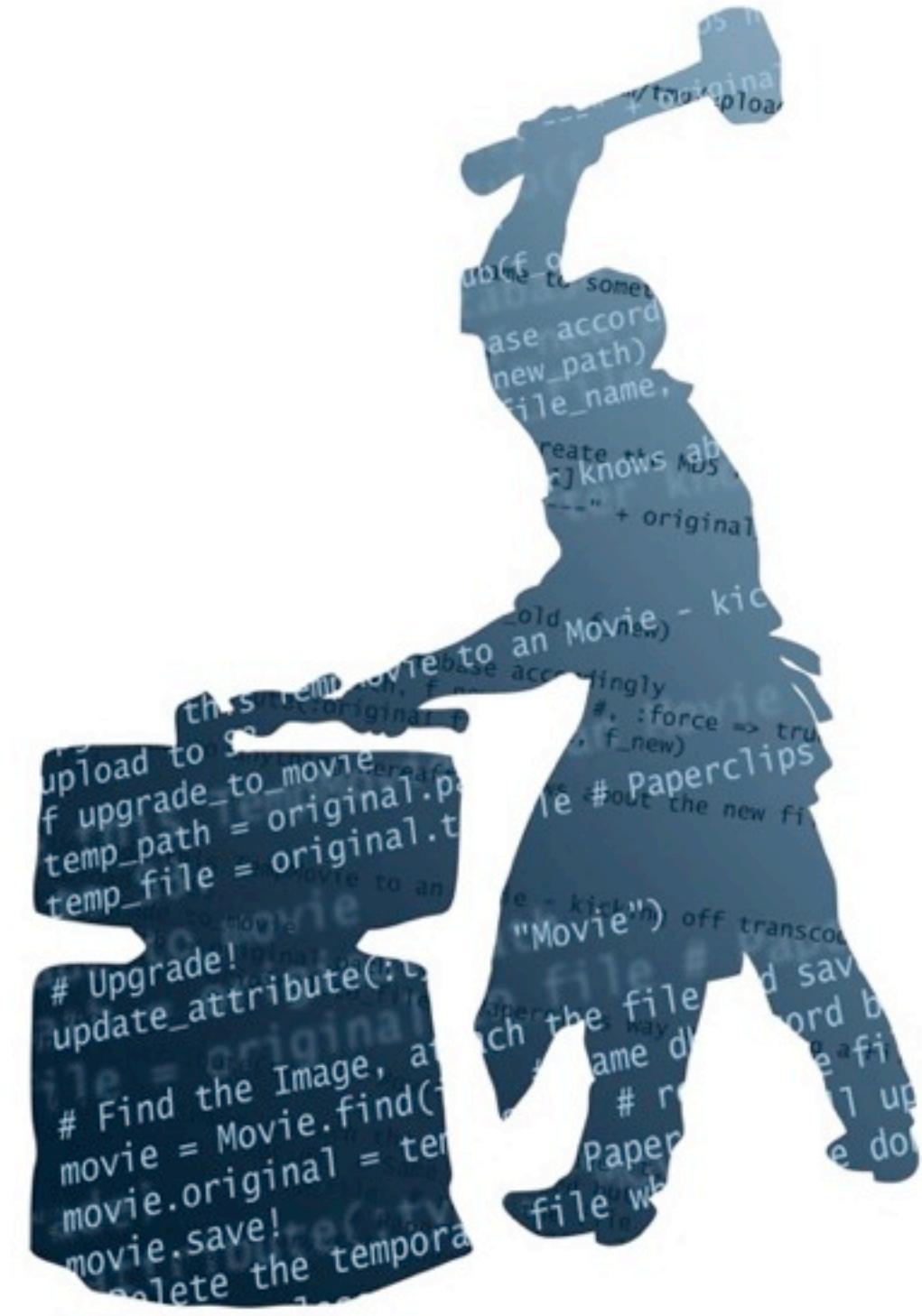




Color Schemes

A better understanding of colors



digital craftsmanship

Motivation



Approximate and explain colors



Approximate and explain colors

#FF8800

hsl(32, 100%, 50%)

rgb(255, 136, 0)

#206676

hsl(191, 57%, 29%)

rgb(32, 102, 118)



Approximate and explain colors

#FF8800

hsl(32, 100%, 50%)

rgb(255, 136, 0)

#206676

hsl(191, 57%, 29%)

rgb(32, 102, 118)

- Will I be able to guess a color value?



Approximate and explain colors

#FF8800

hsl(32, 100%, 50%)

rgb(255, 136, 0)

#206676

hsl(191, 57%, 29%)

rgb(32, 102, 118)

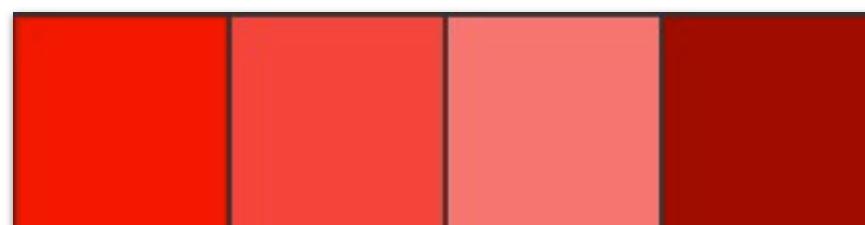
- Will I be able to guess a color value?
- Convert between HEX, RGB and HSL



Write maintainable CSS



Write maintainable CSS



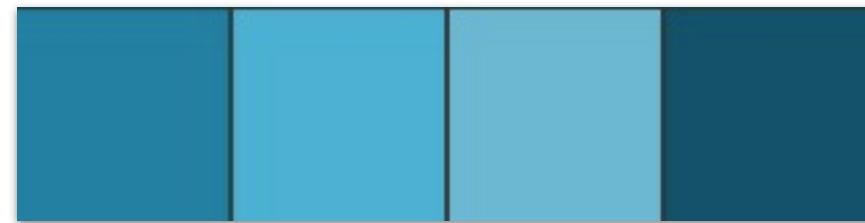
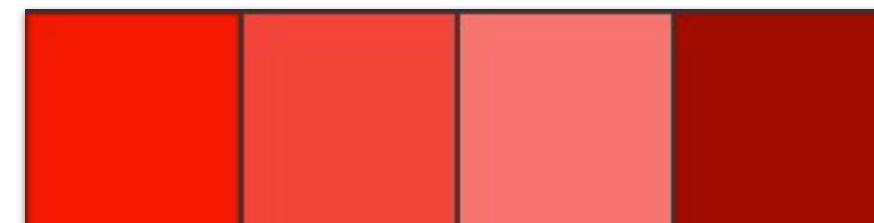
Write maintainable CSS

- Can I write CSS so it's easy to switch from:



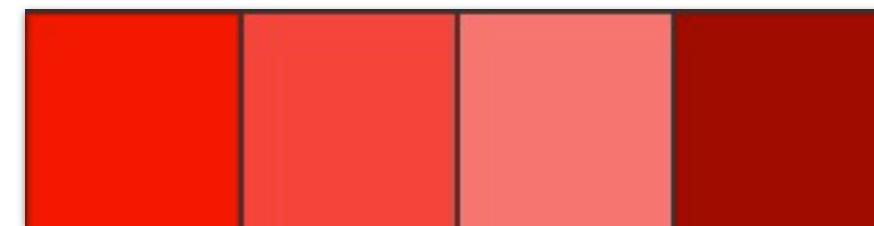
Write maintainable CSS

- Can I write CSS so it's easy to switch from:
- to:

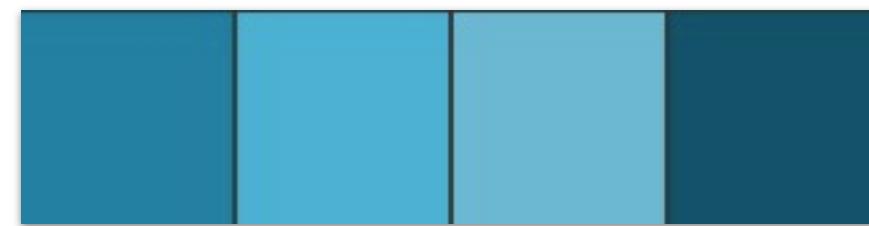


Write maintainable CSS

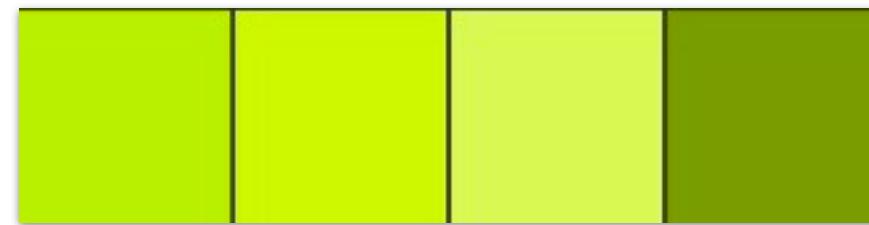
- Can I write CSS so it's easy to switch from:



- to:

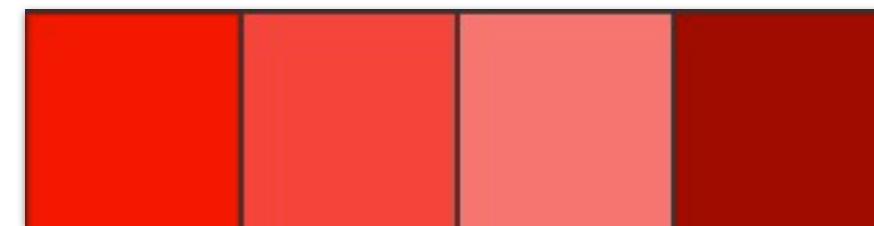


- or to:

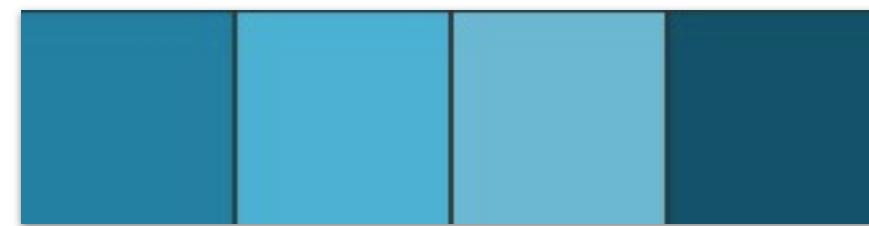


Write maintainable CSS

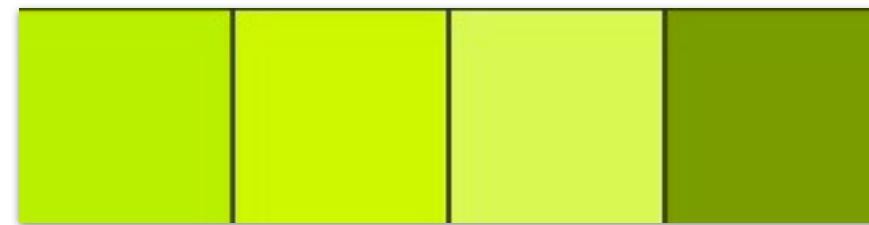
- Can I write CSS so it's easy to switch from:



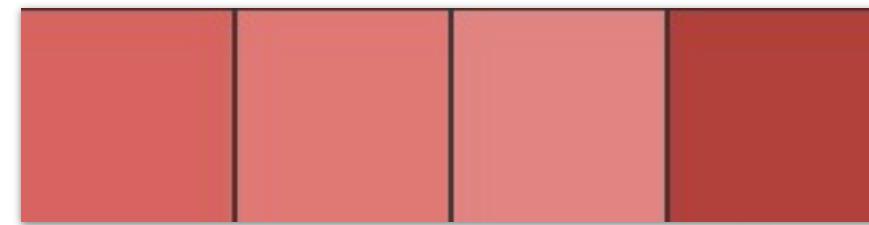
- to:



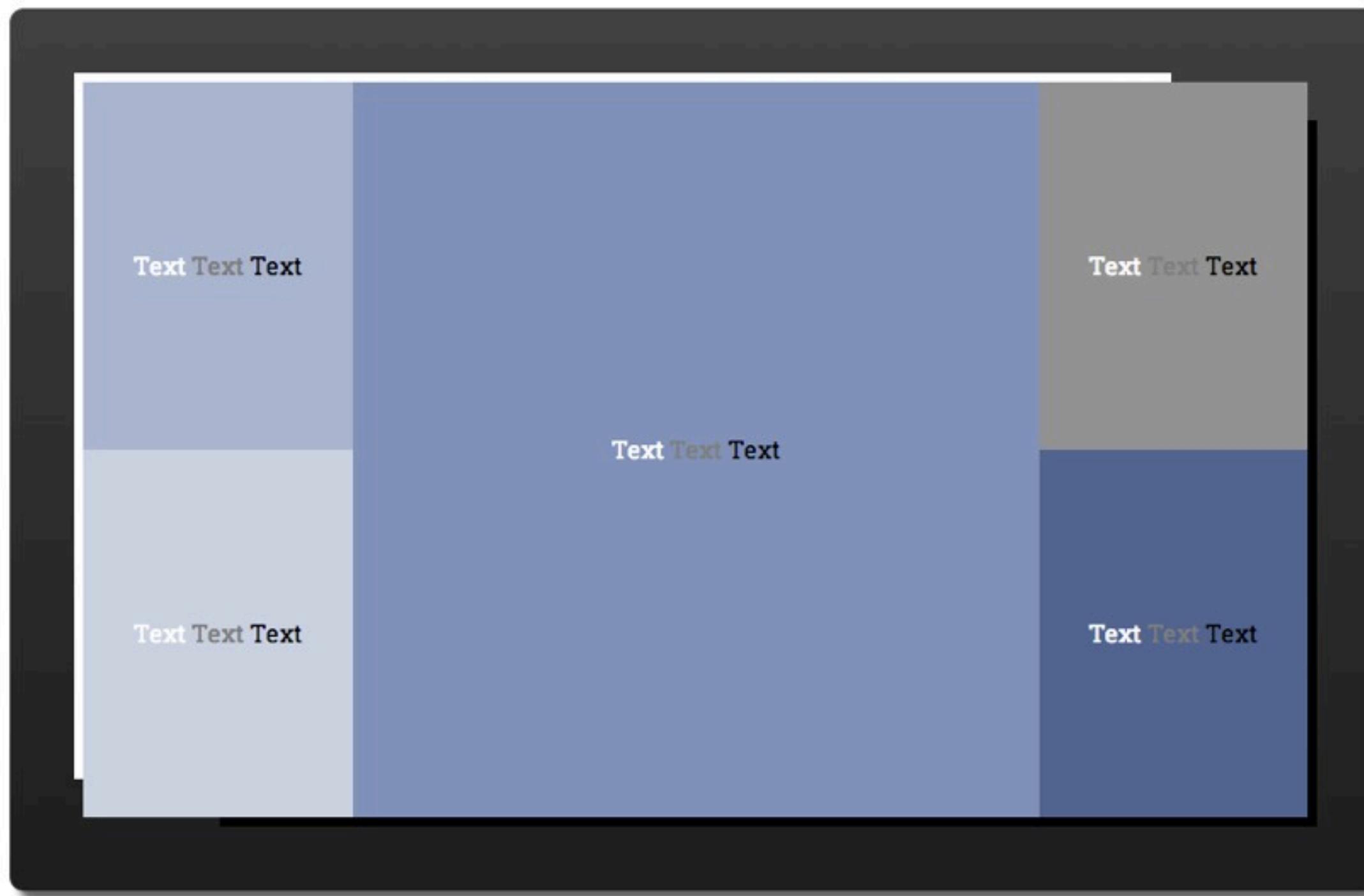
- or to:



- or even to:



