Subject: SqlPerformScript bug: Not doing anything if script starts with "BEGIN;" and ends with "END;"
Posted by zsolt on Fri, 13 Oct 2023 01:12:08 GMT
View Forum Message <> Reply to Message

Try this simple code:

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

using namespace Upp;


CONSOLE_APP_MAIN
{
 String sql_script = R"sql(
  BEGIN;
  create table TBL (
   COL integer
  );
  insert into TBL values(123);
  END;
 )sql";

 Sqlite3Session session;
 String file_name = ConfigFile("SqlPerformScript_Test.db");
 DeleteFile(file_name);
 session.Open(file_name);
 SqlPerformScript(session, sql_script);
 Sql sql(session);
 sql * Select(SqlId("COL")).From(SqlId("TBL"));
 if(!sql.Fetch()){
  Cerr() << EOL << "Nothing in table" << EOL;
  return;
 }
 int v = sql[0];
 Cout() << EOL;
 if(v == 123)
  Cout() << "It is OK. Value is: ";
 else
  Cout() << "Bad value: ";
 Cout() << IntStr(v) << EOL;
}
```

It can be fixed with this change in SqlPerformScript():

```
-  if(c == ';' && level == 0 && !esc) {
```

```
+  if(c == ';' && !esc) {
```

I don't know what this level thing should do, but removing level variable and related code seems to be working well also.
Any idea?

```
@@ -18,32 +18,25 @@ StatementExecutor& SQLStatementExecutor() {
 }
 #endif

 bool SqlPerformScript(SqlSession& session, Stream& script,
                       Gate<int, int> progress_canceled, bool stoponerror)
 {
  String stmt;
- int level = 0;
  bool ok = true;
  bool esc = false;
  while(!script.IsEof()) {
   int c = script.Term();
   if(IsAlpha(c) || c == '_') {
    String id;
    while(IsAlNum(script.Term()) || script.Term() == '_') {
     c = script.Get();
     stmt.Cat(c);
     id.Cat(ToUpper(c));
    }
-   if(!esc) {
-    if(id == "BEGIN")
-     level++;
-    if(id == "END")
-     level--;
-   }
   }
   else
   if(c == '\'') {
    stmt.Cat(c);
    script.Get();
    for(;;) {
     c = script.Get();
@@ -65,15 +58,15 @@ bool SqlPerformScript(SqlSession& session, Stream& script,
    if(script.Term() == '$') {
     script.Get();
     stmt.Cat('$');
     esc = !esc;
    }
   }
```

```
   else
-  if(c == ';' && level == 0 && !esc) {
+  if(c == ';' && !esc) {
    Sql sql(session);
    session.ClearError();
    int q = 0;
    while(stmt[q] == '\r' || stmt[q] == '\n')
     q++;
    stmt = stmt.Mid(q);
    if(!sql.Execute(stmt)) {
@@ -81,20 +74,14 @@ bool SqlPerformScript(SqlSession& session, Stream& script,
     if(stoponerror)
      break;
    }
    stmt.Clear();
    script.Get();
   }
   else {
-   if(!esc) {
-    if(c == '(')
-     level++;
-    if(c == ')')
-     level--;
-   }
    if(c != '\r') {
     if(session.GetDialect() == ORACLE && c == '\n')
      stmt.Cat('\r');
     stmt.Cat(c);
    }
    script.Get();
   }
```