6-Oct-25

Module - 4 / Chapter - 6

Lab project -> 40 in final.

1. Proposal -> 5
2. Schema diagram -> 5
3. E-R diagram -> 6
4. Constraints of the tables -> 6
5. Implementation -> 7
6. Report -> 7
7. Presentation -> 10

4, 5, 6 -> 7 maybe

CRUD

Create
~~Report/Registration~~
Update
Delete

Read (view)

* operation of Simple DB Lab project
* Interface - SDP-2

Module - 04 / Chapter - 6

E-R (Entity-Relationship) Diagram:

* entity sets = Table

[যেসব কোন column multiple value hold করে। (এবং column কে sub-part -> ভাগ করা যায়)

Complex Attributes.

1. Simple : (Age, CT, marks)
   ↓ ↓ ↓ (কে sub-part -> ভাগ করা যায় না, (Ex: Age)

Composite: যেই Attribute এর value কে sub-part -> divide করা যায় যায়

Ex (Name, address))

7-Oct-25

Single-valued attribute: একটি value এবং row ও
থাকে (CGpa, Age)

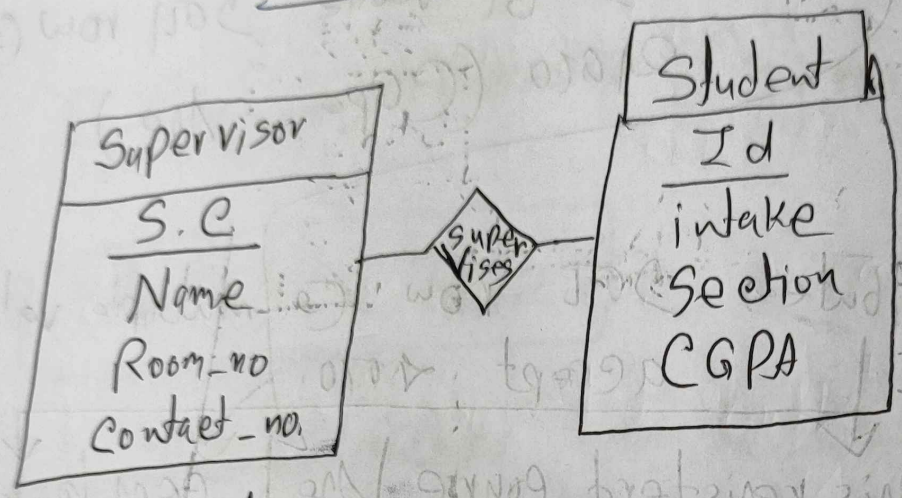
multi-valued attribute, একটি row ও multiple value
accept করে.

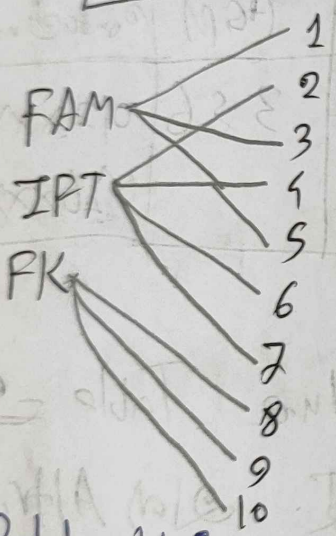
ID	~~d~~ registerd course	Age	CGPA	contact no
1	BAN 101, CSE 201, EE 101, MAT 231	22	3.56	017xxx, 016xxxx

Derived attribute: যেই Attr. এর value Table এ
নাই কিন্তু অন্য Attr. এর
সাহায্যে নিয়ে derive
করা যায়।

ID	DOB	Age
1	10.10.1993	
2	7.5.1999	

Mapping Cardinality Constraints

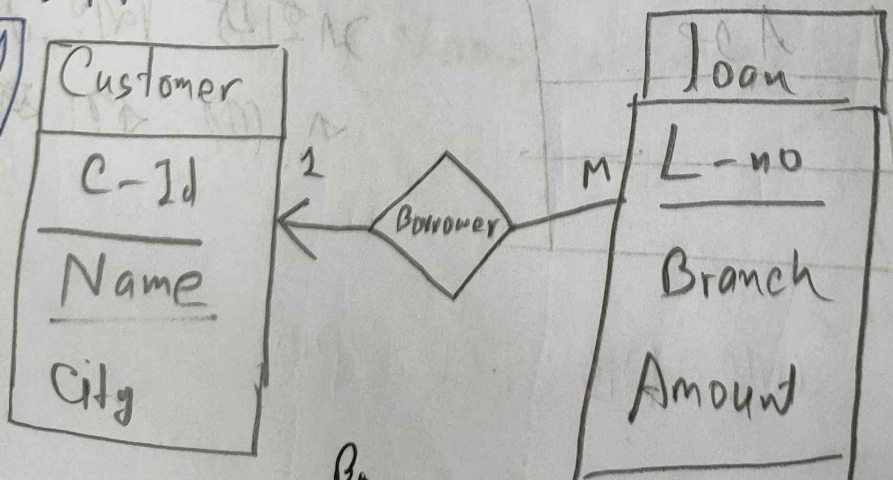


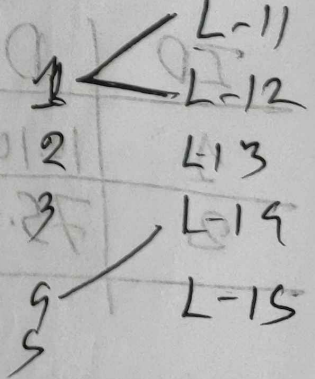


একটা table এর একটা row (entity) আরেকটা table এর কয়টা row (৭) এর সাথে Connected বা যেভাবে represent করে তাকে Mapping Cardinality বলে।