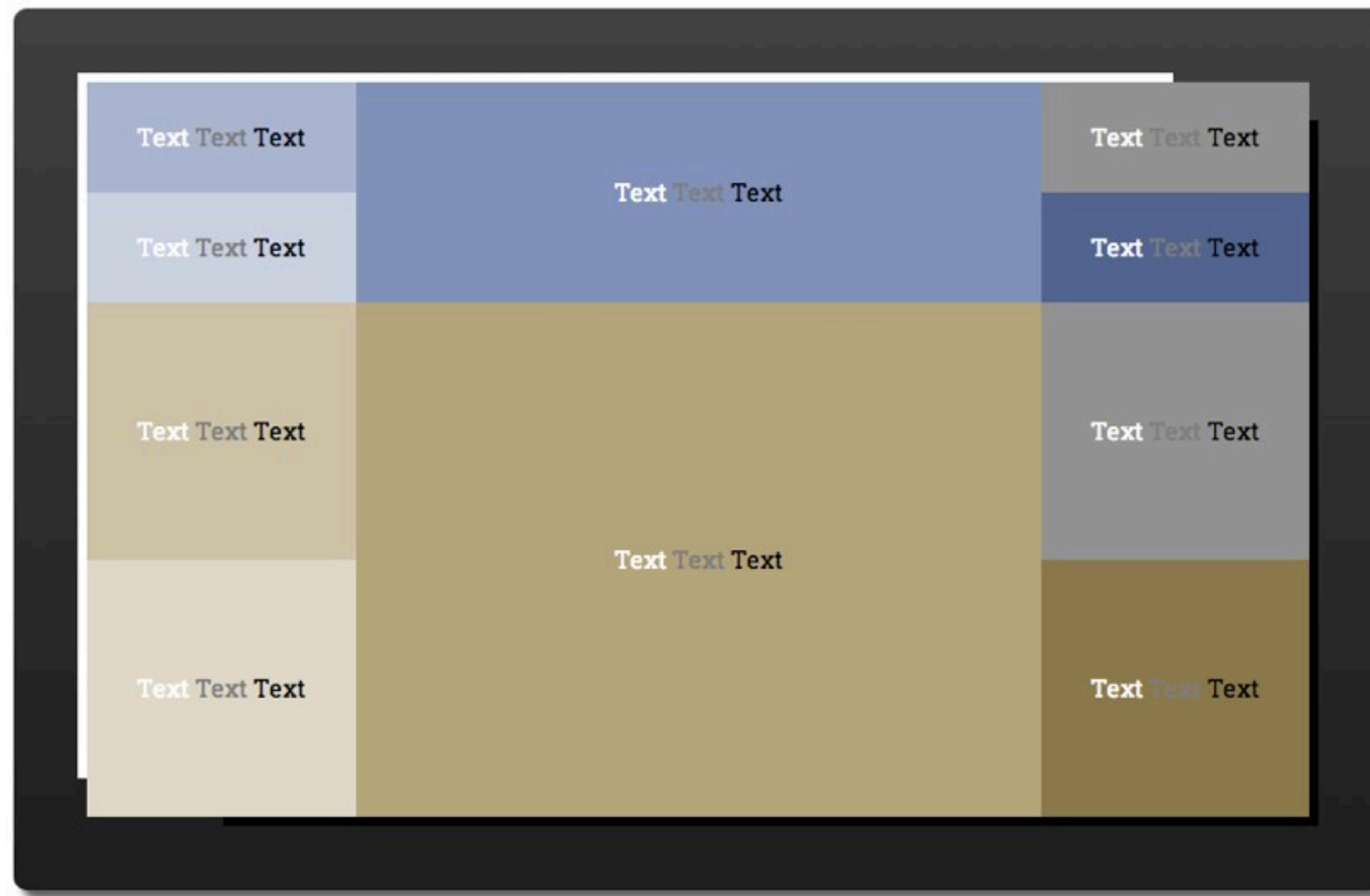
Understanding color schemes



Monochromatic model



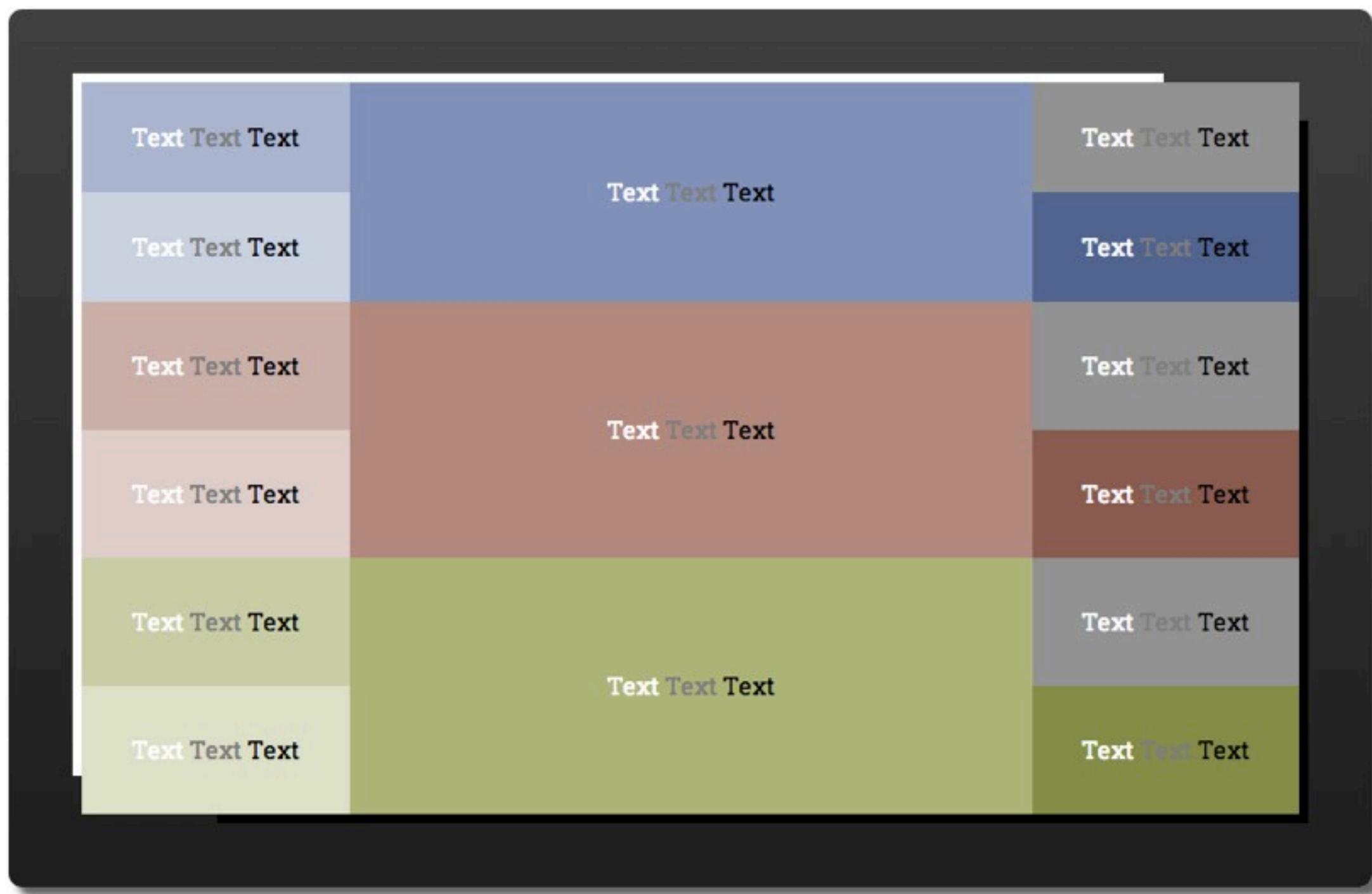
Understanding color schemes



Complementary model



Understanding color schemes



Triad (soft contrast) model



Understanding color schemes



Tetrad (double-contrast) model



Understanding color schemes



Analogic model



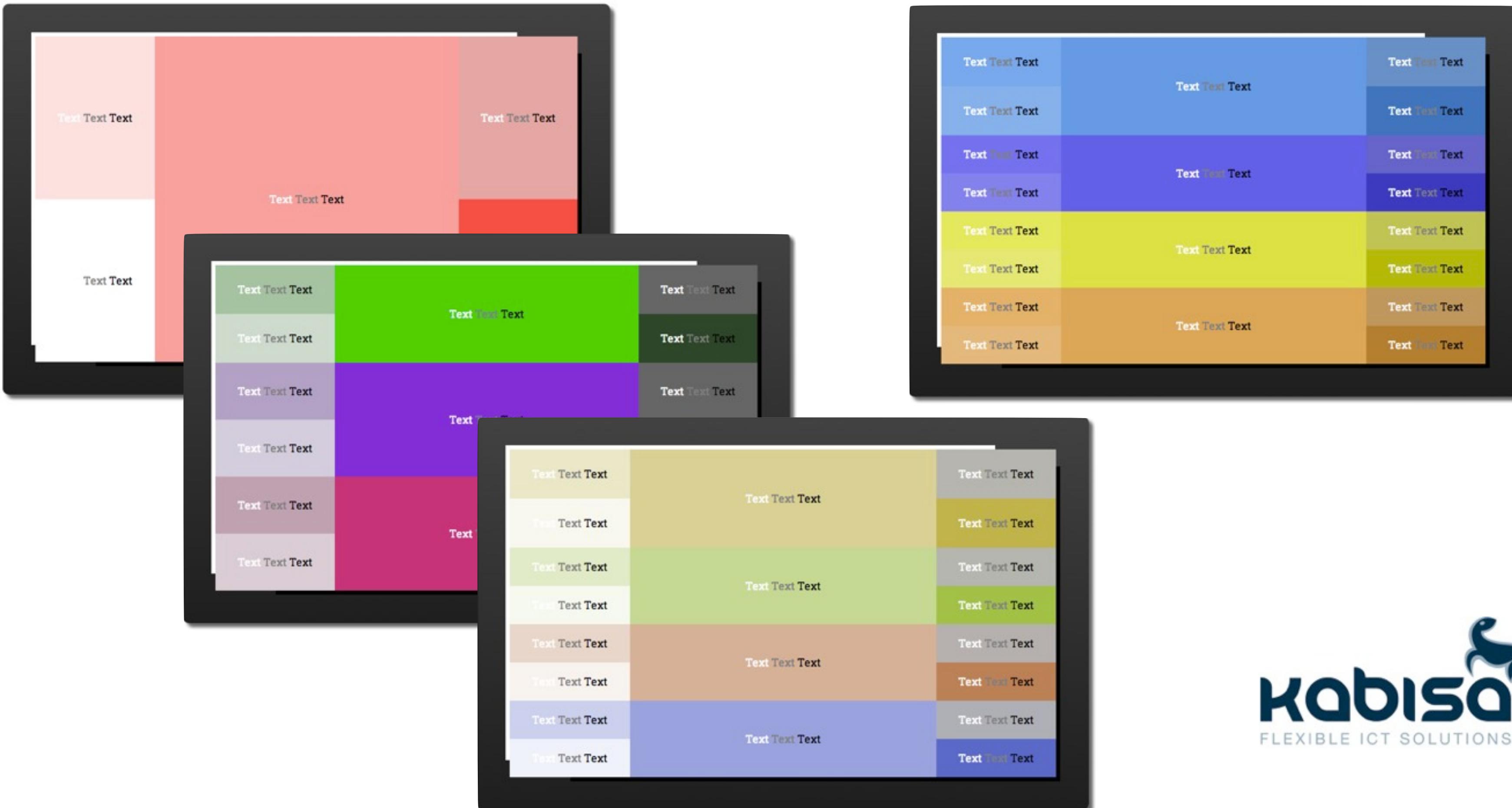
Understanding color schemes



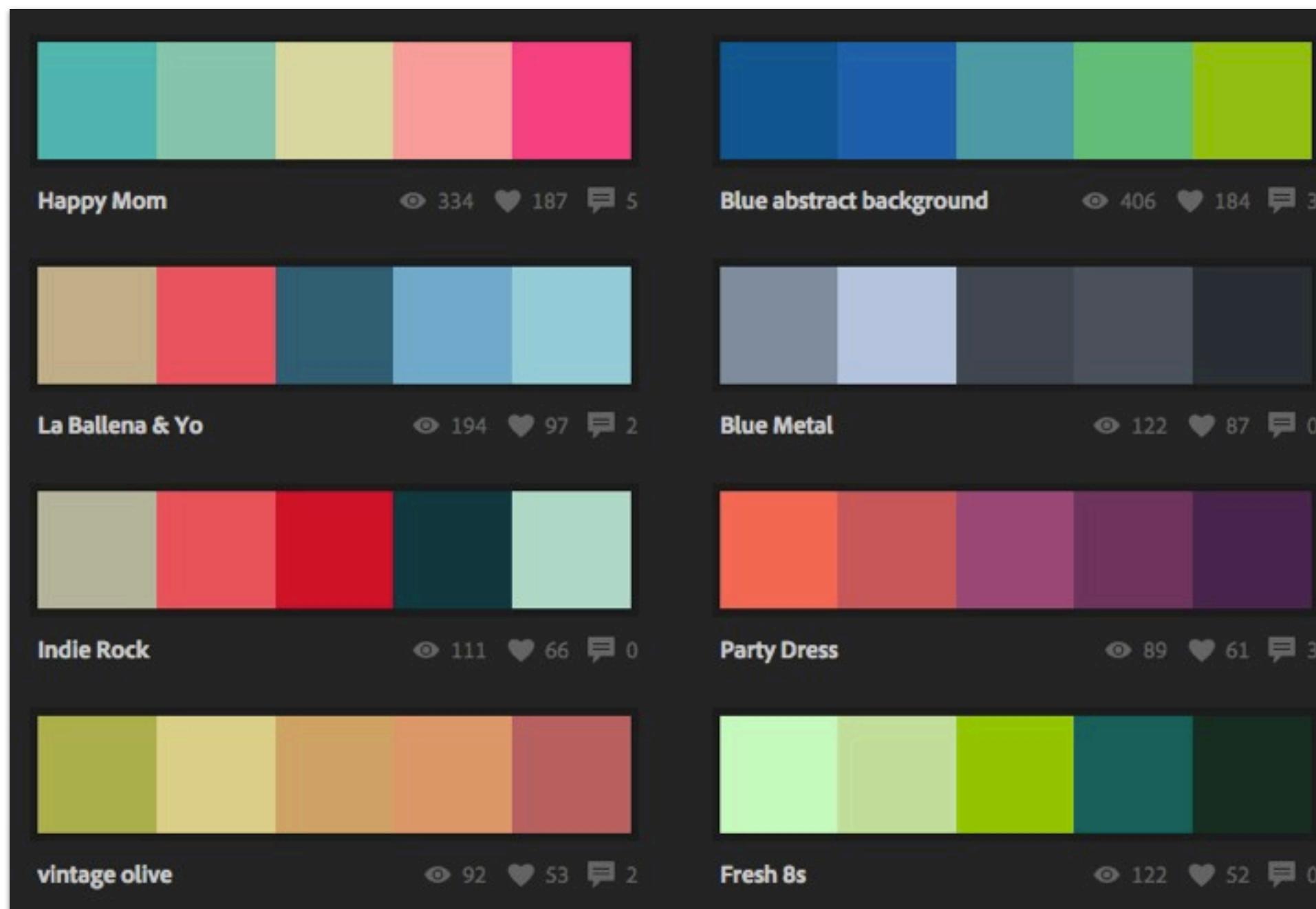
Accented analogic model



And again... maintainable



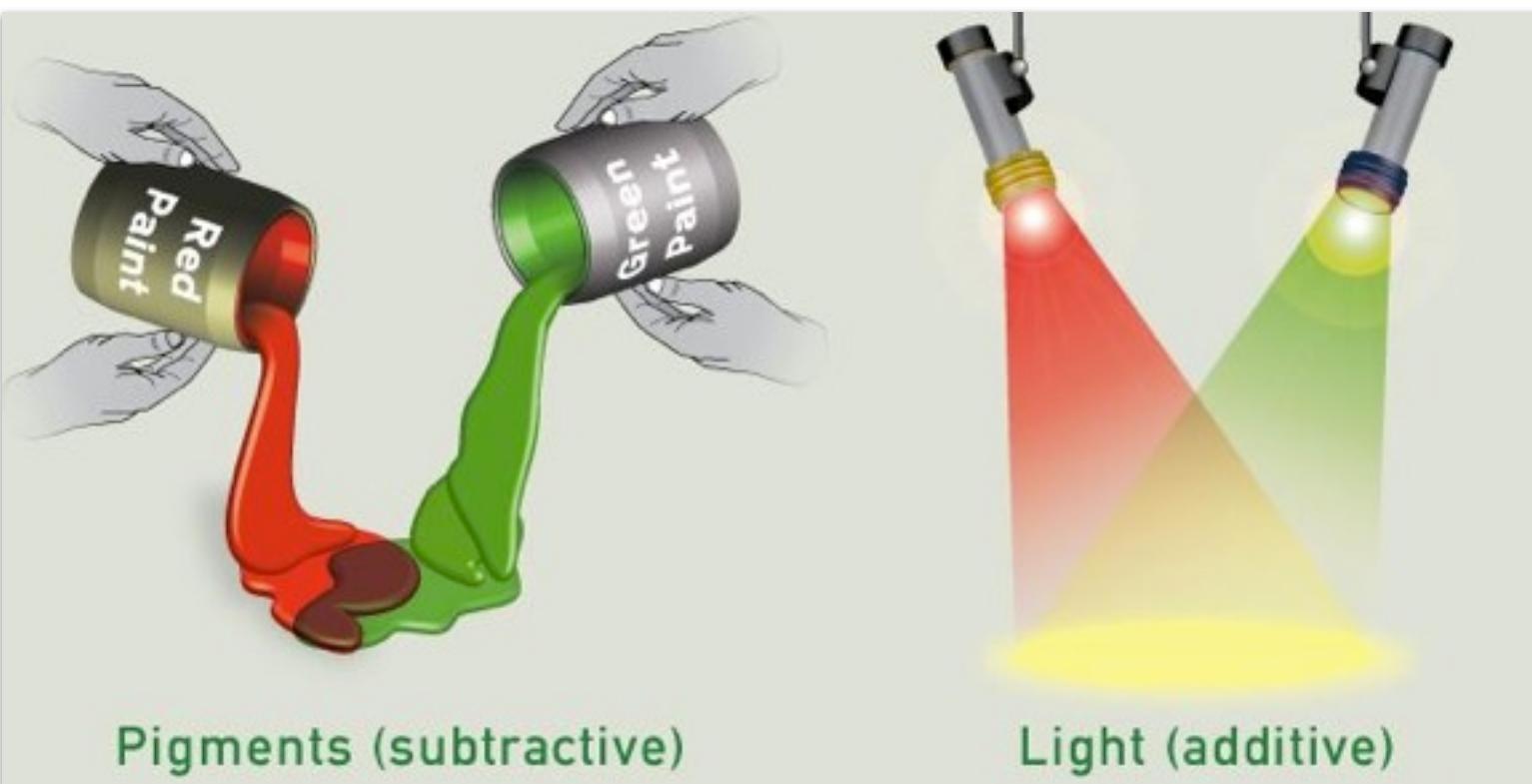
But it's not always derivable



Models, values and wheels



Color Models



RGB - RYB - CMYK

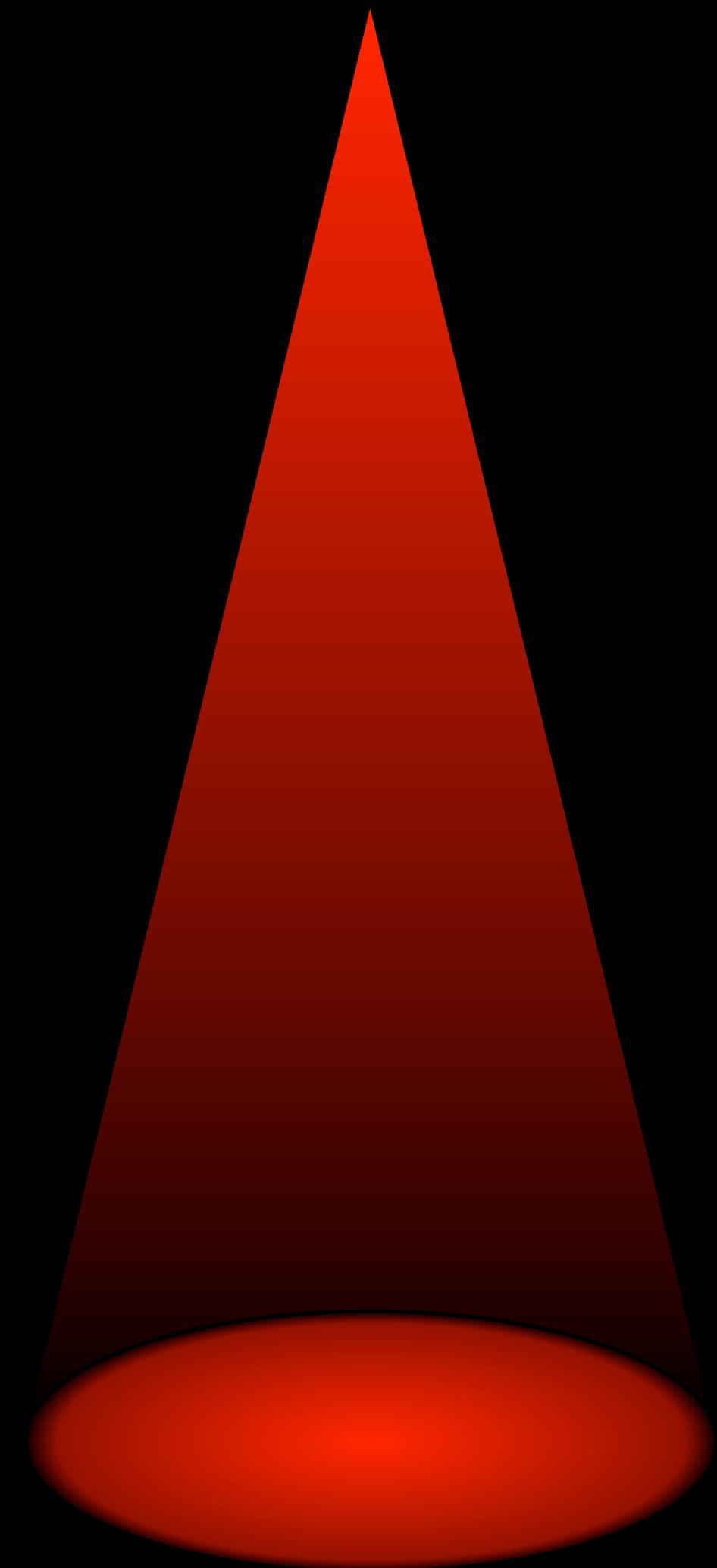


RGB Colors



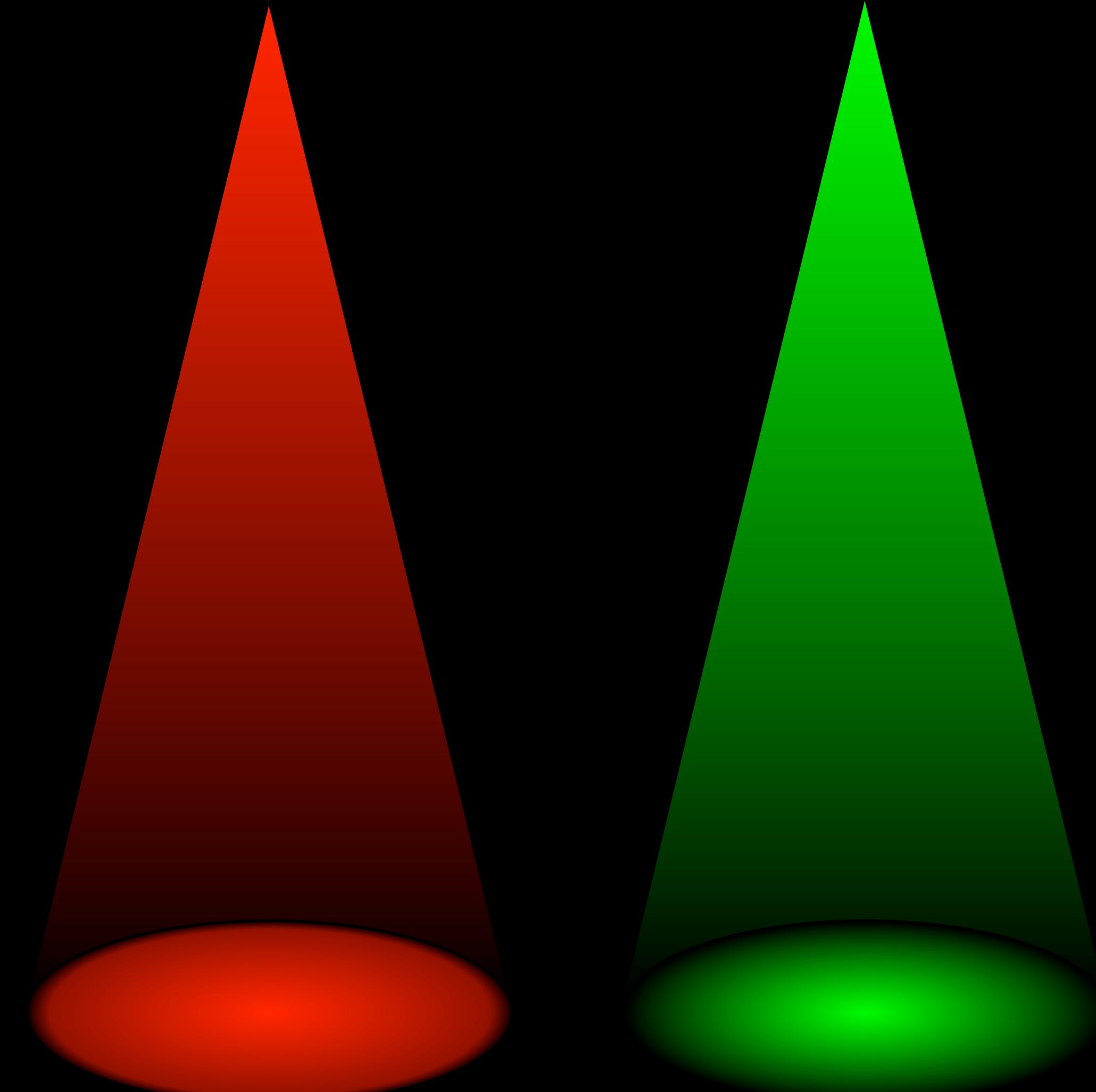
