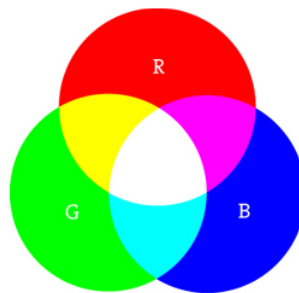


1 Cor

Temos também um suporte incrível a coloração em processing, atendendo a gostos diferentes de modos de trabalho. Para entender RGB Color basta se lembrar de quando você estava no primário e a tia Cássia dava como tarefa misturar as cores com os dedos e ver no que dava.

Diagrama de Venn:



- Vermelho + Verde = Amarelo
- Vermelho + Azul = Rosa
- Verde + Azul = Azul Esverdeado
- Vermelho + Verde + Azul = Branco
- Sem cores = Preto

1.1 Estudo da coloração

Cor:

- `background()`

- background(VALOR, VALOR, VALOR, ALPHA)
- colorMode()
 - colorMode(MODO, COMPRIMENTO, COMPRIMENTO, COMPRIMENTO);
- stroke()
 - stroke(VALOR, VALOR, VALOR, ALPHA)
- fill()
 - fill(VALOR, VALOR, VALOR, ALPHA)
- tint()
 - tint(VALOR, VALOR, VALOR, ALPHA)
- noFill()
- noStroke()
- noTint()

1.2 Exercício 1

```
noStroke();
colorMode(HSB, 100);
for (int i = 0; i < 100; i++) {
  for (int j = 0; j < 100; j++) {
    stroke(i, j, 100);
    point(i, j);
  }
}
```

1.3 Exercício 2

```
noStroke();
colorMode(RGB, 100);
for (int i = 0; i < 100; i++) {
```

```

    for (int j = 0; j < 100; j++) {
        stroke(i, j, 0);
        point(i, j);
    }
}

```

1.4 Exercício 3

Nesse exercício e no código abaixo foi feito um exemplo de Saturação misturando um pouco o que aprendemos no primeiro em **oficina-processing-00**. A A Saturação é o quanto a cor é intensa ou não e representa a quantidade de escala cincenta. Ou seja, uma cor saturada diz-se uma cor pura e uma cor insaturada tem grau de escala cinzenta (porcentagem). O que é bastante interessante trabalhar.

```

int barWidth = 5;
int lastBar = -1;

void setup() {
    size(200, 200);
    colorMode(HSB, width, height, 100);
    noStroke();
}

void draw() {
    int whichBar = mouseX / barWidth;
    if (whichBar != lastBar) {
        int barX = whichBar * barWidth;
        fill(barX, mouseY, 66);
        rect(barX, 0, barWidth, height);
        lastBar = whichBar;
    }
}

```

1.5 Exercício 4

```
PImage b;  
b = loadImage("tarrafa.png");  
image(b, 0, 0);  
tint(0, 153, 204);  
image(b, 50, 0);
```