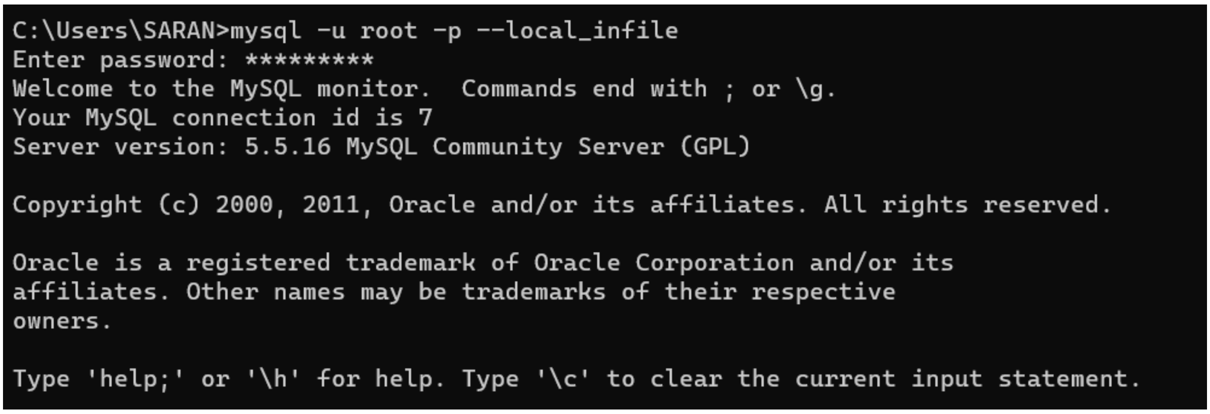
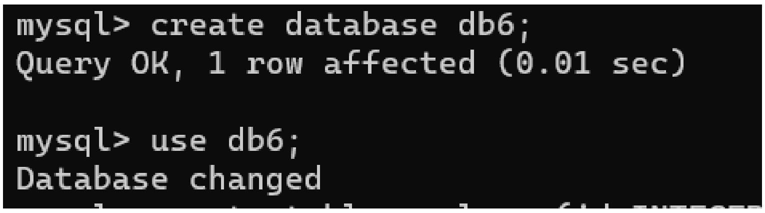
Practice Lab 6

2010110193

1.

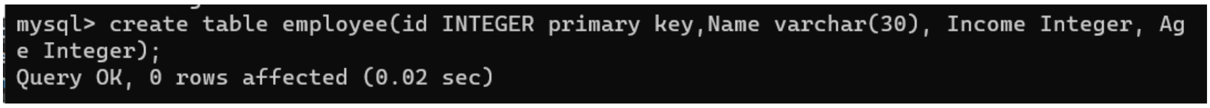


```
C:\Users\SARAN>mysql -u root -p --local_infile
Enter password: *********
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 7
Server version: 5.5.16 MySQL Community Server (GPL)

Copyright (c) 2000, 2011, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```



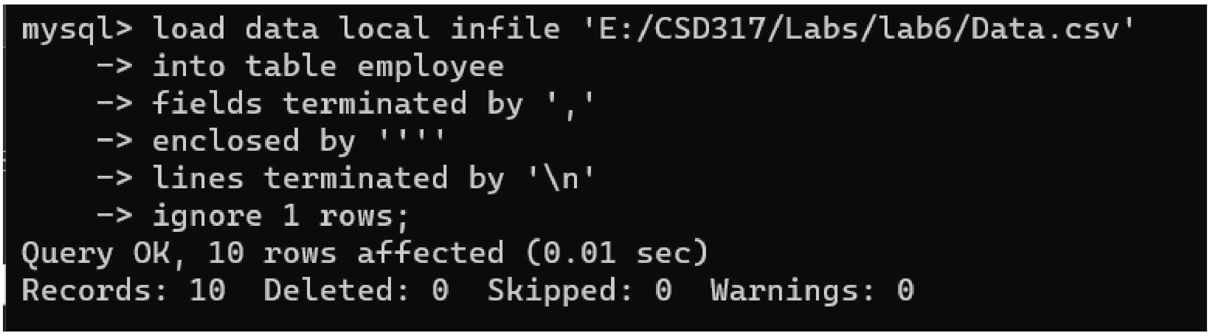
```
mysql> create database db6;
Query OK, 1 row affected (0.01 sec)

mysql> use db6;
Database changed
```

2.



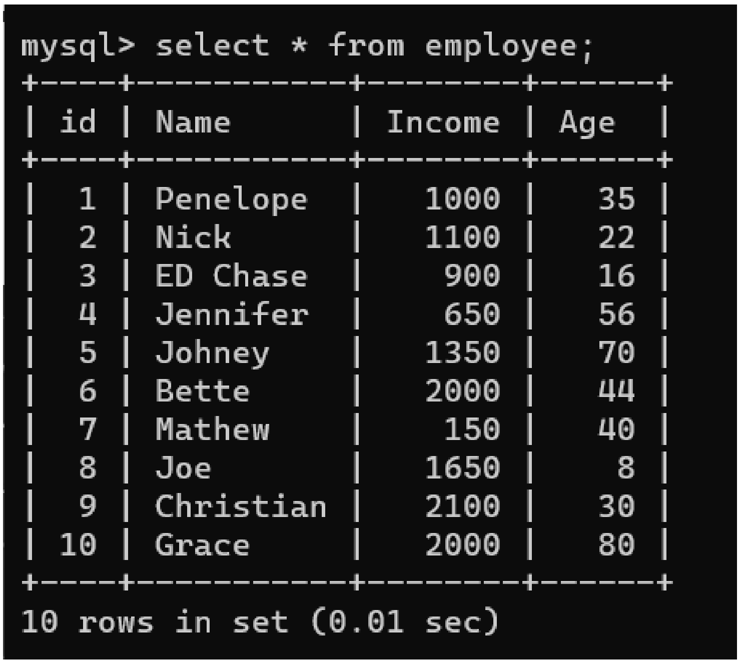
```
mysql> create table employee(id INTEGER primary key,Name varchar(30), Income Integer, Ag
e Integer);
Query OK, 0 rows affected (0.02 sec)
```

3.



```
mysql> load data local infile 'E:/CSD317/Labs/lab6/Data.csv'
    -> into table employee
    -> fields terminated by ','
    -> enclosed by ''''
    -> lines terminated by '\n'
    -> ignore 1 rows;
Query OK, 10 rows affected (0.01 sec)
Records: 10  Deleted: 0  Skipped: 0  Warnings: 0
```

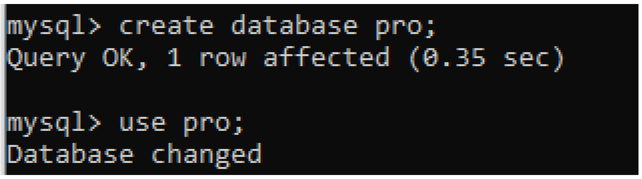
4.



```
mysql> select * from employee;
+----+-----------+--------+------+
| id | Name      | Income | Age  |
+----+-----------+--------+------+
|  1 | Penelope  |   1000 |   35 |
|  2 | Nick      |   1100 |   22 |
|  3 | ED Chase  |    900 |   16 |
|  4 | Jennifer  |    650 |   56 |
|  5 | Johney    |   1350 |   70 |
|  6 | Bette     |   2000 |   44 |
|  7 | Mathew    |    150 |   40 |
|  8 | Joe       |   1650 |    8 |
|  9 | Christian |   2100 |   30 |
| 10 | Grace     |   2000 |   80 |
+----+-----------+--------+------+
10 rows in set (0.01 sec)
```

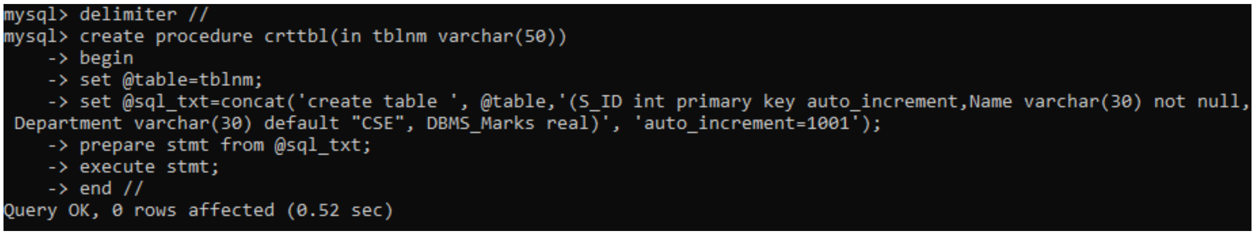
Stored procedure

1.

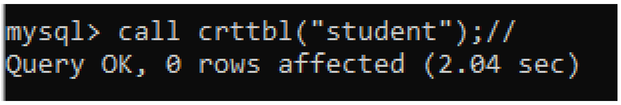


```
mysql> create database pro;
Query OK, 1 row affected (0.35 sec)

mysql> use pro;
Database changed
```

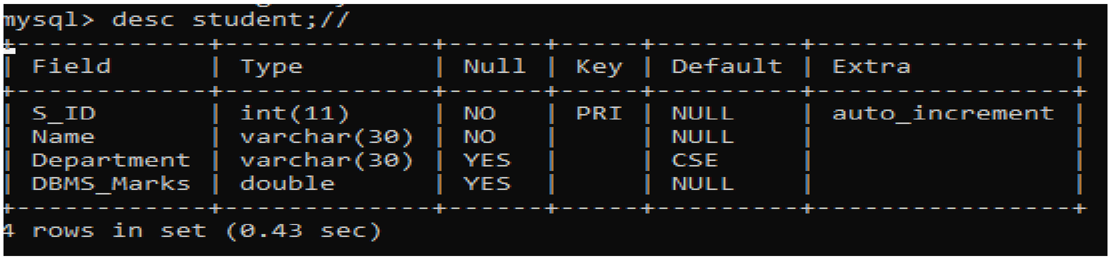
2.



```
mysql> delimiter //
mysql> create procedure crttbl(in tblnm varchar(50))
    -> begin
    -> set @table=tblnm;
    -> set @sql_txt=concat('create table ', @table,'(S_ID int primary key auto_increment,Name varchar(30) not null,
 Department varchar(30) default "CSE", DBMS_Marks real)', 'auto_increment=1001');
    -> prepare stmt from @sql_txt;
    -> execute stmt;
    -> end //
Query OK, 0 rows affected (0.52 sec)
```

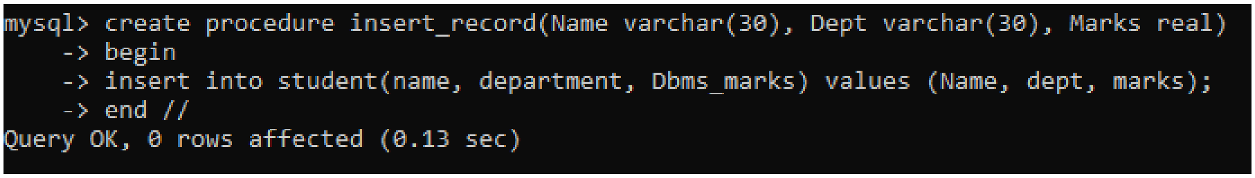


```
mysql> call crttbl("student");//
Query OK, 0 rows affected (2.04 sec)
```



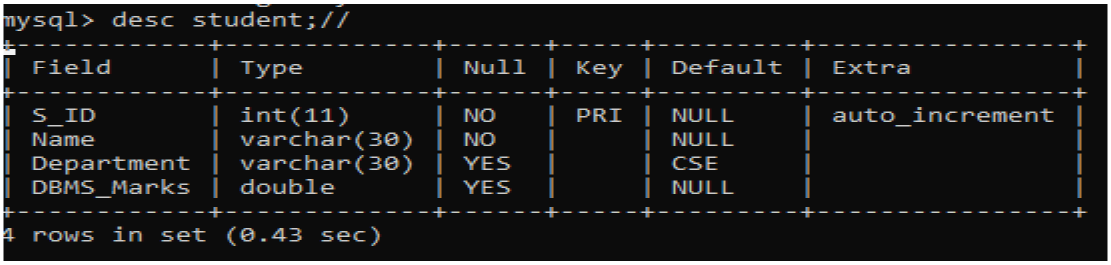
```
mysql> desc student;//
+------------+-------------+------+-----+---------+----------------+
| Field      | Type        | Null | Key | Default | Extra          |
+------------+-------------+------+-----+---------+----------------+
| S_ID       | int(11)     | NO   | PRI | NULL    | auto_increment |
| Name       | varchar(30) | NO   |     | NULL    |                |
| Department | varchar(30) | YES  |     | CSE     |                |
| DBMS_Marks | double      | YES  |     | NULL    |                |
+------------+-------------+------+-----+---------+----------------+
4 rows in set (0.43 sec)
```

3.



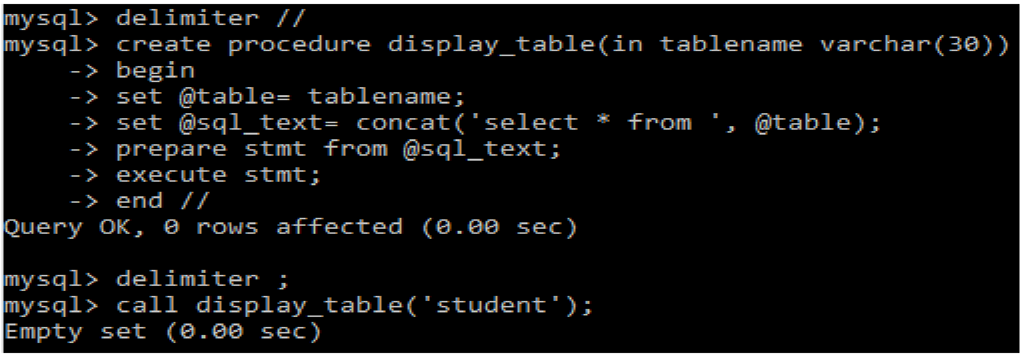
```
mysql> create procedure insert_record(Name varchar(30), Dept varchar(30), Marks real)
    -> begin
    -> insert into student(name, department, Dbms_marks) values (Name, dept, marks);
    -> end //
Query OK, 0 rows affected (0.13 sec)
```

4.



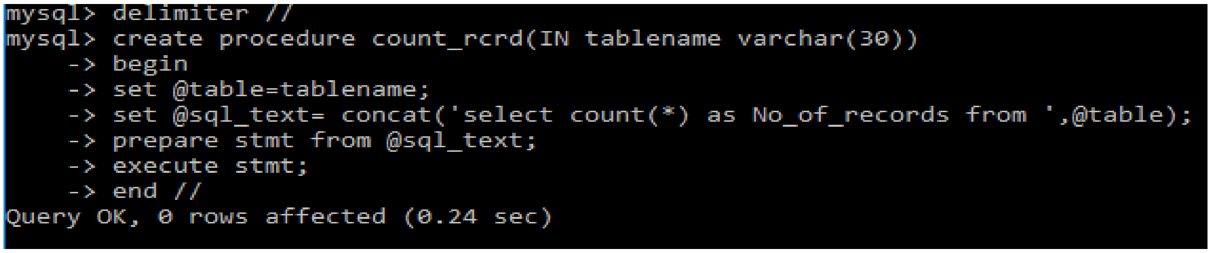
```
mysql> desc student;//
+------------+-------------+------+-----+---------+----------------+
| Field      | Type        | Null | Key | Default | Extra          |
+------------+-------------+------+-----+---------+----------------+
| S_ID       | int(11)     | NO   | PRI | NULL    | auto_increment |
| Name       | varchar(30) | NO   |     | NULL    |                |
| Department | varchar(30) | YES  |     | CSE     |                |
| DBMS_Marks | double      | YES  |     | NULL    |                |
+------------+-------------+------+-----+---------+----------------+
4 rows in set (0.43 sec)
```

5.



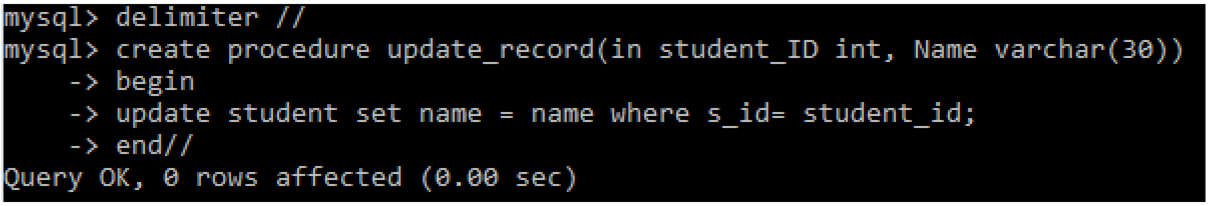
```
mysql> delimiter //
mysql> create procedure display_table(in tablename varchar(30))
    -> begin
    -> set @table= tablename;
    -> set @sql_text= concat('select * from ', @table);
    -> prepare stmt from @sql_text;
    -> execute stmt;
    -> end //
Query OK, 0 rows affected (0.00 sec)