RGB Colors

RGB Colors



`rgb(255, 0, 0)`

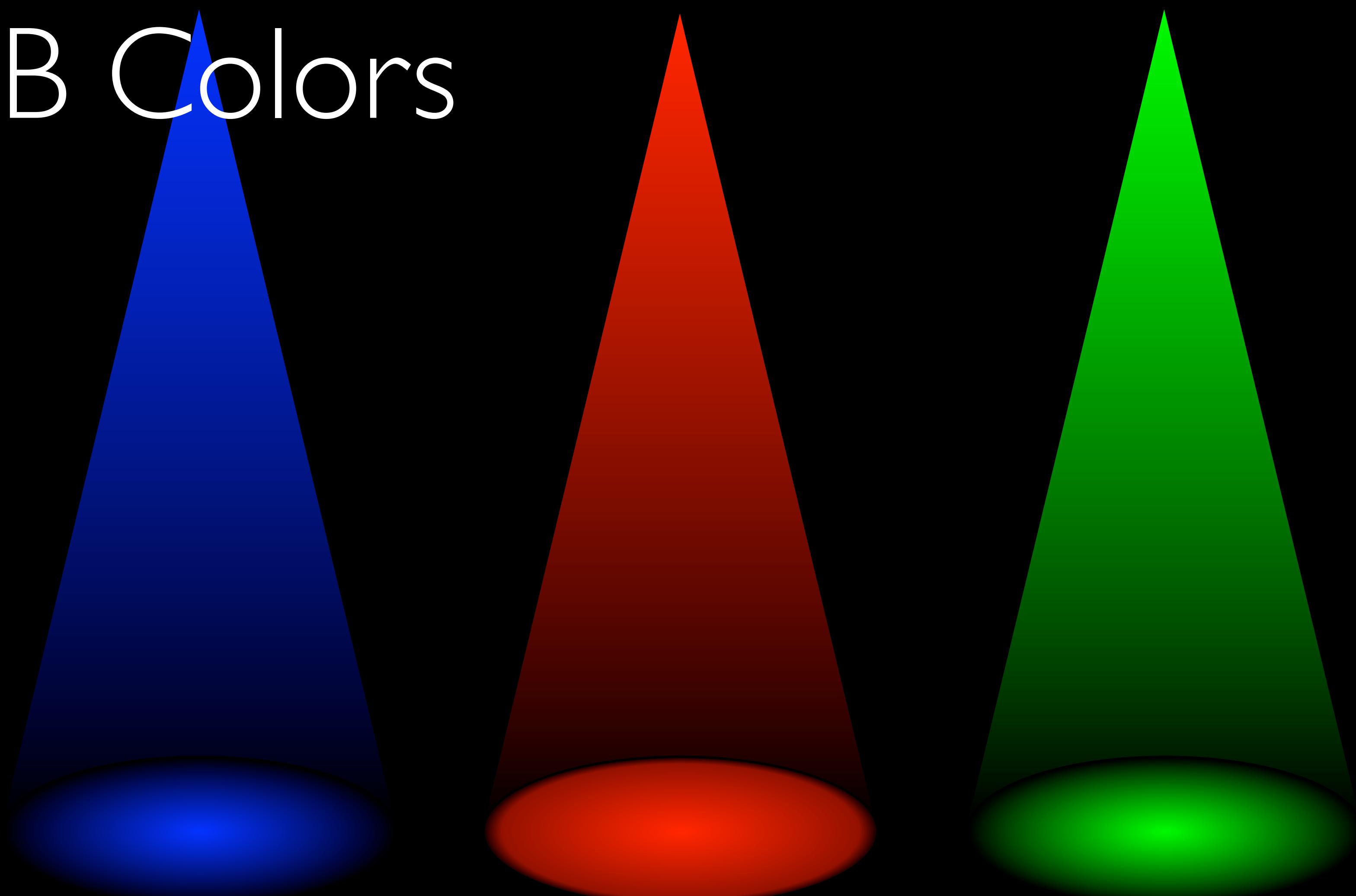
RGB Colors



`rgb(255, 0, 0)`

`rgb(0, 255, 0)`

RGB Colors

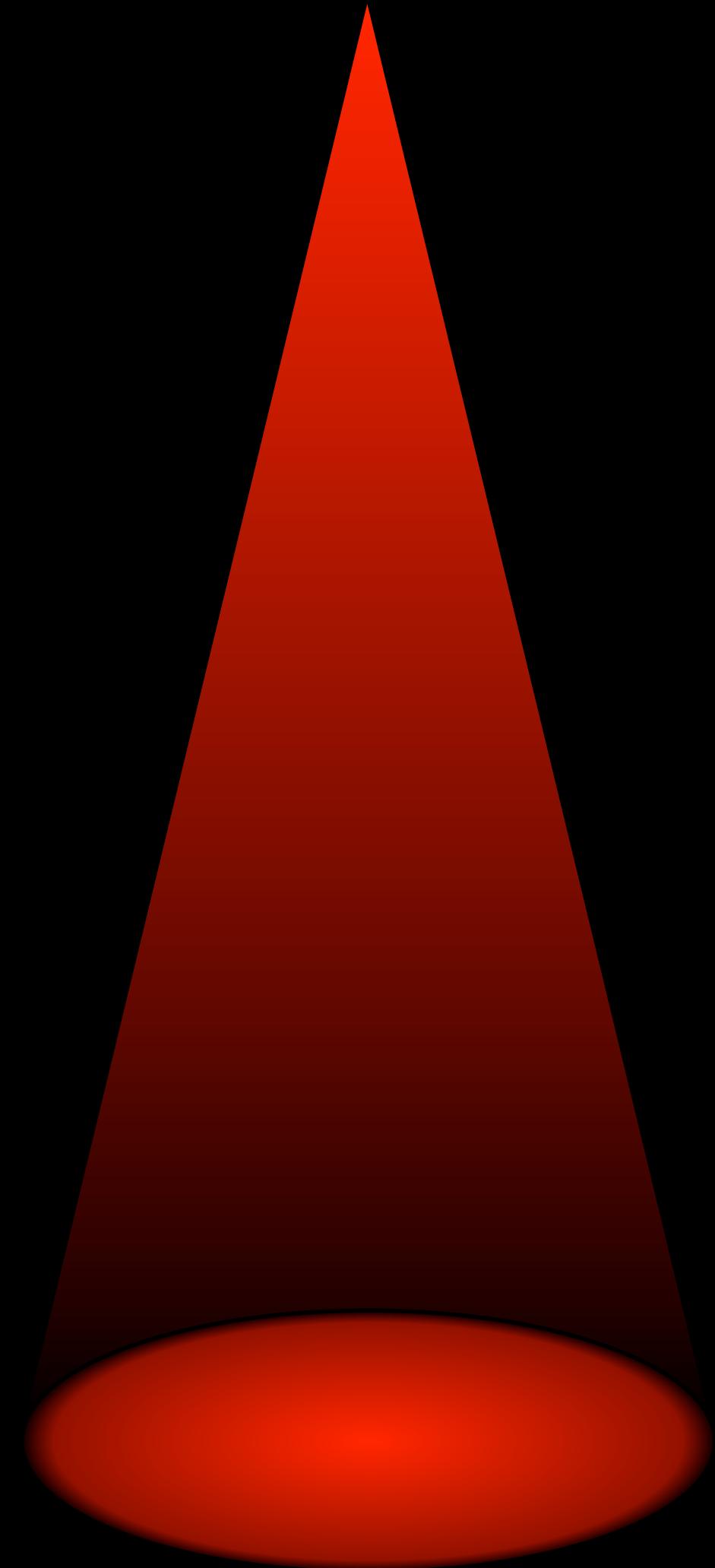


`rgb(0, 0, 255)`

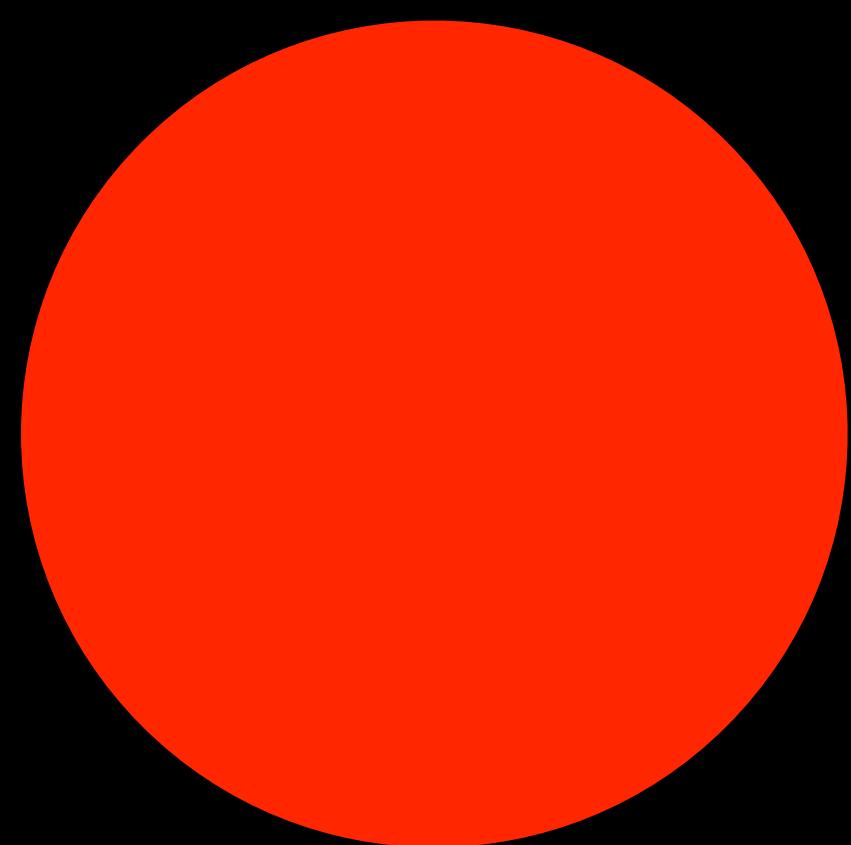
`rgb(255, 0, 0)`

`rgb(0, 255, 0)`

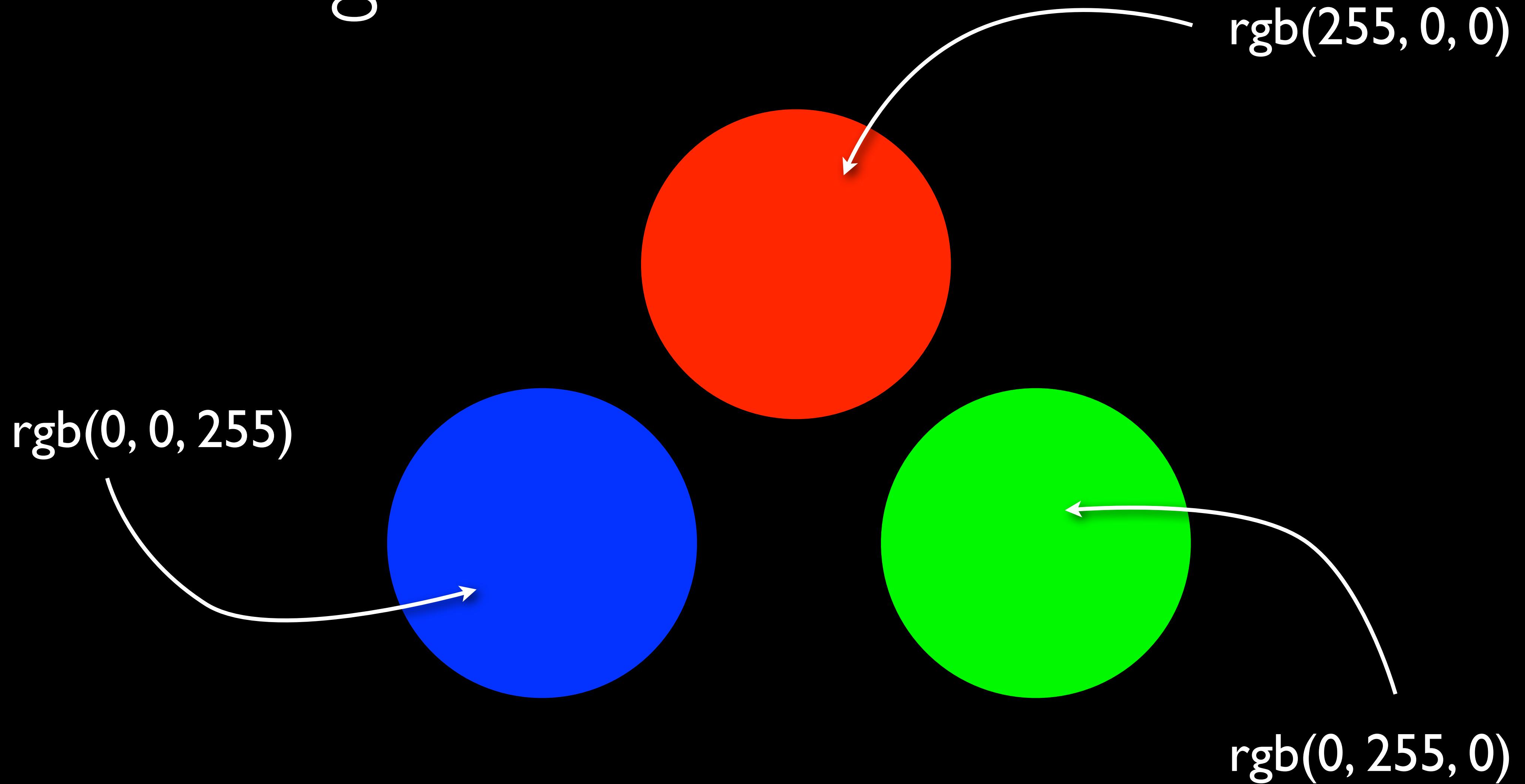
RGB Colors



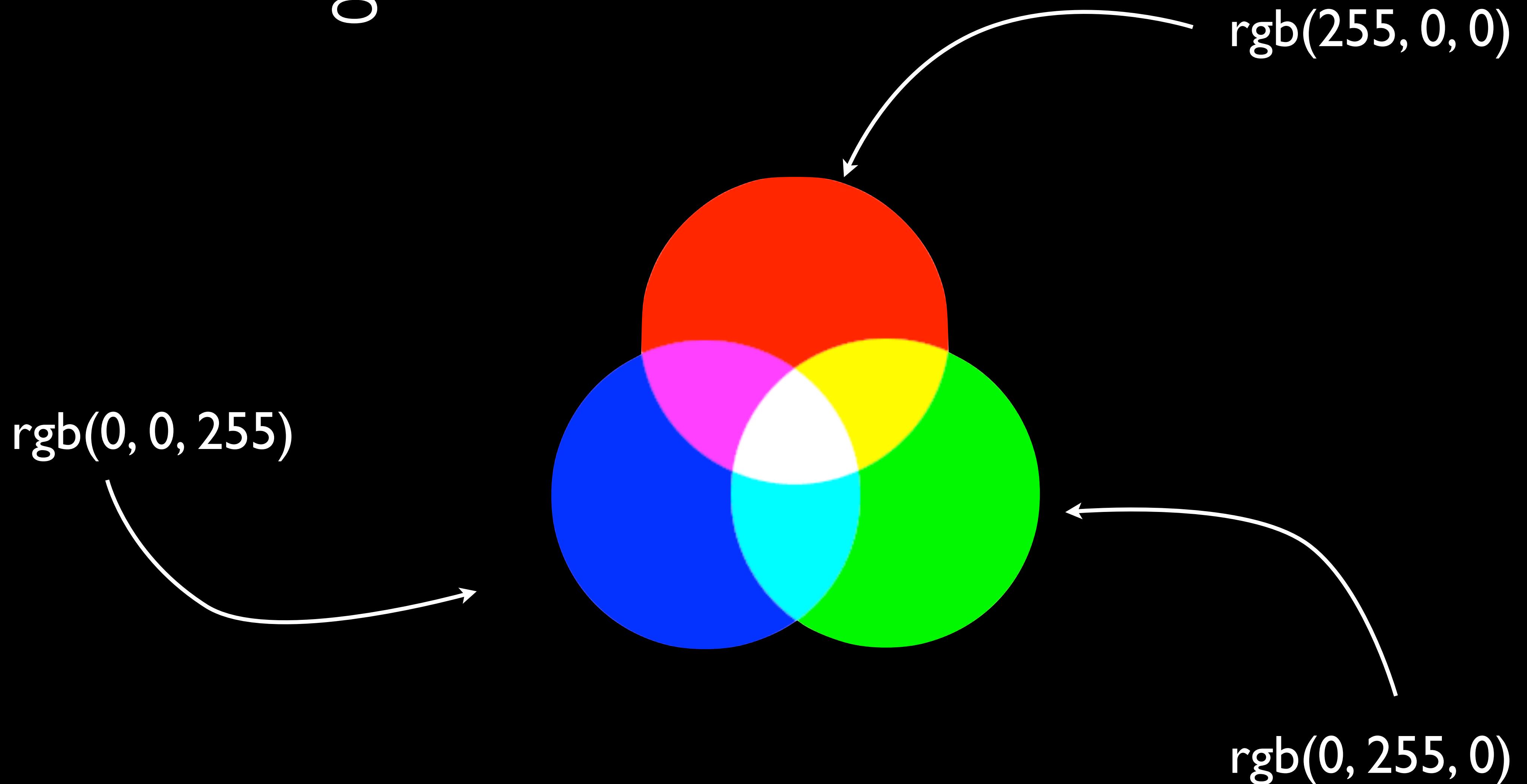
Adding Colors



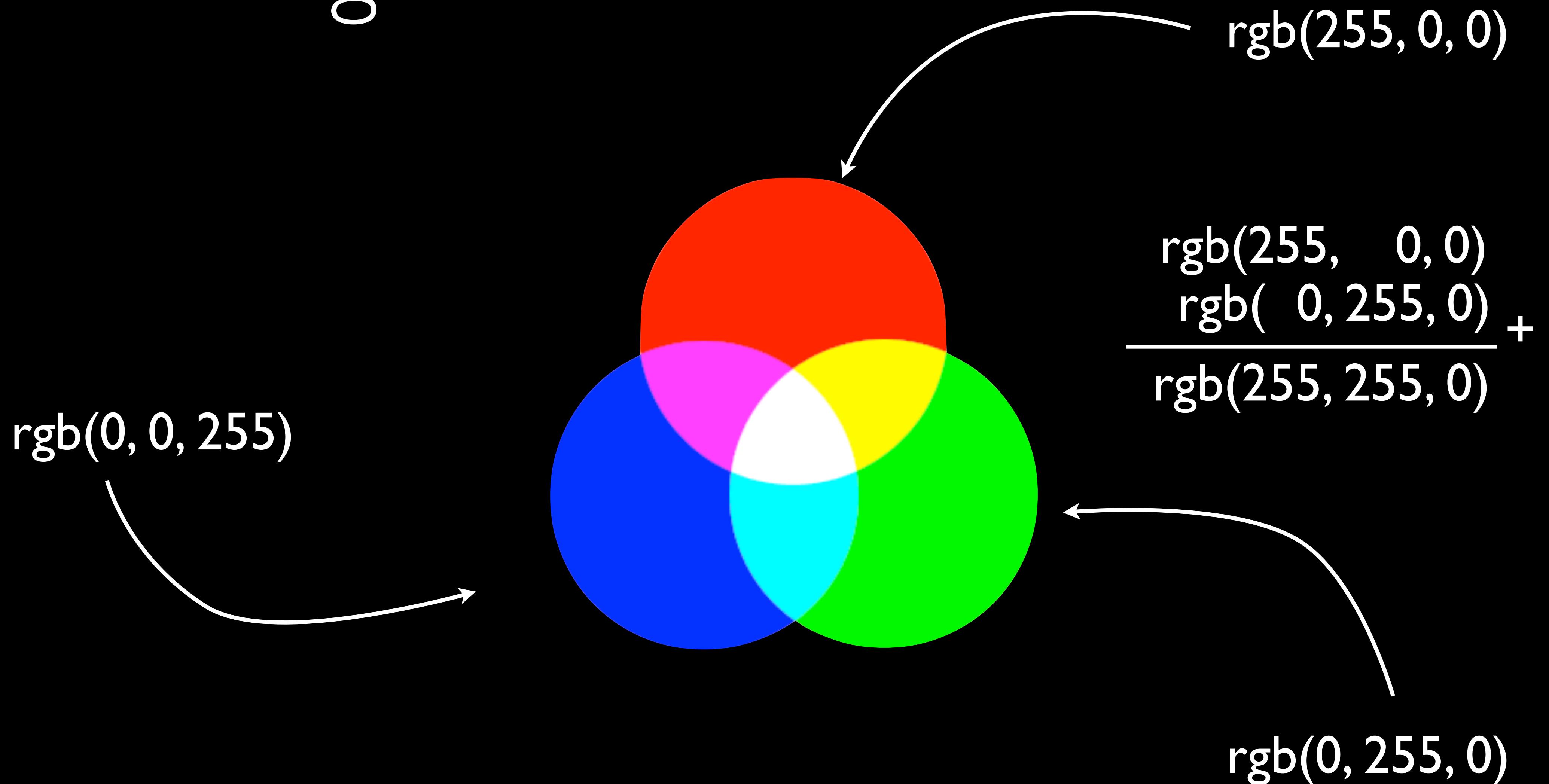
Adding Colors



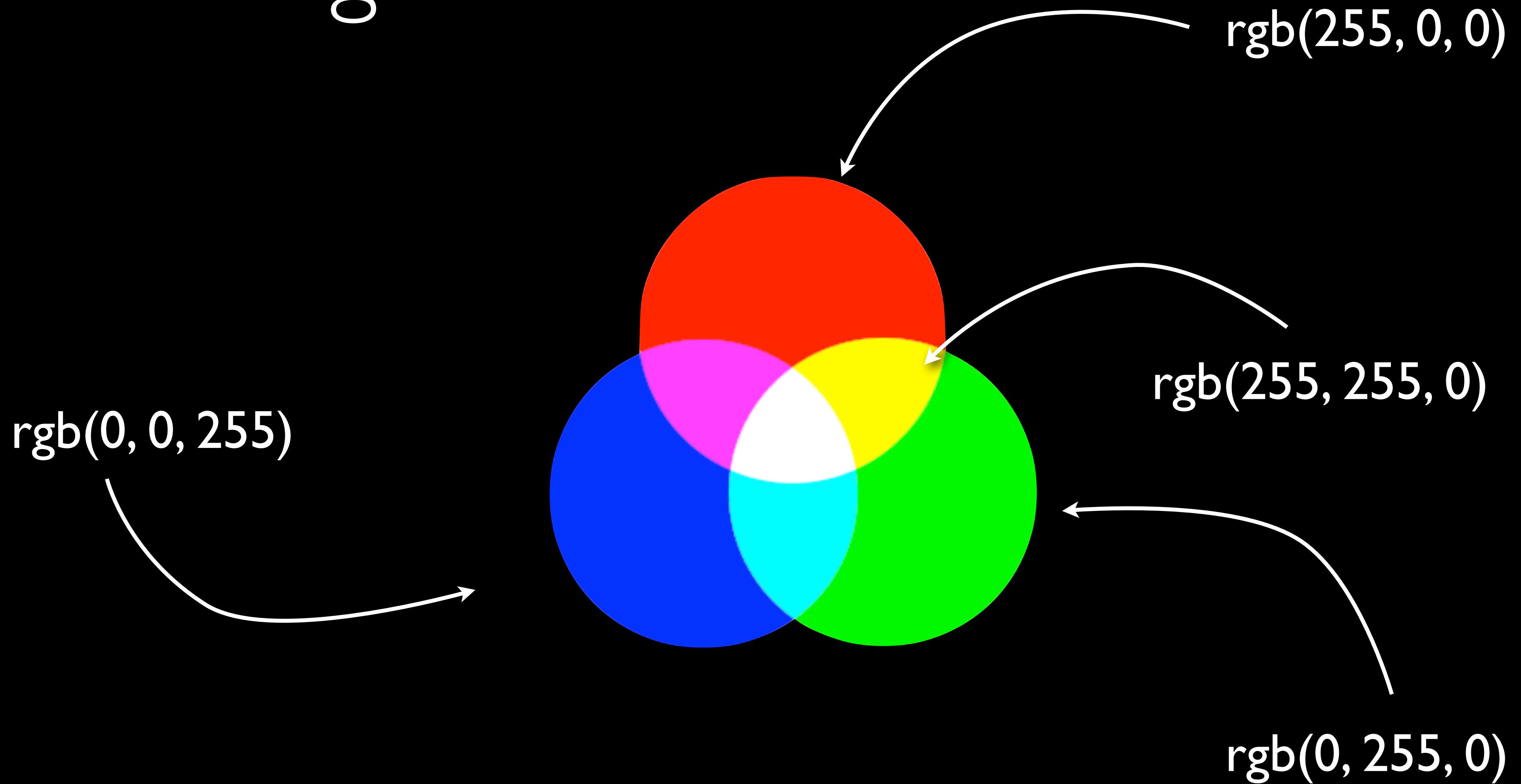
Adding Colors



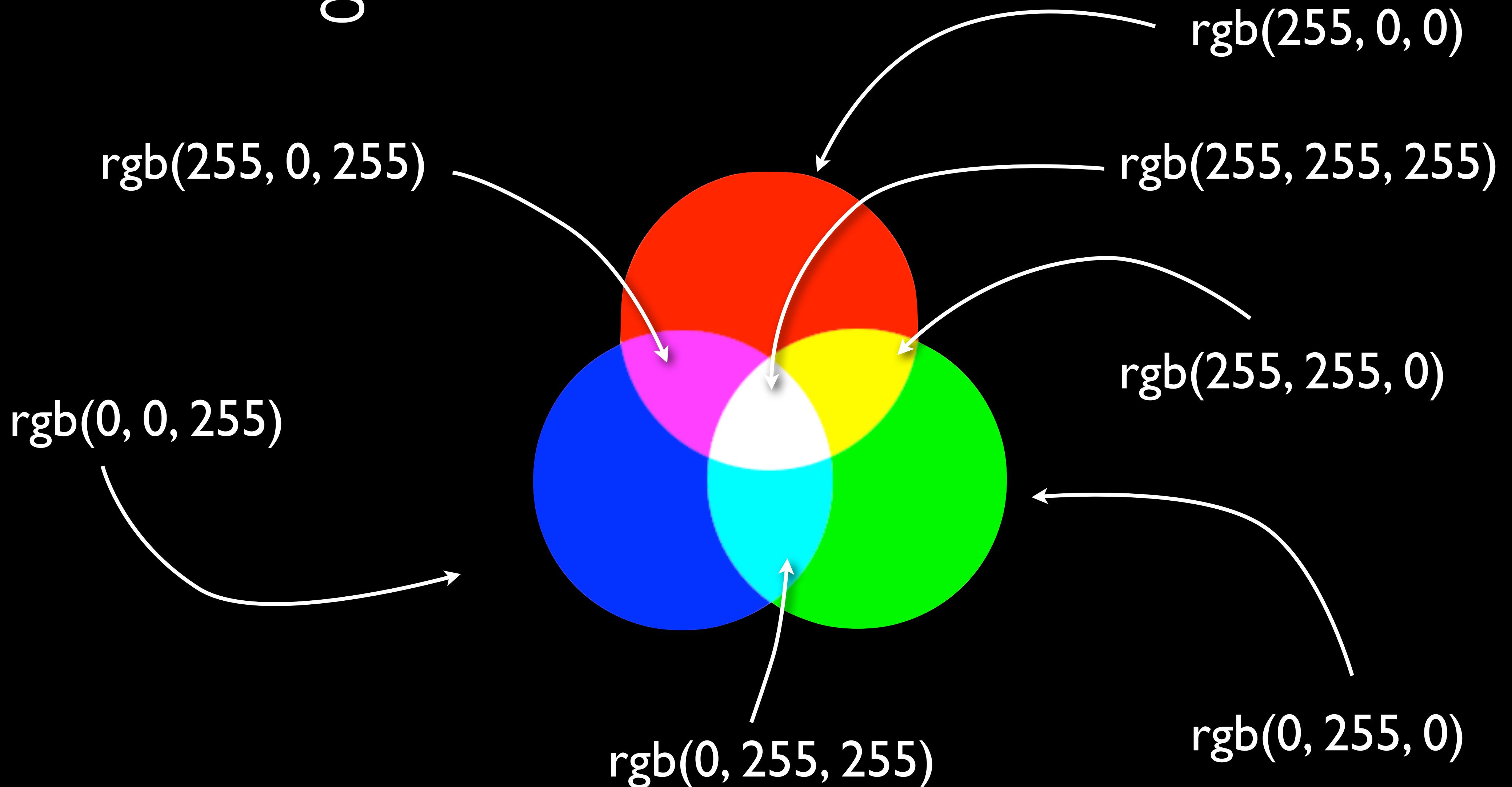
Adding Colors



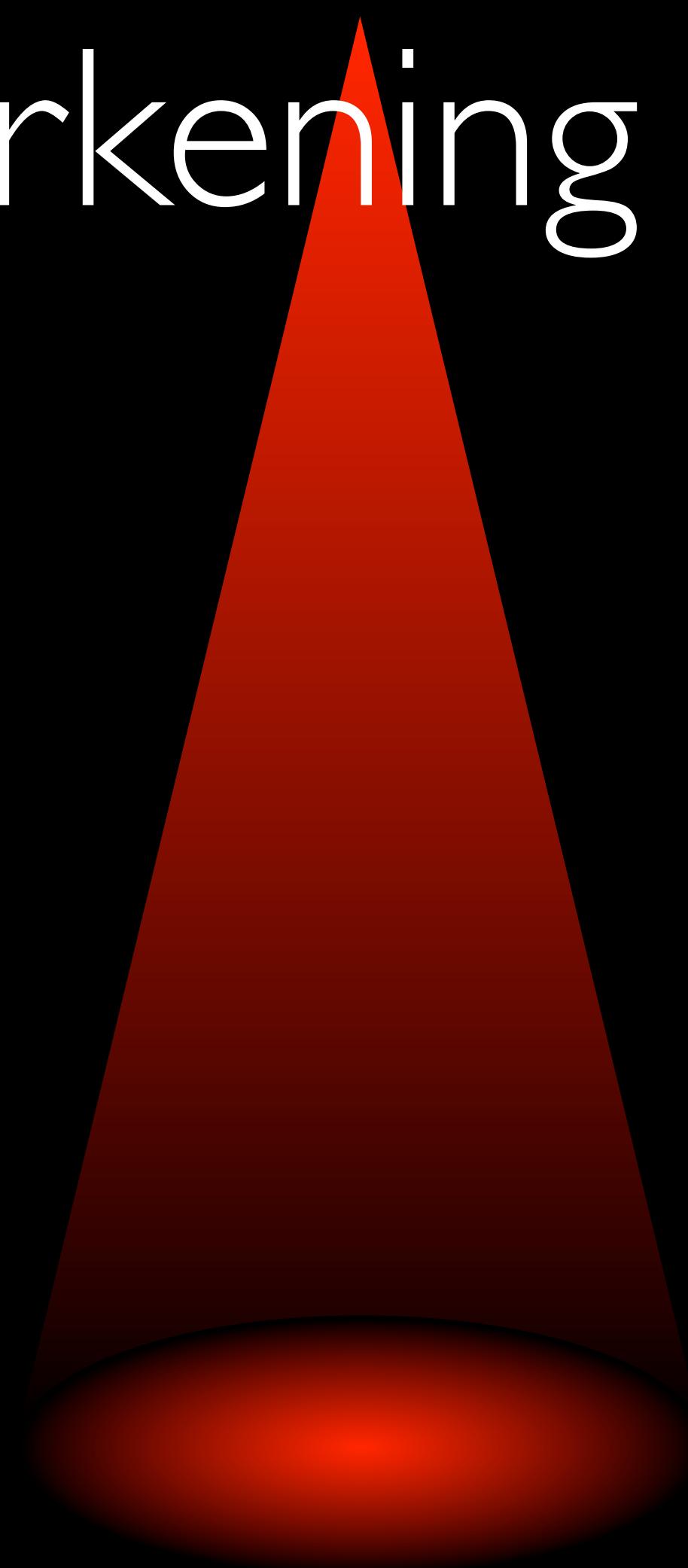
Adding Colors



Adding Colors

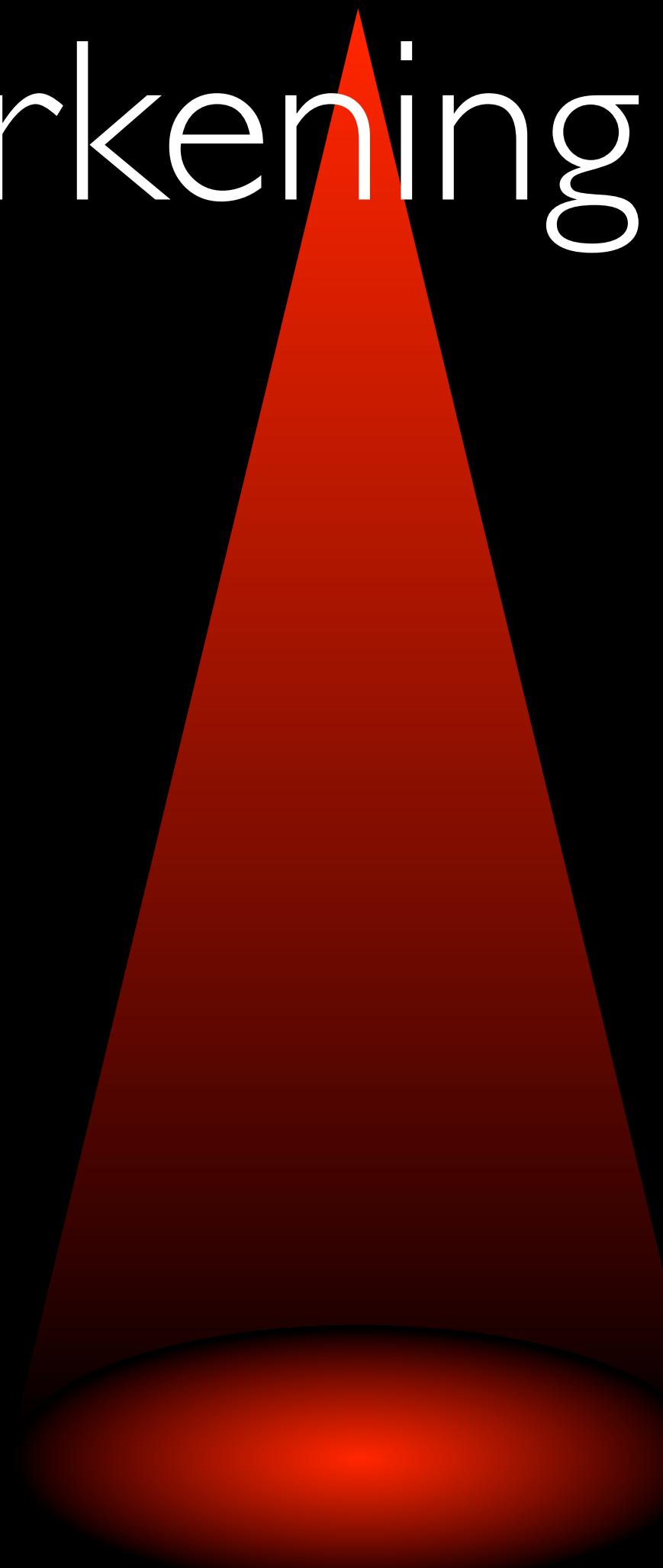


Darkening Colors

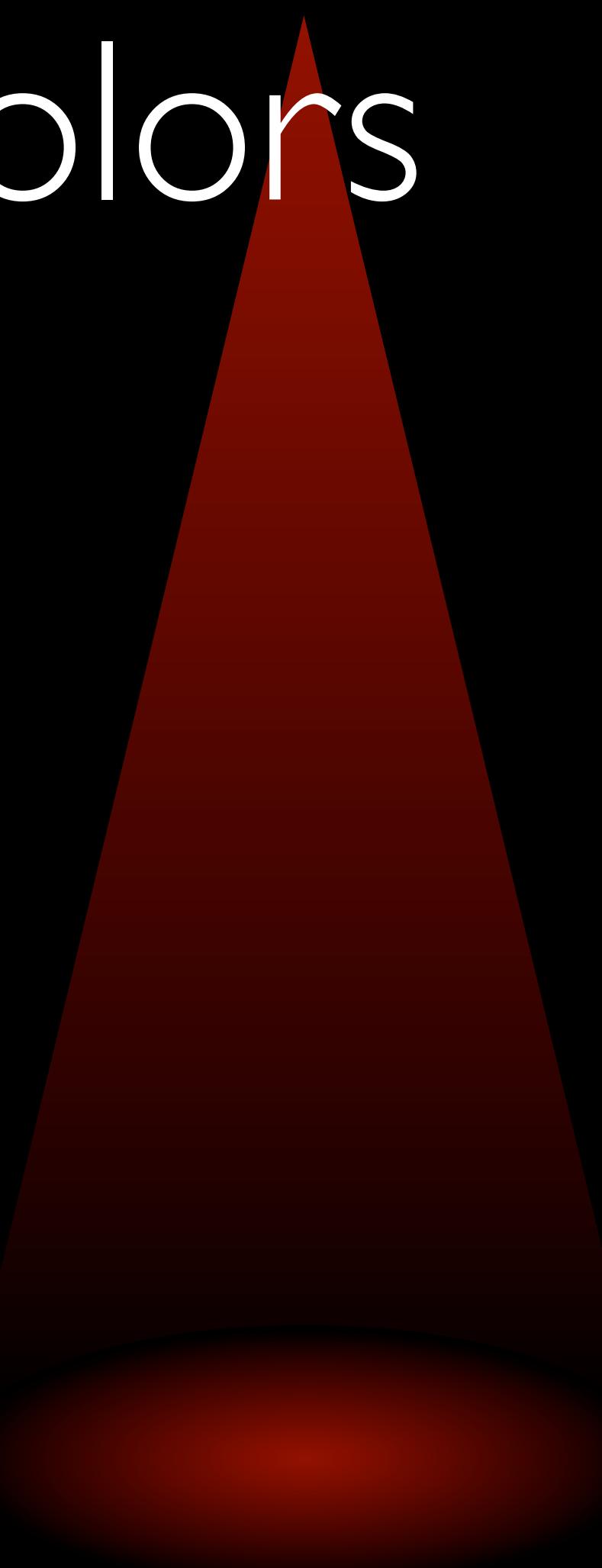


rgb(255, 0, 0)

