

IMFVJ1 – Prática laboratorial

Aula 1

- Apresentação
- Instalação software
- Cor
- Coordenadas ecrã



Apresentação



Professor



Disciplina



Avaliação



Senhora da Graça

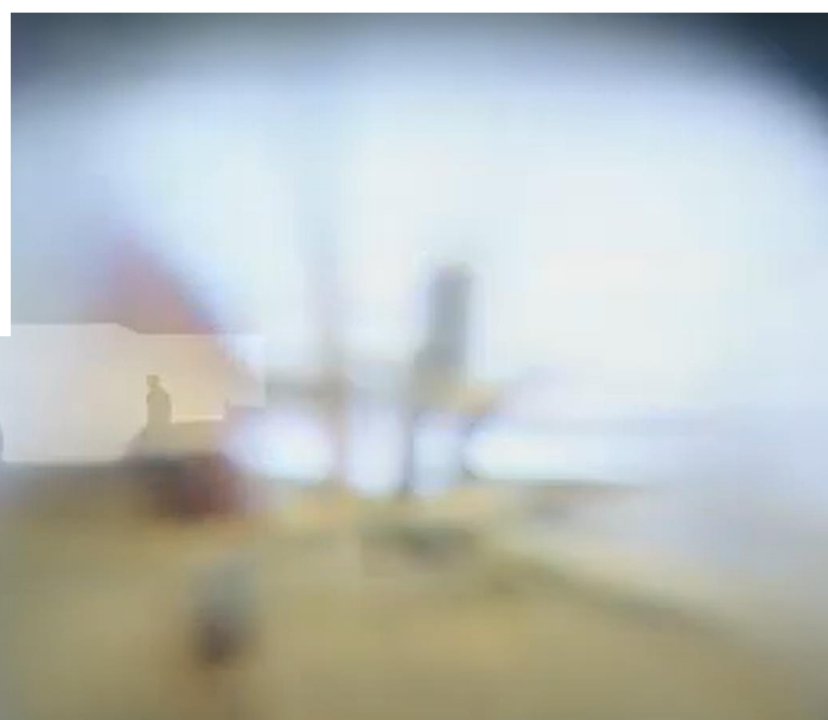


xTNZ Promo



Where is Lourenço Marques? - the population of 'story-tellers'

MSc/PhD Arts and Computational Technologies



EasyPopulation Unity Asset store



- <https://assetstore.unity.com/packages/tools/behavior-ai/easy-population-102334>
- <https://www.youtube.com/watch?v=GKX0Rfr3gDc>

GhostDance



<https://ghostdance.ulusofona.pt/>

Laboratório IMFVJ1

Sistemas de coordenadas cartesianas.

Múltiplos espaços de coordenadas: utilidade prática, exemplos, transformações.

Bases de trigonometria.

Vectores: definição, propriedades, operações.

Matrizes: definição e interpretação geométrica, determinantes, inversa; matrizes homogéneas.

Transformações lineares: rotação, escalamento, projecção;

Hierarquias e combinação de transformações

Rotação em 3D: forma matricial, ângulos de Euler e quaterniões.

Probabilidades e distribuições matemáticas

Funções paramétricas

Primitivas geométricas
Matemática para gráficos 3d

Exemplos em NumPy e Pygame p5

Avaliação

- Todas as sessões, presença e trabalho realizado na sala

- Avaliação contínua

Componente teórica (50%)

(nota mínima 7 valores na componente teórica e nota mínima de 5 valores no último teste teórico de avaliação):

1 a 4 testes teóricos

Componente prática (50%)

(nota mínima de 7 valores na componente prática) Exercícios resolvidos em sala de aula, ou trabalhos disponíveis no moodle

- Segunda época

Componente teórica - 10 valores (nota mínima 7): Exame [10v] Componente prática - 10 valores (nota mínima 7): Projeto [10v]

- Época especial

Componente teórica - 10 valores (nota mínima 7): Exame [10v] Componente prática - 10 valores (nota mínima 7): Projeto [10v] A avaliação de época especial está disponível de acordo com o artigo 19º do Regulamento Geral de Avaliação da Universidade Lusófona.

