Normal forms cont. & intro to transactions

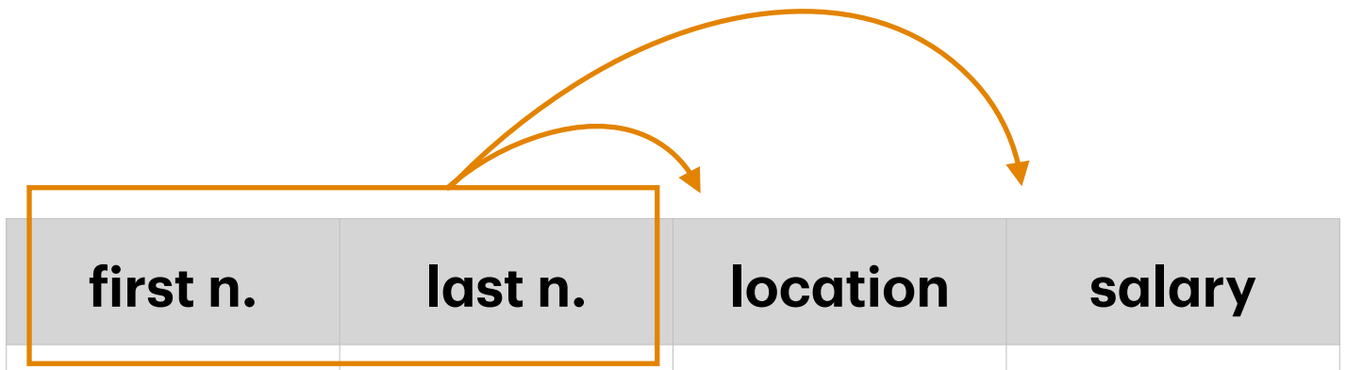
Remy Wang 04/24/2025

superkey $X \to \{A_1, \ldots, A_5\}$

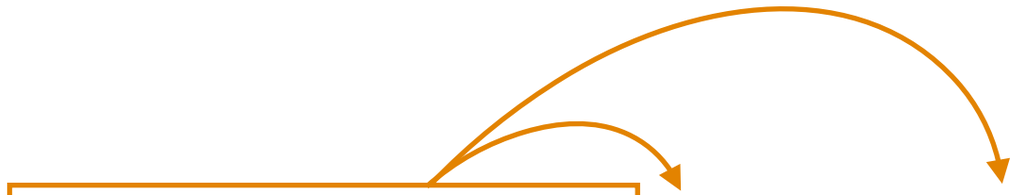
A1	A2	A3	A4	A5

PK

name	location	salary
remy	LA	$30
vincent	LA	$20



first n.	last n.	location	salary
remy	w	LA	$30
dan	s	seattle	$50
dan	o	zurich	$50



first n.	last n.	location	salary	course
remy	w	LA	$30	143
remy	w	LA	$30	240
remy	w	LA	$30	249
dan	s	seattle	$50	344
dan	s	seattle	$50	444
dan	o	zurich	$50	101
dan	o	zurich	$50	113

name	job	location	salary	tax %
remy	prof	LA	$30	20
dan	prof	seattle	$50	15
vincent	TA	LA	$20	10

$$\text{job} \rightarrow \text{salary}$$

$$\text{name} \rightarrow \text{location} \quad \Longrightarrow \quad \text{name, job} \rightarrow \text{tax \%}$$

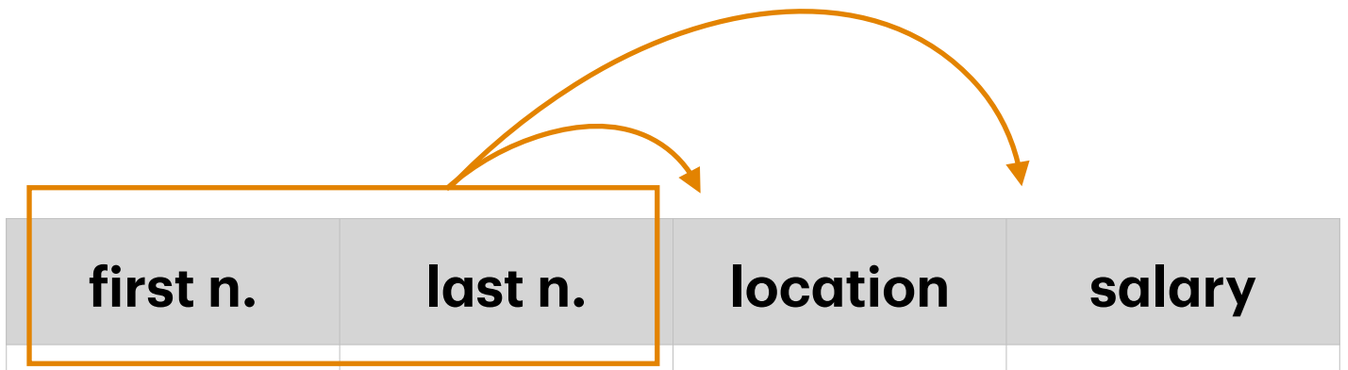
$$\text{location, salary} \rightarrow \text{tax \%}$$

