

# Computação Gráfica na Web

## Introdução à biblioteca THREE.js

Lucas Diniz da Costa

Universidade Federal de Juiz de Fora (UFJF)  
2021



# Agenda

---

- Introdução
- Vantagens
- Aplicações
- Material de apoio
- Planejamento do minicurso

# Introdução

---

- Este minicurso visa estabelecer uma introdução à Computação Gráfica na web através do uso da biblioteca Three.js;
- A Three.js possui muitas ferramentas e artifícios interessantes para desenho e renderização de elementos visuais na web.
- Com o forte crescimento do ambiente web, a biblioteca Three.js veio para facilitar o desenvolvimento de aplicações 2D e 3D na web.

# Three.js

---

- Foi desenvolvida em 2010 por Ricardo Cabello (Mr. doob)
- Criada inicialmente para mostrar o potencial de gráficos 3D nos navegadores web.
- Utiliza o WebGL como estrutura nos navegadores de internet, ou seja, uma biblioteca para facilitar o uso do OpenGL na web.
- Disponível em: <https://threejs.org/>

# Three.js

[three.js](#) r120

**Learn**

- [documentation](#)
- [examples](#)
- [editor](#)

**Community**

- [questions](#)
- [discord](#)
- [forum](#)
- [slack](#)
- [twitter](#)

**Code**

- [github](#)
- [download](#)
- [devtools](#)

**Resources**

- [Three.js Fundamentals](#)
- [Discover Three.js](#)
- [Learn Three.js](#)

submit project

# Vantagens

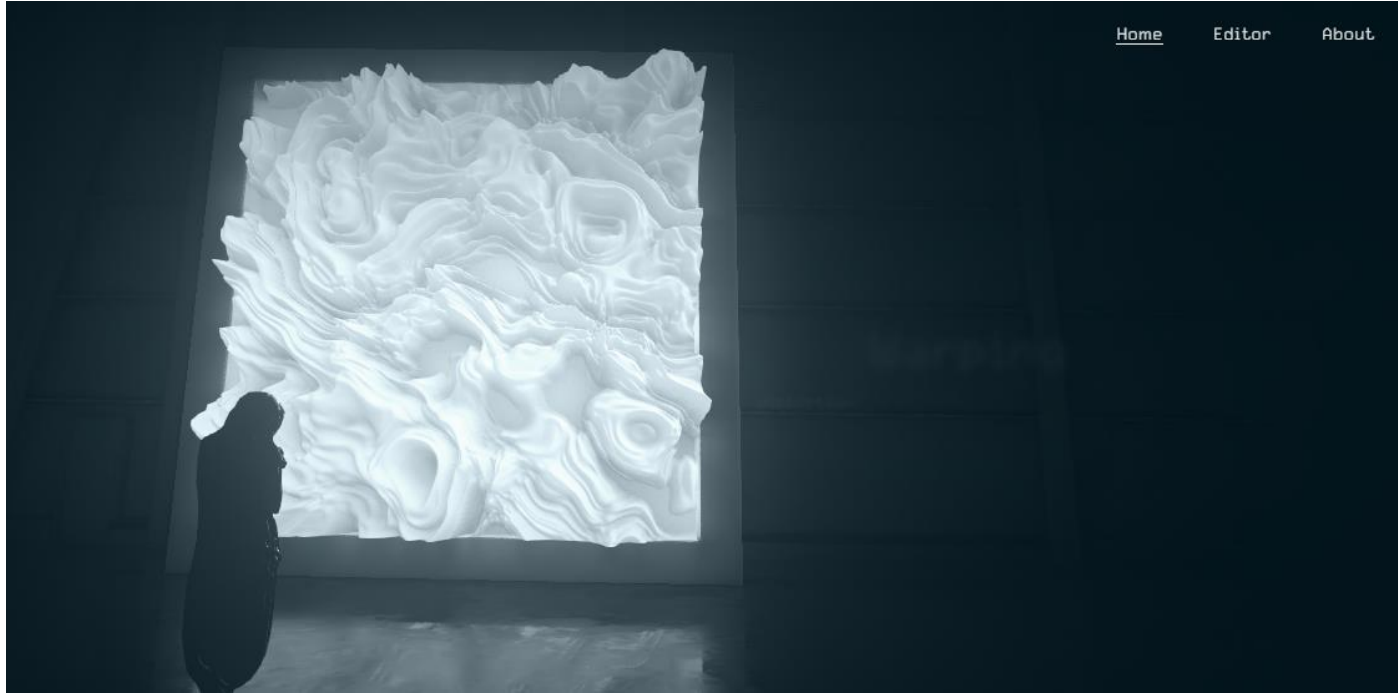
---

- Biblioteca de fácil configuração
- Executa em navegadores de internet
- Executam nos dispositivos móveis
- Atualizações frequentes
- Possibilidade de criar projetos mais complexos em 3D
- Alta gama de bibliotecas que complementam as funcionalidades da Three.js

---

# Aplicações

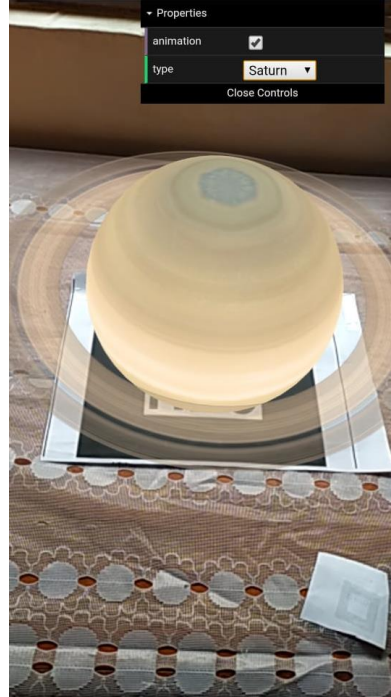
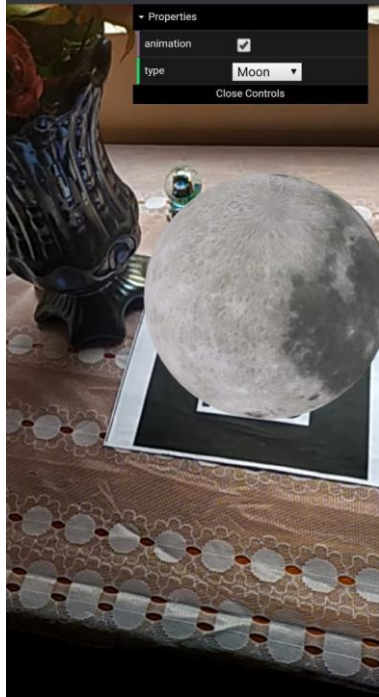
# Simulação de Fluidos