Ferramentas (para IMFVJ1)

Python

Visual
Studio Code

P5.PyPI

Socrative

Moodle

Cursos

Licenciaturas

Mestrados

Mestrados Integrados

Recursos

Guia de Acolhimento

Biblioteca

Click - Portal de e-Learning

🏠 Página Inicial

eMail - Lisboa

eMail - Porto

NetPA

Moodle

Colibri

Avadoc

Portal do Col

Ficha Docent

Office 365

Intranet

Serviços

fernando
lopes

Bolsas PRR | Caminhos disciplinas

Estão abertas as candidaturas às Bolsas de Estudo do PI da candidatura a um dos cursos num estabelecimento

Candidatar-se

VIEW COURSE

Ano Letivo 2022/2023

Introdução à Matemática e Física para...

Ano Letivo 2023/2024

Introdução à Matemática e Física para...

Ano Letivo 2022/2023

Introdução à Matemática e Física para...

Ano Letivo 2023/2024

Introdução à Matemática e Física para...

Avaliação Contínua

- HW1 - Mini-Biografia
- HW 2A (para estudantes sem Matemática A concluída)
- HW3 - Álgebra de matrizes
- HW2B - Funções (Estudantes com Aprovação Matemática A)

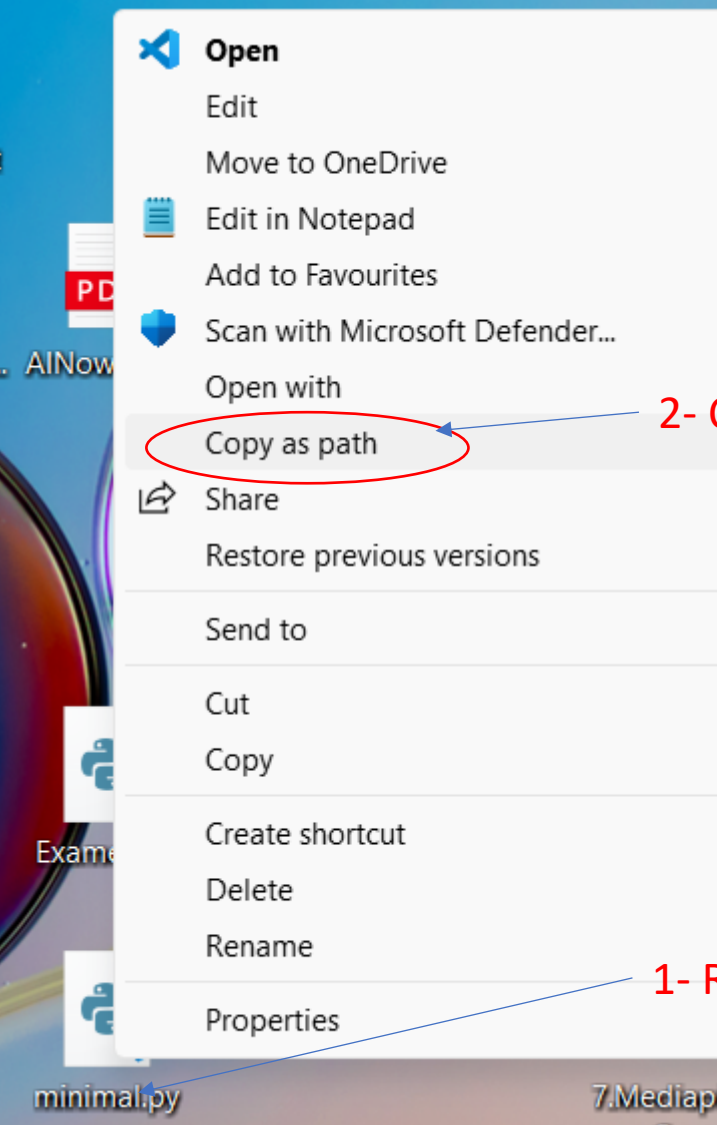
Laboratório de implementação em python

Este módulo é relativo à implementação dos conceitos matemáticos e físicos, usando a linguagem python.

