Part I: Demonstration for ROLLBACK & COMMIT

Before the practices of ROLLBACK and COMMIT, let us figure out the system variable – AUTOCOMMIT. A reference is the Cookbook Recipe 8.2 Step 2: "How to change the values of server system variables with SET statement?"

In this document, we use MySQL command to change the variable values. Note: The default value of AUTOCOMMIT is ON. For our practice, we need to change AUTOCOMMIT value every time when we connect to MySQL.

Recipe 8.2

How to access the server system variables ?

Step 2 How to change the values of server system variables with SET statement ?

Dynamic server system variables can be change with `SET GLOBAL` or `SET SESSION` commands.

In this set we shall examine a value of `autocommit` variable and we shall set its `SESSION` value to `1`. It means that each modification will be automatically committed immediatelly after every data manipulation operation.

First, use a statement `SHOW VARIABLES LIKE 'autocommit';` to find what is the present value of the variable.

```
SHOW VARIABLES LIKE 'autocommit';
```

The results are the following.

```
+---------------+-------+
| Variable_name | Value |
+---------------+-------+
| autocommit    | ON    |
+---------------+-------+
1 row in set (0.05 sec)
```

As expected, the present value of `autocommit` variable is `ON` It means that modifications are automatically committed immediately after each data manipulation statement without processing of `COMMIT` statement.

With the variable 'AUTOCOMMIT' set to the value of ON, each statement processed by MySQL will COMMIT immediately and automatically. You can have a try to make modifications to an existing relational table in database CSIT882. A demonstration is given below by using the example database for lecture classes.

Please connect as the user csit115 and use the database csit115, if you do not have csit882 in your system.

Practice 1: see if ROLLBACK works when the variable AUTOCOMMIT = ON.

Step 1, log in MySQL as a root user from the operation system terminal.

```
mysql –u root –p –v –c
```

Step 2, use a database csit882 (or csit115). We will use csit882 for demonstration.

```
use csit882;
```

Step 3, show current tables.(Suppose you have created the tables then loaded data correctly.)

```
show tables;
```

```
+-------------------+
| Tables_in_csit882 |
+-------------------+
| COURSE            |
| DEPARTMENT        |
| EMPLOYEE          |
| ENROLMENT         |
| STUDENT           |
+-------------------+
```

If your relational tables differ from the sample database shown above, process dbdrop_deptCourse.sql then dbcreate_deptCourse.sql to refresh the database.

Step 4, change variable AUTOCOMMIT to ON,

```
SET SESSION AUTOCOMMIT = 'ON';
```

and double check its value is ON.

```
show variables like 'AUTOCOMMIT';
```

```
+---------------+-------+
| Variable_name | Value |
+---------------+-------+
| autocommit    | ON    |
+---------------+-------+
```

Step 5, choose a relational table to make a few changes with SQL statement. Here we choose DEPARTMENT to insert a new row, then try to use rollback to reverse the processing.

```
SELECT * FROM DEPARTMENT;
```

```
+----------------------+------+-------------+---------+------------+
| name                 | code | total_staff | chair   | budget     |
+----------------------+------+-------------+---------+------------+
| School of Accounting | SACC |          40 | Jennie  |       NULL |
| School of Arts       | SA   |          30 | NULL    |  7000000.0 |
| School of CIT        | SCIT |          40 | Willy   | 10000000.0 |
| School of Mathematics | SM  |          15 | David   |       NULL |
| School of Physics    | SP   |          20 | Peter   |  5000000.0 |
+----------------------+------+-------------+---------+------------+
```

```
INSERT INTO DEPARTMENT VALUES ('Test', 'ST', 1, NULL, NULL);
ROLLBACK;
SELECT * FROM DEPARTMENT;
```

```
+----------------------+------+-------------+--------+-------------+
| name                 | code | total_staff | chair  | budget      |
+----------------------+------+-------------+--------+-------------+
| School of Accounting | SACC |          40 | Jennie |        NULL |
| School of Arts       | SA   |          30 | NULL   |   7000000.0 |
| School of CIT        | SCIT |          40 | Willy  | 10000000.0 |
| School of Mathematics | SM   |          15 | David  |        NULL |
| School of Physics    | SP   |          20 | Peter  |   5000000.0 |
| Test                 | ST   |           1 | NULL   |        NULL |
+----------------------+------+-------------+--------+-------------+
```

From this practice, we can see ROLLBACK does not work, because AUTOCOMMIT is ON. Then, we change the DEPARTMENT table back to the original status by processing the following statement for future practice.

```
DELETE FROM DEPARTMENT WHERE name = 'Test';
```

Practice 2: practice ROLLBACK when the variable AUTOCOMMIT = OFF.

