RTF

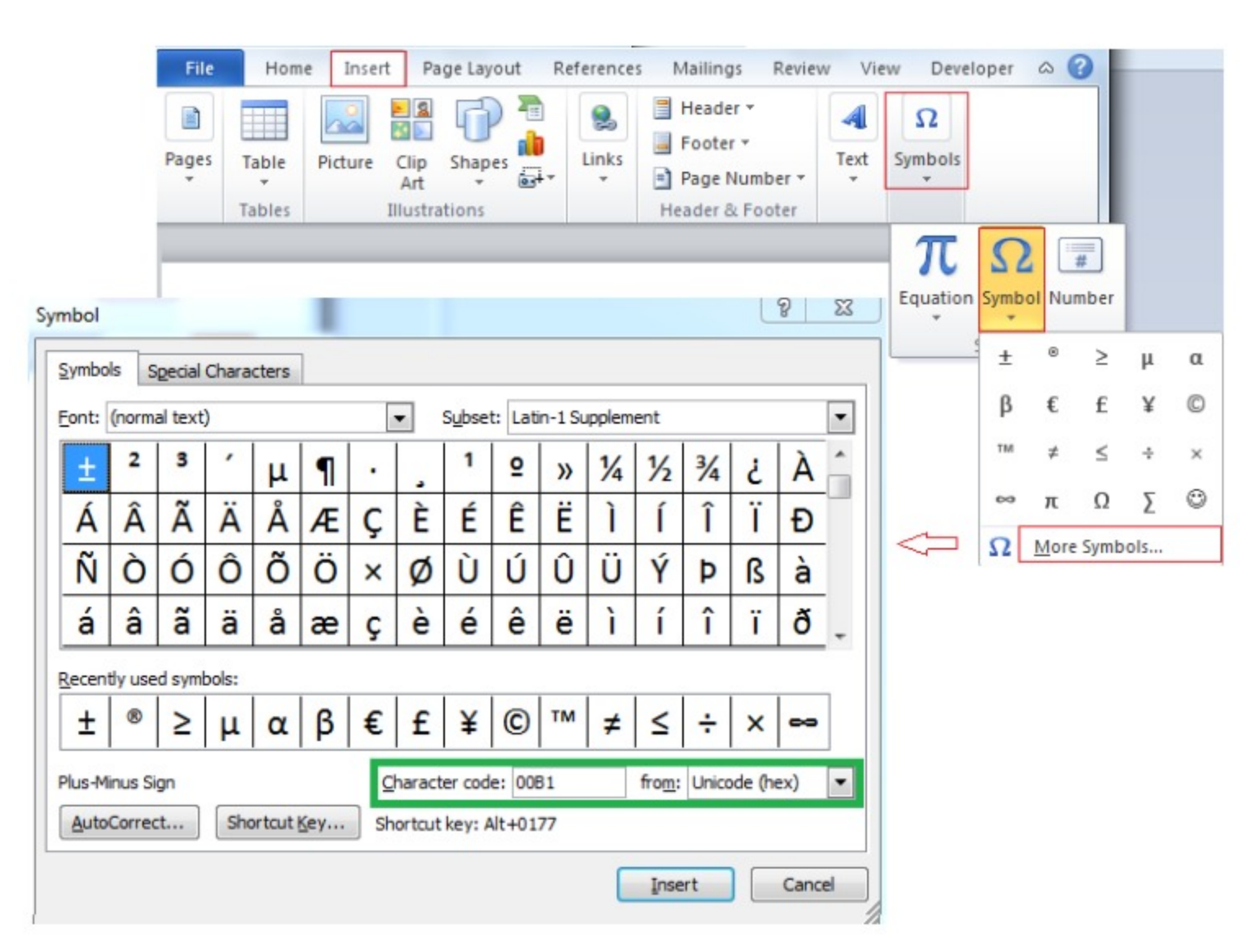
How to customize RTF output using Unicode and RTF code

PUBLISHED ON JULY 2, 2017 BY J.L. COMMENTS OFF

This post will discuss how to use unicode and RTF code to customize our tables. Unicode can be applied to insert special characters such as Greek letts (μ), greater than or equal to (≥), plus-minus sign(±), and registered trademark (®) into our RTF output. RTF code can be used to create lines, a blank line and even turn normal text into subscript or superscript.

Unicode for special characters

Before moving forward to create special characters, we need to know the corresponding unicode for each special character. And we can get this by using Microsoft Word. Click on **Insert** tab -> **Symbols** -> **Symbol** -> **More Symbols...** to trigger out **Symbol** dialog box. If you click on any symbol, you can see that the unicode for this symbol will appear at the bottom-right corner of the **Symbol** dialog box. For example, the unicode of plus-minus symbol is 00B1.



