

1. Good Evening

2. Lecture begins at 9:05pm

3. Topic - Backlog & Misc (Part 1)

---

## AGENDA

1. View ]

✓ ☆

2. CONDITIONALS → IF, CASE

✓ ☆

IFNULL, COALESCE

3. Windowing Fns, PARTITION BY

✓ ☆

4. Union & Union ALL

✓ ☆

5. CTE ▷ Recursive CTE

✓

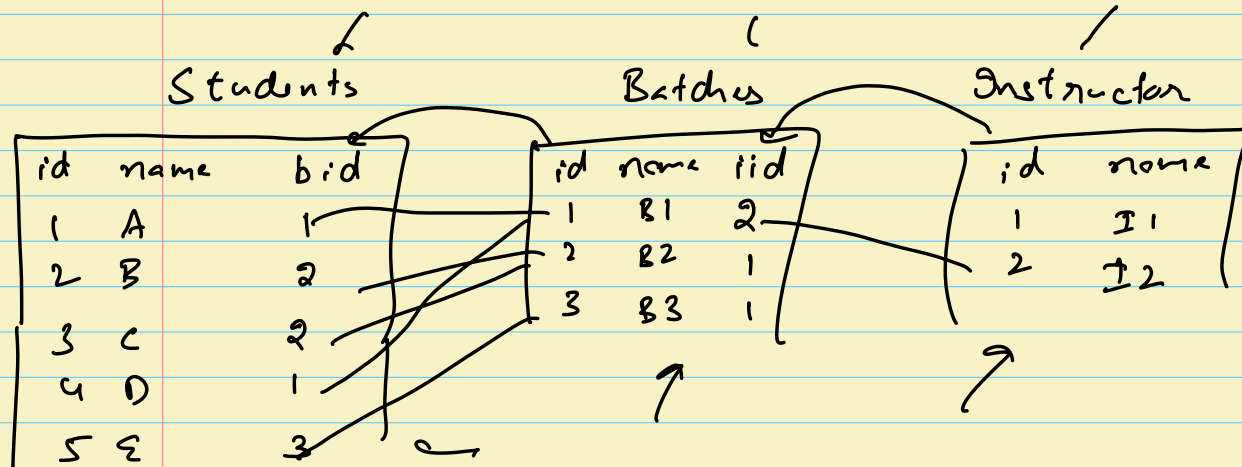
6. Serializable ]

✓

7. Deadlocks ]

✓

## Views : Stored Queries.



Business Analyst might request some kind of data e.g. which instructor is teaching which students.

- [He doesn't know how to write query
- [ Devs should not show db without proper authorization. ✓

tid	iname	sid	sname	
1	I1	2	B	S1
1	I1	3	C	S2
1	I1	5	E	S3
2	I2	1	A	S4
2	I2	4	D	S5

CREATE Inst\_Student\_vw VIEW AS

```
SELECT i.id, i.name, s.id, s.name  
FROM Instructor i  
JOIN Batches b ON i.id = b.iid  
JOIN Students s ON b.iid = s.bid
```

Select \* FROM Inst\_Student\_vw.

Q1. Can views have data?

↳ Usually no. But if we create an index on a view will lead to the view having its own data. Such a view is called materialised view.

Details of Materialized Views  
Insert via Views

Monday.

1. Views are stored queries.

2. Views don't store data.

3. Views can store data if we create an index on it

Q 4. Such a view is called materialised view.

5. Materialised view will store lookup data.

6. Can view be used for delete,

insert & update also? : Yes

with conditions. { view must represent mandatory column of all tables }

Tbl1  
id name call  
Tbl2  
id name call2

Create VIEW V1 AS

Select id, name, call, id, name

FROM Tbl1

JOIN Tbl2 ON Tbl1.id = Tbl2.id

# CONDITIONALS

1 IF

## — CASE

→ IFNOLL

## — COALESCE

IF (price > 100, "price", "Inexpensive")

# CASE

WHEN price < 50 THEN "LI"

WHEN  $p_{\text{row}} < 100$  THEN "L2"

WHEN  $\rho_{\text{pic}} < 200$  THEN "L3"

3573 "ly"