mysql> delimiter ;
mysql> call display_table('student');
Empty set (0.00 sec)
```

6.

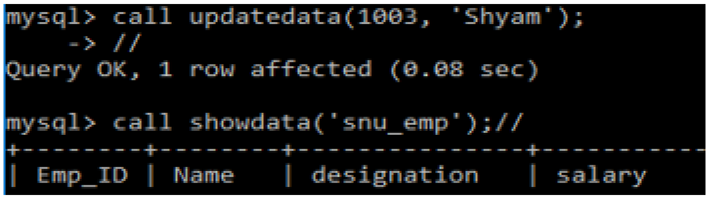


```
mysql> delimiter //
mysql> create procedure count_rcrd(IN tablename varchar(30))
    -> begin
    -> set @table=tablename;
    -> set @sql_text= concat('select count(*) as No_of_records from ',@table);
    -> prepare stmt from @sql_text;
    -> execute stmt;
    -> end //
Query OK, 0 rows affected (0.24 sec)
```

7.



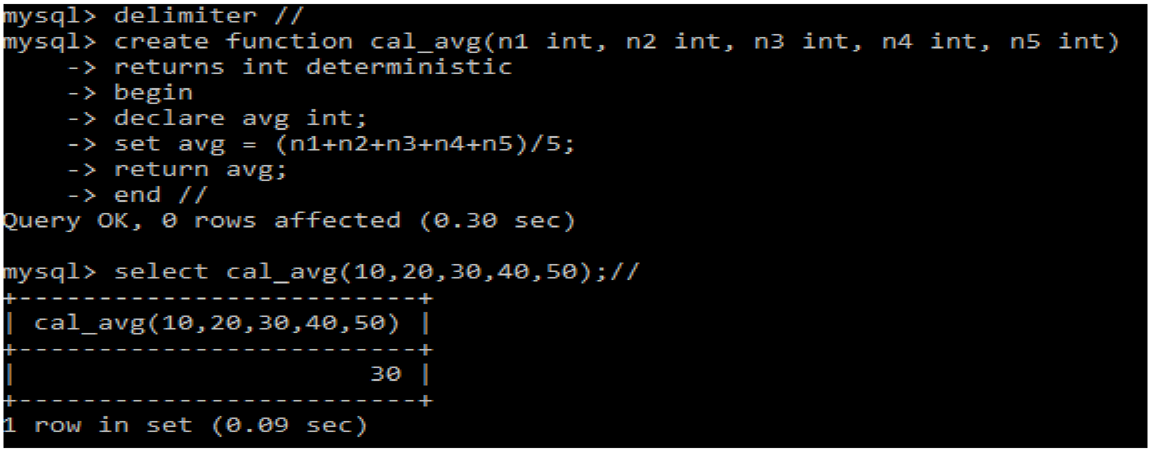
```
mysql> delimiter //
mysql> create procedure update_record(in student_ID int, Name varchar(30))
    -> begin
    -> update student set name = name where s_id= student_id;
    -> end//
Query OK, 0 rows affected (0.00 sec)
```



```
mysql> call updatedata(1003, 'Shyam');
    -> //
Query OK, 1 row affected (0.08 sec)

mysql> call showdata('snu_emp');//
+--------+--------+---------------+----------
| Emp_ID | Name   | designation   | salary
```

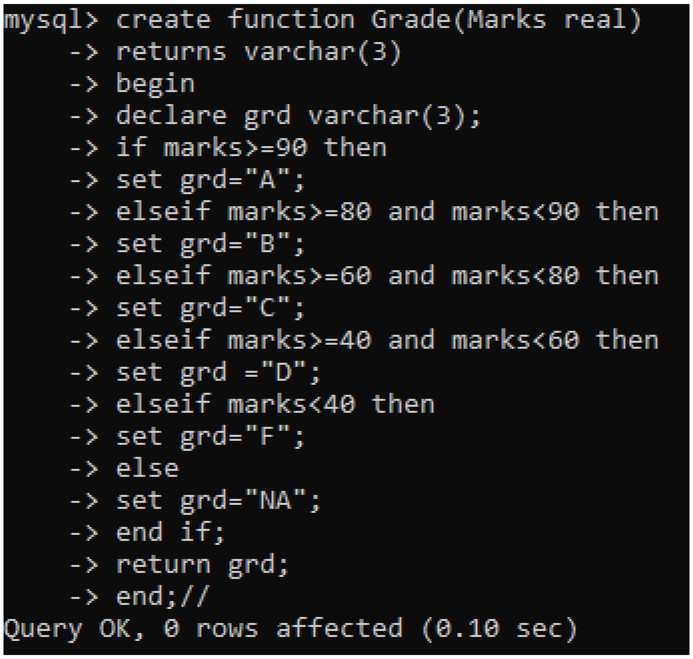
Stored Function

1.

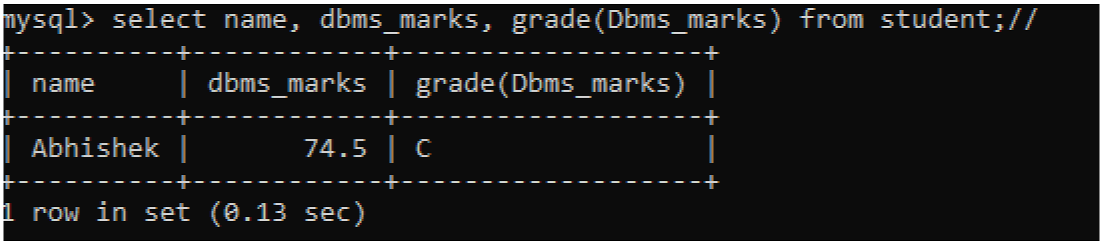


```
mysql> delimiter //
mysql> create function cal_avg(n1 int, n2 int, n3 int, n4 int, n5 int)
    -> returns int deterministic
    -> begin
    -> declare avg int;
    -> set avg = (n1+n2+n3+n4+n5)/5;
    -> return avg;
    -> end //
Query OK, 0 rows affected (0.30 sec)

mysql> select cal_avg(10,20,30,40,50);//
+-------------------------+
| cal_avg(10,20,30,40,50) |
+-------------------------+
|                      30 |
+-------------------------+
1 row in set (0.09 sec)
```

2.

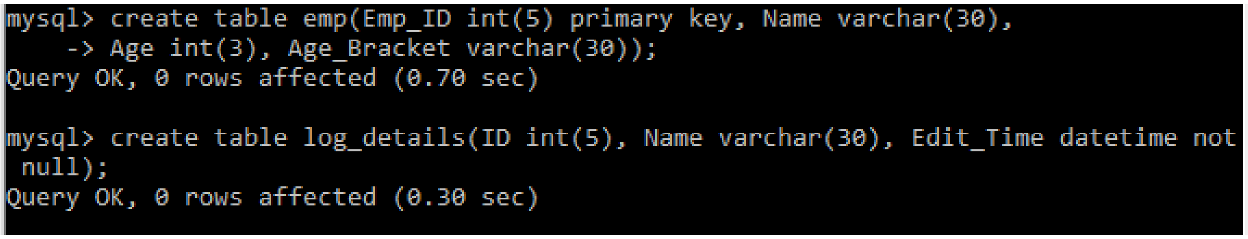


```
mysql> create function Grade(Marks real)
    -> returns varchar(3)
    -> begin
    -> declare grd varchar(3);
    -> if marks>=90 then
    -> set grd="A";
    -> elseif marks>=80 and marks<90 then
    -> set grd="B";
    -> elseif marks>=60 and marks<80 then
    -> set grd="C";
    -> elseif marks>=40 and marks<60 then
    -> set grd ="D";
    -> elseif marks<40 then
    -> set grd="F";
    -> else
    -> set grd="NA";
    -> end if;
    -> return grd;
    -> end;//
Query OK, 0 rows affected (0.10 sec)
```



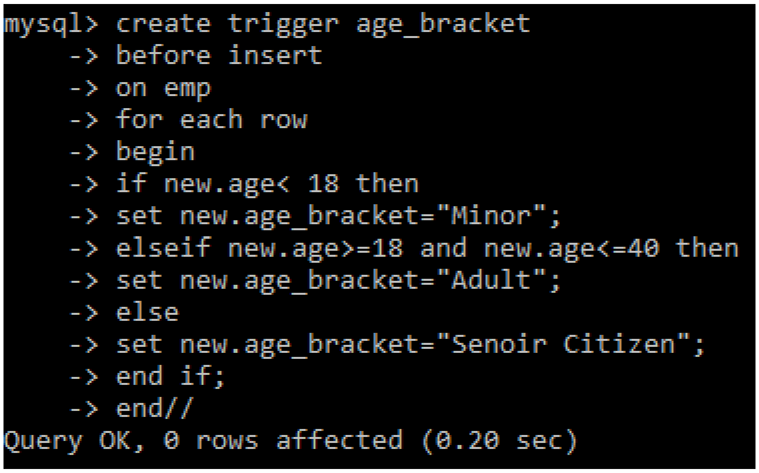
```
mysql> select name, dbms_marks, grade(Dbms_marks) from student;//
+----------+------------+-------------------+
| name     | dbms_marks | grade(Dbms_marks) |
+----------+------------+-------------------+
| Abhishek |       74.5 | C                 |
+----------+------------+-------------------+
1 row in set (0.13 sec)
```

Solution Practice Lab#8

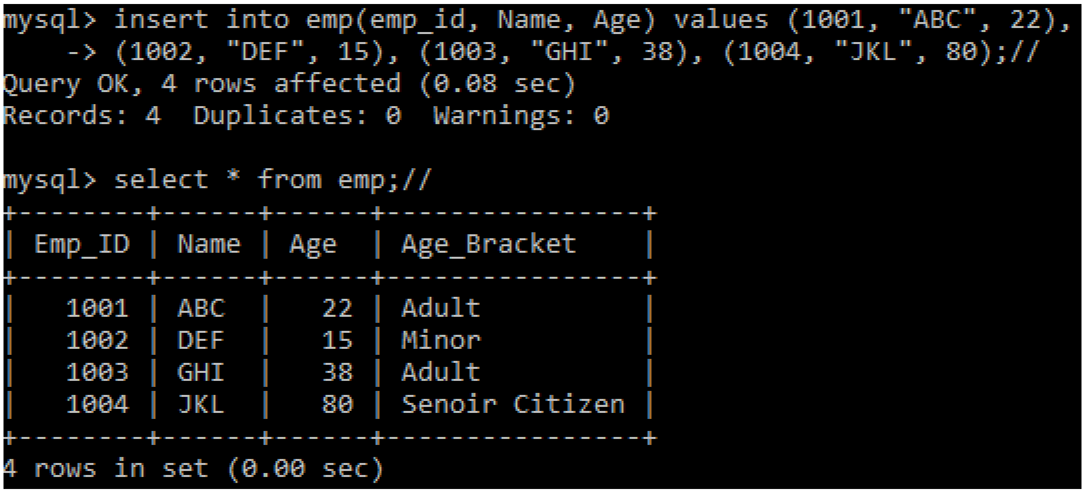


```
mysql> create table emp(Emp_ID int(5) primary key, Name varchar(30),
    -> Age int(3), Age_Bracket varchar(30));
Query OK, 0 rows affected (0.70 sec)

mysql> create table log_details(ID int(5), Name varchar(30), Edit_Time datetime not
 null);
Query OK, 0 rows affected (0.30 sec)
```

1

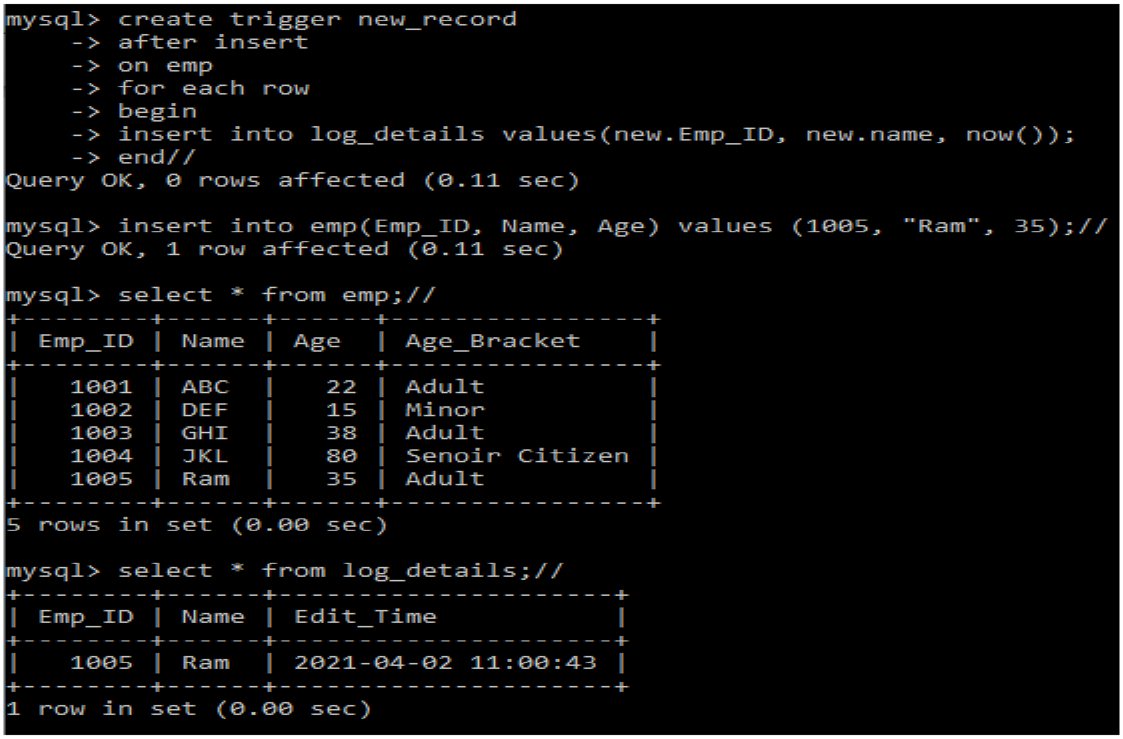


