

Prazer, Processing

...

Programação Criativa - Aula 2
01/10/2015

Conteúdo

- Introdução do Processing
- Projetos inspiradores
- Introdução ao PDE
- Canvas
- Elementos gráficos básicos
- Cores
- Salvando sua criação

2001
Ben Fry
Casey Reas



processing.org

Processing

[Cover](#)[Download](#)[Exhibition](#)[Reference](#)[Libraries](#)[Tools](#)[Environment](#)[Tutorials](#)[Examples](#)[Books](#)[Handbook](#)[Overview](#)[People](#)[Shop](#)[» Forum](#)[» GitHub](#)[» Issues](#)[» Wiki](#)[» FAQ](#)[» Twitter](#)[» Facebook](#)

Welcome to Processing 3! Dan explains the new features and changes; the links Dan mentions are on the [Vimeo page](#).

[» Download Processing](#)[» Browse Tutorials](#)[» Visit the Reference](#)

Processing is a flexible software sketchbook and a language for learning how to code within the context of the visual arts. Since 2001, Processing has promoted software literacy within the visual arts and visual literacy within technology. There are tens of thousands of students, artists, designers, researchers, and hobbyists who use Processing for learning and prototyping.

» Free to download and open source

[» Exhibition](#)[Large Napkin](#)[by Pixtil](#)[Light Kinetics](#)[by Espadaysantacruz](#)[Pathfinder](#)[by Princemio](#)