📄 Instalação das ferramentas de software

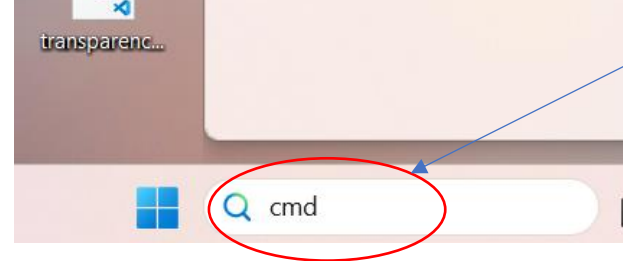
Modo comando

Criar e executar o primeiro programa,



2- Copiar endereço

1- Right Click



3- Abrir linha de comando

```
Command Prompt
Microsoft Windows [Version 10.0.22631.4169]
(c) Microsoft Corporation. All rights reserved.
C:\Users\ruifi>cd "C:\Desktop\minimal.py"
```

4.1 - Escrever cd

4.2- Fazer Ctrl+V (paste do endereço)

```
Command Prompt
Microsoft Windows [Version 10.0.22631.4169]
(c) Microsoft Corporation. All rights reserved.
C:\Users\ruifi>cd "C:\Desktop"
```

4.3 – Remover o nome do ficheiro

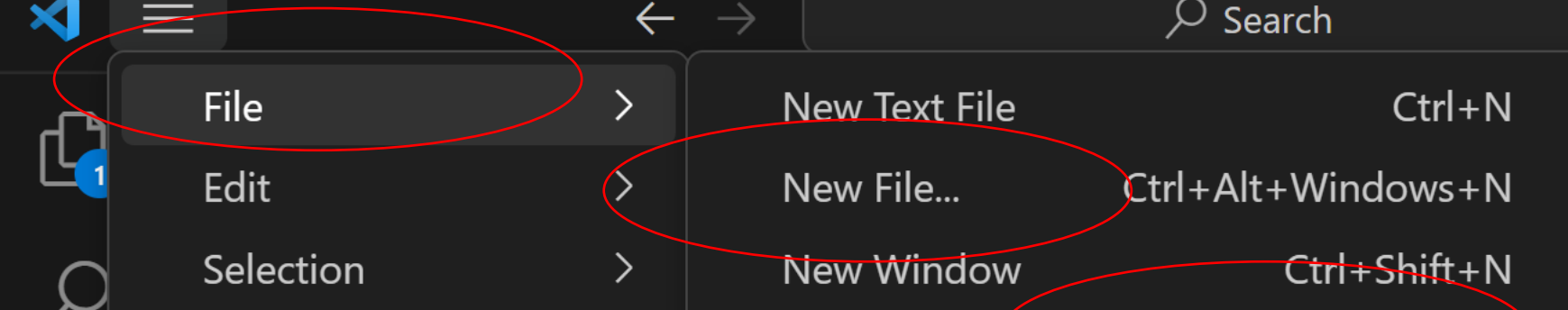
4.4 - Carregar em Enter

```
Command Prompt
Microsoft Windows [Version 10.0.22631.4169]
(c) Microsoft Corporation. All rights reserved.
C:\Users\ruifi>cd "C:\Desktop"
C:\Desktop>python minimal.py
```

