**[Colors](https://github.com/mbostock/d3/wiki/Colors)**

* [d3.rgb](https://github.com/mbostock/d3/wiki/Colors#wiki-d3_rgb) - 指定一种颜色，创建一个RGB颜色对象。支持多种颜色格式的输入。
* [rgb.brighter](https://github.com/mbostock/d3/wiki/Colors#wiki-rgb_brighter) - 增强颜色的亮度，变化幅度由参数决定。
* [rgb.darker](https://github.com/mbostock/d3/wiki/Colors#wiki-rgb_darker) - 减弱颜色的亮度，变化幅度由参数决定。
* [rgb.hsl](https://github.com/mbostock/d3/wiki/Colors#wiki-rgb_hsl) - 将RGB颜色对象转化成HSL颜色对象。
* [rgb.toString](https://github.com/mbostock/d3/wiki/Colors#wiki-rgb_toString) - RGB颜色转化为字符串格式。
* [d3.hsl](https://github.com/mbostock/d3/wiki/Colors#wiki-d3_hsl) - 创建一个HSL颜色对象。支持多种颜色格式的输入。
* [hsl.brighter](https://github.com/mbostock/d3/wiki/Colors#wiki-hsl_brighter) - 增强颜色的亮度，变化幅度由参数决定。
* [hsl.darker](https://github.com/mbostock/d3/wiki/Colors#wiki-hsl_darker) - 减弱颜色的亮度，变化幅度由参数决定。
* [hsl.rgb](https://github.com/mbostock/d3/wiki/Colors#wiki-hsl_rgb) - 将HSL颜色对象转化成RGB颜色对象。
* [hsl.toString](https://github.com/mbostock/d3/wiki/Colors#wiki-hsl_toString) - HSL颜色转化为字符串格式。
* [d3.lab](https://github.com/mbostock/d3/wiki/Colors#wiki-d3_lab) - 创建一个Lab颜色对象。支持多种颜色格式的输入。
* [lab.brighter](https://github.com/mbostock/d3/wiki/Colors#wiki-lab_brighter) - 增强颜色的亮度，变化幅度由参数决定。
* [lab.darker](https://github.com/mbostock/d3/wiki/Colors#wiki-lab_darker) - 减弱颜色的亮度，变化幅度由参数决定。
* [lab.rgb](https://github.com/mbostock/d3/wiki/Colors#wiki-lab_rgb) - 将Lab颜色对象转化成RGB颜色对象。
* [lab.toString](https://github.com/mbostock/d3/wiki/Colors#wiki-lab_toString) - Lab颜色转化为字符串格式。
* [d3.hcl](https://github.com/mbostock/d3/wiki/Colors#wiki-d3_hcl) - 创建一个HCL颜色对象。支持多种颜色格式的输入。
* [hcl.brighter](https://github.com/mbostock/d3/wiki/Colors#wiki-hcl_brighter) - 增强颜色的亮度，变化幅度由参数决定。
* [hcl.darker](https://github.com/mbostock/d3/wiki/Colors#wiki-hcl_darker) - 减弱颜色的亮度，变化幅度由参数决定。
* [hcl.rgb](https://github.com/mbostock/d3/wiki/Colors#wiki-hcl_rgb) - 将HCL颜色对象转化成RGB颜色对象。
* [hcl.toString](https://github.com/mbostock/d3/wiki/Colors#wiki-hcl_toString) - HCL颜色转化为字符串格式。

实现可视化要经常和颜色打交道。虽然你的电脑显示屏懂得很多的颜色，但这对通过js来配置颜色帮助不大。所以d3为RGB和HSL两种配色方案都提供了两种呈现方法，实现颜色空间的插值，和颜色的明暗。为了实现更多的颜色配置方案，请查阅维基百科 [RGB](http://en.wikipedia.org/wiki/RGB_color_model) 和 [HSL](http://en.wikipedia.org/wiki/HSL_and_HSV).

