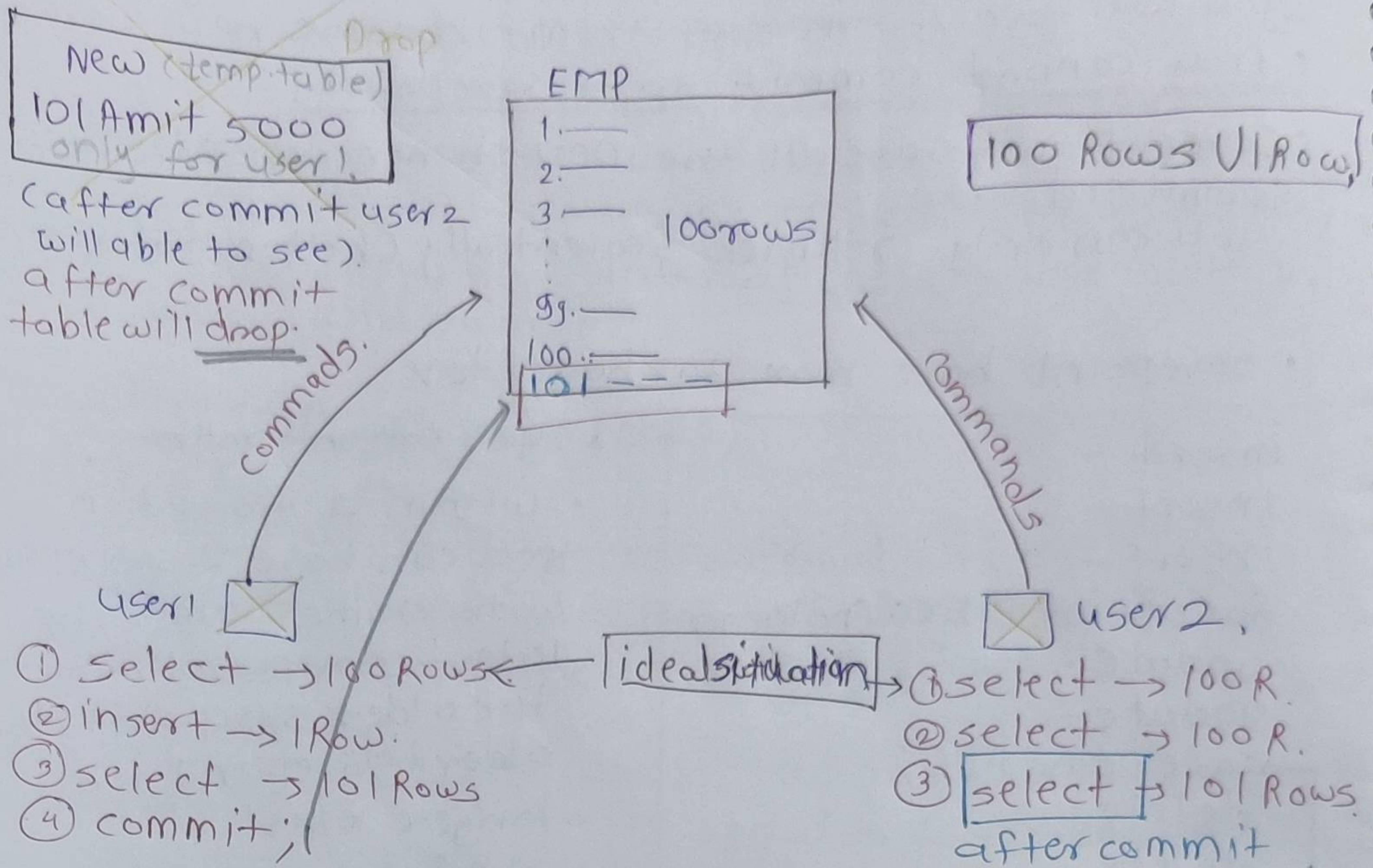


Read and write consistency:-

Case: 1 (for insert)