```
mysql> create trigger age_bracket
    -> before insert
    -> on emp
    -> for each row
    -> begin
    -> if new.age< 18 then
    -> set new.age_bracket="Minor";
    -> elseif new.age>=18 and new.age<=40 then
    -> set new.age_bracket="Adult";
    -> else
    -> set new.age_bracket="Senoir Citizen";
    -> end if;
    -> end//
Query OK, 0 rows affected (0.20 sec)
```



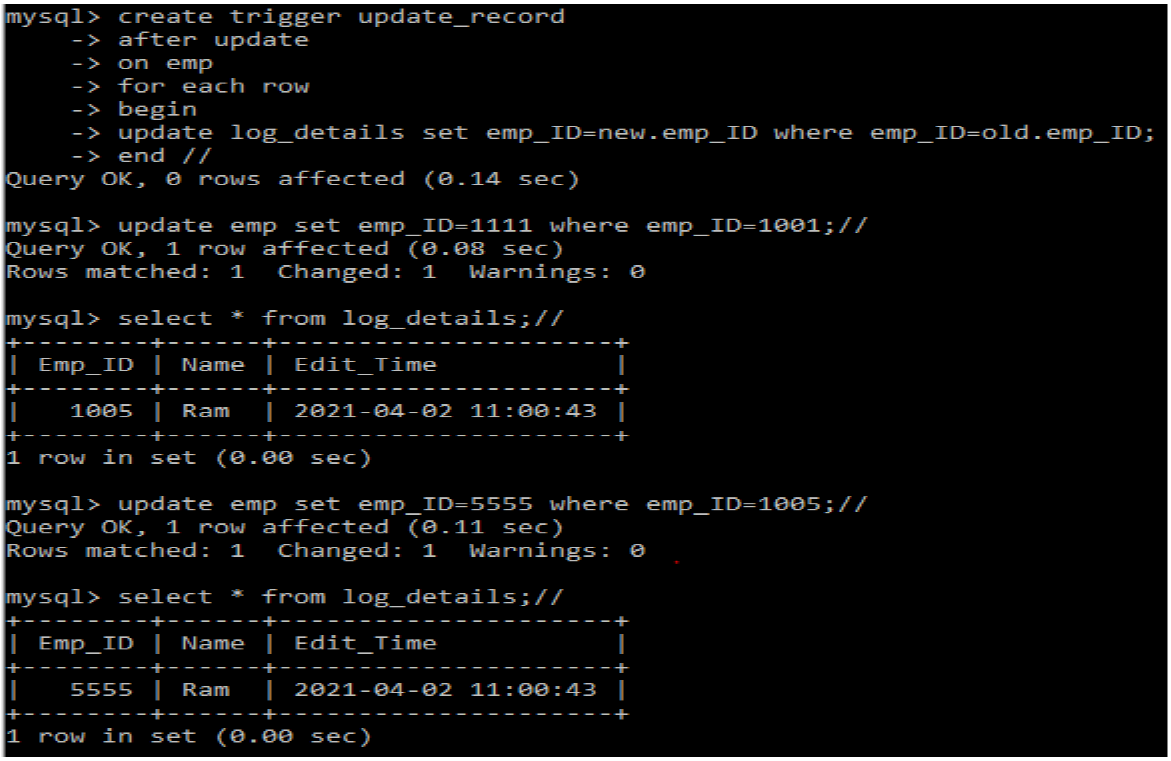
```
mysql> insert into emp(emp_id, Name, Age) values (1001, "ABC", 22),
    -> (1002, "DEF", 15), (1003, "GHI", 38), (1004, "JKL", 80);//
Query OK, 4 rows affected (0.08 sec)
Records: 4  Duplicates: 0  Warnings: 0

mysql> select * from emp;//
+--------+------+------+----------------+
| Emp_ID | Name | Age  | Age_Bracket    |
+--------+------+------+----------------+
|   1001 | ABC  |   22 | Adult          |
|   1002 | DEF  |   15 | Minor          |
|   1003 | GHI  |   38 | Adult          |
|   1004 | JKL  |   80 | Senoir Citizen |
+--------+------+------+----------------+
4 rows in set (0.00 sec)
```

2



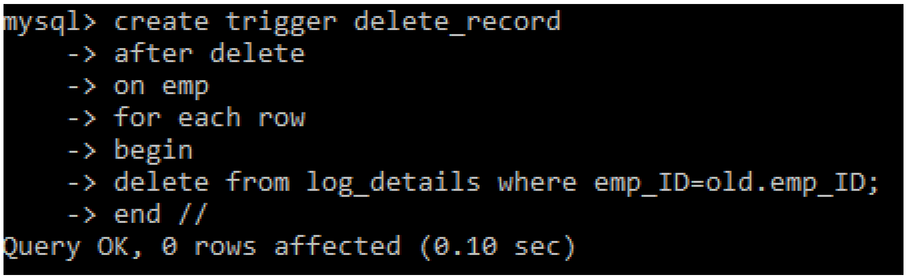
```
mysql> create trigger new_record
    -> after insert
    -> on emp
    -> for each row
    -> begin
    -> insert into log_details values(new.Emp_ID, new.name, now());
    -> end//
Query OK, 0 rows affected (0.11 sec)

mysql> insert into emp(Emp_ID, Name, Age) values (1005, "Ram", 35);//
Query OK, 1 row affected (0.11 sec)

mysql> select * from emp;//
+--------+------+------+-----------------+
| Emp_ID | Name | Age  | Age_Bracket     |
+--------+------+------+-----------------+
|   1001 | ABC  |   22 | Adult           |
|   1002 | DEF  |   15 | Minor           |
|   1003 | GHI  |   38 | Adult           |
|   1004 | JKL  |   80 | Senoir Citizen  |
|   1005 | Ram  |   35 | Adult           |
+--------+------+------+-----------------+
5 rows in set (0.00 sec)

mysql> select * from log_details;//
+--------+------+---------------------+
| Emp_ID | Name | Edit_Time           |
+--------+------+---------------------+
|   1005 | Ram  | 2021-04-02 11:00:43 |
+--------+------+---------------------+
1 row in set (0.00 sec)
```

3

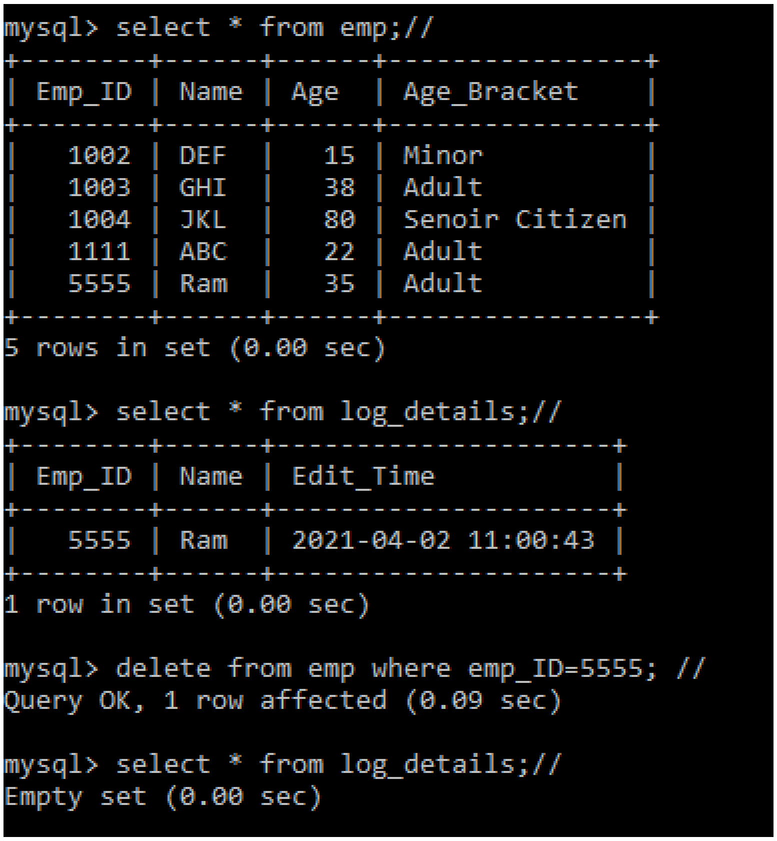


```
mysql> create trigger update_record
    -> after update
    -> on emp
    -> for each row
    -> begin
    -> update log_details set emp_ID=new.emp_ID where emp_ID=old.emp_ID;
    -> end //
Query OK, 0 rows affected (0.14 sec)

mysql> update emp set emp_ID=1111 where emp_ID=1001;//
Query OK, 1 row affected (0.08 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from log_details;//
+--------+------+---------------------+
| Emp_ID | Name | Edit_Time           |
+--------+------+---------------------+
|   1005 | Ram  | 2021-04-02 11:00:43 |
+--------+------+---------------------+
1 row in set (0.00 sec)

mysql> update emp set emp_ID=5555 where emp_ID=1005;//
Query OK, 1 row affected (0.11 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from log_details;//
+--------+------+---------------------+
| Emp_ID | Name | Edit_Time           |
+--------+------+---------------------+
|   5555 | Ram  | 2021-04-02 11:00:43 |
+--------+------+---------------------+
1 row in set (0.00 sec)
```

4



```
mysql> create trigger delete_record
    -> after delete
    -> on emp
    -> for each row
    -> begin
    -> delete from log_details where emp_ID=old.emp_ID;
    -> end //
Query OK, 0 rows affected (0.10 sec)
```



```
mysql> select * from emp;//
+--------+------+------+----------------+
| Emp_ID | Name | Age  | Age_Bracket    |
+--------+------+------+----------------+
|   1002 | DEF  |   15 | Minor          |
|   1003 | GHI  |   38 | Adult          |
|   1004 | JKL  |   80 | Senoir Citizen |
|   1111 | ABC  |   22 | Adult          |
|   5555 | Ram  |   35 | Adult          |
+--------+------+------+----------------+
5 rows in set (0.00 sec)

mysql> select * from log_details;//
+--------+------+---------------------+
| Emp_ID | Name | Edit_Time           |
+--------+------+---------------------+
|   5555 | Ram  | 2021-04-02 11:00:43 |
+--------+------+---------------------+
1 row in set (0.00 sec)

mysql> delete from emp where emp_ID=5555; //
Query OK, 1 row affected (0.09 sec)

mysql> select * from log_details;//
Empty set (0.00 sec)
```

5

Show triggers;

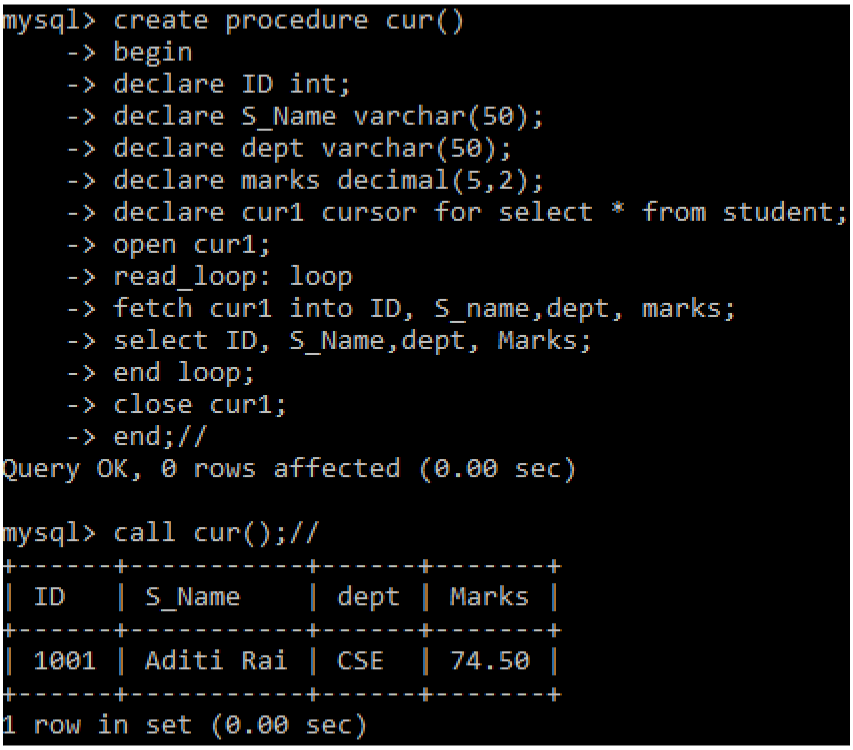
Drop trigger trigger name

Solution Practice Lab#9

Cursor

1.

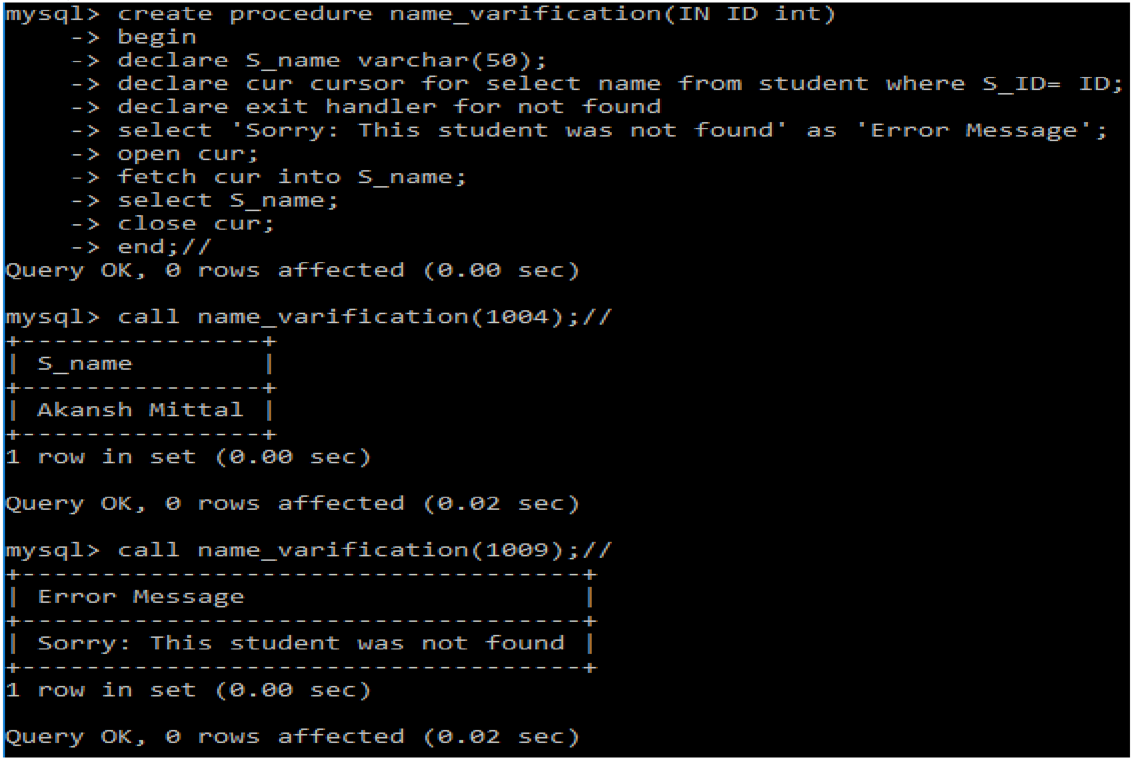
```
mysql> create procedure cur()
   -> begin
   -> declare ID int;
   -> declare S_Name varchar(50);
   -> declare dept varchar(50);
   -> declare marks decimal(5,2);
   -> declare cur1 cursor for select * from student;
   -> open cur1;
   -> read_loop: loop
   -> fetch cur1 into ID, S_name,dept, marks;
   -> select ID, S_Name,dept, Marks;
   -> end loop;
   -> close cur1;
   -> end;//
Query OK, 0 rows affected (0.00 sec)
```



