

# BricMusic——一款轻便的音乐播放器

学习 · 刚刚 0阅读 · 0点赞



BricMusic是轻量的，跨平台的音乐播放器，在Windows10和Ubuntu20.04均实机试验，这个文档是为BricMusic for Windows而写的。

她真的很小巧玲珑~!—



logo

## 配置文件

Windows下所有的配置文件和附加脚本都需要被保存在BricMusic.exe的同级目录下，其中settings.lua是必须的。

# 详解settings.lua

BricMusic的配置文件settings.lua是一个基于Lua语言的脚本文件。

- **AudioPaths**： **必须存在的变量**，这是一个作用域为全局的表，内部只允许字符串，作为BricMusic的默认播放目录。

这里**值得一提**的是，文件路径和**url**都是可以作为播放路径添加到这里的。（注意需要使用/而非\）

```
AudioPaths = {  
"C:/Users/Brick/Music/example.mp3",  
"http://127.0.0.1/Music/example.mp3",  
}
```

- **ThemeColor**： 这是一个作用域为全局的表或字符串，会改变播放器的主色调

--这两种方法均是可以使用的

```
ThemeColor = "#E799B0"  
ThemeColor = {  
Red = 0xE7,  
Green = 0x99,  
Blue = 0xB0  
}
```

- 这里是有一些彩蛋的，试试**那五个姑娘**
- **DefaultMenuPath**： 这个变量本身并不会被播放器使用，仅出现在示例脚本中，与下面的**Extension**配合使用，它需要指名一个含音频文件的路径
- **Extension**： 这个变量本身并不会被播放器使用，仅出现在示例脚本中，找出DefaultMenuPath中包含表内后缀名的文件

```
AudioPaths = {}
```

--这段代码会在"C:/Users/Brick/Music"这个路径下寻找所有的mp3、flac、wav作为后缀的文件并加入到AudioPath中

```
DefaultMenuPath = "C:/Users/Brick/Music"
```

```
Extension = {  
"mp3",  
"flac",  
"wav"  
}  
require("lfs")
```

```
local re = {}
```

```
for i, v in pairs(Extension) do
    re[i] = ".*%." .. v
end
for file in lfs.dir(DefaultMenuPath) do
    if file ~= "." and file ~= ".." then
        for i, v in pairs(re) do
            if string.find(file, v, 0) then
                local f = DefaultMenuPath .. "/" .. file
                table.insert(AudioPaths, f)
            end
        end
    end
end
end
end
```



## 如果我想放一首歌听

- 如果设置好了AudioPaths，直接打开BricMusic.exe就会使用播放默认列表。
- 你可以用BricMusic.exe作为音频的默认文件

找到一个音频文件，右键->打开方式->（设置为默认打开方式->）选择其他应用->（更多应用->）在这台电脑上查找其他应用->找到BricMusic.exe->双击她！

也可以捻一撮选定多个音频文件拖到她头上喂到她嘴里，你选定的所有文件，就会是这次打开的临时播放列表，你也可以试试在有BricMusic启动的时候再使用BricMusic打开音频文件，多了解下，自然会跟她熟稔了（她已经是我的形状子）。

- 写一个.blu歌单文件，然后打开它
- 保存现在的播放列表为一个.blu文件，在以后打开它(Shoobie do lang lang)



## 怎么写一个.blu歌单文件

- 基础版

与settings.lua相类似，这里也需要一个全局的字符串的表**MenuList**作为播放列表，填入文件路径或者url均可。

也可以右键右下角托盘图标，把当前的文件保存下来，这都可以。

- 进阶版

这里就是给会写Lua的同学准备的了。

.blu文件会共处同一个虚拟机环境，但是只有第一个被加载的.blu会支持下面这个很有趣的功能：

```
function func(str)
    local path = str
    function retfun()
        return path
    end
end
```

```
MenuList = {  
  func("C:/Users/Brick/Music/example.mp3"),  
}
```

懂哥已经知道要干什么了

第一个被加载的.blu中的MenuList除了音频文件的路径，还支持存放无参数且以音频文件路径为返回值的函数（回调函数/闭包），这么做的目的是，是让

**.blu支持动态的执行。**

## 图标都是干啥用的

她都已经是我的形状了，我相信怎么用还是显而易见的，右下角托盘有显示，退出，保存三种功能。再小图标展开的时候，你可以滑动滚轮以增大或减小声音，点击图标切换静音和非静音的状态。

## 联系我

除了在发布平台上，你还可以给huahui36113@163.com发邮件。

如果有音频相关的BUG，可以直接把播放出问题的音频和发布的版本号发我。

欢迎大家帮我提供测试。

## 版权相关的

使用动态链接库的协议附在发布包里了，未经许可，软件不可以租借、转卖、从事商业用途，不得使用本软件从事非法行为，使用软件出现的任何损失，与本人无关，不对此负责。

别卖，别搞事，别扯上我

刘福百川

2022.6.22

