## 概述

### 相关插件

基于核心：

◆Drill\_CoreOfWindowCharacter 窗口字符 - 窗口字符核心

插件本体：

◆Drill\_DialogCharOuterGlow 窗口字符 - 外发光效果

◆Drill\_DialogCharOuterBorder 窗口字符 - 描边效果

这里专门介绍描边效果和外发光效果。

## 外发光

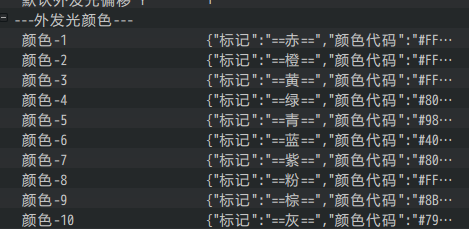
### 发光设置

#### 1）窗口字符

大部分窗口都能支持发光的窗口字符设置。

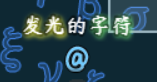
|  |  |
| --- | --- |
| **窗口字符** | **描述** |
| \dDCOG[1] | 之后的文字使用颜色1的外发光。 |
| \dDCOG[1:5:1:1] | 之后的文字使用颜色1，厚度5，偏移(1,1)的外发光。 |
| \dDCOGr | 之后的文字恢复外发光设置。 |
| \dDCOGoff | 之后的文字关闭外发光效果。 |
| \og[1] | 与\dDCOG[1]一样，简写形式。 |
| \os[5] | 之后的文字改变外发光厚度5，简写形式。 |
| \fr | 重置之后文字所有设置。包括恢复外发光设置。 |

发光的颜色需要在插件中配置，才能对应上。



如下图，前后包裹的文本，能够具有外发光效果。





|  |
| --- |
| 注意“\dDCOGr”与“\dDCOGoff”的区别，由于插件默认参数设置为所有文本都不发光，所以两者的效果是一样的。  但如果你的游戏一开始就开启了发白光，那么“\dDCOGr”是恢复一开始的白光，而“\dDCOGoff”是关闭发光。 |

#### 2）文本开关

插件提供了 所有文本和对话框 的自定义外发光开关，

能针对特定区域的文本进行发光自定义。

具体效果可以去 窗口字符管理层 看看发光介绍。



如果你同时用插件指令开启了发光，并且使用了窗口字符设置发光，那么优先按窗口字符的设置来。

### 描边设置

#### 1）窗口字符

大部分窗口都能支持描边的窗口字符设置。

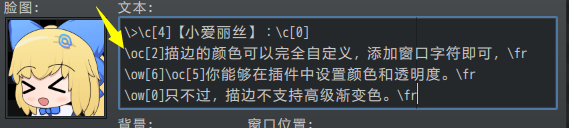
|  |  |
| --- | --- |
| **窗口字符** | **描述** |
| \dDCOB[1] | 之后的文字使用描边颜色1。 |
| \dDCOB[1,4] | 之后的文字使用描边颜色1，厚度4。 |
| \dDCOBr | 之后的文字恢复描边设置。 |
| \dDCOBoff | 之后的文字关闭描边效果。 |
| \oc[1] | 之后的文字使用描边颜色1，简写形式。 |
| \ow[4] | 之后的文字描边厚度为4，简写形式。 |
| \fr | 重置之后文字所有设置。包括恢复描边设置。 |

