**Pygame 官方文档 - pygame.scrap**

pygame.scrap

pygame模块用于支持剪切板（ clipboard)  
[pygame.scrap.init](https://blog.csdn.net/Enderman_xiaohei/article/details/88536254#init%28%29) - 初始化scrap模块  
[pygame.scrap.get](https://blog.csdn.net/Enderman_xiaohei/article/details/88536254#get%28%29) - 从剪贴板获取指定类型的数据  
[pygame.scrap.get\_types](https://blog.csdn.net/Enderman_xiaohei/article/details/88536254#get_types%28%29) - 获取可用剪贴板类型的列表  
[pygame.scrap.put](https://blog.csdn.net/Enderman_xiaohei/article/details/88536254#put%28%29) - 将数据放入剪贴板  
[pygame.scrap.contains](https://blog.csdn.net/Enderman_xiaohei/article/details/88536254#contains%28%29) - 检查剪贴板中是否有某种类型  
[pygame.scrap.lost](https://blog.csdn.net/Enderman_xiaohei/article/details/88536254#lost%28%29) - 检查剪贴板当前是否归应用程序所有  
[pygame.scrap.set\_mode](https://blog.csdn.net/Enderman_xiaohei/article/details/88536254#set_mode%28%29) - 设置剪贴板访问模式

实验！：这意味着这个API可能会在以后的pygame版本中发生变化或消失。 如果你使用它，你的代码将破坏下一个pygame版本。  
scrap模块用于从剪贴板中获取和放置东西。 因此，您可以在pygame和其他应用程序类型之间复制和粘贴内容。 它定义了一些基本的自有数据类型

SCRAP\_PPM

SCRAP\_PBM

SCRAP\_BMP

SCRAP\_TEXT

放在剪贴板中，并允许使用自己定义的剪贴板类型。 SCRAP\_PPM，SCRAP\_PBM和SCRAP\_BMP适用于与其他应用程序共享的surface缓冲区，而SCRAP\_TEXT是纯文本剪贴板类型的别名。

SCRAP\_ \*类型引用以下MIME类型并注册这些类型以及此类数据的默认操作系统类型：

SCRAP\_TEXT text / plain 用于纯文本

SCRAP\_PBM image / pbm 用于PBM编码的图像数据

SCRAP\_PPM image / ppm 用于PPM编码的图像数据

SCRAP\_BMP image / bmp 用于BMP编码的图像数据

根据平台的不同，在将数据放入剪贴板时会自动注册其他类型，以保证与其他应用程序的一致共享行为。 以下列出的类型可以用作字符串，以传递给相应的pygame.scrap模块函数。  
对于Windows平台，将自动支持其他类型并解析为其内部定义：

text / plain; charset = utf-8用于UTF-8编码的文本

audio / wav 用于WAV编码音频

image / tiff 用于TIFF编码的图像数据

对于X11平台，自动支持其他类型并解析为其内部定义：

UTF8\_STRING用于UTF-8编码的文本

text / plain; charset = utf-8 用于UTF-8编码的文本

COMPOUND\_TEXT 用于COMPOUND文本

如前所述，您可以为剪贴板定义自己的类型，但这些类型可能无法被其他应用程序使用。 因此，粘贴到剪贴板中的数据

pygame.scrap.put ("own\_data", data)

只能由应用程序使用，应用程序在剪贴板中查询“own\_data”类型。  
pygame 1.8的新特性。 目前仅适用于Windows，X11和Mac OS X. 在Mac OSX上，目前仅有文本可用 - 下一版本将支持其他类型。

pygame.scrap.init() [↶](https://blog.csdn.net/Enderman_xiaohei/article/details/88536254#title)

初始化scrap模块  
init () -> None  
如果失败，尝试初始化scrap模块并引发异常。 请注意，此模块需要设置显示surface，因此您必须确保先前使用pygame.display.set\_mode()获取了一个。  
[搜索pygame.scrap.init的示例](https://github.com/search?q=pygame.scrap.init&type=Code)

pygame.scrap.get() [↶](https://blog.csdn.net/Enderman_xiaohei/article/details/88536254#title)