- Fonte: <https://cineshader.com/>



# Realidade Aumentada



- Fonte: <https://avrgroup.github.io/vrtools/projects/solar-system.html>

# Realidade Virtual



- Fonte: <https://mixedreality.mozilla.org/hello-webxr/>

# Modelagem de cidades



- Fonte: <https://www.harp.gl/>

# Amostra de produtos



- Fonte: <https://renaultespace.littleworkshop.fr/>

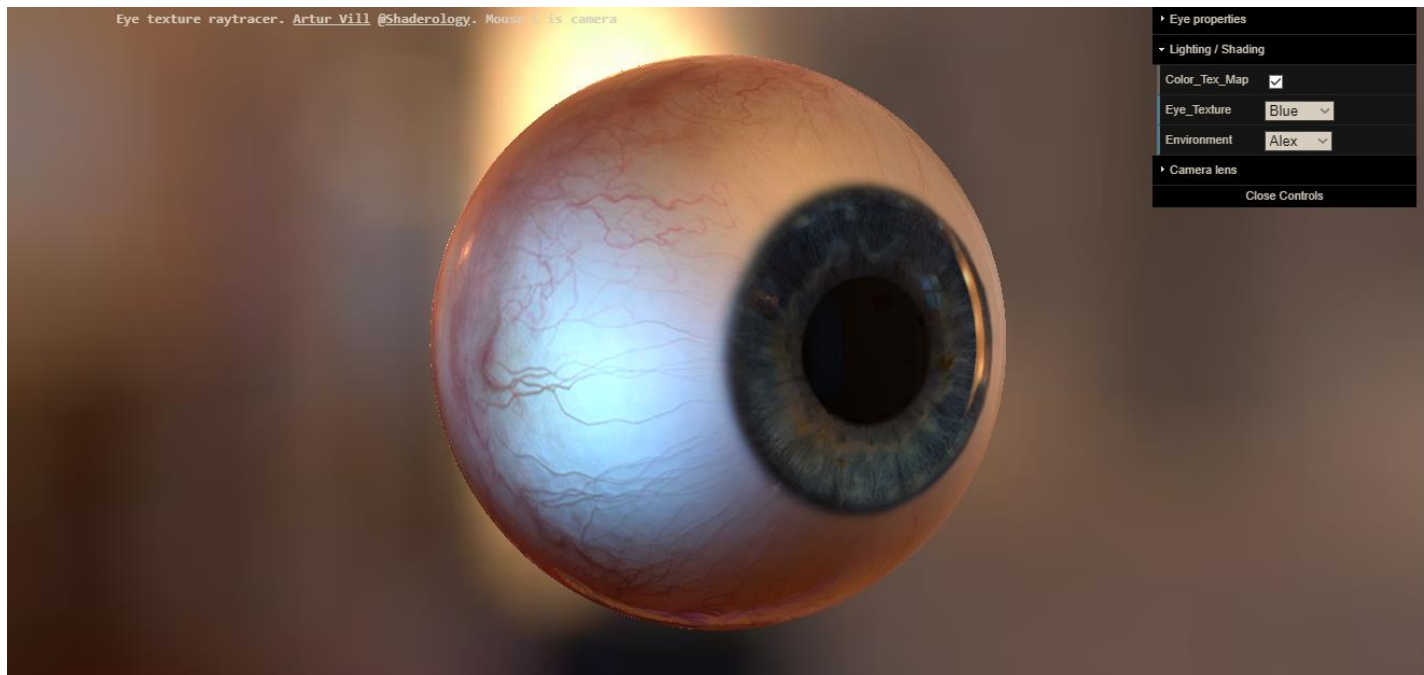


# Amostra de produtos



- Fonte: <http://www.simonreeves.com/projects/db5/>

# Alto detalhamento

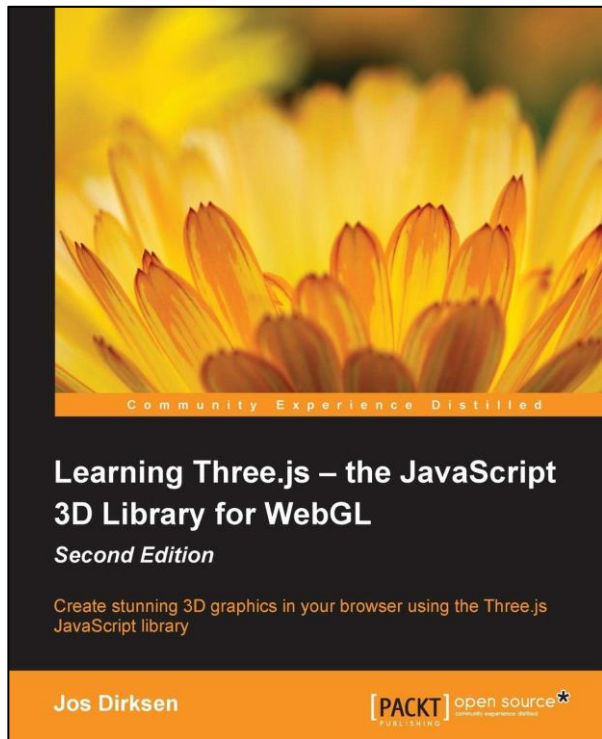


- Fonte: <http://www.vill.ee/eye/>

---

# Material de apoio

# Material de apoio



- Autor: Jos Dirksen
- Nome: **Learning Three.js – the Javascript 3D Library for WebGL**
- Repositório dos exemplos do livro:
- Second Edition - 2015:
- <https://github.com/josdirksen/learning-threejs>
- Third Edition - 2018:
- <https://github.com/PacktPublishing/Learn-Three.js-Third-Edition>



# Planejamento

---

- Apresentação do minicurso;
- Configuração do ambiente de desenvolvimento;
- Aplicação básica em Three.js;
- Câmeras e Luzes;
- Geometrias e Materiais;
- Carregamento de objetos 3D;
- Aplicação do Sistema Solar;
- Outros recursos presentes na Three.js e bibliotecas auxiliares;

---

# Configuração do ambiente de desenvolvimento

# Estrutura básica de um projeto

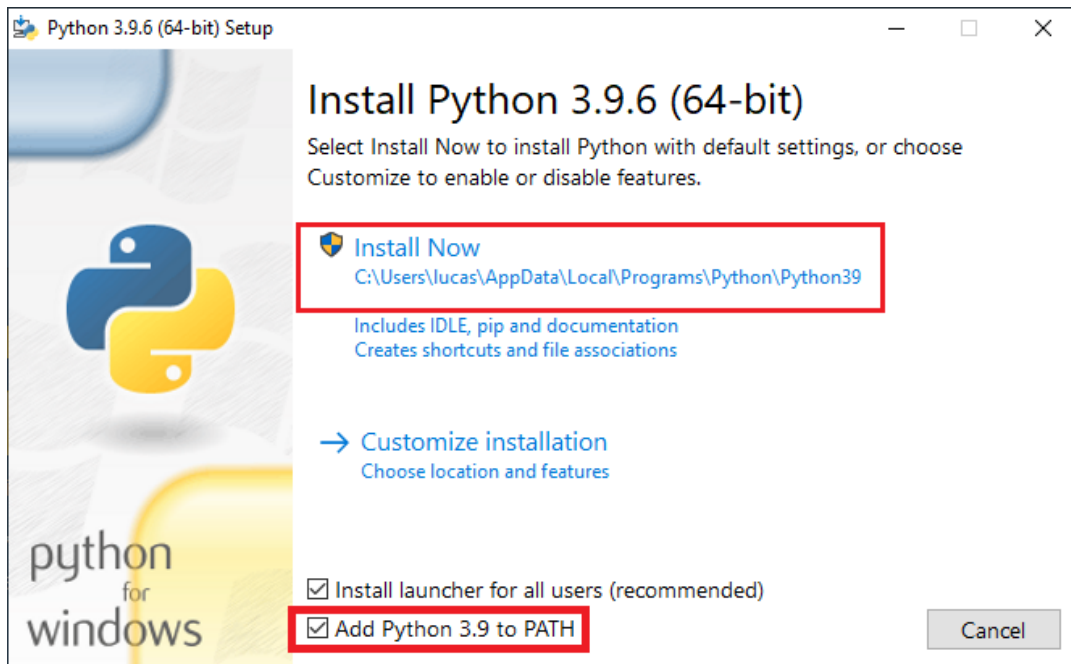
- **Arquivo HTML:** Página HTML que irá carregar as bibliotecas e dispor a interface da Three.js;
- **Servidor local ou online:** Para o carregamento de texturas e objetos é fundamental, pois sem ele, por medida de segurança os navegadores podem impedir o carregamento de arquivos externos;
- **Biblioteca Three.js;**
- Eventualmente o uso de bibliotecas auxiliares para aprimorar o projeto será necessário.

# Servidor local (Alternativa 01)

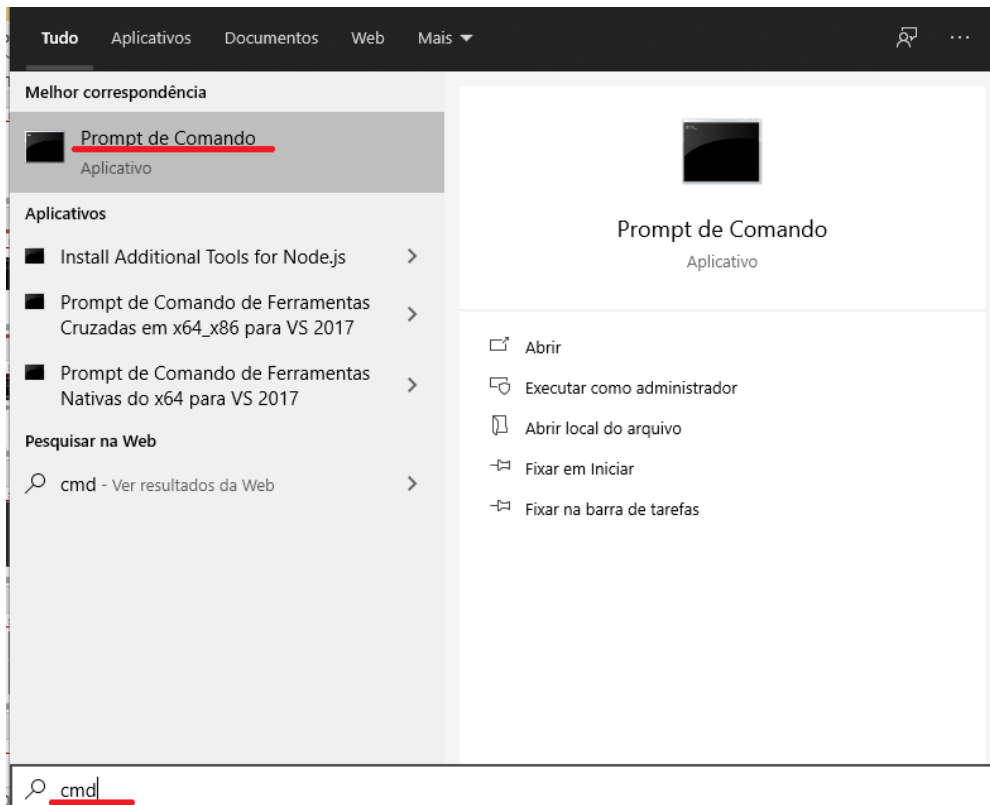


Site para download: <https://www.python.org/downloads/>

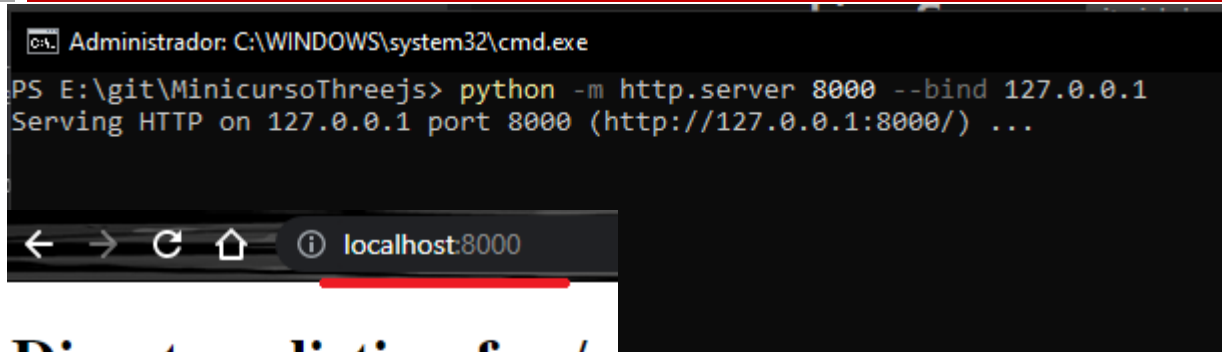
# Servidor local (Alternativa 01)