ΣND

## Students

id	name	psp		id	name	psp
1	A	50	↙ P3	1	A	No
2	B	60	↘ P3	2	B	No
3	C	70	↙ P2	3	C	No
4	D	80	→ P1	4	D	Yes
5	E	90	→ P0	5	E	Yes

```

SELECT id, name,
       IF (psp ≥ 80, "Yes", "No")
       AS ELIGIBLE
FROM Students;

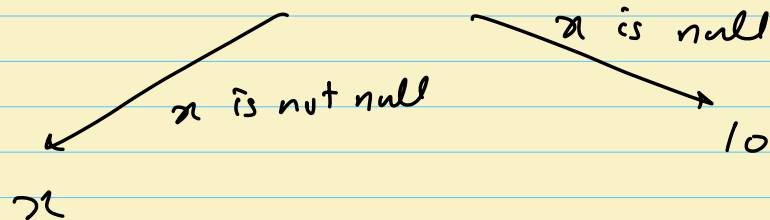
```

```

SELECT id, name,
       CASE
         WHEN psp ≥ 90 THEN "P0"
         WHEN psp ≥ 80 THEN "P1"
         WHEN psp ≥ 70 THEN "P2"
         ELSE "P3"
       END AS Priority.
FROM STUDENTS

```

IFNULL ( x, 10 )



COALESCE ( x, y, z, 10 )

if x is not null  $\rightarrow$  x

" " x is null, but y is not  $\rightarrow$  y

if x & y are null, but z is not  $\rightarrow$  z

if x, y & z are null  $\rightarrow$  10

$\rightarrow$  IF, CASE, IFNULL, COALESCE

views

- $\rightarrow$  Materialized Views
- $\rightarrow$  CRUD on views
- $\rightarrow$

BREAK = 10:15 to 10:25

## WINDOWING FNS X PARTITION BY

Students				Row-N	R	DR
id	name	brd	psp			
1	A	1	80 1	1	1	1
2	B		50 3	3	3	3
3	C		70 2	2	2	2
4	D	2	30 2	6	6	5
5	E		40 1	5	5	4
6	F	3	30 2	7	6	5
7	G		50 1	4	3	3
8	H		10 3	8	8	6

SELECT id, name, psp,  
ROW-NUMBER() OVER (psp DESC)  
 FROM Students

↓

RANK()

DENSE\_RANK()

↑ ORDER BY



```
SELECT id, first-name, brd, psp  
ROW-NUMBER() OVER (PARTITION BY did ORDER BY  
FROM students psp DESC)
```

---

UNION & UNION ALL

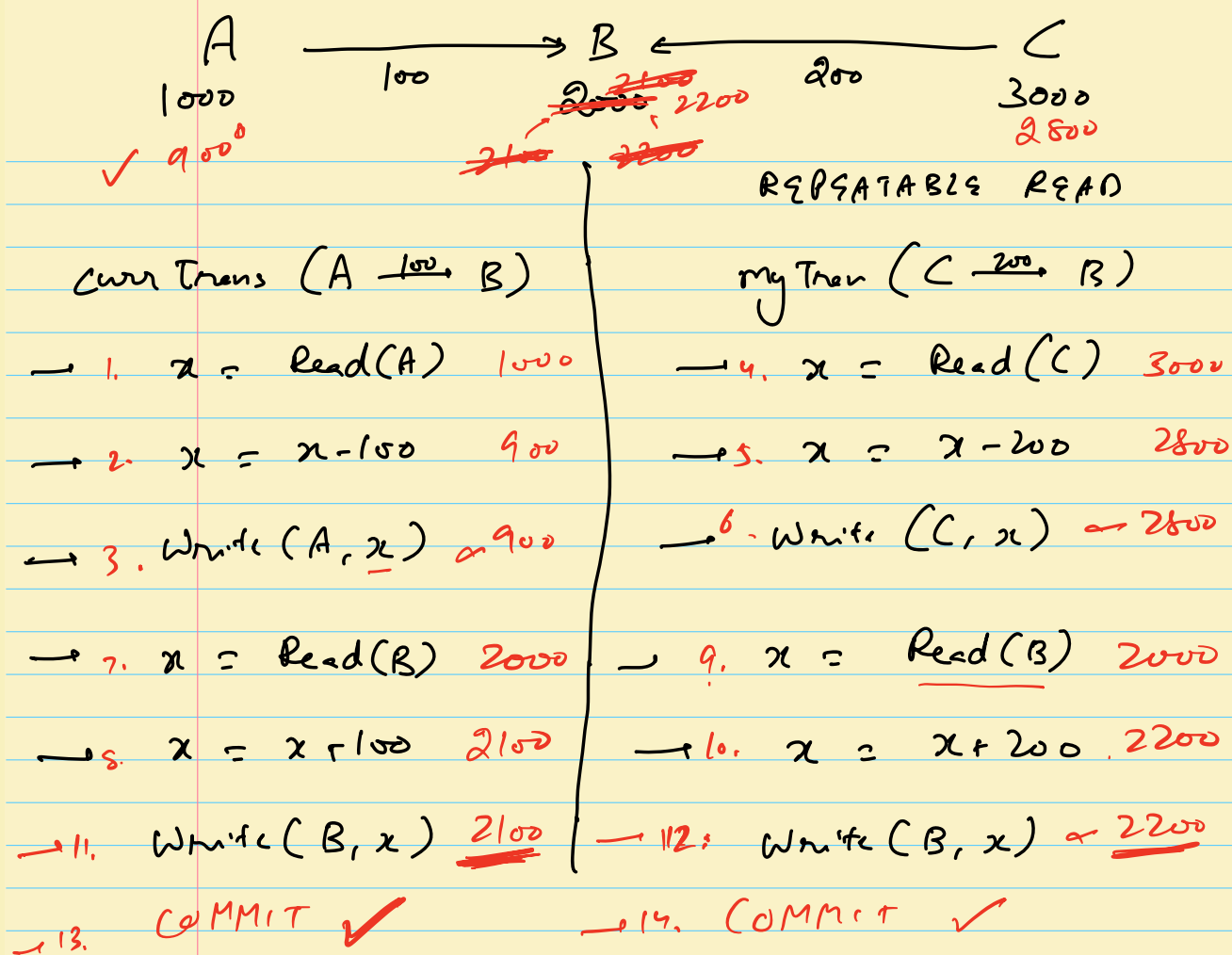
↳ On Workbench.

