Subject: Re: How to use aliases with SqlExp?
Posted by mirek on Wed, 19 Sep 2018 11:50:03 GMT

View Forum Message <> Reply to Message

Patisab wrote on Wed, 12 September 2018 08:37Hello,

I have a little problem with SqlExp when i want to use an alias for a table used twice in same request. It's something like this :

SELECT t1.id, t1.col1, t1.col2, t1.father, t2.col1, t2.col2
FROM table1 AS t1
INNER JOIN table1 AS t2 ON t2.id = t1.father

In .sch file i declare this :

TABLE_(TABLE1)
 INT_ (ID)   PRIMARY_KEY AUTO_INCREMENT NOT_NULL
 STRING_ (COL1,10) UNIQUE NOT_NULL INDEX
 STRING_ (COL2,50)
 INT_ (FATHER)   UNIQUE NOT_NULL
END_TABLE

The SqlExp used is like this :

SQL & Select(ID.Of(TABLE),COL1.Of(TABLE),COL2.Of(TABLE),COL1.Of(T2),COL2.Of(T2))
.From(TABLE)
.InnerJoin(TABLE).AsTable(T2).On(ID.Of(T2) == FATHER.Of(TABLE));

Compiling send this message :

error: 'T2' was not declared in this scope

Could someone have an idea to solve this problem? Well, i know it is possible to take an other way but i want to use SqlExp... as possible it is ;)

Thank very much for your help.

Best regards

Well, obviously, being an identified in C++ code, it has to be declared somewhere.

For schema IDsthe magic trick is done by that '_' after the type name (STRING_). This says "And also declare this as SqlId variable".

If you are using ID that is not in .sch file, you have to declare it yourself, e.g.

SqlId T1("T1");

as there is some repetition (you have to type T1 twice), there is also a macro that can be used for the same purpose

SQLID(T1);

Alternatively, it is even possible to use C literal instead

COL1.Of("T2")

Last hint: There is a relatively recently added alternative to 'As', operator(), which is especially useful it you are reading more columns from single table:

Select(TABLE(ID ,COL1, COL2), T2(COL1, COL2))

(unfortunately, you cannot use "T2" here, you have to declare it as SQLID).

Mirek