```
mysql> create procedure cur()
    -> begin
    -> declare ID int;
    -> declare S_Name varchar(50);
    -> declare dept varchar(50);
    -> declare marks decimal(5,2);
    -> declare cur1 cursor for select * from student;
    -> open cur1;
    -> read_loop: loop
    -> fetch cur1 into ID, S_name,dept, marks;
    -> select ID, S_Name,dept, Marks;
    -> end loop;
    -> close cur1;
    -> end;//
Query OK, 0 rows affected (0.00 sec)

mysql> call cur();//
+------+-----------+------+-------+
| ID   | S_Name    | dept | Marks |
+------+-----------+------+-------+
| 1001 | Aditi Rai | CSE  | 74.50 |
+------+-----------+------+-------+
1 row in set (0.00 sec)
```

2.

```
mysql> create procedure name_varification(IN ID int)
   -> begin
   -> declare S_name varchar(50);
   -> declare cur cursor for select name from student where S_ID= ID;
   -> declare exit handler for not found
   -> select 'Sorry: This student was not found' as 'Error Message';
   -> open cur;
   -> fetch cur into S_name;
   -> select S_name;
   -> close cur;
   -> end;//
Query OK, 0 rows affected (0.00 sec)
```



```
mysql> create procedure name_varification(IN ID int)
    -> begin
    -> declare S_name varchar(50);
    -> declare cur cursor for select name from student where S_ID= ID;
    -> declare exit handler for not found
    -> select 'Sorry: This student was not found' as 'Error Message';
    -> open cur;
    -> fetch cur into S_name;
    -> select S_name;
    -> close cur;
    -> end;//
Query OK, 0 rows affected (0.00 sec)

mysql> call name_varification(1004);//
+---------------+
| S_name        |
+---------------+
| Akansh Mittal |
+---------------+
1 row in set (0.00 sec)

Query OK, 0 rows affected (0.02 sec)

mysql> call name_varification(1009);//
+-----------------------------------+
| Error Message                     |
+-----------------------------------+
| Sorry: This student was not found |
+-----------------------------------+
1 row in set (0.00 sec)

Query OK, 0 rows affected (0.02 sec)
```

Transactions:

How to make autocommit off:

Show variables like ‘autocommit’;

Set autocommit=0;

Now check autocommit, it would be **off** now

2.

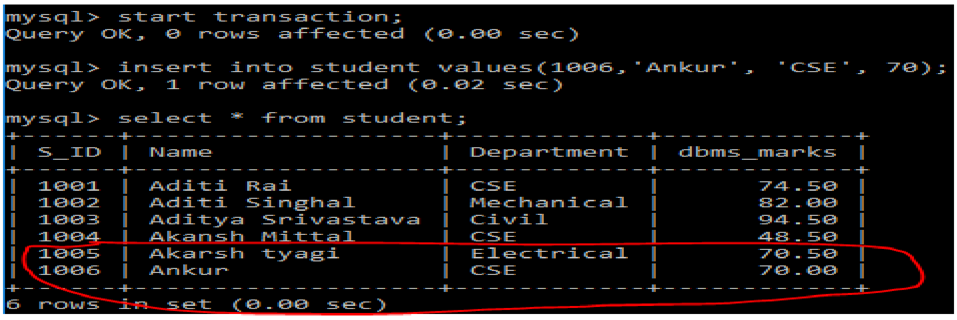
Savepoint sp1;

Savepoint sp2

Rollback to sp1;

3. ii.

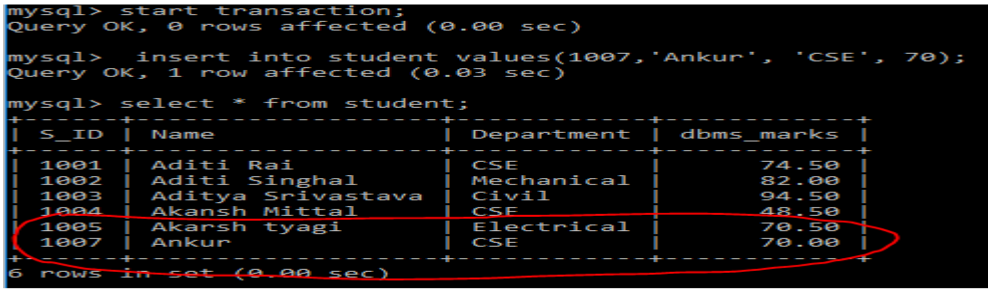
Window 1.



```
mysql> start transaction;
Query OK, 0 rows affected (0.00 sec)

mysql> insert into student values(1006,'Ankur', 'CSE', 70);
Query OK, 1 row affected (0.02 sec)

mysql> select * from student;
+------+---------------------+------------+------------+
| S_ID | Name                | Department | dbms_marks |
+------+---------------------+------------+------------+
| 1001 | Aditi Rai           | CSE        |      74.50 |
| 1002 | Aditi Singhal       | Mechanical |      82.00 |
| 1003 | Aditya Srivastava   | Civil      |      94.50 |
| 1004 | Akansh Mittal       | CSE        |      48.50 |
| 1005 | Akarsh tyagi        | Electrical |      70.50 |
| 1006 | Ankur               | CSE        |      70.00 |
+------+---------------------+------------+------------+
6 rows in set (0.00 sec)
```

1007 is not available

window2.

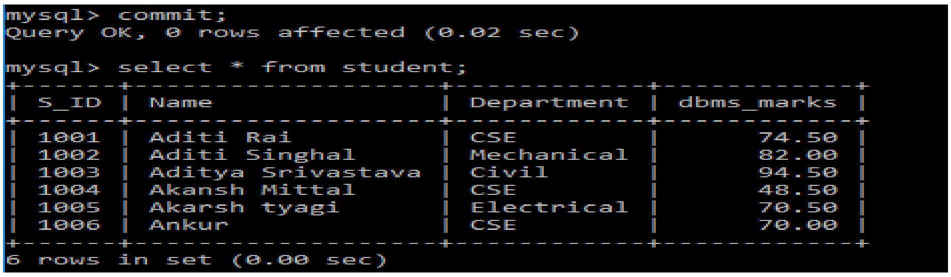


```
mysql> start transaction;
Query OK, 0 rows affected (0.00 sec)

mysql>  insert into student values(1007,'Ankur', 'CSE', 70);
Query OK, 1 row affected (0.03 sec)

mysql> select * from student;
+------+---------------------+------------+------------+
| S_ID | Name                | Department | dbms_marks |
+------+---------------------+------------+------------+
| 1001 | Aditi Rai           | CSE        |      74.50 |
| 1002 | Aditi Singhal       | Mechanical |      82.00 |
| 1003 | Aditya Srivastava   | Civil      |      94.50 |
| 1004 | Akansh Mittal       | CSE        |      48.50 |
| 1005 | Akarsh tyagi        | Electrical |      70.50 |
| 1007 | Ankur               | CSE        |      70.00 |
+------+---------------------+------------+------------+
6 rows in set (0.00 sec)
```

1006 is not available

After committing

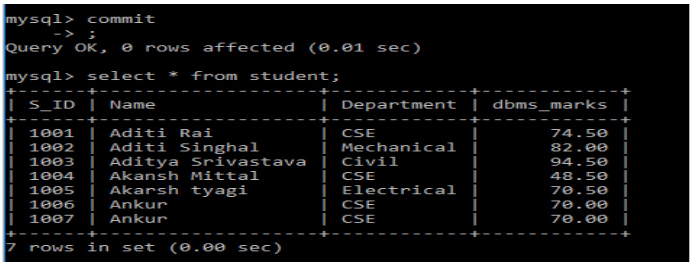
Window 1:



```
mysql> commit;
Query OK, 0 rows affected (0.02 sec)

mysql> select * from student;
+------+---------------------+------------+------------+
| S_ID | Name                | Department | dbms_marks |
+------+---------------------+------------+------------+
| 1001 | Aditi Rai           | CSE        |      74.50 |
| 1002 | Aditi Singhal       | Mechanical |      82.00 |
| 1003 | Aditya Srivastava   | Civil      |      94.50 |
| 1004 | Akansh Mittal       | CSE        |      48.50 |
| 1005 | Akarsh tyagi        | Electrical |      70.50 |
| 1006 | Ankur               | CSE        |      70.00 |
+------+---------------------+------------+------------+
6 rows in set (0.00 sec)
```

after comiting 1007 is visible

Window 2:



```
mysql> commit
    -> ;
Query OK, 0 rows affected (0.01 sec)

mysql> select * from student;
+------+---------------------+------------+------------+
| S_ID | Name                | Department | dbms_marks |
+------+---------------------+------------+------------+
| 1001 | Aditi Rai           | CSE        |      74.50 |
| 1002 | Aditi Singhal       | Mechanical |      82.00 |
| 1003 | Aditya Srivastava   | Civil      |      94.50 |
| 1004 | Akansh Mittal       | CSE        |      48.50 |
| 1005 | Akarsh tyagi        | Electrical |      70.50 |
| 1006 | Ankur               | CSE        |      70.00 |
| 1007 | Ankur               | CSE        |      70.00 |
+------+---------------------+------------+------------+
7 rows in set (0.00 sec)
```

after comiting the trans. 1006 is visible

Same respective affect will impact rest of the operations.

Introduction to Database Systems (CSD202)

Practice Lab#11

Date: 22/04/20

You have to upload the screen shot (black screen) of output with the queries one by one, in a word doc & file name must be->Roll No _Practice Lab#11.

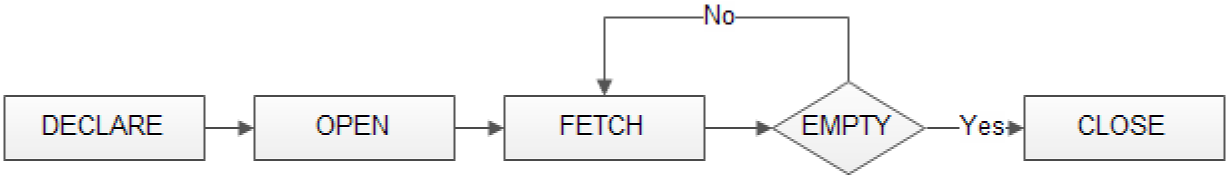
Cursor: -

"A cursor allows you to iterate a set of rows returned by a query and process each row individually." To handle a result set inside a stored procedure, you use a cursor.

Working with MySQL cursor

The following diagram illustrates how MySQL cursor works.

- **MySQL Cursor Steps**



- First, **declare** a cursor by using the DECLARE statement:

  DECLARE cursor_name CURSOR FOR SELECT_statement;

  The cursor declaration must be after any variable declaration. If you declare a cursor before the variable declarations, MySQL will issue an error. A cursor must always associate with a SELECT statement.

- Next, **open** the cursor by using the OPEN statement. The OPEN statement initializes the result set for the cursor, therefore, you must call the OPEN statement before fetching rows from the result set.

  OPEN cursor_name;

- Then, use the **FETCH** statement to retrieve the next row pointed by the cursor and move the cursor to the next row in the result set.

  FETCH cursor_name INTO variables list;

After that, check if there is any row available before fetching it.

- Finally, deactivate the cursor and release the memory associated with it using the **CLOSE** statement:

  CLOSE cursor_name;

It is a good practice to always close a cursor when it is no longer used.

When working with MySQL cursor, you must also declare a NOT FOUND handler to handle the situation when the cursor could not find any row.

Because each time you call the FETCH statement, the cursor attempts to read the next row in the result set. When the cursor reaches the end of the result set, it will not be able to get the data, and a condition is raised. The handler is used to handle this condition.

To declare a **NOT FOUND** handler, you use the following syntax:

DECLARE CONTINUE HANDLER FOR NOT FOUND **SET finished = 1**;

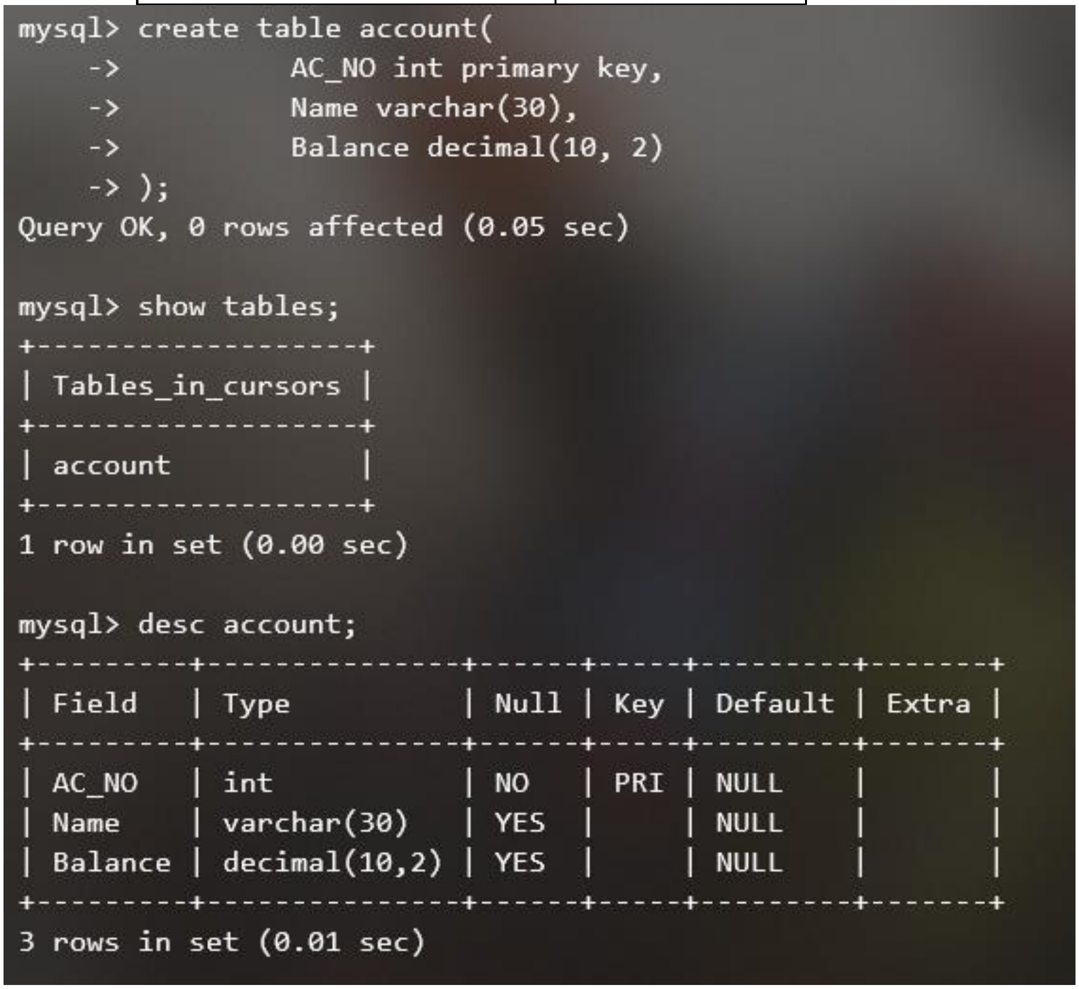
The finished is a variable to indicate that the cursor has reached the end of the result set. Notice that the handler declaration must appear after variable and cursor declaration inside the stored procedures. *(Cursor is very time consuming and not much efficient though)*

Now start your lab

Use DBMS2020 for all problems.

1. Create a table account

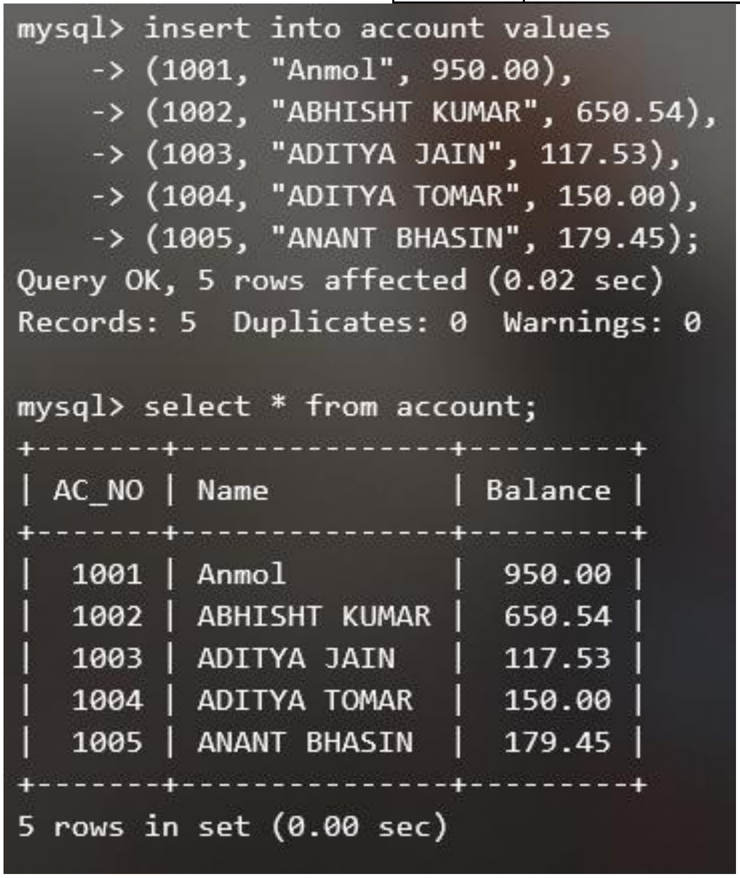
Attribute(Field/Column) Name	**Data Declaration**
AC_NO	Int primary key
Name	Varchar(30)
Balance	Numeric



