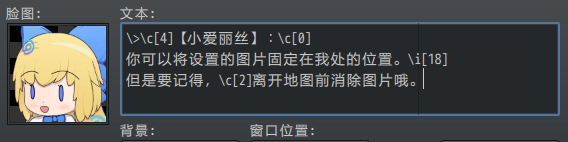
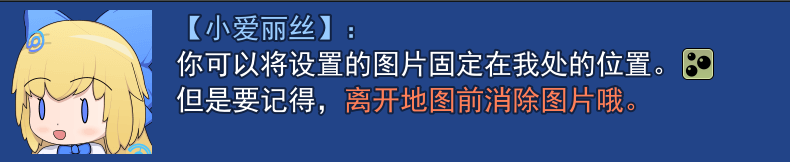
**概述**

**简介**

窗口字符，即绘制在窗口中的字符。

并不是所有字符在窗口中绘制的结果都一样，不同的字符有着特定的功能效果。





**作用域**

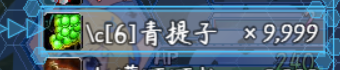
**1）留意窗口的支持情况**

注意，并不是所有文本都可以随意添加窗口字符。

具体要看实际游戏中，哪些窗口能够支持指定的 窗口字符 。

下图的窗口字符，因为不支持，所以直接原封不动地显示了。





**2）脚本相关**

脚本开发者注意，单张Sprite贴图中画的字符，**不能** 实现窗口字符的功能。

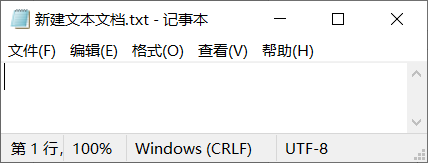
窗口字符顾名思义，只能在窗口中用的字符。

如果你希望贴图中能够实现窗口字符以及其各项功能，那么你必须建立一个窗口Window\_base来当做贴图使用。

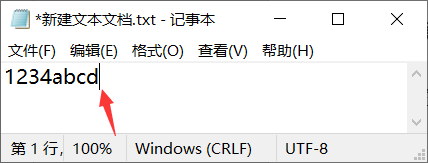
**窗口字符**

**原理**

你可以把一个窗口看成一个记事本，记事本里面有 **文本域和光标** 。

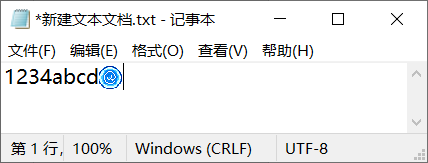


每输入一个字符，光标就能向前移动。

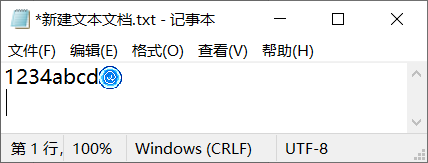


如果我输入一个图标\i[4]，那么就会变成下图。

（记事本并不能真的放图片，这里只是举例。）



如果我按一下回车键，那么就换行。



下面，我们可以总结出一些简单的结论：

1.窗口字符必须绘制在文本域中，每个字符的位置与光标有关系。

2.字符可以换行，换行位置与文本域相关。

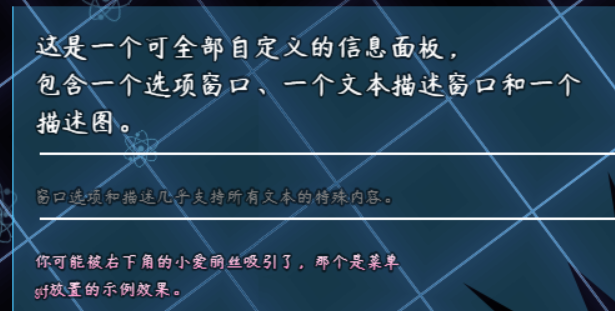
3.每个字符都有宽度，这个宽度由字符自己决定，图标字符也占宽度。

**定义**

**文本域**：是指绘制文本、图标、图片字符的区域，文本域有高度和宽度，字符如果越出了高宽边界，则会被遮挡。

**光标**：文本域中确定下一个字符位置的竖线，光标根据字符的宽度确定下一个位置，并且光标可以换行。

所有窗口都具有 文本域和光标 。



窗口有许多有趣的扩展字符，但是，它的底层依然是文本域和光标，不能跳出这个范围玩出新花样。

比如，窗口中**无法绘制**出动态字符、动态图片。

再比如，窗口**无法显示**超出文本域边界外的字符。

（选项窗口切换时，是将字符全部清理，再全重新绘制一遍，所以看起来好像能超边界）

**字符分类**

窗口中所有的字符，可以分成三类：一般字符、指代字符、效果字符。

**一般字符**：常规绘制的字符。

**指代字符**：用于表示一串字符串的字符，比如数字的值、名称等。

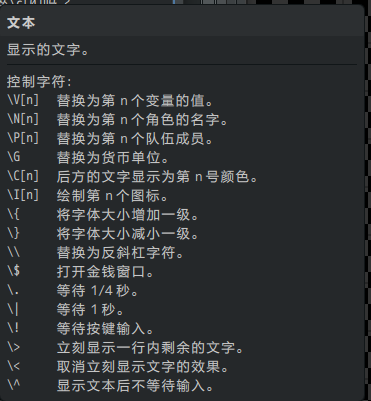
**效果字符**：用于表示执行一个具体功能的字符，比如消息输入、修改颜色、缩放、绘制图标等。（消息输入字符 属于 效果字符）



