

Syntax

Syntax

Syntax is for syntax highlighting. Because you can define the highlighting rules yourself, Syntax is suitable for any programming language or markup language like HTML.

Syntax Script

All color rules are stored in a script *Syntax.txt*. You can edit this script with any text editor including Syntax itself. When Syntax starts it loads *Syntax.txt* and sets the color rules for highlighting according to the script.

You will notice that the rules for the following languages are already defined:

HTML

Pascal

Java

JavaScript

/ Comment Lines

Every line in the script that begins with the / character is ignored by the rule compiler and treated as a comment line.

NEWRULE=HTML Tags

A *NewRule* statement shall always be the first line of the syntax script. It denotes the beginning of a new color rule. The part after the equals sign is the name of the new rule. This name will appear in the rules list. You can check and uncheck any rule.

Add New Rule before setting properties. You can have as many rules as you want in the script. All statements belonging to one rule shall be kept together in a block.

After a *NewRule* statement you must enter one of the three highlighting type statements:

- dtBeginEnd=yes
 - dtKeyShow=yes
 - dtBeginLineEnd=yes
-

dtBeginEnd=yes

Denotes that the rule will be of the begin marker and end marker type.

KBE=<>

Giving a begin and an end marker. You can also add a color. Begin marker, end marker and color shall be separated by a semicolon. Begin and end markers do not have to be single characters. They can also consist of more than one character.

The following are valid *KBE* statements.

KBE=<>;olive

KBE=(*);maroon

KBE=(*,*)

dtKeyShow=yes

Denotes that the rule will be of the Keywords type. Every keyword that you define in the rule will be highlighted.

KW=procedure

Denotes a keyword definition in a *dtKeyShow* rule. You can have as many *KW=keyword* statements in a rule as you want. Just place them underneath each other on separate lines. The following are valid *KW* statements:

KW=end;pink

KW=end;

KW=function

dtBeginLineEnd=yes

Denotes that the rule will be of the Begin to Line end

KLE=//

Denotes a marker definition in a *dtBeginLineEnd* definition.

The whole line after the marker characters *//* will be highlighted. You can add a color if you want:

KLE=//;silver

Enabled=yes

Denotes that the rule is enabled. You can always enable and disable a rule by checking/unchecking it in the rules list. Enter *Enabled=no* when you don't want the rule to be enabled.

You would typically enter this statement after the highlighting type statement.

DefKeyColor=blue

Denotes the default highlighting color of a key or marker in the rule. You can override this color per key or marker as explained further on. The color can be any of the following standard windows color name:

black, maroon, green, olive, navy, purple, teal, gray, silver, red, lime, yellow, blue, fuchsia, aqua

zOrder=1

Denotes the preference order of different rules that are enabled at the same time. A rule with a higher *zOrder* has preference over a rule with a lower *zOrder*.

Partword=yes

Denotes that also keywords being part of a word will be highlighted.

KeyCase=yes

Denotes that only words matching the Case of the Key will be highlighted. If you don't want this then use *KeyCase=no*.

