

## Base Language - Bug #6389

### LENGTH function with COLUMN option

05/23/2022 01:39 PM - Constantin Asofiei

<b>Status:</b>	New	<b>Start date:</b>	
<b>Priority:</b>	High	<b>Due date:</b>	
<b>Assignee:</b>	Ovidiu Maxiniuc	<b>% Done:</b>	20%
<b>Category:</b>		<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>		<b>case_num:</b>	
<b>billable:</b>	No	<b>version:</b>	
<b>vendor_id:</b>	GCD		
<b>Description</b>			

#### History

#2 - 05/23/2022 01:40 PM - Constantin Asofiei

- Priority changed from Normal to High

The LENGTH(..., COLUMN) function in TextOps.columnLength needs to be implemented.

#3 - 05/23/2022 09:19 PM - Ovidiu Maxiniuc

- Status changed from New to WIP

The FWD conversion automatically detects the length unit/type and report an error for COLUMN. This can be easily addresses. My problem is what does this truly means and whether is there a difference from CHARACTER units. I did some tests with OE and I could not find any example where these are different.

I had some various assumption like:

- naïve: the string is trimmed, so even if "abc " has 4 characters it will actually need only 3 columns to be displayed, because the last space is not visible. False, LENGTH("abc ", "COLUMN") = 4. I tried other characters like CR / LF / NBSP with same result. Even the ~b (bell / beep) requires a special column to be displayed;
- the TAB character. Normally, on TTYs, the \t is used to arrange text in columns and, by default, a single character will push the rest of the string up to 8 columns to the right. In this case, "abc~t" would require 8 columns, and "a~tbc" - 10. The answer is negative: LENGTH("abc~t", "COLUMN") = 4. And any other combination.
- some special characters (maybe CJK ?) which are wide enough and cannot be displayed on a single column could require two (multiple) columns to be displayed. I am not aware of any of these and my searching on the internet provided a lot of unrelated results :(. Maybe some emojis?

My only solution now is to make the LENGTH function to return the same result as for CHARACTER unit. When we encounter such an example in which the values differ between these two units, I will handle it myself. If you have any other vision of what this COLUMN is intended, please let me know.

**#4 - 05/24/2022 12:24 AM - Constantin Asofiei**

The conversion is fixed in 6129a. Please make the changes in 6129a.

**#5 - 05/24/2022 06:32 AM - Greg Shah**

Marian: Please review [#6389-3](#) and post any thoughts.

**#6 - 05/24/2022 07:29 AM - Marian Edu**

Greg Shah wrote:

Marian: Please review [#6389-3](#) and post any thoughts.

Greg, I've never used COLUMN type on LENGTH function before and doing a couple of quick tests I have to agree with Ovidiu and just make that return the same result as when CHARACTER type is used for time being and then worry more about it when we have a use-case that fails... I've couldn't thought of anything new, trying different codepages (internal/stream) for client startup, chui/gui doesn't really make any difference so no idea when the results might be different nor why this was added in the first place :(

**#7 - 05/24/2022 11:42 AM - Constantin Asofiei**

I think this is related to chinese characters:

```
def var ch as char.  
ch = CHR(14990001, "cp936", "UTF-8").  
message length(ch, "column").
```

will show '2', but I'm not sure how true this is (if this is a real displayable char or not) as I can't make a display ch work.

Some details here: <https://knowledgebase.progress.com/articles/Article/P108864> and here: <https://stackoverflow.com/questions/3634627/how-to-know-the-preferred-display-width-in-columns-of-unicode-characters>

**#8 - 05/24/2022 11:56 AM - Ovidiu Maxiniuc**

Thank you. My search were in vain. So it is the 3rd bullet (CJK glyphs).

I tried to launch the pro/prowin with the CP parameters from KB but I am getting the following errors:

```
DO NOT CONTINUE. Character set cp936 requires DBE PROGRESS. You may corrupt files or databases. (3624)  
Unable to open word-break table file 247. (2736)  
The word-rule file specified by the -ttwrdrul parameter is invalid. (9258)
```

before the process quits :( I did not specified any -ttwrdrul. I am investigating the issue.

## #9 - 05/24/2022 02:07 PM - Ovidiu Maxiniuc

I added runtime support for LENGTH(..., 'COLUMN'). For the moment it delegates to LENGTH(..., 'CHARACTER') routine. A message will be logged in this event. Committed to 6129a as r13875.

Some notes. I did further investigations based on Constantin example in note 7.  
Using the ChUI client, and altering the code to look like:

```
message ch "character"      length(ch, "character").
message ch "char"           length(ch, "char").
message ch "raw"            length(ch, "raw").
message ch "column"         length(ch, "column").
```

it prints (I redirected the output to a file stream):

```
üï character 2
üï char 2
üï raw 2
üï column 2
```

I think we have now a bit of understanding of the reason why 2 is returned in this case. However, on GUI, the output is:

```
üï character 0
üï char 0
üï raw 2
üï column 2
```

If not redirected to file, there is no character printed on screen. Isn't that strange?

It is also strange that the RAW mode returns 2. The character at hand is 14990001 = 0xE4BAB1 and this does not seem to fit in the 2 bytes range (0-65535).

The funny thing is, FWD will (correctly ???) decode it as `üï`.

A second issue here is related to other differences of the result when the same code is executed on ChUI compared to GUI. Consider the code:

```
message ch 'trim("+++") + "+++" length(ch, trim("+++") + "++").
```

This requires the detection of the unit type to be decided at runtime. The output is different, again. On ChUI the result is the same like CHARACTER unit:

```
üï trim("+++") + "+++" 2
```

but on GUI, I encountered the following error (with message output):

```
** The data type argument value must be "raw", "character" or "column". (1186)
üï trim("+++") + "+++"
```

Because the implementation of the LENGTH function reside in com.goldencode.p2j.util package, the attempt to use LogicalTerminal.isChui() will create unwanted dependency to com.goldencode.p2j.ui.\*;. So I commented out the code, but kept it.

**#10 - 05/24/2022 02:08 PM - Ovidiu Maxiniuc**

- *Status changed from WIP to Review*

- *% Done changed from 0 to 100*

**#11 - 05/24/2022 02:20 PM - Marian Edu**

Maybe the sample attached to this KB entry could help.

[[<https://community.progress.com/s/article/P193813>]]

**#12 - 06/07/2022 07:39 AM - Constantin Asofiei**

- *Status changed from Review to New*

- *% Done changed from 100 to 20*

I'm placing this back to NEW, as we don't have an answer or solution for Chinese characters yet.