

The "Check" constraint is used to limit the value range that can be placed in a column.

e.g. Age int check (Age >= 18).

Select command.

from से वला याल देता है।

(2).

Ans: S1.

सिवा S1 id निकले पर compiler Coagre होता है।

कि T₁ में से का T₂ में से।

cross product ने दो table same नहीं समझते।
अगर (2) different table हो सकते हैं।

DEFAULT

The "DEFAULT" constraint is used to set a default value for a column.

CITY varchar(255) DEFAULT 'Delhi'.

Self Join:

→ से इसका cross Product

Date: / / Page no: / /

S_id	C_id	Since
S ₁	C ₁	2016
S ₂	C ₂	2017
S ₁	C ₂	2017

S_id	C_id	since
S ₁	C ₁	2016
S ₂	C ₂	2017
S ₁	C ₂	2017

Study.

Study.

Ques. find the student id who is enrolled in atleast two courses.

Select (T₁.S_id) from Study as T₁, Study as T₂
where T₁.S_id = T₂.S_id

and

T₁.C_id <> T₂.C_id

↓

not equal to.

The table with the foreign key is called the child table, and
 The table with the Primary Key is called referenced or
 Parent table

Date: / / Page no: _____

find the Emp name who worked in a department
 having location same as their address?

"Dasbhoole"

DEFAULT SET

Persons

ALTER TABLE

EMP				Dept.		
E_No	E_name	Address		Dept_no	Location	E_no
1	Ram	Delhi		D ₁	Delhi	1
2	Vasun	Chd		D ₂	Pune	2
3	Ravi	Chd		D ₃	Patna	4
4	Amit	Delhi				

Select E_name from Emp, Dept where
 and Emp.E_no = Dept.E_no. and
 Emp.address = Dept.location

(Ans Ram)

Emp.			Dept		
1	Ram	Delhi	D ₁	Delhi	1
1	Ram	Delhi	D ₂	Pune	2
1	Ram	Delhi	D ₃	Patna	4
2	Vasun	Chd	D ₁	Delhi	1
2	Vasun	Chd	D ₂	Pune	2
2	Vasun	Chd	D ₃	Patna	4
3	Ravi	Chd	D ₁	Delhi	1
3	Ravi	Chd	D ₂	Pune	2
3	Ravi	Chd	D ₂	Patna	4
4	Amit	Delhi	D ₁	Delhi	1
4	Amit	Delhi	D ₂	Pune	2
4	Amit	Delhi	D ₃	Patna	4

Left Outer Join.

It gives the matching rows and the rows which are in left table but not in right table.

Emp_no.	E-name	Dept_no.		Dept_no.	D-name	Loc
E ₁	Vasun	D ₁		D ₁	IT	Delhi
E ₂	Armit	D ₂		D ₂	HR	Hyd
E ₃	Ravi	D ₁		D ₃	Finance	Pune
E ₄	Nitin	-				

Select emp_no, ename, d_name, loc from
 emp left Outer Join dept ON
 (emp.dept_no. = dept.dept_no.)

Ans:

Emp_no.	E-name	d-name	Loc
E ₁	Vasun	IT	Delhi
E ₂	Armit	HR	Hyd
E ₃	Ravi	IT	Delhi
E ₄	Nitin	-	

- * IN NATURAL JOIN and EQUIJOIN, tuples without a matching (or related) tuples are eliminated from the join result.
- * Tuples with a null in the join attributes are also eliminated.

NATURAL JOIN can be INNER, Left Outer, RightOuter
The default is innerJoin.

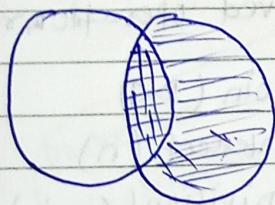
Date: / / Page no: _____

Right Outer Join.

It gives the matching rows and the rows which are in Right table but not in left table.

EMP			DEPT.		
EMP_no.	E_name	Dept No.	Dept No.	D_Name	Location
E ₁	Varun	D ₁	D ₁	IT	Delhi
E ₂	Amrit	D ₂	D ₂	HR	Hyd.
E ₃	Ravi	D ₃	D ₃	Finance	Pune
			D ₄	Testing	Noida

Select emp_no., e-name, d-name, loc from
emp Right Outer Join dept
on (emp.dept_no. = dept.dept_no.)



Ans.

EMP_No	E_Name	D_name	LOC.
E ₁	Varun	D ₁	Delhi
E ₂	Amrit	D ₂	Hyd.
E ₃	Ravi	D ₃	Pune
-	-	D ₄	Noida

FULL OUTER JOIN = LOJ U ROJ.

Relational Algebra (1970)

→ Procedural Query language

OR

→ Formal Query language

- ① What to do
- ② How to do

~~Operations~~

Operators

Basic operations

- Projection (Π)
- Selection (σ)
- Cross Product (\times)
- Union (U)
- Rename (ρ)
- Set Difference (-)

Derived operations

- Join (\bowtie)
- Intersect (\cap)
- Division ($/, \div$)

Projection (Π)

e.g. $\Pi_{\text{Name}}(\text{Student})$ 

RollNo.	Name	Age
1	A	20
2	B	21
3	A	19

Query: Retrieve the roll no. from table (student)

Aggregate function: count just counts the number of rows, who removing duplicates

Selection (σ) σ

Date: _____ / _____ / _____ Page no. _____

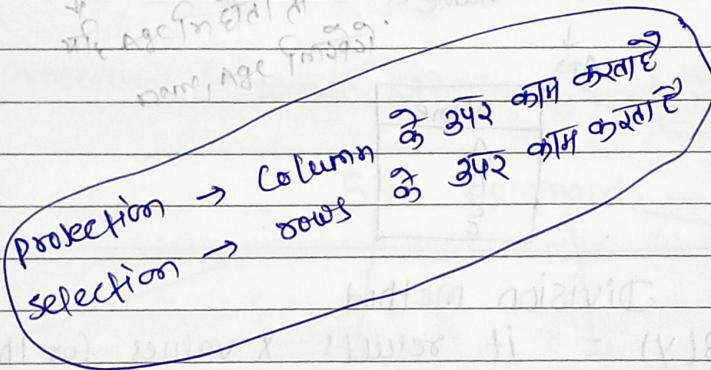
Query: Retrieve the name of Student whose Rollno = '2'

$\pi_{\text{Rollno} = '2'} (\text{Student})$

→ AFS Rollno Name age
2 B 21

$\pi_{\text{name Rollno} = '2'} (\text{Student})$

→ B.



