Transactions - 2

↳ commit & Rollback

Isolation levels

→ read uncomitted

→ read comitted

→ repeatable read

→ serializable

deadlocks

Commit and Rollback

Trans^n
↓
set of db calls

single query } → Transaction
↳ runs → disk
commit;

(autocommit)
↓
false

commit;

permanent {
———————
———————
———————
———————
} → rollback → Atomicity

commit;
////////
rollback;

Isolation levels

✓ set session transaction isolation level _ ___:

Read uncommitted

T1	T2
① select * from film where film.id = 12;	
title = "xyz"	
	② select * from film where film.id = 12.
	title = "xyz"
③ update title = "abc" film.id = 12	④
changes are not committed yet.	"abc"

Dirty Read :-
Whenever a tran^n end up reading or using a data which was not committed.

Read committed

T1

T2

① film_id = 12

title = "xyz"

② selet film_id = 12

title = "xyz"

③ update
title = "abc"
film_id = 12

selet film_id = 12
↓
xyz
abc

④

commit;

postgres — default
mysql → repetable-read

Break: 5 min

10:37 pm

txn1   **txn2**

Students

--- psp... email-sent   send
     82 — 79   true   emails
     75   psp < 80
     56
     69
     89

                                              txn 2

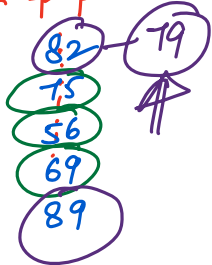   Stud { HW

list<students> = select * from students
                  where psp < 80;

4 stu

                email service (student)

update   students
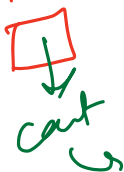set   email sent = true.
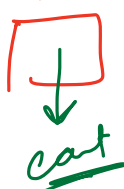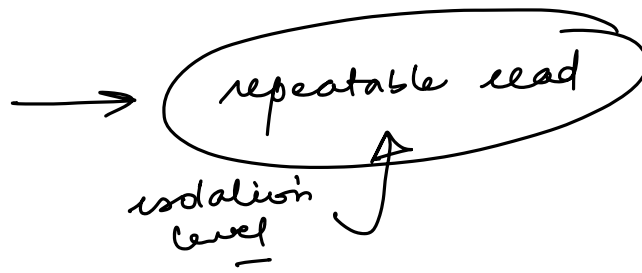when   psp < 80;

                                            non-repetable

select * fr.

5 students   → update

P1   P2   product

P3   product ↑↓↑↓
                    cart
cart

cart

$\longrightarrow$ (repeatable read)

isolation
level

snapshot taken

film.id = 1
$\downarrow$
title: film1 $\longrightarrow$ 3film

txn1 — report about the items

txn2

→ select price, Quatly from
item;
{ 50   4
  80   3

→ 2 more items 2 got sold

select sum(price * Qu) from
item;

insert a new row

→ update 440

items   Pric   Quat
item 1 $ 50   4
item 2 $ 80   3 → 5

item 3 $ 100   2

phantom read

TA

pool .

HR

TA TA TA TA

TX1

Tx2

TA ?