Relationship

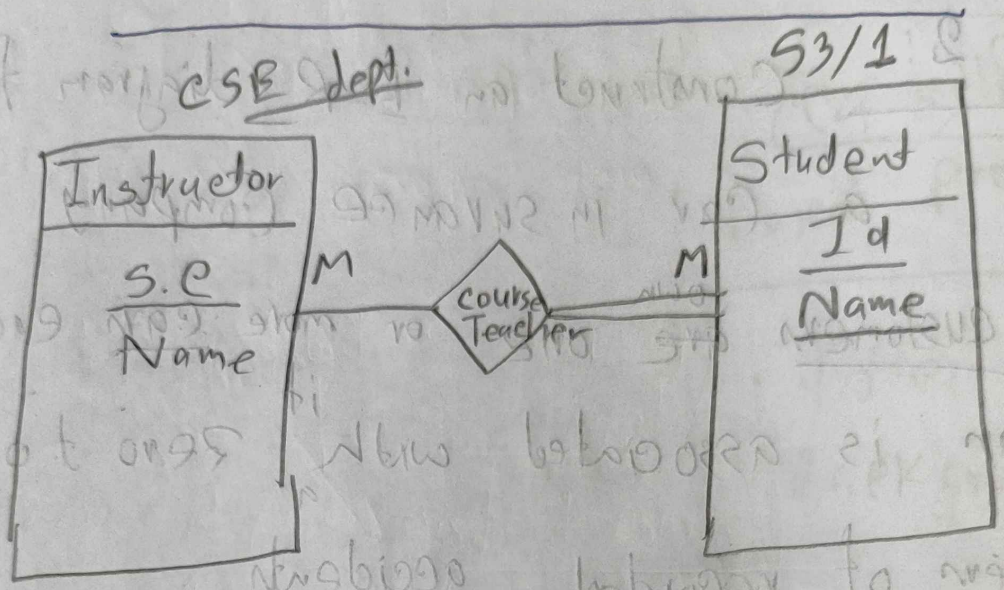
Mapping C. 2





Borrower/Borrows

Total and Partial Participation



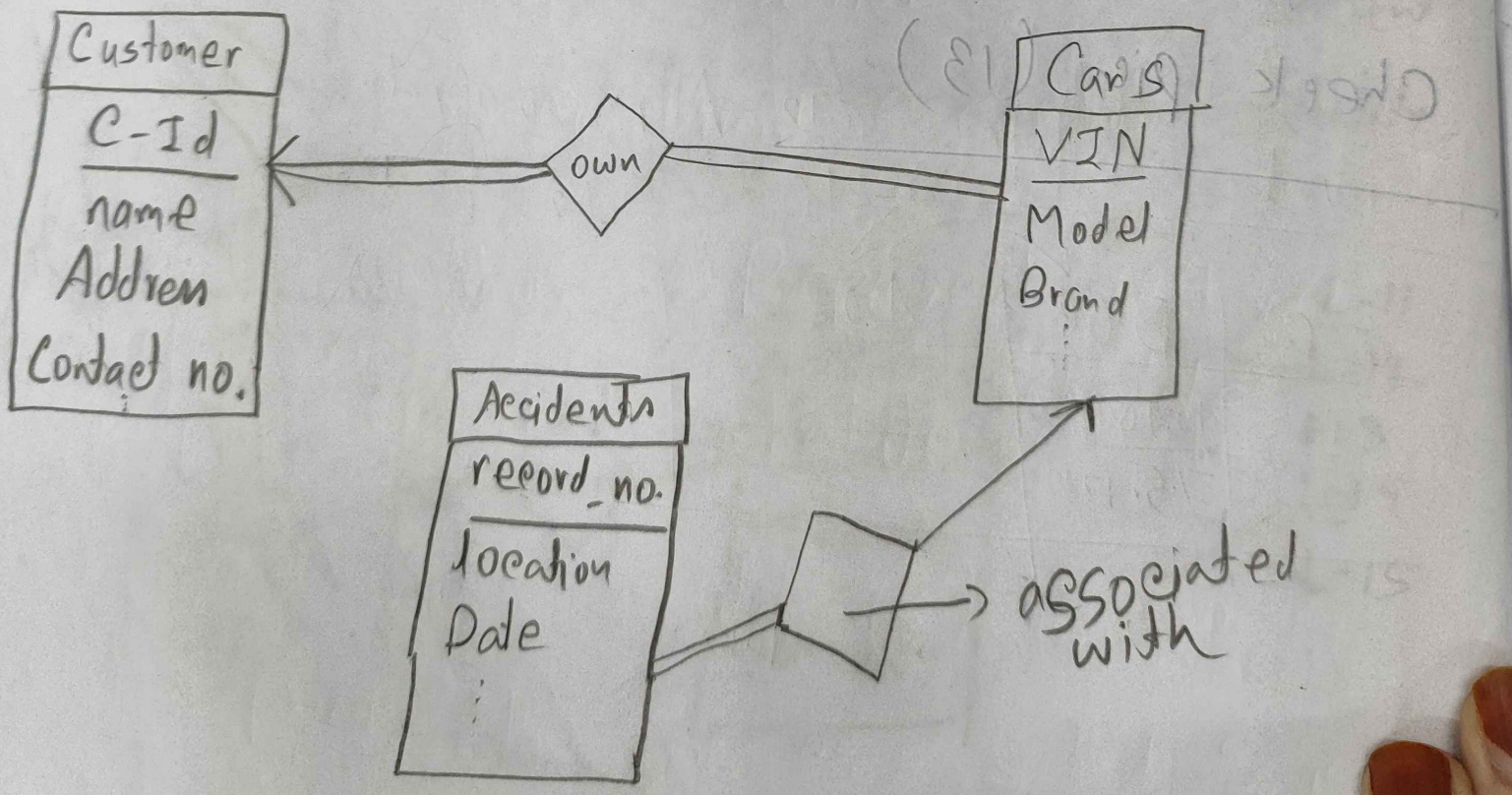
13-OCT-25

Check pic (13)

Check Pic (14) | 14-OCT-25

VIN -> Vehicle Identification num.

Example 2: Construct an E-R diagram for a car insurance company whose customers ~~are~~ own one or more cars each. Each car is associated with it zero to any numbers of recorded accidents.



