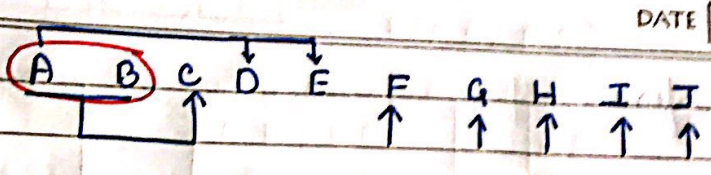
$A \to DE$

$B \to F$

$F \to GH$

$D \to IJ$

DATE



$(AB)^+ = \{A, B, C, D, E, F, G, H, I, J\}$

AB ABCDEFGHIJ BFGH FGH DIJ

$AB \rightarrow C$

$A \rightarrow DE$ PD

$B \rightarrow F$ PD

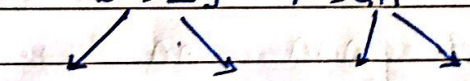
$F \rightarrow GH$

$D \rightarrow IJ$

AB ABC ADEIJ BFGH FGH DIJ (2nd NF)

AB ABC $A \rightarrow DE$ $B \rightarrow F$

$D \rightarrow IJ$ $F \rightarrow GH$



ADE DIJ BF FGH

AB ABC ADE DIJ BF FGH (3rd NF)

④ Table form questions.

Student course details :-

Sid	cid	sname	sdob	cname	cost	marks	grade

First normal form

Sid	cid	SFN	SMN	SLN	SDob	Smon	Syear	cname	cost	marks
								grade		

DATE

Second normal form.

Find the candidate key.

$(sid, cid)^+ = \{sid, cid, sname, sdob, cname, cost, marks, grade\}$

table 1

sid	cid

table 2

sid	sname (FN/MN/LN)	sdob (DD/MM)

table 3

cid	cname	cost

table 4

sid	cid	marks	grade

DATE

Third normal form

table 1

sid	cid

table 2

sid	sname	sdob

table 3

cid	cname	cost

table 4

sid	cid	marks

table 5

marks	grade

↓

BCNF :- in the above 3NF tables sid cid, sid, cid, marks are the determinants of students, course, marks, grade. These determinants are keys as well. Since determinants are the keys the given table is in BCNF.

PAGE

classmate

(10m)

DATE

Detailed explanation of the problem:

① Determinant :- a determinant is that attribute of an entity which determines the value of other attributes.

Eg:- marks determines grade and hence the marks attribute is the determinant, represented as :-

marks → grade.

※ all determinants are not keys, however all keys are determinants.

Eg:- marks is a determinant but it is not a key because marks cannot determine the name, dob of a student.

On the contrary 'sid' is a determinant which can determine all the attributes and hence is called the key.

∴ key is a determinant, that determine all attributes.

② Partial functional dependancy :- when the key attribute of an entity does not completely determine an attribute then it is said that, the attribute is partially functionally dependant on the key attribute, in other words if the key attribute is a candidate key and if an attribute is not dependant on all the components of the candidate key then it is said

PAGE

classmate

to be
key attr

Eg:- in
is
only on
(so just

② Total
attribut
the at
such i
key at
candid
on
then it
functio

Eg:-

④ Tra
attri
attrib
then
trans
attrib

FD
cid, si
na

cla

DATE

to be partially functionally dependent on the key attribute.

Eg:- in the above identified candidate keys is (sid cid) an attribute snam depends only on sid and coursnam depends on cid). (so just part of the key).

③ Total functional dependency :- when the key attribute of an entity completely determines the attributes then it is said that the attribute is totally functionally dependent on the key attribute, in other words if the key is a candidate key and if an attribute depends on all the components of the candidate key then it is said that the attribute is totally functionally dependant on the CK.

Eg:- (sid, cid) → marks.

④ Transitive functional dependency :- If an attribute of an entity depends on the key attribute indirectly through another attribute then it is said that the attribute is transitively functionally dependant on the key attribute.

(cid, sid) → marks.

marks → grade.

FD that can be derived :-

cid, sid → marks sid → snam, dob.

marks → grade cid → cnam, pos, cnam

PAGE

classmate

DATE

② Convert the table upto BCNF.

