

QGIS Application - Feature request #7510

NULL string should be valid in label expressions

2013-04-04 07:12 AM - John Sankey

Status: Closed	
Priority: Normal	
Assignee:	
Category:	
Pull Request or Patch supplied:	Resolution: wontfix
Easy fix?: No	Copied to github as #: 16460
Description	
A multiline label "line1" \ ' "line2" shows nothing if either one of the fields is null. I'm told that in 'Master' the construct "line1" '-n' CASE WHEN length("line2") > 0 THEN "line2" ELSE ' ' END is provided to deal with this. This is an unnecessary complication in the case of strings. A null string is a valid value in all programming languages I know, and it should be in QGIS too.	
Related issues:	
Related to QGIS Application - Bug report # 7380: Different behavior for "Fiel...	Closed 2013-03-17
Duplicated by QGIS Application - Bug report # 8369: Null propagation of field...	Closed 2013-07-27

History

#1 - 2013-04-04 07:31 AM - Nathan Woodrow

The || (double pipe) is string concatenation which returns null because NULL doesn't contain a state or value so it can't know what to do.

QgsExpression follows the normal SQL rules with the final result being NULL. Postgres and TSQL (and the others) all follow this rule.

```
DECLARE @string1 as varchar(max), @string2 as varchar(max)
SET @string1 = 'Hello'
SET @string2 = NULL

SELECT @string1 + @string2
```

Results in NULL

Python also doesn't like it:

```
>>> string1 = "Hello"
>>> string2 = None
>>> string1 + string2
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: cannot concatenate 'str' and 'NoneType' objects
```

There is a new format() function that is cleaner than doing || all over the place. We could add NULL to " replacement in there.

#2 - 2013-04-04 07:35 AM - Jürgen Fischer

- Resolution set to wontfix
- Status changed from Open to Closed

| A null string is a valid value in all programming languages I know, and it should be in QGIS too.

Not in SQL. And that's what the expressions behave like.

| `"line1"||'-n'||CASE WHEN length("line2") > 0 THEN "line2" ELSE ' ' END`
| is provided to deal with this. This is an unnecessary complication in the case of strings.

Indeed - not sure if that even works.

A simpler form would be `coalesce(line1||^`
`;'')||coalesce(line2,')` (which also suppresses the new line if the first line is NULL)

#3 - 2013-08-01 04:31 PM - Alister Hood

- OS version deleted (10.6.8)
- Operating System deleted (Mac OSX)

Perhaps we need some prominent documentation on NULL values...