English

Module 15: (Transaction)

(book 14/15)

(C-S) কি slide)

ACID

Consistence → সামঞ্জস্য

A = 100
B = 200

A+B = 300

A = 50
B = 250

A+B = 300

Isolation: যেসব Concurrent Schedule যে Conflict serializable হয় ~~ইত্যাদি~~ তারাই ~~C.S~~ Concurrent. S Isolation property maintain করে

Durability: ~~DB তে পরিবর্তন change~~

Project proposal

Title

Background:

Motivation:-

Key features:-

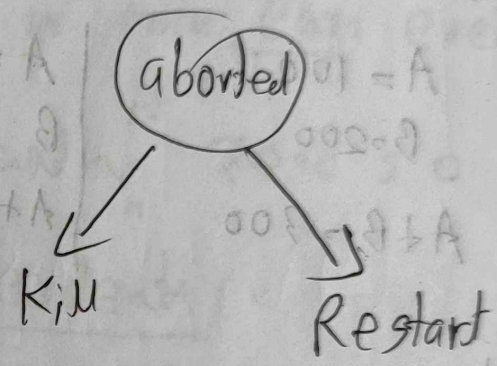
Used tools & language:

5 → marks

20-OCT-25

Atomicity → সব (৬টা) operation হলে then db update হবে নইলে হবে না.

(2) Commit Command



Concurrent ও বিপরীত serial

Schedule 3 → Concurrent execution.

প্রতিটি Transaction ও Represent করে schedule.

(Local buffer)

S-2

A = 200, B = 400 ; A + B = 600

T_2	T_1
A = 200	A = 180
Temp = 20	A = 130 (local buffer)
A = 180 (local buffer)	DB: A = 130
DB: A = 180	B = 420
B = 400	B = 470
B = 420	DB: B = 470
DB: B = 420	A + B = 130 + 470 = 600
A + B = 480 + 420 = 600	

* Consistent.

* Serial হলে কী guarrented যে সঠিক consistent.

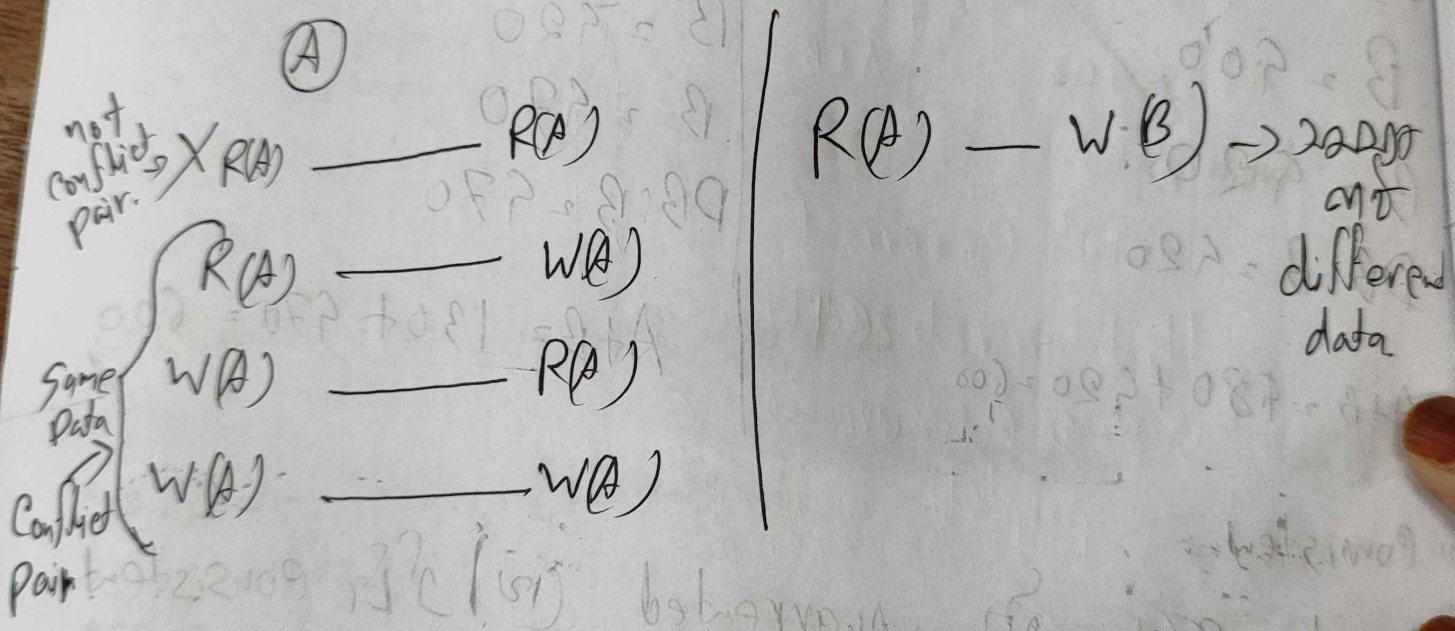
* Concurrent = Consistent হলে আর না হলে না.

21 OCT 25

Precedence graph

Next ... CT

Conflict serializability:-

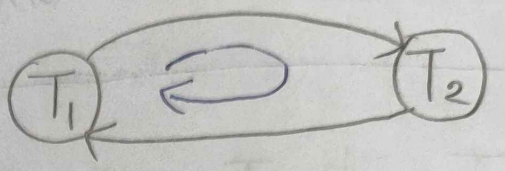


If a concurrent schedule (S) can be converted to a serial ~~seria~~ schedule (S') by avoiding conflict pairs then we can say that S is Conflict Serializable.

Schedule - 4

T_1	T_2
R(A)	
	R(A)
	W(A)
	R(B)
W(A)	
R(B)	
W(B)	
	W(B)

Precedence graph

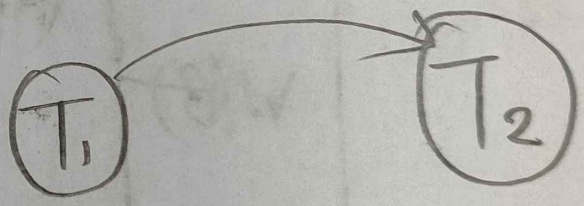


Cycle ⇒ Not conflict serializable.

↳ Inconsistant.