BNCF

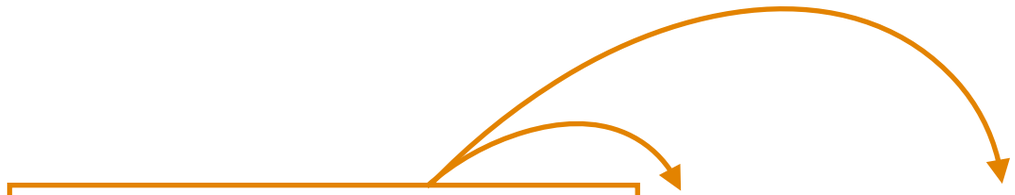
$$X \to Y : \begin{cases} Y \subseteq X \text{ (trivial FD)} \\ X \text{ is a superkey} \end{cases}$$

PK

name	location	salary
remy	LA	$30
vincent	LA	$20



first n.	last n.	location	salary
remy	w	LA	$30
dan	s	seattle	$50
dan	o	zurich	$50



first n.	last n.	location	salary	course
remy	w	LA	$30	143
remy	w	LA	$30	240
remy	w	LA	$30	249
dan	s	seattle	$50	344
dan	s	seattle	$50	444
dan	o	zurich	$50	101
dan	o	zurich	$50	113

name	job	location	salary	tax %
remy	prof	LA	$30	20
dan	prof	seattle	$50	15
vincent	TA	LA	$20	10

$$\text{job} \rightarrow \text{salary}$$

$$\text{name} \rightarrow \text{location} \quad \Longrightarrow \quad \text{name, job} \rightarrow \text{tax \%}$$