π छोसा बाए
में प्रोटो।.
प्रदेश सेवा
करते।

Set Difference (-)

$$A - B = A \text{ but not } B = A \cap B'$$

$$A - B \neq B - A \leftarrow \text{commutative of } \cap$$

- (1) No. of Columns must be Same in no.
- (2) Domain of every Column must be Same.

Rollno	Name
1	A
2	B
3	C

Student

EmpNo.	Name
7	E
1	A

Employee

Rollno.	Name
2	B
3	C

(Student - Employee)

Ques. Name of person who is a Student but not Employee

$\pi_{\text{name}}(\text{Student}) - \pi_{\text{name}}(\text{Employee}) \rightarrow$

Union

- 1) No. of columns must be same in no.
- 2) Domain of every column

RollNo.	Name
1	A
2	B
3	C

(Student)

EmpNo.	Name
7	E
1	A

(Employee)

RollNo.	Name
1	A
2	B
3	C
7	I.

(Student) U (Employee)

$\Pi_{name}(\text{Student}) \cup \Pi_{name}(\text{Employee})$

Ans

Name
A
B
C
E

Division method

$A(x, y) / B(y) =$ it results x values for that there should be tuple $\langle x, y \rangle$ for every y value of relation B.

Sid	Cid
S ₁	C ₁
S ₂	C ₁
S ₁	C ₂
S ₃	C ₂

Enrolled (E)

Cid
C ₁
C ₂

Course

Query: Retrieve Sid of students who enrolled in every course

every
all

using division method
use E(DT).

$$\Pi_{Sid}(\text{Enrolled}) - (\Pi_{Sid}(((\Pi_{Sid}(\text{Enrolled})) \times \Pi_{Cid}(\text{course})) - (\text{Enrolled})))$$

S₁ S₂
S₂ S₃
S₃
(S₁ Ans)

S₁ C₁
S₁ C₂
S₂ C₁
S₂ C₂
S₃ C₁
S₃ C₂

S₁ C₁
S₂ C₁
S₁ C₂
S₃ C₂

ALTER TABLE tablename RENAME Column columnnameletternew

Structure Query language (SQL).

- SQL is domain-specific language. → what to do.
- SQL is a declarative language
- DDL, DML, DCL, TCL
- Keys and constraints. Exist/Not exist
- Operators (like, between, in, Not in), conditional
- Clauses (distinct, order by, group by, from having)
- aggregate functions → max, min, count, avg, sum, distinct. null की गई count करें।
- Joins and Nested Query. null नहीं होता (जोड़े दिए तो table में null दिया जाता है)
- PL SQL (Triggers, function, cursor, procedures). (प्रोग्राम)

Procedural Language SQL

Distinct (Sum (Salary))
different
Salary
of sum.

SQL Commands.

DDL

- Create
- Alter
- Drop
- Truncate
- Rename

Data manipulation language

- select
- insert
- update
- delete

Data control language

- Grant
- Revoke

Transaction control language

- Commit
- Rollback
- Savepoint

constraints

- primary key
- foreign key
- check → fixed domain
- unique
- default → default value
- Not Null

मुझे को क्या करना चाहिए पर विशेष वॉल्यूम
 इसके लिए क्या करना चाहिए पर विशेष वॉल्यूम

Alter Command

- Add Columns
- Remove Columns
- Modify datatype
- Modify datatype length
- Add constraints
- Remove constraints
- Rename column / table

ID	name
----	------

Student

ID	name	address

ALTER TABLE Student ADD address VARCHAR(30);

Difference b/w Alter and update

Alter

Update

(1)

DDL

(1) DML

(2)

works only on Structure

(2) works only on Data

Difference b/w Delete, Drop, Truncate

यह लॉग में अंदर है

अपने बाद delete करता है

↓
Rollback
(rollback)

(1)

DML

tuples will
delete

(slower)

Drop

DDL

Drop table Student

Whole schema
delete

DDL

Truncate Student

All rows will delete in
one time but not structure
(faster)

Truncate ← Roll back
(invalid))

आपस में Relation है

Relation in inner and outer query.

Date: / / Page no. / /

Correlated Subquery (Synchronized Query).

- It is a subquery that uses values from outer query.
- Top to down approach.

OB (IB)

e.g.

EMP

Eid	Name	Address
1	A	Delhi
2	B	Pune
3	A	Chd
4	B	Delhi
5	C	Pune
6	D	Mumbai
7	E	Hyd

DEPL

Did	Dname	Eid
D1	HR	1
D2	IT	2
D3	MRKT	3
D4	Testing	4

Q. Find all employees detail who work in a department.

Select * from EMP where

- exists (Select * from DEPT where
dept.Eid = EMP.Eid);

अगर यह if true आ गया तो true Return करो।

तो दोनों डेटासेट के value अन्दर वाले शारे value के लिए
एक-एक करके दर्ज करो।

Ans

Eid	Name	Address
1	A	Delhi
2	B	Pune
3	A	Chd
4	B	Delhi

Correlated Subquery

Nested Subquery

\rightarrow Bottom up \leftarrow query inner query

Select * from Emp where
e_id in (select e_id
from dept)

Start from
group of
value E

e_id	name
1	A
2	B
3	C
4	D
5	E

Dept

dept_no	name	e_id
D1	IT	1
D2	HR	2
D3	MKT	3

Ques) find detail of all employees
who worked in any department?