# Servidor local (Alternativa 01 - Windows)



# Servidor local (Alternativa 01 - Windows)



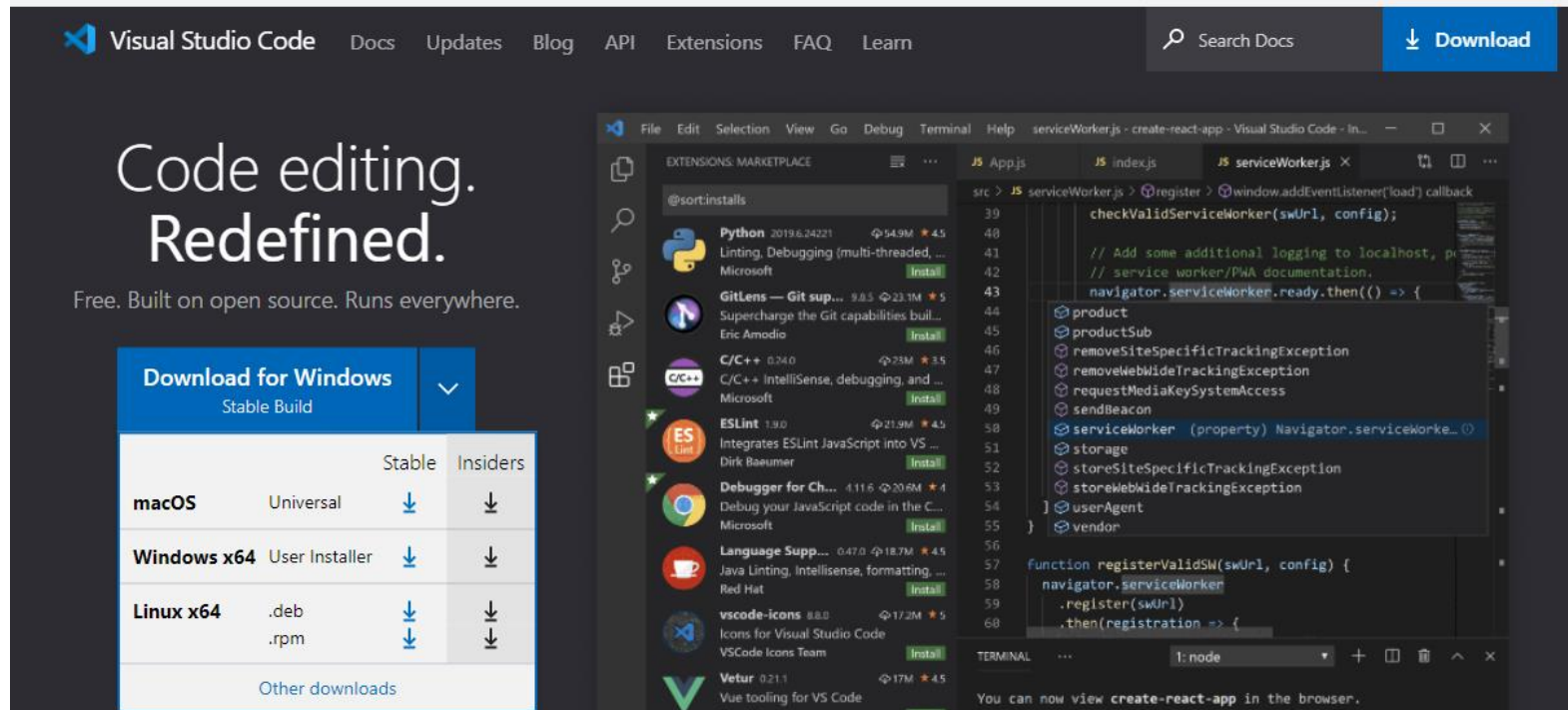
The image shows a Windows command prompt window titled "Administrador: C:\WINDOWS\system32\cmd.exe". The command prompt shows the command `python -m http.server 8000 --bind 127.0.0.1` being executed, with the output "Serving HTTP on 127.0.0.1 port 8000 (http://127.0.0.1:8000/) ...". Below the command prompt, a web browser window is shown with the address bar displaying "localhost:8000". The browser content shows a directory listing for the root directory, listing files and directories: [assets/](#), [cameras.html](#), [exemploGeral.html](#), [geometrias.html](#), [libs/](#), [luzes.html](#), [materiais.html](#), and [Referencia.txt](#).

## Directory listing for /

- [assets/](#)
- [cameras.html](#)
- [exemploGeral.html](#)
- [geometrias.html](#)
- [libs/](#)
- [luzes.html](#)
- [materiais.html](#)
- [Referencia.txt](#)

- Para iniciar o servidor:
  - `python -m http.server 8000 --bind 127.0.0.1`
  - Se atentando de estar na pasta/diretório do projeto.
- O servidor local será inicializado na porta **8000**.
  - Para acessar o projeto, no navegador utilize a url: <http://localhost:8000/>
- Ou use o script presente na pasta do projeto.

# Servidor local (Alternativa 02)



The image shows the Visual Studio Code website on the left and a screenshot of the Visual Studio Code interface on the right.

**Visual Studio Code Website:**

- Header: Visual Studio Code, Docs, Updates, Blog, API, Extensions, FAQ, Learn, Search Docs, Download.
- Text: Code editing. Redefined. Free. Built on open source. Runs everywhere.
- Download for Windows (Stable Build):

		Stable	Insiders
macOS	Universal	↓	↓
Windows x64	User Installer	↓	↓
Linux x64	.deb .rpm	↓ ↓	↓ ↓
- Other downloads

**Visual Studio Code Interface:**

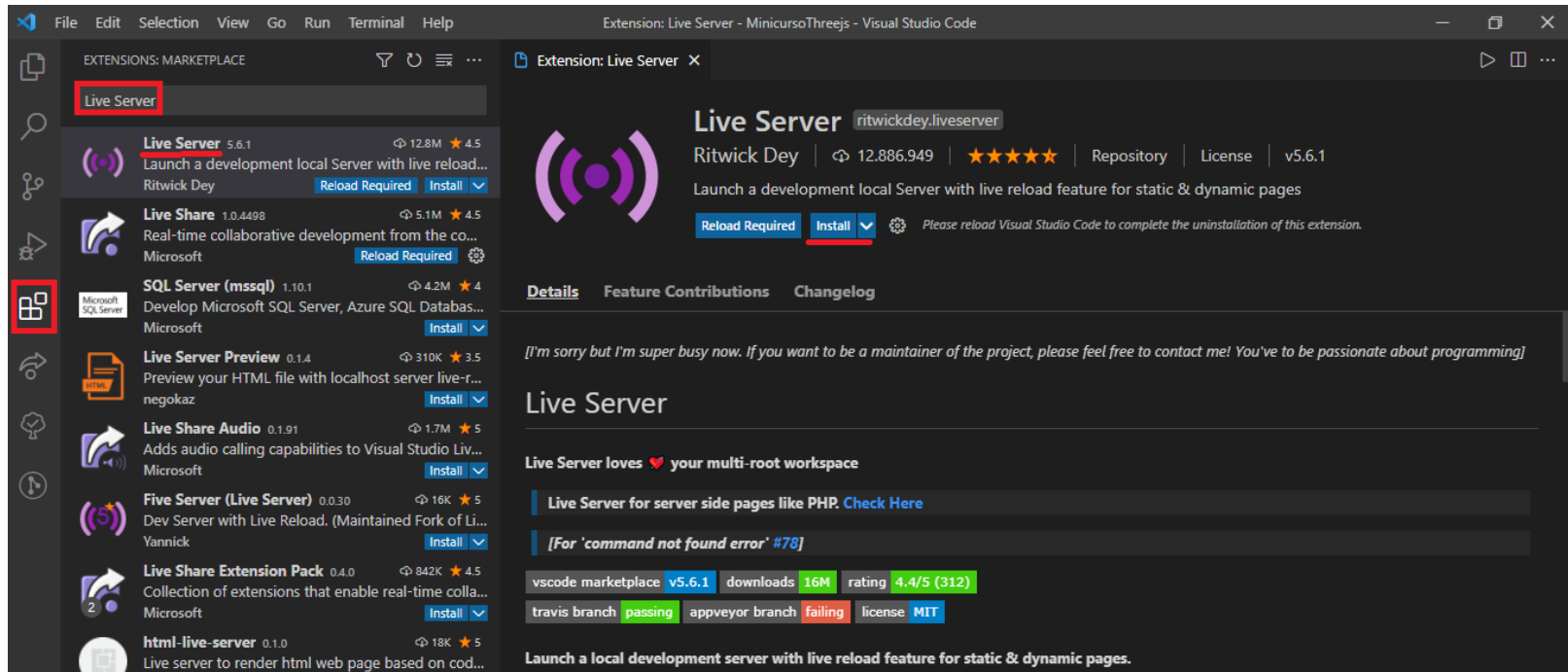
- File Edit Selection View Go Debug Terminal Help
- EXTENSIONS: MARKETPLACE: @sortinstalls, Python, GitLens, C/C++, ESLint, Debugger for Chrome, Language Support, vs-code-icons, Vetur.
- JS App.js: 

```
src > JS serviceWorker.js > register > window.addEventListener('load') callback
39
40
41 // Add some additional logging to localhost, p
42 // service worker/PWA documentation.
43 navigator.serviceWorker.ready.then(() => {
44
45   product
46   productSub
47   removeSiteSpecificTrackingException
48   removeWebWideTrackingException
49   requestMediaKeySystemAccess
50   sendBeacon
51   serviceWorker (property) Navigator.serviceWorke...
52   storage
53   storeSiteSpecificTrackingException
54   storeWebWideTrackingException
55   userAgent
56   vendor
57
58 function registerValidSW(swUrl, config) {
59   navigator.serviceWorker
60     .register(swUrl)
61     .then(registration => {
```
- TERMINAL: 1: node, You can now view create-react-app in the browser.