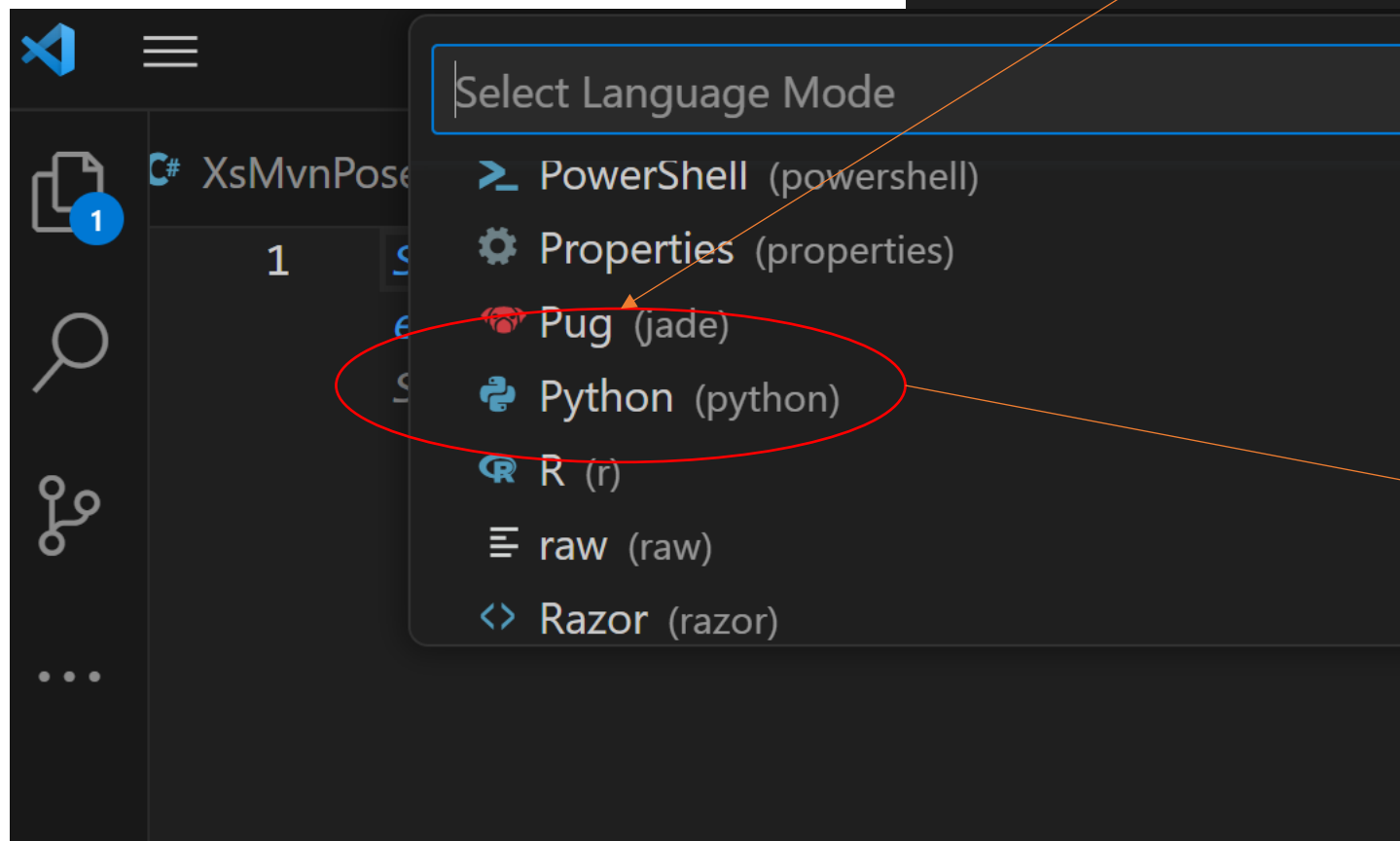
5 – Chamar o python para executar o ficheiro (python minimal.py)

Visual Studio Code

Criar e executar o primeiro programa,
eventualmente instalar extensões para python



1 *Select a language, or fill with template, or open a editor to get started.
Start typing to dismiss or don't show this again.*



No canto inferior direito irá aparecer uma janela a dizer para instalar um extensão se esta ainda não estiver instalada. Confirme a instalação.