Joins

\rightarrow Top down approach

Select * from emp where
exists (select id from dept
where emp.e_id = dept.e_id);

Select attributes from
emp, dept
where emp.e_id =
dept.e_id;

Joins \rightarrow fast ETL ETL
compared to correlated
subquery

Join \rightarrow cross product + conditions
Join \rightarrow buffered join
Outer Buffer join

who worked in any department?

Question.

Date: / / Page no.:

EMP.

E_id	E_name	Dept.	Salary.
1	Ram	HR	10000
2	Amrit	MRKT	20000
3	Ravi	HR	30000
4	Nitin	MRKT	40000
5	Vasun	IT	50000

Ques Write a SQL Query to display maximum Salary from Emp table.

Select Max(Salary) from Emp;

Ques Write a SQL Query to display Employee name who is taking maximum salary

Select E_name from Emp where Salary =
(Select Max(Salary) from Emp);

Get inner query execute first

Ques Write a SQL query to display Second highest salary from Emp table?

Select Max(Salary) from Emp
where Salary < ? (select Max(Salary) from Emp);

↓ not equal to.

Ques) Write an SQL Query from the given table.

Date: / / Page no:

Ques) Write a SQL query to display Employee name who is taking second highest salary?

Select E-name from emp → in first row
where salary =
(\approx)

Select max(salary) from emp
where salary < (select max(salary)
from emp);

Ques). Write a query to display all the dept names along with no. of emps working in that?

Select dept, count(*) from emp
group by dept;

group by (dept);

... 3 से कम वर्ती कंवर
अग्रégate function के प्रयोग किये!

Ques) Write a query to display all the dept names where no. of emps are less than 2.

Select dept from emp group by dept
having: count(*) < 2;

having के साथ where का क्षेत्र
किसी where के table पर कैसे करते हैं?

1 = 1 true ✓
1 = 2 false ✗

1 = (2, 3, 4, 5) X in MySQL
1 = (1, 2, 3, 4, 5) X include for the first 4

Date: / / Page no: _____

Ques) Write a query to display all the employees name in the department where the no. of employees < 2 ?

Select E-name from Emp where dept IN
(Select dept from Emp group by dept
having count(*) < 2);

Ques) Write a query to display highest salary department wise and name of emp who is taking that salary?

Select E-name from Emp where Salary
IN
(Select max(Salary) from Emp
group by dept);

30000
40000
50000

In | Not In

Eid	E-name	Address	Eid	Pid	Pname	Location
1	Ravi	Chd	1	P1	IOT	Banglore
2	Vasun	Delhi	5	P2	BIG Data	Delhi
3	Nitin	Pune	3	P3	Retail	Mumbai
4	Robin	Banglore	4	P4	Android	Hyderabad
5	Army	Chd				

Q. find detail of Emp whose address is either Delhi or Chd or pune?

Select * from Emp where Address

In ('Delhi', 'Chd', 'Pune');

NOT IN निवारणी हो सकते हैं।
not included होता है।

Any, All जैसे नहीं
query का Nested query है।

Q. find the name of Emp who are working on a project?

Select Ename from Emp where Eid

In (Select Distinct(Eid) from Project);

Exists | Not Exists → Correlated में डीट्री में काम आता है |
 Date: false / false

Q. Find the detail of Emp who is working on at least one project ?

Select * from Emp where Eid

Exists (Select Eid from project where
 $(\text{Emp} \cdot \text{Eid} \neq \text{Project} \cdot \text{Eid})$);

यही output का कोटि लिखा था अन्दर के query में निकलते हैं तो
 यह यह बताता है कि correlated Nested query है।

Find N^{th} Highest Salary Using SQL

Select id, salary from Emp e1

where $N-1 = (\text{Select count (distinct salary)}$
 from Emp e2

where $e2.\text{Salary} > e1.\text{Salary}$)

Emp e1

Emp e2

ID	Salary
1	10000
2	20000
3	20000
4	30000
5	40000
6	50000

ID	Salary
1	10000
2	20000
3	20000
4	30000
5	40000
6	50000

Ques). You need to display last name of employees who have 'A' as second character in their names. Which SQL statement display the required result?

- A) select last_name from emp where last_name like '_A%'
- B) select last_name from emp where last_name = '*A%'. X
- C) Select last_name from emp where last_name like '%.A%'
equal to it \rightarrow
 \downarrow
अंतीम प्रथम
पर character
पर आवृत्ति.
- D) select last_name from emp where last_name like '%A.'

Ques). A command to remove relation from SQL database

- A) Delete table < table name >
- B) Drop table < table name >
- C) Erase table < table name >
- D) Alter table < table name >

Ques). In the following, Schema R is R(a, b)

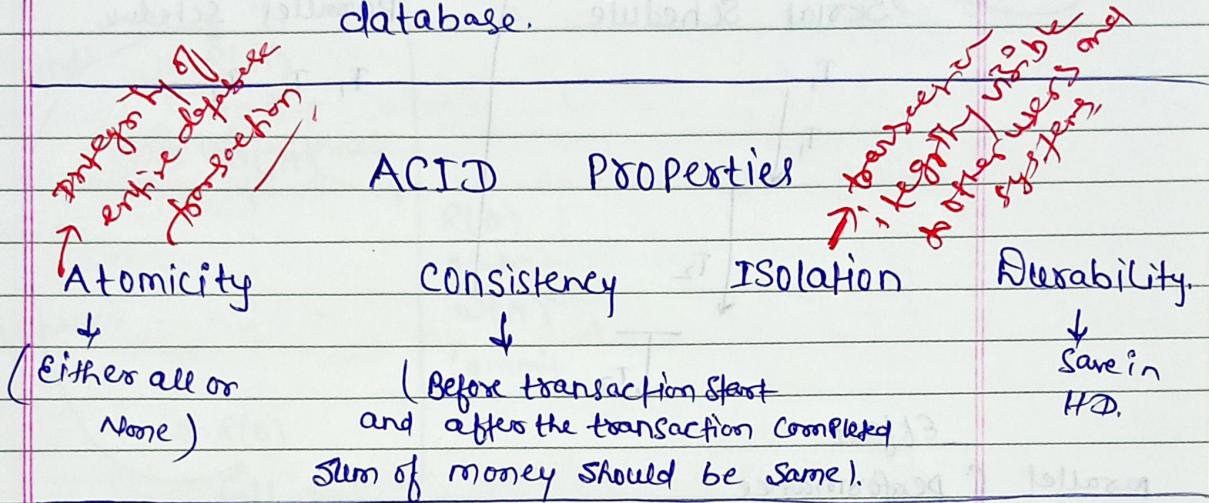
- Q1. select * from R
- Q2. (select * from R) intersect (select * from R)
- Q3. select distinct * from R.