- Site de download: <https://code.visualstudio.com/>

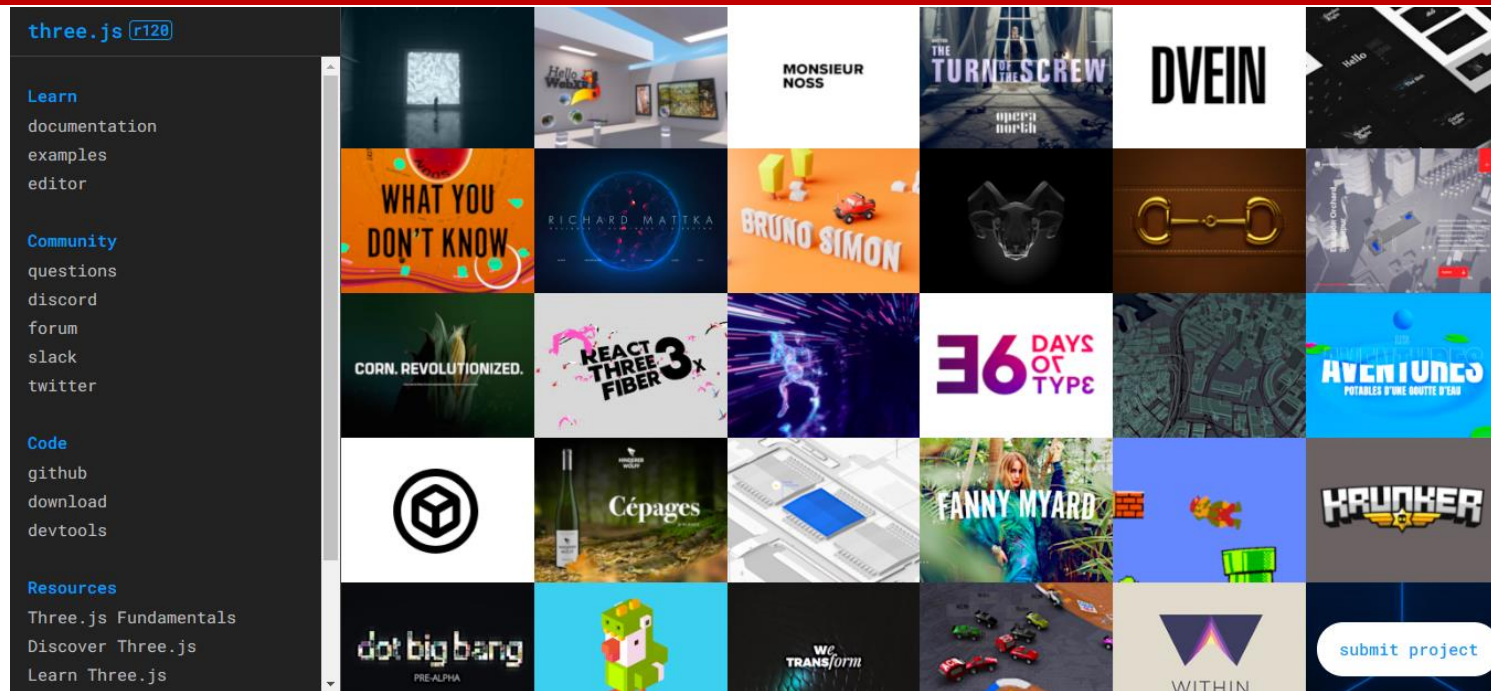


# Servidor local (Alternativa 02)



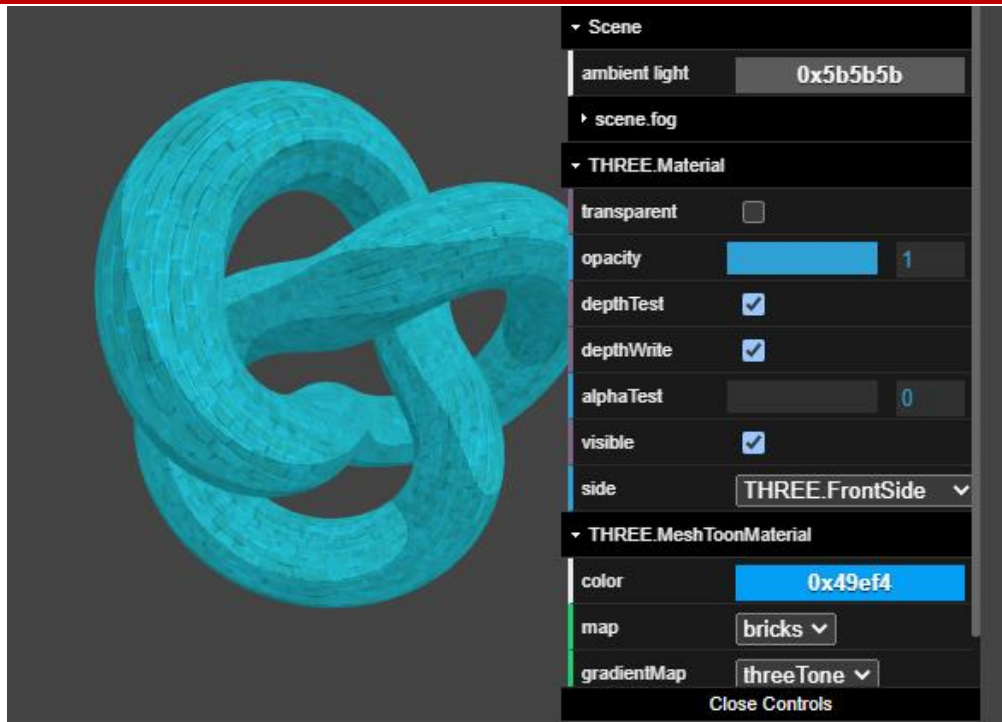
- Instalação da extensão Live Server

# Three.js



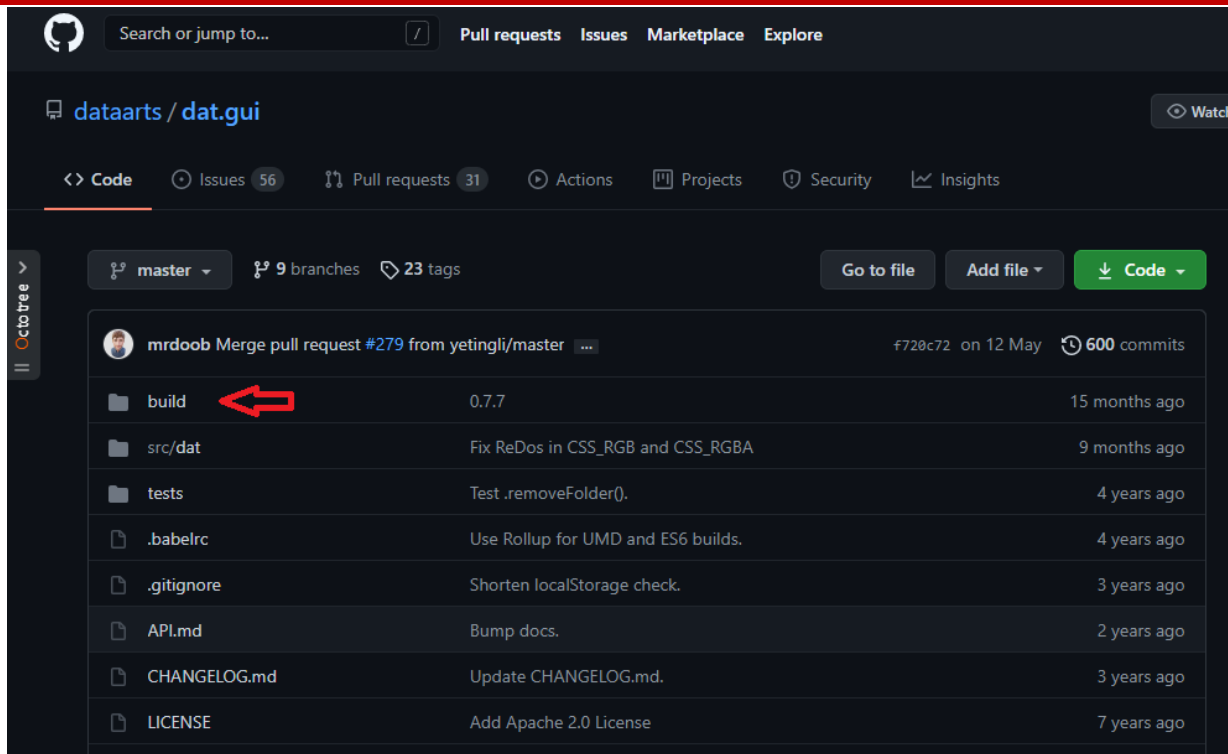
- Fonte: <https://threejs.org/>

# Dat.gui.js



- Fonte: <https://github.com/dataarts/dat.gui>

# Dat.gui.js

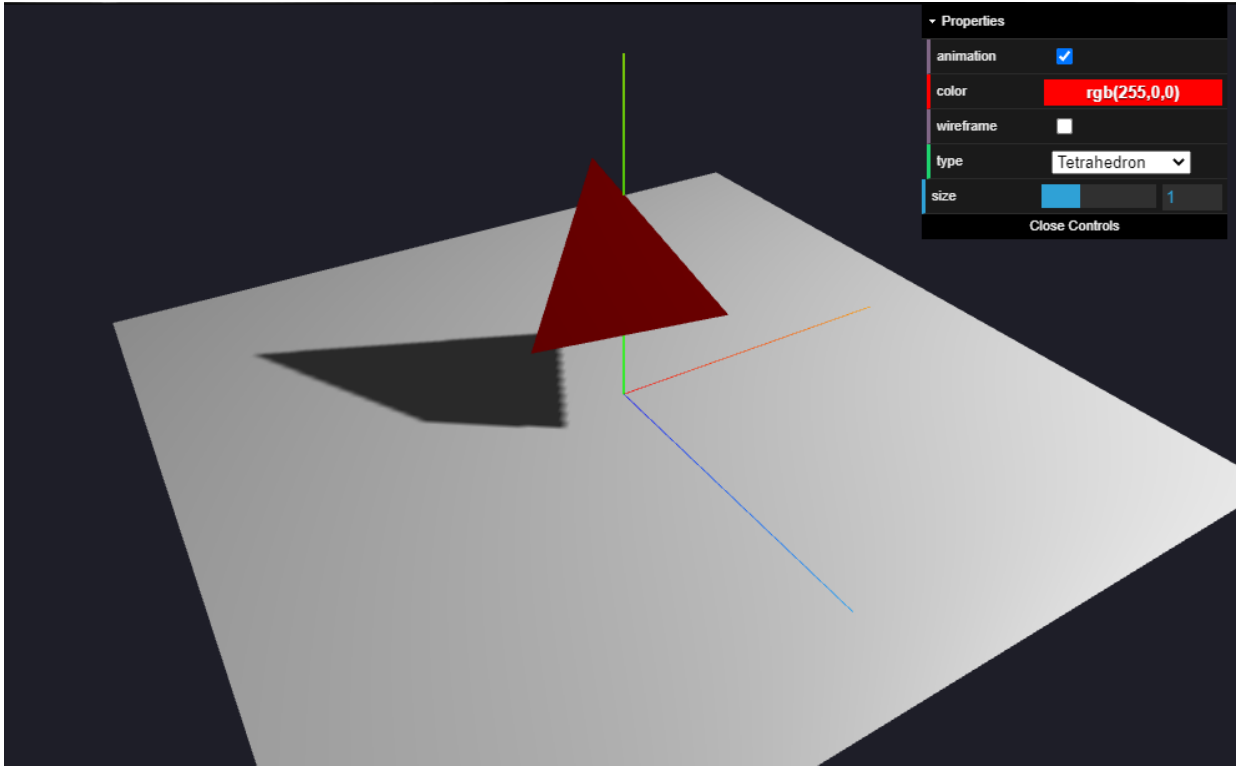


- Fonte: <https://github.com/dataarts/dat.gui>

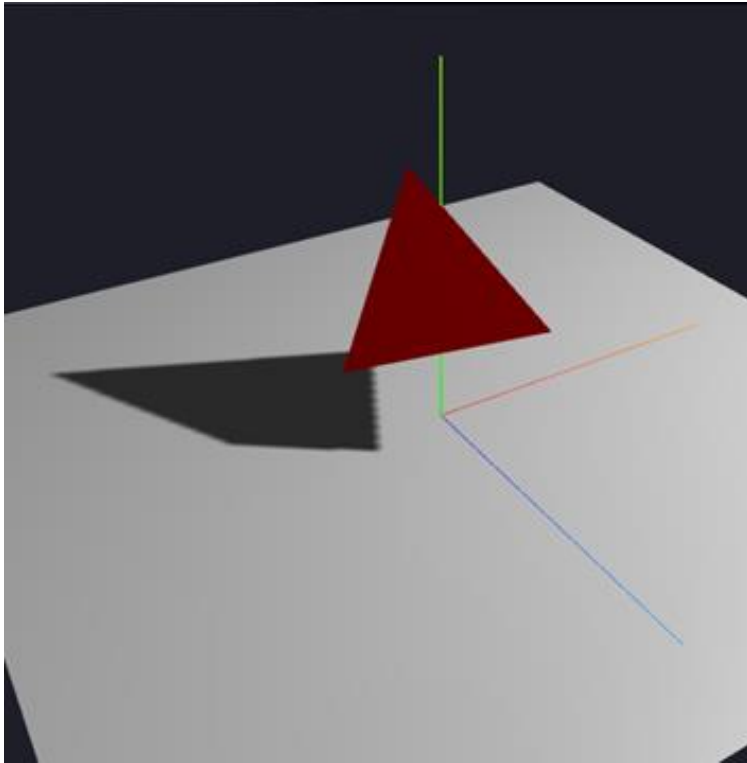
---

# Aplicação básica em THREE.js

# Requisitos base

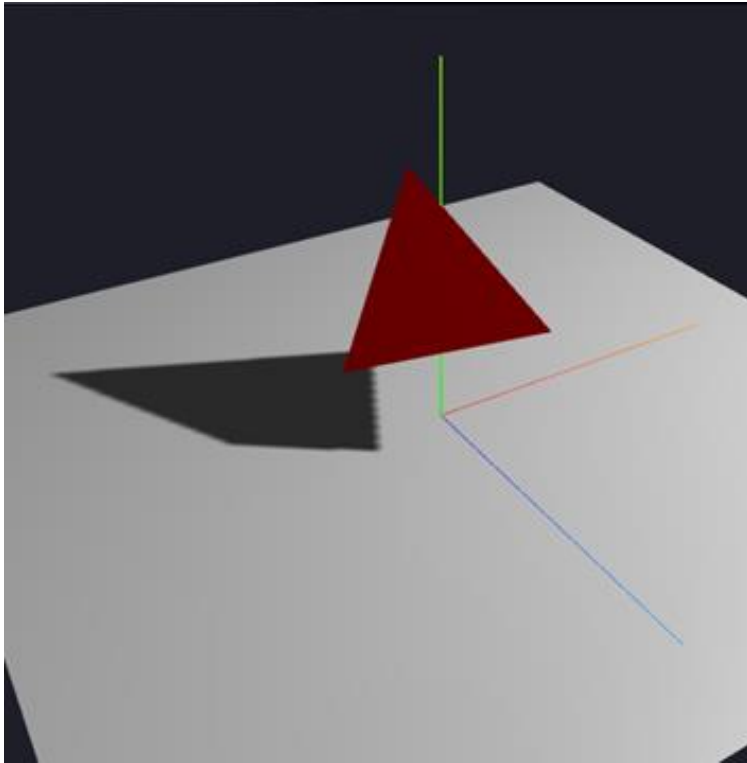


# Requisitos base



- **Scene (Cena):** Armazena toda a estrutura da aplicação;

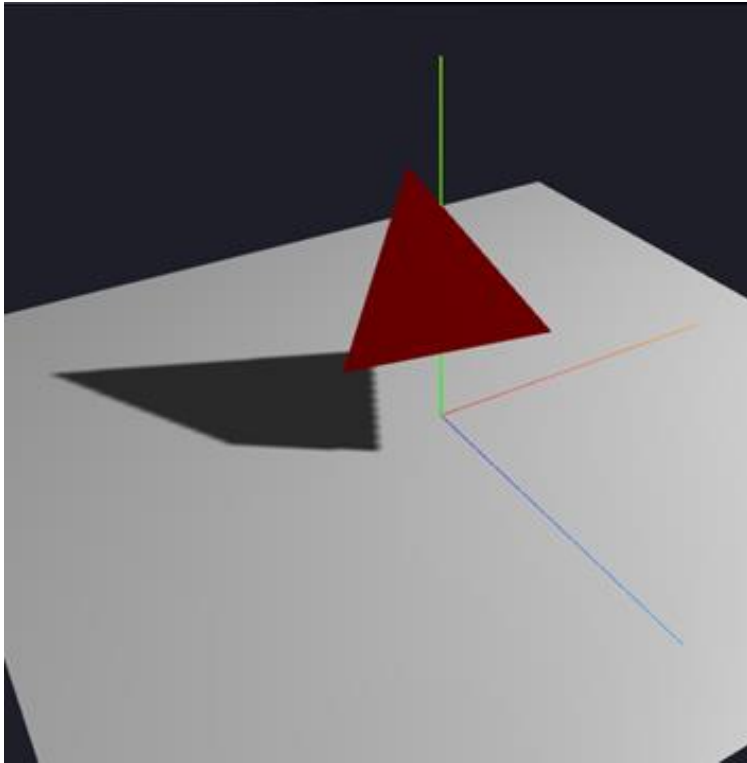
# Requisitos base



- **Scene (Cena):** Armazena toda a estrutura da aplicação;
- **Renderer (Renderizador):** Desenha e processa toda a informação;

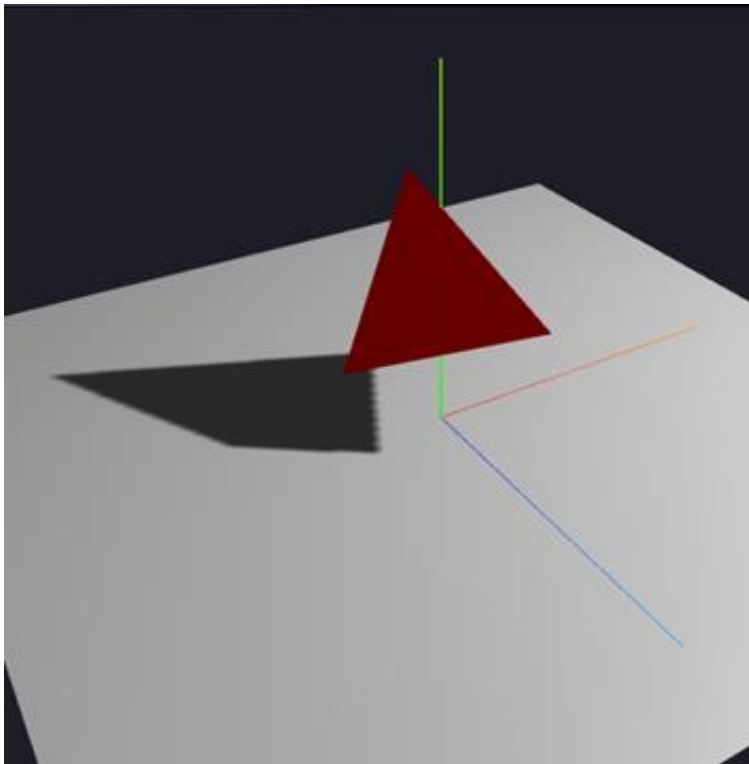