Projetos Inspiradores

Instalações Interativas

My Little Piece of Privacy



Sound Machines



Delicate Boundaries

Light Kinetics

source: <http://espadaysantacruz.com/Light-kinetics>

Projetos Gráficos



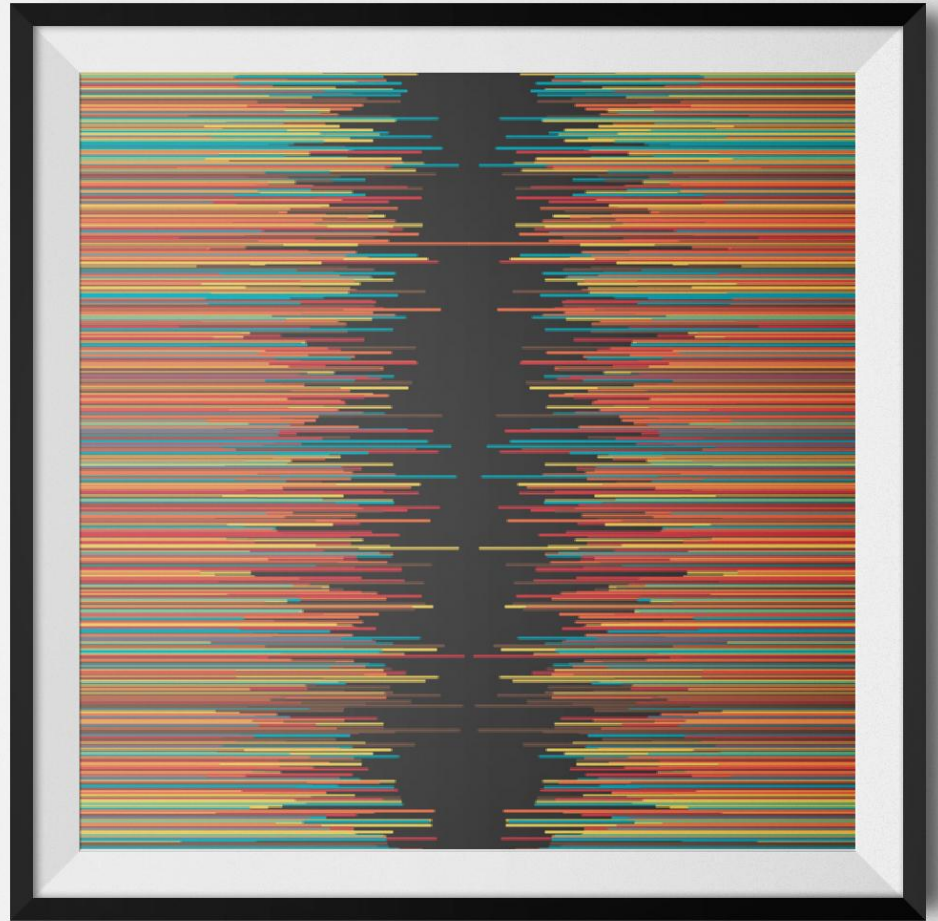
MediaLab Identity





catarses metafóricas

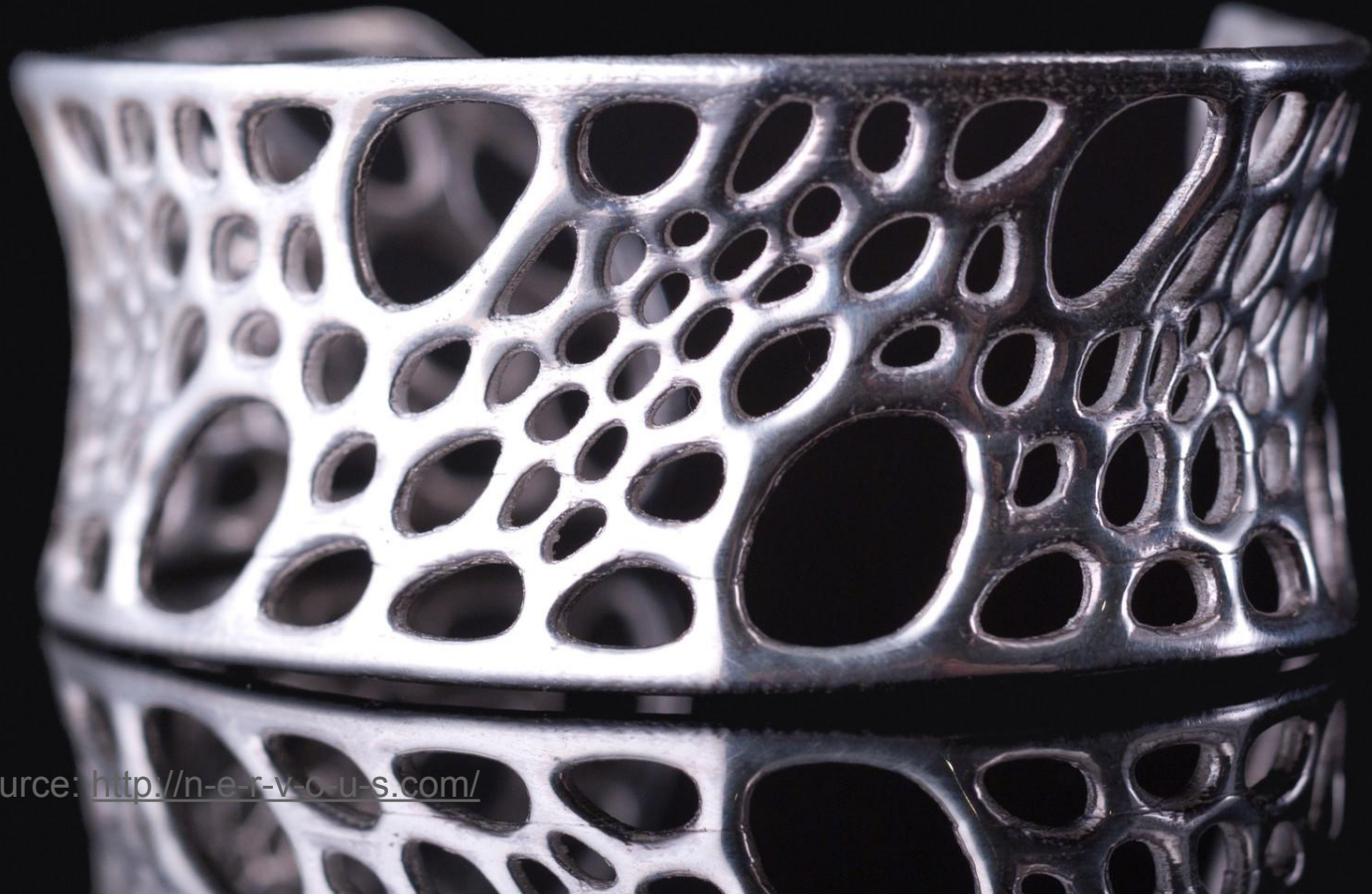
code2pixels



source: <http://code2pixels.tumblr.com>

Aplicações 3D

Cell Cycle



source: <http://n-e-r-v-o-u-s.com/>

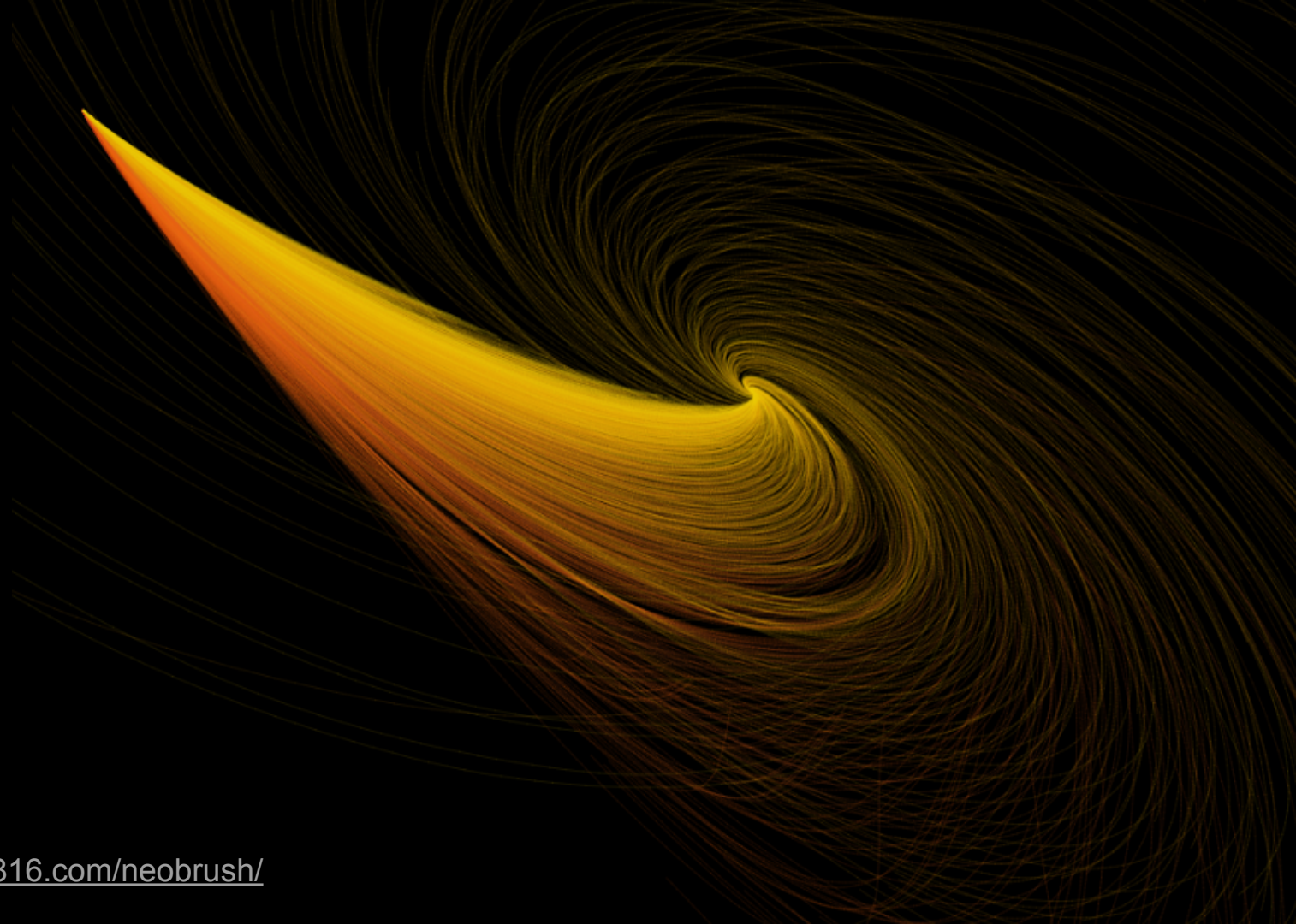
Digital Natives



source: <http://www.plummerfernandez.com/Digital-Nativesv>

Web (p5.js)