```
mysql> create table account(
    ->          AC_NO int primary key,
    ->          Name varchar(30),
    ->          Balance decimal(10, 2)
    -> );
Query OK, 0 rows affected (0.05 sec)

mysql> show tables;
+-------------------+
| Tables_in_cursors |
+-------------------+
| account           |
+-------------------+
1 row in set (0.00 sec)

mysql> desc account;
+---------+---------------+------+-----+---------+-------+
| Field   | Type          | Null | Key | Default | Extra |
+---------+---------------+------+-----+---------+-------+
| AC_NO   | int           | NO   | PRI | NULL    |       |
| Name    | varchar(30)   | YES  |     | NULL    |       |
| Balance | decimal(10,2) | YES  |     | NULL    |       |
+---------+---------------+------+-----+---------+-------+
3 rows in set (0.01 sec)
```

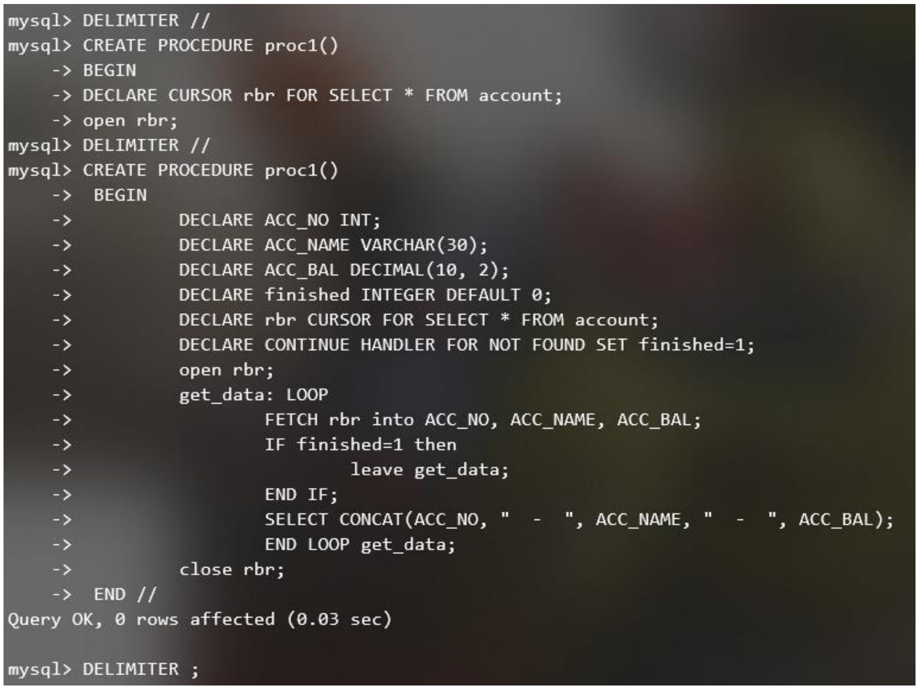
2. Insert below mention data to the table.

AC_NO	Name	Balance
1001	Anmol	950.50
1002	ABHISHT KUMAR	650.24
1003	ADITYA JAIN	117.53
1004	ADITYA TOMAR	150.00
1005	ANANT BHASIN	179.45

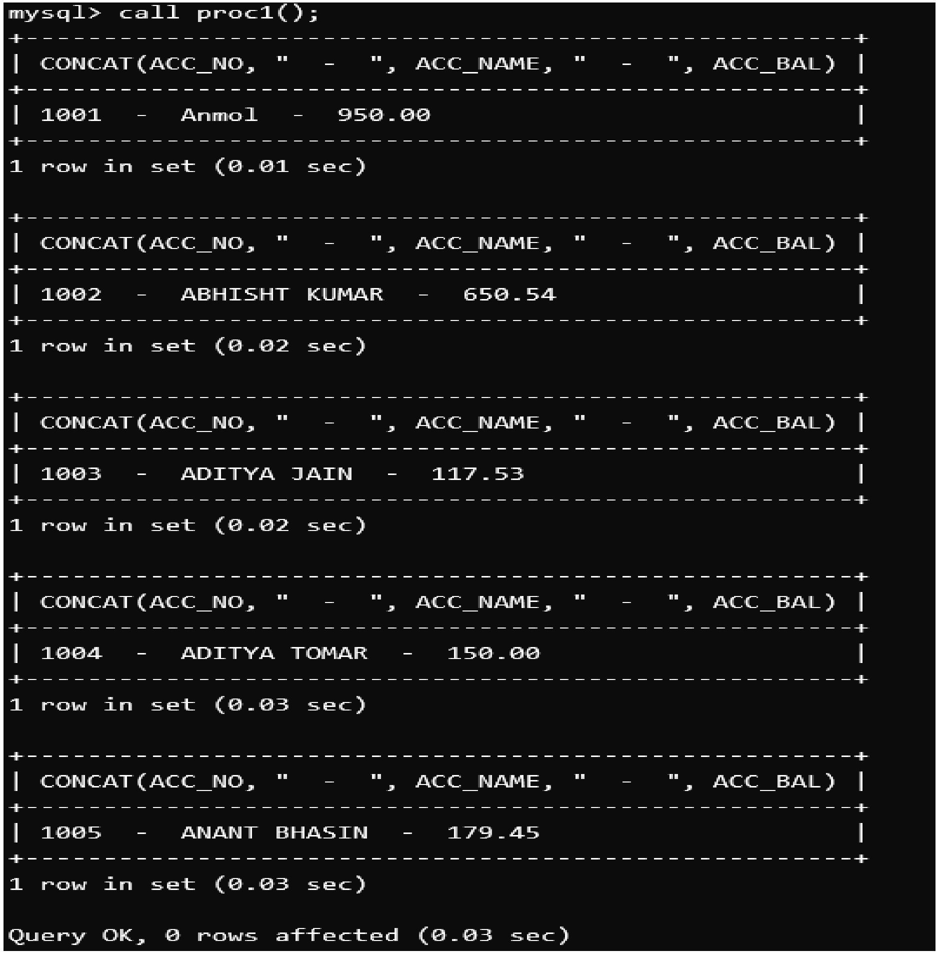


```
mysql> insert into account values
    -> (1001, "Anmol", 950.00),
    -> (1002, "ABHISHT KUMAR", 650.54),
    -> (1003, "ADITYA JAIN", 117.53),
    -> (1004, "ADITYA TOMAR", 150.00),
    -> (1005, "ANANT BHASIN", 179.45);
Query OK, 5 rows affected (0.02 sec)
Records: 5  Duplicates: 0  Warnings: 0

mysql> select * from account;
+-------+---------------+---------+
| AC_NO | Name          | Balance |
+-------+---------------+---------+
|  1001 | Anmol         |  950.00 |
|  1002 | ABHISHT KUMAR |  650.54 |
|  1003 | ADITYA JAIN   |  117.53 |
|  1004 | ADITYA TOMAR  |  150.00 |
|  1005 | ANANT BHASIN  |  179.45 |
+-------+---------------+---------+
5 rows in set (0.00 sec)
```

3. Write a PL/SQL block to Create an explicit cursor to display the details of all customer's **ROW BY ROW**

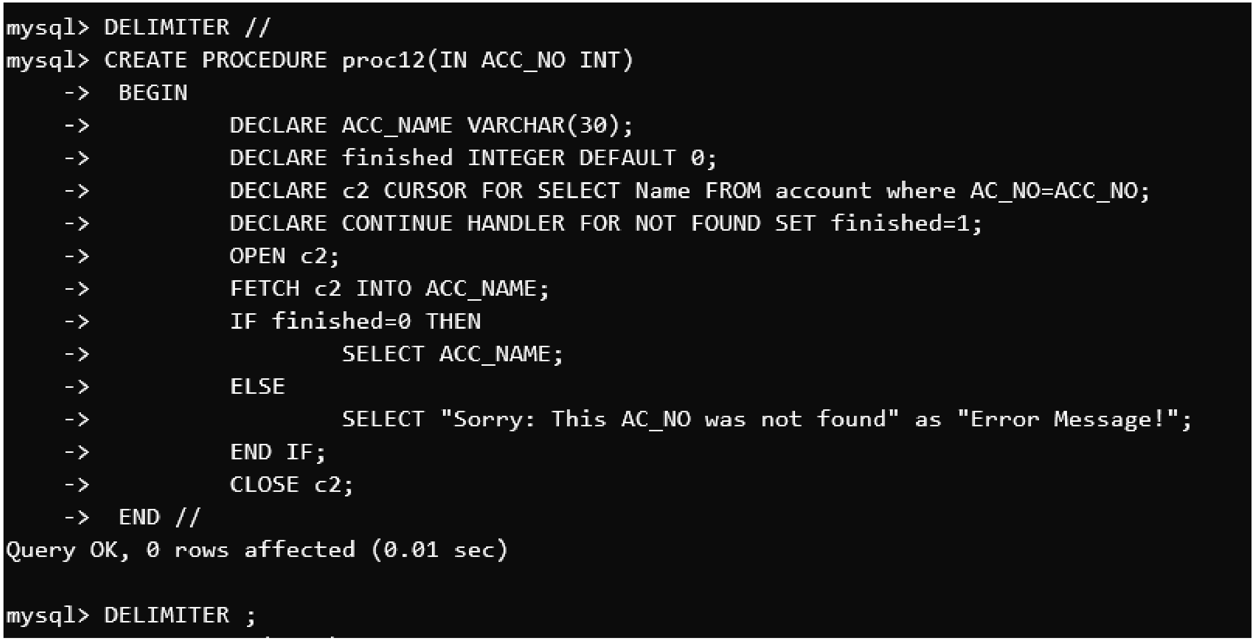