Step 1, change variable AUTOCOMMIT to OFF,

```
SET SESSION AUTOCOMMIT = 'OFF';
```

and double check its value is OFF.

```
show variables like 'AUTOCOMMIT';
```

```
+---------------+-------+
| Variable_name | Value |
+---------------+-------+
| autocommit    | OFF   |
+---------------+-------+
```

Step 2, make modifications to csit882.DEPARTMENT again.

Step 2-1, display the current contents in DEPARTMENT table by processing the select statement and make sure the DEPARTMENT table looks like below:

```
SELECT * FROM DEPARTMENT;
+----------------------+------+-------------+--------+-------------+
| name                 | code | total_staff | chair  | budget      |
+----------------------+------+-------------+--------+-------------+
| School of Accounting | SACC |          40 | Jennie |        NULL |
| School of Arts       | SA   |          30 | NULL   |   7000000.0 |
| School of CIT        | SCIT |          40 | Willy  | 10000000.0 |
| School of Mathematics | SM   |          15 | David  |        NULL |
| School of Physics    | SP   |          20 | Peter  |   5000000.0 |
+----------------------+------+-------------+--------+-------------+
```

Step 2-2, insert a new row into DEPARTMENT table and display the table.

```
INSERT INTO DEPARTMENT VALUES ('Test', 'ST', 1, NULL, NULL);
SELECT * FROM DEPARTMENT;
+----------------------+------+-------------+--------+-------------+
| name                 | code | total_staff | chair  | budget      |
+----------------------+------+-------------+--------+-------------+
| School of Accounting | SACC |          40 | Jennie |        NULL |
```

```
| School of Arts         | SA   |          30 | NULL   |  7000000.0 |
| School of CIT          | SCIT |          40 | Willy  | 10000000.0 |
| School of Mathematics | SM   |          15 | David  |       NULL |
| School of Physics      | SP   |          20 | Peter  |  5000000.0 |
| Test                   | ST   |           1 | NULL   |       NULL |
+------------------------+------+-------------+--------+------------+
```

Step 2-3, process ROLLBACK and display the table again.

```
ROLLBACK;
SELECT * FROM DEPARTMENT;
```

```
+------------------------+------+-------------+--------+------------+
| name                   | code | total_staff | chair  | budget     |
+------------------------+------+-------------+--------+------------+
| School of Accounting   | SACC |          40 | Jennie |       NULL |
| School of Arts         | SA   |          30 | NULL   |  7000000.0 |
| School of CIT          | SCIT |          40 | Willy  | 10000000.0 |
| School of Mathematics | SM   |          15 | David  |       NULL |
| School of Physics      | SP   |          20 | Peter  |  5000000.0 |
+------------------------+------+-------------+--------+------------+
```

Now the table is rolled back to the original status and insertion has been reversed.

Practice 3: practice COMMIT when the variable AUTOCOMMIT = OFF.

Step 1, change variable AUTOCOMMIT to OFF,

```
SET SESSION AUTOCOMMIT = 'OFF';
```

and double check its value is OFF.

```
show variables like 'AUTOCOMMIT';
```

```
+---------------+-------+
| Variable_name | Value |
+---------------+-------+
| autocommit    | OFF   |
+---------------+-------+
```

Step 2, make modifications to csit882.DEPARTMENT again.