- A) Q₁, Q₂, Q₃ produce same result.
- B) only Q₁, Q₂ produce same result.
- C) only Q₂, Q₃ produce same result.
- D) Q₁, Q₂, Q₃ produce different result.

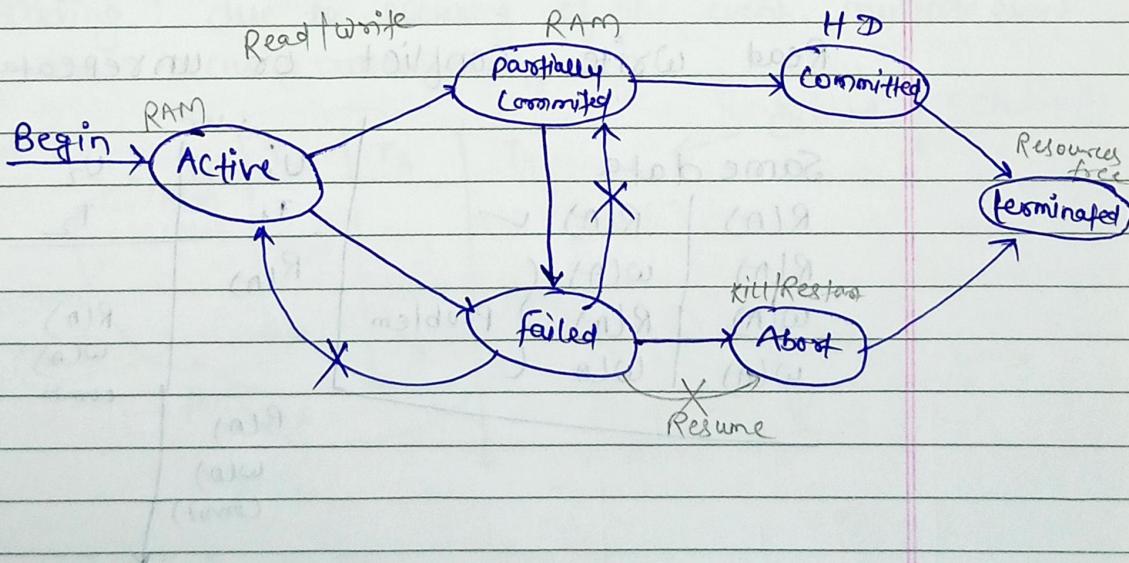
TRANSACTION

Transaction : It is a set of operations used to perform a logical unit of work.

→ A transaction generally represent change in database.

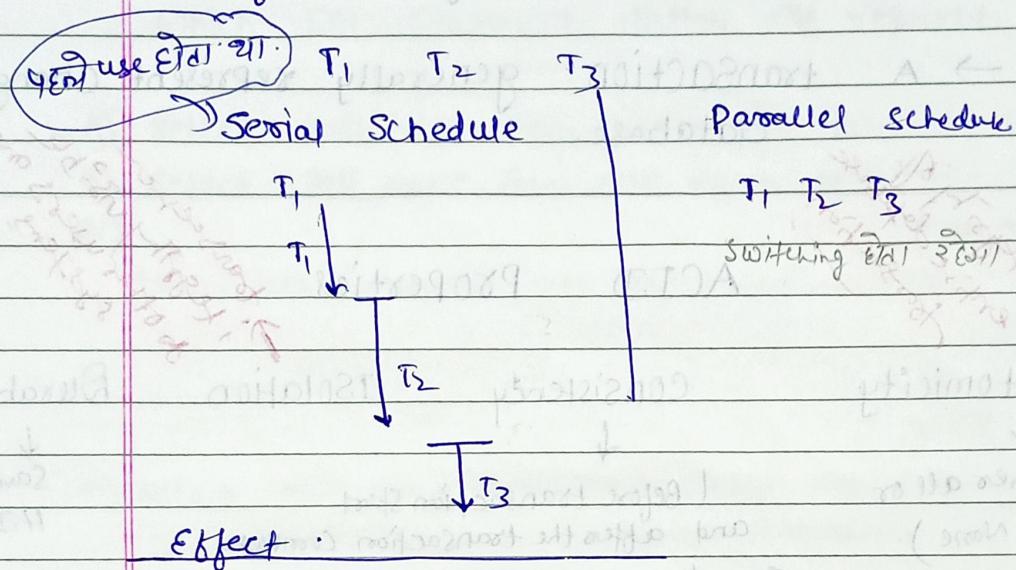


TRANSACTION STATES.



Schedule:

→ It is chronological execution sequence of multiple transactions.

