**MageINI GM上的超强ini文件脚本**

Mage松仁玉米

　　MageINI是一款基于GM8和GMS的ini扩展脚本，所有函数均通过GM内置的函数完成，支持任意位置的INI文件操作，支持从文件载入，或直接从字符串载入ini，支持自定义ini标识符等等强大的功能。

**关于函数说明中的参数写法：**

　　在下面的函数说明中，每一个函数我都会说明函数的名字，参数，以及返回值，还有这个函数的功能，以及注意事项，说明函数的名字参数和返回值我会使用这种格式：

函数名(参数1类型 参数1描述,参数2类型 参数2描述,…) : 返回值类型 返回值描述

　　其中，有些函数的参数可能会是string型或real型，在说明中我将会使用string|real来表示。返回值也有返回string或real的情况，同样使用这种方式表示。

　　为了方便大家看懂说明，一些ini常用的术语，我会使用“片段”来代表ini中的section，使用“键”来代表ini中的key，使用“值”来代表ini中的value。一个标准的ini文件结构如下：

[section]

key1=value1

key2=value2

key3=value3

;this is a common statement

　　这里的section就是一个片段，key1，key2，key3均为键，value1，value2，value3均表示该键下的值。

**函数说明：**

INI\_init()

　　初始化函数，在游戏开始的时候执行一次。（MD这个初始化函数都不想多说了，是个扩展应该都会带上这个东西。）

INI\_open(string fname) : real iniHandle

　　打开一个ini文件，参数fname是ini文件的路径，支持相对路径和绝对路径，打开后返回ini文件的句柄，以后在操作该文件的时候会用到这个句柄。如果文件存在则直接打开读取，文件不存在会在该ini文件关闭的时候创建一个新的文件。如果返回-1则表示发生错误，使用INI\_get\_error函数可以获取错误信息。

INI\_load\_from\_string(string str) : real iniHandle

　　直接从字符串中载入ini，如果你有一个字符串，这个字符串完全遵守ini的标准，你可以直接将这个字符串载入，该函数返回ini的句柄，以后再操作这个ini的时候会用到这个句柄。如果返回-1则表示发生错误，使用INI\_get\_error函数可以获取错误信息。

INI\_get\_error() : string errorMessage

　　返回当前是否有错误，返回值为错误描述的字符串。

INI\_close(real iniHandle) : real|string 1|iniString

　　关闭一个ini句柄，当对ini操作结束的时候，使用这个函数来关闭并释放ini句柄。当你的ini是使用INI\_open函数来打开一个文件时，这个函数会将操作后的ini文件进行保存，也就是说，如果不使用关闭函数，ini文件操作后是不会被保存在你的硬盘上的。保存完后，该函数会返回1。如果你使用的是INI\_load\_from\_string载入了一个字符串，那么这个函数会返回操作后的ini字符串而不是返回一个实数。注意：使用这个函数所保存的ini文件或返回的ini字符串都会把源ini中的注释移除！

INI\_read(real iniHandle,string section,string|real key,string default) : string value

　　当key参数为string型时，从指定的ini句柄中指定的片段的键读取一个字符串，当键或片段不存在时，会返回default值。弱该片段下存在多个key键时，会返回第一个key键的值。当key参数为real型时，从指定的ini句柄中指定的片段读取第key个键的值。当key为1时会读取该片段下第一个键的值。Key为2时会读取该片段下第二个键的值，以此类推，如果key的值大于该片段下存在键的总数或取值不当（key<=0时），则会返回default值。

INI\_read\_n(real iniHandle,string section,string key,real n,string default) : string value

　　从指定的ini句柄中指定的片段的第n个key键读取一个字符串，这个函数只适用于一个片段下有多个相同的键时才使用。如果片段或键不存在或n超出了key键的数量，则返回default。

INI\_write (real iniHandle,string section,string key,string value)

　　在指定的ini句柄的指定片段的键下写入一个值，如果该片段下存在这个键，将会覆盖，如存在多个则会覆盖第一个。

INI\_add(real iniHandle,string section,string key,string value)

　　在指定的ini句柄的指定片段的键下写入一个值，如果片段下存在这个键，则会继续追加，并不会产生覆盖操作。

INI\_section\_count(real iniHandle) : real count

　　返回指定的ini句柄下一共有多少个片段。

INI\_key\_count(real iniHandle,string section) : real count

　　返回指定的ini句柄下的指定片段下共有多少个键。

INI\_key\_same\_count(real iniHandle,string section,string key) : real count

　　返回指定的ini句柄下的指定片段下共有多少个名字为key的键。

INI\_section\_delete(real iniHandle,string section)

　　删除指定ini句柄下的section片段，同时也会删除该片段下的所有键。

INI\_key\_delete(real iniHandle,string section,string key)

　　删除指定ini句柄下section片段内所有名字为key的键。

INI\_key\_delete\_n(real iniHandle,string section,string key,real n)

　　删除指定ini句柄下section片段内第n个名字为key的键。如果n<=0或n大于该键的总数目则不会删除任何键。

INI\_get\_string(real iniHandle) : string iniStr

　　输出指定ini句柄的ini格式字符串。使用该函数会清除原ini字串中的所有注释。

INI\_save\_file(real iniHandle,string fname)

　　将ini句柄以ini的格式保存在fname文件中。使用该函数会清除原ini的所有注释。

**自定义ini标识符：**

　　你可以自定义你的ini标识符，所谓的ini标识符就是片段名的中括号＂[]＂，键和值的连接符号＂=＂使用这一扩展你可以轻松的改变这些标识符。这些符号不仅只影响你将ini存放到文件或者是输出ini字符串，它也会影响你读取ini文件。一个标准的ini文件，如果你打算使用自定义的标识符，则不会读入任何数据。

这里提供了几个全局变量让你改变ini的标识符：

　　INI\_SECTION\_SIGN\_LEFT ： INI片段的左括号，默认值为＂[＂

　　INI\_SECTION\_SIGN\_RIGHT ： INI片段的右括号，默认值为＂]＂

　　INI\_ITEM\_SIGN ： INI的键与值的连接符，默认值为＂=＂，有些ini文件可能会是＂:＂，对于这种ini文件，在打开之前一定要先更改此变量为＂:＂。

　　INI\_NOTE\_SIGN ： INI的注释符，当ini文件的一行的第一个字符是这个符号时，该行不会被程序所解析。

最后再说几句：

如果你在这个扩展中发现了啥BUG的话，欢迎联系作者，QQ386083738，本扩展完全开源，各位也可以自己修复一些。

GMS上的话，是理论上支持的，因为函数内部用的都是ds\_list系列，ds\_map系列以及file\_text系列，GMS应该也支持。

Git地址：<https://git.coding.net/mage/MageINI.git>

官方网站：<http://www.magecorn.com>

Gamemaker开发者之家：<http://www.gamemake.org>