Step 2-1, display the current contents in DEPARTMENT table by processing the select statement and make sure the DEPARTMENT table looks like below:

```
SELECT * FROM DEPARTMENT;
```

```
+------------------------+------+-------------+--------+------------+
| name                   | code | total_staff | chair  | budget     |
+------------------------+------+-------------+--------+------------+
| School of Accounting   | SACC |          40 | Jennie |       NULL |
| School of Arts         | SA   |          30 | NULL   |  7000000.0 |
| School of CIT          | SCIT |          40 | Willy  | 10000000.0 |
| School of Mathematics | SM   |          15 | David  |       NULL |
| School of Physics      | SP   |          20 | Peter  |  5000000.0 |
+------------------------+------+-------------+--------+------------+
```

Step 2-2, insert a new row into DEPARTMENT table and display the table.

```
INSERT INTO DEPARTMENT VALUES ('Test', 'ST', 1, NULL, NULL);
SELECT * FROM DEPARTMENT;
```

```
+----------------------+------+-------------+--------+------------+
| name                 | code | total_staff | chair  | budget     |
+----------------------+------+-------------+--------+------------+
| School of Accounting | SACC |          40 | Jennie |       NULL |
| School of Arts       | SA   |          30 | NULL   |  7000000.0 |
| School of CIT        | SCIT |          40 | Willy  | 10000000.0 |
| School of Mathematics | SM  |          15 | David  |       NULL |
| School of Physics    | SP   |          20 | Peter  |  5000000.0 |
| Test                 | ST   |           1 | NULL   |       NULL |
+----------------------+------+-------------+--------+------------+
```

Step 2-3, process COMMIT and display the table.

```
COMMIT;
SELECT * FROM DEPARTMENT;
```

```
+----------------------+------+-------------+--------+------------+
| name                 | code | total_staff | chair  | budget     |
+----------------------+------+-------------+--------+------------+
| School of Accounting | SACC |          40 | Jennie |       NULL |
| School of Arts       | SA   |          30 | NULL   |  7000000.0 |
| School of CIT        | SCIT |          40 | Willy  | 10000000.0 |
| School of Mathematics | SM  |          15 | David  |       NULL |
| School of Physics    | SP   |          20 | Peter  |  5000000.0 |
| Test                 | ST   |           1 | NULL   |       NULL |
+----------------------+------+-------------+--------+------------+
```

Step 2-4, process ROLLBACK and display the table.

```
ROLLBACK;
SELECT * FROM DEPARTMENT;
```

```
+----------------------+------+-------------+--------+------------+
| name                 | code | total_staff | chair  | budget     |
+----------------------+------+-------------+--------+------------+
| School of Accounting | SACC |          40 | Jennie |       NULL |
| School of Arts       | SA   |          30 | NULL   |  7000000.0 |
| School of CIT        | SCIT |          40 | Willy  | 10000000.0 |
| School of Mathematics | SM  |          15 | David  |       NULL |
| School of Physics    | SP   |          20 | Peter  |  5000000.0 |
| Test                 | ST   |           1 | NULL   |       NULL |
+----------------------+------+-------------+--------+------------+
```

The result indicates that once a statement is committed by COMMIT, ROLLBACK does not work.

Then, we change the DEPARTMENT table back to the original status by processing the following statement for future practice.

```
DELETE FROM DEPARTMENT WHERE name = 'Test';
COMMIT;
```

Please connect as the user csit115 and use the database csit115, if you do not have csit882 in your system.

Part II: Demonstration for Transactions (For CSIT882)

A partially ordered set of read, write operations on the database items is called as a transaction. This demonstration shows some cases to simulate two transactions interleaving their operations. In the demonstration we use the example database for lecture classes which is available on CSIT882 Moodle site.

Step 1, open two separate terminal windows. log in as a root user in Terminal window 1, and log in as a csit882 user in Terminal window 2. Terminal 1 acts as transaction T1. Terminal 2 acts as transaction T2. We are simulating a situation where there are two different transactions T1 and T2.

```
mysql –u root –p –v –c
```

Step 2, use database csit882 as the default database in both transactions. We use csit882 for demonstration.

```
use csit882;
```

Step 3, show current tables. (Suppose you have created the tables then loaded data correctly.)

```
show tables;
```

```
+-------------------+
| Tables_in_csit882 |
+-------------------+
| COURSE            |
| DEPARTMENT        |
| EMPLOYEE          |
| ENROLMENT         |
| STUDENT           |
+-------------------+
```

If your relational tables differ from the sample database shown above, process dbdrop_deptCourse.sql then dbcreate_deptCourse.sql to refresh the database.

