
Subject: Simple SQL question

Posted by [koldo](#) on Fri, 25 Feb 2011 11:43:57 GMT

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

Hello all

I wanted to select all registers from columns Name, Surname and Location that contains either text1 or text2.

For example, if the table is:

Name Surname Location

Mary Poppins Bruxelles

with text1 = "pin" and text2 = "elle", the second register would be selected.

How could it be done?

Subject: Re: Simple SQL question

Posted by [mirek](#) on Fri, 25 Feb 2011 19:00:27 GMT

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

koldo wrote on Fri, 25 February 2011 06:43Hello all

I wanted to select all registers from columns Name, Surname and Location that contains either text1 or text2.

For example, if the table is:

Name Surname Location

Mary Poppins Bruxelles

with text1 = "pin" and text2 = "elle", the second register would be selected.

How could it be done?

<http://www.postgresql.org/docs/8.3/static/functions-matching.html>

I guess it is even SQL standard.

In SqlExp, use Like.

Mirek

Subject: Re: Simple SQL question

Posted by [koldo](#) on Sun, 27 Feb 2011 08:19:02 GMT

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

Hello Mirek

For the question it could be used:

```
SQL.Execute("select *, [Name] || '.' || [Surname] || '.' || [Location] as FULLDATA from PERSONAS  
where FULLDATA like '%" + pin1 + "%' and FULLDATA like '%" + pin2 + "%");
```

The '.' is just to avoid that "aryPop" would match with register MaryPoppinsBruxelles.

Is it possible to do this and use ALIAS with SqlExp or it has to be done with plain SQL?

Subject: Re: Simple SQL question

Posted by [mirek](#) on Fri, 04 Mar 2011 11:00:58 GMT

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

koldo wrote on Sun, 27 February 2011 03:19Hello Mirek

For the question it could be used:

```
SQL.Execute("select *, [Name] || '.' || [Surname] || '.' || [Location] as FULLDATA from PERSONAS  
where FULLDATA like '%" + pin1 + "%' and FULLDATA like '%" + pin2 + "%");
```

The '.' is just to avoid that "aryPop" would match with register MaryPoppinsBruxelles.

Is it possible to do this and use ALIAS with SqlExp or it has to be done with plain SQL?

I have to admin I do not know what '[' and ']' mean in [Name]

However, this excluded:

```
Select((NAME|".|SURNAME).As(FULLDATA)).Where(Like(FULLDATA, "%"+pin1+"%");
```

should work.

Mirek

Subject: Re: Simple SQL question

Posted by [koldo](#) on Fri, 04 Mar 2011 12:36:41 GMT

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

Hello Mirek

There are some problems (I use sqlite):

- Instead of FULLDATA it has to be used "FULLDATA", as it is a new column so it is not in .sch file

- NAME|".|SURNAME is translated to "NAME + '.' + SURNAME". It does not produce any SQL error, but no register is found.

However this works: "NAME || '.' || SURNAME"

- I would like to select all columns plus FULLDATA. How is it done in the Select()?

- A comment: As() is replaced by "" in sqlite. sqlite understands it well, although it also understands the classic "AS"

Thank you for your answer.

Subject: Re: Simple SQL question

Posted by [mirek](#) on Sat, 05 Mar 2011 08:10:58 GMT

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

koldo wrote on Fri, 04 March 2011 07:36Hello Mirek

There are some problems (I use sqlite):

- Instead of FULLDATA it has to be used "FULLDATA", as it is a new column so it is not in .sch file

Then define it locally:

```
SqlId FULLDATA("FULLDATA");
```

(perhaps I would rather use something more simple as "F").

Quote:

- NAME|".|SURNAME is translated to "NAME + '.' + SURNAME". It does not produce any SQL error, but no register is found.

However this works: "NAME || '.' || SURNAME"

That is weird, in SqlExp '|' should be string cat. It could be a bug in SqlExp for specific target DB - what DB is it?

(FYI, it is quite simple, check operator| in SqlExp and see how SqlCase is used to provide different code for different DBs....)

Quote:

- I would like to select all columns plus FULLDATA. How is it done in the Select()?

Select(SqlAll(), FULLDATA)

should work here.

Mirek

Subject: Re: Simple SQL question
Posted by [koldo](#) on Sat, 05 Mar 2011 14:11:26 GMT
[View Forum Message](#) <> [Reply to Message](#)

Thank you Mirek

Yes, there is a problem in operator| for sqlite. The fix would be in SqlVal.cpp:

```
SqlVal operator|(const SqlVal& a, const SqlVal& b) {  
    return SqlVal(a, SqlCase(ORACLE, " || ")(PGSQL, " || ")(SQLITE3, " || ")(" + "), b, SqlS::MUL); //  
    Added (SQLITE3, " || ")  
}
```

In addition, please could you add function coalesce?. The functions could be:

```
SqlVal Coalesce(const SqlVal& exp1, const SqlVal& exp2) {  
    return SqlFunc("coalesce", exp1, exp2);  
}
```

```
SqlVal Coalesce(const SqlVal& exp1, const SqlVal& exp2, const SqlVal& exp3) {  
    return SqlFunc("coalesce", exp1, exp2, exp3);  
}
```

```
SqlVal Coalesce(const SqlVal& exp1, const SqlVal& exp2, const SqlVal& exp3, const SqlVal&  
exp4) {  
    return SqlFunc("coalesce", exp1, exp2, exp3, exp4);  
}
```

It is compatible with at least sqlite, mysql, postgre, oracle ans sql server.

Subject: Re: Simple SQL question
Posted by [mirek](#) on Sun, 06 Mar 2011 18:00:54 GMT
[View Forum Message](#) <> [Reply to Message](#)

Both applied, thanks.

Coalesce is similar Nvl, but that works only for 2 parameters. It looks like by the time we created SqlExp, this was not standardized (e.g. coalesce is not present before Oracle 9i).

Subject: Re: Simple SQL question
Posted by [koldo](#) on Sun, 06 Mar 2011 20:31:20 GMT
[View Forum Message](#) <> [Reply to Message](#)

mirek wrote on Sun, 06 March 2011 19:00Both applied, thanks.

Coalesce is similar Nvl, but that works only for 2 parameters. It looks like by the time we created SqlExp, this was not standardized (e.g. coalesce is not present before Oracle 9i).
Thank you Mirek