# Requisitos base



- **Scene (Cena):** Armazena toda a estrutura da aplicação;
- **Renderer (Renderizador):** Desenha e processa toda a informação;
- **Camera:** Possibilita a visualização do cenário;

# Requisitos base



- **Scene (Cena):** Armazena toda a estrutura da aplicação;
- **Renderer (Renderizador):** Desenha e processa toda a informação;
- **Camera:** Possibilita a visualização do cenário;
- **Looping de Animação:** Cria o movimento e continuidade da cena;

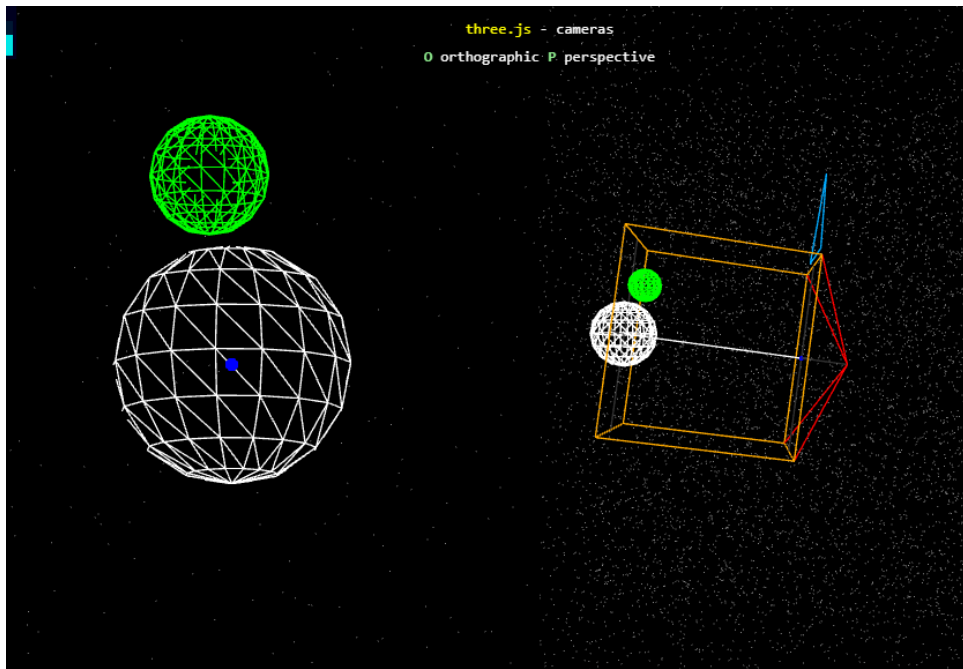
---

# Elementos auxiliares para a construção da cena

---

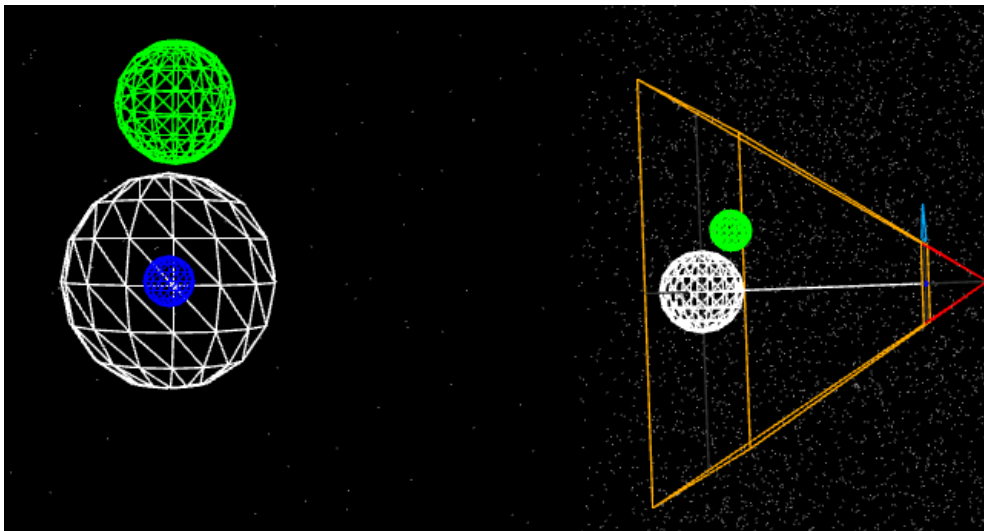
# Câmeras

# OrthographicCamera



- O tamanho de um objeto na imagem renderizada permanece **constante**, independentemente de sua distância da câmera.
- Ideal para fazer elementos 2D na interface do usuário.
- Fonte:  
<https://threejs.org/docs/?q=camera#api/en/cameras/OrthographicCamera>
- [https://threejs.org/examples/#webgl\\_camera](https://threejs.org/examples/#webgl_camera)