**Rmmv中默认字符**

所有窗口字符可以去看看[窗口字符表](#综合字符表)。

Rmmv中默认提供的窗口字符如下图所示：



**消息输入字符**：用于对话框一个输入/等待功能的字符，只在 对话框 中有效。

（上图中”\.”之后的字符都是消息输入字符）

**插件扩展字符**

**相关插件**

与窗口字符相关的插件如下：

Drill\_CoreOfColor 系统 - 颜色核心

YEP\_MessageCore 对话框 - 消息核心

Drill\_DialogSpecialCharSize 对话框 - 特殊字符大小控制器

Drill\_DialogTextAlign 对话框 - 文本居中

Drill\_DialogTextBigImage 对话框 - 大图片字符

Drill\_DialogReferenceChar 对话框 - 自定义指代字符

具体扩展的额外窗口字符可以去看看[窗口字符表](#综合字符表)。

**表达式（窗口辅助核心）**

表达式 和 窗口字符 是不同的两个结构，功能相似，但作用域不一样。

**1）写法区别**

表达式的格式为”<某功能:…:…>”。

窗口字符的格式为”\xxx”或”\xxx[10]”。

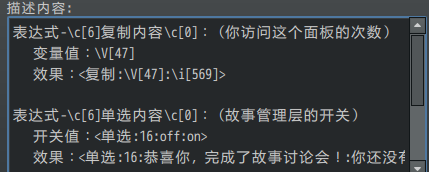
**2）只特定位置有效**

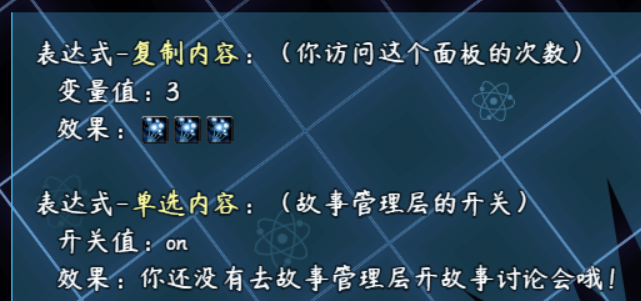
只有 窗口辅助核心的特定子插件 才能使用表达式。

在一般的窗口中**无效**。

相关子插件会对表达式的支持情况进行说明。

（下图为全自定义面板的表达式转换功能。）





**光标偏移字符**

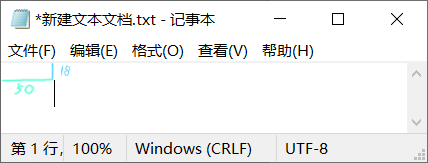
你需要先了解 [文本域与光标](#文本域与光标) ，不要急。

光标字符 \px[10] 和 \py[10]（单位像素） 可以将指定的图片绘制在文本域中。

注意，偏移位置不能填负数。