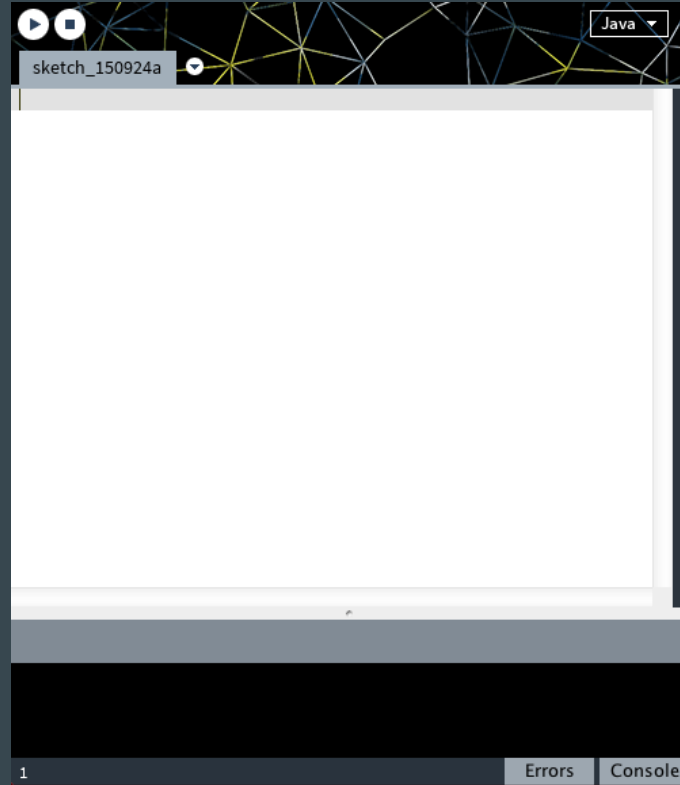
NeoBrush



source: <http://fun.the816.com/neobrush/>

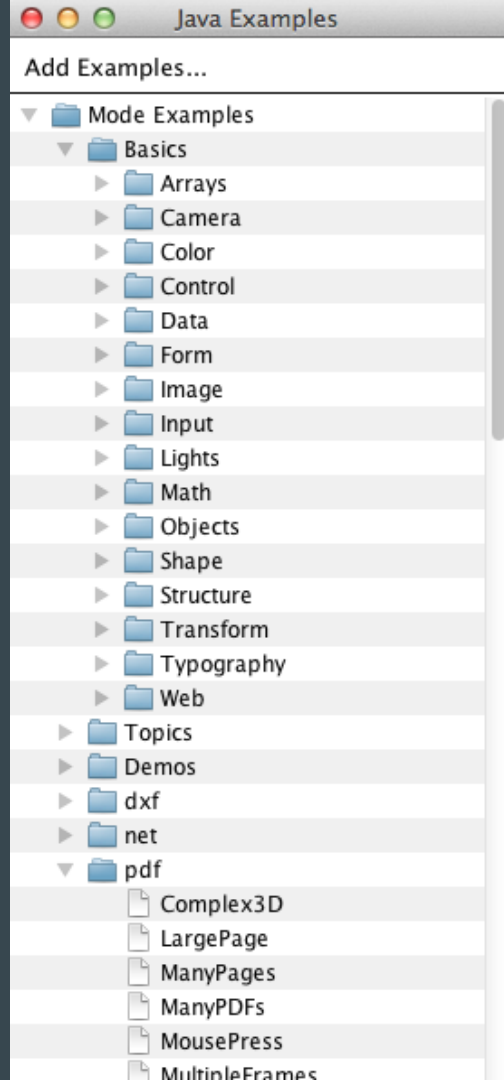
Introdução ao PDE

Processing Development Environment



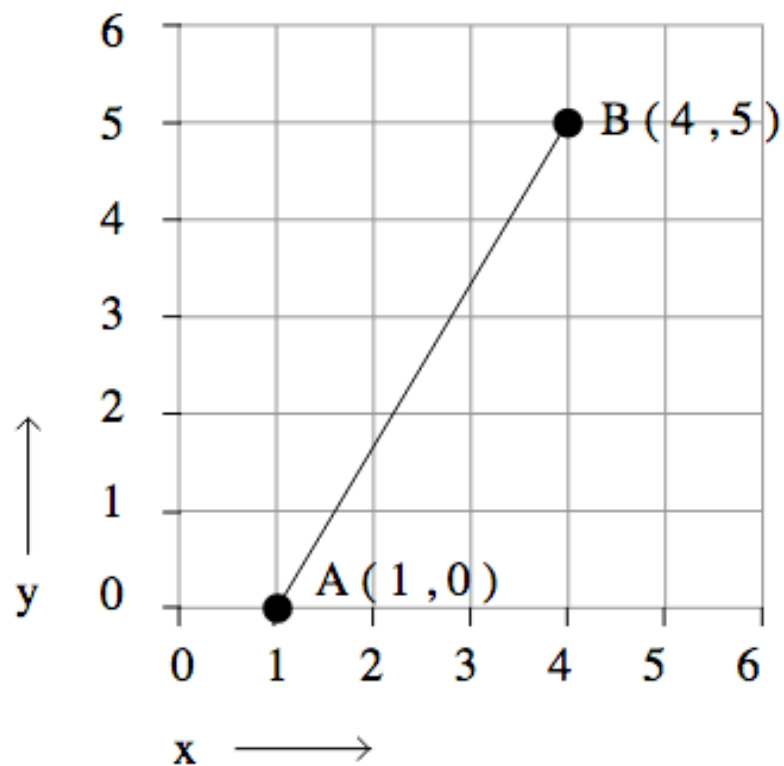
Exemplos

(Arquivo -> Exemplos)

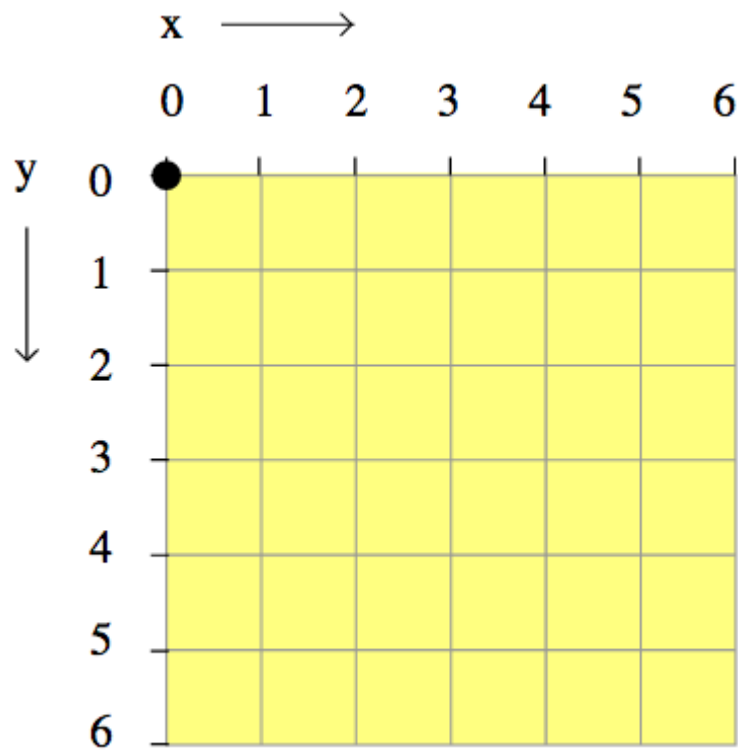


Canvas

Plano cartesiano



Canvas



Elementos gráficos básicos

Ponto

Desenhando um ponto

Comando:

point(x,y);

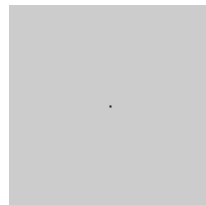
x -> Coordenada do eixo x

y -> Coordenada do eixo y

Exercício 1

Desenhe um ponto no centro do seu canvas

Resultado:



Como alterar o tamanho do canvas

Comandos iniciais para alterar o tamanho do canvas.

```
size(width, height);
```

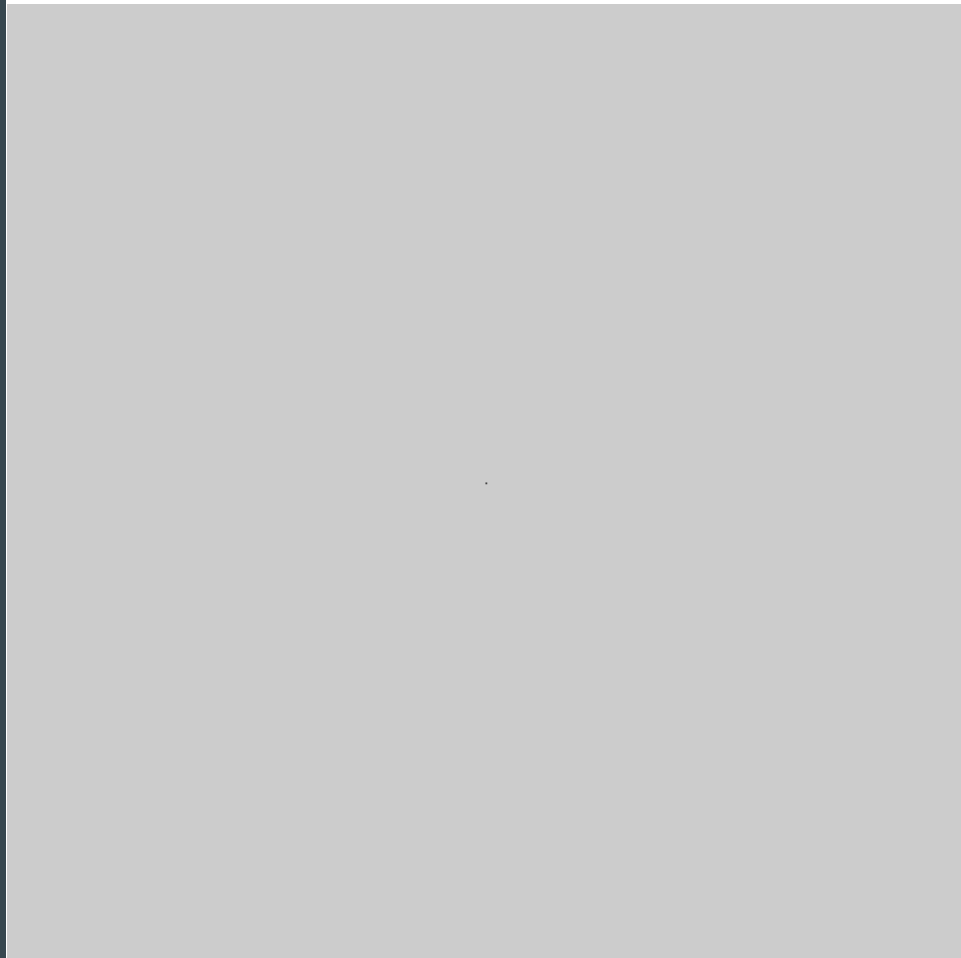
width/height -> largura/altura em pixels

