Having a set of points with a 'code' attribute consisting of a 2-char code+5digits I wanted to select all dutch points (starting with NL).

Reading `substr(string,start[,length])` I tried:

eg: `code = "NL1234"`

`substring("code", 0, 2) => 'NL'`

then I thought to start on position 1 so did

`substring("code", 1, 2) => 'NL'`

I actually started questioning because I started of with

`substring("code", 0, 3) => 'NL'` thinking it was from-till

then tried

`substring("code", 0, 2) and also came to 'NL'`

So there are three ways to find the first 2 chars :-)

Isn't this confusing? Should an expression not either be zero-based or one-based?

We could argue that expressions are not for devs/coders so we deliberately choose for 1-based. But I think then we should put that in the Help text (so peeps like me should not try zero) and using zero then should work.

But personally I would just prefer zero-based...Mmm, testing LibreOffice here, I see that using a zero as start actually returns an error)...

So this is actually a feature?

Associated revisions

Revision 2bed7ab0 - 2019-02-07 11:17 AM - Jürgen Fischer

fix substr help (fixes #21192)

Revision eb8e36c2 - 2019-02-08 12:30 AM - Jürgen Fischer

fix substr help (fixes #21192)

(cherry picked from commit 2bed7ab0018696dba76d2c8053d22633a271486e)
Not reproducible here:

```plaintext
subst('NL1234',0,2) => 'NL'
subst('NL1234',1,2) => 'NL'
subst('NL1234',0,3) => 'NL1'
```

#2 - 2019-02-07 01:05 AM - Jürgen Fischer

- Status changed from Open to Feedback

#3 - 2019-02-07 10:08 AM - Richard Duivenvoorde

Mmm, strange. So the last example I cannot reproduce here either.

But my point that it is not clear if it is zero based or one based is still valid:

```plaintext
subst('NL1234',0,2) => 'NL'
subst('NL1234',1,2) => 'NL'
```

But we could consider this a 'feature'...

Problem would have been bigger if the second parameter would have been the end-stop instead of the length....

Close?

#4 - 2019-02-07 11:17 AM - Jürgen Fischer

Richard Duivenvoorde wrote:

```
Mmm, strange. So the last example I cannot reproduce here either.

But my point that it is not clear if it is zero based or one based is still valid:

subst('NL1234',0,2) => 'NL'
subst('NL1234',1,2) => 'NL'
```

But we could consider this a 'feature'...

Problem would have been bigger if the second parameter would have been the end-stop instead of the length....

Close?

The help doesn't say where the indexes start - but the examples show the current behavior (ie. index start at 1 - like in postgresql for instance; postgresql however produces an empty string on `substring('NL1234',0,1)` and 'N' on `substring('NL1234',0,2)`). Except

`subst('HELLO WORLD',6) → 'WORLD'`
Applied in changeset commit: qgis|2bed7ab0018696dba76d2c8053d22633a271486e.