注意：虽然你可以直接操作颜色，当时也需要参考一下D3对于[interpolateRgb](https://github.com/mbostock/d3/wiki/Transitions" \l "wiki-d3_interpolateRgb), [interpolateHsl](https://github.com/mbostock/d3/wiki/Transitions#wiki-d3_interpolateHsl) 和 [scales](https://github.com/mbostock/d3/wiki/Scales) 等内置函数的实现。

如果你想查阅调色盘，请查阅 [ordinal scales](https://github.com/mbostock/d3/wiki/Ordinal-Scales) 。

## RGB

[#](https://github.com/mbostock/d3/wiki/Colors" \l "wiki-d3_rgb) d3.**rgb**(*r*, *g*, *b*)

通过输入的参数r, g 和 b，创建一个RGB类型颜色对象。参数必须是在0-255之间的整数。你可以通过访问返回的颜色对象的r, g and b 属性值来获取参数r, g, b的值。

[#](https://github.com/mbostock/d3/wiki/Colors#wiki-d3_rgb) d3.**rgb**(*color*)

通过解析输入的字符串参数，创建一个RGB类型颜色对象。如果参数 color 不是一个字符串，参数就会被强制类型转换为字符串类型。因此，该构造函数可以用来创建一个已经存在的颜色对象的副本，或者是将[d3.hsl](https://github.com/mbostock/d3/wiki/Colors#wiki-d3_hsl)强制类型转换为RGB。字符串参数可以有多种形式：

• rgb decimal - "rgb(255,255,255)"

• hsl decimal - "hsl(120,50%,20%)"

• rgb hexadecimal - "#ffeeaa"

• rgb shorthand hexadecimal - "#fea"

• named - "red", "white", "blue"

输出的颜色将以红、绿和蓝的整数通道形式存储（整数范围：[0,255]）。颜色通道可以通过颜色对象的属性 r, g 和 b 访问到。可支持的列表[named colors](http://www.w3.org/TR/SVG/types.html#ColorKeywords) 可以通过CSS指定。如果是HSL 空间的颜色，可以转换成在RGB空间相同类型的值，和 [hsl.rgb](https://github.com/mbostock/d3/wiki/Colors" \l "wiki-hsl_rgb) 类似。

[#](https://github.com/mbostock/d3/wiki/Colors" \l "wiki-rgb_brighter) rgb.**brighter**([*k*])

返回一个亮度高的颜色拷贝。每个颜色通道值将乘以0.7 ^ -k. 如果参数 k 被忽略，将使用默认值1。通道值上限值255，下限值30.

[#](https://github.com/mbostock/d3/wiki/Colors" \l "wiki-rgb_darker) rgb.**darker**([*k*])

返回一个亮度低的颜色拷贝。每个颜色通道值将乘以0.7 ^ k. 如果参数 k 被忽略，将使用默认值1。

[#](https://github.com/mbostock/d3/wiki/Colors" \l "wiki-rgb_hsl) rgb.**hsl**()

返回一个HSL空间的等值颜色对象。请查阅 [d3.hsl](https://github.com/mbostock/d3/wiki/Colors#wiki-d3_hsl) 了解更多关于返回颜色对象的信息. 文档 [CSS3 Color Module Level 3](http://www.w3.org/TR/css3-color/#hsl-color) 中有关于RGB到HSL转换的信息。该函数是上述转化的逆操作。

[#](https://github.com/mbostock/d3/wiki/Colors" \l "wiki-rgb_toString) rgb.**toString**()

将RGB颜色转换成一个十六进制数的字符串，如 such as "#f7eaba"。

## HSL

[#](https://github.com/mbostock/d3/wiki/Colors" \l "wiki-d3_hsl) d3.**hsl**(*h*, *s*, *l*)

通过输入的参数h, s 和 l，创建一个HSL颜色对象。创建Constructs a new HSL color with the specified hue h, saturation s and lightness l. 其中色度h取值范围[0,360]。饱和度和亮度取值范围 [0,1](https://github.com/mbostock/d3/wiki/not%20percentages)。 你可以通过访问返回的颜色对象的h, s and l 属性值来获取颜色的通道属性.

