



ODC 20
26

Omnis Developer Conference

Make Text Stand Out.

Maurin Wannags,
Omnis Software
Germany GmbH



Content

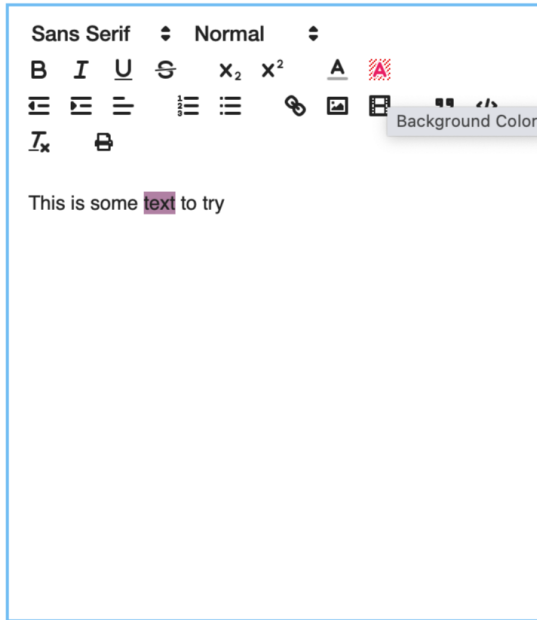
Make Text Stand Out.

- Highlighting Text in Omnis Studio
- Dynamic Text Highlighting
- Approach: JSON String Modulation
- Demonstration

Highlighting Text in Omnis Studio

Make Text Stand Out.

Highlighting Text in Omnis Studio

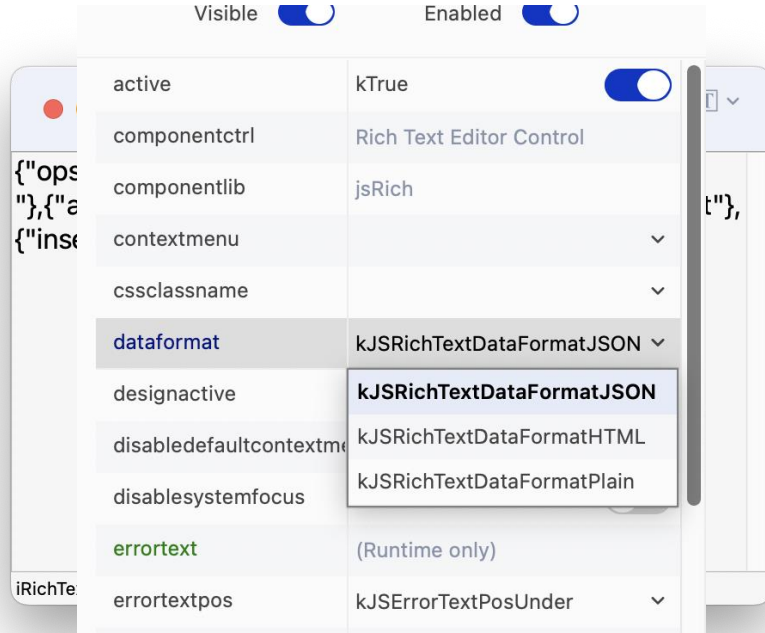


- Not directly supported in most fields/components
- Exception: the Rich Text Editor
 - *Background color*

Dynamic Text Highlighting

Make Text Stand Out.

Dynamic Text Highlighting



- To be able to see formatting in the linked variable or row, the \$dataformat of the Rich Text Editor should be set to:
 - *kJSRichTextDataFormatJSON*
 - *kJSRichTextDataFormatHTML*
- In this case (JSON), the formatting is implemented using a “background attribute”.

Make Text Stand Out.

Dynamic Text Highlighting

```
1  {
2  |   "ops": [
3  |     {
4  |       |   "insert": "This is some text to try\n"
5  |       |   }
6  |     ]
7  |   }
8
9
10
11
12
13
14
15
16
```

- Locate the search text in the JSON string/object
- Close the JSON object
- Insert a new JSON object containing the found text and the “background attribute”
- Reopen the JSON object

Make Text Stand Out.

Dynamic Text Highlighting

```
{
  "ops": [
    {
      "insert": "This is some "
    },
    {
      "attributes": {"background": "#e60000"},
      "insert": "text"
    },
    {
      "insert": " to try\n"
    }
  ]
}
```

- Either edit the linked variable directly ("Field Reference")
- Or load the JSON string into the linked variable as the result

Approach: JSON String Modulation

Make Text Stand Out.

Approach: JSON String Modulation

```
# here the magic begins ;-)
Begin text block
If pCaseSensitive
  Text:[left(pText,pos(pWord,pText)-1)]
Else
  Text:[left(pText,pos(low(pWord),low(pText))-1)]
End If
Text:},{"attributes":{"background":pColor
Text:},"insert":iDummy
Text:},{"insert":pCaseSensitive
  Text:[mid(pText,pos(pWord,pText)+len(pWord))]
Else
  Text:[mid(pText,pos(low(pWord),low(pText))+len(pWord))]
End If
End text block
Get text block pText
# note that the calling method needs to replace the dummy string back to the word
```

- Take case into account (pCaseSensitive)
- To prevent an infinite loop – since the word would be found over and over again – a "dummy word" is used.

Demonstration

Link to the Tech Note / Library:

<https://www.omnis.net/developers/resources/technotes/tnjs0017.jsp>



Thank you very much!