Here I have made a summary of special characters and how to create it in both **ODS RTF** and **ODS TAGSETS.RTF** destination. Please note that the sign before **{unicode** should be the same as that in **ods escapechar** statement.

− Click here to hide/show code

```
data unicode;
input @1 code $1-70;
datalines;
~{unicode 03B1} ~{unicode 03B2} ~{unicode 03B3} ~{unicode 03B4}
~{unicode 03B5} ~{unicode 03B6} ~{unicode 03B7} ~{unicode 03B8}
~{unicode 03BB} ~{unicode 03BC} ~{unicode 03BD} ~{unicode 03BE}
~{unicode 03BF} ~{unicode 03C3} ~{unicode 03C6} ~{unicode 03C0}
~{unicode 2265} ~{unicode 2264} ~{unicode 2260} ~{unicode 2030}
~{unicode 2248} ~{unicode 2261} ~{unicode 00B1}
~{unicode 221A} ~{unicode 221E} ~{unicode 2229} ~{unicode 222B}
~{unicode 00B0}C ~{unicode 03BC}mol/L
~{unicode 00AE} ~{unicode 00A9} ~{unicode 0040} ~{unicode 2122}
~{unicode 002B} ~{unicode 002D} ~{unicode 00D7} ~{unicode 00F7}
~{unicode 00B9} ~{unicode 00B2} ~{unicode 00B3} ~{unicode 0060}
~{unicode 2153} ~{unicode 2154}
~{unicode 00BC} ~{unicode 00BD} ~{unicode 00BE}
~{unicode 2155} ~{unicode 2156} ~{unicode 2157} ~{unicode 2158}
~{unicode 2159} ~{unicode 215A}
~{unicode 215B} ~{unicode 215C} ~{unicode 215D} ~{unicode 215E}
~{unicode 2190} ~{unicode 2191} ~{unicode 2192} ~{unicode 2193}
~{unicode 2194} ~{unicode 2195} ~{unicode 21A8}
~{unicode 2196} ~{unicode 2197} ~{unicode 2198} ~{unicode 2199}
~{unicode 25A1} ~{unicode 25AA} ~{unicode 25AB}
~{unicode 25CA} ~{unicode 25CB} ~{unicode 25CC} ~{unicode 25CF}
run;

ods escapechar="~" ;
options orientation=landscape nodate nonumber noquotelenmax;
options formchar="|—|+|—+=|-/\<>*";
ods rtf file="D:\test.rtf" nogtitle nogfootnote;
proc report data=unicode style(column)=[just=l];
run;
ods rtf close;

ods escapechar="~" ;
options orientation=landscape nodate nonumber noquotelenmax;
options formchar="|—|+|—+=|-/\<>*";
ods tagsets.rtf file="D:\test.rtf" uniform nogtitle nogfootnote device=ACTXIMG;
proc report data=unicode style(column)=[just=l];
run;
ods tagsets.rtf close;
```

By submiting above code, you will get an output as below. I also added the corresponding unicode which can make it easier for you to check in future.

symbol	unicode
α β γ δ	~{unicode 03B1} ~{unicode 03B2} ~{unicode 03B3} ~{unicode 03B4}
ε ζ η θ	~{unicode 03B5} ~{unicode 03B6} ~{unicode 03B7} ~{unicode 03B8}
λ μ ν ξ	~{unicode 03BB} ~{unicode 03BC} ~{unicode 03BD} ~{unicode 03BE}
ο σ φ π	~{unicode 03BF} ~{unicode 03C3} ~{unicode 03C6} ~{unicode 03C0}
≥ ≤ ≠ ‰	~{unicode 2265} ~{unicode 2264} ~{unicode 2260} ~{unicode 2030}
≈ ≡ ±	~{unicode 2248} ~{unicode 2261} ~{unicode 00B1}
√ ∞ ∩ ∫	~{unicode 221A} ~{unicode 221E} ~{unicode 2229} ~{unicode 222B}
°C μmol/L	~{unicode 00B0}C ~{unicode 03BC}mol/L
® © @ ™	~{unicode 00AE} ~{unicode 00A9} ~{unicode 0040} ~{unicode 2122}
+ - × ÷	~{unicode 002B} ~{unicode 002D} ~{unicode 00D7} ~{unicode 00F7}
¹ ² ³ `	~{unicode 00B9} ~{unicode 00B2} ~{unicode 00B3} ~{unicode 0060}
⅓ ⅔	~{unicode 2153} ~{unicode 2154}
¼ ½ ¾	~{unicode 00BC} ~{unicode 00BD} ~{unicode 00BE}
⅕ ⅖ ⅗ ⅘	~{unicode 2155} ~{unicode 2156} ~{unicode 2157} ~{unicode 2158}
⅙ ⅚	~{unicode 2159} ~{unicode 215A}
⅛ ⅜ ⅝ ⅞	~{unicode 215B} ~{unicode 215C} ~{unicode 215D} ~{unicode 215E}
← ↑ → ↓	~{unicode 2190} ~{unicode 2191} ~{unicode 2192} ~{unicode 2193}
↔ ↕ ↨	~{unicode 2194} ~{unicode 2195} ~{unicode 21A8}
↖ ↗ ↘ ↙	~{unicode 2196} ~{unicode 2197} ~{unicode 2198} ~{unicode 2199}