```
mysql> DELIMITER //
mysql> CREATE PROCEDURE proc1()
    -> BEGIN
    -> DECLARE CURSOR rbr FOR SELECT * FROM account;
    -> open rbr;
mysql> DELIMITER //
mysql> CREATE PROCEDURE proc1()
    ->  BEGIN
    ->          DECLARE ACC_NO INT;
    ->          DECLARE ACC_NAME VARCHAR(30);
    ->          DECLARE ACC_BAL DECIMAL(10, 2);
    ->          DECLARE finished INTEGER DEFAULT 0;
    ->          DECLARE rbr CURSOR FOR SELECT * FROM account;
    ->          DECLARE CONTINUE HANDLER FOR NOT FOUND SET finished=1;
    ->          open rbr;
    ->          get_data: LOOP
    ->                  FETCH rbr into ACC_NO, ACC_NAME, ACC_BAL;
    ->                  IF finished=1 then
    ->                          leave get_data;
    ->                  END IF;
    ->                  SELECT CONCAT(ACC_NO, "  -  ", ACC_NAME, "  -  ", ACC_BAL);
    ->                  END LOOP get_data;
    ->          close rbr;
    ->  END //
Query OK, 0 rows affected (0.03 sec)

mysql> DELIMITER ;
```

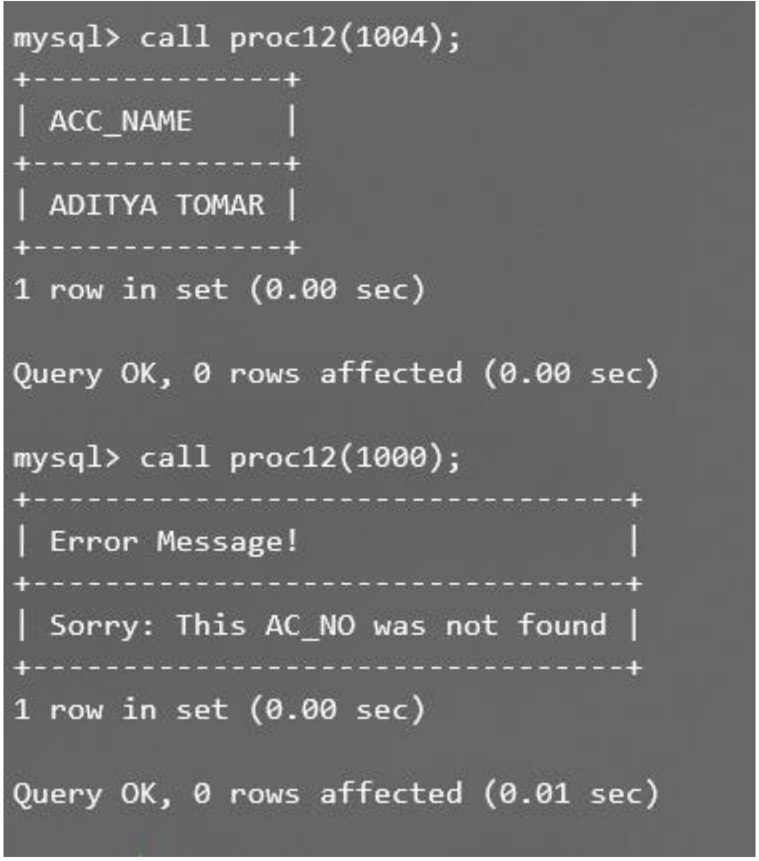


```
mysql> call proc1();
+----------------------------------------------------+
| CONCAT(ACC_NO, "  -  ", ACC_NAME, "  -  ", ACC_BAL) |
+----------------------------------------------------+
| 1001  -  Anmol  -  950.00                           |
+----------------------------------------------------+
1 row in set (0.01 sec)

+----------------------------------------------------+
| CONCAT(ACC_NO, "  -  ", ACC_NAME, "  -  ", ACC_BAL) |
+----------------------------------------------------+
| 1002  -  ABHISHT KUMAR  -  650.54                   |
+----------------------------------------------------+
1 row in set (0.02 sec)

+----------------------------------------------------+
| CONCAT(ACC_NO, "  -  ", ACC_NAME, "  -  ", ACC_BAL) |
+----------------------------------------------------+
| 1003  -  ADITYA JAIN  -  117.53                     |
+----------------------------------------------------+
1 row in set (0.02 sec)

+----------------------------------------------------+
| CONCAT(ACC_NO, "  -  ", ACC_NAME, "  -  ", ACC_BAL) |
+----------------------------------------------------+
| 1004  -  ADITYA TOMAR  -  150.00                    |
+----------------------------------------------------+
1 row in set (0.03 sec)

+----------------------------------------------------+
| CONCAT(ACC_NO, "  -  ", ACC_NAME, "  -  ", ACC_BAL) |
+----------------------------------------------------+
| 1005  -  ANANT BHASIN  -  179.45                    |
+----------------------------------------------------+
1 row in set (0.03 sec)

Query OK, 0 rows affected (0.03 sec)
```

4. Write a PL/SQL block to Create an explicit cursor to display the name by passing the AC_NO, if given AC_NO does not exist in the record, it will just display an Error Message “Sorry: This AC_NO was not found”. **(FOR NOT FOUND scenario)**



```
mysql> DELIMITER //
mysql> CREATE PROCEDURE proc12(IN ACC_NO INT)
    ->  BEGIN
    ->          DECLARE ACC_NAME VARCHAR(30);
    ->          DECLARE finished INTEGER DEFAULT 0;
    ->          DECLARE c2 CURSOR FOR SELECT Name FROM account where AC_NO=ACC_NO;
    ->          DECLARE CONTINUE HANDLER FOR NOT FOUND SET finished=1;
    ->          OPEN c2;
    ->          FETCH c2 INTO ACC_NAME;
    ->          IF finished=0 THEN
    ->                  SELECT ACC_NAME;
    ->          ELSE
    ->                  SELECT "Sorry: This AC_NO was not found" as "Error Message!";
    ->          END IF;
    ->          CLOSE c2;
    ->  END //
Query OK, 0 rows affected (0.01 sec)

mysql> DELIMITER ;
```



```
mysql> call proc12(1004);
+--------------+
| ACC_NAME     |
+--------------+
| ADITYA TOMAR |
+--------------+
1 row in set (0.00 sec)

Query OK, 0 rows affected (0.00 sec)

mysql> call proc12(1000);
+---------------------------------+
| Error Message!                  |
+---------------------------------+
| Sorry: This AC_NO was not found |
+---------------------------------+
1 row in set (0.00 sec)

Query OK, 0 rows affected (0.01 sec)
```

Transaction:

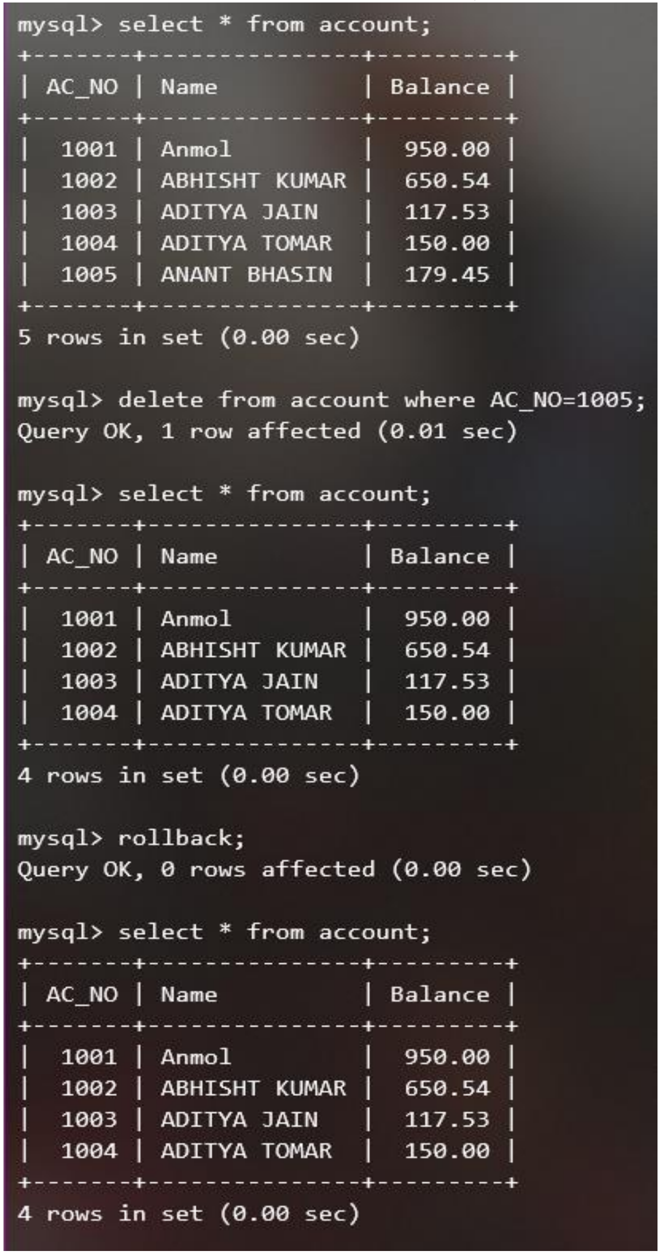
Transaction is very important concept, before starting the transaction you must be aware of **ACID** properties. I am briefing the acid properties, (though you have done in the class).

A- Atomicity: - All or nothing.
C- Consistency: - No constraint violated (Enforce all the integrity and data constraints).
I- Isolation: - Users (Session's) don't affect each other.
D- Durability: - Once data is committed, it is permanent.

A DBMS that supports all four is said to be **"ACID-Compliant".**

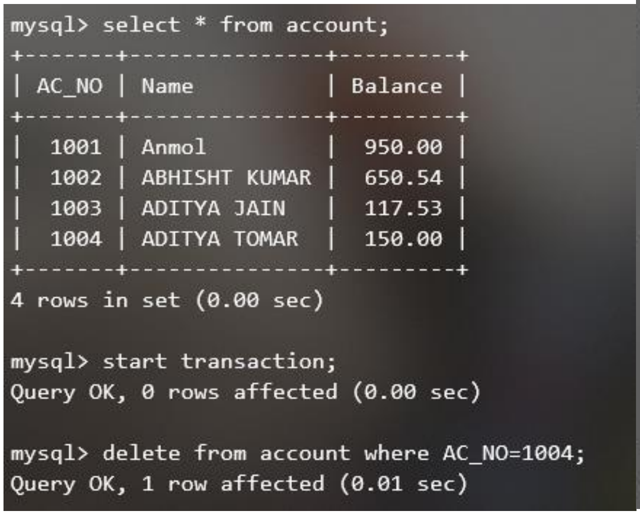
Now perform some operations:-

1. Delete one record from account table (without starting the transaction) and **rollback** it.

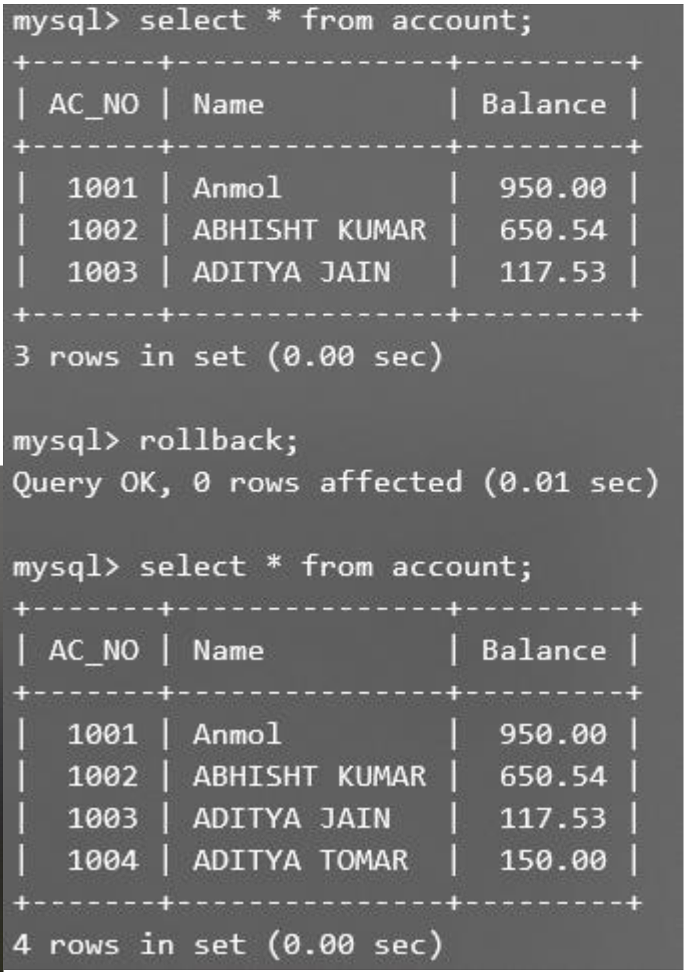


```
mysql> select * from account;
+-------+--------------+---------+
| AC_NO | Name         | Balance |
+-------+--------------+---------+
|  1001 | Anmol        |  950.00 |
|  1002 | ABHISHT KUMAR|  650.54 |
|  1003 | ADITYA JAIN  |  117.53 |
|  1004 | ADITYA TOMAR |  150.00 |
|  1005 | ANANT BHASIN |  179.45 |
+-------+--------------+---------+
5 rows in set (0.00 sec)

mysql> delete from account where AC_NO=1005;
Query OK, 1 row affected (0.01 sec)

mysql> select * from account;
+-------+--------------+---------+
| AC_NO | Name         | Balance |
+-------+--------------+---------+
|  1001 | Anmol        |  950.00 |
|  1002 | ABHISHT KUMAR|  650.54 |
|  1003 | ADITYA JAIN  |  117.53 |
|  1004 | ADITYA TOMAR |  150.00 |
+-------+--------------+---------+
4 rows in set (0.00 sec)

mysql> rollback;
Query OK, 0 rows affected (0.00 sec)

mysql> select * from account;
+-------+--------------+---------+
| AC_NO | Name         | Balance |
+-------+--------------+---------+
|  1001 | Anmol        |  950.00 |
|  1002 | ABHISHT KUMAR|  650.54 |
|  1003 | ADITYA JAIN  |  117.53 |
|  1004 | ADITYA TOMAR |  150.00 |
+-------+--------------+---------+
4 rows in set (0.00 sec)
```

2. Start the transaction & delete one record and **rollback** it.

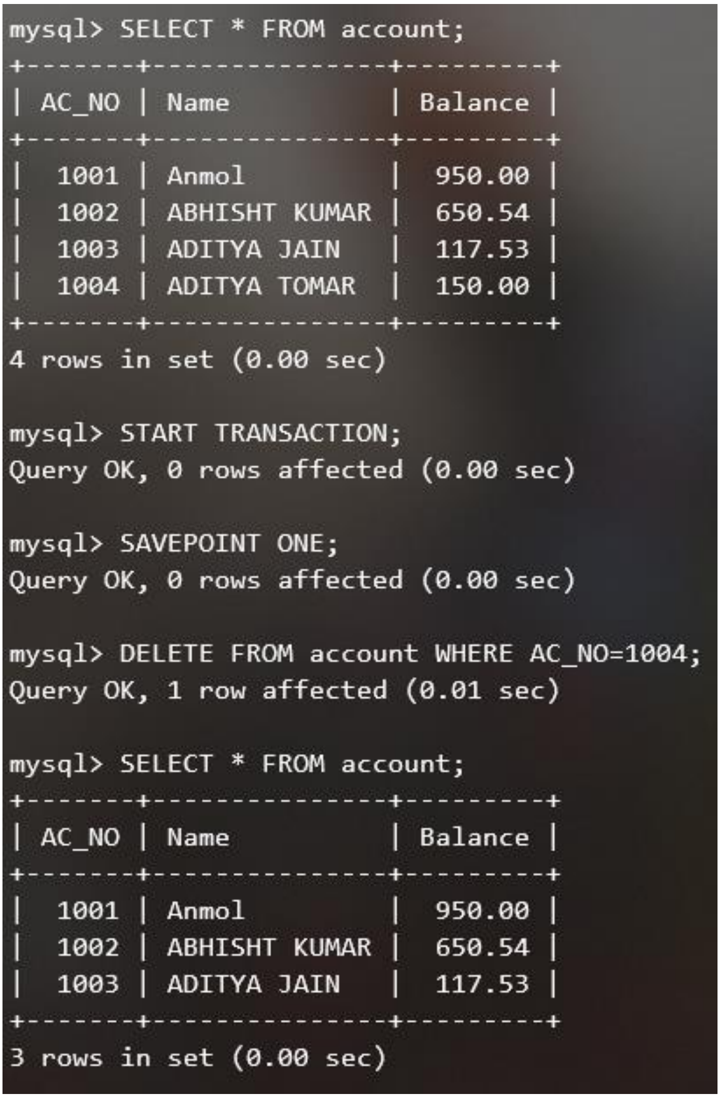


