

If an application tuple is inserted for a student with $GPA > 3.9$ to NCSU ⇒ set decision to 'Y'

CREATE TRIGGER AutoAccept
AFTER INSERT ON Apply

REFERENCING NEW AS NewApp

WHEN (NewApp.location = 'NCSU')

~~condition~~
AND $3.9 < (\text{SELECT GPA FROM Student WHERE ID} = \text{NewApp.ID})$

UPDATE Apply

SET decision = 'Y'

WHERE ID = NewApp. ID

AND location

= NewApp. Location

FOR EACH Row;



REFERENCING NEW TABLE AS New Apps

[NO "WHEN" CONDITION]

UPDATE Appy

SET decision = 'Y'

WHERE (ID, location)

IN (SELECT ID, location

FROM New Apps)

"condition"
AND location = 'NCVD'
AND 3.9 < ();



If campus enrollment increases from below 7000, delete all "new" applications to the campus (dated after 2/15/20), and set all 'Y' decisions for applications before 2/15/20 to 'U'

CREATE OR REPLACE TRIGGER TooMany
 event [AFTER UPDATE OF enrollment]
 ON campus

REFERENCING OLD AS OldVal
 cond [NEW AS NewVal
 WHEN (OldVal.enrollment < 7000
 AND NewVal.enrollment >= 7000)



BEGIN

DELETE FROM Apply
WHERE location = New Val. location
AND date > 2/15/20;

UPDATE Apply

SET decision = 'U'

WHERE location =

old Val. location

AND decision = 'Y'

END

FOR EACH Row;



New topic :

TRANSACTIONS



Example 6.41 in book 2nd ed

Accounts (acctNo, balance)
xfer \$100: 120 → 453]

UPDATE Accounts

SET balance = balance - 100
WHERE acctNo = 120;

Some problem
problem

UPDATE Accounts

SET balance = balance + 100
WHERE acctNo = 453;

Success



failure

Transactions:

Solution to problems of

- bad interactions, and
- atomicity

Def: A transaction is

one or more operations
on db system that must be
executed

- atomically, and

- (typically) in a

serializable manner



ACID transactions

- * Atomic
- * Consistent
- * Isolated
- * Durable



Each SQL statement
issued at terminal or
in application program
- is a transaction
- unless you say to the system
`(START TRANSACTION S)`



Always assume
that in an individual/
transaction

- all its operations will be
executed by the PMS
in the stated order



[Slide 8]

State 1

Sells: bar beer price

'J'	'B'	2.50
'J'	'M'	3.00

State 2

(after (del))

Sells: bar beer price
 [empty relation]

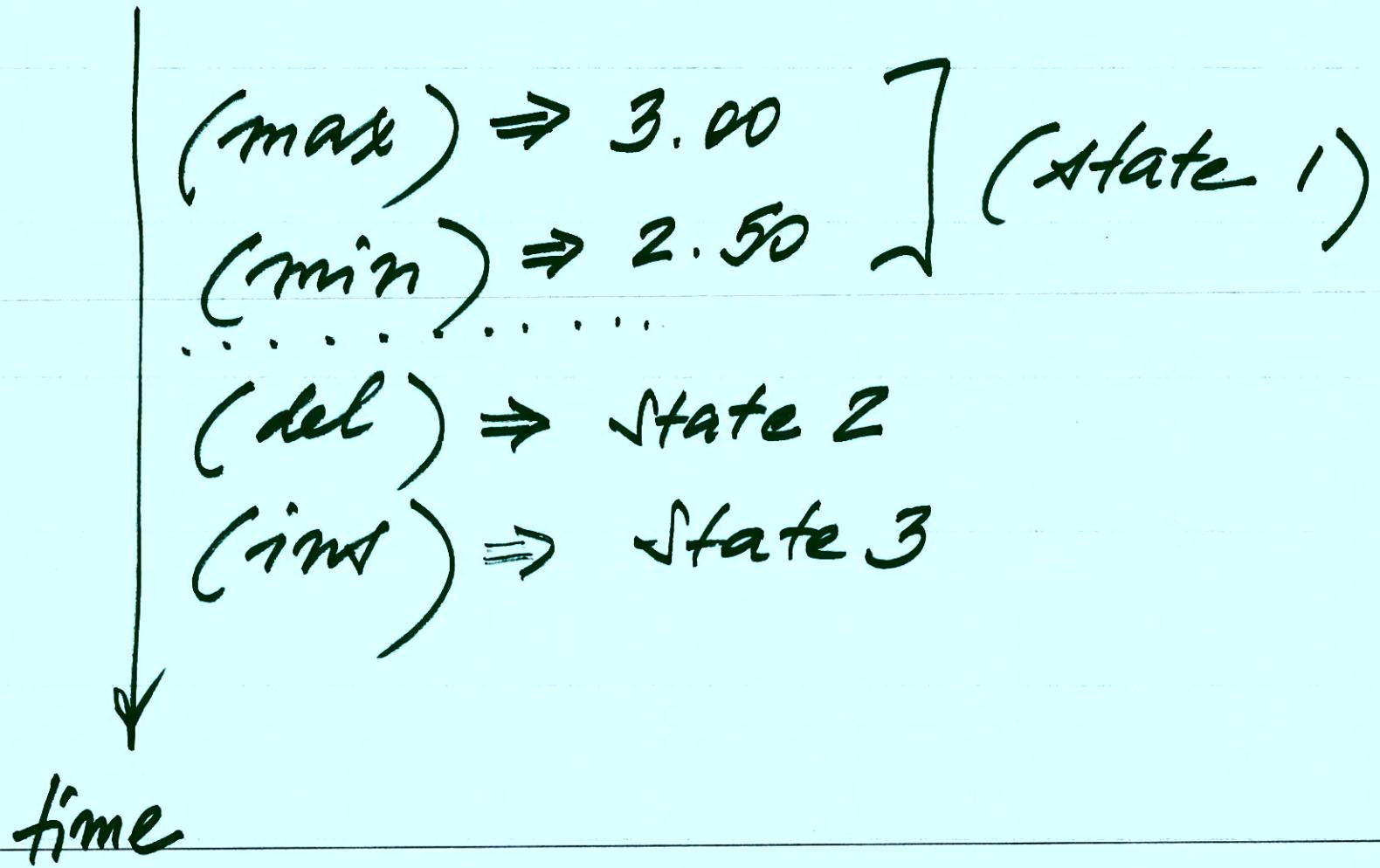
State 3

(after ('ins'))

Sells: bar beer price

'J'	'H'	3.50
-----	-----	------





(del) \Rightarrow state 2
(ins) \Rightarrow state 3
...
(max) $\Rightarrow 3.50$
(min) $\Rightarrow 3.50$

time

[on state 3]