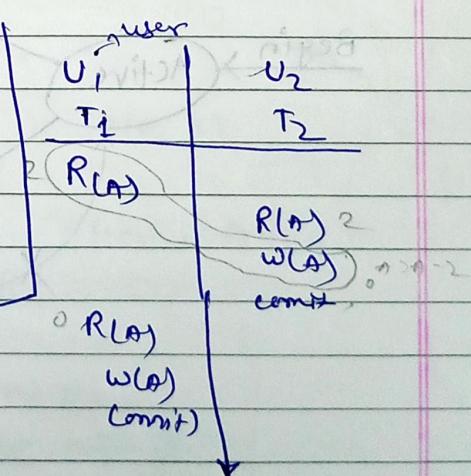


Effect

parallel schedule { performance
if high { throughput: No. of transactions executed per unit time

Read Write Conflict or Unrepeatable Read.

Same data	
R(A)	R(A) ✓
R(A)	W(A) {
W(A)	R(A) { Problem
W(A)	W(A)

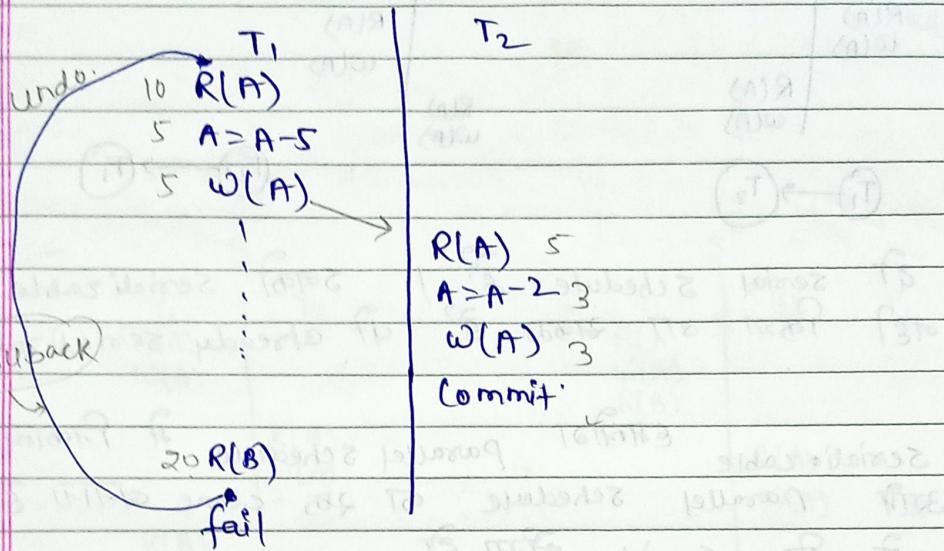


Recoverability

④ Recoverable Schedule
③ Strict Recoverable Schedule

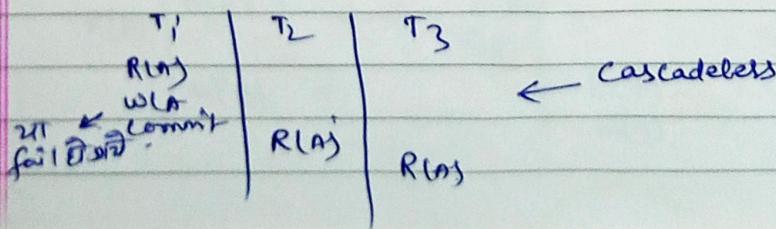
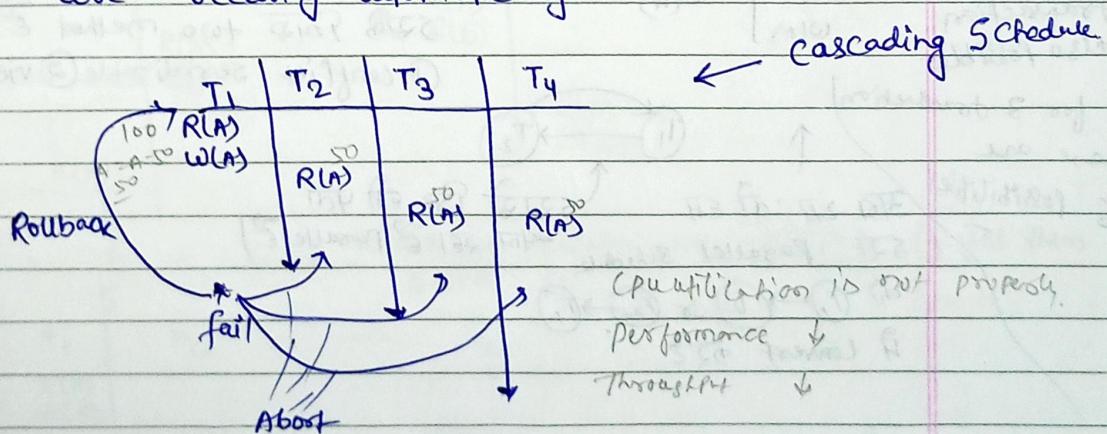
Date: / / Page no: _____

① IRRECOVERABLE SCHEDULE.



② Cascading Schedule vs ③ Cascadeless Schedule

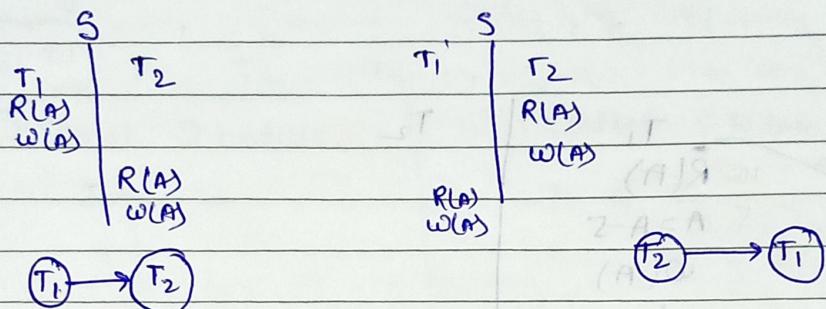
Cascading: due to occurring of one event, multiple events are occurring automatically.



Serializability.

