

Pygame - Dev Notes - Colour

A brief intro to using colour with Python's Pygame module.

Contents

- Intro
- Mixing colours

Intro

There is a clearly defined pattern we may use to work with colours in Pygame.

Each colour is defined using the standard RGB primary colours, which include

- *Red*
- *Green*
- *Blue*.

We may then start to create our secondary colours as variants and combinations of these three primary colours. e.g.

- cyan = blue + green
- magenta = blue + red
- yellow = green + red

We may also create base colours for *black* and *white*. e.g.

- black = no colours
- white = red + green + blue

Mixing colours

To create a particular colour, we're effectively defining how much of each primary colour we require mixing.

This mixing uses a known scale from **0** to **255** for each primary colour, therefore giving a possible 256 points per colour on the scale.

We may define each of the above colours, for example, as follows

- red = `rgb(255, 0, 0)`
- green = `rgb(0, 255, 0)`
- blue = `rgb(0, 0, 255)`
- cyan = `rgb(0, 255, 255)`
- magenta = `rgb(255, 0, 255)`
- yellow = `rgb(255, 255, 0)`
- black = `rgb(0, 0, 0)`
- white = `rgb(255, 255, 255)`

If we consider the sheer number of colour options for this scale, we end up with

- $256 * 256 * 256 = 16,777,216$

So, we have over 16 million possible colour variations for our game's design and rendering.