

---

Subject: Schema description documentation not matching code behavior

Posted by [mingodad](#) on Thu, 10 Apr 2014 17:10:17 GMT

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

---

I'm reading the documentation on:

U++ SQL

Basic Use and Description

And it says :

Note: In U++, we tend to use uppercase for all database/table/column names. Also, some databases, like PostgreSQL, have case-sensitive database, table, and column names. Writing queries can be a pain when you have names that have a mix of upper and lower-case letters. By default, unless you use quotes, all names in the macros will result in lower-case names for all tables/columns created.

But in reality in all examples the names are not converted to lower case, they remain upper case.

Which one is wrong the documentation or the code ?

Thanks in advance for your time, attention and great work !

Here is an attempt to make the names lowercase:

----

--- /home/mingo/upp/uppsrc/Sql/SqlSchema.h

+++ /tmp/SqlSchema-0.4.h

@ @ -29,7 +29,6 @ @

```
void Object(const char *text, const char *drop);
```

```
- String NormalizeName(const char *name);  
void Table(const char *name);  
void TableSuffix(const char *suffix);  
void Column(const char *type, const char *name);
```

----

--- /home/mingo/upp/uppsrc/Sql/SqlSchema.cpp

+++ /tmp/SqlSchema-0.3.cpp

@ @ -60,15 +60,9 @ @

```
Upgrade() << text << '\n';
```

```
}
```

```
-String SqlSchema::NormalizeName(const char *name)
```

```
-{
```

```
- String nm = name;
```

```
- return ToLower(nm);
```

```
-}
```

```
-
```

```
void SqlSchema::Table(const char *name) {  
    FlushTable();
```

```

- table = NormalizeName(name);
+ table = name;
  table_suffix = Null;
  Schema() << Expand("create table @t (\n");
  SchemaDrop() << Expand("drop table @t;\n");
@@ -87,12 +81,12 @@

  items = 0;
  type = _type;
- column = prefix + NormalizeName(name);
+ column = prefix + name;
  attribute.Clear();
}

void SqlSchema::ColumnArray(const char *type, const char *name, int _items) {
- Column(type, NormalizeName(name));
+ Column(type, name);
  items = _items;
}
----

```

---

Subject: Re: Schema description documentation not matching code behavior  
 Posted by [mirek](#) on Mon, 14 Apr 2014 12:26:15 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

mingodad wrote on Thu, 10 April 2014 17:10 I'm reading the documentation on:  
 U++ SQL  
 Basic Use and Description

And it says :

Note: In U++, we tend to use uppercase for all database/table/column names. Also, some databases, like PostgreSQL, have case-sensitive database, table, and column names. Writing queries can be a pain when you have names that have a mix of upper and lower-case letters. By default, unless you use quotes, all names in the macros will result in lower-case names for all tables/columns created.

But in reality in all examples the names are not converted to lower case, they remain upper case.

Which one is wrong the documentation or the code ?

Thanks in advance for your time, attention and great work !

Here is an attempt to make the names lowercase:

```

----
--- /home/mingo/upp/uppsrc/Sql/SqlSchema.h
+++ /tmp/SqlSchema-0.4.h
@@ -29,7 +29,6 @@

```

```

void Object(const char *text, const char *drop);

- String NormalizeName(const char *name);
void Table(const char *name);
void TableSuffix(const char *suffix);
void Column(const char *type, const char *name);
----
--- /home/mingo/upp/uppsrc/Sql/SqlSchema.cpp
+++ /tmp/SqlSchema-0.3.cpp
@@ -60,15 +60,9 @@
    Upgrade() << text << '\n';
}

-String SqlSchema::NormalizeName(const char *name)
-{
- String nm = name;
- return ToLower(nm);
-}
-
void SqlSchema::Table(const char *name) {
    FlushTable();
- table = NormalizeName(name);
+ table = name;
    table_suffix = Null;
    Schema() << Expand("create table @t (\n");
    SchemaDrop() << Expand("drop table @t;\n");
@@ -87,12 +81,12 @@

    items = 0;
    type = _type;
- column = prefix + NormalizeName(name);
+ column = prefix + name;
    attribute.Clear();
}

void SqlSchema::ColumnArray(const char *type, const char *name, int _items) {
- Column(type, NormalizeName(name));
+ Column(type, name);
    items = _items;
}
----

```

Documentation is wrong here... Even in the point of PGSQL being case sensitive (it is not).

Mirek

---