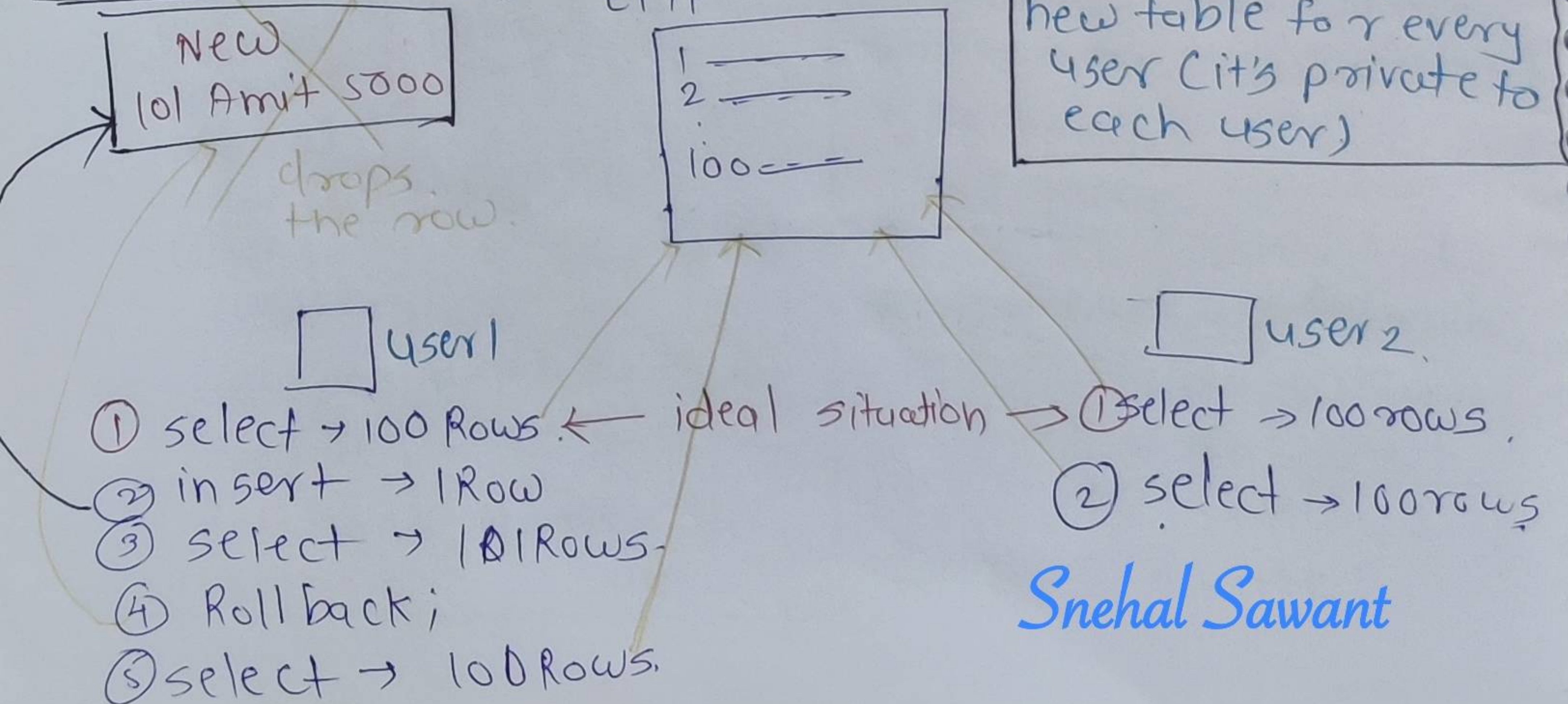


• In a multi-user environment, when you select from a table, you can view only the committed data of other users.

Plus (union)

The changes made by you.