Darkening Colors

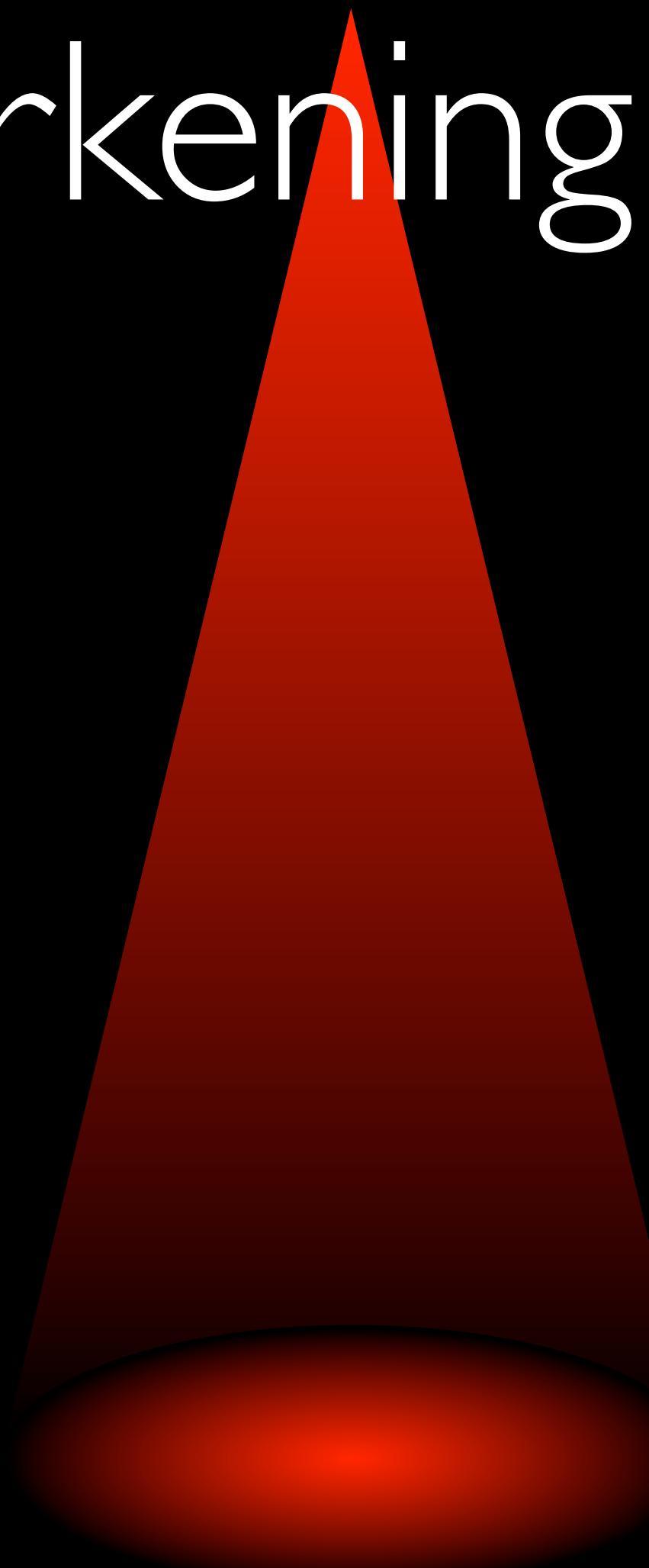


rgb(255, 0, 0)

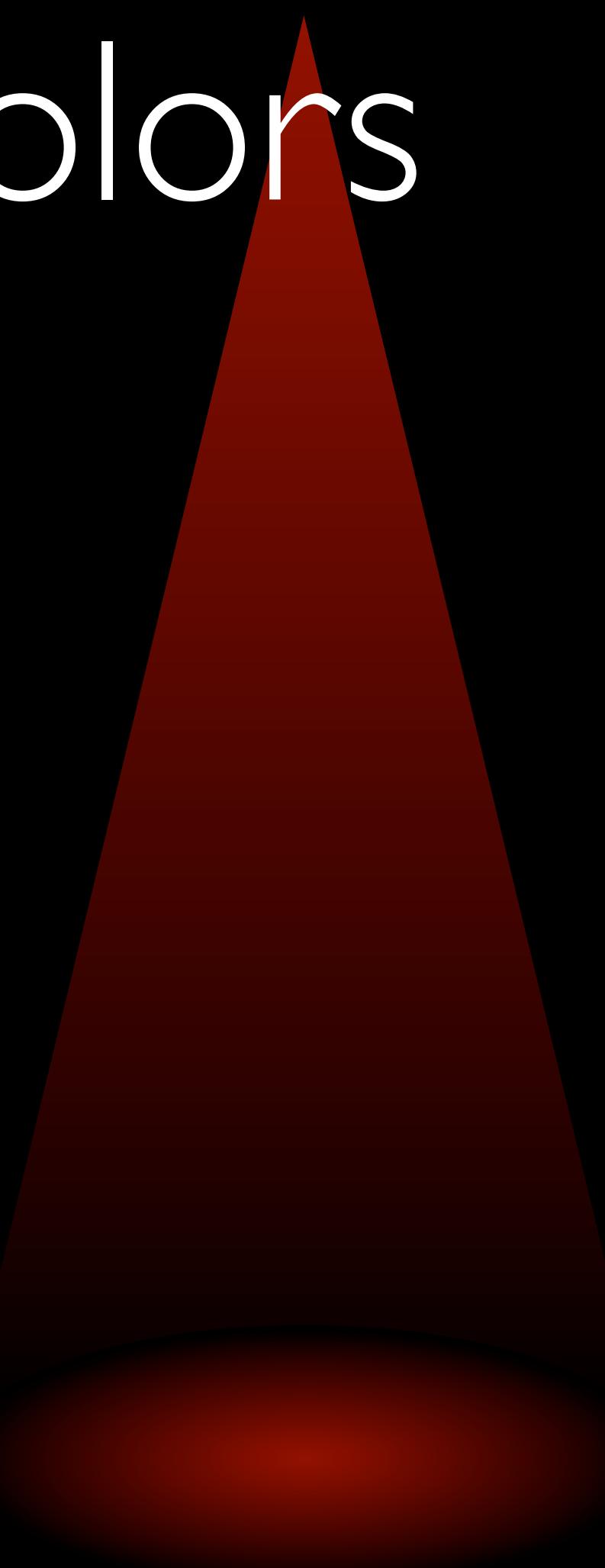


rgb(127, 0, 0)

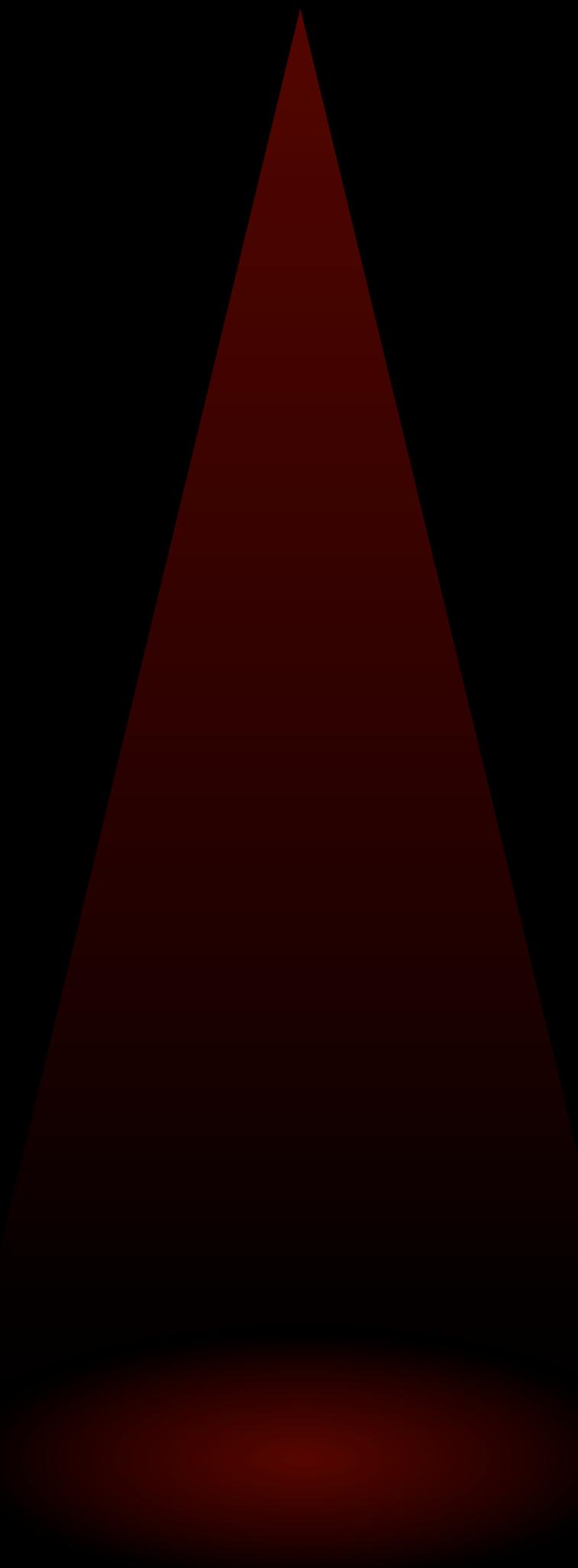
Darkening Colors



`rgb(255, 0, 0)`



`rgb(127, 0, 0)`



`rgb(63, 0, 0)`

Adding and Darkening Colors

`rgb(255, 0, 0)`

`rgb(0, 255, 0)`

`rgb(255, 255, 0)`

Adding and Darkening Colors

`rgb(255, 0, 0)`

`rgb(0, 127, 0)`

`rgb(255, 127, 0)`

Adding and Darkening Colors

`rgb(255, 0, 0)`

`rgb(0, 127, 0)`

`rgb(255, 127, 0)`

Adding and Darkening Colors

`rgb(127, 0, 0)`

`rgb(0, 255, 0)`

`rgb(127, 255, 0)`

Adding and Darkening Colors

`rgb(0, 0, 127)`

`rgb(0, 255, 0)`

`rgb(0, 255, 127)`

Adding and Darkening Colors

`rgb(0, 0, 255)`

`rgb(0, 255, 0)`

`rgb(0, 255, 255)`

Adding and Darkening Colors

`rgb(0, 0, 255)`

`rgb(0, 127, 0)`

`rgb(0, 127, 255)`

Adding and Darkening Colors

$\text{rgb}(127, 0, 0)$

$\text{rgb}(0, 0, 255)$

$\text{rgb}(127, 0, 255)$

Adding and Darkening Colors

$\text{rgb}(255, 0, 0)$

$\text{rgb}(0, 0, 255)$

$\text{rgb}(255, 0, 255)$

Adding and Darkening Colors

$\text{rgb}(255, 0, 0)$

$\text{rgb}(0, 0, 127)$

$\text{rgb}(255, 0, 127)$

Adding and Darkening Colors

$\text{rgb}(255, 0, 0)$

$\text{rgb}(0, 0, 255)$

$\text{rgb}(0, 0, 255)$

$\text{rgb}(255, 255, 255)/\text{white}$

Adding and Darkening Colors

$\text{rgb}(127, 0, 0)$

$\text{rgb}(0, 0, 127)$

$\text{rgb}(0, 0, 127)$

$\text{rgb}(127, 127, 127)/\text{grey}$

Pastel Colors

`rgb(255, 0, 0)`

`rgb(255, 0, 0)`

Pastel Colors

rgb(127, 0, 0)

rgb(127, 0, 0)

Pastel Colors

`rgb(255, 0, 0)`

`rgb(0, 0, 127)`

`rgb(0, 0, 127)`

`rgb(255, 127, 127)`

Pastel Colors

`rgb(255, 0, 0)`

`rgb(0, 0, 191)`

`rgb(0, 0, 191)`

`rgb(255, 191, 191)`

Color, tints and shades



Color, tints and shades

- Pick as base color
(e.g. `rgb(255, 0, 0)`)



Color, tints and shades

- Pick as base color
(e.g. `rgb(255, 0, 0)`)
- Decrease value of base color to create shades
(e.g. `rgb(127, 0, 0)`)



Color, tints and shades

- Pick as base color
(e.g. `rgb(255, 0, 0)`)
- Decrease value of base color to create shades
(e.g. `rgb(127, 0, 0)`)
- Increase value of other colors equivalently to create tints
(e.g. `rgb(255, 127, 127)`)



Color, tints and shades

- Pick as base color
(e.g. `rgb(255, 0, 0)`)
- Decrease value of base color to create shades
(e.g. `rgb(127, 0, 0)`)
- Increase value of other colors equivalently to create tints
(e.g. `rgb(255, 127, 127)`)

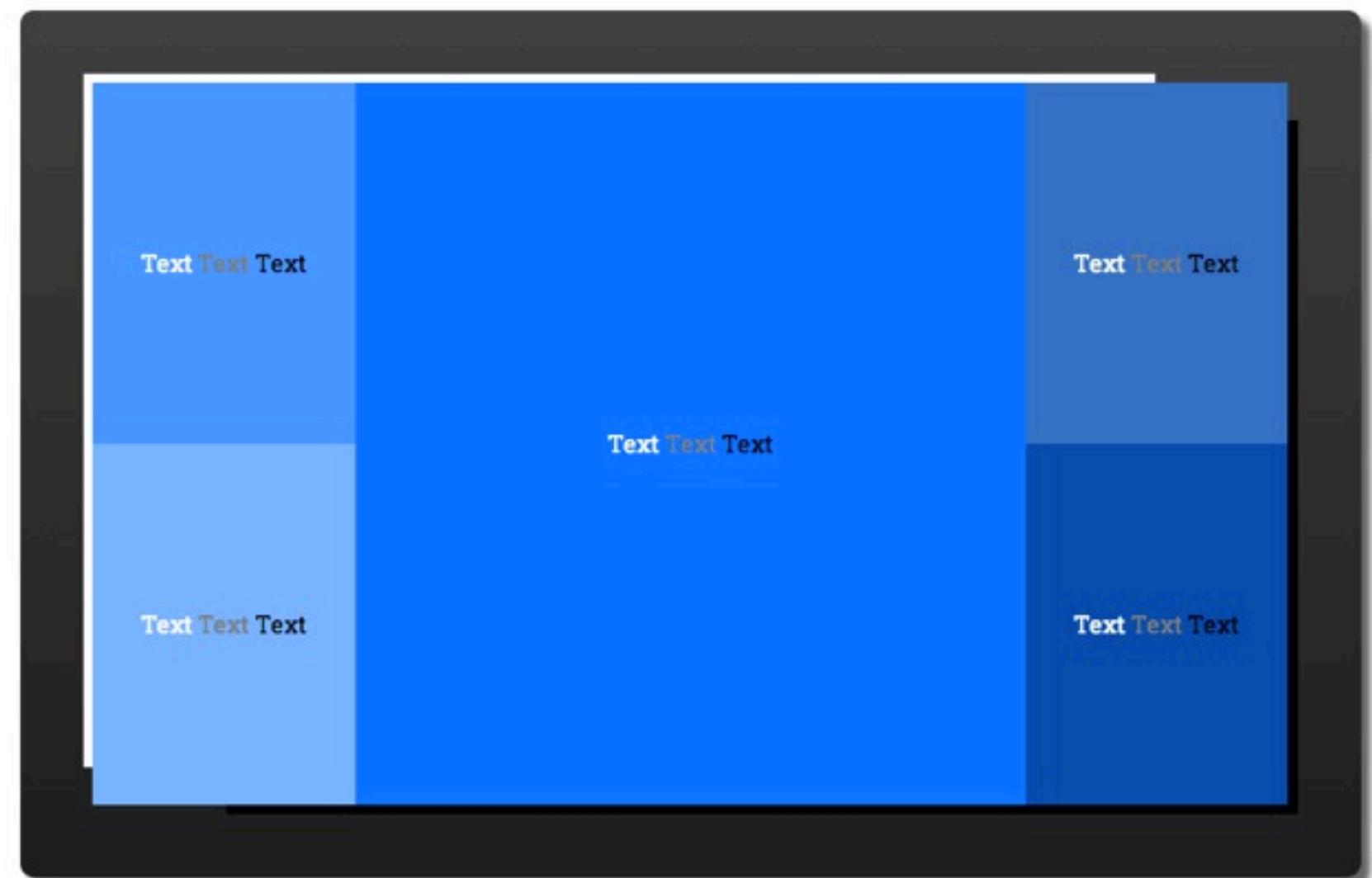


Color, tints and shades



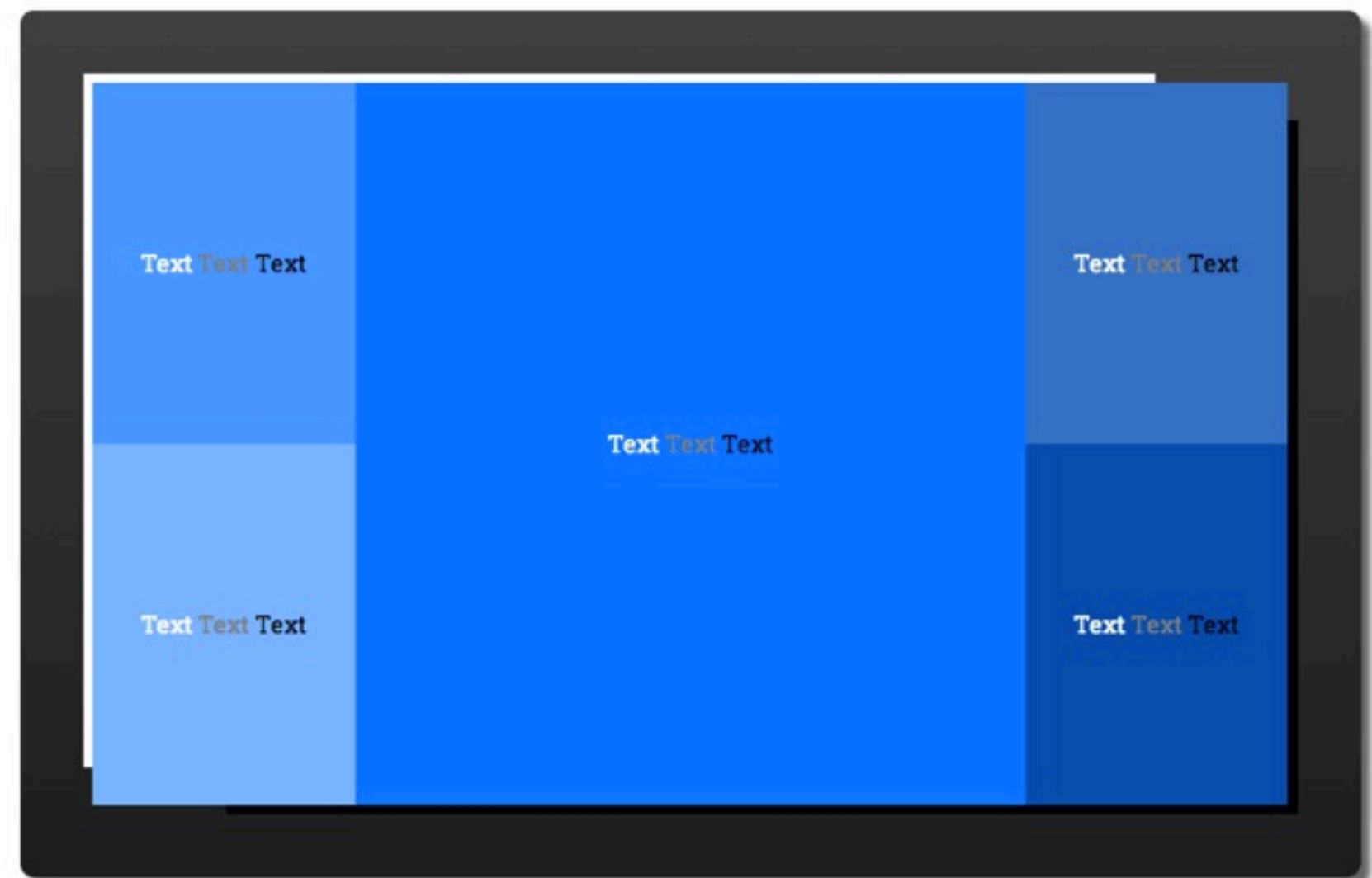
Color, tints and shades

- Not so easy when base color is a combination of R, G and B
(e.g. `rgb(0, 119, 255)`)



Color, tints and shades

- Not so easy when base color is a combination of R, G and B
(e.g. `rgb(0, 119, 255)`)
- Hang on, HSL to the rescue



From RGB to HEX

`rgb(123, 32, 101)`

`#7B2065`



RGB Color Wheel

rgb(255, 0, 0)

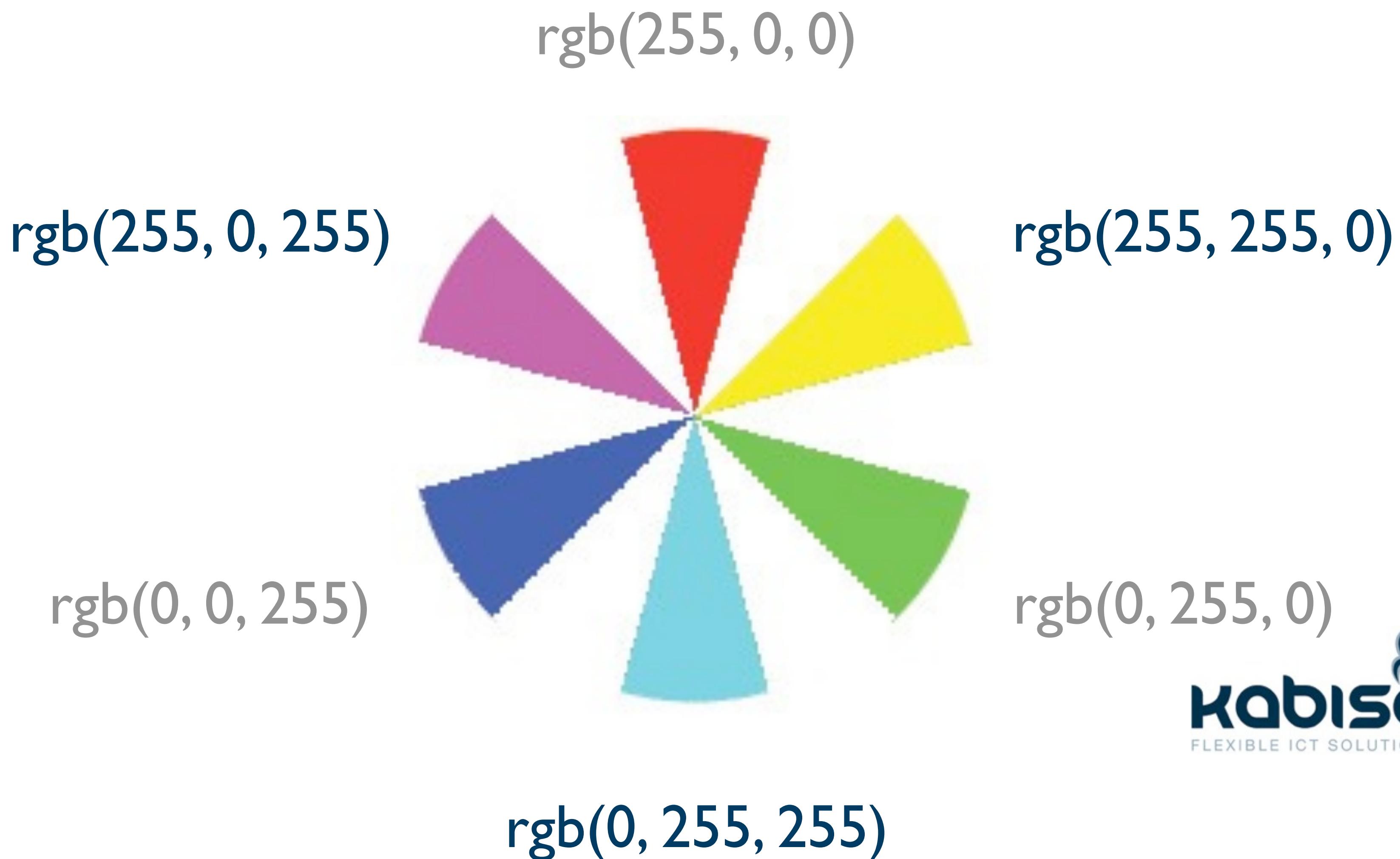


rgb(0, 0, 255)

