

关于本文档

文档使用 Markdown 编写，使用 [Docsify](#) 解析为网页，[源码](#) 在 github 上开源，目前由开发者维护。

API 稳定性

由于 AutoX.js 处于活跃的更新和开发状态，API 可能随时有变动，我们用 **稳定性** 来标记模块、函数的稳定性。

这些标记包括：

稳定性：弃用

弃用的函数、模块或特性，
在未来的更新中将很快会被移除或更改。应该在脚本中移除对这些函数的使用，以免后患。



稳定性：实验

实验性的函数、模块或特性，
在未来的更新中可能会更改或移除。应该谨慎使用这些函数或模块，或者仅用作临时或测试用途。



稳定性：稳定

稳定的函数、模块或特性，
在未来的更新中这些模块已有的函数一般不会被更改，会保证后向兼容性。

如何阅读本文档

先看一个例子，下面是 [基于控件的操作](#) 的章节中 `input` 函数的部分说明。

`input([i,]text)`

- `i` {number} 表示要输入的为第 $i + 1$ 个输入框
- `text` {string} 要输入的文本

`input` 表示函数名，`([i,]text)` 表示要传入两个参数：`i`，`text`。`[i,]` 表示 `i` 是可选参数，即 `i` 可有可无。

下方是函数说明，`i` 表示参数名称，`{number}` 表示参数 `i` 的类型为数值，`表示要输入...` 是对参数 `i` 的详细说明

js

```
//执行这个语句会在屏幕上的第2个输入框处输入"啦啦啦"。  
input(1, "啦啦啦")
```

js

```
//这个语句会在屏幕上所有输入框输入"嘿嘿嘿"。  
input("嘿嘿嘿");
```

我们再看第二个例子。图片和图色处理中 `detectsColor` 函数的部分说明。

`images.detectsColor(image, color, x, y[, threshold = 16, algorithm = "diff"])`

- `image` {Image} 图片
- `color` {number} | {string} 要检测的颜色
- `x` {number} 要检测的位置横坐标
- `y` {number} 要检测的位置纵坐标
- `threshold` {number} 颜色相似度临界值，默认为16。取值范围为0~255。
- `algorithm` {string} 颜色匹配算法，包括：
 - `equal` : 相等匹配，只有与给定颜色 `color` 完全相等时才匹配。
 - `diff` : 差值匹配。与给定颜色的R、G、B差的绝对值之和小于 `threshold` 时匹配。
 - `rgb` : rgb欧拉距离相似度。与给定颜色 `color` 的rgb欧拉距离小于等于 `threshold` 时匹配。

- **rgb** : 加权rgb欧拉距离匹配(**LAB Delta E**)。
- **hs** : hs欧拉距离匹配。hs为HSV空间的色调值。

threshold = 16 表示如果不指定该参数, 则 **threshold** 的值为 **16**

js

```
images.detectsColor(captureScreen(), "#112233", 100, 200)  
//相当于  
images.detectsColor(captureScreen(), "#112233", 100, 200, 16, "r
```



js

```
images.detectsColor(captureScreen(), "#112233", 100, 200, 64)  
//相当于  
images.detectsColor(captureScreen(), "#112233", 100, 200, 64, "r
```



调用有可选参数及默认值的函数时请不要写上方括号和等于号。

[< 上一页](#)[首页](#)[首页](#)[下一页 >](#)[用户界面 - UI](#)[基础](#)