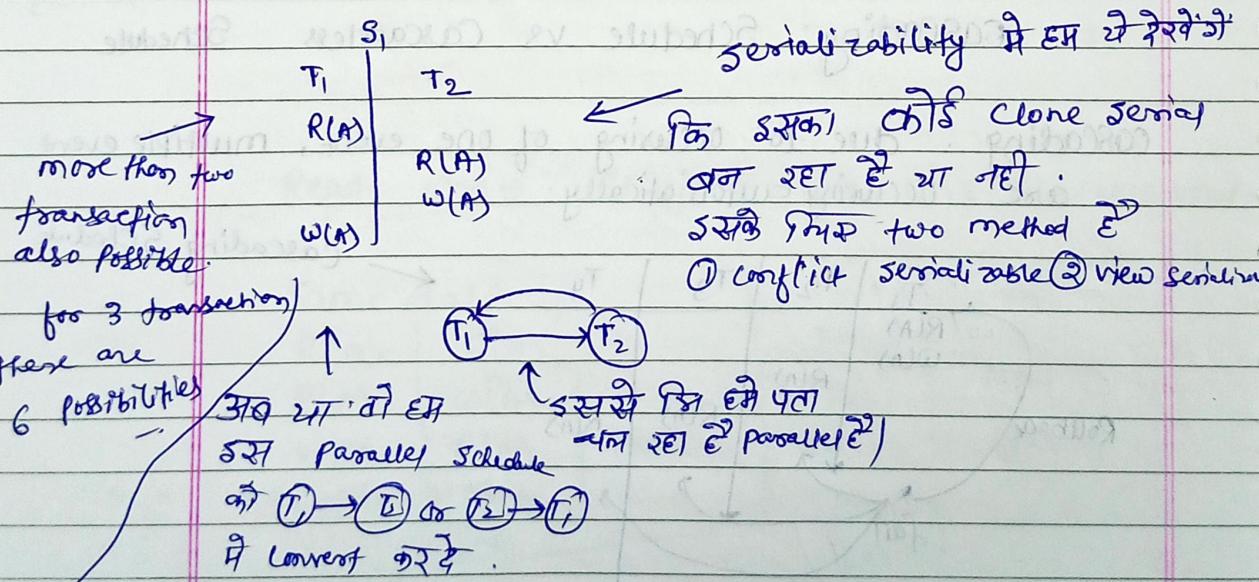
Date: / / Page no:

→ An ability of a Schedule to become Serializable.



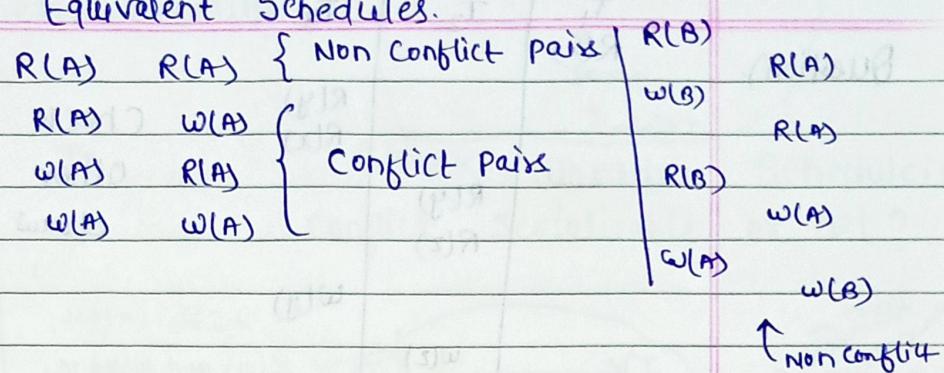
दो serial schedule हैं। उनकी serializable
नहीं किया जा सकता है वो already serializable हैं।

serializable schedule में प्रक्रिया हैं।
असमान parallel schedule का कोई clone नहीं है।
जो कि serial हो सकता है।



Conflict Equivalent Schedules.

Date: / / Page no. _____



S		S'	
T_1	T_2	T_1	T_2
$R(A)$		$R(A)$	
$W(A)$		$W(A)$	
	$R(A)$		$R(A)$
	$W(A)$		$W(A)$
		$R(B)$	

Q. Find out $S \equiv S'$

Ans

S		(S')	
T_1	T_2	T_1	T_2
$R(A)$		$R(A)$	
$W(A)$		$W(A)$	
	$R(A)$		$R(A)$
	$W(A)$		$W(A)$
		$R(B)$	

adjacent

Check out Adjacent Non conflict Pair if yes then swipe it out.

S		S'	
T_1	T_2	T_1	T_2
$R(B)$		$R(A)$	
$W(A)$		$W(A)$	
	$R(A)$		$R(A)$
	$W(A)$		$W(A)$

→ proved
 $S \equiv S'$

यदि यहीं एक ही रूप है
 तो यहीं Serializable

रुपये में कूद लिए कर सकता है conflict बन देता है
जब parallel होने पर भी है।

Date: / / Page no.:

Question).

T_1	T_2	T_3
$R(x)$		
	$R(y)$	
	$R(z)$	
		$R(y)$ $R(x)$
	$w(z)$	
		$w(y) \rightarrow$ serializable $R(y)$ and $w(y)$ - दोनों serializable
$R(z)$		
$w(x)$		
$w(z)$		

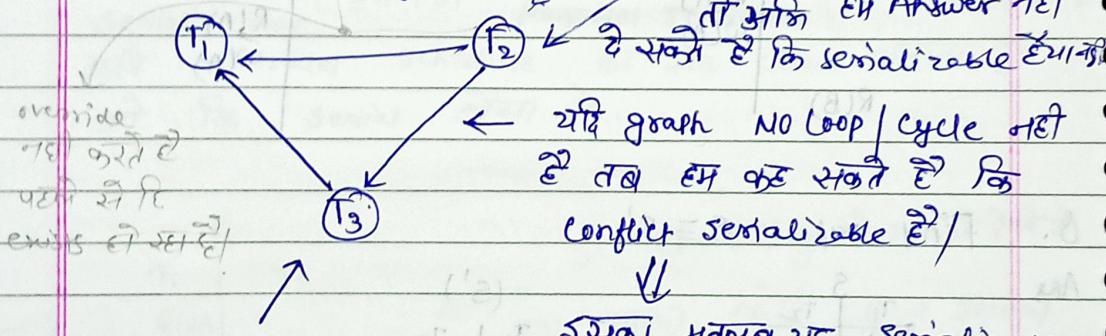
Check conflict pairs in other transactions and draw edge.