Projno	Projname	Empno	Empname	Rate category	hourly rate
1023	networking	101	Vincent	A	60
		102	Pauline	B	50
		103	Charles	C	40
1050	DB	101	Vincent	A	60
		107	David	B	50

⇓

First normal form

Proj no	Projname	Empno	Empname	Rate category	hourly rate
1023	networking	101	Vincent	A	60
1023	networking	102	Pauline	B	50
1023	networking	103	Charles	C	40
1050	DB	101	Vincent	A	60
1056	DB	107	David	B	50

⇓

Second normal form

Partial dependency.

tab 1 :- Projno | Emp no.

tab 2 :- Empno | Empname | Rate-category. | hourly rate.

DATE

table 3 : | projno | projname |

⇓

Third normal form :-

(-transitive dependency)

table 1 : | projid | empno |

table 2 :- | empno | empname |

table 3 : | projno | projname |

table 4 : | rate category | hourly rate |

⇓

all determinant are super keys so in BCNF.

FD :-
projid → projname.
empid → empno.
rate category → hourly rate

③ Consider the following relation

car_sale (car#, date_sold, salesman#, comission%, discount_amt)

assume that a car may be sold by multiple salesperson and hence (car#, salesman#) is the PK

additional dependencies are :-

date_sold → discount_amt

salesman# → comission.

PAGE

classmate

DATE

Based on the given primary key is this relation in 1NF, 2NF or 3NF. Why or why not how would you successfully normalize it completely

car#	date_sold	salesman#	commision %	discount amt

(composite attribute)
so split into day, month and year

Second normal form (Partial dependencies)

tab1 :-

salesman#	car#

tab2 :-

car#	date_sold	month	year	discount_amt

tab3 :-

salesman#	commission%

3rd NF.

tab1 :-

salesman#	car#

tab2 :-

car#	date_sold	monthsold	year_sold

tab3 :-

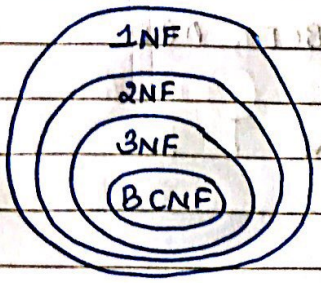
salesman#	comision #

tab4 :-

date_sold	discount amt

DATE

How to identify normal form.



① BCNF is checked first (shortcut)

$X \longrightarrow Y$

↓

Should be a superkey.

② Third NF

$X \longrightarrow Y$

↓

Should be a superkey or 'Y' should be prime.

Problems :-

① R(A,B,C,D,E,F,G,H)

$AB \rightarrow C$ candidate key :- AB

$A \rightarrow DE$ (PD)

$B \rightarrow F$ (PD)

$F \rightarrow GH$ (TD)

FD is in 1st NF.

② R(A,B,C,D,E)

$CE \rightarrow D$ candidate key :- CE

$D \rightarrow B$ (TD)

$C \rightarrow A$ (PD)

FD is in 1st NF.

PAGE

DATE

③ R(A,B,C,D,E,F)

AB → C (PD) CK:- BCD, ABD
DC → AE (PD)
E → F ✓

1st NF.

④ R(A,B,C,D,E,G,H,I)

AB → C CK:- ABD
BD → EF
AD → GH
A → I

1st NF.

⑤ R(ABCDE)

AB → CD ✓ CK:- AB, BD, BC
D → A ✓ (Prime attribute)
BC → DE ✓

3rd ~~1st~~ NF

⑥ R(A,B,C,D,E)

BC → ADE BC, CD
D → B (prime attribute)

~~1st NF.~~ 3rd NF.

classmate PAGE

DATE

⑦ R(V, W, X, Y, Z)

X → YV (PD) CK: VW, XW

Y → Z

Z → Y

VW → X ✓ (BCNF)

X → Y [2nd NF]

↓ ↓

Part of CK non-prime

1st NF.

⑧ R(A, B, C, D, E, F)

ABC → D (BCNF) CK: ABC, ACD.

ABD → E. BCNF ✗

CD → F (PD) 3NF ✗

CDF → B

BF → D

1NF

⑨ R(A, B, C)

A → B

B → C A, B, C BCNF ✓

C → A

DATE

(10) R(A,B,C,D,E,F)

$A \rightarrow BCDEF$

$BC \rightarrow ADEF$

$DEF \rightarrow ABC$

CK:- A, BC, DEF

BCNF ✓

(11) R(A,B,C)

$AB \rightarrow C$

$C \rightarrow A$

CK:- AB, BA

3rd NF

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

$A \rightarrow B$

$BC \rightarrow E$

$DE \rightarrow A$

CK:- ACD, BCD, CDE

3rd NF

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

$AB \rightarrow CD$

$D \rightarrow A$ (partial)

$BC \rightarrow DE$

CK:- AB, BC, BD.