Aceda ao git da disciplina

<https://github.com/virtualeiro/pygame-imfvj1-2024-2025>

e copie o ficheiro

[minimal.py](#)

```
import pygame
pygame.init()
screen = pygame.display.set_mode((640, 480))
pygame.display.set_caption("Test Window")
clock = pygame.time.Clock()
```

```
running = True
while running:
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            running = False
```

```
screen.fill((0, 0, 0))
pygame.display.flip()
clock.tick(60)
```

```
pygame.quit()
```

```

pygame.init()
screen = pygame.display.set_mode((0,0))
pygame.display.set_caption("Test 1")
clock = pygame.time.Clock()

running = True
while running:
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            running = False

    screen.fill(0)
    pygame.display.flip()
    clock.tick(60)

pygame.quit()

```


Estrutura de um programa (minimal.py)

```
import pygame
```

← Importa biblioteca pygame

```
pygame.init()
```

```
screen = pygame.display.set_mode((640, 480))
```

```
pygame.display.set_caption("Test Window")
```

```
BLACK = (0, 0, 0)
```

```
clock = pygame.time.Clock()
```

```
running = True
```

```
while running:
```

```
    for event in pygame.event.get():
```

```
        if event.type == pygame.QUIT:
```

```
            running = False
```

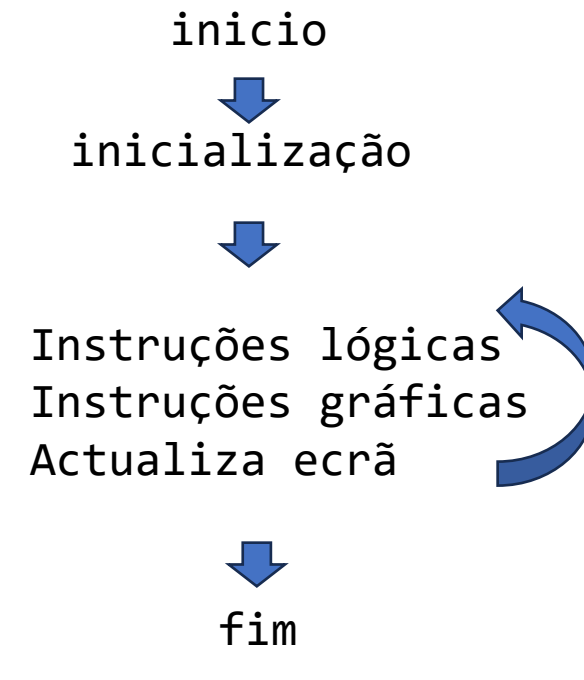
```
    screen.fill(BLACK)
```

```
    pygame.display.update()
```

```
    clock.tick(60)
```

← Actualiza ecrã

```
pygame.quit()
```



← Ciclo principal:

Cria um loop que permite executar as instruções repetitivamente, até ser satisfeita uma condição de paragem

← Monitoriza todos os eventos-sistema que aconteceram (foram registados) desde a ultima vez que esta instrução foi executada

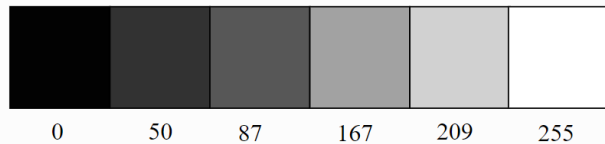
Cor

Preto e branco ou cinzentos.

0 - Preto,

255 - Branco.

Valores intermédios – Escala de cinzas de preto a branco.