RTF code for special output

RTF code is very useful as it can be used to give our text some special styles. For example, **\sub** is used to create subscripts while **\super** can be used to create superscripts. **\b** turns on bold while **\b0** turns off bold. **\qc** can center aligned the text. **\li200** can insert 200 twips indent before the text. Here is also a summary of special outputs and corresponding RTF code. The example applies **ODS ESCAPECHAR =** statement together with the raw-text function **R** to pass raw text to RTF only. **/RTF** means that destination other than RTF ignores the raw text.

− Click here to hide/show code

```
data unicode;
input @1 code $1-90;
datalines;
This is subscript T~R/RTF"\sub" 2 ~R/RTF"\plain\fs18" How to switch to Normal text
This is superscript T~R/RTF"\super" 2 ~R/RTF"\plain\fs18" How to switch to Normal text
~R/RTF"\ql" sample for left aligned
~R/RTF"\qc" sample for center aligned
~R/RTF"\qr" sample for right aligned
~R/RTF"\li0" sample for no indent
~R/RTF"\li200" sample for 200 twips indent
~R/RTF"\li400" sample for 400 twips indent
sampe for~R/RTF"\~" black space
~R/RTF"\b" sample for bold ~R/RTF"\b0" How to switch to normal text
~R/RTF"\i" italicized text ~R/RTF"\i0" switch to normal text
~R/RTF"\ul" underline text ~R/RTF"\ul0" switch to normal text
~R/RTF"\highlight2" sample for highlight
Sample line 1 ~R/RTF"\line" Sample line 2
~R/RTF"\strike" This is striked
~R/RTF"\brdrt\brdrs\brdrw14" This is a top line for a table cell or table row
~R/RTF"\brdrb\brdrs\brdrw14" This is a bottom line for a table cell or table row
run;

ods escapechar="~" ;
options orientation=landscape nodate nonumber noquotelenmax;
options formchar="|—|+|—+=|-/\<>*";
ods rtf file="&tabledir/test.rtf" nogtitle nogfootnote;
proc report data=unicode style(column)=[just=l];
run;
ods rtf close;

ods escapechar="~" ;
options orientation=landscape nodate nonumber noquotelenmax;
options formchar="|—|+|—+=|-/\<>*";
ods tagsets.rtf file="&tabledir/test.rtf" uniform nogtitle nogfootnote device=ACTXIMG;
proc report data=unicode style(column)=[just=l];
run;
ods tagsets.rtf close;
```

And here is a one-to-one mapping illustration.

Ouput	RTF Code
This is subscript T_2 How to switch to Normal text	~R/RTF"\sub" 2 ~R/RTF"\plain\fs18" How to switch to Normal text
This is superscript T^2 How to switch to Normal text	This is superscript T~R/RTF"\super" 2 ~R/RTF"\plain\fs18" How to switch to Normal text
sample for left aligned	~R/RTF"\ql" sample for left aligned
sample for center aligned	~R/RTF"\qc" sample for center aligned
sample for right aligned	~R/RTF"\qr" sample for right aligned
sample for no indent	~R/RTF"\li0" sample for no indent
sample for 200 twips indent	~R/RTF"\li200" sample for 200 twips indent
sample for 400 twips indent	~R/RTF"\li400" sample for 400 twips indent
sampe for blank space	sampe for~R/RTF"\~" blank space
sample for bold How to switch to normal text	~R/RTF"\b" sample for bold ~R/RTF"\b0" How to switch to normal text
italicized text switch to normal text	~R/RTF"\i" italicized text ~R/RTF"\i0" switch to normal text
underline text switch to normal text	~R/RTF"\ul" underline text ~R/RTF"\ul0" switch to normal text
[illegible]	~R/RTF"\highlight2" sample for highlight
Sample line 1 Sample line 2	Sample line 1 ~R/RTF"\line" Sample line 2
~~This is striked~~	~R/RTF"\strike" This is striked
This is a top line for a table cell or table row	~R/RTF"\brdrt\brdrs\brdrw14" This is a top line for a table cell or table row
This is a bottom line for a table cell or table row	~R/RTF"\brdrb\brdrs\brdrw14" This is a bottom line for a table cell or table row

Apply template to aviod use of ~R/RTF

It is annoying to type **~R/RTF** every time we need to apply RTF code. And we can fix this issue by adding **protectspecialchars=off** to template. Here is an example of our template and how to use it.