Step 4, choose a table to practice transactions, such as the relational table DEPARTMENT.

```
SELECT * FROM DEPARTMENT;
```

```
+----------------------+------+-------------+--------+------------+
| name                 | code | total_staff | chair  | budget     |
+----------------------+------+-------------+--------+------------+
| School of Accounting | SACC |          40 | Jennie |       NULL |
| School of Arts       | SA   |          30 | NULL   |  7000000.0 |
| School of CIT        | SCIT |          40 | Willy  | 10000000.0 |
| School of Mathematics | SM  |          15 | David  |       NULL |
| School of Physics    | SP   |          20 | Peter  |  5000000.0 |
+----------------------+------+-------------+--------+------------+
```

Step 5 (IMPORTANT), turn OFF autocommit in both terminals.

```
SET SESSION AUTOCOMMIT = 'OFF';
```

and double check the value is OFF.

```
show variables like 'AUTOCOMMIT';
```

```
+---------------+-------+
| Variable_name | Value |
+---------------+-------+
| autocommit    | OFF   |
+---------------+-------+
```

Step 6, practice transition cases below.

Case 1: T1 (autocommit = 'OFF') -> T2 (autocommit = 'OFF') -> T1 (commit) -> T2

T1:

```
INSERT INTO DEPARTMENT VALUES ('Test', 'ST', 1, NULL, NULL);
SELECT * FROM DEPARTMENT;
```

```
+----------------------+------+-------------+---------+------------+
| name                 | code | total_staff | chair   | budget     |
+----------------------+------+-------------+---------+------------+
| School of Accounting | SACC |          40 | Jennie  |       NULL |
| School of Arts       | SA   |          30 | NULL    |  7000000.0 |
| School of CIT        | SCIT |          40 | Willy   | 10000000.0 |
| School of Mathematics| SM   |          15 | David   |       NULL |
| School of Physics    | SP   |          20 | Peter   |  5000000.0 |
| Test                 | ST   |           1 | NULL    |       NULL |
+----------------------+------+-------------+---------+------------+
```

T2:

```
SELECT * FROM DEPARTMENT;
```

```
+----------------------+------+-------------+---------+------------+
| name                 | code | total_staff | chair   | budget     |
+----------------------+------+-------------+---------+------------+
| School of Accounting | SACC |          40 | Jennie  |       NULL |
| School of Arts       | SA   |          30 | NULL    |  7000000.0 |
| School of CIT        | SCIT |          40 | Willy   | 10000000.0 |
| School of Mathematics| SM   |          15 | David   |       NULL |
| School of Physics    | SP   |          20 | Peter   |  5000000.0 |
+----------------------+------+-------------+---------+------------+
```

T1:

```
COMMIT;
SELECT * FROM DEPARTMENT;
```

```
+----------------------+------+-------------+---------+------------+
| name                 | code | total_staff | chair   | budget     |
+----------------------+------+-------------+---------+------------+
| School of Accounting | SACC |          40 | Jennie  |       NULL |
| School of Arts       | SA   |          30 | NULL    |  7000000.0 |
| School of CIT        | SCIT |          40 | Willy   | 10000000.0 |
| School of Mathematics| SM   |          15 | David   |       NULL |
| School of Physics    | SP   |          20 | Peter   |  5000000.0 |
| Test                 | ST   |           1 | NULL    |       NULL |
+----------------------+------+-------------+---------+------------+
```

T2:

```
SELECT * FROM DEPARTMENT;
```

```
+----------------------+------+-------------+---------+------------+
| name                 | code | total_staff | chair   | budget     |
+----------------------+------+-------------+---------+------------+
| School of Accounting | SACC |          40 | Jennie  |       NULL |
| School of Arts       | SA   |          30 | NULL    |  7000000.0 |
| School of CIT        | SCIT |          40 | Willy   | 10000000.0 |
+----------------------+------+-------------+---------+------------+
```

```
| School of Mathematics | SM   |          15 | David  |       NULL |
| School of Physics     | SP   |          20 | Peter  |  5000000.0 |
| Test                  | ST   |           1 | NULL   |       NULL |
+-----------------------+------+-------------+--------+------------+
```

(The user represented by T2 may need to exit then reconnect to the MySQL server again to be able to see the modifications made and committed by T1.)