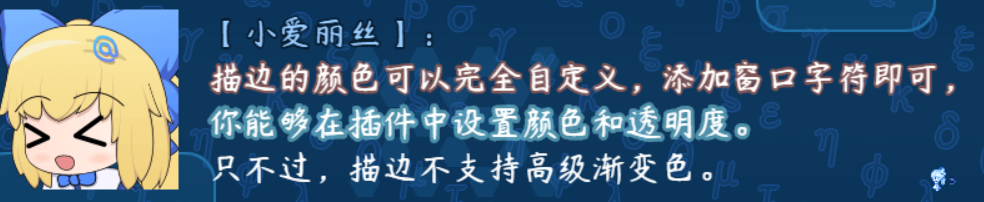
描边的颜色需要在插件中配置，才能对应上。

（描边可以设置颜色透明度）



如下图，前后包裹的文本，能够具有描边效果。

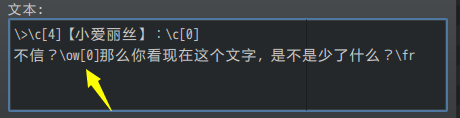




|  |
| --- |
| 游戏默认设置就有一层厚度为4的黑色半透明描边，用于突出文本，一般玩家很难察觉到，但如果文字窗口是白色背景，描边与不描边的区别就明显了。 |