Please use view
graphable

if look then
most serializable
conflict

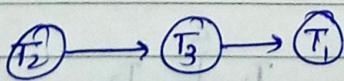
Soln.

precedence graph



to अकिनी ET Answer TET
दो सक्षम हो कि serializable होना चाहिए
यदि graph NO loop/cycle नहीं
हो तब हम कह सकते हैं कि
Conflict serializable है।

To find out serializable
firstly find out the
vertex whose indegree
is zero and draw it and
Remove and then check indegree zero and Repeat the above
process



Ans -

View Serializability

Date: / / CS Page no.: _____

No loop | No cycle

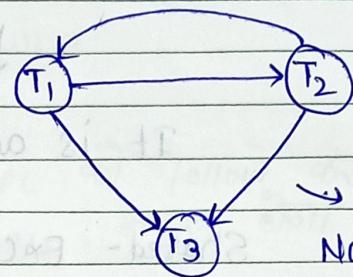
↓

serial

↓ consistent

Check whether Schedule is conflict serializable or not?

S		T ₃
T ₁	T ₂	
R(A)		
W(A)	W(A)	W(A)



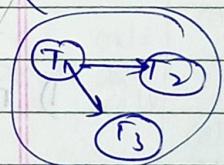
loop \Rightarrow not serializable

loop then
Non conflict
Serializable

now check using view.

change slightly in S.

S			T ₁	T ₂	T ₃
T ₁	T ₂	T ₃	R(A)	W(A)	W(A)
R(A)					
W(A)	W(A)	W(A)			



finally both get output zero.

So if it's not a conflict serializable. It is a view serializable. So it's a serializable.

Shared Exclusive Locking Protocol.

Concurrency Control protocol:

Basic aim is to achieve serializability and recoverability.



It is achieved by locking protocol.

Shared - Exclusive locking.

→ Shared LOCK(S) ⇒ If transaction locked data item in Shared mode then allowed to read only.

→ Exclusive Lock(X) ⇒ If transaction locked data item in Exclusive mode then allowed to Read and write both.

Problem in S/X locking

- 1) may not sufficient to produce only serializable schedule.
- 2) may not free from Irrecoverability.
- 3) may not free from dead lock.
- 4) may not free from starvation.

→ Request

	S	X
S	Yes	No
X	No	No

compatibility table.

Shared & Shared

Resource की संख्या

infinite time

don't wait

deadlock

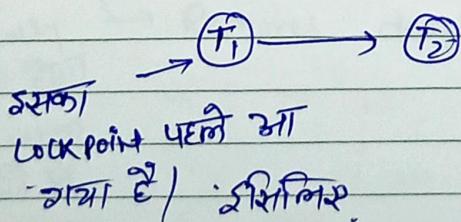
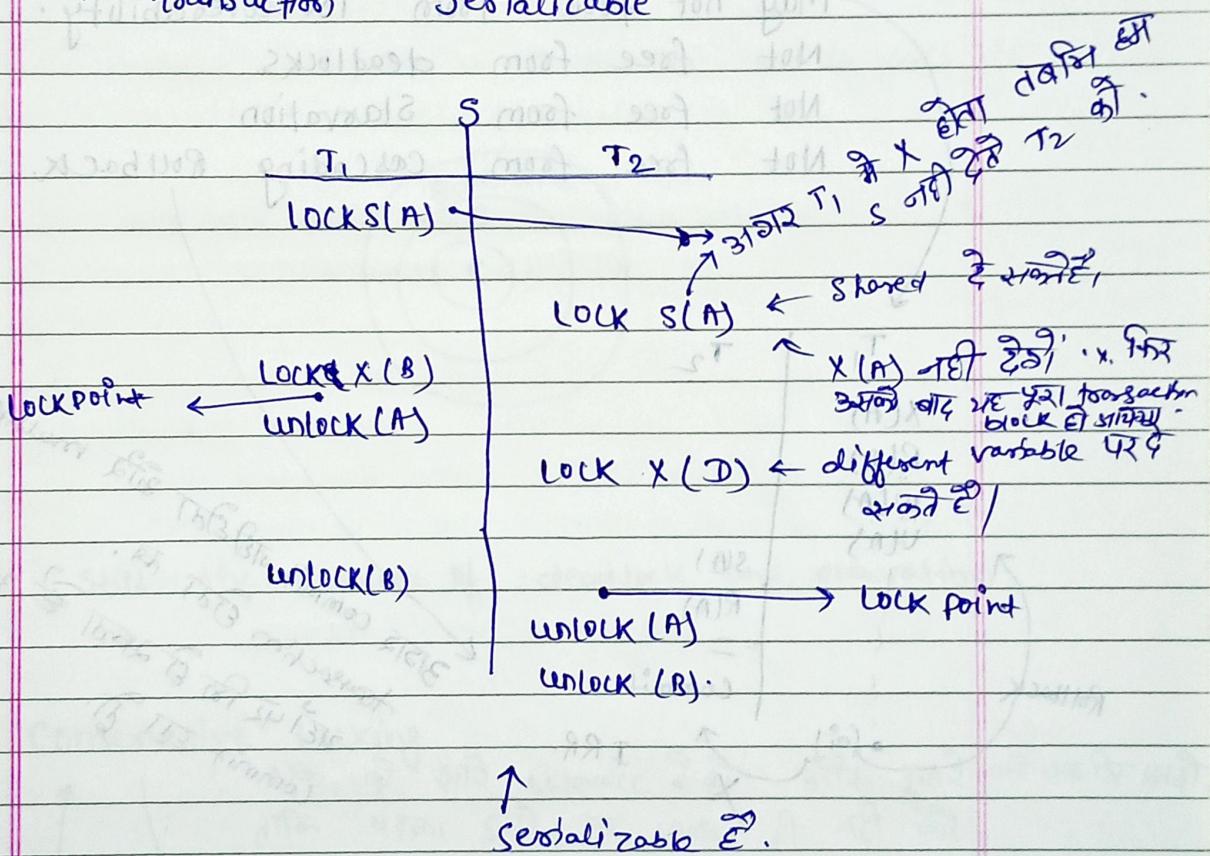
modification in S/X