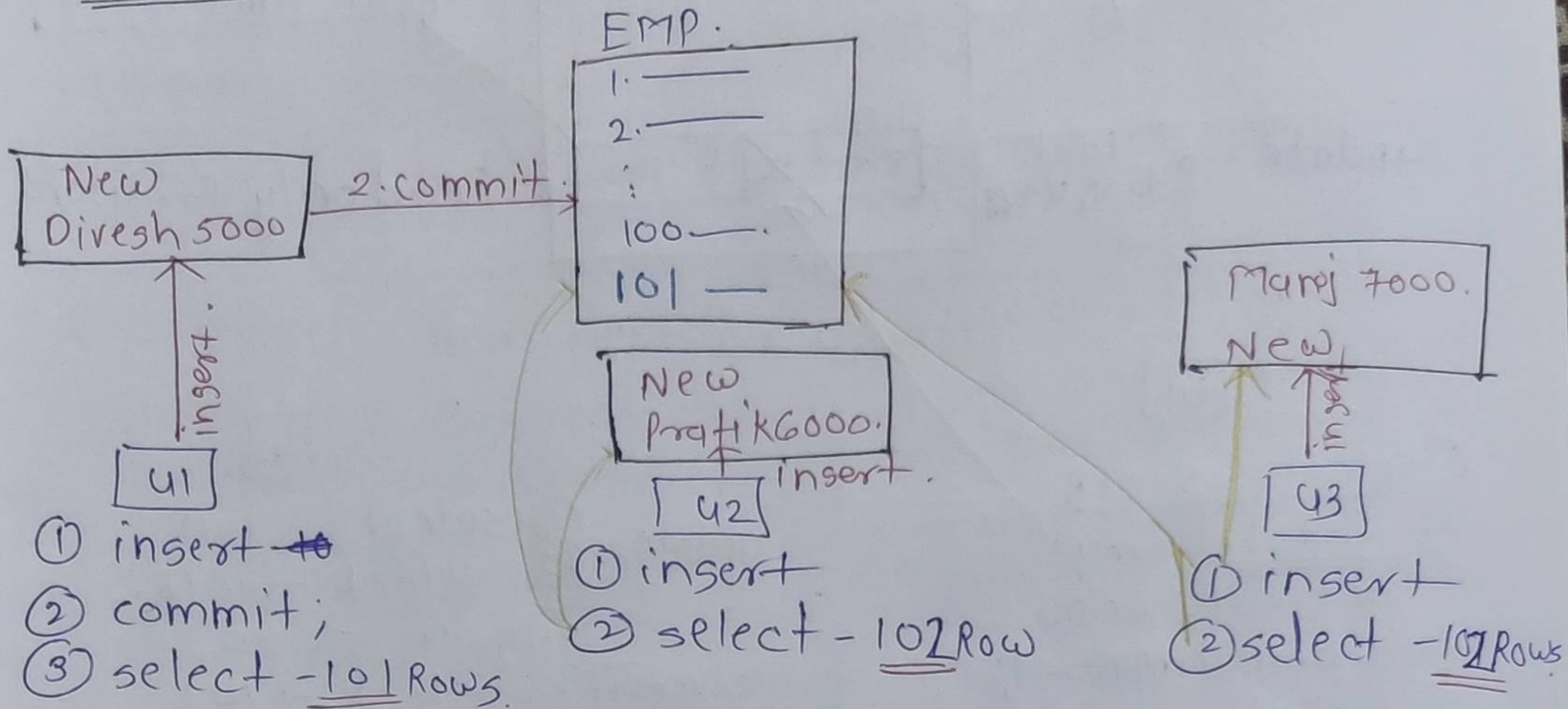
Case-2 (for insert)