如果你可以设置厚度0去掉描边，但最好使用“\dDCOBoff”字符。

（作者我已作了兼容，二种字符的实际效果是完全一样的）

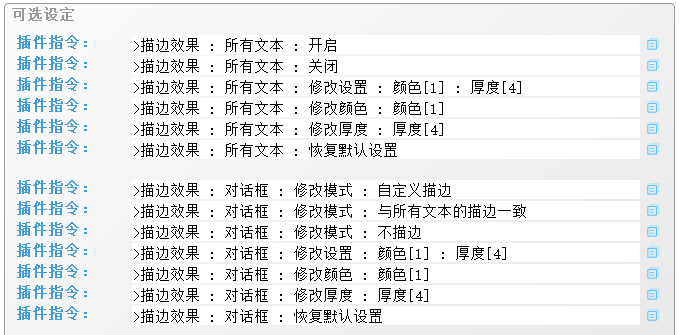


#### 2）文本开关

插件提供了 所有文本和对话框 的自定义描边开关，

能针对特定区域的文本进行描边自定义。

具体效果可以去 窗口字符管理层 看看描边介绍。



如果你同时用插件指令开启了描边，并且使用了窗口字符设置描边，那么优先按窗口字符的设置来。

### 效果叠加

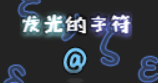
#### 1）字体效果叠加

外发光或描边可以与下面插件效果叠加：

◆Drill\_DialogFontFace 窗口字符 - 字体管理器

比如 外发光+字体。



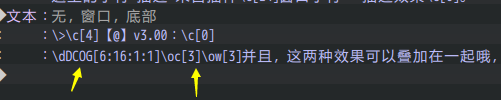


比如 描边+字体。





值得一提的是，外发光与描边本身能够相互叠加。





#### 2）轮播文本效果叠加

外发光或描边可以与下面插件效果叠加：

◆Drill\_EventText 行走图 - 事件漂浮文字

该插件具有轮播效果，每间隔一段时间变化播放不同的窗口字符。

叠加后，能够实现不断变化的 ”发光” 效果。

#### 3）其它效果叠加

除了上述的效果，外发光与描边还能与其它插件的窗口字符效果叠加。

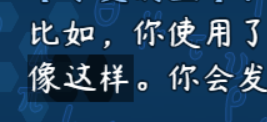
具体你可以去看看窗口字符表，一个个试试看。

### 文本域边界问题

如果你给 字符块、窗口文本域 设置了外发光或描边，注意不要设置太厚。

因为文本域是一个矩形的绘制区域，是有长宽边界的。

如果外发光或描边太厚，容易看出被切割的边沿。

## 从零开始设计（DIY）

### 设计一个外发光文字

#### 1. 设置一个目标

首先你需要先设想或看到一个目标，来根据目标的字体情况来设想你的字体。

比如2022年6月5日作者我今天偶然跑去看音游录播，

初音未来游戏《Project DIVA MEGA 39's +》。

载入界面时出现了下面这些艺术字。于是作者我尝试模仿它：



#### 2. 结构规划/流程梳理

在规划窗口字符之前，你需要先了解有哪些窗口字符，建议去 窗口字符管理层 转一圈，这样能快速学习并掌握到大部分能搭配的窗口字符。

首先，**外发光**。

需要设置文字外发光，颜色为青色或蓝色。

然后，**字体**。

目标字体大小很大，需要用到窗口字符核心来调整字体。

而且目标字体非常粗，可以用上网找的免费商用的英文粗体字体。

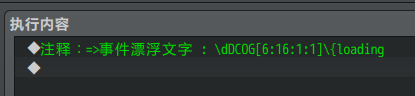
最后，**描边**。

游戏中所有绘制的字体，都默认有一层黑色描边边框，这里我们需要去掉它，所以用到了描边插件。

#### 3. 快速设置

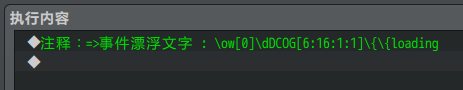
这里，我们先把外发光和字体大小设置上，然后看看效果。

（你可以用 \{ \} 扩大缩小字符，也可以用 \fs[24] 指定字符大小）



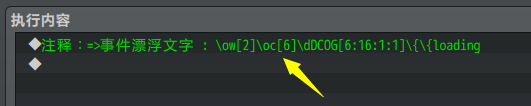


然后，再弄大点，设置描边厚度为0。看看效果：





单独的白色和外发光，颜色太淡了，还是要再加一层蓝色的描边才好些。





#### 4. 字体设置

下面需要字体插件：

◆Drill\_DialogFontFace 窗口字符 - 字体管理器

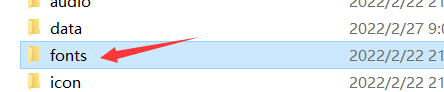
首先，要找到一款能买免费商用，且粗体与上面相当的字体。

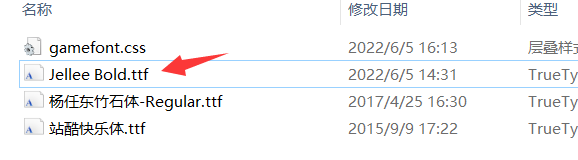
于是我找到了字体：Jellee Bold 。



|  |
| --- |
| 配置字体的方式，可以结合“23.窗口字符 > 关于字体管理器.docx”来看。 |

将这个字体文件放入font文件夹。





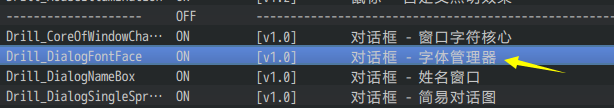
用记事本打开 gamefont.css文件，添加下图的文本结构。

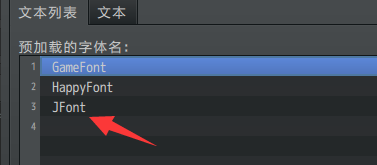
为方便减少后面写字符的数量，起名为 JFont 。



|  |
| --- |
| 注意，字体名称，是双击打开字体文件后，显示的字体名称。  **不是** 字体的文件名 。（有时候下载来的文件名和字体名可能不一样） |

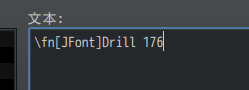
在预加载中添加新的字体，JFont名称。





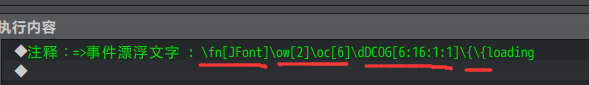
这样就配置完成了。

你可以在对话框中，使用窗口字符”\fn[JFont]”，看看字体载入后的效果。





最后，将这个字体效果添加到之前写的 窗口字符 列表中。



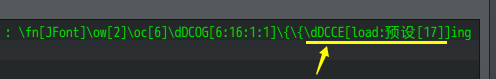
这样就基本完成了。



#### 5. 其它效果组合

设置之后，你还可以将此效果与跳动字符的插件结合：

◆Drill\_DialogCharContinuedEffect 窗口字符 - 字符块持续动作效果



### 设计一个来回闪烁的展板文字

你可以去文档 ”7.行走图 > 关于事件漂浮文字.docx” 的章节，

采用 轮播文本+外发光 来设计一个类似霓虹灯的文字。