Then, we change the DEPARTMENT table back to the original status by processing the following statement for future practice.

```
DELETE FROM DEPARTMENT WHERE name = 'Test';
COMMIT;
```

Case 2: T1 (autocommit = 'ON') -> T2 (autocommit = 'ON')

Before practicing Case 2, make sure the autocommit is set to 'ON' in both terminals.

```
SET SESSION AUTOCOMMIT = 'ON';
show variables like 'AUTOCOMMIT';
+---------------+-------+
| Variable_name | Value |
+---------------+-------+
| autocommit    | ON    |
+---------------+-------+
```

T1:

```
INSERT INTO DEPARTMENT VALUES ('Test', 'ST', 1, NULL, NULL);
SELECT * FROM DEPARTMENT;
+-----------------------+------+-------------+--------+------------+
| name                  | code | total_staff | chair  | budget     |
+-----------------------+------+-------------+--------+------------+
| School of Accounting  | SACC |          40 | Jennie |       NULL |
| School of Arts        | SA   |          30 | NULL   |  7000000.0 |
| School of CIT         | SCIT |          40 | Willy  | 10000000.0 |
| School of Mathematics | SM   |          15 | David  |       NULL |
| School of Physics     | SP   |          20 | Peter  |  5000000.0 |
| Test                  | ST   |           1 | NULL   |       NULL |
+-----------------------+------+-------------+--------+------------+
```

T2:

```
SELECT * FROM DEPARTMENT;
+-----------------------+------+-------------+--------+------------+
| name                  | code | total_staff | chair  | budget     |
+-----------------------+------+-------------+--------+------------+
| School of Accounting  | SACC |          40 | Jennie |       NULL |
| School of Arts        | SA   |          30 | NULL   |  7000000.0 |
| School of CIT         | SCIT |          40 | Willy  | 10000000.0 |
| School of Mathematics | SM   |          15 | David  |       NULL |
| School of Physics     | SP   |          20 | Peter  |  5000000.0 |
| Test                  | ST   |           1 | NULL   |       NULL |
+-----------------------+------+-------------+--------+------------+
```

Then, we change the DEPARTMENT table back to the original status by processing the following statement for future practice.

```
DELETE FROM DEPARTMENT WHERE name = 'Test';
```

Case 3: T1 (autocommit = 'OFF') -> T2 (autocommit = 'OFF' & ROLLBACK;) -> T1

Before practicing Case 3, make sure the autocommit is set to 'OFF' in both terminals.

```
SET SESSION AUTOCOMMIT = 'OFF';
show variables like 'AUTOCOMMIT';
+---------------+-------+
| Variable_name | Value |
+---------------+-------+
| autocommit    | OFF   |
+---------------+-------+
```

T1:

```
INSERT INTO DEPARTMENT VALUES ('TestT1', 'ST1', 1, NULL, NULL);
SELECT * FROM DEPARTMENT;
+----------------------+------+-------------+--------+------------+
| name                 | code | total_staff | chair  | budget     |
+----------------------+------+-------------+--------+------------+
| School of Accounting | SACC |          40 | Jennie |       NULL |
| School of Arts       | SA   |          30 | NULL   |  7000000.0 |
| School of CIT        | SCIT |          40 | Willy  | 10000000.0 |
| School of Mathematics| SM   |          15 | David  |       NULL |
| School of Physics    | SP   |          20 | Peter  |  5000000.0 |
| TestT1               | ST1  |           1 | NULL   |       NULL |
+----------------------+------+-------------+--------+------------+
```