```
mysql> select * from account;
+-------+--------------+---------+
| AC_NO | Name         | Balance |
+-------+--------------+---------+
|  1001 | Anmol        |  950.00 |
|  1002 | ABHISHT KUMAR |  650.54 |
|  1003 | ADITYA JAIN  |  117.53 |
|  1004 | ADITYA TOMAR |  150.00 |
+-------+--------------+---------+
4 rows in set (0.00 sec)

mysql> start transaction;
Query OK, 0 rows affected (0.00 sec)

mysql> delete from account where AC_NO=1004;
Query OK, 1 row affected (0.01 sec)
```

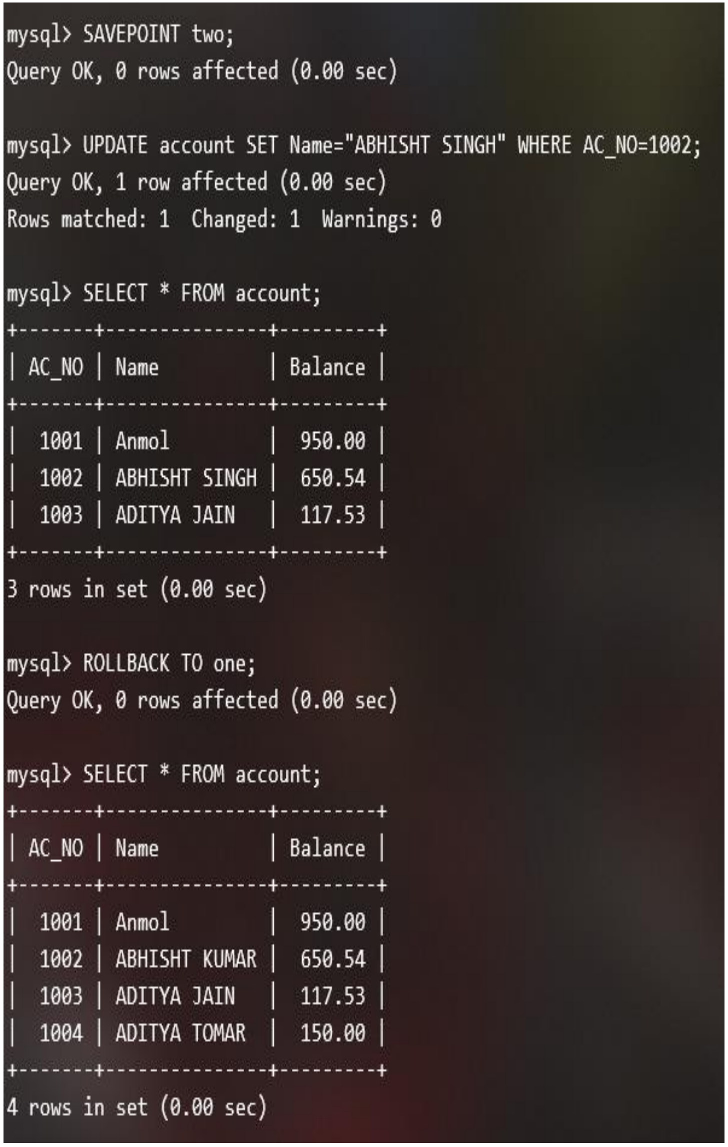


```
mysql> select * from account;
+-------+--------------+---------+
| AC_NO | Name         | Balance |
+-------+--------------+---------+
|  1001 | Anmol        |  950.00 |
|  1002 | ABHISHT KUMAR |  650.54 |
|  1003 | ADITYA JAIN  |  117.53 |
+-------+--------------+---------+
3 rows in set (0.00 sec)

mysql> rollback;
Query OK, 0 rows affected (0.01 sec)

mysql> select * from account;
+-------+--------------+---------+
| AC_NO | Name         | Balance |
+-------+--------------+---------+
|  1001 | Anmol        |  950.00 |
|  1002 | ABHISHT KUMAR |  650.54 |
|  1003 | ADITYA JAIN  |  117.53 |
|  1004 | ADITYA TOMAR |  150.00 |
+-------+--------------+---------+
4 rows in set (0.00 sec)
```

3. **Use Save points**: - Start transaction update one record save this change to savepoint1, insert new record & save this change to savepoint2 and so on…. Now rollback to a particular part of transaction.



```
mysql> SELECT * FROM account;
+-------+--------------+---------+
| AC_NO | Name         | Balance |
+-------+--------------+---------+
|  1001 | Anmol        |  950.00 |
|  1002 | ABHISHT KUMAR |  650.54 |
|  1003 | ADITYA JAIN  |  117.53 |
|  1004 | ADITYA TOMAR |  150.00 |
+-------+--------------+---------+
4 rows in set (0.00 sec)

mysql> START TRANSACTION;
Query OK, 0 rows affected (0.00 sec)

mysql> SAVEPOINT ONE;
Query OK, 0 rows affected (0.00 sec)

mysql> DELETE FROM account WHERE AC_NO=1004;
Query OK, 1 row affected (0.01 sec)

mysql> SELECT * FROM account;
+-------+--------------+---------+
| AC_NO | Name         | Balance |
+-------+--------------+---------+
|  1001 | Anmol        |  950.00 |
|  1002 | ABHISHT KUMAR |  650.54 |
|  1003 | ADITYA JAIN  |  117.53 |
+-------+--------------+---------+
3 rows in set (0.00 sec)
```

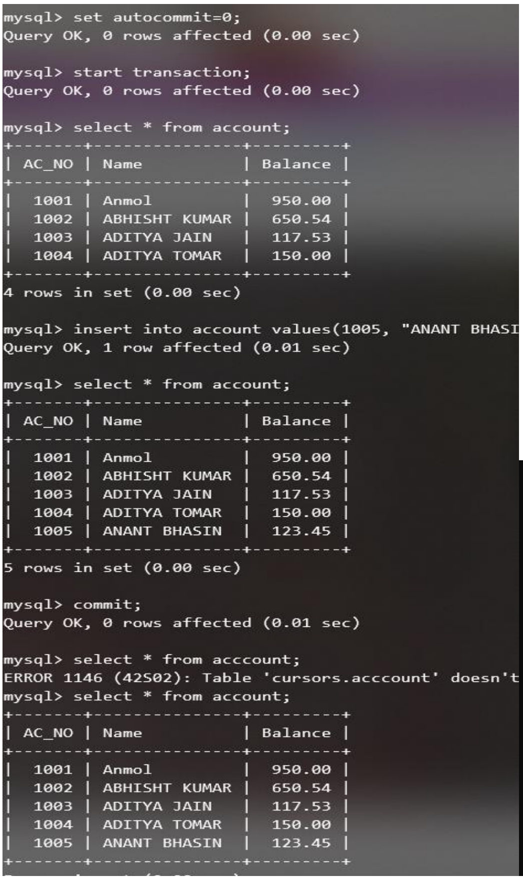


```
mysql> SAVEPOINT two;
Query OK, 0 rows affected (0.00 sec)

mysql> UPDATE account SET Name="ABHISHT SINGH" WHERE AC_NO=1002;
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> SELECT * FROM account;
+-------+--------------+---------+
| AC_NO | Name         | Balance |
+-------+--------------+---------+
|  1001 | Anmol        |  950.00 |
|  1002 | ABHISHT SINGH |  650.54 |
|  1003 | ADITYA JAIN  |  117.53 |
+-------+--------------+---------+
3 rows in set (0.00 sec)

mysql> ROLLBACK TO one;
Query OK, 0 rows affected (0.00 sec)

mysql> SELECT * FROM account;
+-------+--------------+---------+
| AC_NO | Name         | Balance |
+-------+--------------+---------+
|  1001 | Anmol        |  950.00 |
|  1002 | ABHISHT KUMAR |  650.54 |
|  1003 | ADITYA JAIN  |  117.53 |
|  1004 | ADITYA TOMAR |  150.00 |
+-------+--------------+---------+
4 rows in set (0.00 sec)
```

4. Now open one more mysql command line window. Access the same table in both the command line screen's and execute below mention statements(Make auto-commit off in both the screen's & don't forget to start transaction every time in every screen/session):

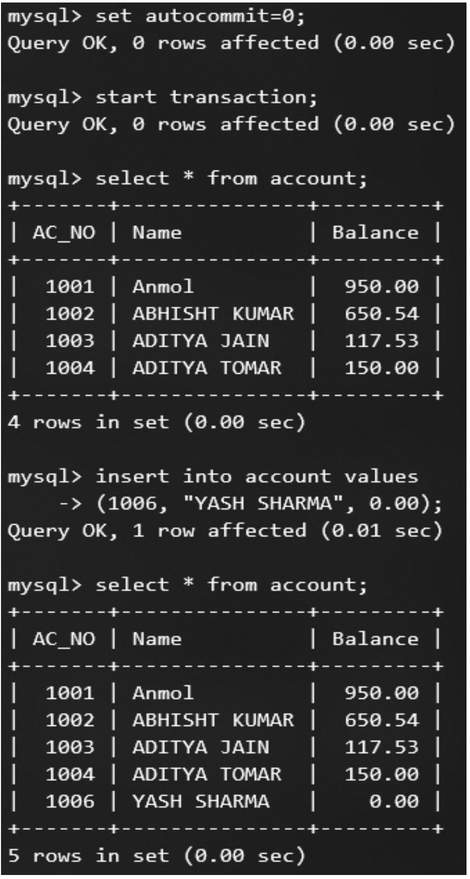
   i. Insert a new record in window 1, Insert another **(new)** record in window 2 & display the table before committing and after committing in each window.

WINDOW – 1

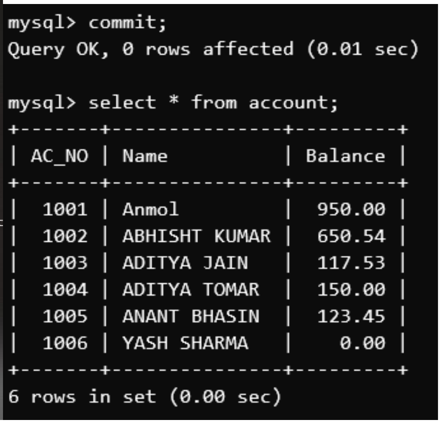


```
mysql> set autocommit=0;
Query OK, 0 rows affected (0.00 sec)

mysql> start transaction;
Query OK, 0 rows affected (0.00 sec)

mysql> select * from account;
+-------+---------------+---------+
| AC_NO | Name          | Balance |
+-------+---------------+---------+
|  1001 | Anmol         |  950.00 |
|  1002 | ABHISHT KUMAR |  650.54 |
|  1003 | ADITYA JAIN   |  117.53 |
|  1004 | ADITYA TOMAR  |  150.00 |
+-------+---------------+---------+
4 rows in set (0.00 sec)

mysql> insert into account values(1005, "ANANT BHASI
Query OK, 1 row affected (0.01 sec)

mysql> select * from account;
+-------+---------------+---------+
| AC_NO | Name          | Balance |
+-------+---------------+---------+
|  1001 | Anmol         |  950.00 |
|  1002 | ABHISHT KUMAR |  650.54 |
|  1003 | ADITYA JAIN   |  117.53 |
|  1004 | ADITYA TOMAR  |  150.00 |
|  1005 | ANANT BHASIN  |  123.45 |
+-------+---------------+---------+
5 rows in set (0.00 sec)

mysql> commit;
Query OK, 0 rows affected (0.01 sec)

mysql> select * from acccount;
ERROR 1146 (42S02): Table 'cursors.acccount' doesn't
mysql> select * from account;
+-------+---------------+---------+
| AC_NO | Name          | Balance |
+-------+---------------+---------+
|  1001 | Anmol         |  950.00 |
|  1002 | ABHISHT KUMAR |  650.54 |
|  1003 | ADITYA JAIN   |  117.53 |
|  1004 | ADITYA TOMAR  |  150.00 |
|  1005 | ANANT BHASIN  |  123.45 |
+-------+---------------+---------+
```

WINDOW – 2 (2 SCREENSHOTS)



```
mysql> set autocommit=0;
Query OK, 0 rows affected (0.00 sec)

mysql> start transaction;
Query OK, 0 rows affected (0.00 sec)

mysql> select * from account;
+-------+---------------+---------+
| AC_NO | Name          | Balance |
+-------+---------------+---------+
|  1001 | Anmol         |  950.00 |
|  1002 | ABHISHT KUMAR |  650.54 |
|  1003 | ADITYA JAIN   |  117.53 |
|  1004 | ADITYA TOMAR  |  150.00 |
+-------+---------------+---------+
4 rows in set (0.00 sec)

mysql> insert into account values
    -> (1006, "YASH SHARMA", 0.00);
Query OK, 1 row affected (0.01 sec)

mysql> select * from account;
+-------+---------------+---------+
| AC_NO | Name          | Balance |
+-------+---------------+---------+
|  1001 | Anmol         |  950.00 |
|  1002 | ABHISHT KUMAR |  650.54 |
|  1003 | ADITYA JAIN   |  117.53 |
|  1004 | ADITYA TOMAR  |  150.00 |
|  1006 | YASH SHARMA   |    0.00 |
+-------+---------------+---------+
5 rows in set (0.00 sec)
```

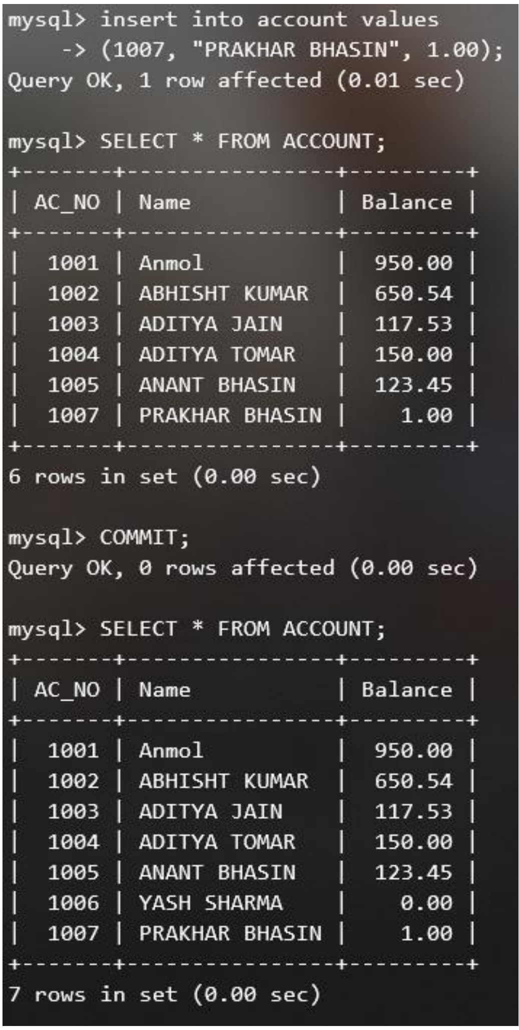