**1）位置影响**

偏移字符能够直接修改光标的位置，如下图，设置了\px[50]\py[18]，即光标左移50像素（绿色线），下移18像素（青色线）。



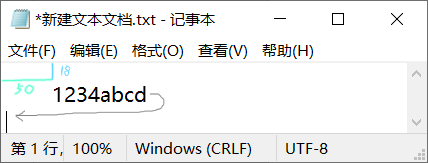
后面打出的字符，就是下面的情况：



**2）换行影响**

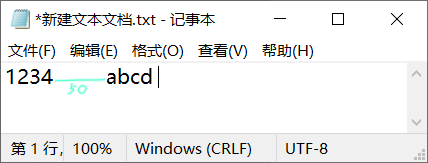
光标偏移后，能够直接对整行的位置造成影响，如下图。

注意，光标偏移的px像素值（绿色线）会清零，偏移的py值（蓝色线）会叠加。



**3）断行影响**

如果光标偏移字符写在中间，比如”1234\px[50]abcd”，可以造成”断开”的情况：



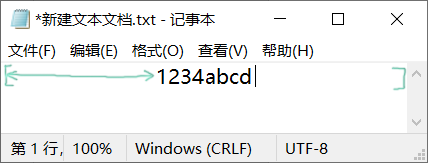
**文本居中**

你需要先了解 [文本域与光标](#文本域与光标) ，不要急。

文本居中，能根据当前的宽度

**1）居中原理**

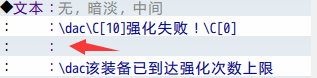
居中与光标偏移性质一样，识别窗口的宽度和字符的宽度后，自动偏移合适的距离。

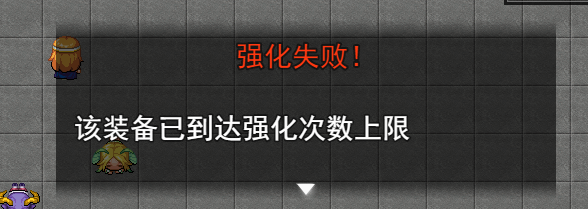


这也是不能同时写两个\dac\dac字符的原因，因为光标距离叠加后，会出界。

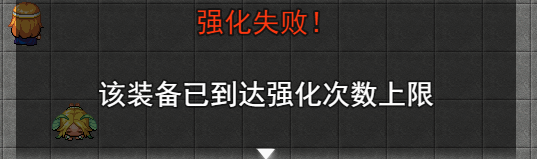
**2）使用注意**

在使用文本居中插件时，如果隔行进行居中，那么第二行要 加空格 ，不然就会造成居中失效。（居中插件不能识别空行情况，必须要有字符。）





加上空格后就能正常居中了。



**大图片字符**

你需要先了解 [文本域与光标](#文本域与光标) ，不要急。

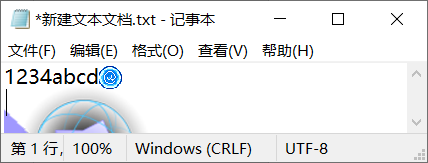
图片字符 \dimg[1]（直接绘制图片） 或 \dimg[10,10,1]（偏移一段距离后绘制图片） 可以将指定的图片绘制在文本域中。