$$\text{location, salary} \rightarrow \text{tax \%}$$

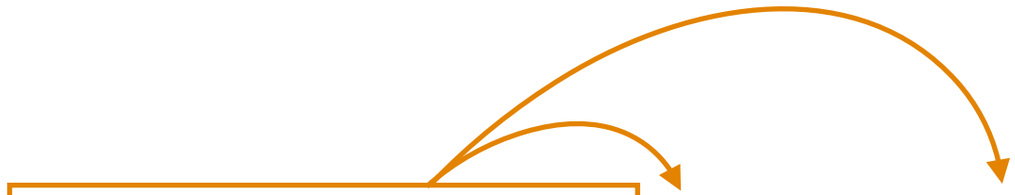
Decomposition

Factor out violating FDs

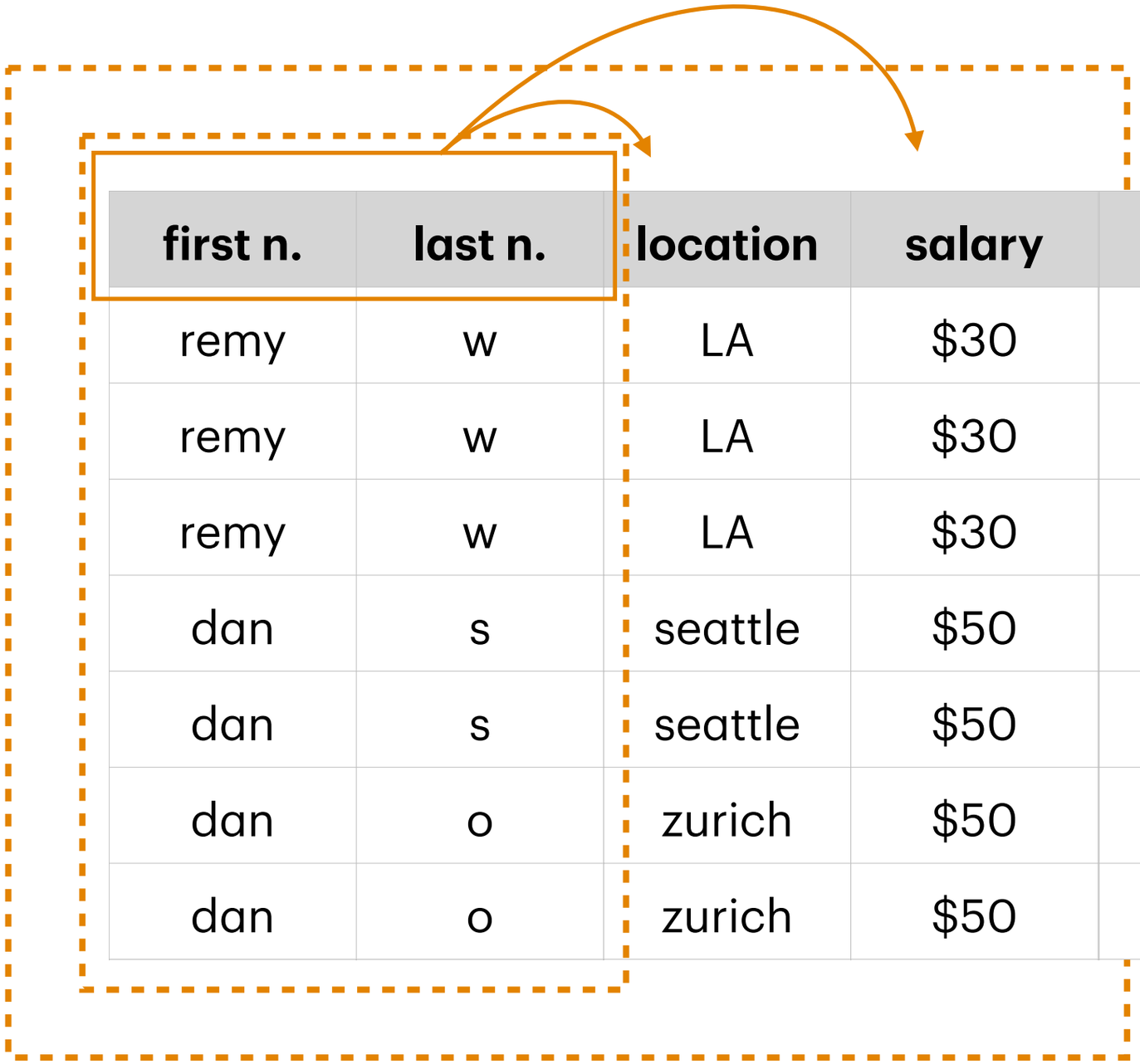
$$X \to Y : \begin{cases} Y \subseteq X \text{ (trivial FD)} \\ X \text{ is a superkey} \end{cases}$$