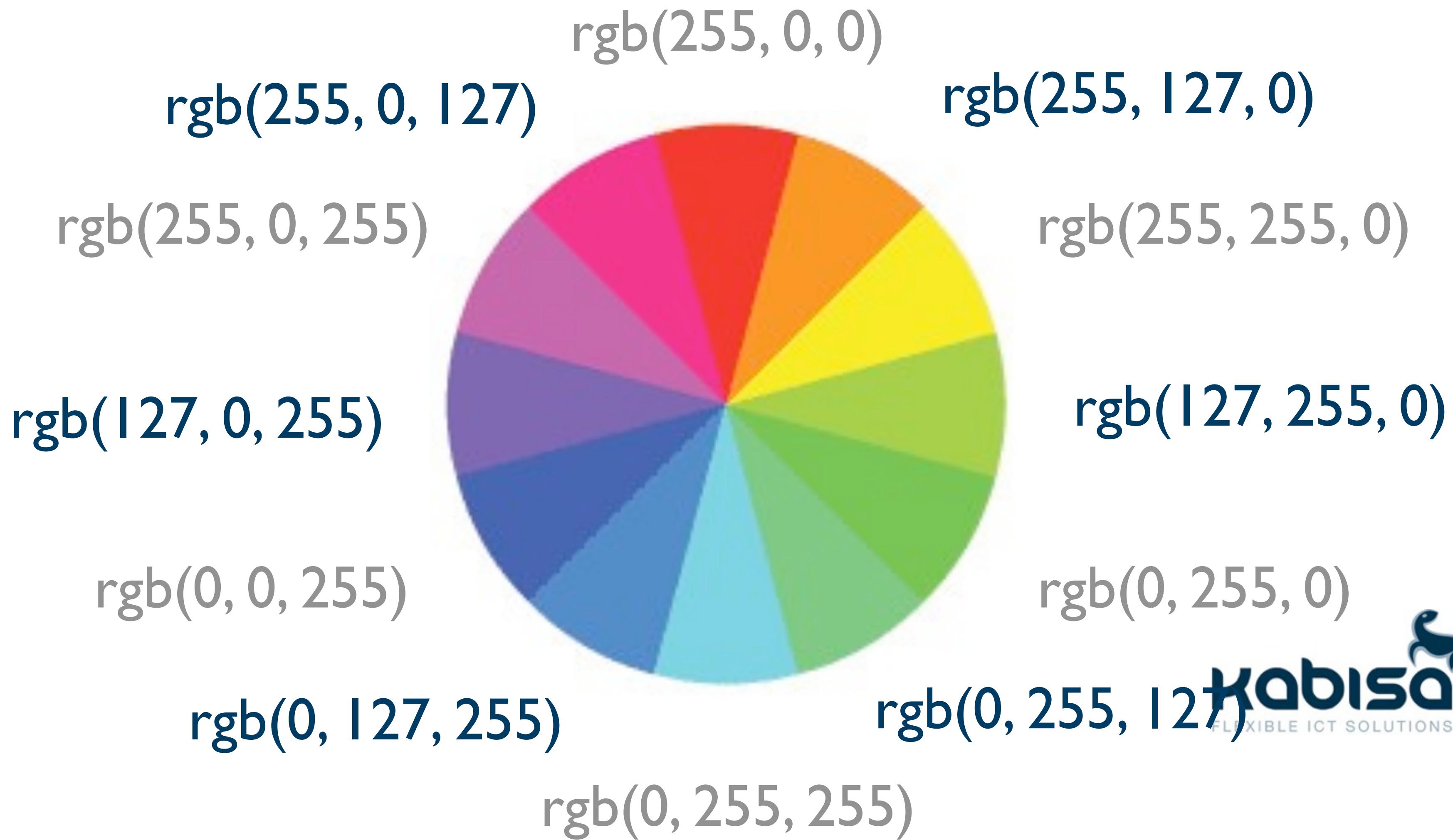
rgb(0, 255, 0)



RGB Color Wheel



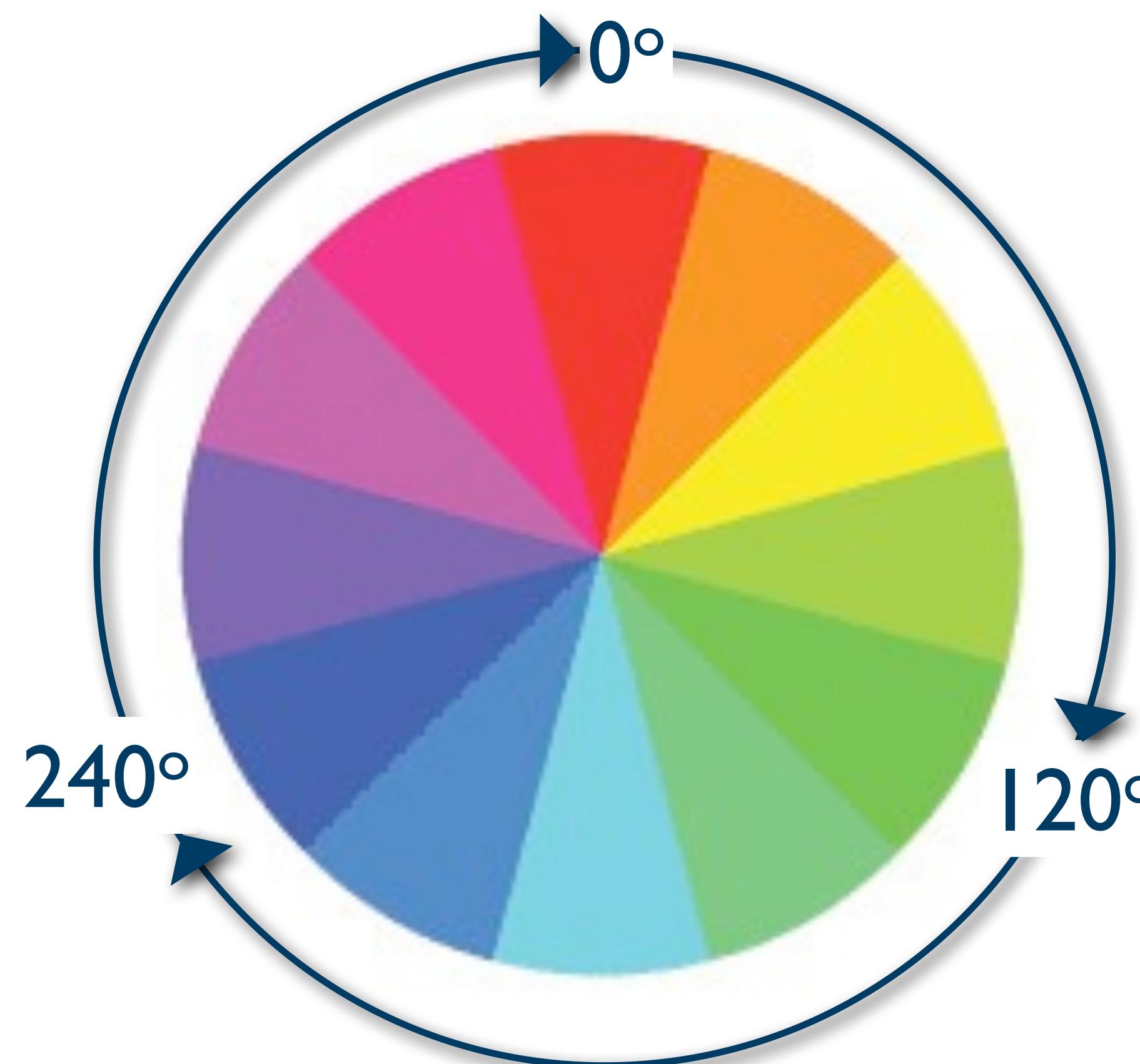
RGB Color Wheel



Hue



Hue



HSL



HSL

- Hue is the H in HSL



HSL

- Hue is the H in HSL
 - $0^\circ \leq \text{Hue} \leq 360^\circ$



HSL

- Hue is the H in HSL
 - $0^\circ \leq \text{Hue} \leq 360^\circ$



HSL

- Hue is the H in HSL
 - $0^\circ \leq \text{Hue} \leq 360^\circ$
- Saturation



HSL

- Hue is the H in HSL
 - $0^\circ \leq \text{Hue} \leq 360^\circ$
- Saturation
 - The colorfulness of a color relative to its own brightness



HSL

- Hue is the H in HSL
 - $0^\circ \leq \text{Hue} \leq 360^\circ$
- Saturation
 - The colorfulness of a color relative to its own brightness
 - $0^\circ \leq \text{Saturation} \leq 100^\circ$



HSL

- Hue is the H in HSL
 - $0^\circ \leq \text{Hue} \leq 360^\circ$
- Saturation
 - The colorfulness of a color relative to its own brightness
 - $0^\circ \leq \text{Saturation} \leq 100^\circ$



HSL

- Hue is the H in HSL
 - $0^\circ \leq \text{Hue} \leq 360^\circ$
- Saturation
 - The colorfulness of a color relative to its own brightness
 - $0^\circ \leq \text{Saturation} \leq 100^\circ$
- Lightness



HSL

- Hue is the H in HSL
 - $0^\circ \leq \text{Hue} \leq 360^\circ$
- Saturation
 - The colorfulness of a color relative to its own brightness
 - $0^\circ \leq \text{Saturation} \leq 100^\circ$
- Lightness
 - The light/darkness of a color from white to black



HSL

- Hue is the H in HSL
 - $0^\circ \leq \text{Hue} \leq 360^\circ$
- Saturation
 - The colorfulness of a color relative to its own brightness
 - $0^\circ \leq \text{Saturation} \leq 100^\circ$
- Lightness
 - The light/darkness of a color from white to black
 - $0^\circ \leq \text{Lightness} \leq 100^\circ$



HSL

- Hue is the H in HSL
 - $0^\circ \leq \text{Hue} \leq 360^\circ$
- Saturation
 - The colorfulness of a color relative to its own brightness
 - $0^\circ \leq \text{Saturation} \leq 100^\circ$
- Lightness
 - The light/darkness of a color from white to black
 - $0^\circ \leq \text{Lightness} \leq 100^\circ$



HSL



HSL

- It's easier to express color using HSL than RGB



HSL

- It's easier to express color using HSL than RGB
- Guess the color by setting the H-value



HSL

- It's easier to express color using HSL than RGB
- Guess the color by setting the H-value
- Adjust the saturation and lightness



HSL

- It's easier to express color using HSL than RGB
- Guess the color by setting the H-value
- Adjust the saturation and lightness
- Monochrome color schemes are very easy to create



Color, tints and shades



Color, tints and shades

- Pick as base color
(e.g. HSL(0, 100%, 50%))



Color, tints and shades

- Pick as base color
(e.g. HSL(0, 100%, 50%))
- Change saturation to move to grey
(e.g. HSL(0, 60%, 50%))



Color, tints and shades

- Pick as base color
(e.g. HSL(0, 100%, 50%))
- Change saturation to move to grey
(e.g. HSL(0, 60%, 50%))
- Change lightness to lighten or darken the color
(e.g. HSL(0, 100%, 73%) or
HSL(0, 100%, 33%))

Color, tints and shades

- Pick as base color
(e.g. HSL(0, 100%, 50%))
- Change saturation to move to grey
(e.g. HSL(0, 60%, 50%))
- Change lightness to lighten or darken the color
(e.g. HSL(0, 100%, 73%) or
HSL(0, 100%, 33%))



Easy to change scheme



Easy to change scheme

- Replace Hue with a different value

HSL(0, 100%, 50%)

HSL(0, 60%, 50%)

HSL(0, 100%, 73%)

HSL(0, 100%, 33%)



Easy to change scheme

- Replace Hue with a different value

HSL(0, 100%, 50%)

HSL(0, 60%, 50%)

HSL(0, 100%, 73%)

HSL(0, 100%, 33%)

HSL(212, 100%, 50%)

HSL(212, 60%, 50%)

HSL(212, 100%, 73%)

HSL(212, 100%, 33%)



Sass



Sass

```
$hue      : 0;  
$saturation: 100%;  
$lighting  : 50%;  
$base-color: hsl($hue, $saturation, $lighting);  
  
.base-color {  
  background: $base-color;  
}  
.variant-1 {  
  background: adjust-color($base-color, $saturation: -40%);  
}  
.variant-2 {  
  background: adjust-color($base-color, $lightness: 23%);  
}  
.variant-3 {  
  background: adjust-color($base-color, $lightness: -17%);  
}
```



Sass

```
$hue      : 0;  
$saturation: 100%;  
$lighting  : 50%;  
$base-color: hsl($hue, $saturation, $lighting);  
  
.base-color {  
  background: $base-color;  
}  
.variant-1 {  
  background: adjust-color($base-color, $saturation: -40%);  
}  
.variant-2 {  
  background: adjust-color($base-color, $lightness: 23%);  
}  
.variant-3 {  
  background: adjust-color($base-color, $lightness: -17%);  
}
```



Change base color

```
$hue      : 212;  
$saturation: 100%;  
$lighting  : 50%;  
$base-color: hsl($hue, $saturation, $lighting);
```

```
.base-color {  
  background: $base-color;  
}  
.variant-1 {  
  background: adjust-color($base-color, $saturation: -40%);  
}  
.variant-2 {  
  background: adjust-color($base-color, $lightness: 23%);  
}  
.variant-3 {  
  background: adjust-color($base-color, $lightness: -17%);  
}
```



FLEXIBLE ICT SOLUTIONS

Color Schemes



Monochromatic

- Based on a single color tint, uses only variations made by changing its saturation and brightness



Monochromatic

- Based on a single color tint, uses only variations made by changing its saturation and brightness

```
$hue      : 212;  
$saturation: 100%;  
$lighting  : 50%;  
$base-color: hsl($hue, $saturation, $lighting);  
  
.pri-0 {  
  background: $base-color;  
}  
.pri-1 {  
  background: adjust-color($base-color, $saturation: -40%);  
}  
.pri-2 {  
  background: adjust-color($base-color, $lightness: 23%);  
}  
.pri-3 {  
  background: adjust-color($base-color, $lightness: -17%);  
}
```



FLEXIBLE ICT SOLUTIONS

Monochromatic

- Based on a single color tint, uses only variations made by changing its saturation and brightness



```
$hue      : 212;  
$saturation: 100%;  
$lighting  : 50%;  
$base-color: hsl($hue, $saturation, $lighting);  
  
.pri-0 {  
  background: $base-color;  
}  
.pri-1 {  
  background: adjust-color($base-color, $saturation: -40%);  
}  
.pri-2 {  
  background: adjust-color($base-color, $lightness: 23%);  
}  
.pri-3 {  
  background: adjust-color($base-color, $lightness: -17%);  
}
```

Complementary

- The primary color is supplemented with its complement (color on the opposite side of the wheel)



Complementary

- The primary color is supplemented with its complement (color on the opposite side of the wheel)



Complementary

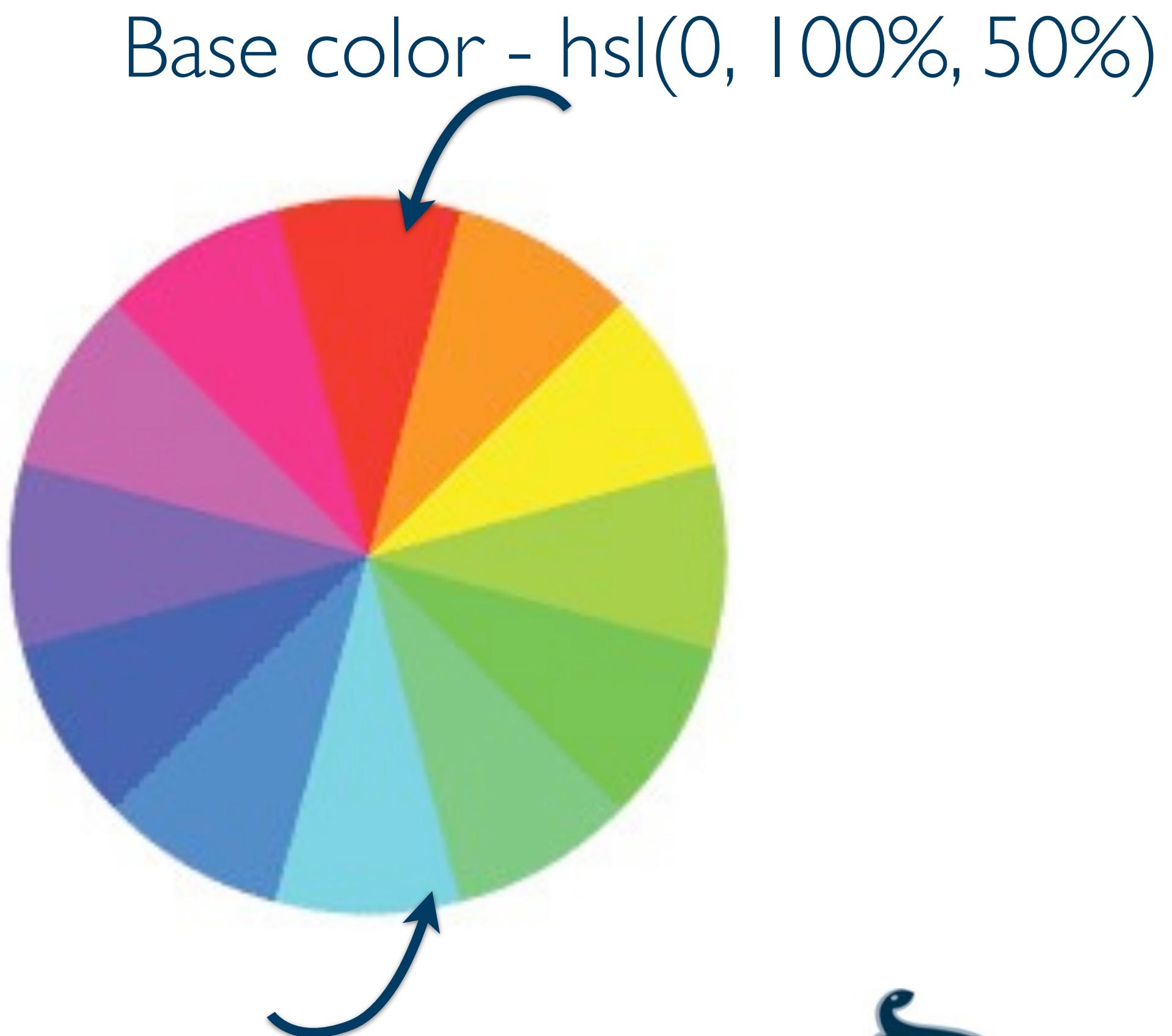
- The primary color is supplemented with its complement (color on the opposite side of the wheel)

Base color - hsl(0, 100%, 50%)



Complementary

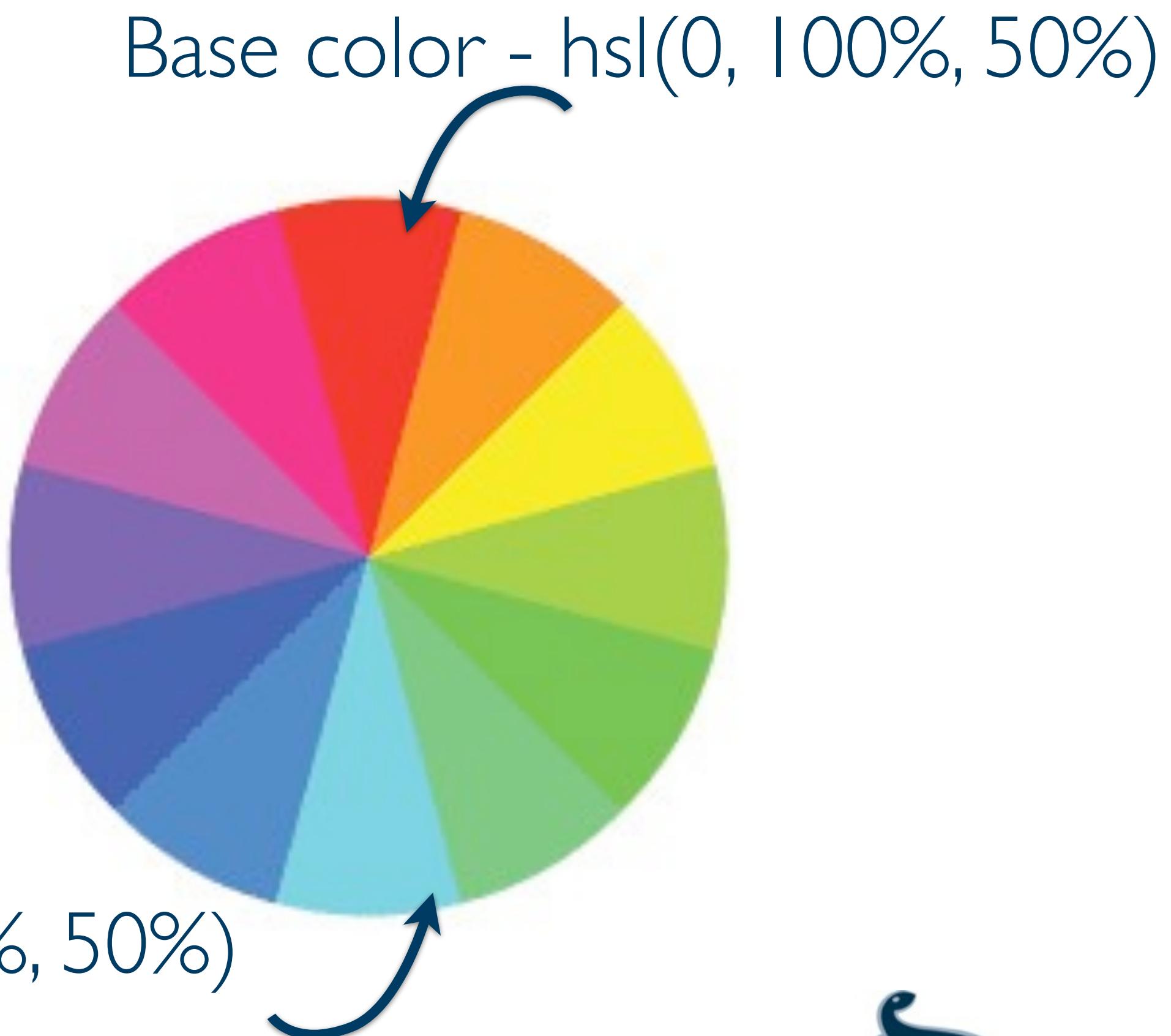
- The primary color is supplemented with its complement (color on the opposite side of the wheel)



Complementary

- The primary color is supplemented with its complement (color on the opposite side of the wheel)

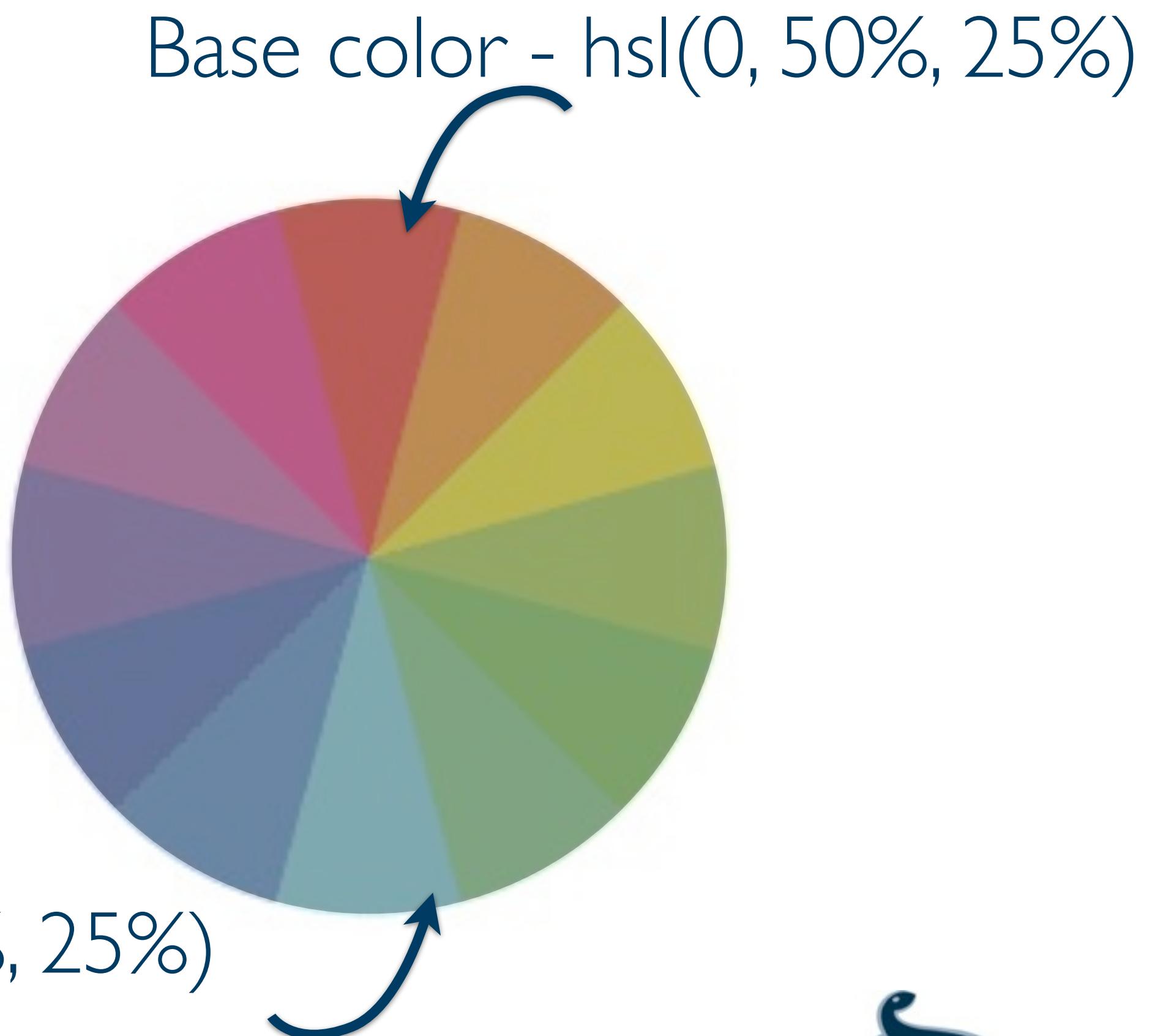
Complement - $\text{hsl}(180, 100\%, 50\%)$



Complementary

- The base color is supplemented with its complement (color on the opposite side of the wheel)