Mohir

Alang

HR1

HR4

RU

uncommitted — DR

RC

non — reo

RR

repeat

S
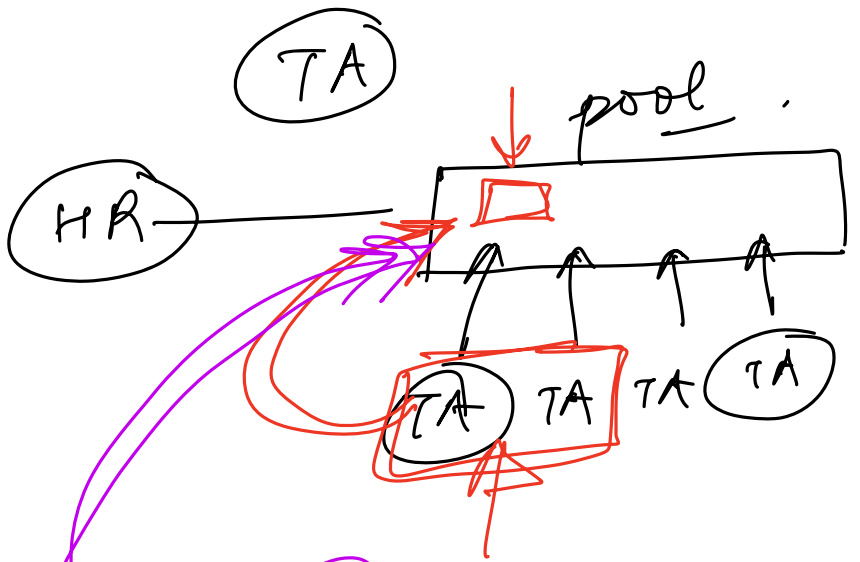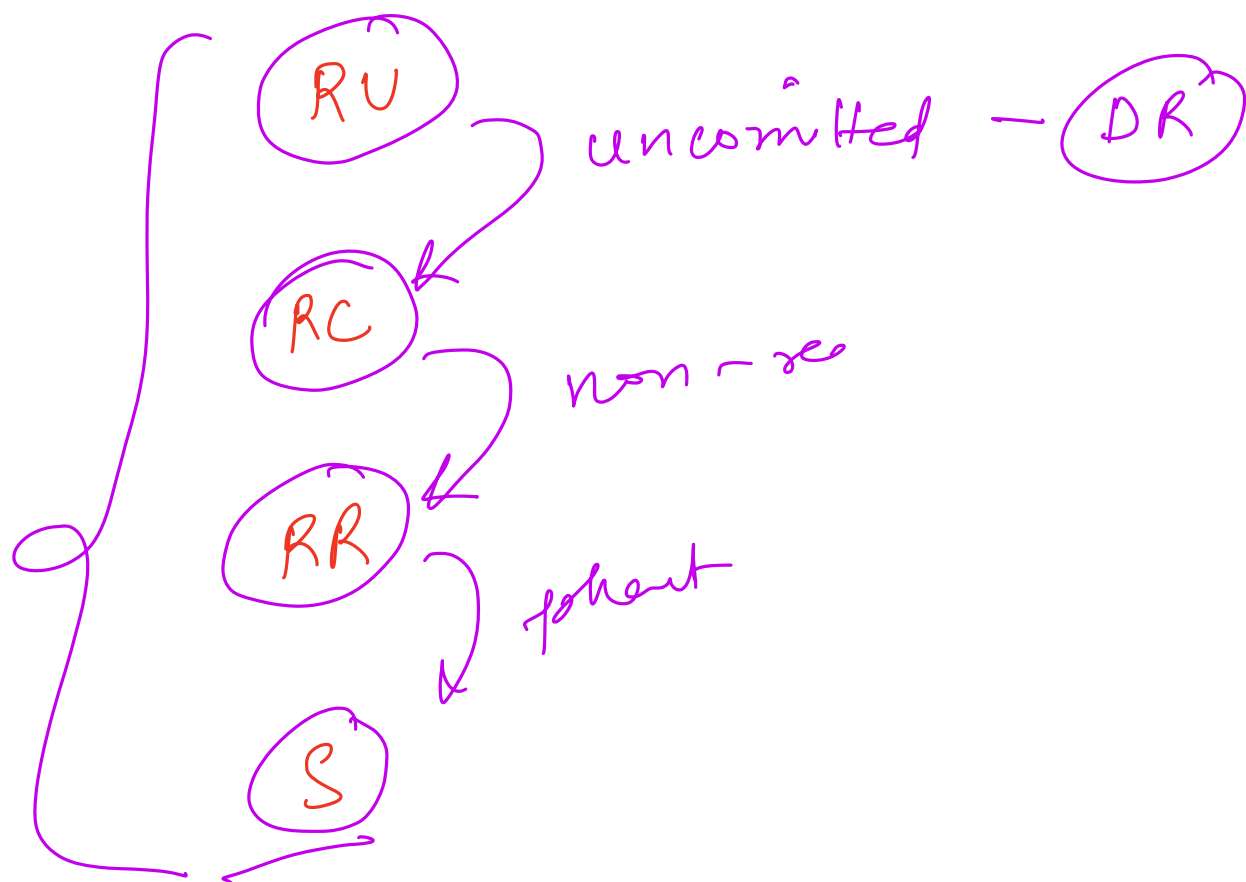
autocommit = false true

{
———————
———————
———————
———————

Commit

Start transaction;
———————
———————
———————
———————
———————

Commit