# PerspectiveCamera

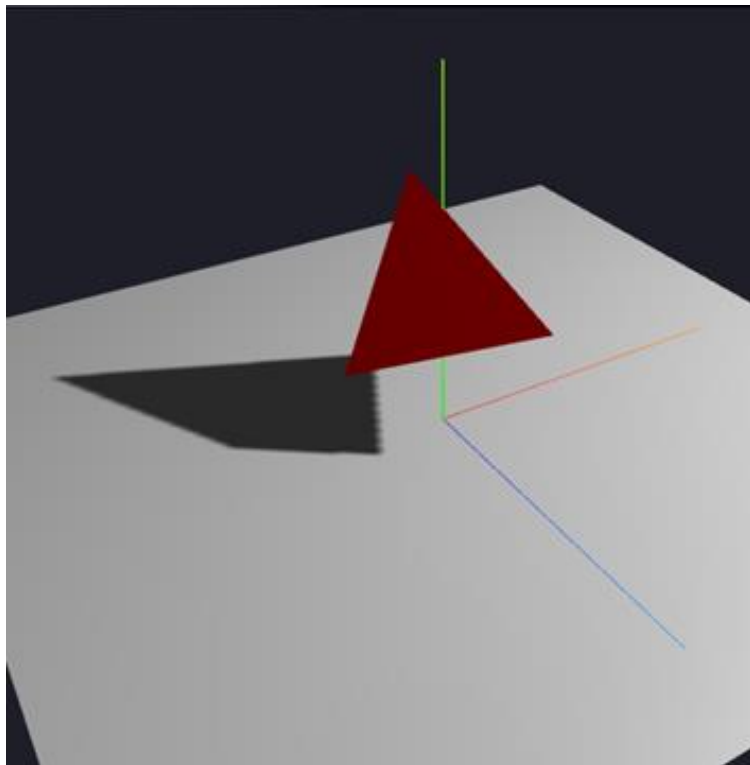


- É a câmera mais usada para o ambiente 3D
- Representa o olho humano e a distorção na imagem
- Fonte:  
<https://threejs.org/docs/?q=camera#api/en/cameras/PerspectiveCamera>
- [https://threejs.org/examples/#webgl\\_camera](https://threejs.org/examples/#webgl_camera)

---

# Luzes

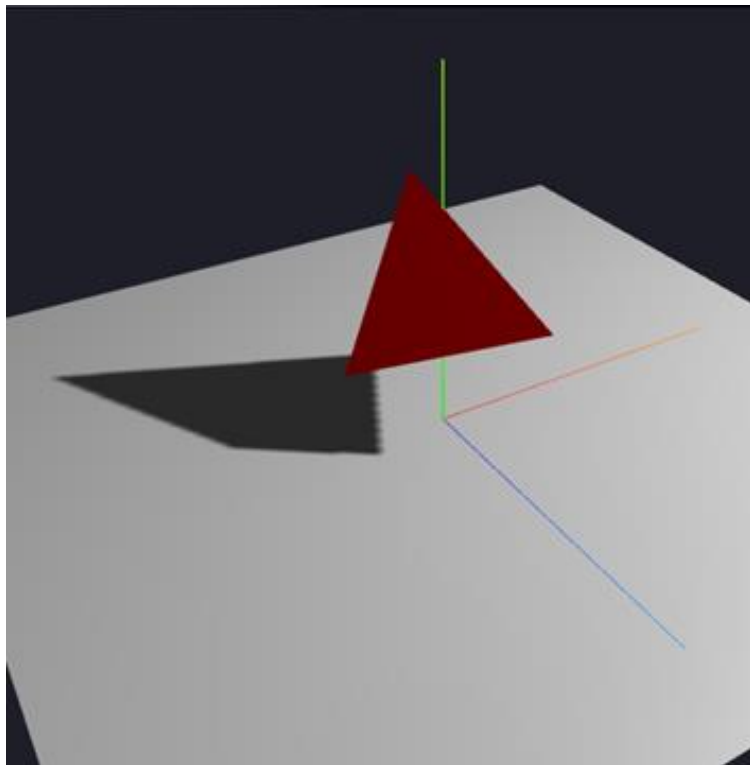
# AmbientLight



- Ilumina todos os objetos igualmente
- **Não** faz sombra e não tem direção
- Fonte:
- <https://threejs.org/docs/?q=light#api/en/lights/AmbientLight>

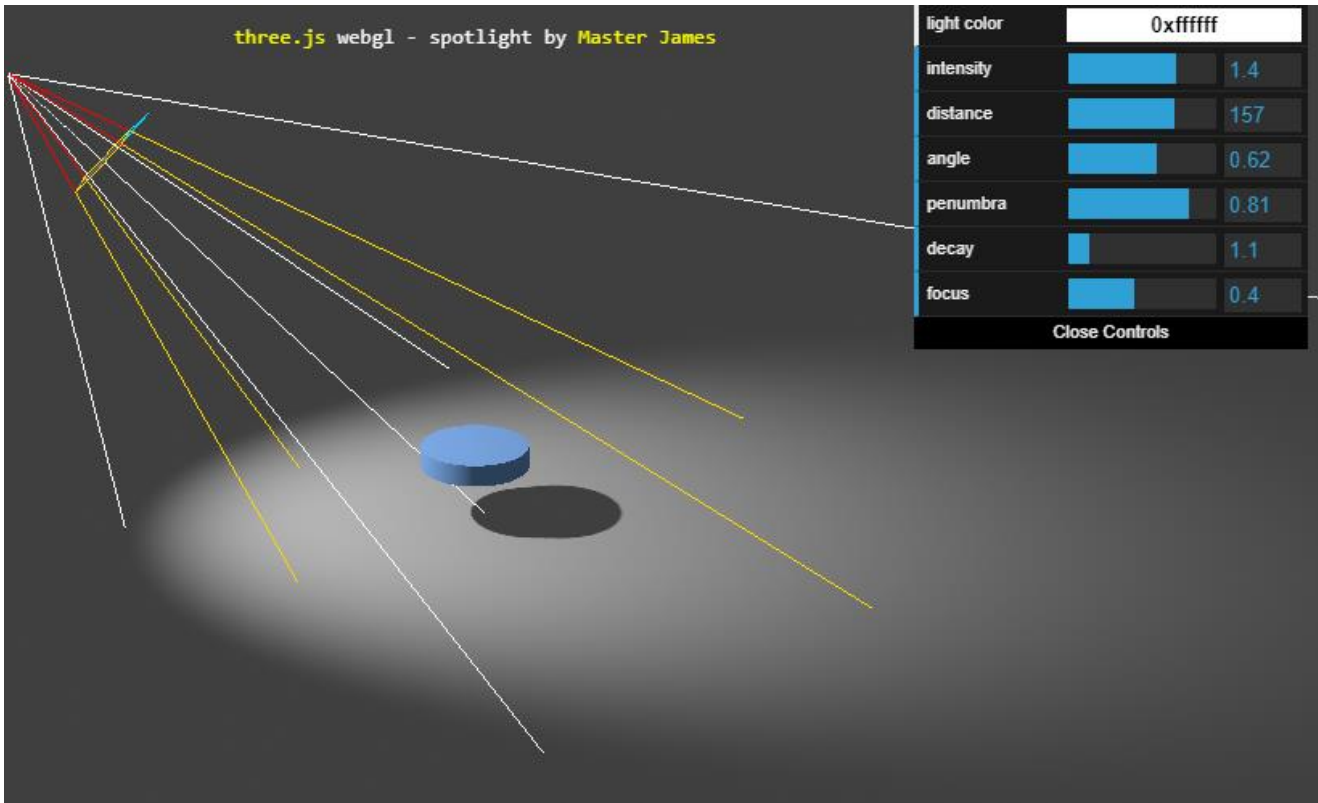


# SpotLight



- Ilumina em uma direção
- Faz sombra e apresenta uma angulação de emissão
- Fonte:
- [https://threejs.org/examples/#webgl\\_lights\\_spotlight](https://threejs.org/examples/#webgl_lights_spotlight)
- [https://threejs.org/examples/#webgl\\_lights\\_spotlight](https://threejs.org/examples/#webgl_lights_spotlight)

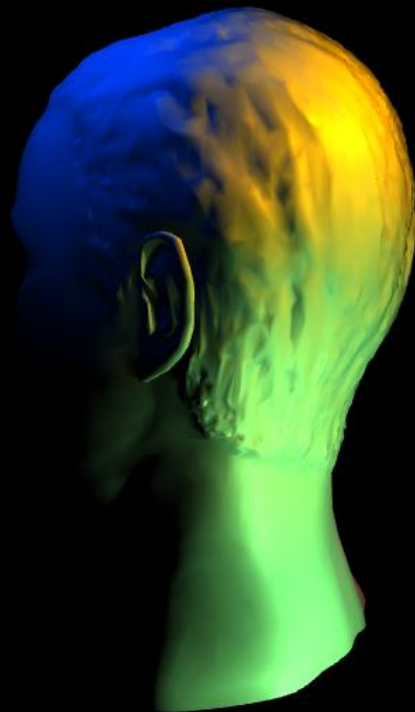
# SpotLight



# PointLight

three.js - point lights WebGL demo.