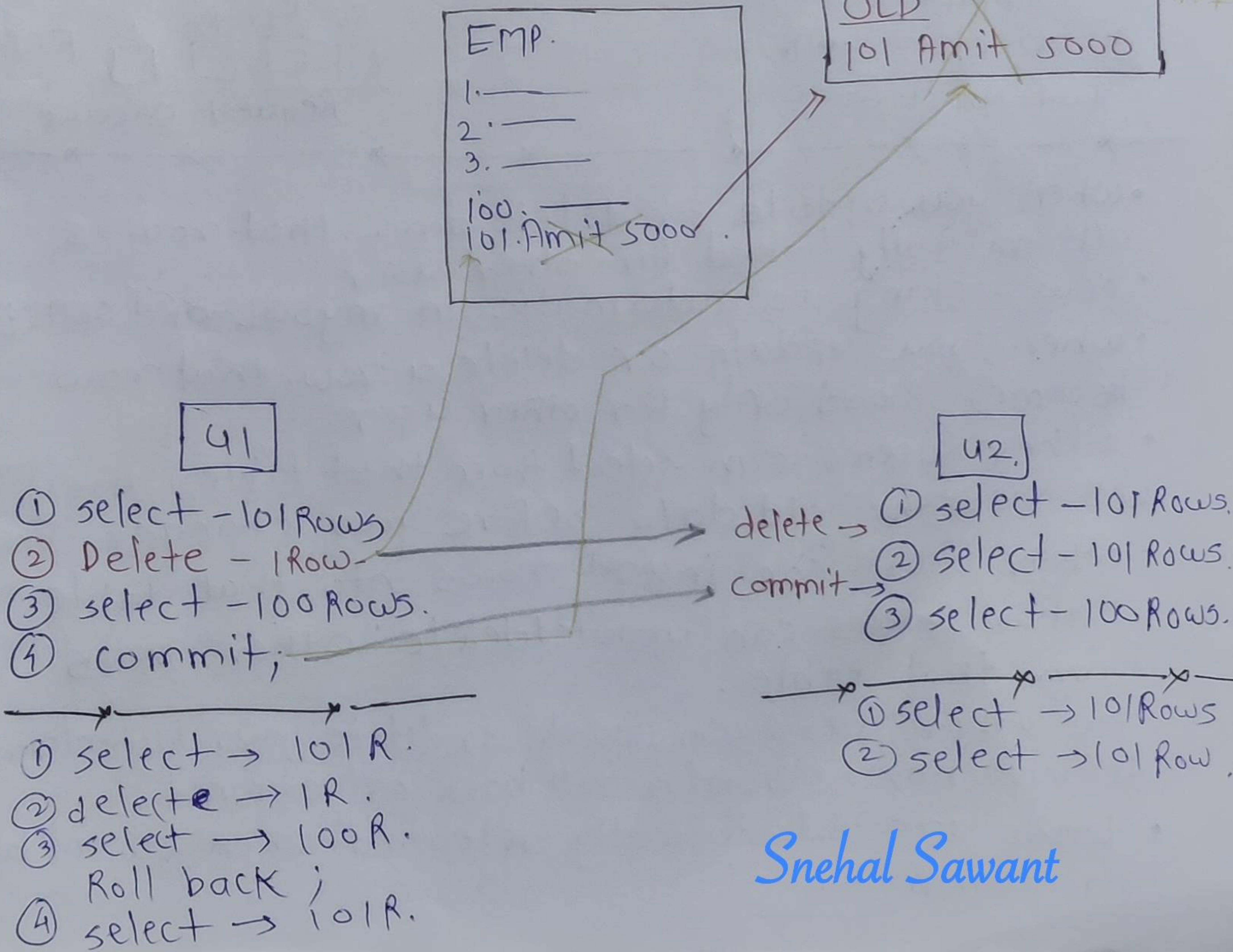
Snehal Sawant

- There is a separate new table for every user.
(it's private to each user).

