Subject: Sql Lite schema with DATE type
Posted by NattyRoots on Thu, 09 May 2019 09:23:48 GMT
View Forum Message <> Reply to Message

Hello,

I'm trying to create a SQL Lite 3 database where I would store the insertion Date. Here's my current .sch :

```
TABLE_ (SESSION)
 INT  (SESSION_ID) PRIMARY_KEY AUTO_INCREMENT
 STRING_ (BATTLETAG, 200)
 STRING_ (DISCORD_NAME, 200)
 STRING_ (RANK, 4)
 DATE_ (DATE_SESSION) SQLDEFAULT(CURRENT_DATE)
END_TABLE
```

It compiles just fine but when I try to execute any SQL query, I get the following error and my .db file isn't updated :

```
Assertion failed in C:\UPP\upp-mingw-12999\upp\uppsrc\plugin\sqlite3\Sqlite3upp.cpp, line 270
NULL != current_stmt
```

I'm updating my .db file using the method found in the SQL Lite 3 reference :

```
Sqlite3Session sqlite3;
sqlite3.LogErrors(true);
if(!sqlite3.Open(ConfigFile("simple.db"))) {
 Cout() << "Can't create or open database file\n";
 return;
}

#ifdef _DEBUG
 SqlSchema sch(SQLITE3);
 All_Tables(sch);
 if(sch.ScriptChanged(SqlSchema::UPGRADE))
  SqlPerformScript(sch.Upgrade());
 if(sch.ScriptChanged(SqlSchema::ATTRIBUTES)) {
  SqlPerformScript(sch.Attributes());
 }
 if(sch.ScriptChanged(SqlSchema::CONFIG)) {
  SqlPerformScript(sch.ConfigDrop());
  SqlPerformScript(sch.Config());
```

```
 }
 sch.SaveNormal();
#endif
```

Is there a special way to deal with date using Sql Lite 3 in U++ ?

Thank you for your help !

Subject: Re: Sql Lite schema with DATE type
Posted by BioBytes on Thu, 09 May 2019 14:32:36 GMT
View Forum Message <> Reply to Message

Hi NattyRoots,

Did you check that ?

https:// www.ultimatepp.org/forums/index.php?t=msg&goto=43918&amp
;SQ=6ba6f2a48c976cdebd1e926ccffe0f49&

Hope it helps
BR

Biobytes

Subject: Re: Sql Lite schema with DATE type
Posted by NattyRoots on Fri, 10 May 2019 13:05:07 GMT
View Forum Message <> Reply to Message

Hello BioBytes,

Thank you for your answer ! But unfortunately it did not seems to be my problem.

I came up with a solution by storing a timestamp generated in C++ in my column DATE_SESSION, rather than using the SQLDEFAULT() :

.sch :

```
TABLE_ (SESSION)
 INT_  (SESSION_ID) PRIMARY_KEY AUTO_INCREMENT
 STRING_ (BATTLETAG, 200)
 STRING_ (DISCORD_NAME, 200)
 STRING_ (RANK, 4)
 INT_ (DATE_SESSION)
```

END_TABLE

Code :

```
std::chrono::milliseconds ms = std::chrono::duration_cast< std::chrono::milliseconds >(
    std::chrono::system_clock::now().time_since_epoch()
);
Sql sqlInsert;
sqlInsert*Insert(SESSION)(BATTLETAG, battletag)(DISCORD_NAME, userName)(RANK,
rang)(DATE_SESSION, ms.count());
```

Thanks for the help anymway,

Natty