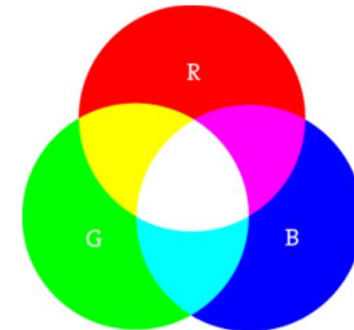


```
screen.fill((0, 0, 0))
```

Define a cor de fundo da janela

RGB

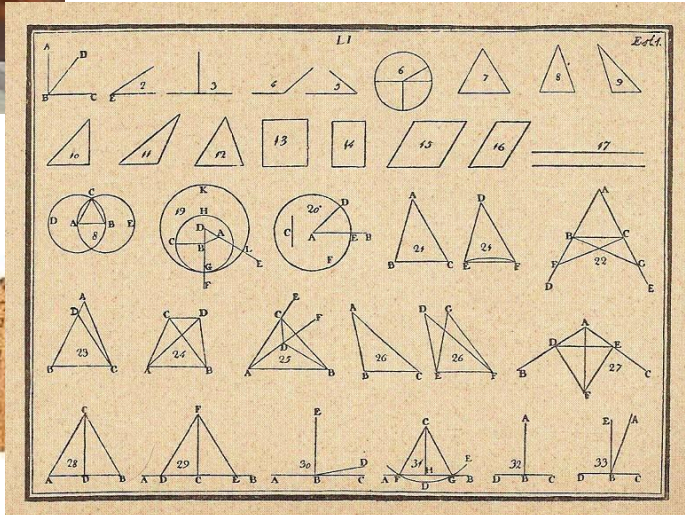
- Red + Green = Yellow
- Red + Blue = Purple
- Green + Blue = Cyan (blue-green)
- Red + Green + Blue = White
- No colors = Black