Multiuser (for insert)

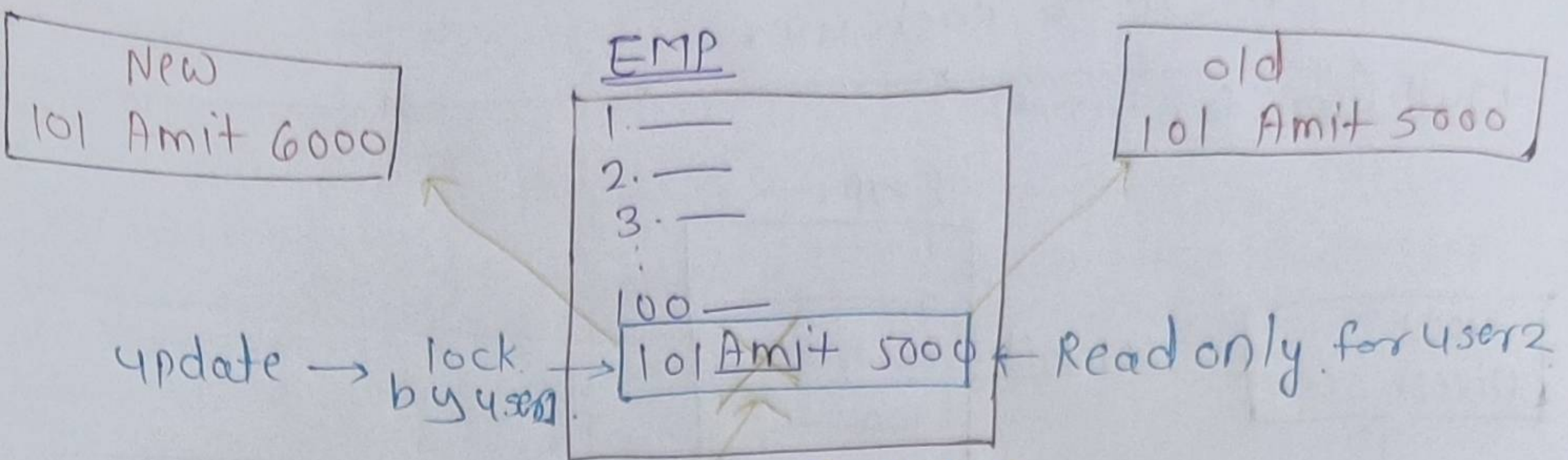


For Delete.



Snehal Sawant

For update:-



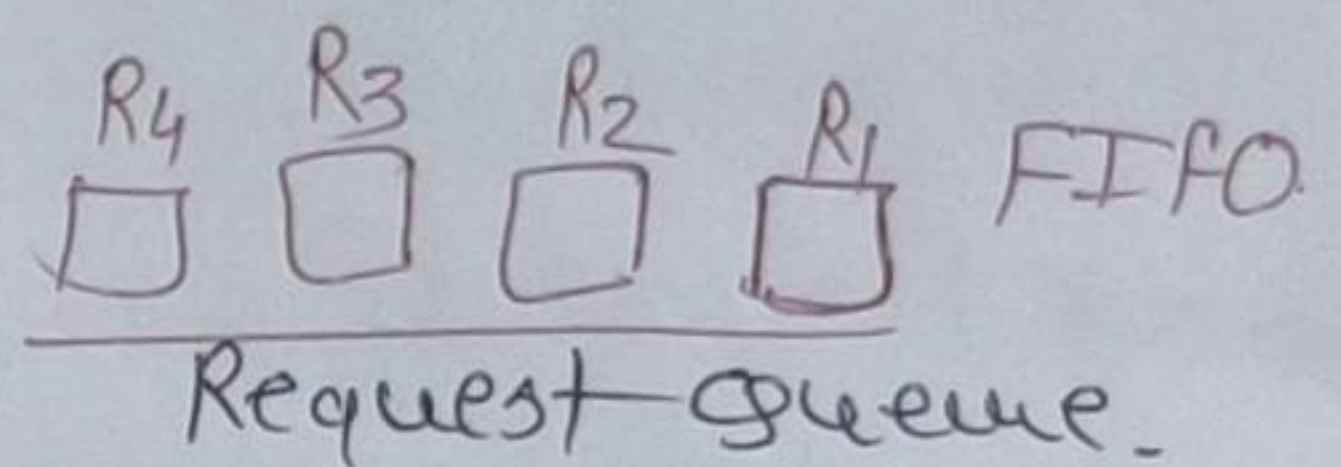
Snehal Sawant

U1

- ① select → 101 R.
- ② update emp
set sal = 6000
where empno = 1;
- ③ select → 101 R.
101 Amit 6000
- ④ commit;
- ⑤ select → 101 R.
101 Amit 6000