Exemplo:

```
size(600,600);
```

Exercício 2

Desenhe um ponto no centro do seu canvas, agora o canvas deve ter tamanho 600x600px



Linha

Desenhando uma linha/reta
(menor distância entre dois
pontos)

Comando:

line(x1,y1, x2,y2);

Primeiro ponto:

x1 -> Coordenada do eixo x

y1 -> Coordenada do eixo y

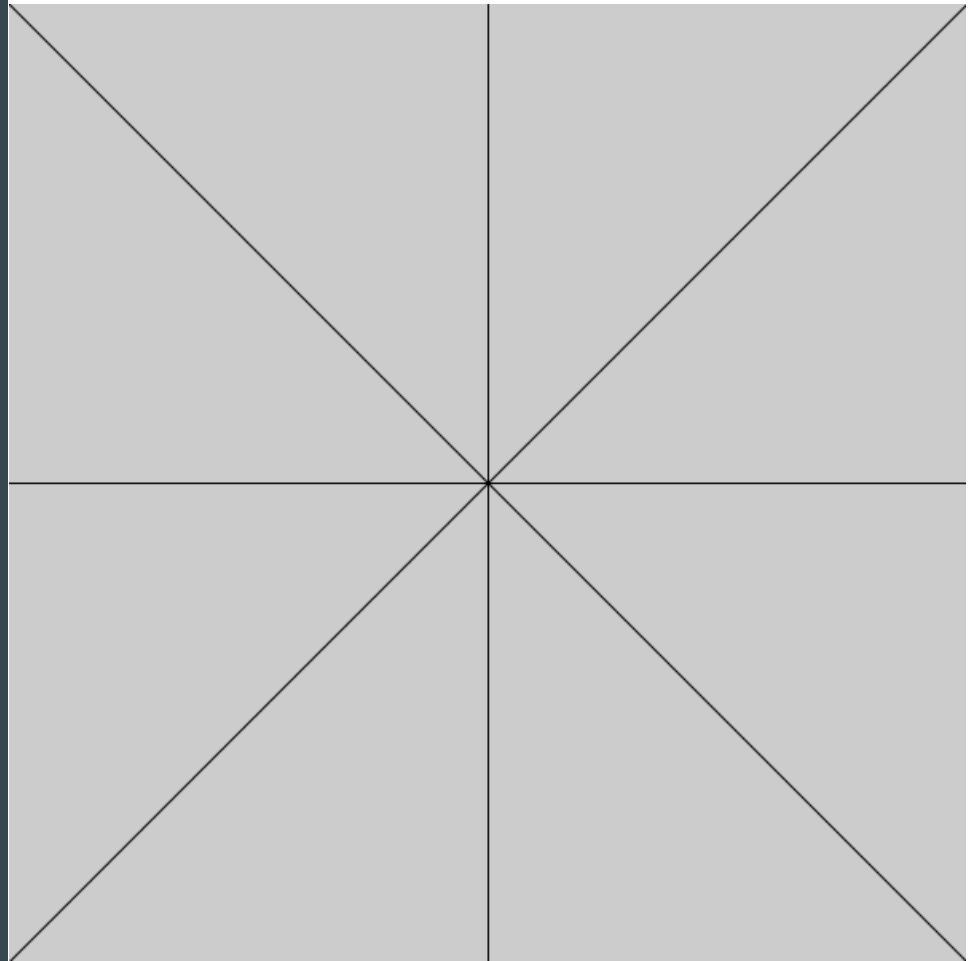
Segundo ponto:

x2 -> Coordenada do eixo x

y2 -> Coordenada do eixo y

Exercício 3

Desenhe 4 linhas que se cruzam
no centro do seu sketch.
O canvas deve ter tamanho
600x600px



Retângulo

Desenhando retângulos e
quadrados

Comando:

rect(x,y, a,b);

x -> Coordenada do eixo x

y -> Coordenada do eixo y

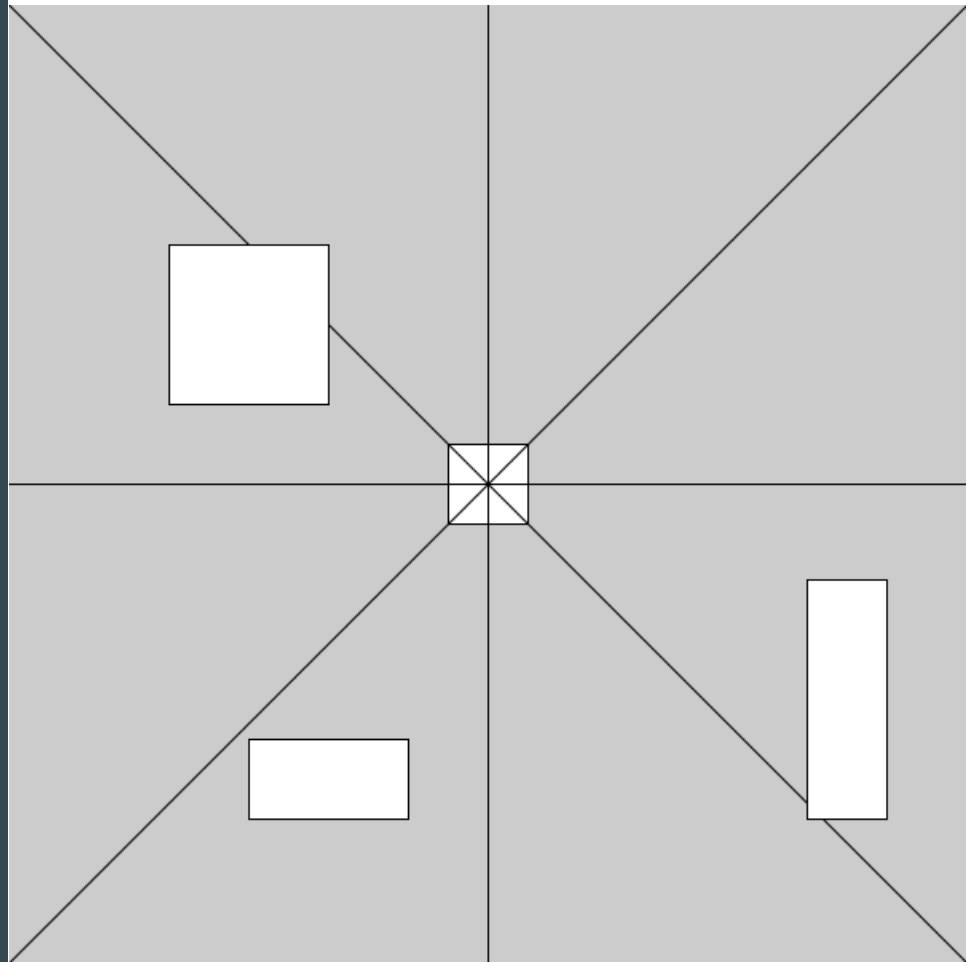
(Coordenadas do canto superior
esquerdo)

a -> largura do retângulo

b -> altura do retângulo

Exercício 4

Desenhe 4 retângulos sobre as linhas desenhadas no exercício anterior. O resultado final deve ficar semelhante à imagem ao lado.