Make new table over $X \cup Y$

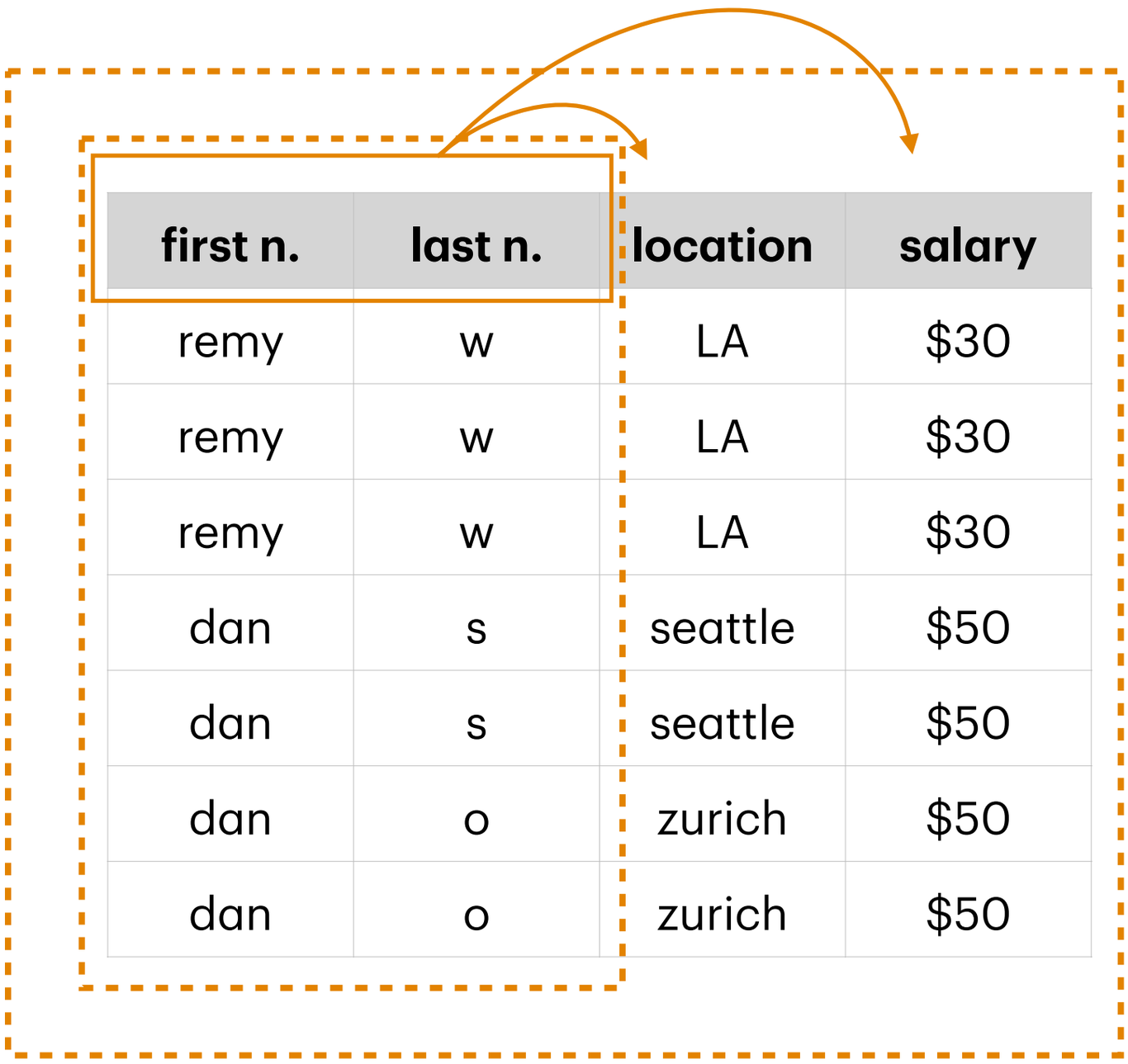
Drop Y from old table (keep X)



first n.	last n.	location	salary	course
remy	w	LA	$30	143
remy	w	LA	$30	240
remy	w	LA	$30	249
dan	s	seattle	$50	344
dan	s	seattle	$50	444
dan	o	zurich	$50	101
dan	o	zurich	$50	113



first n.	last n.	location	salary	course
remy	w	LA	$30	143
remy	w	LA	$30	240
remy	w	LA	$30	249
dan	s	seattle	$50	344
dan	s	seattle	$50	444
dan	o	zurich	$50	101
dan	o	zurich	$50	113



first n.	last n.	location	salary
remy	w	LA	$30
remy	w	LA	$30
remy	w	LA	$30
dan	s	seattle	$50
dan	s	seattle	$50
dan	o	zurich	$50
dan	o	zurich	$50

first n.	last n.	course
remy	w	143
remy	w	240
remy	w	249
dan	s	344
dan	s	444
dan	o	101
dan	o	113

$$\text{job} \rightarrow \text{salary} \quad \text{name} \rightarrow \text{location} \quad \text{location, salary} \rightarrow \text{tax \%}$$

name	job	location	salary	tax %
remy	prof	LA	$30	20
dan	prof	seattle	$50	15
vincent	TA	LA	$20	10

3rd Normal Form

$$X \to Y : \begin{cases} Y \subseteq X \text{ (trivial FD)} \\ X \text{ is a superkey} \\ \forall y \in Y : y \in \text{key} \end{cases}$$

3rd Normal Form

$$X \to Y : \begin{cases} Y \subseteq X \text{ (trivial FD)} \\ X \text{ is a superkey} \\ \forall y \in Y : y \in \text{key} \end{cases}$$

never loses FDs

decomposition in P time

3rd Normal Form

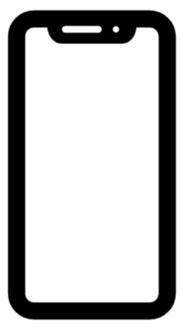
$$X \to Y : \begin{cases} Y \subseteq X \text{ (trivial FD)} \\ X \text{ is a superkey} \\ \forall y \in Y : y \in \text{key} \end{cases}$$