Schedule - 3

T_1	T_2
R(A)	
W(A)	
	R(A)
	W(A)
R(B)	
W(B)	
	R(B)
	W(B)



No cycle ⇒ Conflict serializable

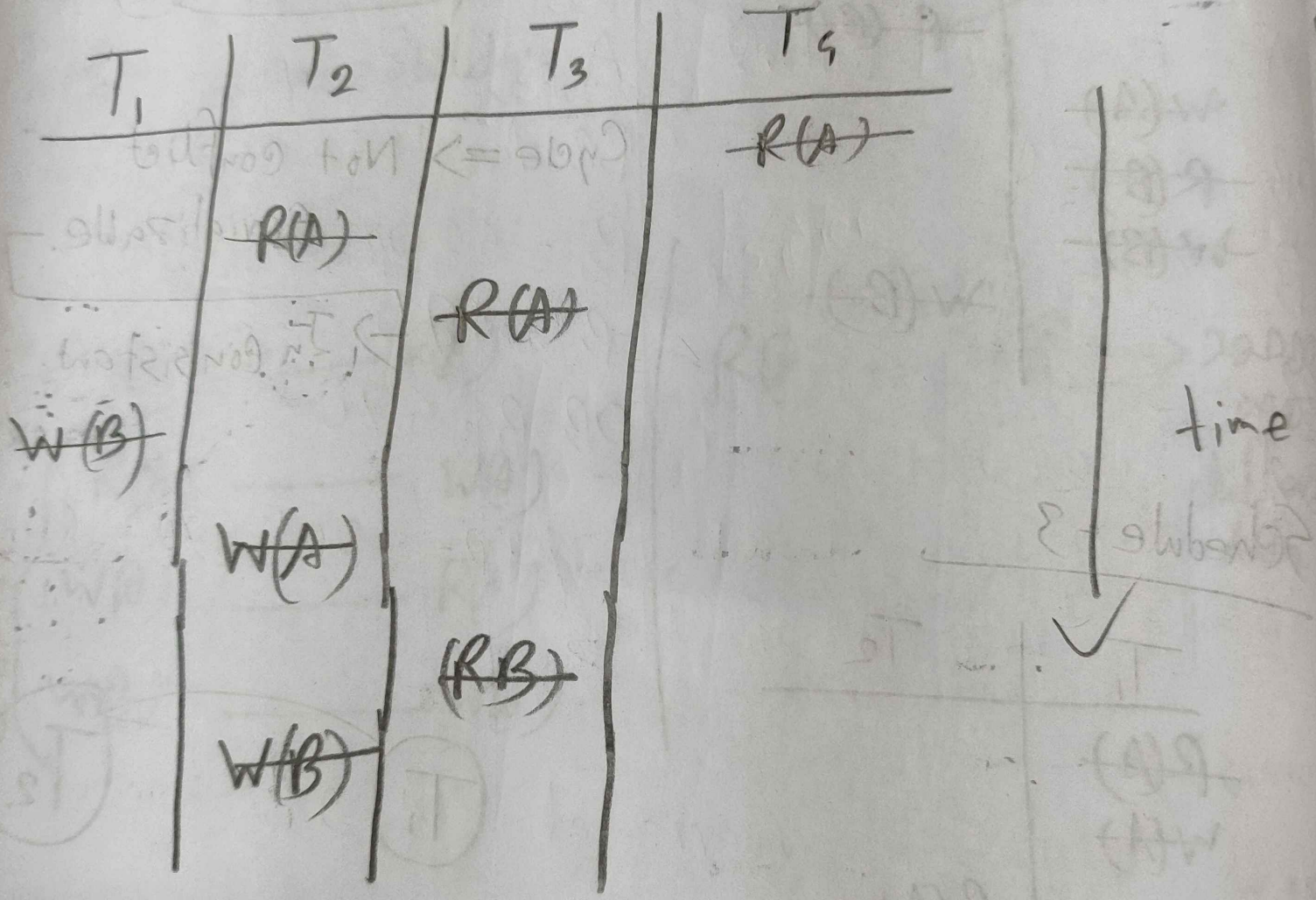
↳ Consistant.

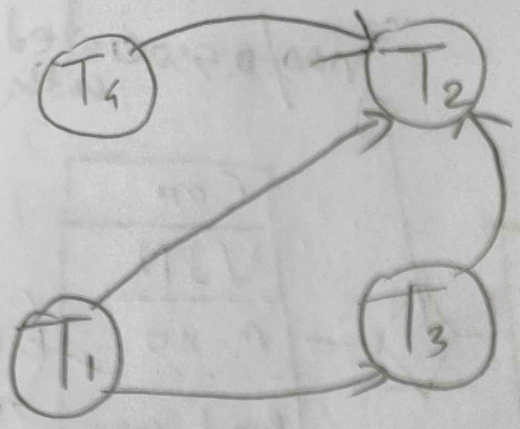
check Pick (22)

27 OCT-25

Conflict serializability:

Q. Is the schedule Conflict serializable?



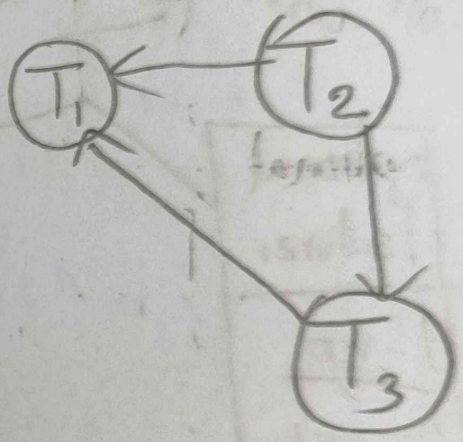


No cycle =

Serializable

#

T_1	T_2	T_3
R(x)		
		R(y)
		R(x)
	R(y)	
	R(z)	
		W(y)
	W(z)	
R(z)		
W(x)		
W(z)		