由于是绘制在文本域，所以经常会出现以下现象：

**1）图片只有一半**

因为文本域的宽度不够，所以只能看到一部分，加高窗口即可。

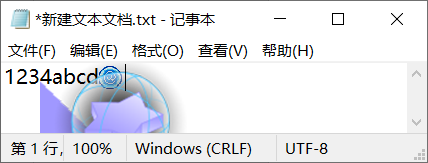
（另外，大图片不占字符宽度，所以光标仍然停留在原位置。）



**2）图片遮挡了前面的字符**

由于可以通过 \dimg[-80,0,1] 设置图片的偏移位置，所以在绘制时，当前光标的字符可以挡住前面的字符。属于正常现象。

当然，遮挡后，就不能再还原了，只能重新绘制。



所以一般都把 图片字符 放在最前面绘制，你也可以根据这个原理主动设置前面的字符的被遮挡效果。（对话管理层以此制作了 阅后即焚 的效果，你可以去看看。）

同样的，后面的图片字符，可以遮挡前面绘制的图片。

**字符串核心**

字符串的窗口字符可以与变量字符组合"\str[\v[21]]"。

通过变量的值来修改不同情况下的字符串。

**1）自定义技能名**

技能块元素可以支持自定义的字符串。（原rmmv窗口不能支持）

修改字符串，切换不同的名字。

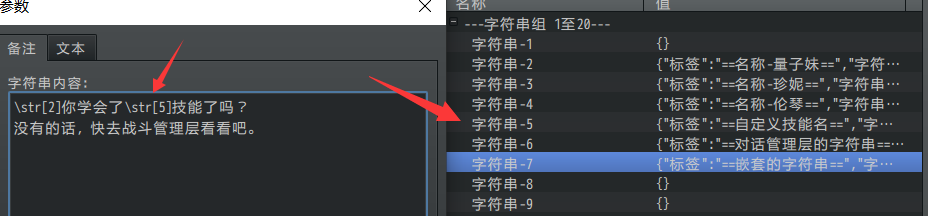


**2）自我嵌套**

字符串内部可以也用\str[]字符进行反复嵌套，但是要注意不能死循环嵌套。

比如3中有\str[2]，2中有\str[1]，而1中又有\str[3]，则会构成死循环嵌套。

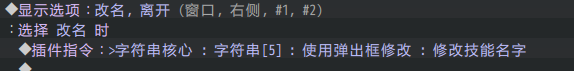
图中的字符串7，套用了前面2、5字符串中的内容。



**3）随时修改**

你可以使用插件指令随时修改字符串的内容，包括嵌套其他字符串的内容。

你还可以让用户自己输入字符串，实现更多组合功能。



**窗口字符表**

所有可用的窗口字符表如下表所示：

（斜杠后面的字符可大写也可小写）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **字符** | | **描述** | **作用范围** | **类型** | **所属插件** |
| \v[21] | | 替换为第n个变量的值。 | 所有窗口 | 指代字符 | Rmmv默认功能 |
| \n[5] | | 替换为第n个角色的名字。 | 所有窗口 | 指代字符 | Rmmv默认功能 |
| \p[1] | | 替换为第n个队伍成员的名字。 | 所有窗口 | 指代字符 | Rmmv默认功能 |
| \g | | 替换为货币单位。  （ 数据库>系统 中设置单位） | 所有窗口 | 指代字符 | Rmmv默认功能 |
| \\ | | 替换为'\'反斜杠字符本身。 | 所有窗口 | 指代字符 | Rmmv默认功能 |
| \c[0] - \c[31] | | 之后文字显示为第n个颜色。 | 所有窗口 | 效果字符 | Rmmv默认功能 |
| \i[136] | | 绘制第n个图标。 | 所有窗口 | 效果字符 | Rmmv默认功能 |
| \{ | | 将字体放大一级。 | 所有窗口 | 效果字符 | Rmmv默认功能 |
| \} | | 将字体缩小一级。 | 所有窗口 | 效果字符 | Rmmv默认功能 |
| \$ | | 打开金钱窗口。(对话中右上角出现一个金钱窗口,结束对话消失) | 只对话框 | 效果字符 | Rmmv默认功能 |
| \. | | 等待 15 帧，四分之一秒。 | 只对话框 | 消息输入字符 | Rmmv默认功能 |
| \| | | 等待 60 帧，一秒。 | 只对话框 | 消息输入字符 | Rmmv默认功能 |
| \! | | 等待按键输入。 | 只对话框 | 消息输入字符 | Rmmv默认功能 |
| \> | | 立刻显示后面文字。（一行内） | 只对话框 | 消息输入字符 | Rmmv默认功能 |