Walt Disney head by David O'Reilly



- Ilumina em todas as direções tal como uma lâmpada
- Fonte:
- <https://threejs.org/docs/?q=light#api/en/lights/PointLight>
- [https://threejs.org/examples/#webgl\\_lights\\_pointlights](https://threejs.org/examples/#webgl_lights_pointlights)

---

# Materials

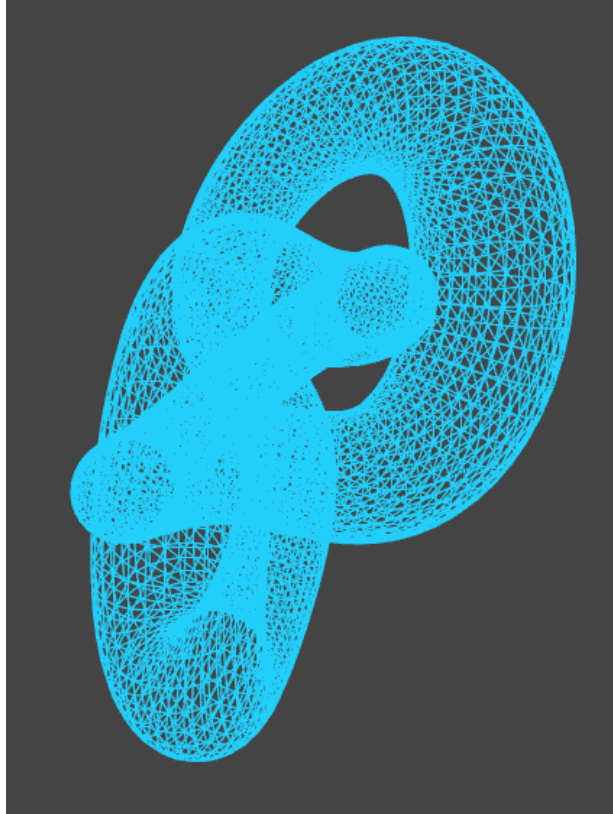
# MeshBasicMaterial



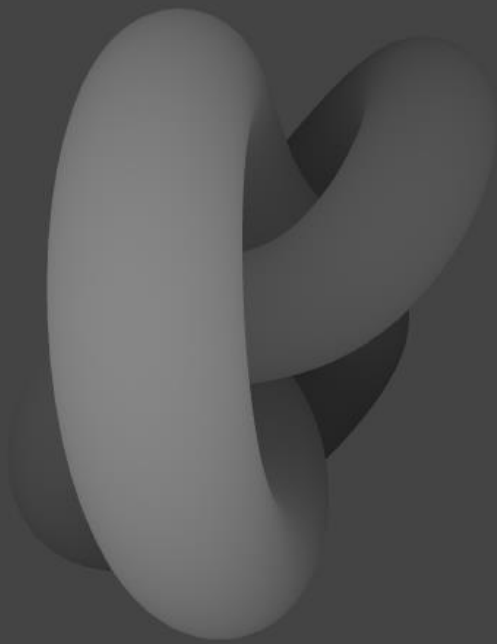
- A luz não afeta o material !!!
- Pode ter uma distorção e não conseguir diferenciar as faces do objeto

- Fonte: <https://threejs.org/docs/scenes/material-browser.html#MeshBasicMaterial>

# \*opção Wireframe

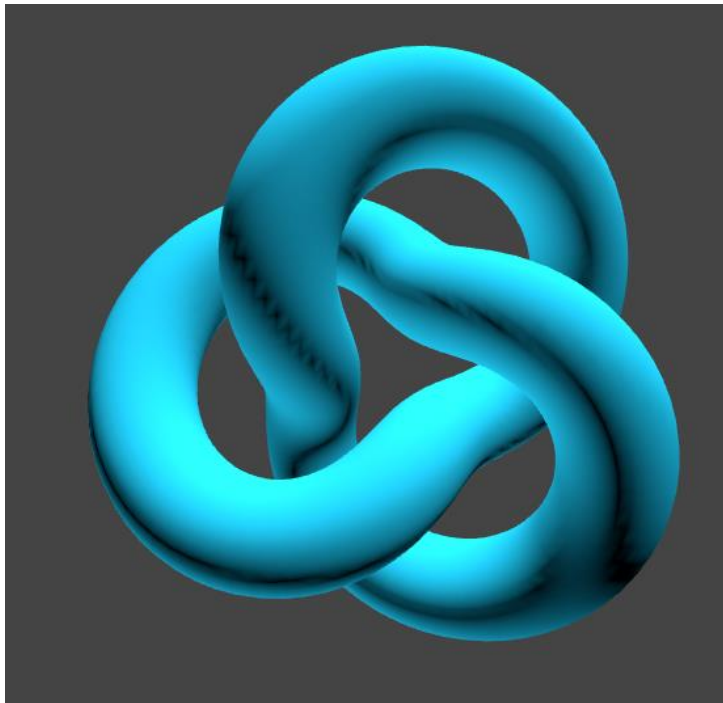


# MeshDepthMaterial



- Usado para materiais que apresentam **profundidade**.
- Quanto mais próximo da câmera, mais claro será a cor.
- Quanto mais afastado da câmera, mais escuro será.

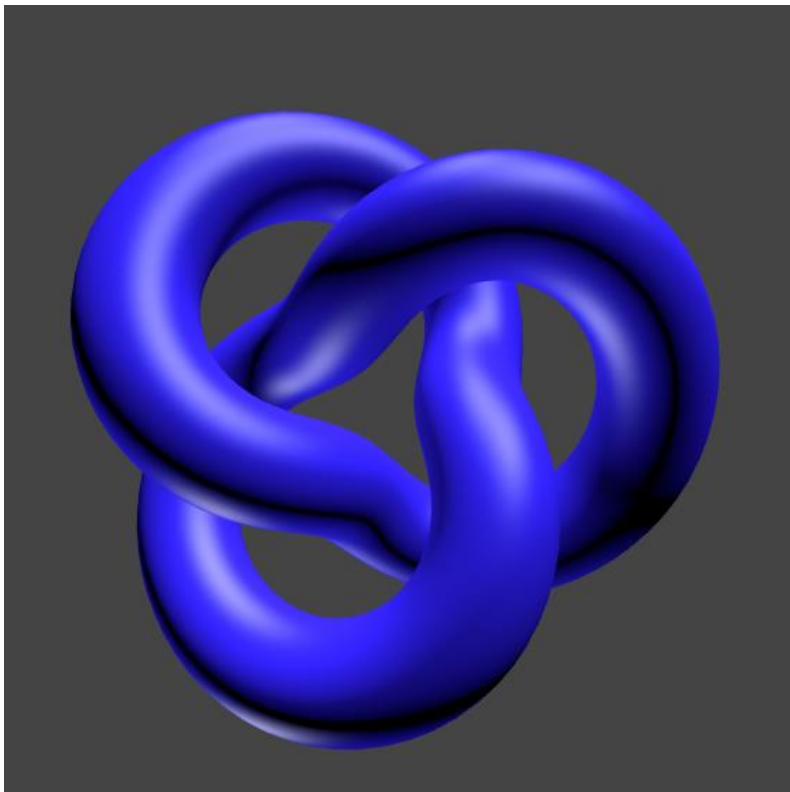
# MeshLambertMaterial



- Apresenta uma profundidade no material.
- Ideal para simular superfícies mais opacas como madeira ou rocha
- Não pode simular superfícies com muito brilho.



# MeshPhongMaterial



- Apresenta o coeficiente especular.
- Logo, o material é indicado para superfícies brilhosas.
- A reflexão de luz pode impactar no desempenho da aplicação.

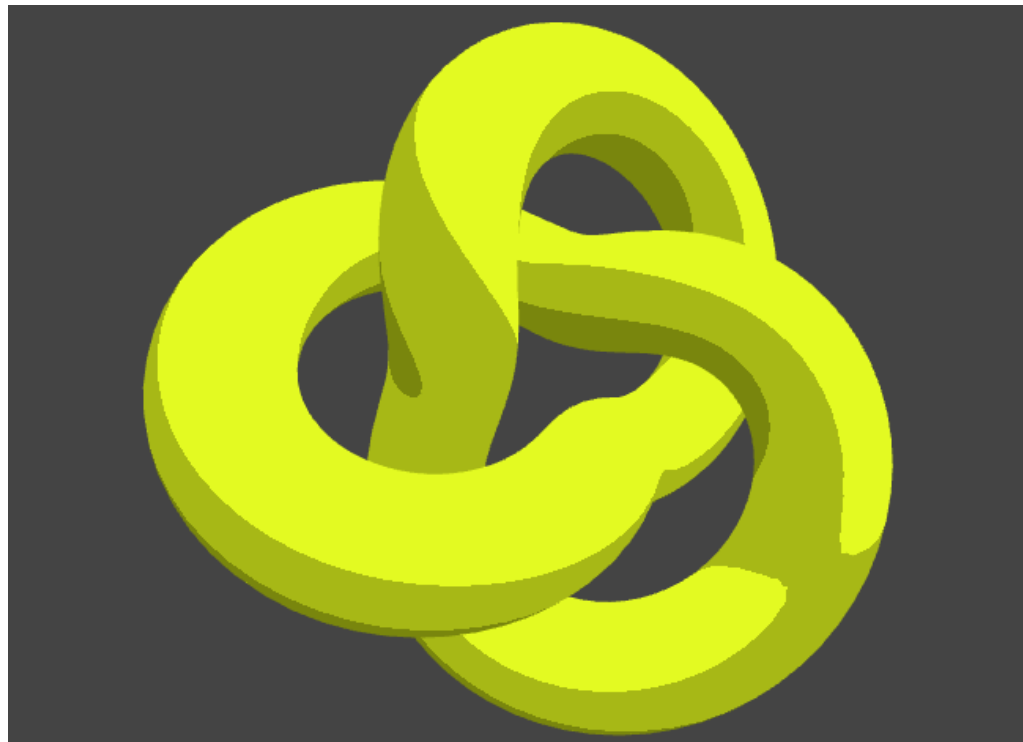
# MeshStandardMaterial



- Apresenta o coeficiente especular.
- Logo, o material é indicado para superfícies brilhosas.
- A reflexão de luz pode impactar no desempenho da aplicação.
- Aparência mais realista em comparação ao **MeshLambertMaterial** e ao **MeshPhongMaterial**.