从剪贴板获取指定类型的数据  
get (type) -> bytes  
从剪贴板返回指定类型的数据。 数据以字节字符串形式返回，可能需要进一步处理，例如解码为Unicode。 如果没有可用的传递类型的数据，则返回None。

text = pygame.scrap.get (SCRAP\_TEXT)

if text:

# 用它来做事情

else:

print "剪贴板中似乎没有文字。"

[搜索pygame.scrap.get的示例](https://github.com/search?q=pygame.scrap.get&type=Code)

pygame.scrap.get\_types() [↶](https://blog.csdn.net/Enderman_xiaohei/article/details/88536254#title)

获取可用剪贴板类型的列表  
get\_types () -> list  
获取具有可用剪贴板类型标识符的字符串列表。 可以在scrap.get()方法中使用每个标识符来获取特定类型的剪贴板内容。 如果剪贴板中没有数据，则返回空列表。

types = pygame.scrap.get\_types ()

for t in types:

if "text" in t:

#内容中包含“text”一词。

# 它可能是文本，所以打印出来。

print pygame.scrap.get(t)

[搜索pygame.scrap.get\_types的示例](https://github.com/search?q=pygame.scrap.get_types&type=Code)

pygame.scrap.put() [↶](https://blog.csdn.net/Enderman_xiaohei/article/details/88536254#title)

将数据放入剪贴板  
put(type, data) -> None  
将特定剪贴板类型的数据放入剪贴板。 数据必须是字符串缓冲区。 类型是一个字符串，用于标识放入剪贴板的数据类型。 这可以是本地的SCRAP\_PBM，SCRAP\_PPM，SCRAP\_BMP或SCRAP\_TEXT值之一，也可以是自己的字符串标识符。  
如果无法将内容放入剪贴板，则该方法会引发异常。

fp = open("example.bmp"，"rb")

pygame.scrap.put(SCRAP\_BMP，fp.read())

fp.close()

＃现在，您可以从其他应用程序中的剪贴板获取图像数据。

pygame.scrap.put(SCRAP\_TEXT，"要复制的文本")

pygame.scrap.put("纯文本"，"要复制的文本")

[搜索pygame.scrap.put的示例](https://github.com/search?q=pygame.scrap.put&type=Code)

pygame.scrap.contains() [↶](https://blog.csdn.net/Enderman_xiaohei/article/details/88536254#title)

检查剪贴板中是否有某种类型  
contains (type) -> bool  
如果传递的类型的数据在剪贴板中可用，则返回True，否则返回False。

if pygame.scrap.contains (SCRAP\_TEXT):

print "剪贴板中有文字。"

if pygame.scrap.contains ("own\_data\_type"):

print "剪贴板中有东西。"

[搜索pygame.scrap.contains的示例](https://github.com/search?q=pygame.scrap.contains&type=Code)

pygame.scrap.lost() [↶](https://blog.csdn.net/Enderman_xiaohei/article/details/88536254#title)

检查剪贴板当前是否归应用程序所有  
lost() -> bool  
如果剪贴板当前归pygame应用程序所有，则返回True，否则返回False。

if pygame.scrap.lost ():

print "我这里再也没有内容了-。 剪贴板由其他人使用。"

[搜索pygame.scrap.lost的示例](https://github.com/search?q=pygame.scrap.lost&type=Code)

pygame.scrap.set\_mode() [↶](https://blog.csdn.net/Enderman_xiaohei/article/details/88536254#title)

设置剪贴板访问模式  
set\_mode(mode) -> None  
设置剪贴板的访问模式。 这仅适用于X11环境，其中鼠标选择（SRAP\_SELECTION）和剪贴板（SCRAP\_CLIPBOARD）的剪贴板模式是可用的。 在其他环境中将模式设置为SCRAP\_SELECTION不会产生任何差异。  
如果传递与SCRAP\_CLIPBOARD或SCRAP\_SELECTION不同的值，则将引发ValueError。  
[搜索pygame.scrap.set\_mode的示例](https://github.com/search?q=pygame.scrap.set_mode&type=Code)

以上文档，自己翻译，可能有误，可参考：[pygame.scrap](https://www.pygame.org/docs/ref/scrap.html)