| \< | | 取消立刻显示。 | 只对话框 | 消息输入字符 | Rmmv默认功能 |
| \^ | | 显示文本后不等待输入。 | 只对话框 | 消息输入字符 | Rmmv默认功能 |
|  |  | | | | |
| \c[101] - \c[199] | | 之后文字显示为自定义颜色。  见 ”关于文本颜色.docx” 。 | 所有窗口 | 效果字符 | Drill\_CoreOfColor  系统-颜色核心 |
| \c[201] - \c[299] | | 之后文字显示为自定义渐变色。  见 ”关于文本颜色.docx” 。 | 所有窗口 | 效果字符 | Drill\_CoreOfColor  系统-颜色核心 |
| \dal | | 该字符之后的文字左对齐。  见 [文本居中](#文本居中) 。 | 所有窗口 | 效果字符 | Drill\_DialogTextAlign  对话框-文本居中 |
| \dac | | 该字符之后的文字居中。  见 [文本居中](#文本居中) 。 | 所有窗口 | 效果字符 | Drill\_DialogTextAlign  对话框-文本居中 |
| \dar | | 该字符之后的文字右对齐。  见 [文本居中](#文本居中) 。 | 所有窗口 | 效果字符 | Drill\_DialogTextAlign  对话框-文本居中 |
| \dimg[1] | | 在光标位置绘制大图片。  见 [大图片字符](#大图片字符) 。 | 所有窗口 | 效果字符 | Drill\_DialogTextBigImage  对话框-大图片字符 |
| \dimg[5,-5,1] | | 在光标位置偏移一点绘制大图片，不影响光标位置。  见 [大图片字符](#大图片字符) 。 | 所有窗口 | 效果字符 | Drill\_DialogTextBigImage  对话框-大图片字符 |
| \str[1] | | 替换为插件中自定义配置的字符串。  见 [字符串核心](#字符串核心) 。 | 所有窗口 | 效果字符 | Drill\_CoreOfString  系统-字符串核心 |
| \ac[1] | | 替换为第n个角色的职业名。 | 所有窗口 | 指代字符 | YEP\_MessageCore  对话框-消息核心 |
| \an[1] | | 替换为第n个角色的昵称（小名） | 所有窗口 | 指代字符 | YEP\_MessageCore  对话框-消息核心 |
| \pc[1] | | 替换为第n个队伍成员的职业名 | 所有窗口 | 指代字符 | YEP\_MessageCore  对话框-消息核心 |
| \pn[1] | | 替换为第n个队伍成员的昵称（小名） | 所有窗口 | 指代字符 | YEP\_MessageCore  对话框-消息核心 |
| \nc[1] | | 替换为第n个职业的名字 | 所有窗口 | 指代字符 | YEP\_MessageCore  对话框-消息核心 |
| \ni[1] | | 替换为第n个物品的名字 | 所有窗口 | 指代字符 | YEP\_MessageCore  对话框-消息核心 |
| \nw[1] | | 替换为第n个武器的名字 | 所有窗口 | 指代字符 | YEP\_MessageCore  对话框-消息核心 |
| \na[1] | | 替换为第n个防具的名字 | 所有窗口 | 指代字符 | YEP\_MessageCore  对话框-消息核心 |
| \ns[1] | | 替换为第n个技能的名字 | 所有窗口 | 指代字符 | YEP\_MessageCore  对话框-消息核心 |
| \nt[1] | | 替换为第n个状态的名字 | 所有窗口 | 指代字符 | YEP\_MessageCore  对话框-消息核心 |
