

QGIS Application - Bug report #18062  
floating point error in round function in Ubuntu

2018-02-08 03:07 PM - Michel Stuyts

<b>Status:</b>	Closed	
<b>Priority:</b>	Normal	
<b>Assignee:</b>		
<b>Category:</b>	Expressions	
<b>Affected QGIS version:</b>	master	<b>Regression?:</b> No
<b>Operating System:</b>		<b>Easy fix?:</b> No
<b>Pull Request or Patch supplied:</b>		<b>Resolution:</b> up/downstream
<b>Crashes QGIS or corrupts data:</b>		<b>Copied to github as #:</b> 25958
<b>Description</b>		
<p>The following procedure gives a floating point error in the results in Ubuntu Linux</p> <ol style="list-style-type: none"><li>1. Open QGIS 2.99 in Ubuntu Linux 16.04 LTS</li><li>2. Create a new shapefile for lines</li><li>3. Start editing this shapefile and create a line</li><li>4. Stop editing the shapefile and save the edits</li><li>5. Go to the layer styling panel and select "Show labels for this layer"</li><li>6. For "Label with" add round(123.456789,2) as value and click "Apply"</li><li>7. The label now shows 123.459999999999</li><li>8. Open attribute table</li><li>9. Open field calculator</li><li>10. Select Create new field, with output field type Real, output field length 10 and Precision 10 (that's the highest possible value). As a expression enter round(123.456789,2).</li><li>11. The output preview says 123.459999999999</li><li>12. The result in the table says 123.4600000000</li></ol> <p>On Windows 10, the result is always the correct 123.46.</p>		

History

#1 - 2018-02-09 12:10 AM - Kory Roberts

Looks similar to these already reported bugs:

#17861

#18007

#2 - 2018-02-09 01:26 AM - Nyal Dawson

- Resolution set to up/downstream
- Status changed from Open to Closed

This is an issue with the underlying Qt 5.6 library, and is not something we can fix in QGIS (believe us, we tried HARD). The ONLY solution is to use a build with a newer version of Qt (5.7 and above are fine) - that may mean updating your distro.

#3 - 2018-02-12 10:33 AM - Michel Stuyts

Since I just want to show it as a label and don't want to calculate with it, for now I'll just use the following custom function in the Function Editor:

```
@qgsfunction(args='auto', group='Custom')  
def str_round(value1, value2, feature, parent):  
    return str(round(value1,value2))
```

This will show the correct result as a string.