U2

- update →
- ① select → 101 R.
 - ② select → 101 R.
101 Amit 5000.
 - ③ select → 101
101 Amit 6000.
- commit



- When you update or delete a row, that row is automatically locked for other users.
- Row locking is automatic in MySQL and ORACLE.
- When you update or delete a row, that row becomes Read-only for other users.
- Other users can select from that table; they will view the old data before your changes.
- Other users can insert rows into that table.
- Other users can update/delete "other" rows from that table.
- No other users can update or delete your locked row, till you have issued a roll back or commit.
- Locks are automatically released when you roll back or commit.

Optimistic Row Locking mechanism of MySQL →

- automatic row locking

Pessimistic row locking →

- you manually lock the rows in advance before issuing update or delete.

To lock the rows manually, you have to use select statement with the FOR UPDATE clause.

e.g. select * from dept for update;

- Locks are automatically released when you rollback or commit.

• select * from emp
where deptno = 10
for update wait;

(by default wait)

• select * from emp
where deptno = 10
for update nowait;

Wait & Nowait options are not available in MySQL.

• select * from emp
where deptno = 10
for update wait 60;

← seconds.

Snehal Sawant

- if row is available, then it will acquire the lock.
- if row is unavailable, then it will wait in the request queue for the specified time period; according it will ~~acquire~~ the lock or abort the operation.

- if row is available, then it will acquire the lock
- if row is unavailable, then it will abort operation.

To try out Row locking in MySQL Workbench; if you get stuck in the request queue. -

- click on Query (menu at the top) → click on stop.

For two query windows in MySQL WB

- click on Query (menu at the top) → new tab to current server → click on it.

Function:

- Routine that returns a value (return statement)
- you can pass parameters/Arguments to a function.
- Function can be overloaded.

EMP	
Varchar(20) FNAME	Varchar(20) ENAME
Arun	Purun
Tarun	Arun
Siran	Kiran
Nutan	Purun.

In ORACLE:-

- select * from emp;
- select fname, lname from emp;
- Concatenate → to join

concat(str1, str2).

select concat(fname, lname) from emp;

To join two strings without blank space bet^h them.
e.g. ArunPurun.

- select concat(concat(fname, ''), lname) from emp;
- inner function
- outer function

- To get space bet^h two string.

eg. Arun Purun.

- max upto 255 levels for function within fun (common for all RDBMS)
- this limit of SQL can be exceeded with the help of views.

In MySQL:-

- Concat(str1, str2, str3, ..., strn).

select concat(fname, ' ' lname) from emp;

Snehal Sawant

- select concat ('Mr. ', fname, ' ', lname) from emp;
Mr. Arun Purun.
- select upper(fname) from emp;
eg. ARUN.
- update emp set fname = upper(fname);

In Oracle:

- select * from emp where upper(fname) = 'ARUN';
soln for case-insensitive query in oracle:
- select * from emp where lower(fname) = 'Arun';
- select initcap(fname) from emp; (Not in MySQL)
eg. Arun.

EMP.	
ENAME	
Arun	Purun
Tarun	Arun
Sirun	Kirun
Nutan	Purun

Snehal Sawant

- select lpad(ename, 25, ' ') from emp;

eg & & & & Arun Purun
 Total 25; Right justified.