parallel growing phase start की स्थिती देख
on compatibility.

Phase Locking (2PL) Protocol in Transaction Control.

- Growing Phase: locks are acquired and no locks are released.
- Shrinking Phase: locks are released and no locks are acquired.

जो प्रति transaction 2-PL की follow कर रहा है उसके
में transaction serializable अवश्य होगा।



Shared के लिए Shared के सभी हैं।

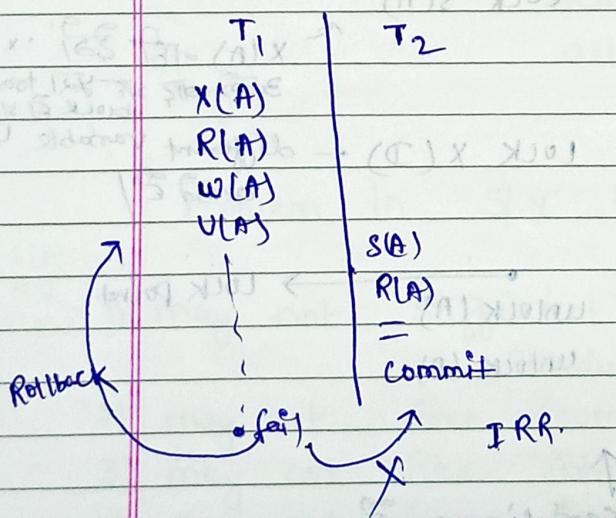
Date: / / Page no:

2 PL (2 phase locking)

Advantages: Always ensures Serializability.

Drawbacks:

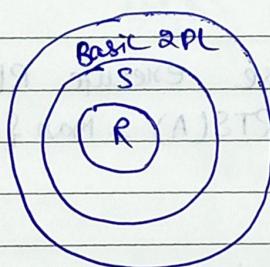
- May not free from recoverability.
- Not free from deadlocks.
- Not free from Starvation.
- Not free from cascading Rollback.



It always produce cascades of recoverable
strict

Strict 2PL: It should satisfy the basic 2PL and all exclusive locks should hold until commit / Abort.

Rigorous 2PL: It should satisfy the basic 2PL and all shared, exclusive locks should hold until commit / Abort.



Still the problem of deadlock and starvation.

• Conservative Locking

: कैसे को भारी Resource हैं दी। और इससे को एक जिसकी
पर्फॉर्मेंस इससे वाह नहीं होती क्योंकि

practically
poor efficiency

anticipation
जीवन से

→ Removes deadlock Problem.

Assumption E |
→ यहां आजी वे उनी
complete PFT होता

Date: / / Page no:

Timestamp Ordering Protocol.

- Unique value assign to every transaction.
- Tells the order (when they enters into System).
- $\text{Read-TS (RTS)} = \text{last (latest) transaction no. which perform Read successfully.}$
- $\text{Write-TS (WTS)} = \text{last (latest) transaction no. which performed Write successfully.}$

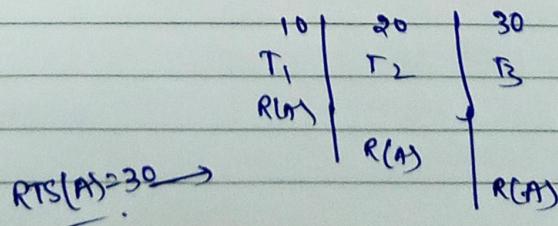
Rules:

- 1) Transaction T_i issues a $\text{Read}(A)$ operation
 - if $\text{WTS}(A) > \text{TS}(T_i)$, Rollback T_i
 - Otherwise execute $R(A)$ operation
Set $\text{RTS}(A) = \max \{ \text{RTS}(A), \text{TS}(T_i) \}$
- 2) Transaction T_i issues $\text{Write}(A)$ operation
 - if $\text{RTS}(A) > \text{TS}(T_i)$ then Rollback T_i
 - if $\text{WTS}(A) > \text{TS}(T_i)$ then Rollback T_i
 - otherwise execute $\text{write}(A)$ operation
Set $\text{WTS}(A) = \text{TS}(T_i)$

e.g.

10:00 10:10 10:15 TS(T_i)
 T_1 T_2 T_3
order → 100 200 300

Older Younger Youngest



old | young

100 | 200

T₁ | T₂

R(A)

w(A)

T₁ | T₂

w(A) | (R(A))

w(A)

w(A)

\rightarrow $R(A), w(A)$ ने T_1, T_2 को priority दे रखे हैं।

\rightarrow $w(A)$ को allow है।

$w(A) \rightarrow$ $w(A)$ को allow है।

(339)

(339)

(339)

case 1

100

T₁

R(A)

$w(A)$
! Commit
Rollback

case 2:

T₁

R(A)

Commit

fail

T₂

w(A)

rollback

commit

100

T₂

w(A)

! Commit

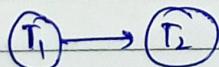
fail

rollback

commit

fail

does not allow.



\rightarrow assumption है।