Ellipse

Desenhando elipses e círculos

Comando:

ellipse(x,y, a,b);

x -> Coordenada do eixo x

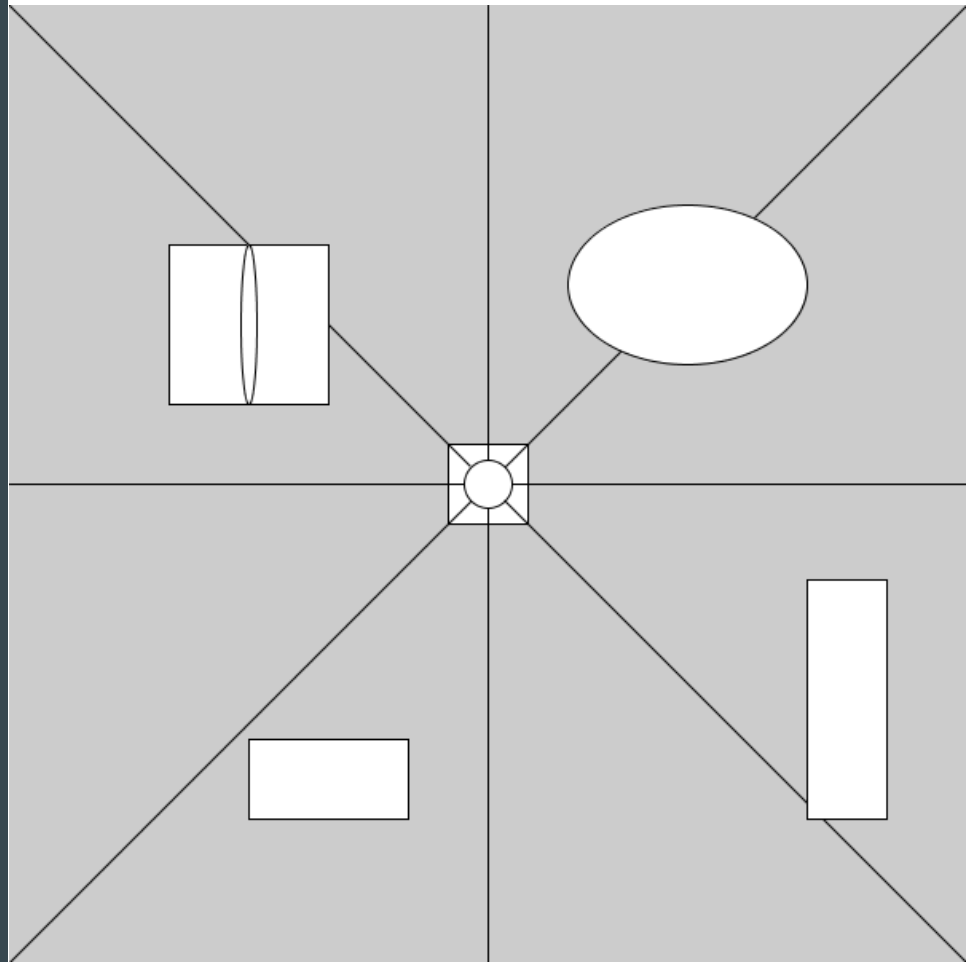
y -> Coordenada do eixo y
(Coordenadas do centro)

a -> largura da ellipse

b -> altura da ellipse

Exercício 5

Desenhe 3 elipses sobre as formas dos exercícios anteriores. O resultado final deve ficar semelhante à imagem ao lado.



Quadriláteros

Desenhando quadriláteros
irregulares

**quad(x1,y1, x2,y2, x3,
y3, x4,y4);**

Primeiro ponto:

x1 -> Coordenada do eixo x

y1 -> Coordenada do eixo y

...

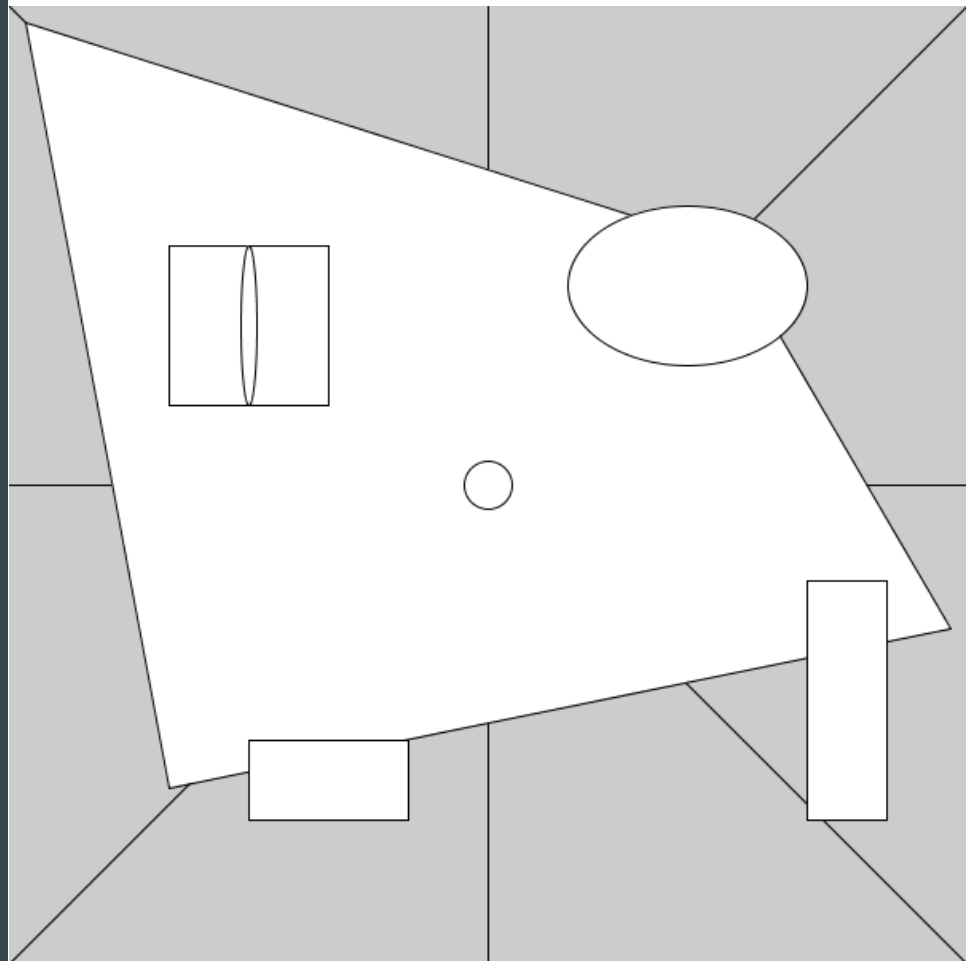
Quarto ponto:

x4 -> Coordenada do eixo x

y4 -> Coordenada do eixo y

Exercício 6

Desenhe 1 quadrilátero irregular atrás das imagens desenhadas e por cima das linhas. O resultado final deve ficar semelhante à imagem ao lado.



Triângulos

Desenhando triângulos.