No cycle =

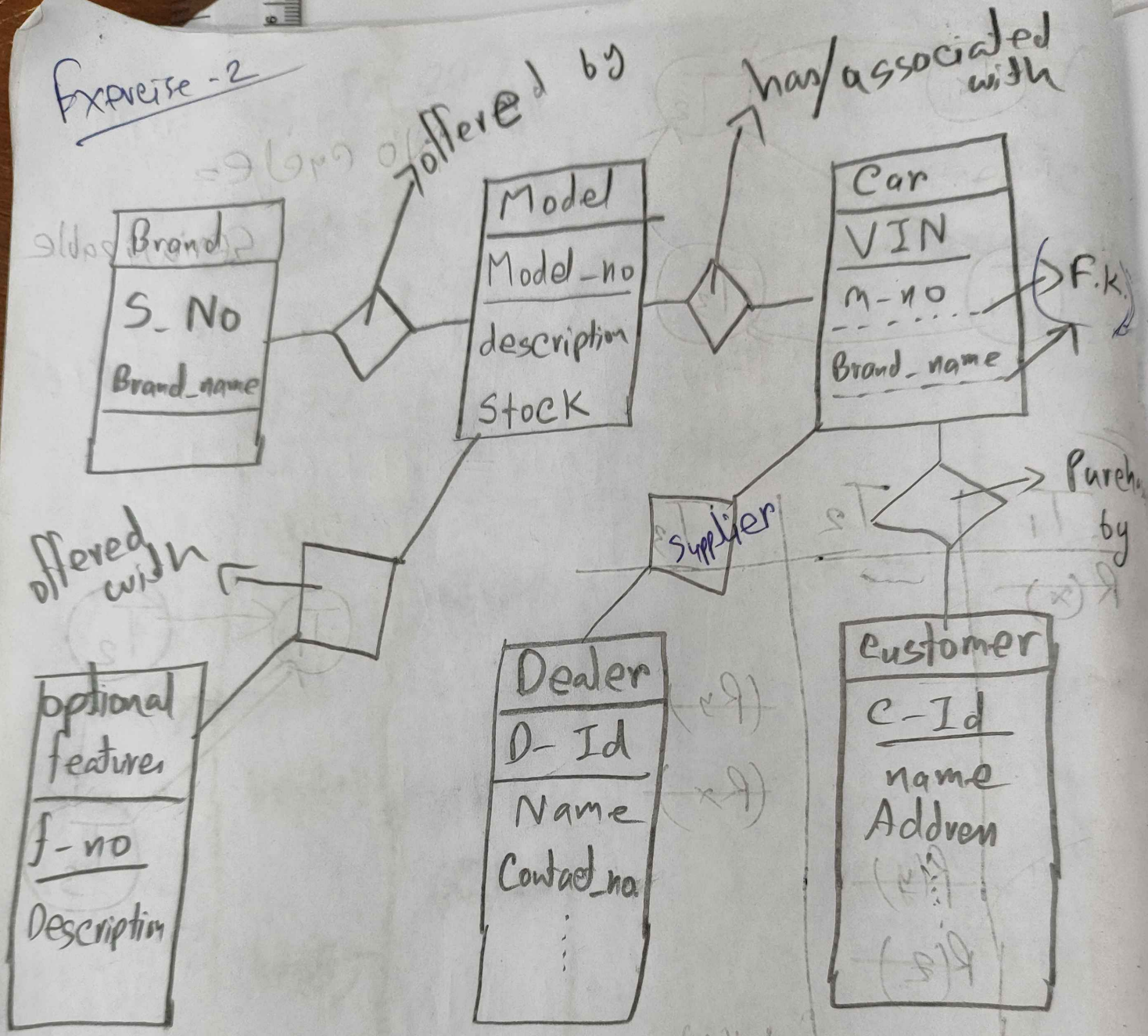
conflict

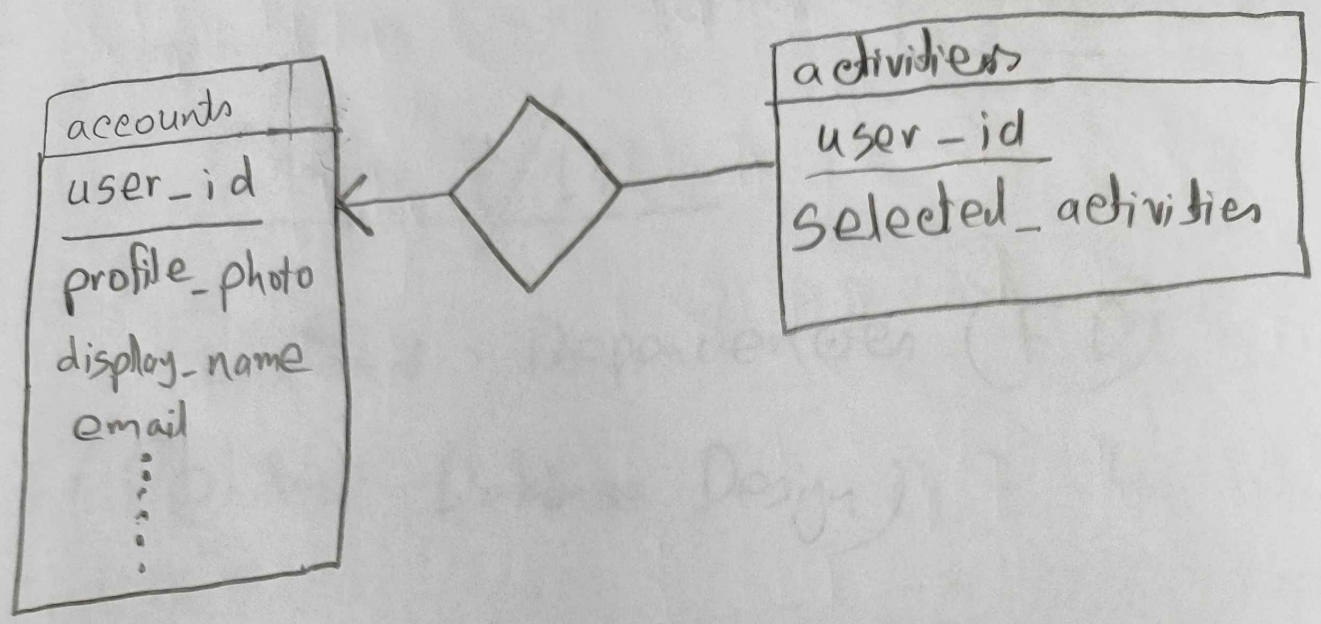
Serializable

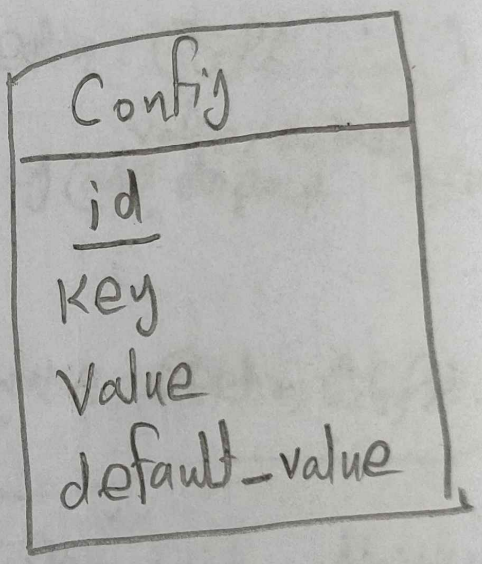
→ Consistency maintain ✓✓

→ isolation Property maintain ✓✓

Exercise - 2


offered by
has/associated with
Brand
S_No
Brand_name
Model
Model_no
description
Stock
Car
VIN
m-no
Brand_name
F.k
offered with
optional features
f-no
Description
Supplier
Dealer
D-Id
Name
Contact_no
Purch
by
Customer
C-Id
name
Addres


accounts
user_id
profile_photo
display_name
email
activities
user_id
selected_activities


Config
id
key
value
default_value

3-Nov-25

Chapter 7/11

Functional Dependencies (F.D)
(Relational Database Design)

Student(Id, Name, Intake, Section, CGPA ......)