Uses:-

1. Right justification.
 2. Billing Receipt.
 3. cheque printing.
- select lpad(ename, 25, '*') from emp;

eg
 * * * * * Arun Purun

- select rpad(²⁵ename, 25, ' ') from emp;
- select rpad(ename, 25, '*') from emp;

Use:-

1. left justification of numeric data.
2. to convert varchar to char (convert variable length to fixed-length).
3. Billing receipt.
4. Cheque printing.

• select ltrim(ename) from emp;

eg. ~~off~~ Arun Purun.

Uses:-

a. left justification.

• select rtrim(ename) from emp;

use:-

a. Convert char to Varchar
(converting fixed length to variable length).

b. right justification of char column.

lpad (rtrim(ename), ..., ...)

• select trim(ename) from emp;
• removes blank spaces from both the sides.

• select substr(ename, 3) from emp;
3 → starting position.

eg. o/p.

un purun.

run Arun.

• select substr(ename, 3, 2) from emp;

eg. o/p

3 → starting position

2 → Number of character.

un
ru.

• select substr(ename, -3) from emp;

it will start from right.

• o/p = run

Snehal Sawant

• select substr(ename, -3, 2) from emp;

eg: o/p un / ru
 un / ru

uses :-

- a. extract a part of the string.
- substr('New Mumbai', 1, 3) → New.
- select replace(ename, 'un', 'xy') from emp;
un → xy o/p = Arxy, purxy,
Tarxy, Axy, ~~Sirxy~~
Sirxy, Kirxy
Nutan purxy.
- select replace(ename, 'un', 'xyz') from emp;
un → xyz
- select replace(ename, 'un', 'x') from emp;
un → x
- select replace(ename, 'un', ' ') from emp;
un → ' '
- select replace(ename, 'un') from emp;
un → (Not supported by MySQL)
(3rd parameter compulsory in MySQL).
- ex. o/p Ar pur.
- select instr(ename, 'un') from emp;
 - returns starting position of string.
 - if string is not found then it returns 0.

uses :-

- a. to check if one string existing in another string.
- instr(experience, 'oracle') if 0 → no oracle exp.
if non zero → has oracle exp.

Varchar.
ENAME
ANJALI
Basit
Chetan

EMP.

- select length (ename) from emp;

o/p: 6
5
6

- select ascii (ename) from emp;
returns ASCII value of 1st char. o/p 65
66
67.
- select ascii (substr (ename, 2)) from emp;
returns ASCII value of 2nd char.
- select ascii (substr (ename, -1)) from emp;

- 'z' → 122.

Snehal Sawant

- select ascii ('z') from emp;
o/p 122
122
122
returns 122 for 3 times.
122 returns for each row.

- select distinct ascii ('z') from emp;
122.
- select sal, sal*12, 3*12, ascii ('z') from emp;
- use mysql;
select distinct ascii ('z') from user;
- select ascii ('z') from dual;

o/p: 122.

- dual is a system table. (common for all RDBMS).
- it contains only 1 row and 1 column.
- dual is a dummy table.
- select substr ('New Mumbai', 1, 3) from dual;
- select 3*12 from dual;
- select 'welcome to CDAC' "MESSAGE" from dual;

- select char(65 using utf8) from dual;

A.

- char(65 using utf8) → where utf8 is the given set for U.S English, else default is binary character set.

- select * from emp where Rname = 'Basit';
it will not work so to work it, ^{wrong spelling.}

- select * from emp where soundex(ename) = soundex('Basit');

- it will work for simple names like.

'arun = arron.

it removes vowels and then compares.

H.W.