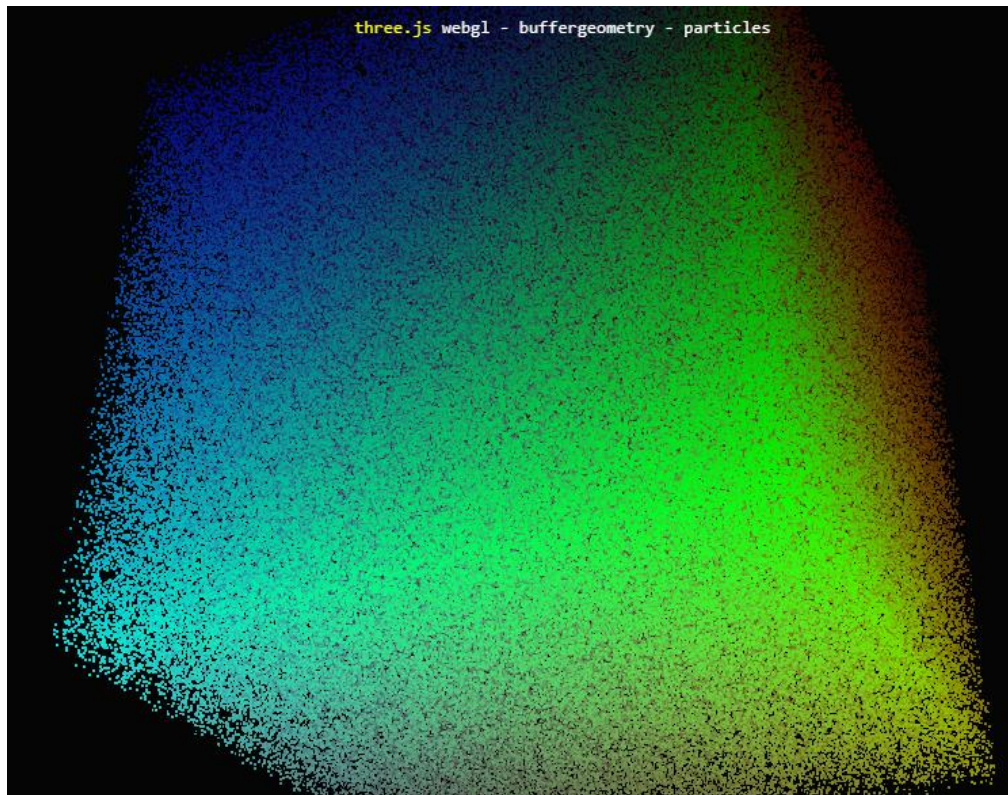
# MeshToonMaterial

---



- Material que deixa um aspecto de desenho animado e cartunesco.

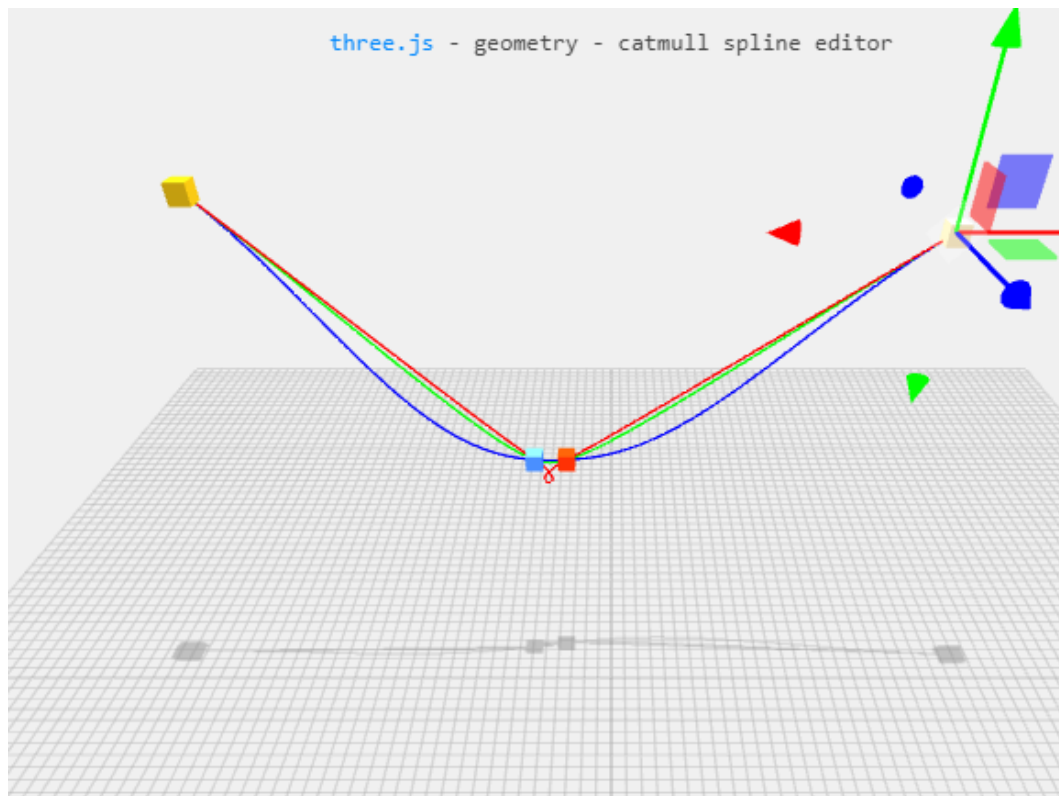
# MeshPointsMaterial



- Material voltado para a classe de pontos.
- Fonte:

[https://threejs.org/examples/#webgl\\_buffergeometry\\_points](https://threejs.org/examples/#webgl_buffergeometry_points)

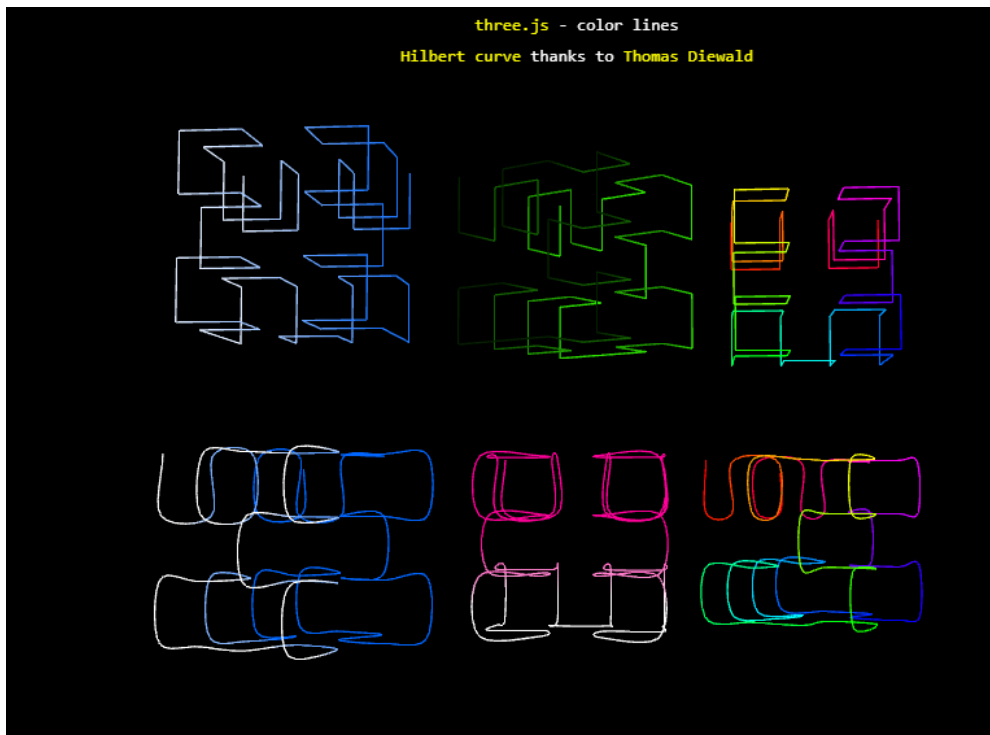
# ShadowMaterial



- Material recebe sombras de outros objetos mas ele mesmo é Transparente
- Fonte:

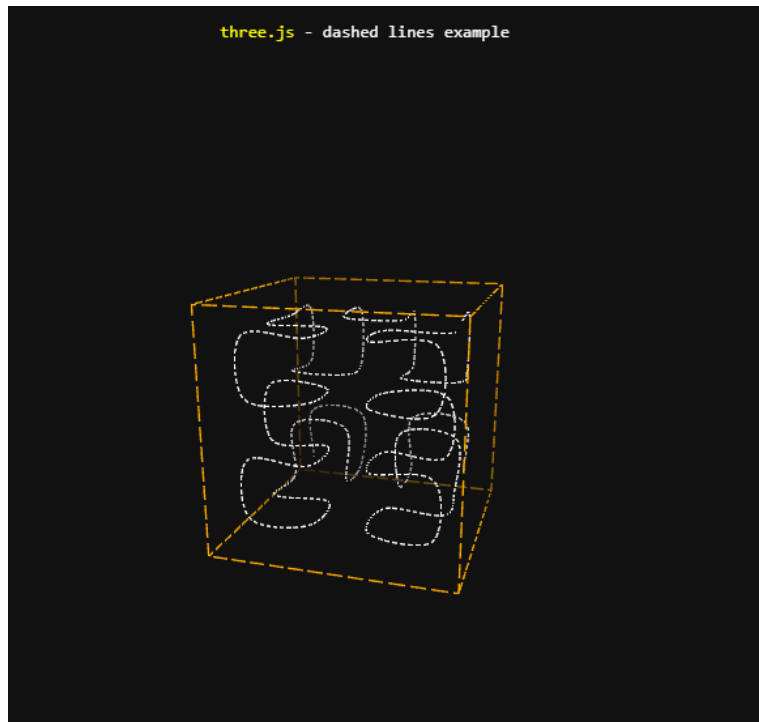
[https://threejs.org/examples/#webgl\\_geometry\\_spline\\_editor](https://threejs.org/examples/#webgl_geometry_spline_editor)

# LineBasicMaterial



- Fonte: [https://threejs.org/examples/#webgl\\_lines\\_colors](https://threejs.org/examples/#webgl_lines_colors)

# LineBasicMaterial



- Fonte: [https://threejs.org/examples/#webgl\\_lines\\_colors](https://threejs.org/examples/#webgl_lines_colors)

---

# Geometrias



---

# Aplicação do Sistema Solar

---

# Outros recursos presentes na Three.js e bibliotecas auxiliares

# Bibliotecas Auxiliares

---

- **Stats.js** – Exibe o FPS na aplicação (Opcional);
- **dat.gui.js** – Exibe um menu para o controle da aplicação e melhoria de interação do usuário com a aplicação;
- Bibliotecas de carregamento de objetos 3D
- **Physijs** – Trabalhar com física em objetos;
- **Oimo.js** – Trabalhar com física em objetos;
- **AR.js** – Trabalhar com realidade aumentada com Three.js;
- **WEBXR** – Trabalhar com realidade virtual no Three.js;
- **AFRAME** - Trabalhar com realidade virtual em conjunto ao Three.js;