→ Id এর উপর বাকি attributes গুলোর value/detail depend করে।

F.D.

Id → Name, Intake, Section, CGPA ......

Id: Determinant attribute

Name, Intake, Section, CGPA ......: Dependent attribute

Id, Name → - - -

$A \longrightarrow BCDE$ ✓ valid dependency

$B \longrightarrow DE$ ✗ Invalid

$D \longrightarrow BC$ ✗

Rule of F.D.

1. **Transitivity rule:** If, $X \to Y$ & $Y \to Z$ then $X \longrightarrow Z$

2. **Union rule:** If $X \longrightarrow Y$ & $X \to Z$ then $X \to YZ$

3. **Decomposition rule:** If $X \longrightarrow YZ$ then $X \longrightarrow Y$ and $X \to Z$

4 Pseudo transitivity : If $X \to Y$ &

$YZ \to W$

then

$XZ \to W$.

5. Augmentation rule: If $X \to Y$ ~~the~~

then

$XZ \to YZ$

#

$R(A, B, C, G, H, I)$

$F = \{A \to B, A \to C, CG \to H, CG \to I, B \to H\}$

Closure of F / F^+ (আরো যেসব F.D. পাওয়া যাবে তা বের করা)

$F^+ = \{$ $A \to BC$ [Union]
$CG \to HI$ [4]
$A \to H$ [Transitivity]
$AG \to H$ [P.T.]
$AG \to I$ [4] $\}$

9-Nov-23

No ~~tidirec~~ bidirectional

$B \rightarrow A$
$D \rightarrow BC$ $\rightleftharpoons$
$E \rightarrow BC$

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

$F = \{ A \rightarrow BC$
$CD \rightarrow E$
$B \rightarrow D$
$E \rightarrow A \}$

Find F^+ / Find the closure of F

$\Rightarrow F^+ = \{ BC \rightarrow E$ [Pseudo transitivity]
$CD \rightarrow A$ [transitivity]
$A \rightarrow B$ & $A \rightarrow C$ [Decomposition]

R(A, B, C)

$F = \{ A \to BC$

$B \to C$

$A \to B$

$AB \to C \}$

$\Rightarrow F^+ = \{ A \to C$ [Decomposition]

Closure of attribute sets

Find $A^+, B^+, (AB)^+$ [A এর ক্লোজার কোন কোন Attribute dependent তা বের করা = A^+]

$\Rightarrow A^+ = ABC$

$B^+ = BC$

$(AB)^+ = ABC$

R(A, B, C, G, H, I)

$F_2 = \{$
$A \to B$
$A \to C$
$CG \to H$
$CG \to I$
$B \to H$ $\}$

Find A^+, B^+, C^+, $(CG)^+$

Sol^n:

$A^+ = ABCH$

~~$B^+ = BAHCGI$~~

$C^+ = C$

$(CG)^+ = CGHI$

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

F = { A → BC
CD → E
B → D
E → A }

Find A+, B+, C+, (CD)+, E+ / F+

⇒

A+ = ABCDE

B+ = BD

C+ = C

(CD)+ = CDEAB

E+ = EABCD

~~F+ = {~~

F+ = check previous page.

10-Nov-2

Finding Super key, Candidate key & Primary key from Functional dependency sets:-

R(A, B, C)

F = { A → BC
B → C
A → B
AB → C }

find SK, CK, PK for this table.

⇒ First calculate the ~~co~~ closure of all determinant part/attribute

∴ A+, B+, (AB)+

$A^+ = ABC$ →
$B^+ = BC$ → Key হবে যেসব attribute
$(AB)^+ = ABC$ → থাকবে ABC তে
এগুলো।

$\therefore$ S.K. = A, AB

CK = A

PK = A

এগুলো হবে

S.K. = A, E, CD

CK = A, E

PK = A / E

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

F = { AB → C
BC → AD
D → E
CF → B }

BCF+
CD+
D+

⇒ (AB)+ = ABCDE.

(BC)+ = BCADE

D+ = DE

(CF)+ = CFBADE ✓

∴ SK = CF
CK = CF
PK = CF

$(ABF)^+ = ABCDEF$

$(BCF)^+ = BCADEF$

∴ ABF, BCF is also possible super key.

R(A, B, C, G, H, I)

F: { A → B
A → C
CG → H
CG → I
B → H }

⇒ $A^+ = ABCH$

$B^+ = BH$

~~$C^+ = C$~~

$(CG)^+ = CGHI$

$(AG)^+ = AGBCHI$

~~$(AG)^+ = ABCHGI$~~

$(ACG)^+ = ACGHIB$

Sk. ⇒ AG, ACG

CK ⇒ AG

PK ⇒ AG

⇒ যেকোন একটা হলেই possibility একটা।

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

F = { AB → C
BC → AD
D → E }

→ $(AB)^+$ = ABCDE

$(BC)^+$ = ~~ABCDE~~ BCADE

D^+ = DE

consider

∴ ~~AB~~ $(ABF)^+$ = ABFCDE

$(BCF)^+$ = BCFADE

S.K. = ABF, BCF

C.K. = ABF, BCF

P.K. = ABF / BCF

11-Nov-25

Q3: R(A, B, C, D, E, H)

$F = \{ A \to B,\ BC \to D,\ E \to C,\ D \to A \}$

⇒

$A^+ = AB$

$(BC)^+ = BCDA$

$E^+ = EC$

$D^+ = DAB$

$(BCEH)^+ = ABCDEH$

S.K. = BCEH

C.K. = BCEH

P.K. = BCEH

V.V.I topic

Canonical cover ***

* যেসব dependency গুলো সবচেয় Prime.