---

## SERIALIZABLE

Four Isolation Levels

1. READ UNCOMMITTED ✓
  2. READ COMMITTED ✓
  3. REPEATABLE READ ✓
  4. SERIALIZABLE
- } → Don't honor locks
- ] → Honors locks.



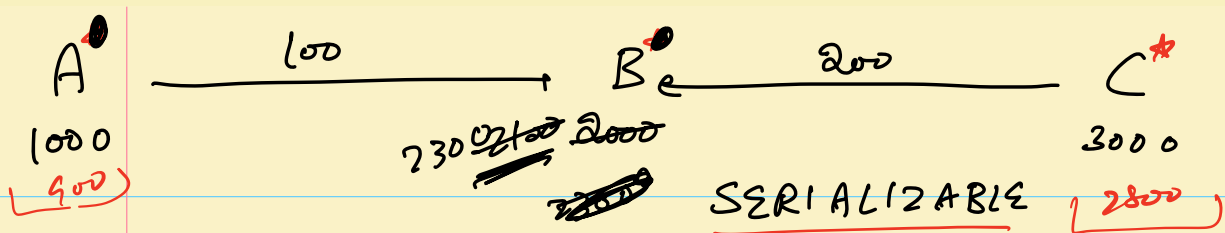
## SOLUTION

1. T1 should take locks

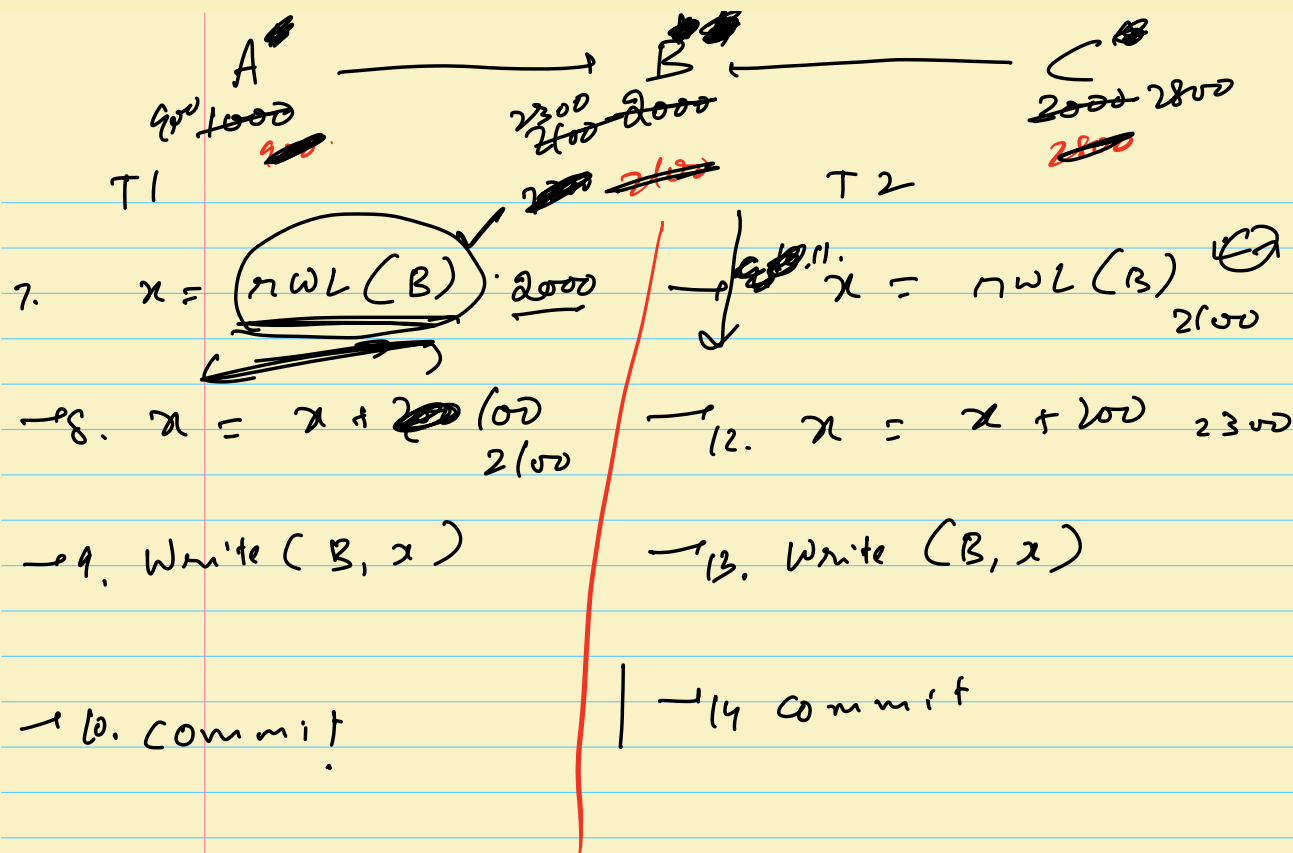
Update, Delete, Insert

Select FOR UPDATE

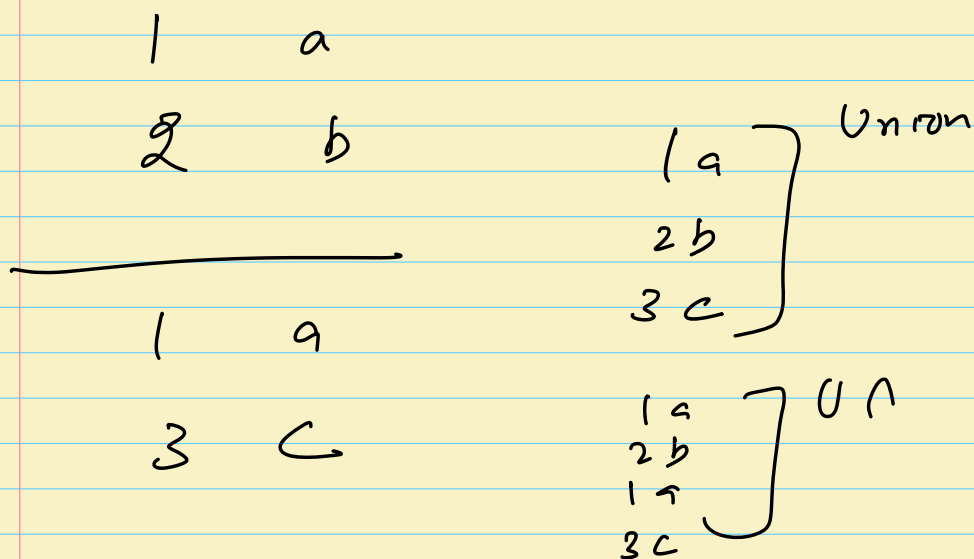
2. T2 should have isolation level for Serializable.



- |  |   |
|--|---|
| <p>T1</p> <p>→ 1. <math>x = \text{readWlock}(A)</math> 1000</p> <p>→ 2. <math>x = x - 100</math> 900</p> <p>→ 3. <math>\text{Write}(A, x)</math> 900</p> <p>→ 7. <math>x = \text{rwl}(B)</math> 2000</p> <p>→ 8. <math>x = x + 100</math> 2100</p> <p>→ 9. <math>\text{Write}(B, x)</math> 2100</p> <p>10. <u>Commit</u></p> | <p>T2</p> <p>→ 4. <math>x = \text{readWlock}(C)</math> 3000</p> <p>→ 5. <math>x = x - 200</math> 2800</p> <p>→ 6. <math>\text{Write}(C, x)</math> 2800</p> <p>→ 11. <math>x = \text{rwl}(B)</math> 2100</p> <p>→ 12. <math>x = x + 200</math> 2300</p> <p>→ 13. <math>\text{Write}(B, x)</math> 2300</p> <p>14. <u>Commit</u></p> |
|--|---|



## DOUBTS



SELECT A

FROM BA

WHERE name = 'B'

FOR UPDATE,