

Ros Developer Documentation

上次修改时间：2022-01-14

一、简介

Ros 语言是一个专门用于在本App中实现自定义绘制图片的轻量语言，它是一门解释型语言，Ros 代码将被解释为 Swift 语言执行。

二、基础语法：

- 1. "{" 代表一个 Ros 语句的开始，"}" 代表一个 Ros 语句的结束。 "{" 和 "}" 应成对出现。
- 2. 任意一个 Ros 语句只有如下两种形式：

- a. 是一个函数调用语句

一个函数调用语句由函数名和参数两大部分组成，参数个数视具体函数而定，例如：

Plain Text

```
1 {函数名, 参数1, 参数2, 参数3}
```

- b. 是多个 Ros 语句

这种语句常作为 "循环" 函数的最后一个参数、"判断" 函数的后两个参数等情景。例如在下面的代码中，2~6 行整体是一个 Ros 语句，作为第二个参数供 "循环" 函数使用，里面有三个小的 Ros 语句。1~7 行整体代码的作用是将 2~6 行的内容循环执行 5 次。

Plain Text

```
1 {循环, 5,
2 {
3 {函数a, 参数1, 参数2}
4 {函数b, 参数1}
5 {函数c, 参数1, 参数2}
6 }
7 }
```

- 3. 将使用中文逗号（“，”）和英文逗号（“,”）来分割一个 Ros 语句中的函数名和各个参数，因此可以混用 “，” 和 “,”。
- 4. "//" 被用来作为注释标志，"//" 后面的当前行的剩余内容将不会被解释执行。
- 5. 除了注释与换行有关外，所有的 空格、换行 等在解释执行时均会被忽略。

三、内置函数说明

{新建画板，画板宽度，画板高度，绘画内容}

函数名称	新建画板
参数1	一个数值，可以是整数或小数，表示此画板的宽度。
参数2	一个数值，可以是整数或小数，表示此画板的高度。
参数3	一个 Ros 语句，应在里面写需要进行的绘制操作。
返回值	一个图片对象，即此画板最终生成的图片。

Plain Text
<pre>1 {新建画板，512，512，{ 2 {画图片，丢，0，0} 3 {画图片，{图像圆角化，头像}，30，200，100，100} 4 }}</pre>

{画图片，一个图片，图片左边距画板距离，图片上边距画板距离，图片画出宽度，图片画出高度}

函数名称	画图片
参数1	一个图片，将要被画的图片。
参数2	一个数值，可以是整数或小数，表示图片将画的位置的左边距画板距离。
参数3	一个数值，可以是整数或小数，表示图片将画的位置的上边距画板距离。
参数4	一个数值，可以是整数或小数，给定图片画出时的宽度。允许与参数5一起省略，省略时则默认使用图片本身的宽度。
参数5	一个数值，可以是整数或小数，给定图片画出时的高度。允许与参数4一起省略，省略时则默认使用图片本身的高度。
返回值	无。

Plain Text

```
1 {画图片，头像，30，200，100，100}  
2  
3 // 省略了参数4和参数5  
4 {画图片，头像，30，200}
```

{图像圆角化，一个图片}

函数名称	图像圆角化
参数1	一个图片，将要被圆角化处理的图片。
返回值	圆角化后的图片。

注：处理后将生成一个新图片，原图片不变。

Plain Text

```
1 {图像圆角化，头像}
```

{取图片宽度}

函数名称	取图片宽度
参数1	一个图片。
返回值	一个数值，表示传入图片的宽度。

Plain Text

```
1 {取图片宽度，头像}
```

{取图片高度}

函数名称	取图片高度
参数1	一个图片。
返回值	一个数值，表示传入图片的高度。

Plain Text

```
1  {取图片高度, 头像}
```

{取颜色, 红色值, 绿色值, 蓝色值, 不透明度}

函数名称	取颜色
参数1	一个整数，必须在 0 ~ 255 之间，表示此颜色的红色通道的值。
参数2	一个整数，必须在 0 ~ 255 之间，表示此颜色的绿色通道的值。
参数3	一个整数，必须在 0 ~ 255 之间，表示此颜色的蓝色通道的值。
参数4	一个小数，必须在 0 ~ 1 之间，表示颜色的不透明度。此参数可以省略，省略时则为1.0，即完全不透明。
返回值	对应的颜色对象。

Plain Text

```
1  // 以下两句话都将返回红色
2
3  {取颜色, 255, 0, 0}
4
5  {取颜色, 255, 0, 0, 1}
```

{循环, 循环次数, 循环变量, 循环内容}

函数名称	循环
参数1	一个整数，表示循环次数。
参数2	一个变量名，这个变量可在循环内使用，从 0 开始递增，第一轮循环为 0，第二轮循环为 1，以此类推。此参数可以省略。
参数3	一个 Ros 语句，应在里面写循环要做的事情。
返回值	无。

Plain Text

```
1  {循环, 5, {
2  {函数a, 参数1, 参数2}
3  {函数b, 参数1}
4  {函数c, 参数1, 参数2}
5  }}
6
7
8  {循环, 5, i, {
9  {画图片, 图片{i}, 0, 0}
10 }}

```

{创建动图，内容}

函数名称	创建动图
参数1	一个 Ros 语句，应在里面写添加动图每一帧的相关操作。
返回值	一个动图对象。

Plain Text

```
1  {创建动图, {
2  {添加帧, 图片a}
3  {添加帧, 图片b}
4  {添加帧, 图片c}
5  }}

```

{添加帧，图片}

函数名称	添加帧
参数1	一个图片对象，将会作为动图的一帧。
返回值	无。

注：此函数一般只会在「创建动图」函数的「内容」参数中使用。

Plain Text

```
1  {添加帧, 图片a}

```

{设置帧延迟，延迟时间}

函数名称	设置帧延迟
参数1	一个数字，表示动图每帧的延迟时间，单位为秒。如果不主动使用此函数来设置延迟时间，将使用默认值：0.1，即每帧延迟0.1秒。
返回值	无。

Plain Text
1 {设置帧延迟，0.1}