− Click here to hide/show code

```
proc template;
define style customtemp;
parent = Styles.RTF;

replace Body from Document /
protectspecialchars = off
asis = on
;

replace NoteContent from NoteContent /
protectspecialchars = off
asis = on
;

replace Header from Header /
protectspecialchars = off
asis = on
;

replace Data from Data /
protectspecialchars = off
asis = on
;
end;
run;

data unicode;
input @1 code $1-70;
datalines;
This is subscript T\sub 2 \plain\fs18 How to switch to Normal text
This is superscript T\super 2 \plain\fs18 How to switch to Normal text
\ql sample for left aligned
\qc sample for center aligned
\qr sample for right aligned
\li0 sample for no indent
\li200 sample for 200 twips indent
\li400 sample for 400 twips indent
sampe for\~ black space
\b sample for bold \b0 How to switch to normal text
\i italicized text \i0 switch to normal text
\ul underline text \ul0 switch to normal text
\highlight2 sample for highlight
Sample line 1\line Sample line 2
\strike This is striked
\brdrt\brdrs\brdrw14 This is a top line for a table cell or table row
\brdrb\brdrs\brdrw14 This is a bottom line for a table cell or table row
run;

ods escapechar="~" ;
options orientation=landscape nodate nonumber noquotelenmax;
options formchar="|—|+|—+=|-/\<>*";
ods rtf file="&base2/test.rtf" style=customtemp nogtitle nogfootnote;
proc report data=unicode style(column)=[just=l];
run;
ods rtf close;

ods escapechar="~" ;
options orientation=landscape nodate nonumber noquotelenmax;
options formchar="|—|+|—+=|-/\<>*";
ods tagsets.rtf file="&base2/test.rtf" style=customtemp uniform nogtitle nogfootnote
device=ACTXIMG;
proc report data=unicode style(column)=[just=l];
run;
ods tagsets.rtf close;
```

Here is the output and corresponding rtf code.

Like the post? Welcome to share or you can subscribe to get latest post. How to subscribe? Go to the top-right corner of this web page to submit your name and email address. If you cannot receive emails from us, please check your spam/junk folder.

If you like this tutorial and found it helpful please feel free to leave us a donation.

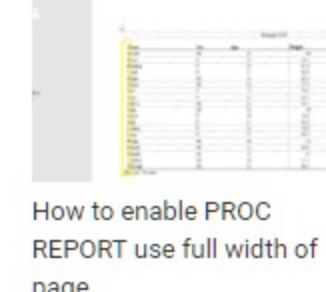
Donate

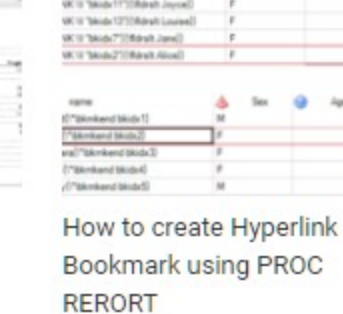
SHARE THIS FACEBOOK TWITTER

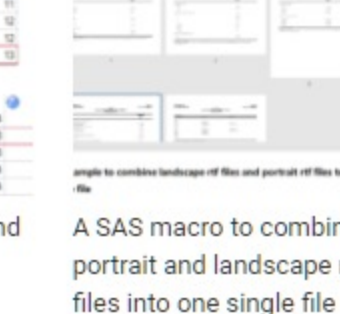
PREVIOUS
How to retrieve comments from word document using VBA

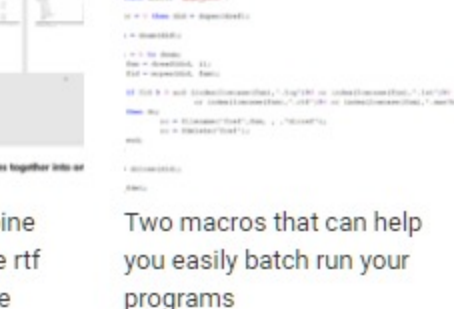
NEXT
Noval approach to derive PARAMTYP and DETYPE for BDS ADaM dataset

RELATED POSTS


How to enable PROC REPORT use full width of page


How to create Hyperlink and Bookmark using PROC RERORT


A SAS macro to combine portrait and landscape rtf files into one single file


Two macros that can help you easily batch run your programs

Name

Email*

Subscribe

RECENT POSTS

Extract System Organ Class of various MedDRA versions from PDF files using Python
Python Installation by using Anaconda
How to add multiple levels of bookmark into PDF via VBA
How to enable PROC REPORT use full width of page
How to reference special characters produced by RTF code via VBA

ARCHIVES

June 2019
May 2019
November 2018
August 2018
July 2018
June 2018
May 2018
April 2018
December 2017
September 2017
August 2017
July 2017
June 2017
May 2017
March 2017
February 2017