

c_dtc.doc

COLLABORATORS

	<i>TITLE :</i> c_dtc.doc		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		December 8, 2024	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	c_dtc.doc	1
1.1	c_dtc.doc	1
1.2	c.datatype/c.datatype	1

Chapter 1

c_dtc.doc

1.1 c_dtc.doc

c.datatype

1.2 c.datatype/c.datatype

NAME

c.datatype - data type for any c source

FUNCTION

This datatype is designed to display C and C++ source codes. It display's different parts of the C source in different style and color. At the moment these parts are :

- STANDARD - any text which, doesn't match the following parts
- COMMENT - any comment such like /* ... */ and // ...
- CPP - any C-PreProcessor keyword like "#define" or "#include"
- KEYWORD - any C/C++ keyword, which isn't handled explicitly
- STORAGE - extern, static, register, auto keywords
- TYPES - basic type keywords like int, char, long etc.
- TYPENAME - any name following a struct, union, class or enum
- STRING - any string or char literal
- NUMBER - any number constant decimal, hex

It uses a parser generated by bison with my yacc grammer. Because it's a parser, it may occur a parse error on some unusual source code. If this happens please send me a description of this parse error and maybe the input file. So I can fix this problem !

PREFS

The c.prefs file is searched first in PROGDIR:Prefs/DataTypes/ and then in Env:DataTypes/ with the following three ReadArgs() templates :

- CPART/A/K, PEN/N/K, R=RED/N/K, G=GREEN/N/K, B=BLUE/N/K, ITALIC/S, BOLD/S, UNDERLINED/S, TEXT/S

CPART is one of the explained cpart names like COMMENT or CPP.

PEN assigns the color with the pen number to the specified part
R,G,B defines a new color for the specified part. This color is
allocated with ObtainBestPenA(...,OBP_Precision,
PRECISION_ICON);

ITALIC,BOLD,UNDERLINED specifies the font style for the part
TEXT treat this CPART as normal text

- GLOBAL/A/S, TABLENGTH/N/K, NONESTEDCOMMENTS/S

GLOBAL indicates, that this line is a global setting. Note: The
/A/S combination isn't supported from ReadArgs(), so I check
it manually !

TABLENGTH - number of spaces to use for a tab !

NONESTEDCOMMENTS - disables nested comments

- INLINEARGS/A/S, KEYWORD/K/A, PATTERN/K/A, LINES/N/K

INLINEARGS indicates ,that this line is a inline args setting

KEYWORD defines the keyword to search for the inlined arguments for
the first lines. You can specify any string, which is compared
using strcmp(). After the keyword follows directly the pattern
specified by PATTERN.

PATTERN defines the pattern to get the inlined arguments. Any char
in the pattern must match the char in the text after the
keyword. To get any arguments you can specify a template using
the '%' char like in scanf() function. Following options are
supported :

%t - stands for the TABLENGTH

%c - stands for NESTEDCOMMENTS or NONESTEDCOMMENTS

LINES - specifies the number of lines from the beginning scanned
for inline arguments. Default is 10.

- USERTYPES/A/S, TYPES/M

USERTYPES indicates, that this line contains user defined types

TYPES defines words to treat as basic types like int or long etc.

AUTHOR

Stefan Ruppert
Windthorststrasse 5
65439 Floersheim am Main
Germany
EMail: ruppert@vs3.informatik.fh-wiesbaden.de

SEE ALSO

text.datatype