Complement - $hsl(180, 50%, 25\%)$



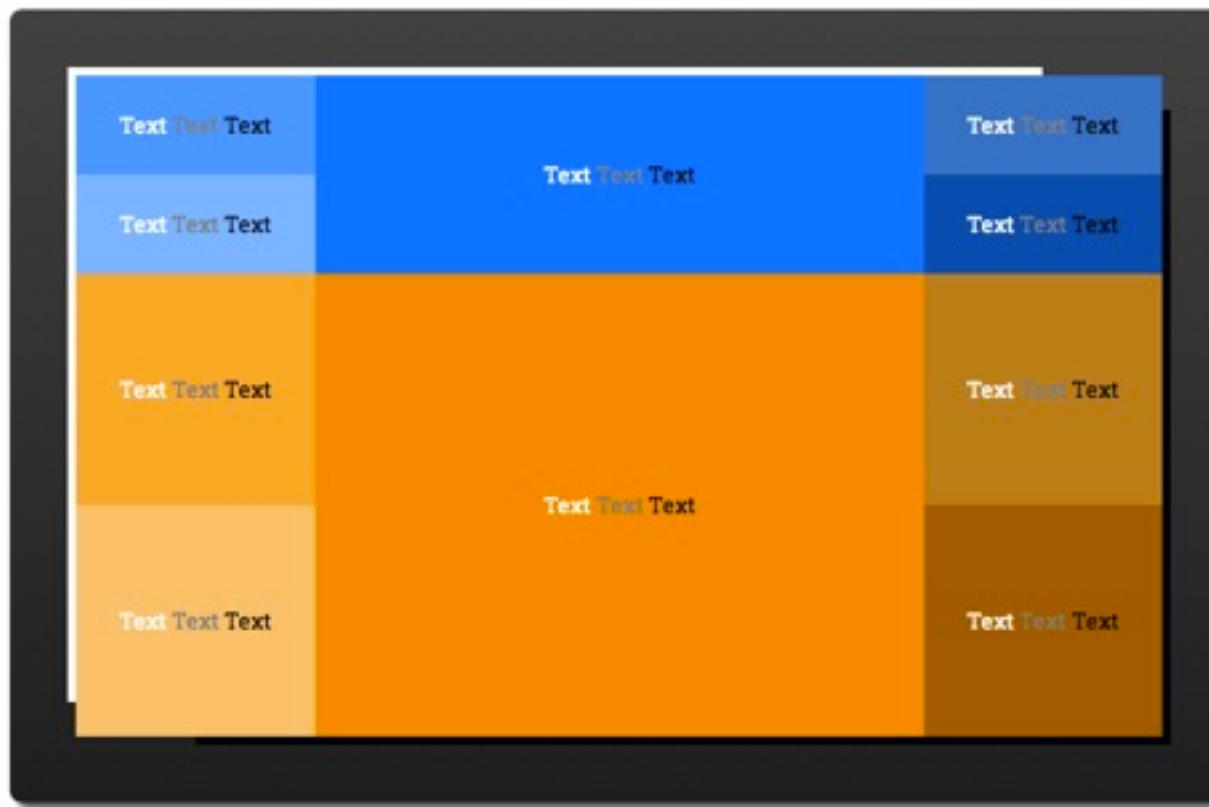
Complementary

```
$base-color: hsl($hue, $saturation, $lighting);  
$compl-color: complement($base-color);  
  
.compl-0 {  
    background: $compl-color;  
}  
.compl-1 {  
    background: adjust-color($compl-color, $saturation: -40%);  
}  
.compl-2 {  
    background: adjust-color($compl-color, $lightness: 23%);  
}  
.compl-3 {  
    background: adjust-color($compl-color, $lightness: -17%);  
}
```



Complementary

```
$base-color: hsl($hue, $saturation, $lighting);  
$compl-color: complement($base-color);  
  
.compl-0 {  
    background: $compl-color;  
}  
.compl-1 {  
    background: adjust-color($compl-color, $saturation: -40%);  
}  
.compl-2 {  
    background: adjust-color($compl-color, $lightness: 23%);  
}  
.compl-3 {  
    background: adjust-color($compl-color, $lightness: -17%);  
}
```



Triad

- The base color is supplemented with two colors, placed identically on both sides of its complement



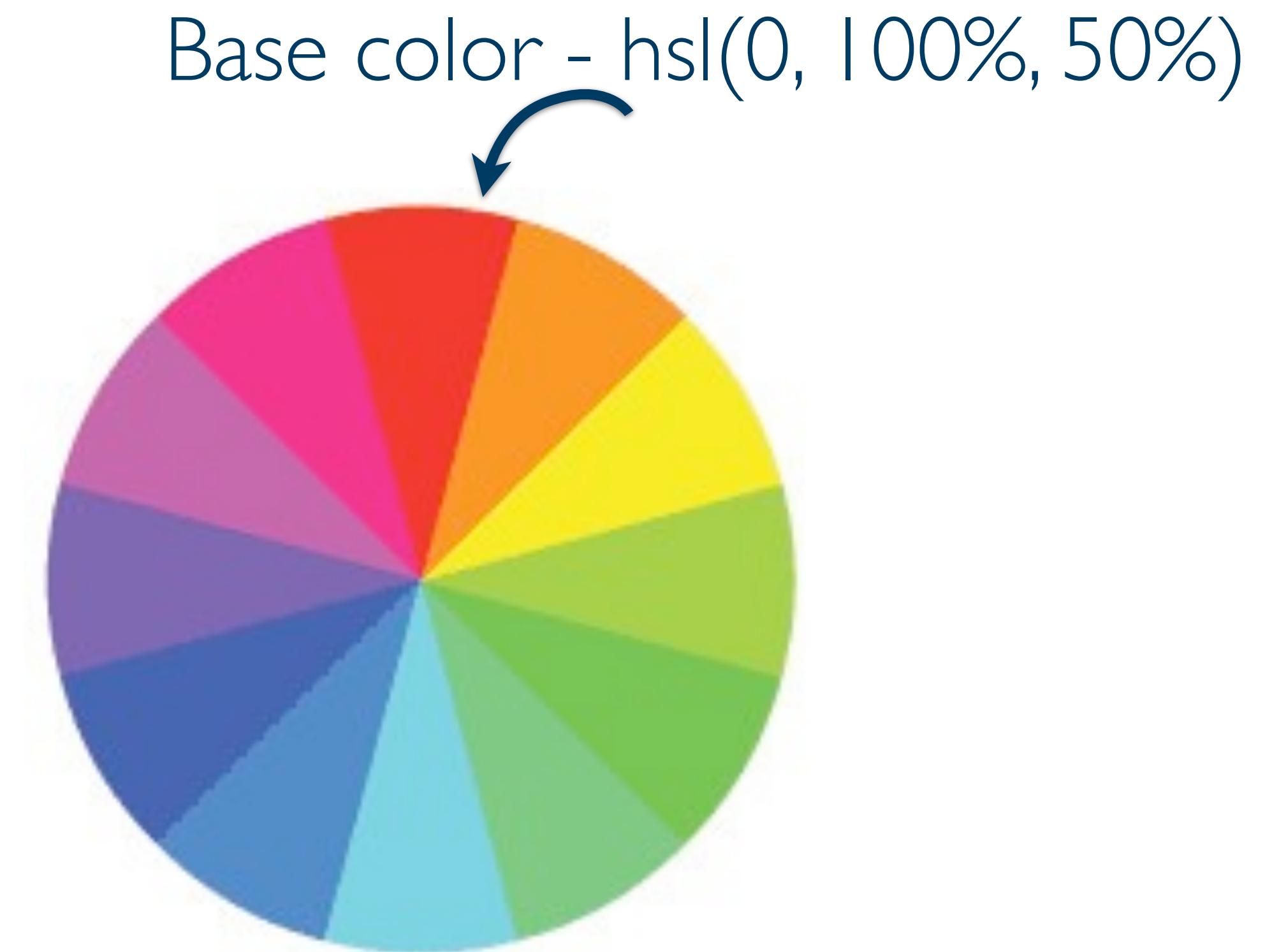
Triad

- The base color is supplemented with two colors, placed identically on both sides of its complement



Triad

- The base color is supplemented with two colors, placed identically on both sides of its complement



Triad

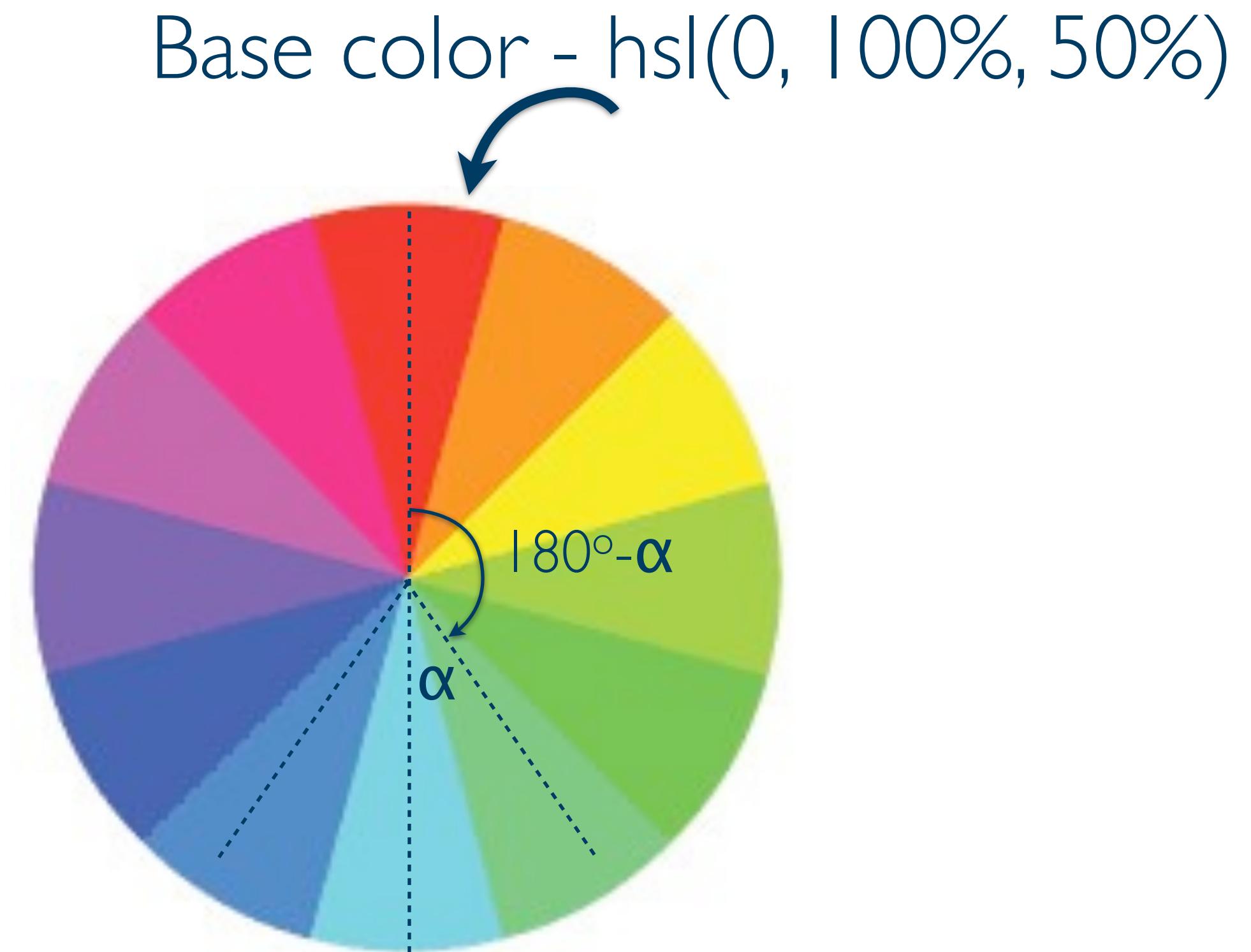
- The base color is supplemented with two colors, placed identically on both sides of its complement

Base color - hsl(0, 100%, 50%)



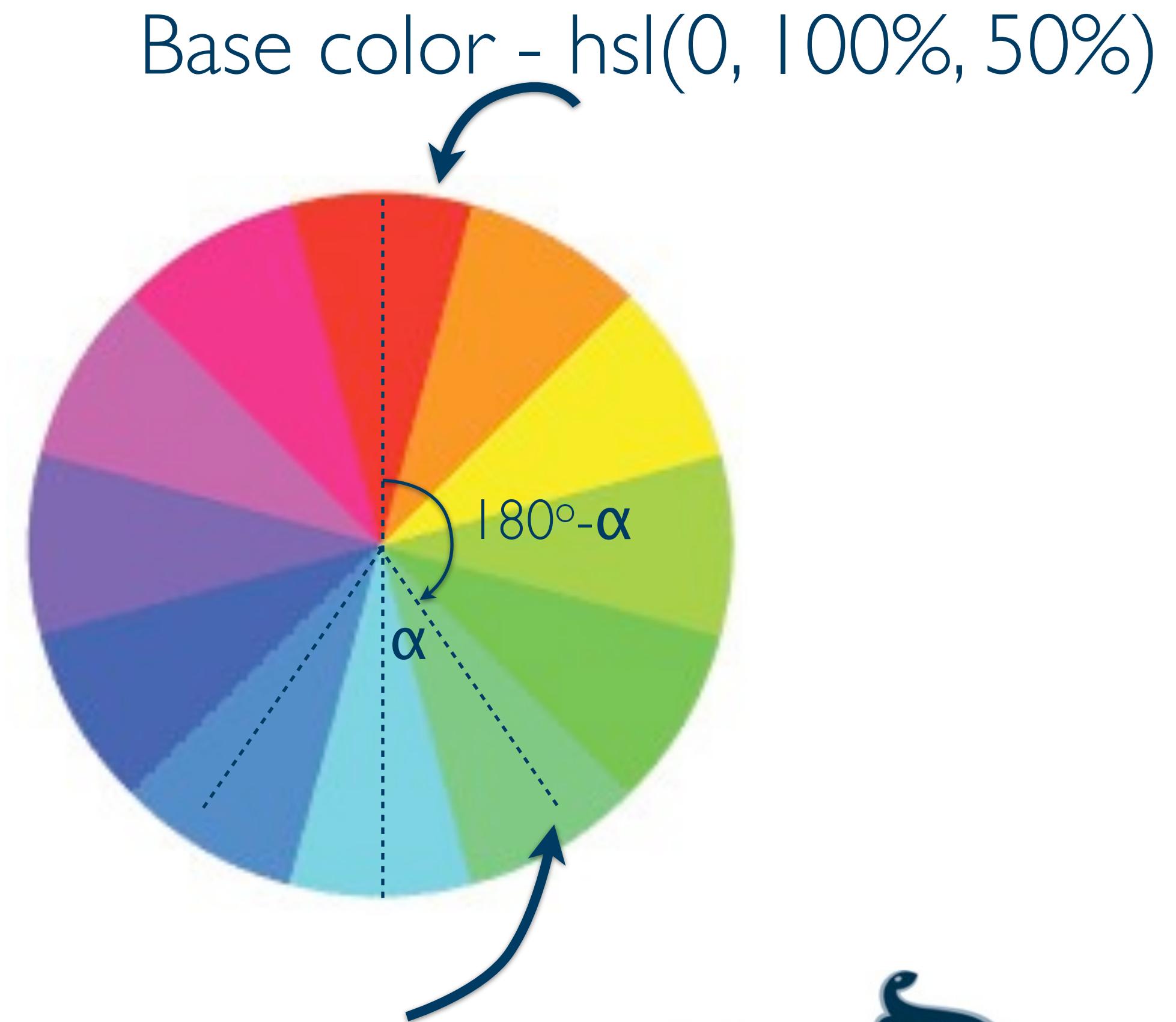
Triad

- The base color is supplemented with two colors, placed identically on both sides of its complement



Triad

- The base color is supplemented with two colors, placed identically on both sides of its complement



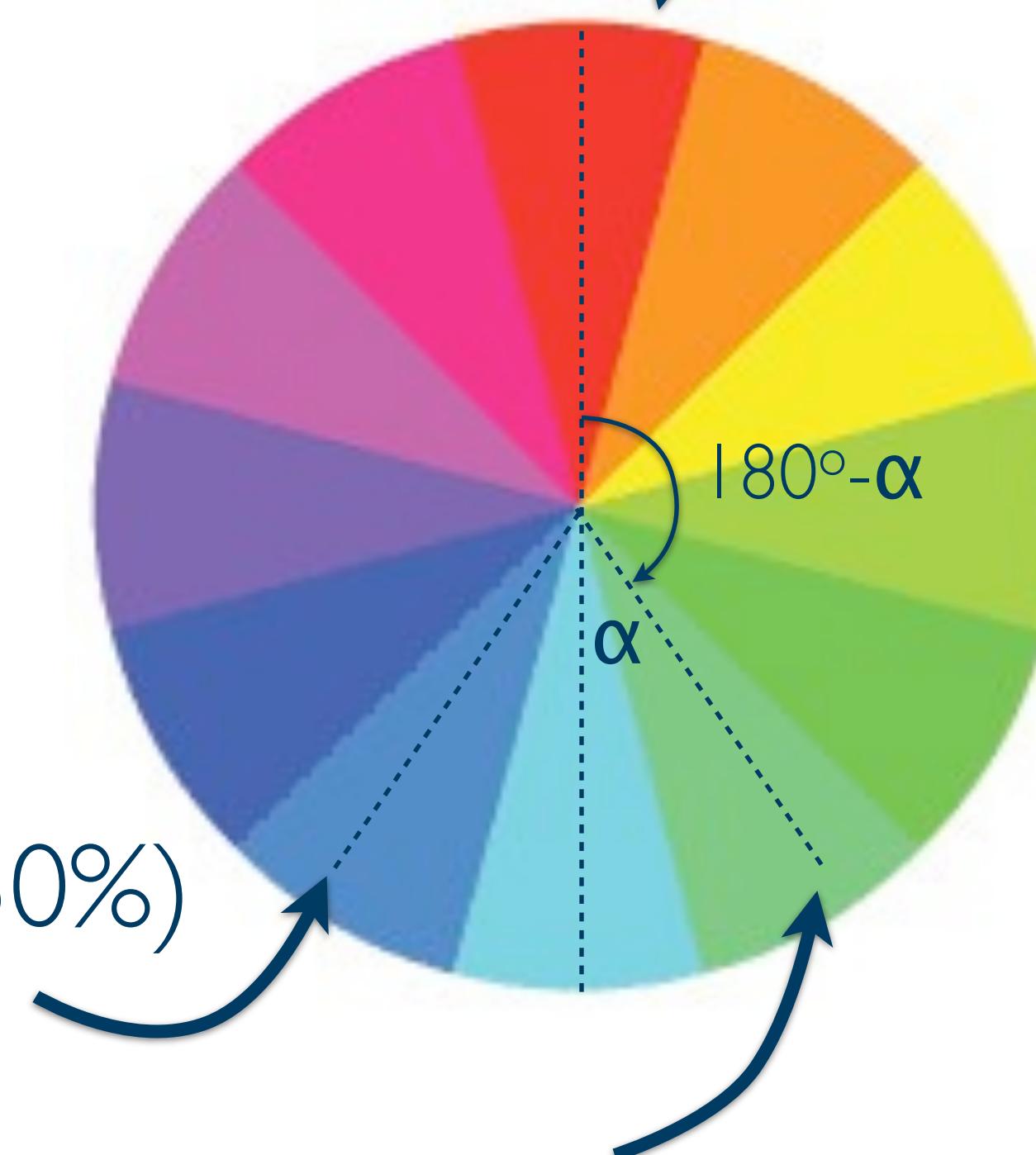
Triad

- The base color is supplemented with two colors, placed identically on both sides of its complement

Secondary 1 - hsl(210, 100%, 50%)

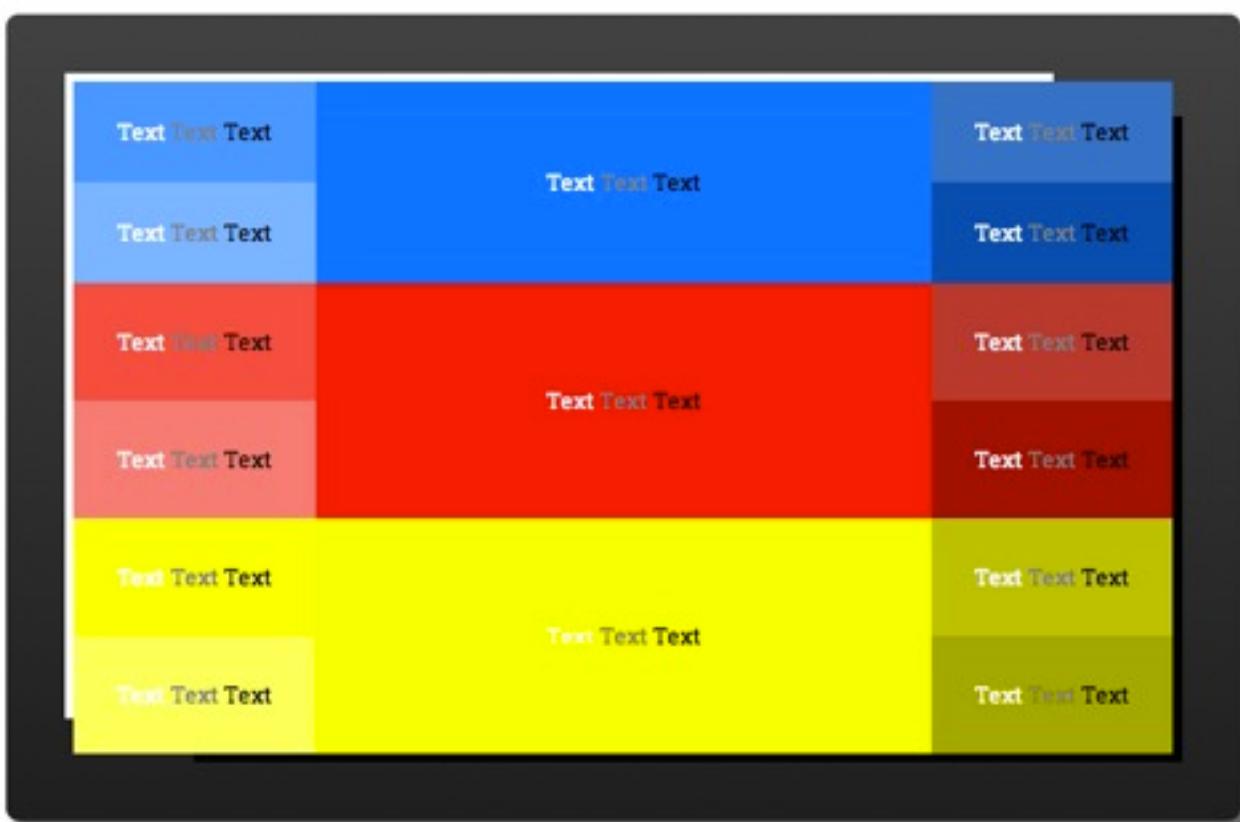
Secondary 2 - hsl(150, 100%, 50%)

Base color - hsl(0, 100%, 50%)

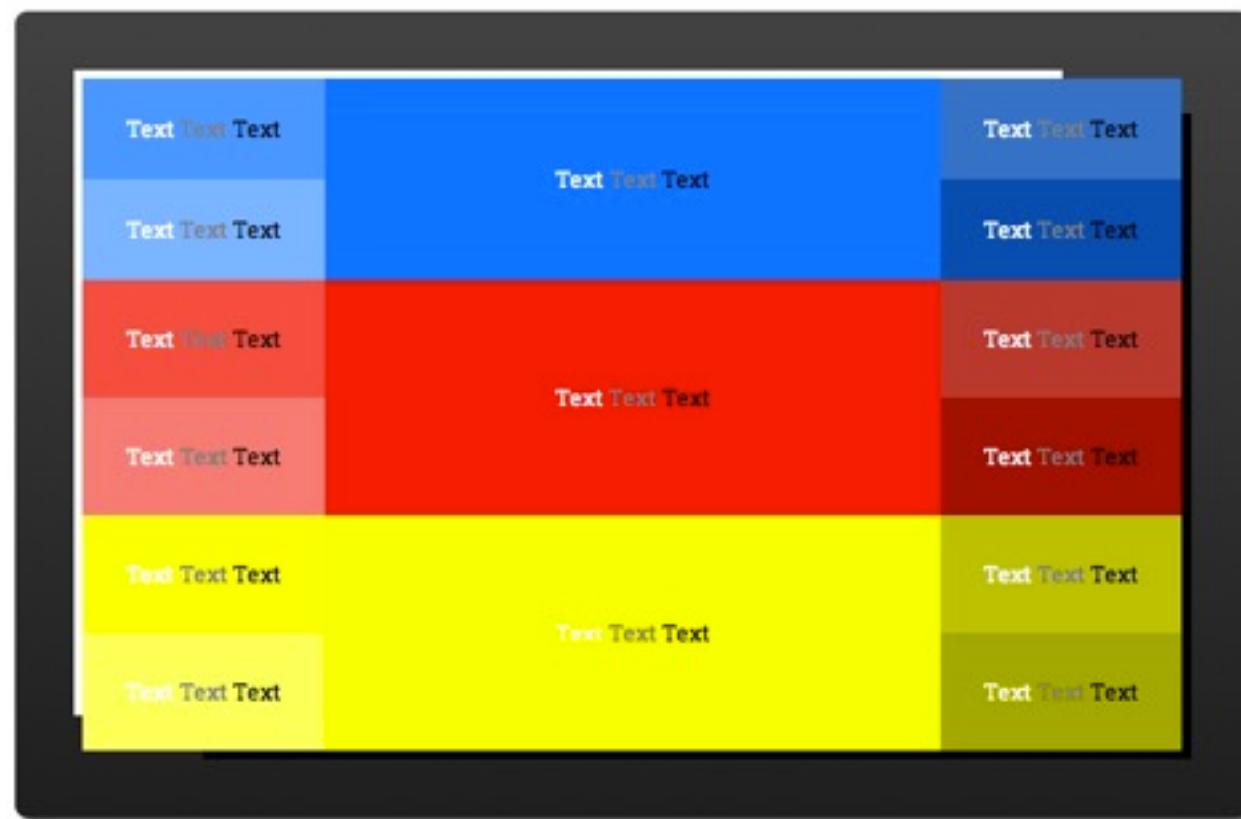


Triad

```
$base-color: hsl($hue, $saturation, $lighting);  
$angle: 30;  
$sec1-color: adjust-hue($base-color, (180 - $angle));  
$sec2-color: adjust-hue($base-color, (180 + $angle));  
  
.sec1-0 {  
    background: $sec1-color;  
}  
.sec1-1 {  
    background: adjust-color($sec1-color, $saturation: -40%);  
}  
.sec1-2 {  
    background: adjust-color($sec1-color, $lightness: 23%);  
}  
.sec1-3 {  
    background: adjust-color($sec1-color, $lightness: -17%);  
}  
....
```



Triad



```
$base-color: hsl($hue, $saturation, $lighting);  
$angle: 30;  
$sec1-color: adjust-hue($base-color, (180 - $angle));  
$sec2-color: adjust-hue($base-color, (180 + $angle));  
  
.sec1-0 {  
    background: $sec1-color;  
}  
.sec1-1 {  
    background: adjust-color($sec1-color, $saturation: -40%);  
}  
.sec1-2 {  
    background: adjust-color($sec1-color, $lightness: 23%);  
}  
.sec1-3 {  
    background: adjust-color($sec1-color, $lightness: -17%);  
}  
....
```