R(A, B, C)

$F = \{ A \to BC,\ B \to C,\ A \to B,\ AB \to C \}$

findout the canonical cover.

=> Step: 1: যেখানে decomposition rule apply করা যাবে সেখানে করতে হবে

Apply decomposition rule if possible.

$F_2 = \{ A \to C,\ B \to C,\ A \to B,\ AB \to C \}$

Step-2: Check whether $A \to C$ is necessary or not.

$A^+ = ABC$

$-(A)^+ = ABC$

So, $A \to C$ isn't necessary.

$F_2 = \{B \to C$
$A \to B$
$AB \to C\}$

Step-3: Check whether $B \to C$ is necessary or not.

$B^+ = BC$

$-(B)^+ = B$

$\therefore B \to C$ is necessary.

$F_2 = \{B \to C$
$A \to B$
$AB \to C\}$

Step-4: Check whether $A \to B$ is necessary or not

$A^+ = ABC$

$- (A)^+ = A$

$\therefore A \to B$ is necessary

$F = \{B \to C$
$A \to B$
$AB \to C\}$

Step-5: Check whether $AB \to C$ is necessary or not

$(AB)^+ = ABC$

$- (AB)^+ = ABC$

$\therefore$ So $AB \to C$ isn't necessary.

So the canonical cover is,

$$F = \{ B \rightarrow C, \quad A \rightarrow B \}$$

17-Nov-25

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

F = { A → BC
CD → E
B → D
E → A }

Using the functional dependencies compute the canonical cover of F.

যা বুঝতে না পারবি।

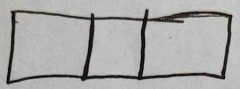
Chp-12 / 14 (7th edition)

(Indexing & Hashing)

indexing এর ক্ষেত্রে structure follow করা হয় a Binary search Tree এর মত।

B+ Tree:

n = order of B+ Tree



⇒ Each leaf node can ~~store~~ contain maximum (n-1) value.

⇒ " " , have to " at least $\lceil \frac{n-1}{2} \rceil$ value.

#Example:

leaf node 2, 3, 5, 7, 11, 17, 19, 23, 29, 31 (n = 4)

Max = 3
Min = 2

Check pin (18)

18-Nov-25

2, 3, 5, 7, 11, 17, 19, 23, 29, 31 (n=6)