```
mysql> commit;
Query OK, 0 rows affected (0.01 sec)

mysql> select * from account;
+-------+---------------+---------+
| AC_NO | Name          | Balance |
+-------+---------------+---------+
|  1001 | Anmol         |  950.00 |
|  1002 | ABHISHT KUMAR |  650.54 |
|  1003 | ADITYA JAIN   |  117.53 |
|  1004 | ADITYA TOMAR  |  150.00 |
|  1005 | ANANT BHASIN  |  123.45 |
|  1006 | YASH SHARMA   |    0.00 |
+-------+---------------+---------+
6 rows in set (0.00 sec)
```

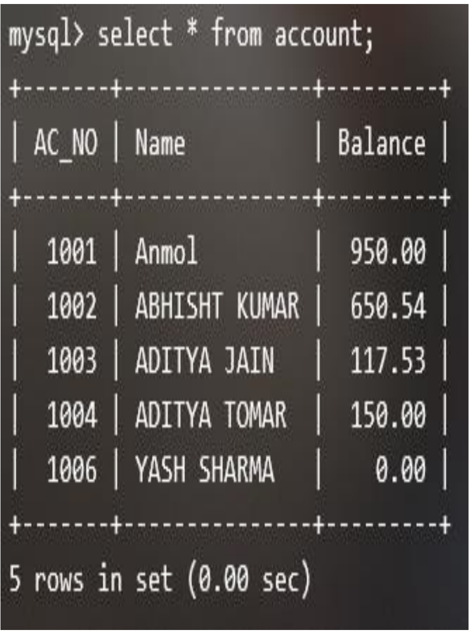
ii. Insert a new record in window 1, Insert the **same** record to the table in window 2 & display the table before committing and after committing in each window.

WINDOW – 1

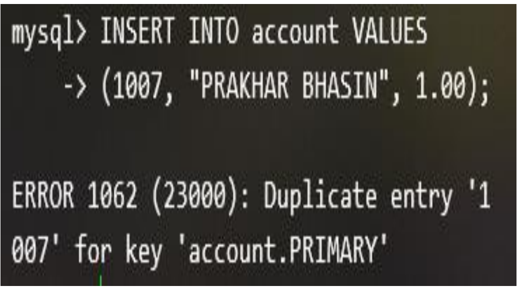
WINDOW – 2 BELOW (2 SCREENSHOTS)



```
mysql> insert into account values
    -> (1007, "PRAKHAR BHASIN", 1.00);
Query OK, 1 row affected (0.01 sec)

mysql> SELECT * FROM ACCOUNT;
+-------+----------------+---------+
| AC_NO | Name           | Balance |
+-------+----------------+---------+
|  1001 | Anmol          |  950.00 |
|  1002 | ABHISHT KUMAR  |  650.54 |
|  1003 | ADITYA JAIN    |  117.53 |
|  1004 | ADITYA TOMAR   |  150.00 |
|  1005 | ANANT BHASIN   |  123.45 |
|  1007 | PRAKHAR BHASIN |    1.00 |
+-------+----------------+---------+
6 rows in set (0.00 sec)

mysql> COMMIT;
Query OK, 0 rows affected (0.00 sec)

mysql> SELECT * FROM ACCOUNT;
+-------+----------------+---------+
| AC_NO | Name           | Balance |
+-------+----------------+---------+
|  1001 | Anmol          |  950.00 |
|  1002 | ABHISHT KUMAR  |  650.54 |
|  1003 | ADITYA JAIN    |  117.53 |
|  1004 | ADITYA TOMAR   |  150.00 |
|  1005 | ANANT BHASIN   |  123.45 |
|  1006 | YASH SHARMA    |    0.00 |
|  1007 | PRAKHAR BHASIN |    1.00 |
+-------+----------------+---------+
7 rows in set (0.00 sec)
```



```
mysql> select * from account;
+-------+---------------+---------+
| AC_NO | Name          | Balance |
+-------+---------------+---------+
|  1001 | Anmol         |  950.00 |
|  1002 | ABHISHT KUMAR |  650.54 |
|  1003 | ADITYA JAIN   |  117.53 |
|  1004 | ADITYA TOMAR  |  150.00 |
|  1006 | YASH SHARMA   |    0.00 |
+-------+---------------+---------+
5 rows in set (0.00 sec)
```



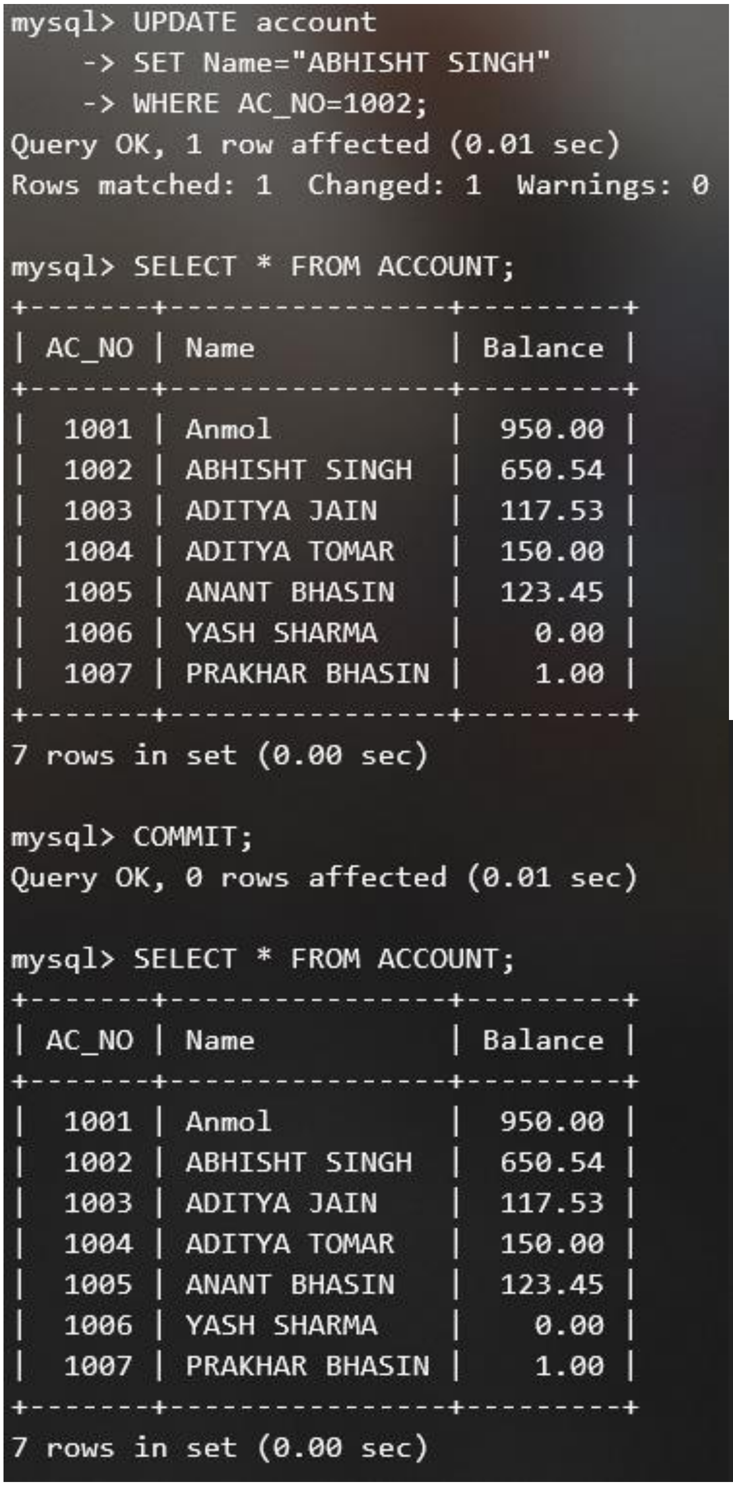
```
mysql> INSERT INTO account VALUES
    -> (1007, "PRAKHAR BHASIN", 1.00);

ERROR 1062 (23000): Duplicate entry '1
007' for key 'account.PRIMARY'
```

iii. Update any record from table in window1 and update the same record in window2 at the same time (before commit in window 1).

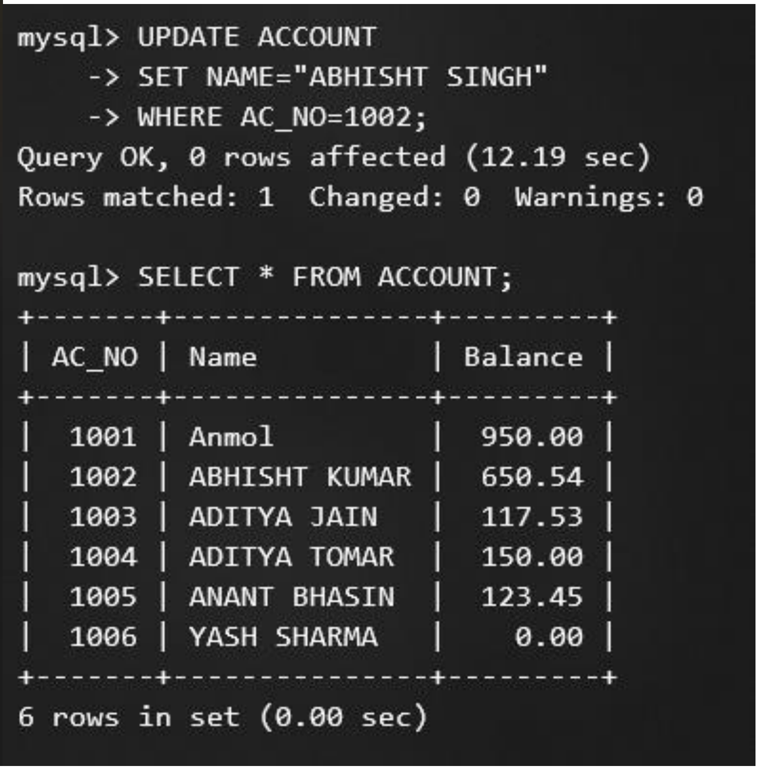
WINDOW – 1

WINDOW – 2

ANS – BEFORE COMMITTING ON WINDOW 1 THE COMMAND IN WINDOW 2 WOULD NOT EXECUTE AND AFTER COMMITTING THE CHANGE DID NOT TAKE PLACE

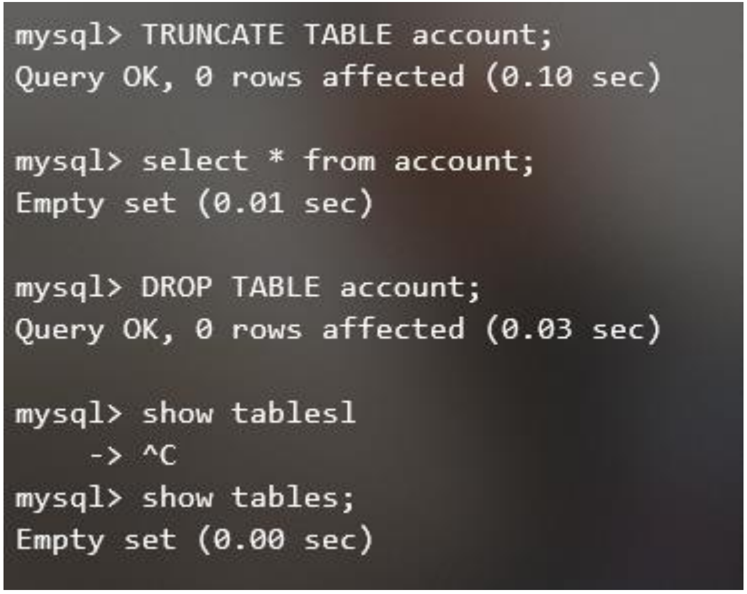


```
mysql> UPDATE account
    -> SET Name="ABHISHT SINGH"
    -> WHERE AC_NO=1002;
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> SELECT * FROM ACCOUNT;
+-------+---------------+---------+
| AC_NO | Name          | Balance |
+-------+---------------+---------+
|  1001 | Anmol         |  950.00 |
|  1002 | ABHISHT SINGH |  650.54 |
|  1003 | ADITYA JAIN   |  117.53 |
|  1004 | ADITYA TOMAR  |  150.00 |
|  1005 | ANANT BHASIN  |  123.45 |
|  1006 | YASH SHARMA   |    0.00 |
|  1007 | PRAKHAR BHASIN |    1.00 |
+-------+---------------+---------+
7 rows in set (0.00 sec)

mysql> COMMIT;
Query OK, 0 rows affected (0.01 sec)

mysql> SELECT * FROM ACCOUNT;
+-------+---------------+---------+
| AC_NO | Name          | Balance |
+-------+---------------+---------+
|  1001 | Anmol         |  950.00 |
|  1002 | ABHISHT SINGH |  650.54 |
|  1003 | ADITYA JAIN   |  117.53 |
|  1004 | ADITYA TOMAR  |  150.00 |
|  1005 | ANANT BHASIN  |  123.45 |
|  1006 | YASH SHARMA   |    0.00 |
|  1007 | PRAKHAR BHASIN |    1.00 |
+-------+---------------+---------+
7 rows in set (0.00 sec)
```



```
mysql> UPDATE ACCOUNT
    -> SET NAME="ABHISHT SINGH"
    -> WHERE AC_NO=1002;
Query OK, 0 rows affected (12.19 sec)
Rows matched: 1  Changed: 0  Warnings: 0

mysql> SELECT * FROM ACCOUNT;
+-------+---------------+---------+
| AC_NO | Name          | Balance |
+-------+---------------+---------+
|  1001 | Anmol         |  950.00 |
|  1002 | ABHISHT KUMAR |  650.54 |
|  1003 | ADITYA JAIN   |  117.53 |
|  1004 | ADITYA TOMAR  |  150.00 |
|  1005 | ANANT BHASIN  |  123.45 |
|  1006 | YASH SHARMA   |    0.00 |
+-------+---------------+---------+
6 rows in set (0.00 sec)
```

5. Now execute some queries to **Truncate & Drop** the table in transaction block (*Now you will be able to identify the difference in DROP, DELETE, & TRUNCATE)*



```
mysql> TRUNCATE TABLE account;
Query OK, 0 rows affected (0.10 sec)

mysql> select * from account;
Empty set (0.01 sec)

mysql> DROP TABLE account;
Query OK, 0 rows affected (0.03 sec)

mysql> show tablesl
    -> ^C
mysql> show tables;
Empty set (0.00 sec)
```

Drop all tables under DBMS2020