Espaço e coordenadas



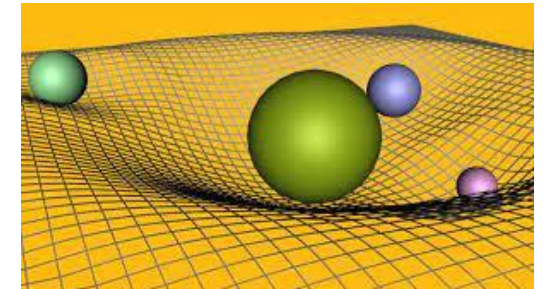
Escola de Atenas



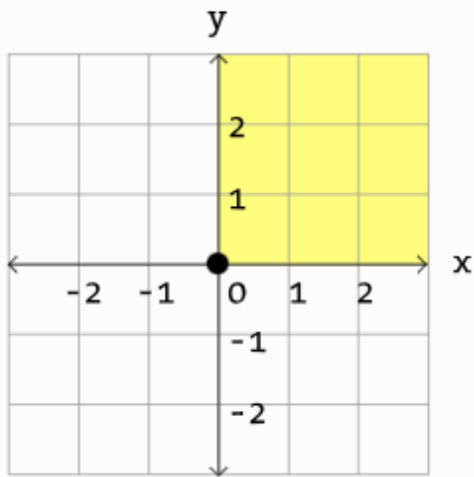
Euclides 300 BC



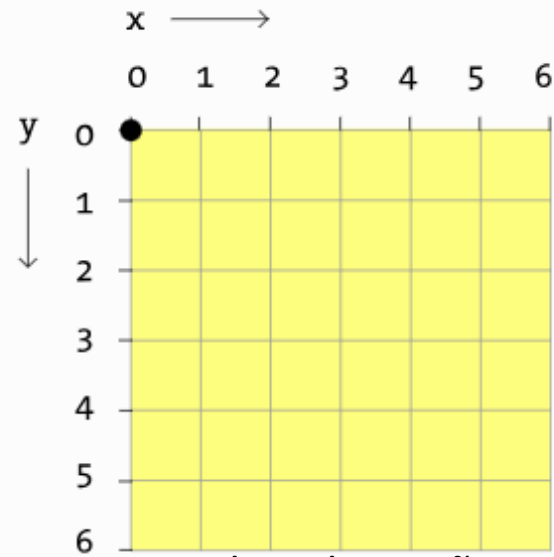
Descartes 1637



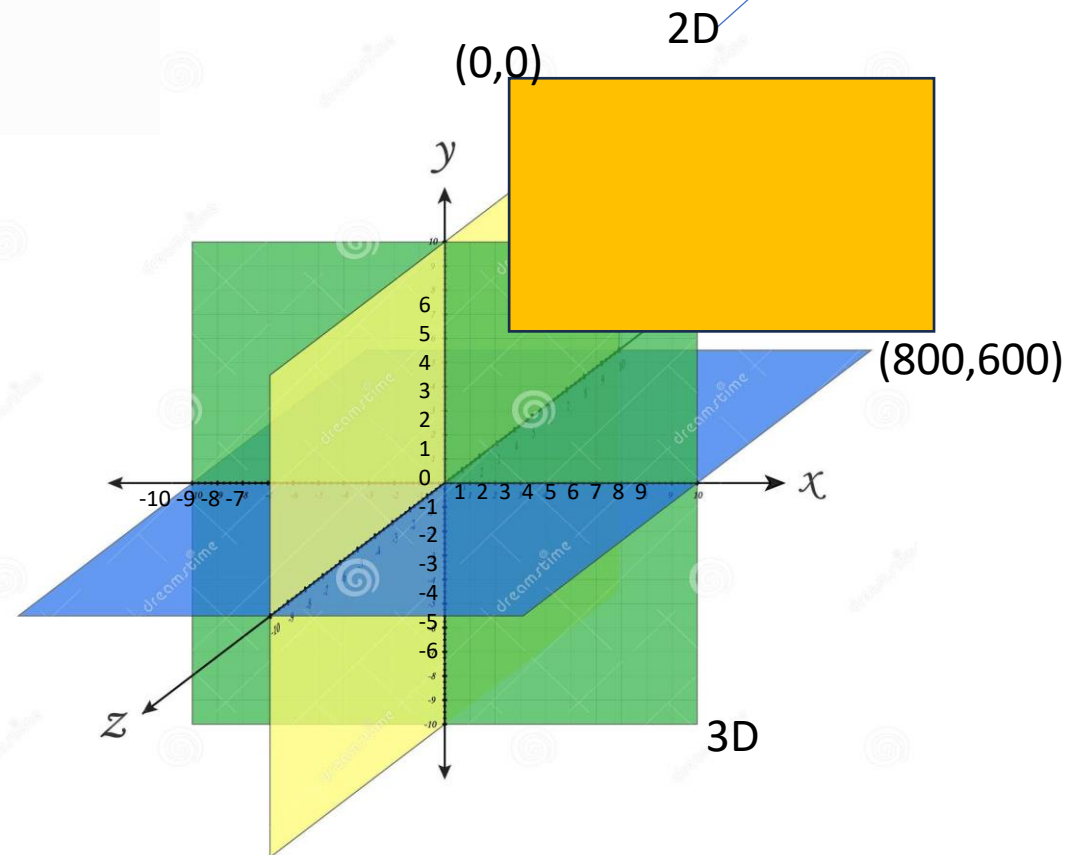
Einstein 1905



Coordenadas Cartesianas



Coordenadas Ecrã



Linhas

```
pygame.draw.line()
```

```
line(surface, color, start_pos, end_pos)
```

```
line(surface, color, start_pos, end_pos, width=1)  
      (r, g, b)      (x, y)      (x, y)
```

```
pygame.draw.line(screen, (255,0,0), (0,0), (640,480),20)
```

Referencia:

<https://www.pygame.org/docs/ref/draw.html#pygame.draw.line>

(2,4)

	X							
	0	1	2	3	4	5	6	7
0					■			
1								
2			■					
3								
4			■					
5	■							
6						■		
7								