Tetrad

- This scheme is made by a pair of colors and their complements



Tetrad

- This scheme is made by a pair of colors and their complements



Tetrad

- This scheme is made by a pair of colors and their complements

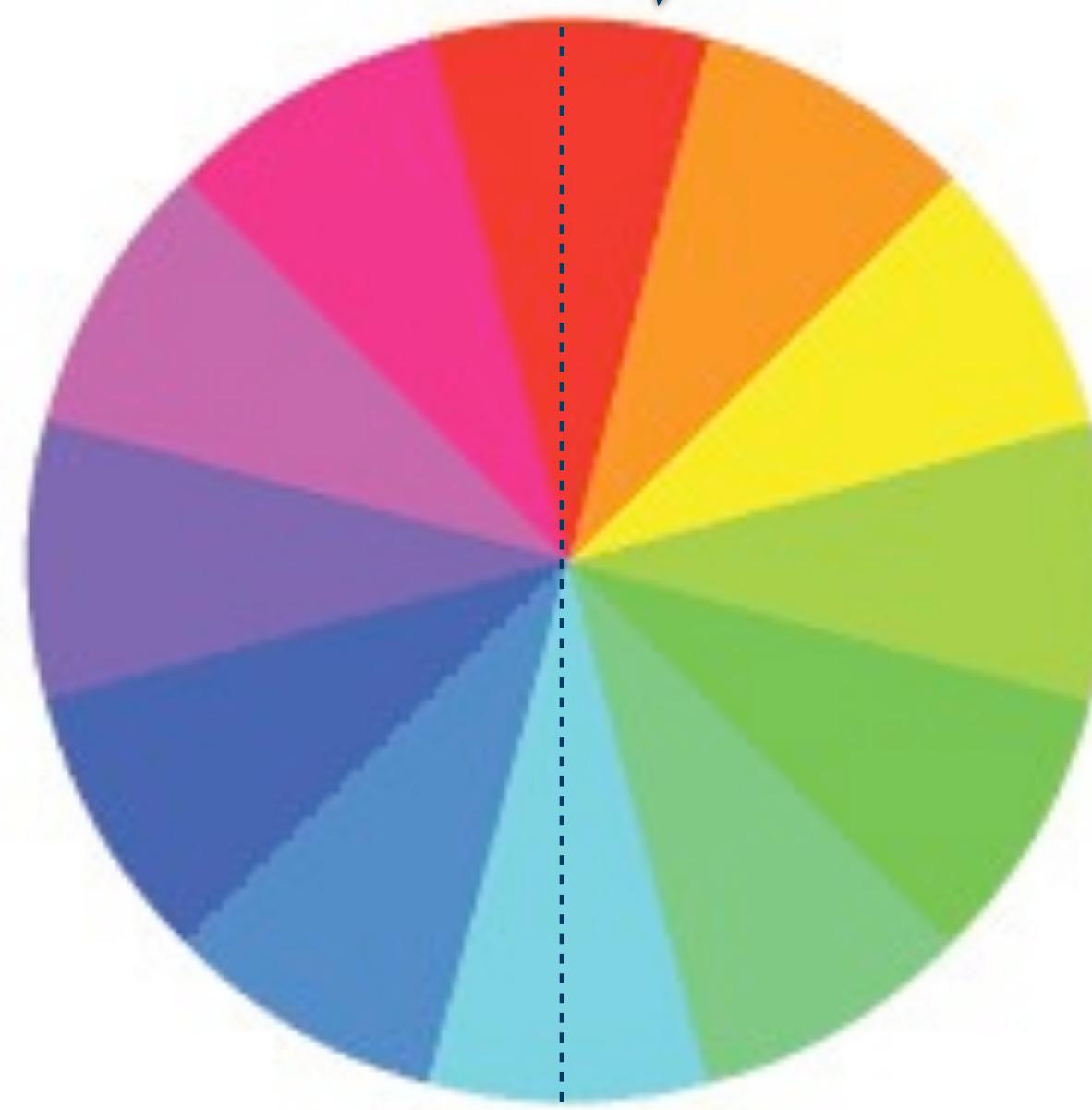
Base color - hsl(0, 100%, 50%)



Tetrad

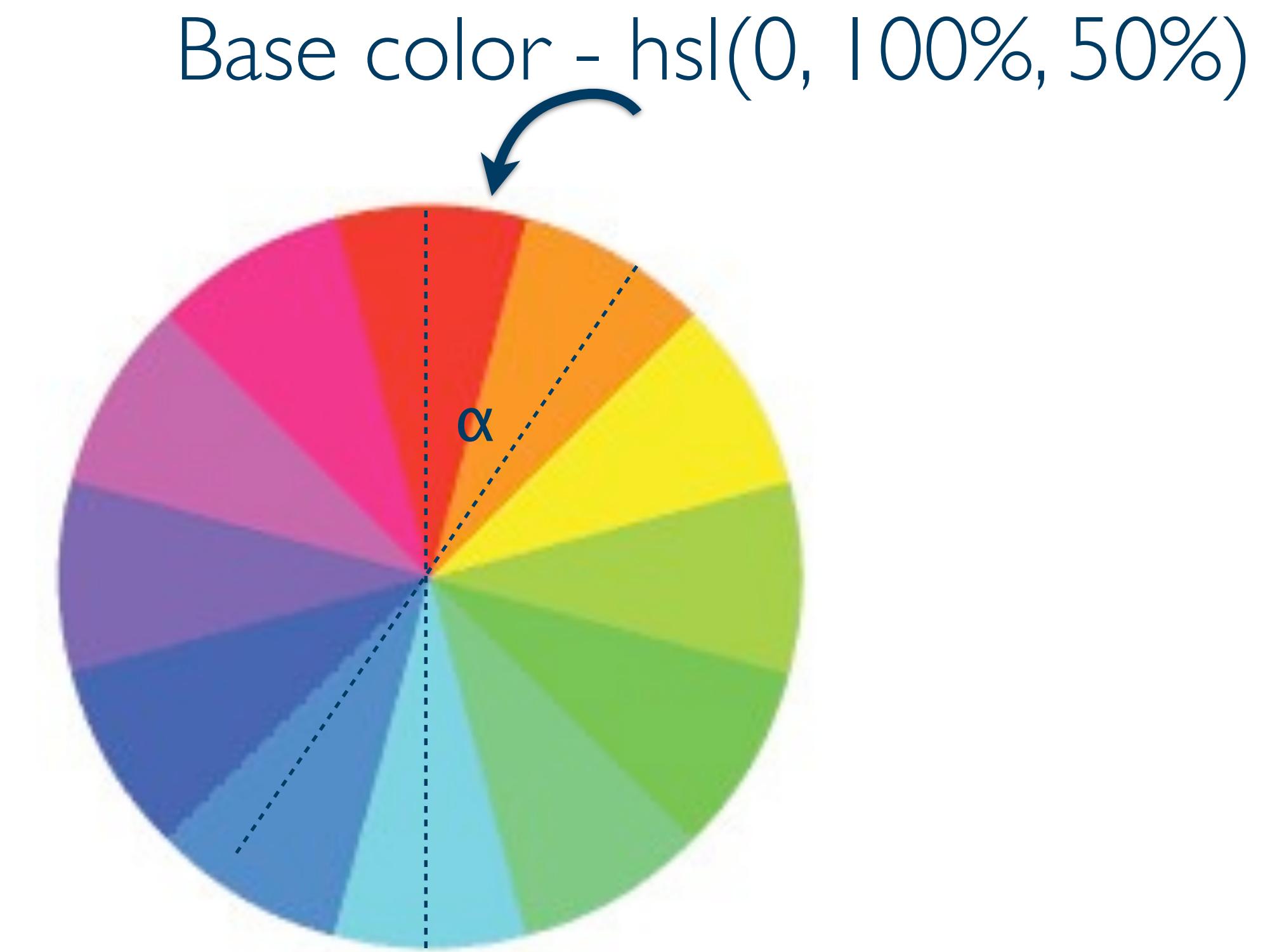
- This scheme is made by a pair of colors and their complements

Base color - hsl(0, 100%, 50%)



Tetrad

- This scheme is made by a pair of colors and their complements



Tetrad

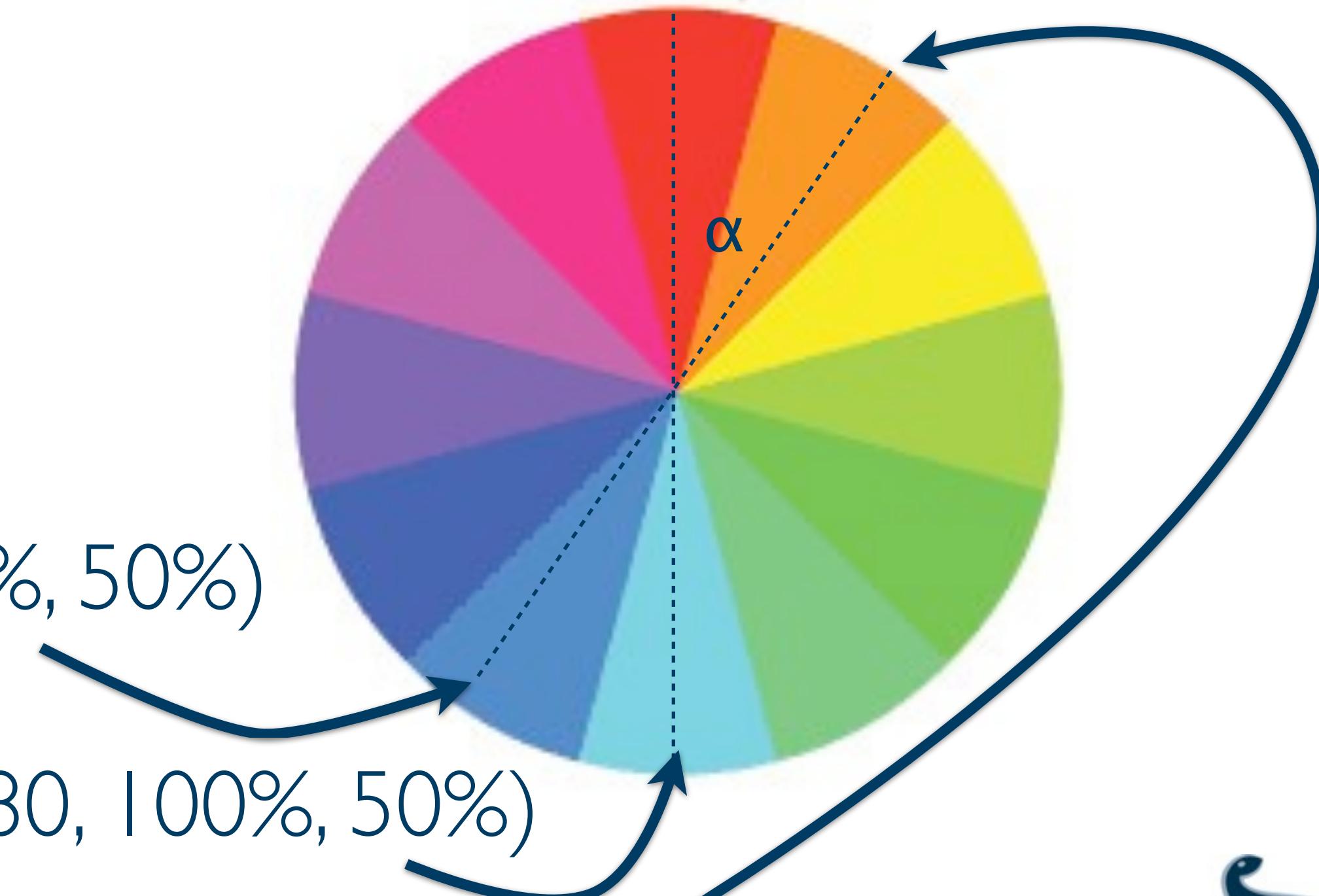
- This scheme is made by a pair of colors and their complements

Secondary I - hsl(210, 100%, 50%)

Complement - hsl(180, 100%, 50%)

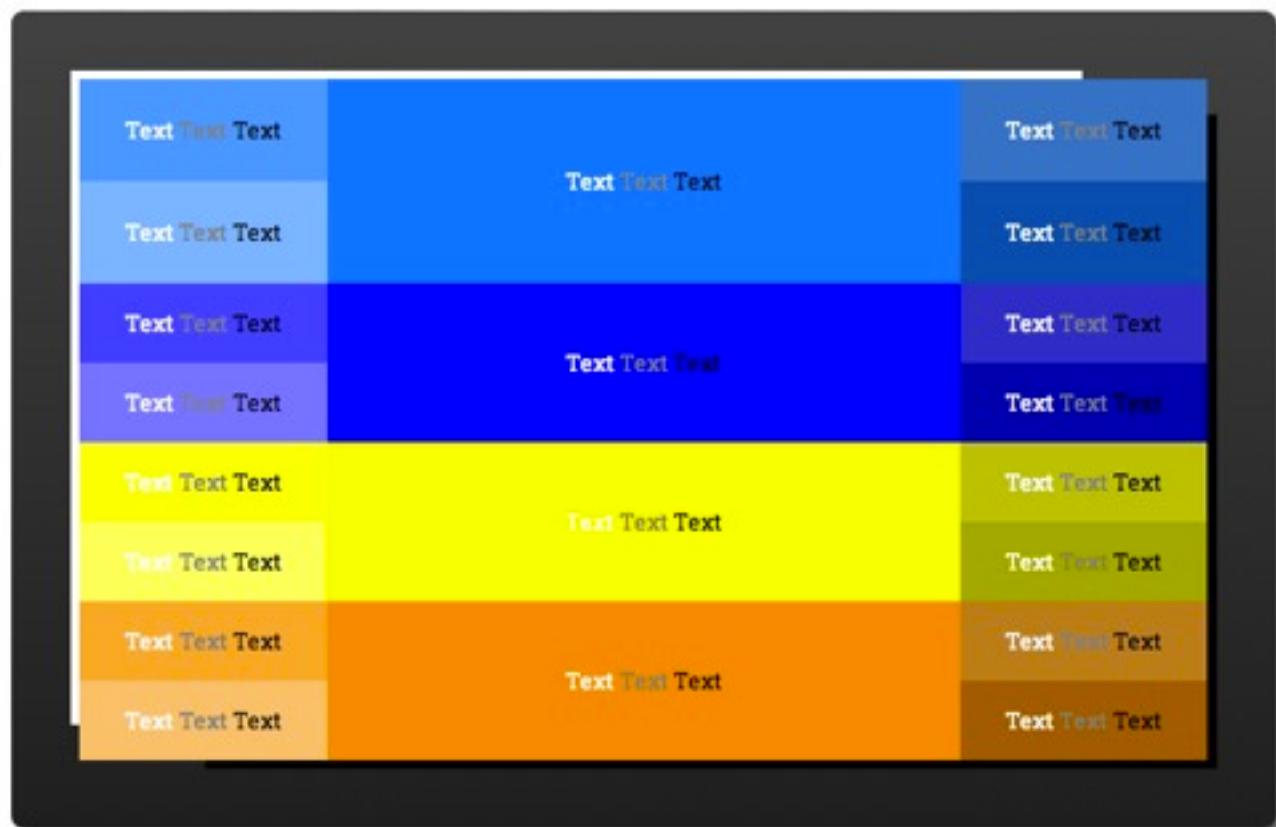
Secondary I - hsl(30, 100%, 50%)

Base color - hsl(0, 100%, 50%)



Tetrad

```
$base-color: hsl($hue, $saturation, $lighting);  
$compl-color: complement($base-color);  
$angle: 30;  
$sec1-color: adjust-hue($base-color, $angle);  
$sec2-color: adjust-hue($compl-color, $angle);  
  
.sec1-0 {  
    background: $sec1-color;  
}  
.sec1-1 {  
    background: adjust-color($sec1-color, $saturation: -40%);  
}  
.sec1-2 {  
    background: adjust-color($sec1-color, $lightness: 23%);  
}  
.sec1-3 {  
    background: adjust-color($sec1-color, $lightness: -17%);  
}  
...  
...
```



Analogic

- This scheme is made by the primary color and its adjacent colors – two colors identically on both sides



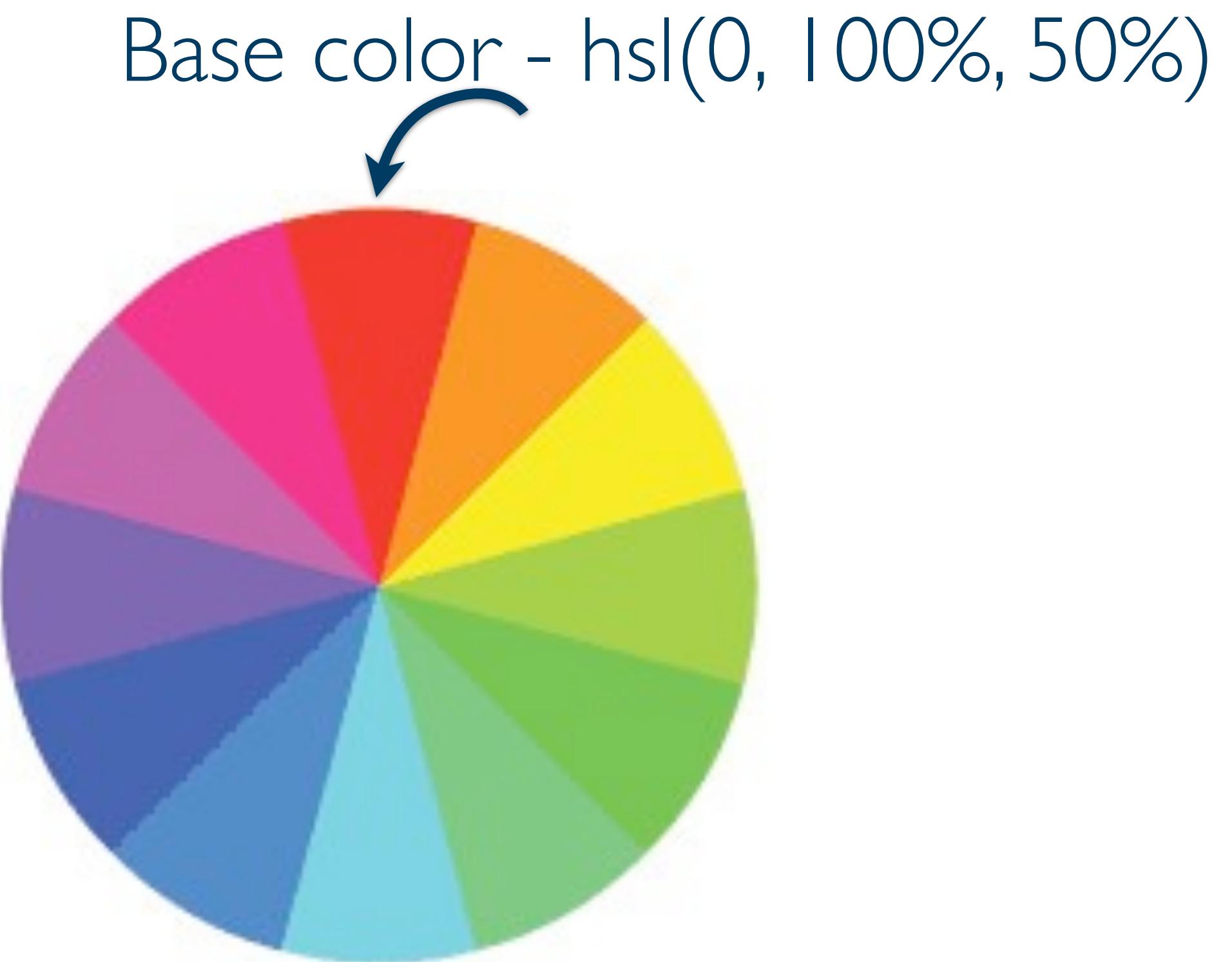
Analogic

- This scheme is made by the primary color and its adjacent colors – two colors identically on both sides



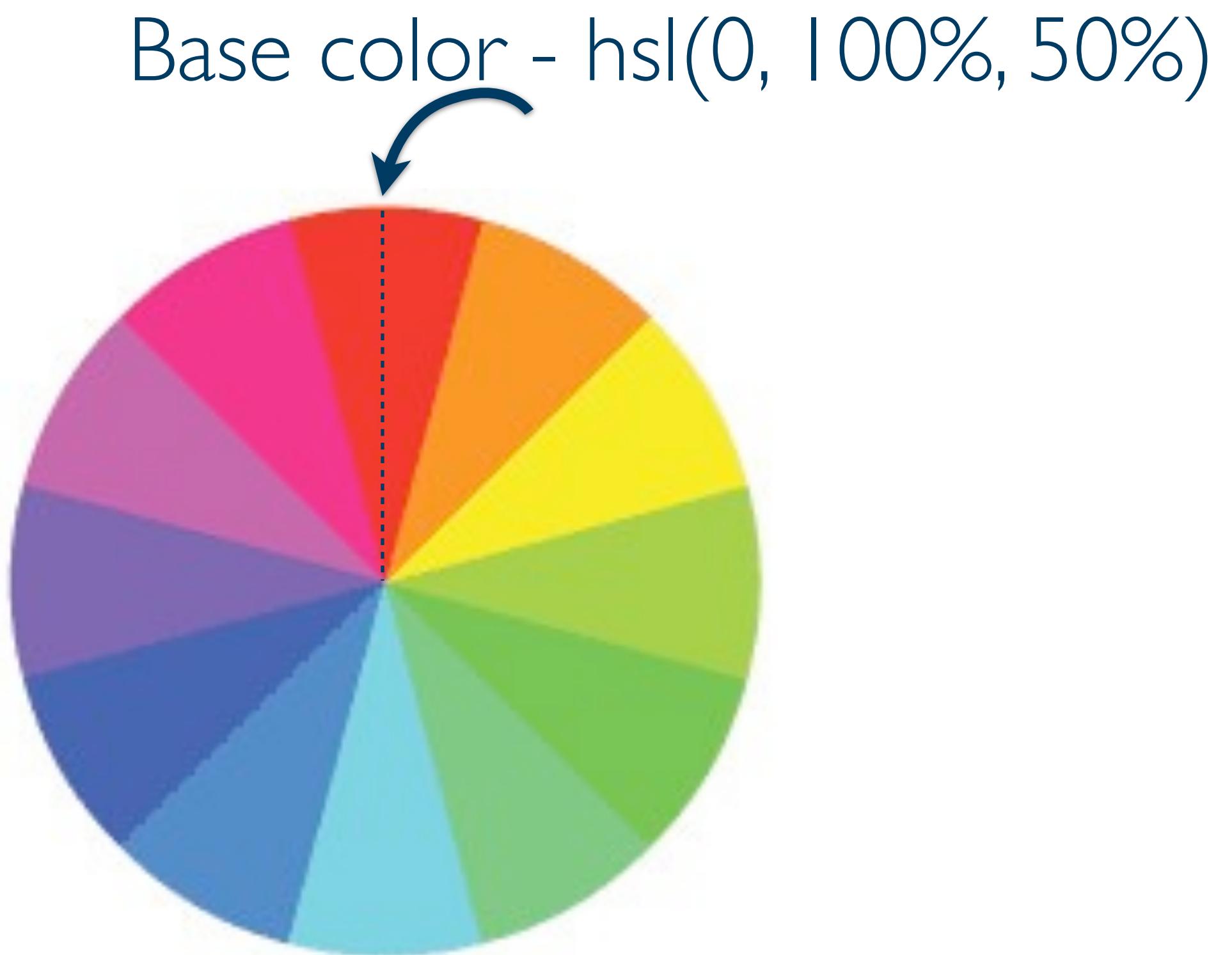
Analogic

- This scheme is made by the primary color and its adjacent colors – two colors identically on both sides



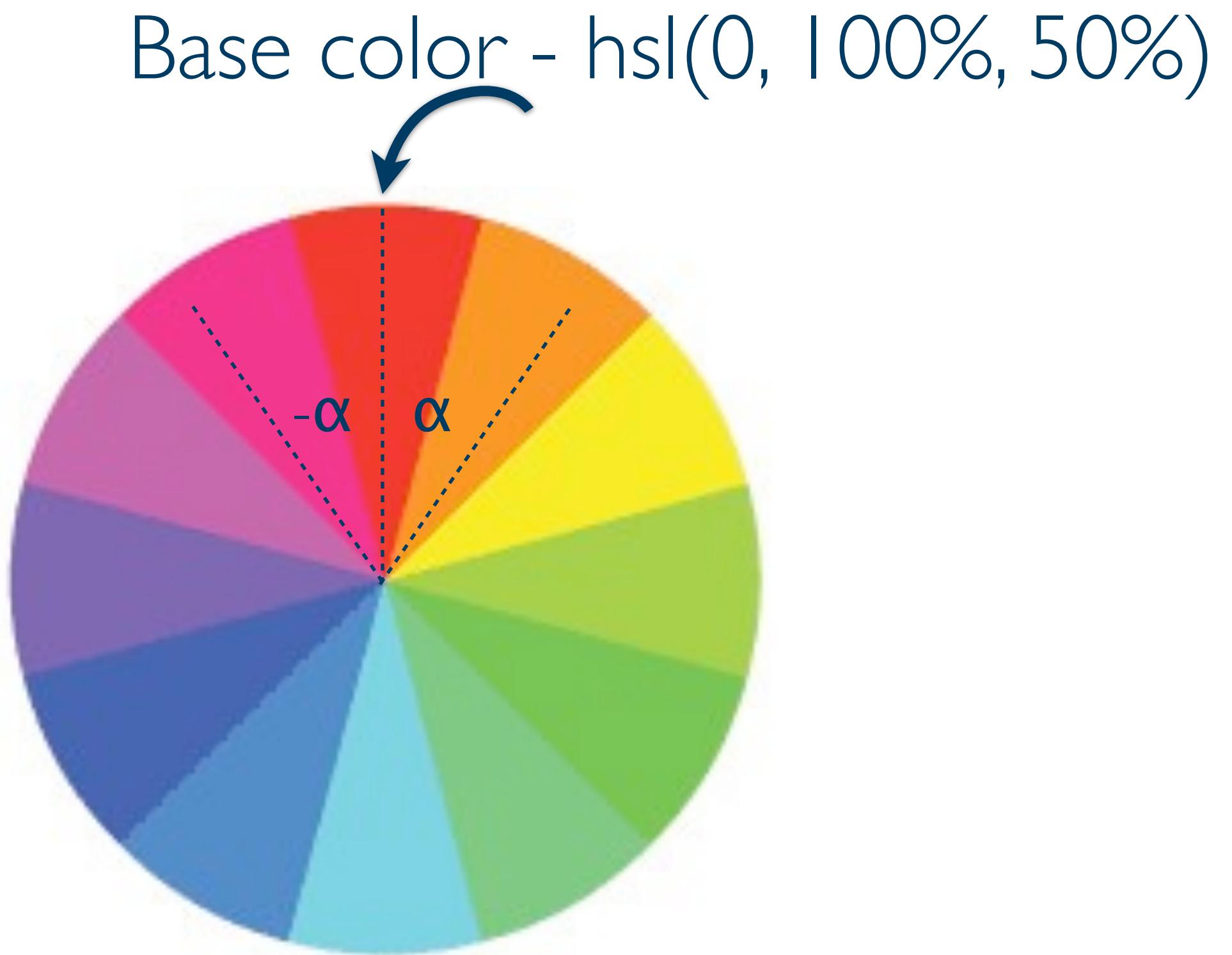
Analogic

- This scheme is made by the primary color and its adjacent colors – two colors identically on both sides