Comando:

triangle(x1,y1, x2,y2, x3,y3);

Primeiro ponto:

x1 -> Coordenada do eixo x

y1 -> Coordenada do eixo y

Segundo ponto:

x2 -> Coordenada do eixo x

y2 -> Coordenada do eixo y

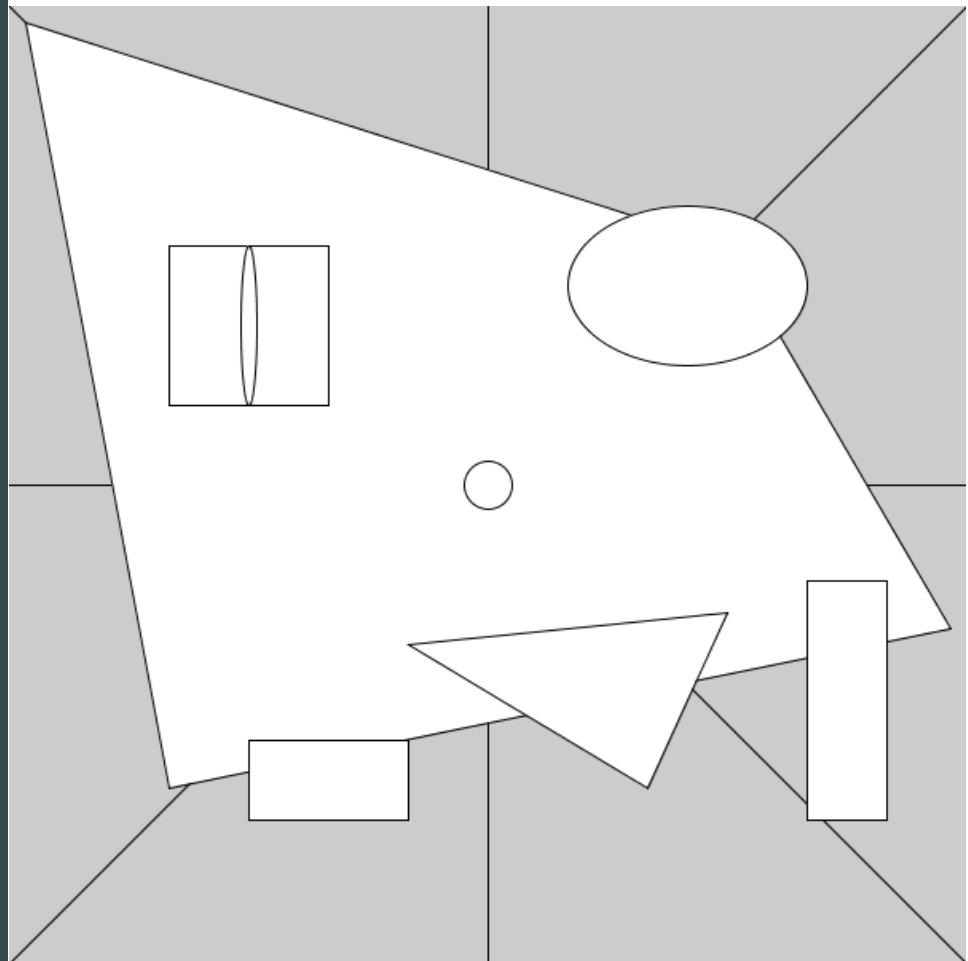
Terceiro ponto:

x3 -> Coordenada do eixo x

y3 -> Coordenada do eixo y

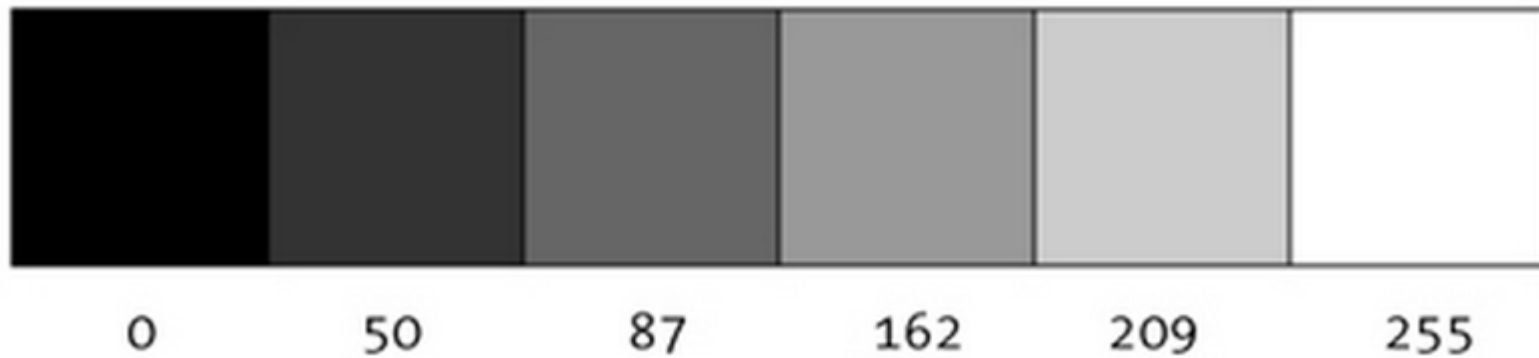
Exercício 7

Desenhe 1 triângulo irregular sobre as formas anteriores. O resultado final deve ficar semelhante à imagem ao lado.

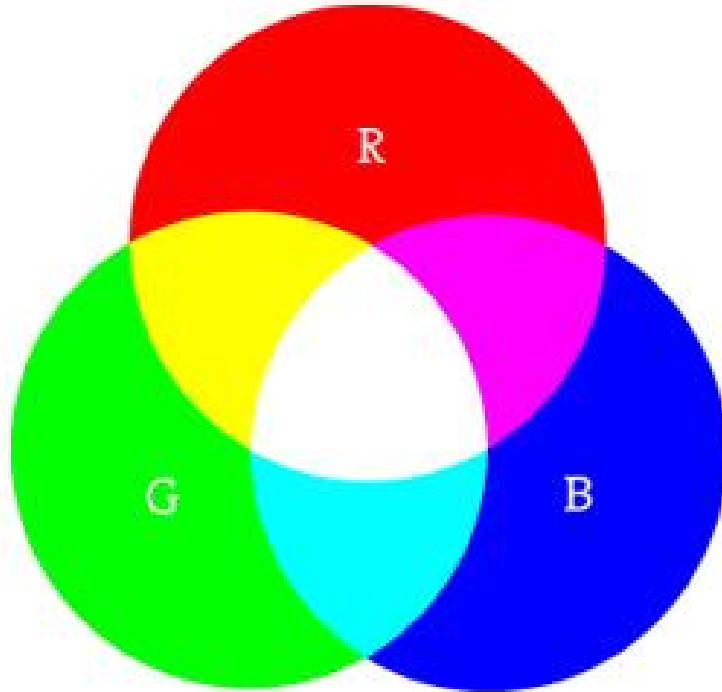


Cores

Tons de cinza



RGB (Red-Green-Blue)



Preenchendo formas

Preencher (cinza)

Preencher as formas com tons de cinza

fill(cinza);

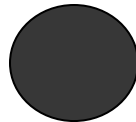
cinza -> valores entre 0 e 255 para colorir em tons de cinza

Exemplo:

fill(222);



fill(55);



Preencher (cores)

Preencher as formas com alguma cor

`fill(r, g, b);`

`fill(hex);`

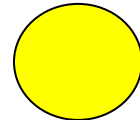
r,b,g -> valores de red, green, blue, respectivamente, cada um entre 0 e 255

hex -> cor em hexadecimal (ex: #373737)

Exemplo:

`fill(#0000C8);`

`fill(255,255,0);`



Preencher com transparência

Preencher as formas com alguma cor transparente

`fill(r, g, b, alpha);`

`fill(hex, alpha);`

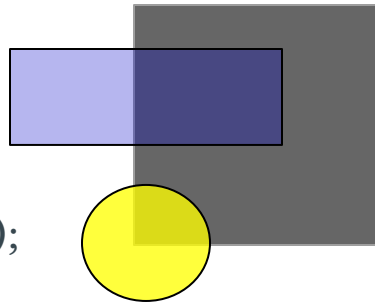
`fill(cinza, alpha);`

alpha -> valor de 0 a 255 que representa a opacidade de um elemento.

