

文件对话框

file_explorer_openfile

一个简单的公共打开对话框, 调用之后将打开一个打开文件的对话框, 让用户选择需要打开的文件, 使用ANSI编码

char * path 初始地址
char * filter 正则表达式匹配文件 (例如, 所有文件使用".*")
char * return_path 返回的地址, 请务必使用MAX_PATH大小的char型数组

如果path输入的是完整的文件路径, 隐含的意思是先打开所在目录, 再定位光标到文件上

打开文件选择对话框

```
int file_explorer_openfile( char * path, char * filter, char * return_path, const char * custom_msg );
```

参数	char * path	初始路径
	char * filter	匹配文件名的正则表达式
	char * return_path	返回的地址, 请务必使用MAX_PATH大小的char型数组
	const char * custom_msg	用户显示的数据, 显示在打开对话框的标题栏上的数据
返回值	int	0获取文件成功, 其他失败

1. 只有返回值为0的时候, return_path中的数据才是有效的. 返回其他值的情况, 可能是路径不存在, 或者是用户选择了取消打开操作.

1. 正则表达式的应用如下例子: 选择JPG和BMP文件

"\\.(JPG|jpg|BMP|bmp)\$"

用回调的方式打开文件, 在用户选定文件后, 回调函数就被调用, 其他参数意义同

file_explorer_openfile

```
int file_explorer_openfile_shell( char * path, char * filter, char * return_path, const char * custom_msg, int (*func)(char *path) );
```

参数	char * path	初始路径
	char * filter	匹配文件名的正则表达式
	char * return_path	返回的地址, 请务必使用MAX_PATH大小的char型数组
	const char * custom_msg	用户显示的数据, 显示在打开对话框的标题栏上的数据
	int (*func)(char *path))	用户选择了文件后, 回调用户函数
返回值	int	0 获取文件成功, 其他失败

1. 用户回调函数的原型为

```
int (*func)(char *path)
```

path是用户选定的文件路径
返回值0----继续浏览文件
1----退出文件浏览

2. 其他参数意义同file_explorer_openfile