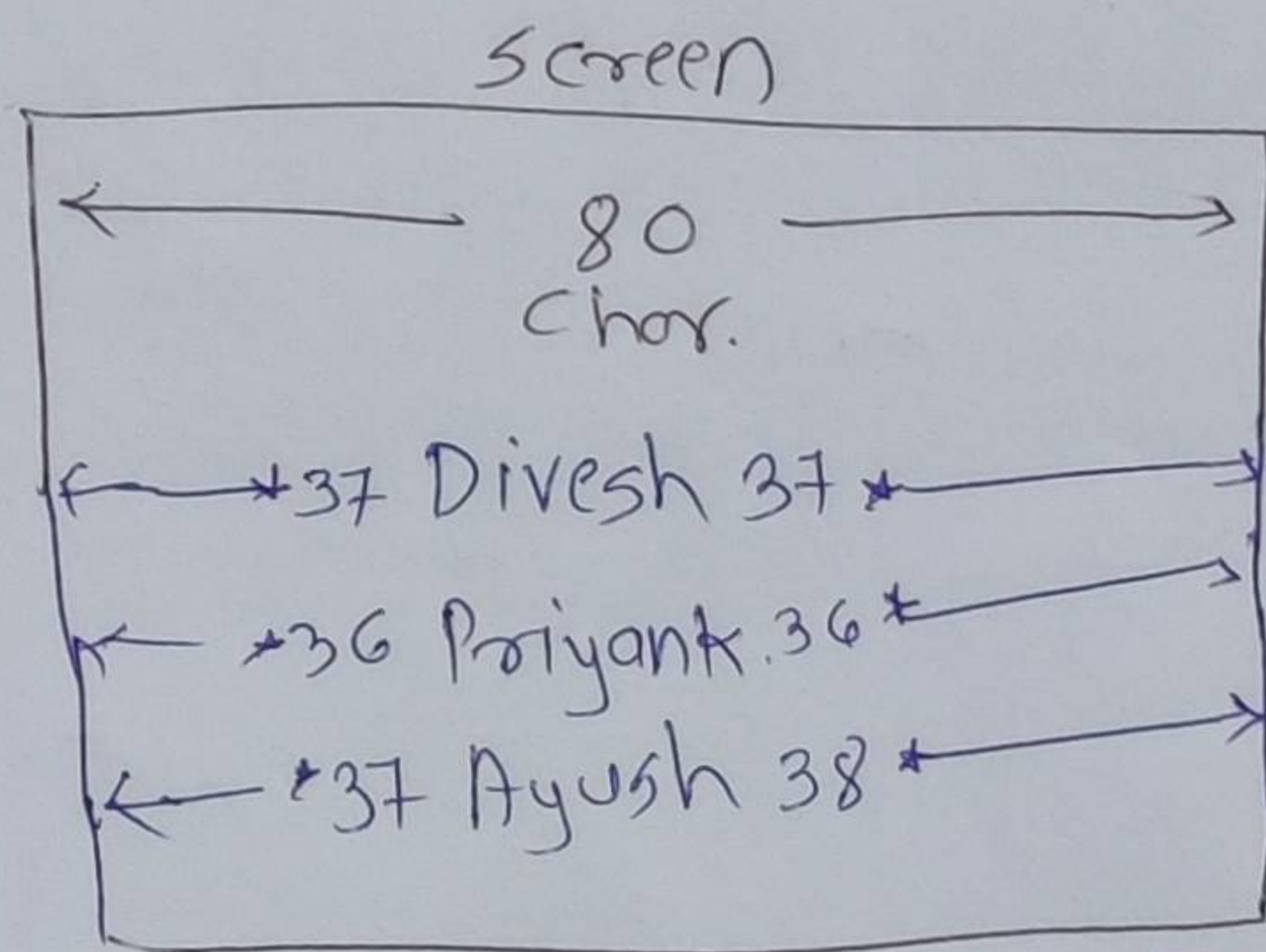
In MySQL:-

1. select → display the Enames initcap.

AYUSH → Ayush.

2. select → display ename center-justified
(Padded with '*')

assume screen-width = 80 char.



Snehal Sawant

3. select → display the ENAME char(20) column
right-justified. assume screen-width = 80 char.

In Oracle:

4. select → display the ENAMES with the vowels removed
(a, e, i, o, u, y)
AYUSH → AH