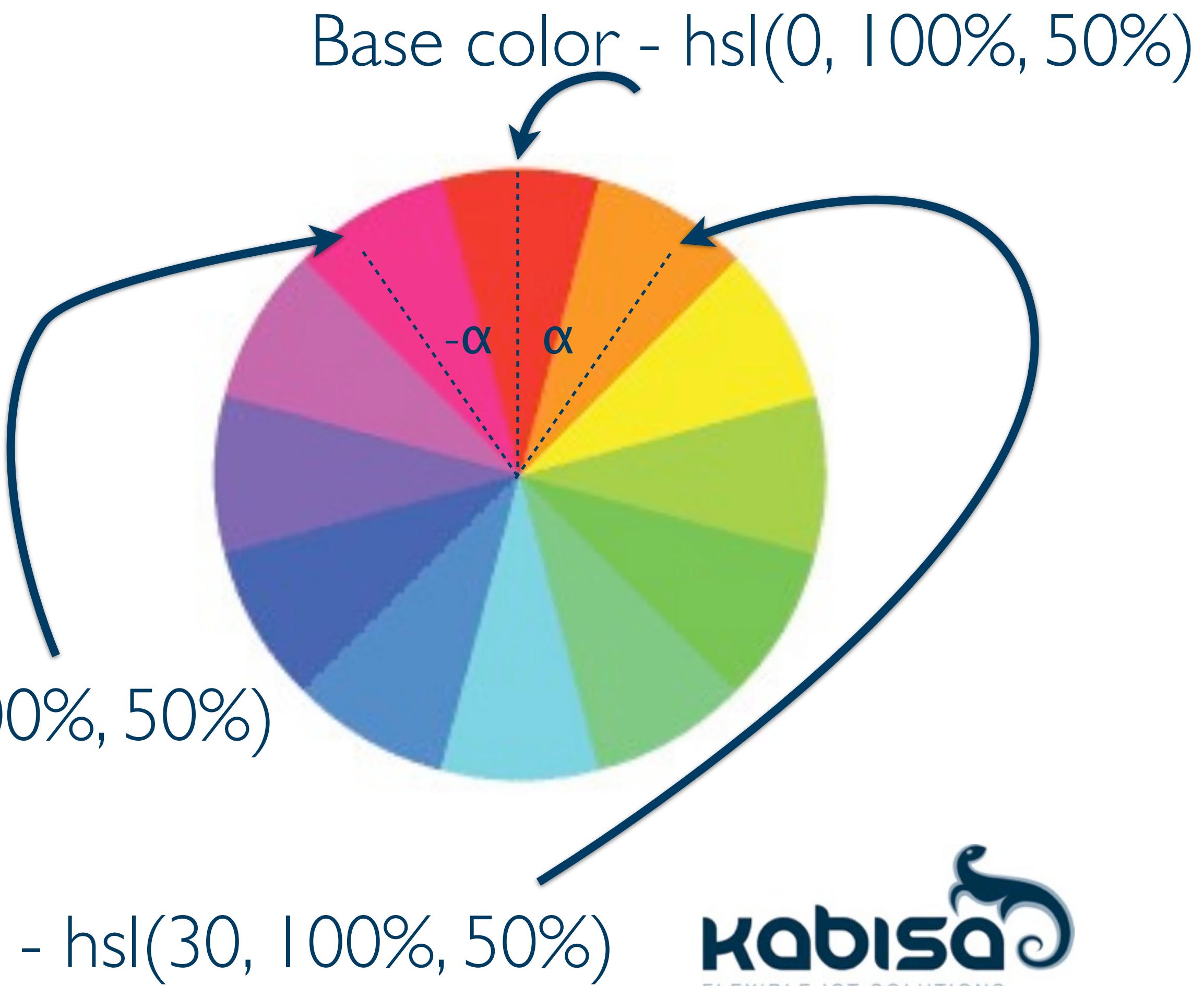
Analogic

- This scheme is made by the primary color and its adjacent colors – two colors identically on both sides



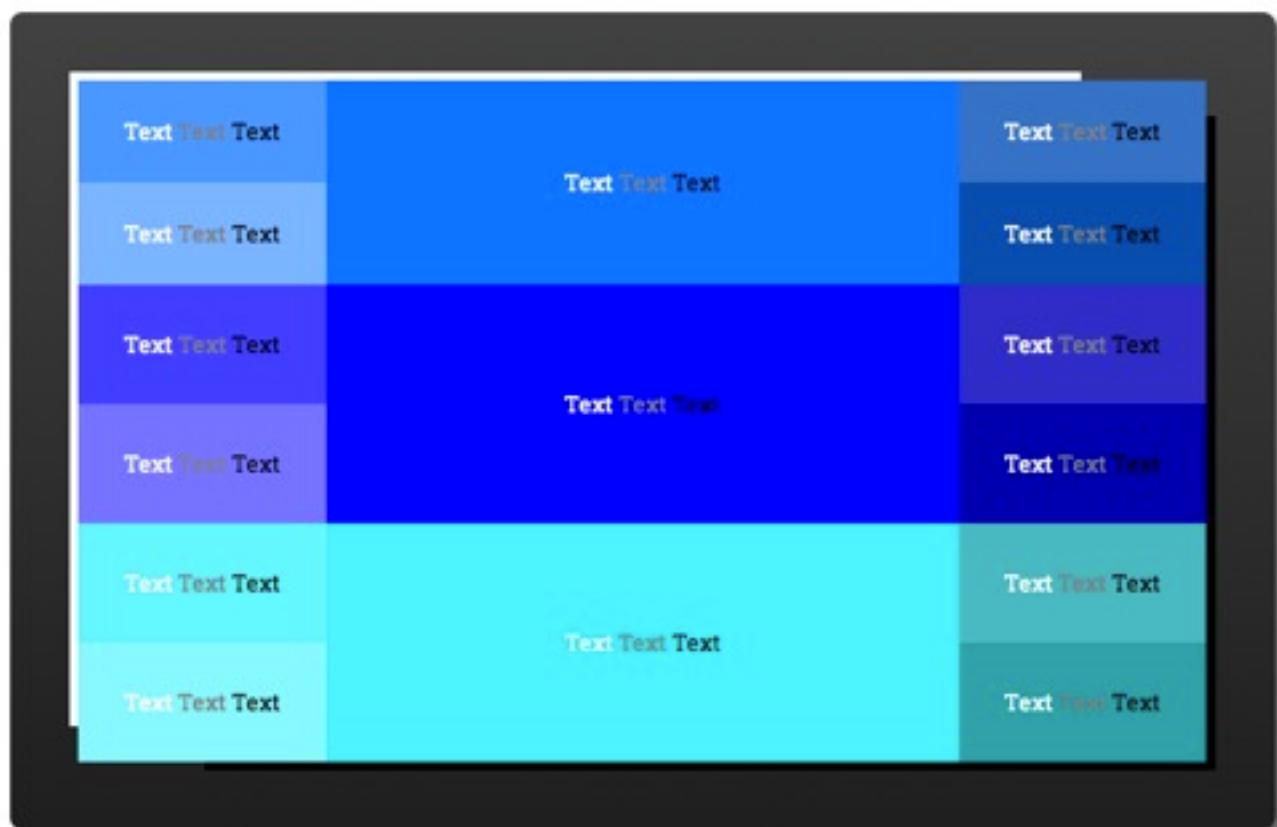
Analogic

- This scheme is made by the primary color and its adjacent colors – two colors identically on both sides



Tetrad

```
$base-color: hsl($hue, $saturation, $lighting);  
$angle: 30;  
$sec1-color: adjust-hue($base-color, $angle);  
$sec2-color: adjust-hue($base-color, (360-$angle));  
  
.sec1-0 {  
    background: $sec1-color;  
}  
.sec1-1 {  
    background: adjust-color($sec1-color, $saturation: -40%);  
}  
.sec1-2 {  
    background: adjust-color($sec1-color, $lightness: 23%);  
}  
.sec1-3 {  
    background: adjust-color($sec1-color, $lightness: -17%);  
}  
...  
...
```



Accented Analogic

- This is the Analogic model with complementary (contrast) color added



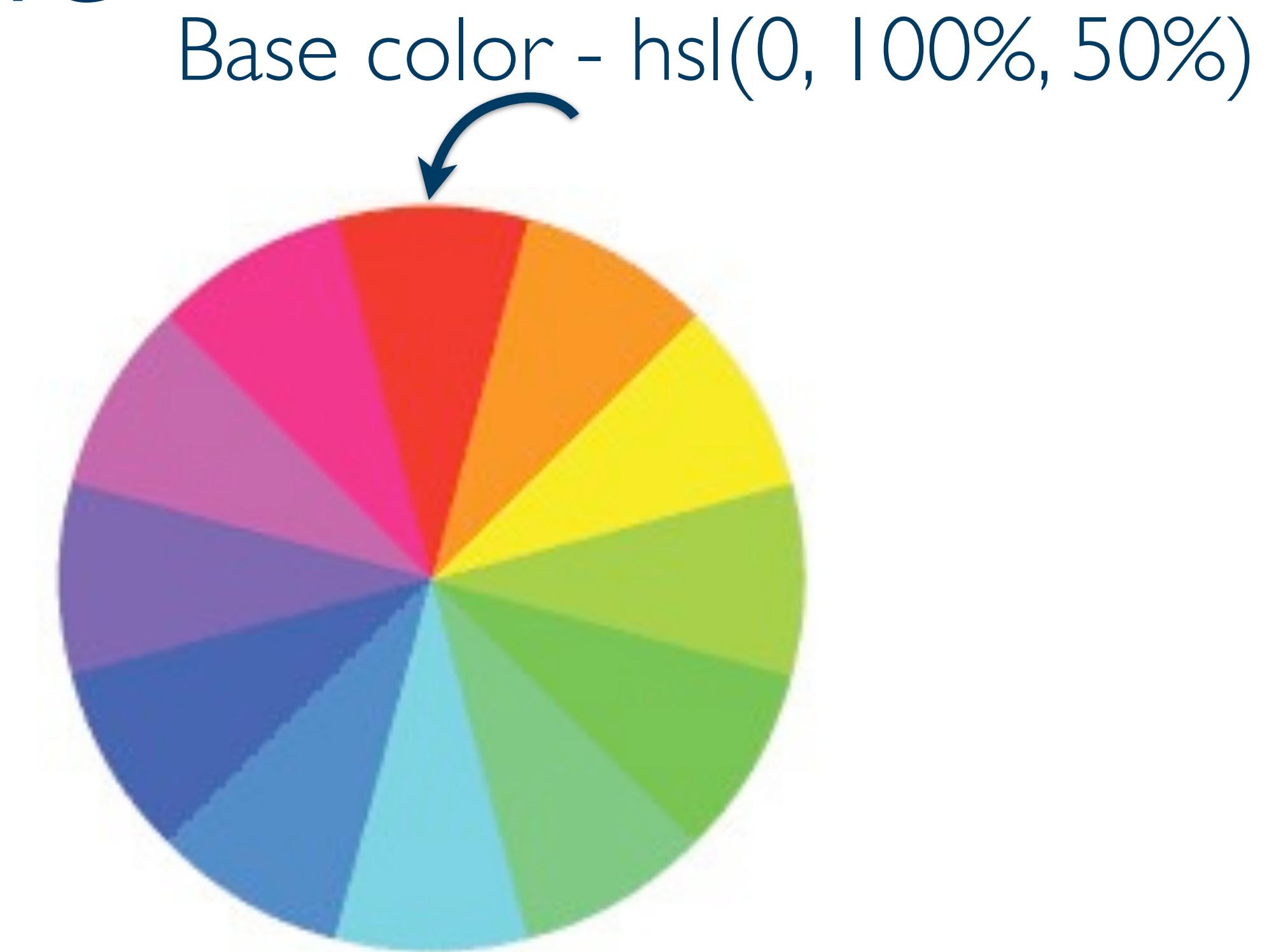
Accented Analogic

- This is the Analogic model with complementary (contrast) color added



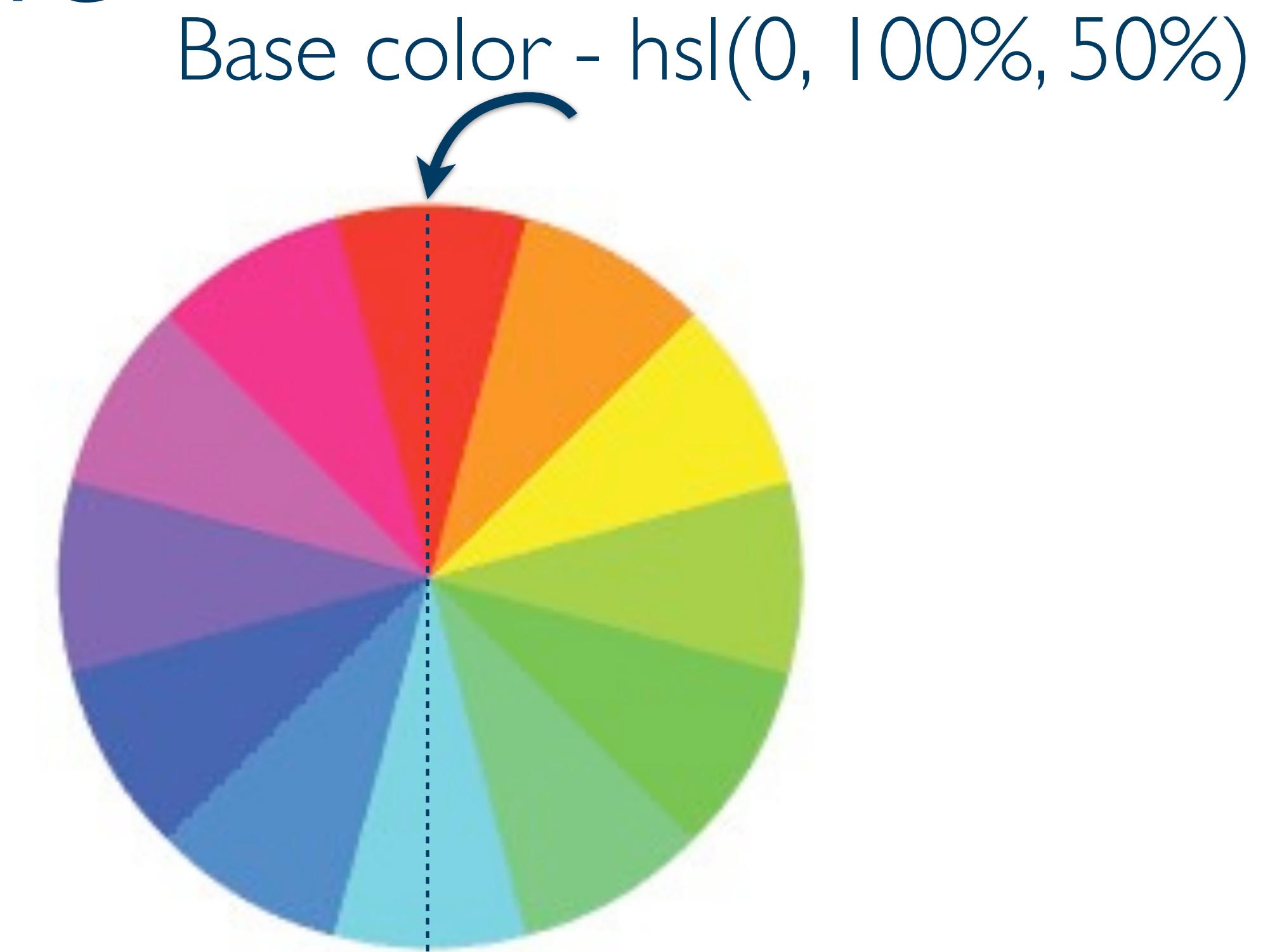
Accented Analogic

- This is the Analogic model with complementary (contrast) color added



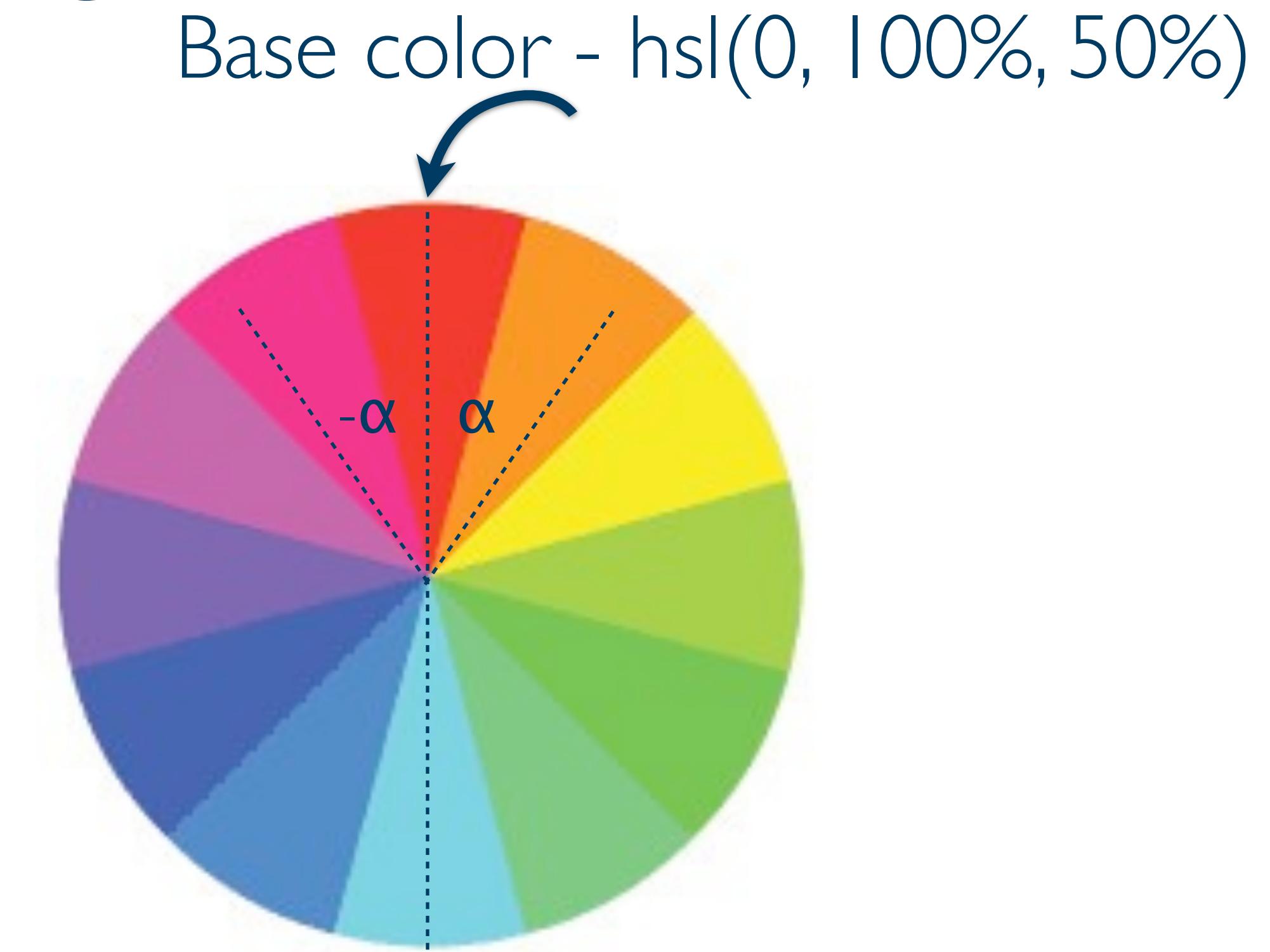
Accented Analogic

- This is the Analogic model with complementary (contrast) color added



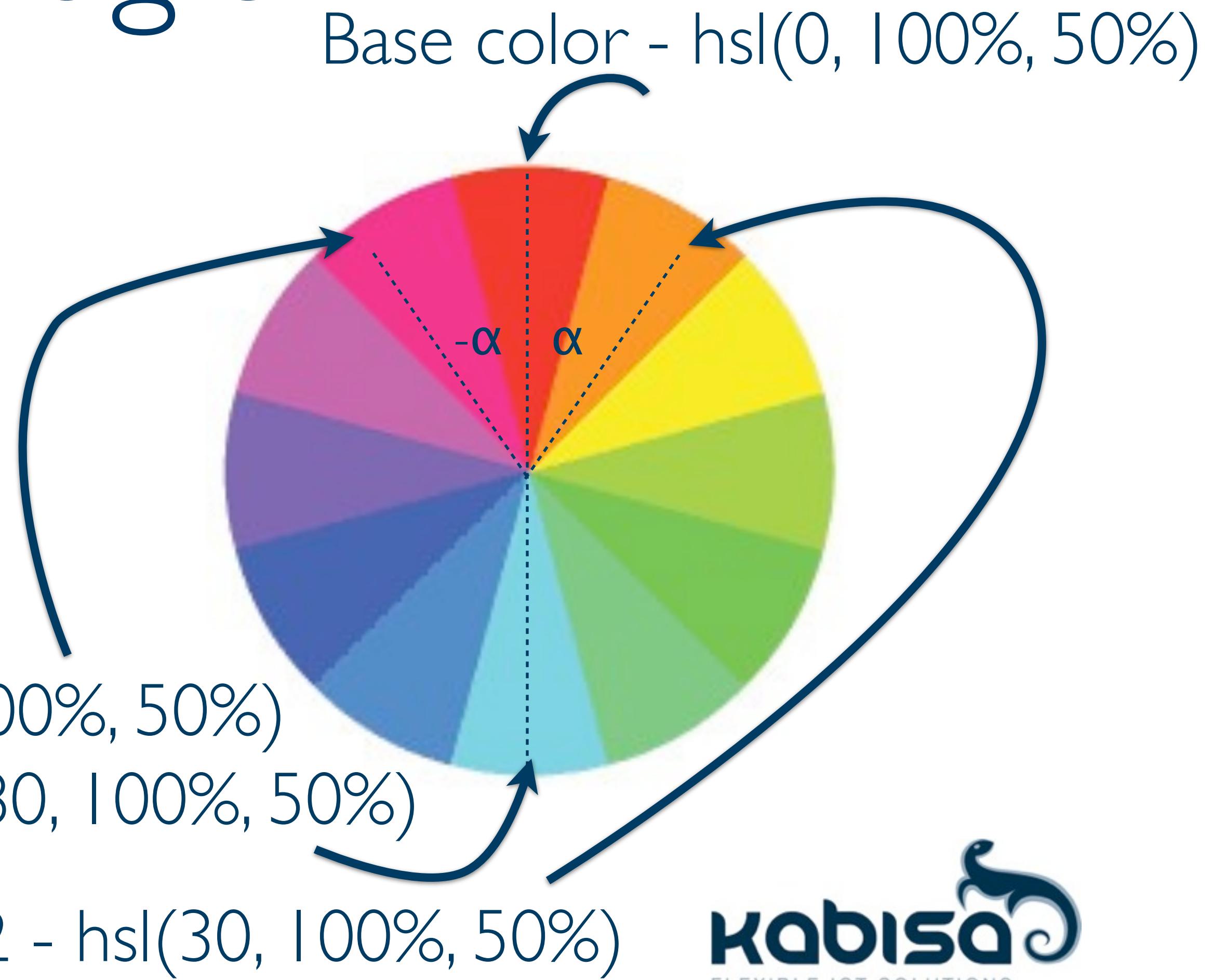
Accented Analogic

- This is the Analogic model with complementary (contrast) color added



Accented Analogic

- This is the Analogic model with complementary (contrast) color added



Accented Analogic

```
$base-color: hsl($hue, $saturation, $lighting);  
$compl-color: complement($base-color);  
$angle: 30;  
$sec1-color: adjust-hue($base-color, $angle);  
$sec2-color: adjust-hue($base-color, (360-$angle));  
  
.sec1-0 {  
    background: $sec1-color;  
}  
.sec1-1 {  
    background: adjust-color($sec1-color, $saturation: -40%);  
}  
.sec1-2 {  
    background: adjust-color($sec1-color, $lightness: 23%);  
}  
.sec1-3 {  
    background: adjust-color($sec1-color, $lightness: -17%);  
}  
...
```



Sample project



Sample project

- <https://github.com/bazzel/color-schemes-scss>



Tooling



Tooling

- Adobe Kuler - <https://kuler.adobe.com>



Tooling

- Adobe Kuler - <https://kuler.adobe.com>
- Color Scheme Designer - <http://colorschemedesigner.com/>



Tooling

- Adobe Kuler - <https://kuler.adobe.com>
- Color Scheme Designer - <http://colorschemedesigner.com/>
- Saturation - <http://saturation.bitgun.com/>



Tooling

- Adobe Kuler - <https://kuler.adobe.com>
- Color Scheme Designer - <http://colorschemedesigner.com/>
- Saturation - <http://saturation.bitgun.com/>
- HSL Color Picker - <http://hslpicker.com>



Tooling

- Adobe Kuler - <https://kuler.adobe.com>
- Color Scheme Designer - <http://colorschemedesigner.com/>
- Saturation - <http://saturation.bitgun.com/>
- HSL Color Picker - <http://hslpicker.com>
- Color Schemer - <https://github.com/Team-Sass/color-schemer>



Tooling

- Adobe Kuler - <https://kuler.adobe.com>
- Color Scheme Designer - <http://colorschemedesigner.com/>
- Saturation - <http://saturation.bitgun.com/>
- HSL Color Picker - <http://hslpicker.com>
- Color Schemer - <https://github.com/Team-Sass/color-schemer>
- Chrome DevTools - 



Sources



Sources

- Powerful Color Manipulation with Sass

<http://nex-3.com/posts/89-powerful-color-manipulation-with-sass>



Sources

- Powerful Color Manipulation with Sass

<http://nex-3.com/posts/89-powerful-color-manipulation-with-sass>

- Causes of Colors

<http://www.webexhibits.org/causesofcolor>