Exemplo:

`fill(#0000C8, 50);`

`fill(255,255,0, 200);`



Cor de fundo do canvas

Comando para alterar a cor de fundo do seu canvas.

```
background(125); // tom de cinza
```

```
background(255,0,0); // vermelho
```

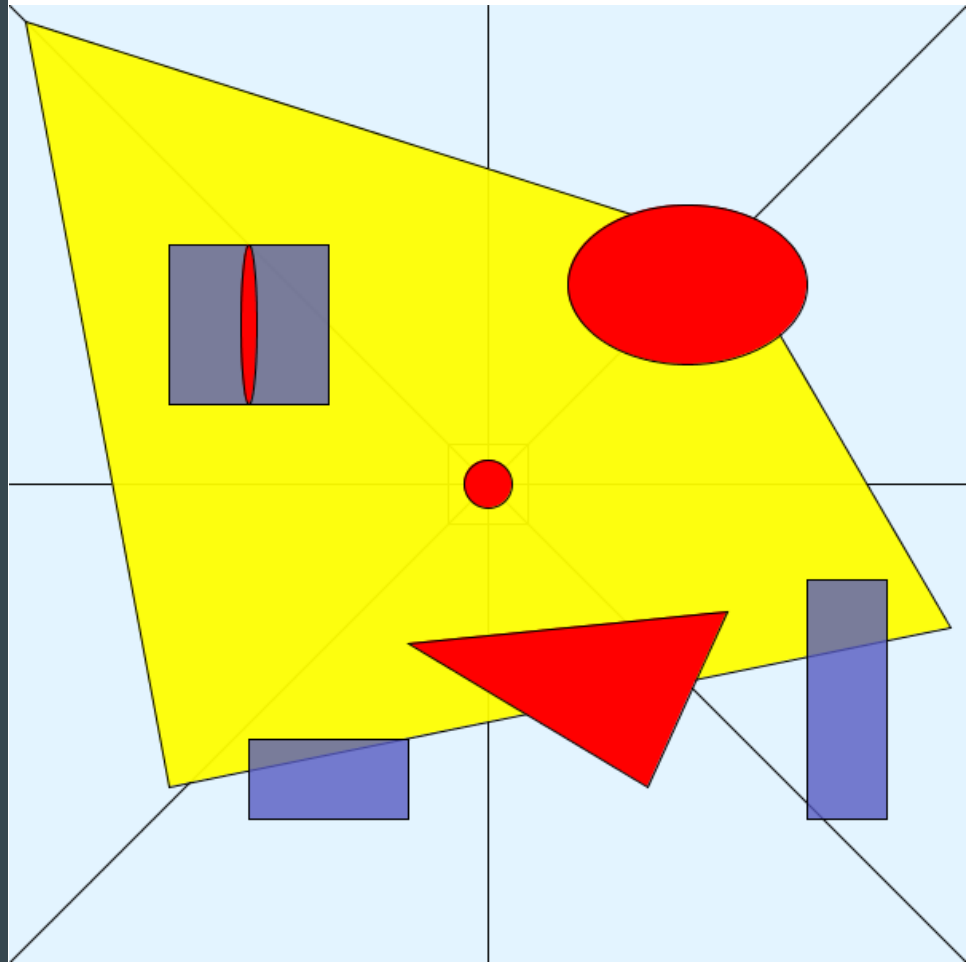
```
background(#00FF00); // verde
```

*Recomendado sempre ser utilizado **depois** do comando de size();*

Exercício 8

Colorir as formas de acordo com o exemplo ao lado.

Lembre-se que as formas são coloridas de acordo com a última instrução de fill().



**Colorindo e manipulando
bordas**

Borda (cinza)

Instrução para alterar a borda para tons de cinza

stroke(cinza);

cinza -> valores entre 0 e 255 para colorir em tons de cinza

Exemplo:

stroke(222);



stroke(55);



Borda (cores)

Alterar a cor da borda

`stroke(r, g, b);`

`stroke(hex);`

r,b,g -> valores de red, green, blue, respectivamente, cada um entre 0 e 255

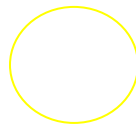
hex -> cor em hexadecimal (ex: #373737)

Exemplo:

`stroke(#0000C8);`



`stroke(255,255,0);`



Colorir borda com transparência

```
stroke(r, g, b, alpha);
```

```
stroke(hex, alpha);
```

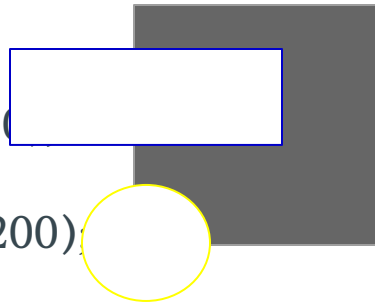
```
stroke(cinza, alpha);
```

alpha -> valor de 0 a 255 que representa a opacidade de um elemento.

Exemplo:

```
stroke(#0000C8, 50)
```

```
stroke(255,255,0, 200);
```



Peso/Grossura da borda

Instrução para alterar a grossura
da borda

strokeWeight(peso);

peso -> grossura da borda (em pixels)

Exemplo:

`strokeWeight(0.5);` // muito fino

`strokeWeight(10);` // **pesado**



Sem borda ou preenchimento

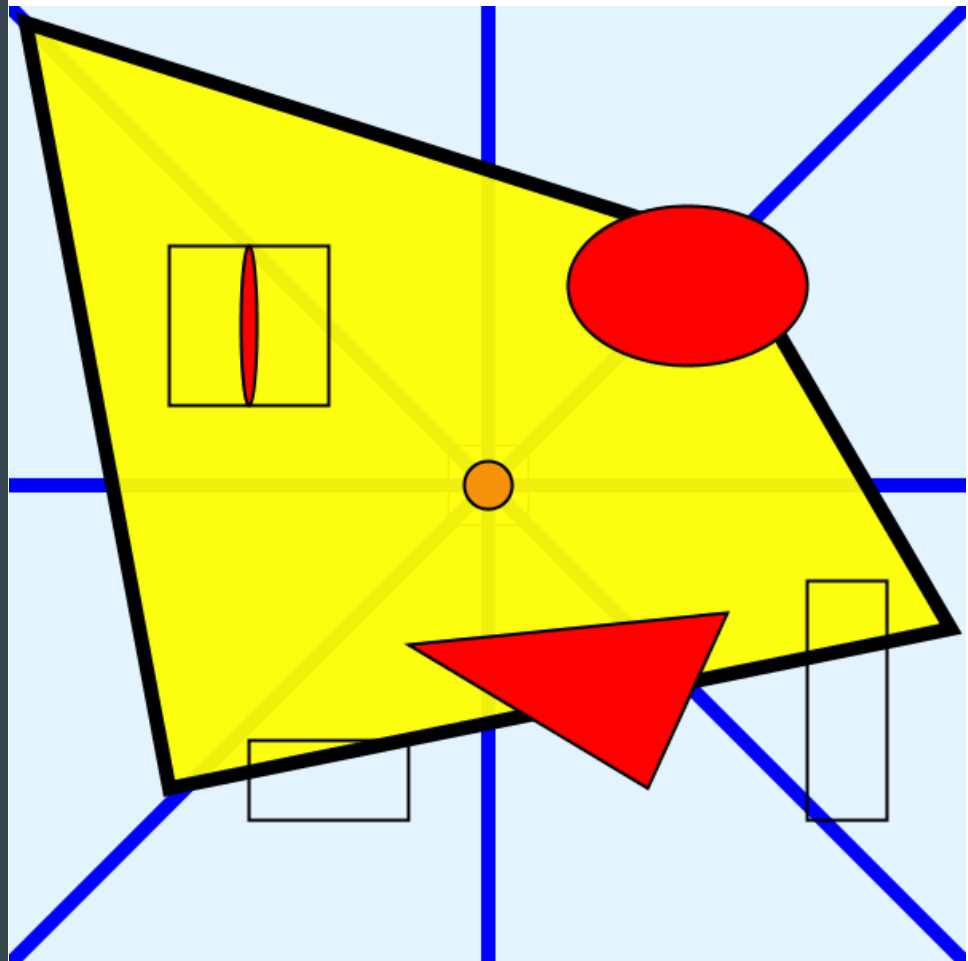
Comandos para não preencher a forma ou não desenhar nenhuma borda

```
noFill();
```

```
noStroke();
```

