

SQL - Clone Tables

There may be a situation when you need an exact copy of a table and the CREATE TABLE ... or the SELECT... commands does not suit your purposes because the copy must include the same indexes, default values and so forth.

If you are using MySQL RDBMS, you can handle this situation by adhering to the steps given below –

- Use SHOW CREATE TABLE command to get a CREATE TABLE statement that specifies the source table's structure, indexes and all.
- Modify the statement to change the table name to that of the clone table and execute the statement. This way you will have an exact clone table.
- Optionally, if you need the table contents copied as well, issue an INSERT INTO or a SELECT statement too.

Example

Try out the following example to create a clone table for **TUTORIALS_TBL** whose structure is as follows –

Step 1 – Get the complete structure about the table.

```
SQL> SHOW CREATE TABLE TUTORIALS_TBL \G;
***** 1. row *****
      Table: TUTORIALS_TBL
Create Table: CREATE TABLE 'TUTORIALS_TBL' (
  'tutorial_id' int(11) NOT NULL auto_increment,
  'tutorial_title' varchar(100) NOT NULL default '',
  'tutorial_author' varchar(40) NOT NULL default '',
  'submission_date' date default NULL,
  PRIMARY KEY ('tutorial_id'),
  UNIQUE KEY 'AUTHOR_INDEX' ('tutorial_author')
) TYPE = MyISAM
1 row in set (0.00 sec)
```

Step 2 – Rename this table and create another table.

```
SQL> CREATE TABLE `CLONE_TBL` (
  -> 'tutorial_id' int(11) NOT NULL auto_increment,
  -> 'tutorial_title' varchar(100) NOT NULL default '',
  -> 'tutorial_author' varchar(40) NOT NULL default '',
```

```
-> 'submission_date' date default NULL,
-> PRIMARY KEY (`tutorial_id`),
-> UNIQUE KEY 'AUTHOR_INDEX' ('tutorial_author')
-> ) TYPE = MyISAM;
Query OK, 0 rows affected (1.80 sec)
```

Step 3 – After executing step 2, you will clone a table in your database. If you want to copy data from an old table, then you can do it by using the INSERT INTO... SELECT statement.

```
SQL> INSERT INTO CLONE_TBL (tutorial_id,
-> tutorial_title,
-> tutorial_author,
-> submission_date)
-> SELECT tutorial_id,tutorial_title,
-> tutorial_author,submission_date,
-> FROM TUTORIALS_TBL;
Query OK, 3 rows affected (0.07 sec)
Records: 3 Duplicates: 0 Warnings: 0
```

Finally, you will have an exact clone table as you wanted to have.