3NF

(14) R(W,X,Y,Z)

$Z \rightarrow W$

$Y \rightarrow XZ$

$XW \rightarrow Y$

CK:- Y, XW, XZ

3rd NF.

PAGE

DATE

(15) R(A, B, C, D, E, F)

AB → C
DC → AE
E → F

CK:- ABD, BCD.

1NF

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

A → B
B → E
C → D

CK:- AE.

1NF

(17) R(V, W, X, Y, Z)

Z → Y
Y → Z
X → YV
VW → Y

CK:- XW

1NF

(18) R(A, B, C, D, E, F)

ABC → D
ABD → E
CD → F (PT)
CDF → B
BF → D

CK:- ABC, ACD

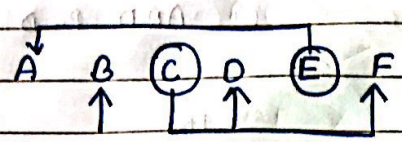
1NF.

PAGE

Gate Questions DATE

(19) R(A, B, C, D, E, F)

$C \to F$
$E \to A$
$EC \to D$
$A \to B$.

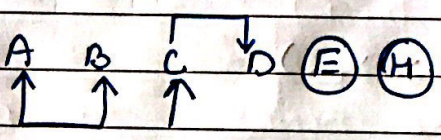


$(CE)^+ = \{C, E, F, A, B, D\}$

1NF.

(20) R(A, B, C, D, E, H)

$A \to B$
$BC \to D$
$E \to C$ (PD)
$D \to A$.



$(EH)^+ = \{E, H, C\}$ X

$(AEH)^+ = \{A, E, H, B, C, D\}$ ✓
$(BEH)^+ = \{B, E, H, C, D, A\}$ ✓
$(CEH)^+ = \{C, E, H\}$ X
$(DEH)^+ = \{D, E, H, A, C, B\}$ ✓

AEH, BEH, DEH.

1NF.

DATE

(21) R(A,B,C,D,E,P,G)

AB → CD
DE → P
C → E
P → C
B → G.

A B C D E P G
↑ ↑ ↑ ↑ ↑

$(AB)^+ = \{A, B, C, D, E, G, P\}$ ✓

1NF

(22) R(A,B,C,D,E,F,G,H)

CH → G
A → BC (PD)
B → CEH
E → A
F → EG

A B C (D) E F G H
↑ ↑ ↑ ↑ ↑ ↑ ↑

AD ✓
BD ✓
CD ✗
DE ✓
DF ✓
DG ✗

DH ✗
DCG ✗
DGH ✗
DCH ✗
DCGH ✗

1NF

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

A → B
B → C
C → BD.

CK: A

∴ since the size of CK is 1.
it will be in 2NF.

DATE

(24) R(A, B, C, D, E, F)

$AB \rightarrow CD$
$CD \rightarrow EF$
$BC \rightarrow DEF$
$D \rightarrow B$
$CE \rightarrow F$

CK:- AB ✓
AC ✗
AD ✓
AE ✗
AF ✗

$(ACE)^+$ ✗
$(ACF)^+$ ✗
$(AEF)^+$ ✗
$(ACEF)^+$ ✗

2NF.

How to normalize a table.

(25) R(A, B, C, D, E, F, G, H, I, J)

$AB \rightarrow C$ (PD)
$A \rightarrow DE$ (PD)
$B \rightarrow F$ (PD)
$D \rightarrow IJ$ (TD)
$F \rightarrow GH$ (TD)

$(AB)^+ = \{A, B, C, D, E, F, G, H, I, J\}$

Table in 1NF.

$R_1(\underline{AB})$

$R_2(\underline{A}, \underline{BC})$

$R_3(\underline{A}, D, E, I, J)$

$R_4(\underline{B}, F, G, H)$

} table in 2NF.

DATE

tables in 3NF.

R_1 (AB)

R_2 (ABC)

R_3 (A DE)

R_4 (D I J)

R_5 (B F)

R_6 (F, G, H)

do not take R_2(A,B) to count.

So, number of tables is (5).