

Subject: Re: ODBC Assertion failed
Posted by [JeyCi](#) on Mon, 21 Dec 2020 14:33:11 GMT
[View Forum Message](#) <> [Reply to Message](#)

Posted by [JeyCi](#) on Mon, 21 Dec 2020 14:33:11 GMT

[View Forum Message](#) [Reply to Message](#)

was this correct in your whole code?

Quote:SQL = myDB;

```
Sql query(myDB);
```

the insertion below is working for me in Sqlite3 - try to adopt the code to your

MS_SQL_Server_connection (your Commit, Rollback & WasError() are included in this code, I just can not check for ms_sql_server & therefore cannot provide adopted example)... but such is my view on insertion

```
#include <Core/Core.h>
```

```
#include <plugin/sqlite3/Sqlite3.h>
```

```
using namespace Upp;
```

CONSOLE APP MAIN

```
{
  Cout() << "Start" << EOL;
```

[illegible]

```
if(!myDB.Open(ConfigFile("DATA.db"))) {
    LOG("Can't create or open database file\n");
    return;
}
```

```
#ifdef _DEBUG
myDB.SetTrace();
#endif
```

```
SQL = myDB;
// ===== initial data
String s = "[{\\"A\\":\\"valA1\\", \\"B\\":\\"valB1\\"}, {\\"A\\":\\"valA2\\", \\"B\\":\\"valB2\\"}]";
ValueMap js=ParseJSON(s); // frankly speaking ValueArray should be perhaps, but it works for
me...
int jsz=js.GetCount();
```

```
// ===== create tbl
SQL.Execute("drop table IF EXISTS tbl;");
SQL.Execute("create table IF NOT EXISTS tbl (A STRING , B STRING, PRIMARY KEY (A));");
```

```
// ===== insert into tbl
Sql query(myDB); // (my_db)
query.SetStatement("INSERT INTO tbl ( A, B) VALUES ( ?, ?)");
// initialize transaction
query.Begin();
    // loop to insert rows
for(size_t i = 0; i < jsz; i++) {
```

```

    query.SetParam(0,~js[i]["A"]);
    query.SetParam(1,~js[i]["B"]);
    query.Execute();
}
// error behavior
if(query.WasError()){
    Cout() << query.GetLastError();
    query.Rollback();
    return; // false;
}
//Save data & unlock db
query.Commit();
Cout() << "Done" << EOL;

// ===== select to console_view
if (query.Execute("SELECT * FROM tbl"))
{
    Cout() << "Call succeeded" << EOL;

    for (int i = 0; i < query.GetColumnCount(); ++i) {
        Cout() << Format("%d: %s", i, query.GetColumnInfo(i).name) << EOL;
    }

    while (query.Fetch()) {
        Cout() << query[0] << " ; " << query[1] << EOL;
    }
}
else
{
    Cout() << "Call failed" << EOL;
    Cout() << SQL.GetLastError() << EOL;
}
}

```

if you put "valA1" to both A-key values in json-string -- you will catch
Quote:SQL logic error

p.s.

I don't know whether SqlMassInsert will be quicker, but using Transactions gives the possibility for Rollback - I am not sure about this possibility in SqlMassInsert, perhaps it is being done automatically for this class - in LOG should be seen, - but I didn't check SqlMassInsert& UseTransaction(bool b = true) == testing needed to compare the speed of 2 approaches (my & SqlMassInsert)

Wish you good luck if my code could be helpful for you... or somebody can notice some mistakes with a fresh_eye