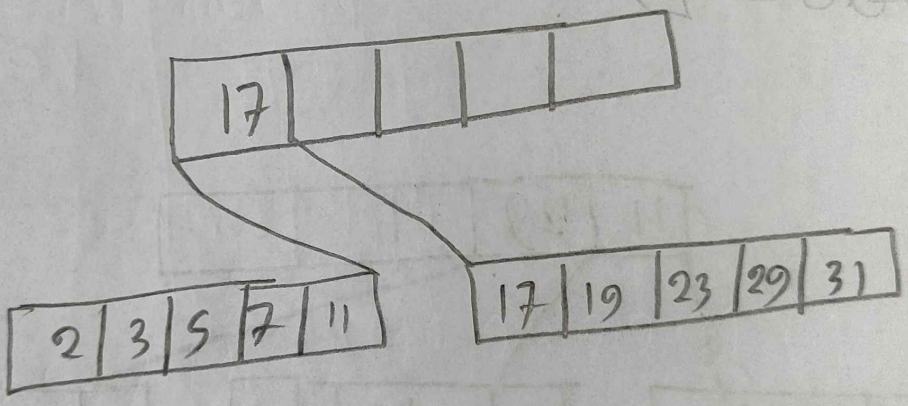
B+ tree

Leaf node

Max (6-1) = 5

Min = $\lceil \frac{6-1}{2} \rceil$ = 3

next week
মঙ্গলবার (25.11.25)
CT-3
Chp-7
+
B+ tree

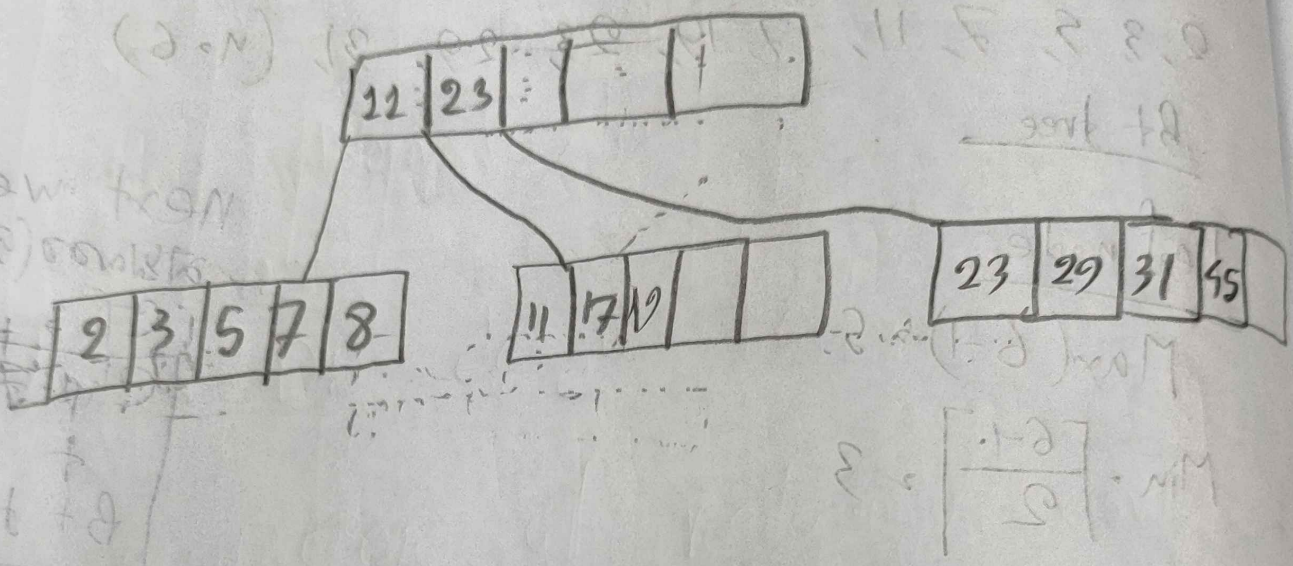


(i) Insert 8

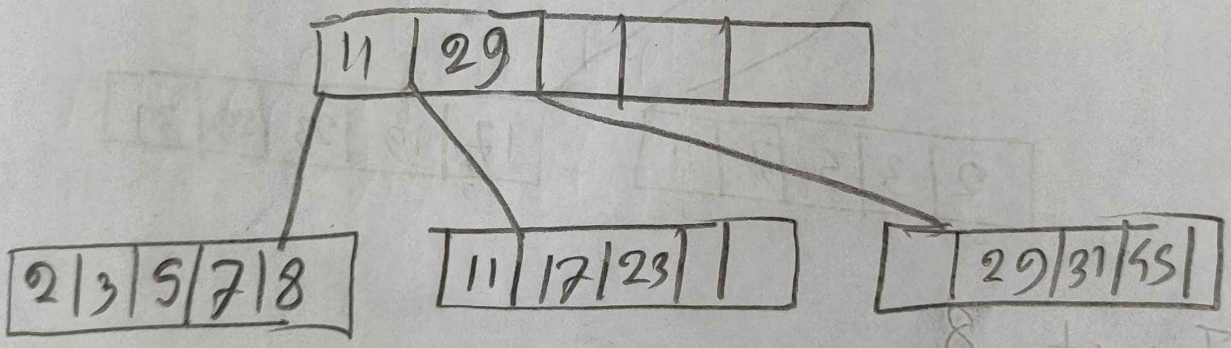
(ii) Insert 45

P.T.O.

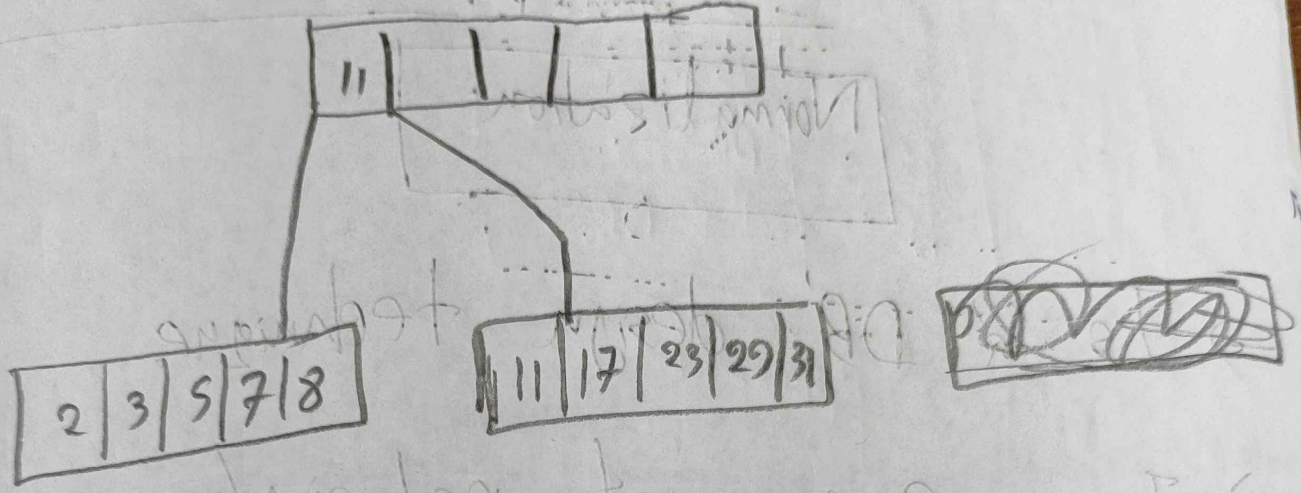
(i) + (ii)



(iii) Delete 29



(IV) Delete 45



Chapter-7 ৭৭ বাবি প্রশ্ন

Topic

Normalization

→ Is a DB design technique.

→ Is a procen of reducing/removing redundancy from DB.

Emp-Id	Name	Contact No.	Dept.	Dept_manager
101	A	0171	Eng	FH
102	B	0173	KR	MR
103	C	0175	MK	MJ
104	D	0176	Eng.	FH
105	E	0179	MK	MJ
101	A	0171	Eng.	FH

① Row wise Redundancy

② Column "

↳ এটা স্বাভাবিক।

↳ Storage এর জন্য harmful.

Emp_Id কে P.K. করলে Solve.

[P.K দিলে row wise redundancy solve হয় না]

তিনটি problem arise করতে:

① Insertion anomaly

② Delete "

③ Update "

NF = Normal Form

1NF, 2NF, 3NF

1NF :- ① An attribute of a table can not hold multiple values.

P.T.O

Student

Std_Id	Name	Course
101	A	OS, DBMS
102	B	OS
103	C	DBMS
104	D	Algo, DBMS

একটা solution approach:

PK →

Std_Id	Name	Course1	Course 2
101	A	OS	DBMS
102	B	OS	NULL
103	C	DBMS	NULL
104	D	Algo	DBMS

আরেকটা solution approach:

Composite PK

Std_Id	Name	Course
101	A	OS
101	A	DBMS
102	B	OS
103	C	DBMS
104	D	Algo
104	D	DBMS

multivalued attribute এর ক্ষেত্রে already 1NF → হবে।

19-Nov-25

2NF:

1. Table must be in 1NF
2. Doesn't allow Partial functional dependency.

Student → or this. → Composite PK

Std_id	Name	Addren	Course no.	Course title	Credit	Grade
101	A	DH	207	DB	3	A+
101	A	DH	208	DB-lab	1.5	A+
102	B	CH	207	DB	3	A+
102	B	CH	208	DB-lab	1.5	A

Std_Id, Course_no → Name, Addren, Course_title, Credit, Grade.

(Complete Functional dependency)

Std_Id → Name, Addren

(Partial Functional dependency)

Course_no → Course_title, Credit

[System 2NF → allowed না]

Table টাকে split করে দেখলাম: (Solution)

PK std

Std_id	Name	Addres
101	A	DH
102	B	CH

PK Course

Course_no	Course_title	Credit
207	DB	3
208	DB-lab	1.5

Composite P.K.

Std_id	Course_no	Grade
101	207	A+
101	208	A+
102	~~208~~ 207	A+
102	208	A