| \ii[1] | | 替换为第n个物品的名字 + 图标 | 所有窗口 | 指代字符 | YEP\_MessageCore  对话框-消息核心 |
| \iw[1] | | 替换为第n个武器的名字 + 图标 | 所有窗口 | 指代字符 | YEP\_MessageCore  对话框-消息核心 |
| \ia[1] | | 替换为第n个防具的名字 + 图标 | 所有窗口 | 指代字符 | YEP\_MessageCore  对话框-消息核心 |
| \is[1] | | 替换为第n个技能的名字 + 图标 | 所有窗口 | 指代字符 | YEP\_MessageCore  对话框-消息核心 |
| \it[1] | | 替换为第n个状态的名字 + 图标 | 所有窗口 | 指代字符 | YEP\_MessageCore  对话框-消息核心 |
| \fr | | 重设之后文字的字体为默认。 | 所有窗口 | 效果字符 | YEP\_MessageCore  对话框-消息核心 |
| \fn<微软雅黑> | | 指定之后的文字字体。 | 所有窗口 | 效果字符 | YEP\_MessageCore  对话框-消息核心 |
| \fs[20] | | 指定之后的文字字体大小为n。 | 所有窗口 | 效果字符 | YEP\_MessageCore  对话框-消息核心 |
| \fb | | 之后的文字字体加粗。  （如果要还原，就加 \fr ） | 所有窗口 | 效果字符 | YEP\_MessageCore  对话框-消息核心 |
| \fi | | 之后的文字字体倾斜。  （如果要还原，就加 \fr ） | 所有窗口 | 效果字符 | YEP\_MessageCore  对话框-消息核心 |
| \oc[6] | | 之后的 文字边线 为第n个颜色。  (只能填rmmv默认颜色0-31) | 所有窗口 | 效果字符 | YEP\_MessageCore  对话框-消息核心 |
| \ow[3] | | 之后的 文字边线 厚度为n像素。  (标准为1像素) | 所有窗口 | 效果字符 | YEP\_MessageCore  对话框-消息核心 |
| \px[10] | | 断行，设置当前字符偏移的x值，单位像素。  见 [光标偏移字符](#光标偏移字符) 。 | 所有窗口 | 效果字符 | YEP\_MessageCore  对话框-消息核心 |
| \py[5] | | 断行，设置当前字符偏移的y值，单位像素。  见 [光标偏移字符](#光标偏移字符) 。 | 所有窗口 | 效果字符 | YEP\_MessageCore  对话框-消息核心 |
| \af[5] | | 该字符把对话框脸图 换成第n个角色脸图。 (只对话框有效) | 只对话框 | 效果字符 | YEP\_MessageCore  对话框-消息核心 |
| \pf[1] | | 该字符把对话框脸图 换成第n个队伍成员脸图。(只对话框有效) | 只对话框 | 效果字符 | YEP\_MessageCore  对话框-消息核心 |
| \w[60] | | 等待 n 帧。 | 只对话框 | 消息输入字符 | YEP\_MessageCore  对话框-消息核心 |
| \n<小爱丽丝> | | 建立一个靠左的姓名框。 | 只对话框 | 效果字符 | YEP\_MessageCore  对话框-消息核心 |
| \nc<小爱丽丝> | | 建立一个居中的姓名框。 | 只对话框 | 效果字符 | YEP\_MessageCore  对话框-消息核心 |
| \nr<小爱丽丝> | | 建立一个靠右的姓名框。 | 只对话框 | 效果字符 | YEP\_MessageCore  对话框-消息核心 |