T2:

```
INSERT INTO DEPARTMENT VALUES ('TestT2', 'ST2', 1, NULL, NULL);
SELECT * FROM DEPARTMENT;
+----------------------+------+-------------+--------+------------+
| name                 | code | total_staff | chair  | budget     |
+----------------------+------+-------------+--------+------------+
| School of Accounting | SACC |          40 | Jennie |       NULL |
| School of Arts       | SA   |          30 | NULL   |  7000000.0 |
| School of CIT        | SCIT |          40 | Willy  | 10000000.0 |
| School of Mathematics| SM   |          15 | David  |       NULL |
| School of Physics    | SP   |          20 | Peter  |  5000000.0 |
| TestT2               | ST2  |           1 | NULL   |       NULL |
+----------------------+------+-------------+--------+------------+

ROLLBACK;
SELECT * FROM DEPARTMENT;
+----------------------+------+-------------+--------+------------+
| name                 | code | total_staff | chair  | budget     |
+----------------------+------+-------------+--------+------------+
| School of Accounting | SACC |          40 | Jennie |       NULL |
| School of Arts       | SA   |          30 | NULL   |  7000000.0 |
| School of CIT        | SCIT |          40 | Willy  | 10000000.0 |
| School of Mathematics| SM   |          15 | David  |       NULL |
| School of Physics    | SP   |          20 | Peter  |  5000000.0 |
+----------------------+------+-------------+--------+------------+
```

Then, we change the DEPARTMENT table back to the original status by processing the following statement for future practice. In terminal T1, process the following statements:

```
DELETE FROM DEPARTMENT WHERE name = 'TestT1';
COMMIT;
```

Case 4: T1 (autocommit = 'OFF') -> T2 (autocommit = 'OFF' & COMMIT;) -> T1

Before practicing Case 4, make sure the autocommit is set to 'OFF' in both terminals.

```
SET SESSION AUTOCOMMIT = 'OFF';
show variables like 'AUTOCOMMIT';
+---------------+-------+
| Variable_name | Value |
+---------------+-------+
| autocommit    | OFF   |
+---------------+-------+
```

T1:

```
INSERT INTO DEPARTMENT VALUES ('TestT1', 'ST1', 1, NULL, NULL);
SELECT * FROM DEPARTMENT;
+----------------------+------+-------------+--------+------------+
| name                 | code | total_staff | chair  | budget     |
+----------------------+------+-------------+--------+------------+
| School of Accounting | SACC |          40 | Jennie |       NULL |
| School of Arts       | SA   |          30 | NULL   |  7000000.0 |
| School of CIT        | SCIT |          40 | Willy  | 10000000.0 |
| School of Mathematics| SM   |          15 | David  |       NULL |
| School of Physics    | SP   |          20 | Peter  |  5000000.0 |
| TestT1               | ST1  |           1 | NULL   |       NULL |
+----------------------+------+-------------+--------+------------+
```

T2:

```
INSERT INTO DEPARTMENT VALUES ('TestT2', 'ST2', 1, NULL, NULL);
COMMIT;
SELECT * FROM DEPARTMENT;
+----------------------+------+-------------+--------+------------+
| name                 | code | total_staff | chair  | budget     |
+----------------------+------+-------------+--------+------------+
| School of Accounting | SACC |          40 | Jennie |       NULL |
| School of Arts       | SA   |          30 | NULL   |  7000000.0 |
| School of CIT        | SCIT |          40 | Willy  | 10000000.0 |
| School of Mathematics| SM   |          15 | David  |       NULL |
| School of Physics    | SP   |          20 | Peter  |  5000000.0 |
| TestT2               | ST2  |           1 | NULL   |       NULL |
+----------------------+------+-------------+--------+------------+
```

T1:

```
SELECT * FROM DEPARTMENT;
+----------------------+------+-------------+--------+------------+
| name                 | code | total_staff | chair  | budget     |
+----------------------+------+-------------+--------+------------+
| School of Accounting | SACC |          40 | Jennie |       NULL |
| School of Arts       | SA   |          30 | NULL   |  7000000.0 |
| School of CIT        | SCIT |          40 | Willy  | 10000000.0 |
| School of Mathematics| SM   |          15 | David  |       NULL |
| School of Physics    | SP   |          20 | Peter  |  5000000.0 |
| TestT2               | ST2  |           1 | NULL   |       NULL |
+----------------------+------+-------------+--------+------------+
```

(The user represented by T1 may need to exit then reconnect to the MySQL server again to be able to see the modifications made and committed by T2 in the last step of Case 4.)

---End of the Demonstration---