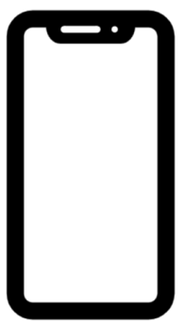
never loses FDs

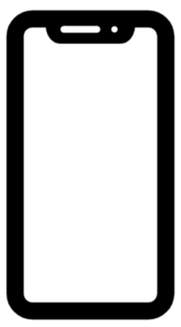
decomposition in P time

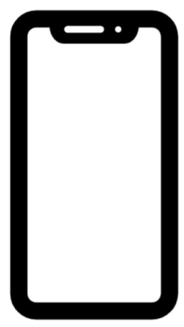
Transactions: it's a wild world out there


python™









exercise 1: free points for all!

atomicity

a TX either completes

or leaves no trace

exercise 2: playing favorites!

consistency

a TX should leave the DB

in a consistent state

exercise 3: armageddon!

isolation

multiple concurrent TX

should not interfere

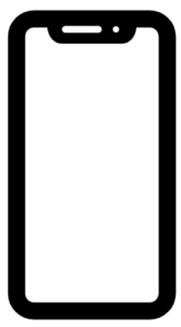
exercise 4: blackout!

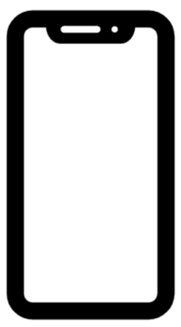
durability

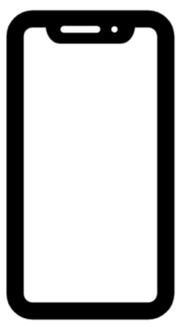
completed TX are forever

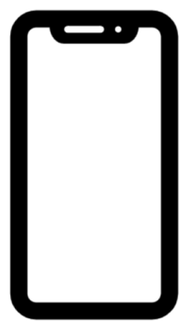
SQLite demo


python™









```
import sqlite3 as sql

con = sql.connect("bank.db")
cur = con.cursor()

res = cur.execute("SELECT * FROM acc")
```

transaction

a set of read & write ops

T1
READ(A, t)
t := t+100
WRITE(A, t)
READ(B, t)
t := t+100
WRITE(B,t)

A,B are elements in the DB

t is a local variable in the app

T1
READ(A, t)
t := t+100
WRITE(A, t)
READ(B, t)
t := t+100
WRITE(B,t)

A,B are elements in the DB

t is a local variable in the app

T2
READ(A, s)
s := s*2
WRITE(A,s)
READ(B,s)
s := s*2
WRITE(B,s)

transaction

a set of read & write ops

A either execute all, or nothing

C

I 2 transactions do not interfere

D

schedule

an interleaving & read/writes

from different TXs

time

T1	T2
READ(A, t)	
	READ(A, s)
	s := s*2
t := t+100	
WRITE(A, t)	
	WRITE(A,s)
	READ(B,s)
	s := s*2
READ(B, t)	
	WRITE(B,s)
t := t+100	
WRITE(B,t)	

serial schedule

"one at a time!"

A = 2
B = 2

time

T1	T2
READ(A, t)	
t := t+100	
WRITE(A, t)	
READ(B, t)	
t := t+100	
WRITE(B,t)	
	READ(A, s)
	s := s*2
	WRITE(A,s)
	READ(B,s)
	s := s*2
	WRITE(B,s)

time

T1	T2
A = 2 B = 2	**READ**(A, s)
	s := s*2
	WRITE(A,s)
	READ(B,s)
A = 4 B = 4	s := s*2
	WRITE(B,s)
	READ(A, t)
	t := t+100
	WRITE(A, t)
	READ(B, t)
	t := t+100
A = 104 B = 104	**WRITE**(B,t)

serializable schedule

equivalent to "one at a time!"

T1	T2	
READ(A, t)		A = 2 B = 2
t := t+100		
WRITE(A, t)		
	READ(A, s)	A = 102 B = 2
	s := s*2	
	WRITE(A,s)	A = 204 B = 2
READ(B, t)		
t := t+100		
WRITE(B,t)		
	READ(B,s)	A = 204 B = 102
	s := s*2	
	WRITE(B,s)	A = 204 B = 204

serial → isolation

serializable = serial

serializable → isolation