[#](https://github.com/mbostock/d3/wiki/Colors#wiki-d3_hsl) d3.**hsl**(*color*)

通过解析输入的字符串参数，创建一个HSL类型颜色对象。如果参数 color 不是一个字符串，参数就会被强制类型转换为字符串类型。因此，该构造函数可以用来创建一个已经存在的颜色对象的副本，或者是将[d3.rgb](https://github.com/mbostock/d3/wiki/Colors#wiki-d3_hsl)强制类型转换为HSL。字符串参数可以有多种形式：

• rgb decimal - "rgb(255,255,255)"

• hsl decimal - "hsl(120,50%,20%)"

• rgb hexadecimal - "#ffeeaa"

• rgb shorthand hexadecimal - "#fea"

• named - "red", "white", "blue"

输出的颜色将以取值范围为[0,360]的色度和取值范围为[0,1]的亮度、饱和度作为属性值存储。红、绿和蓝的整数通道形式存储（整数范围：[0,255]）。颜色通道可以通过颜色对象的属性 h, s 和 l 访问到。可支持的列表[named colors](http://www.w3.org/TR/SVG/types.html" \l "ColorKeywords) 可以通过CSS指定。如果是RGB空间的颜色，可以转换成在HSL空间相同类型的值，和 [rgb.hsl](https://github.com/mbostock/d3/wiki/Colors#wiki-rgb_hsl) 类似。

[#](https://github.com/mbostock/d3/wiki/Colors" \l "wiki-hsl_brighter) hsl.**brighter**([*k*])

返回一个亮度高的颜色拷贝。每个颜色通道值将乘以0.7 ^ -k. 如果参数 k 被忽略，将使用默认值1。

[#](https://github.com/mbostock/d3/wiki/Colors" \l "wiki-hsl_darker) hsl.**darker**([*k*])

返回一个亮度低的颜色拷贝。每个颜色通道值将乘以0.7 ^ k. 如果参数 k 被忽略，将使用默认值1。

[#](https://github.com/mbostock/d3/wiki/Colors" \l "wiki-hsl_rgb) hsl.**rgb**()

返回一个RGB空间的等值颜色对象。请查阅 [d3.rgb](https://github.com/mbostock/d3/wiki/Colors#wiki-d3_rgb) 了解更多关于返回颜色对象的信息. 文档 [CSS3 Color Module Level 3](http://www.w3.org/TR/css3-color/#hsl-color) 中有关于HSL到RGB 转换的信息。该函数是上述转化的逆操作。

[#](https://github.com/mbostock/d3/wiki/Colors" \l "wiki-hsl_toString) hsl.**toString**()

将颜色转换成一个十六进制数的RGB颜色字符串，如 such as "#f7eaba"。

## HCL

[#](https://github.com/mbostock/d3/wiki/Colors" \l "wiki-d3_hcl) d3.**hcl**(*h*, *c*, *l*)

…

[#](https://github.com/mbostock/d3/wiki/Colors#wiki-d3_hcl) d3.**hcl**(*color*)

…

[#](https://github.com/mbostock/d3/wiki/Colors" \l "wiki-hcl_brighter) hcl.**brighter**([*k*])

…

[#](https://github.com/mbostock/d3/wiki/Colors" \l "wiki-hcl_darker) hcl.**darker**([*k*])

…

[#](https://github.com/mbostock/d3/wiki/Colors" \l "wiki-hcl_rgb) hcl.**rgb**()

…

[#](https://github.com/mbostock/d3/wiki/Colors" \l "wiki-hcl_toString) hcl.**toString**()

## 将颜色转换成一个十六进制数的RGB颜色字符串，如 such as "#f7eaba"。L\*a\*b\*

[#](https://github.com/mbostock/d3/wiki/Colors" \l "wiki-d3_lab) d3.**lab**(*l*, *a*, *b*)

…

[#](https://github.com/mbostock/d3/wiki/Colors#wiki-d3_lab) d3.**lab**(*color*)

…

[#](https://github.com/mbostock/d3/wiki/Colors" \l "wiki-lab_brighter) lab.**brighter**([*k*])

…

[#](https://github.com/mbostock/d3/wiki/Colors" \l "wiki-lab_darker) lab.**darker**([*k*])

…

[#](https://github.com/mbostock/d3/wiki/Colors" \l "wiki-lab_rgb) lab.**rgb**()

…

[#](https://github.com/mbostock/d3/wiki/Colors" \l "wiki-lab_toString) lab.**toString**()

将这个 L\*a\*b\* 颜色转换成一个十六进制数的RGB颜色字符串，如 such as "#f7eaba"。