Exercício 9

Altere as bordas de acordo com a imagem ao lado. Note que algumas formas estão sem preenchimento.



Salvando sua criação

Formatos de imagem

Salvando o resultado visual do sketch.

Lembre-se: o comando `saveFrame()` deve ser executado sempre ao final do arquivo.

```
saveFrame("arquivo");
```

arquivo -> nome do seu
arquivo+formato

formato pode ser: jpg, png, tiff, tga

Exemplo:

```
saveFrame("formas.png");
```

Salvando um PDF

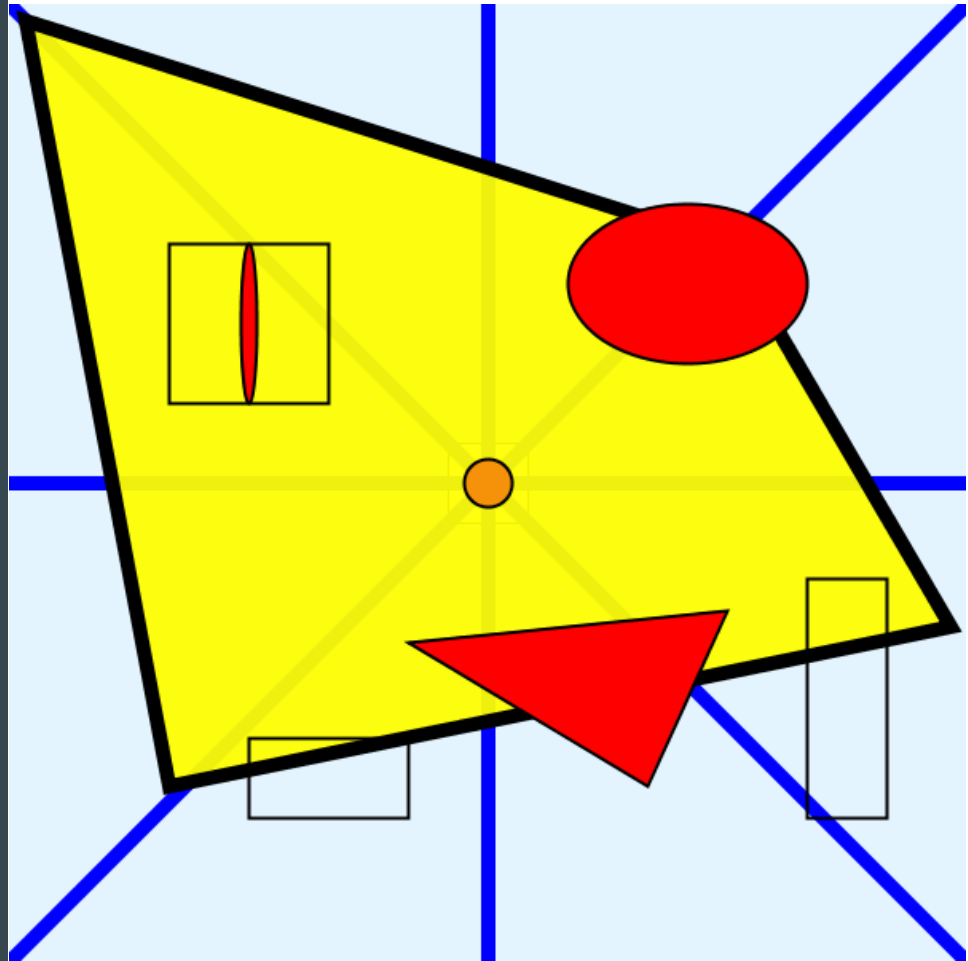
Comandos para salvar um arquivo PDF.

Pode ser importado em programas vetoriais como Illustrator ou Inkscape.

```
import processing.pdf.*;  
  
size(600,600);  
  
beginRecord(PDF, "line.pdf");  
  
// seu código aqui  
  
endRecord();
```

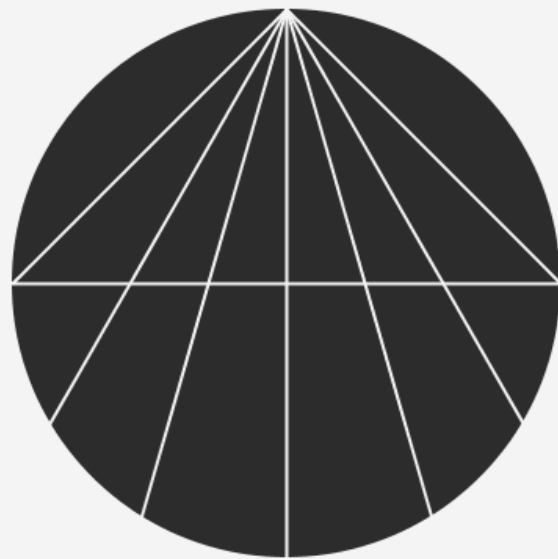
Exercício 10

Salvar a imagem ao lado como um arquivo PNG e também um arquivo PDF.



Exercício 11

Use seu conhecimento para criar a imagem ao lado ou qualquer coisa que vier à sua mente.



Exercício 12

Use seu conhecimento para criar a imagem ao lado ou qualquer coisa que vier à sua mente.



Resumão

Canvas

size()
background()

Formas

point(x,y)
line(x1,y1, x2,y2)
rect(x,y, a,b)
ellipse(x,y, a,b)
quad(x1,y1, x2,y2, x3,y3, x4,y4);
triangle(x1,y1, x2,y2, x3,y3)

Cores

fill(color)
stroke(color)
noFill()
noStroke()
strokeWeight(weight)

Salvando

saveFrame("file.png/jpg/tiff/tga")
import processing.pdf.*;
beginRecord(PDF,"file.pdf");
endRecord();

Exercício 13

Livre! Crie o que você quiser
utilizando o que você aprendeu
hoje (:



Referências

<http://hello.processing.org>

<http://hello.processing.org/guide/>

<https://processing.org/reference/>

<https://processing.org/overview/>

<https://processing.org/tutorials/>

<https://processing.org/tutorials/drawing/>

<https://processing.org/tutorials/color/>

<https://processing.org/examples/>