

Usabilidade, desenvolvimento web, mobile e jogos

Dr. Richard Henrique de Souza [Quinta]

Dr. Saulo Popov Zambiasi [Sexta - Tuma B]

Dr. Richard Henrique de Souza [Sexta - Tuma A]

Cronograma - Resumo

■ Avaliações



1a Avaliação Discursiva(A1) - 08 a 09/05

↓ 24/05 2a oportunidade

□ 2a Avaliação Objetiva (A2) - 11 a 12/06

↓ 21/06 2a oportunidade

□ 3a Avaliação Trabalho (A3) - 14/06

■ Término Regular - 28/06

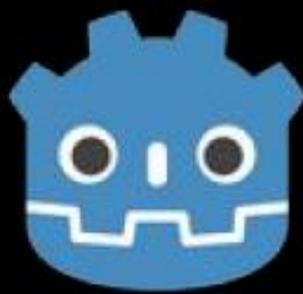


A3

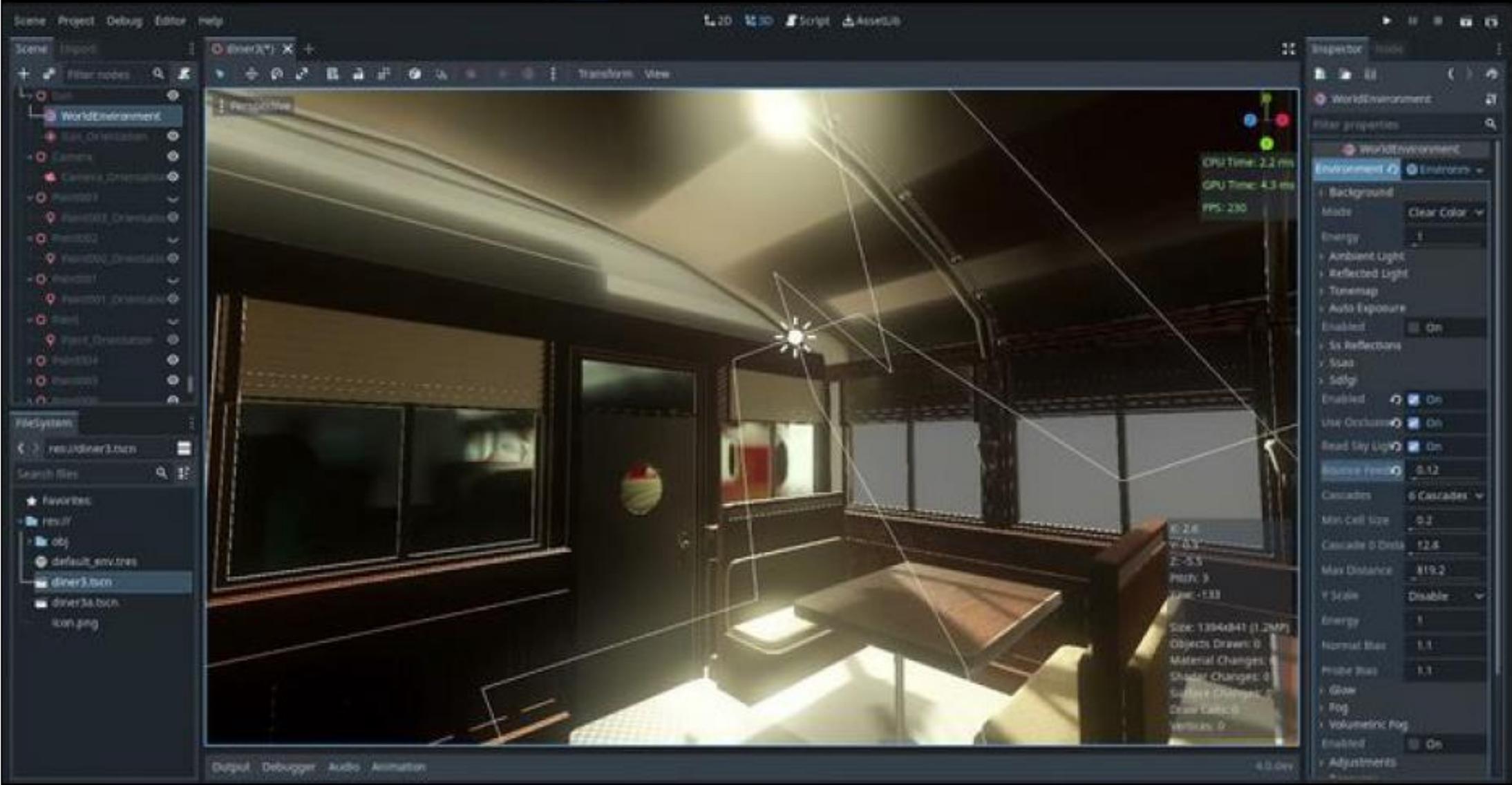
- ▶ **Nessa avaliação deve ser escolhido um dos seguintes temas para se desenvolver um projeto:**
 - ▶ **Desenvolvimento para Web: Html, javascript, bootstrap e um framework como ReactJS ou AngularJS, ou Vue.JS ou Flutter**
 - ▶ **Desenvolvimento para Dispositivos Móveis: Flutter ou React Native**
 - ▶ **Desenvolvimento de Jogos Digitais: Usando um motor de jogo como Godot, Unity 3d, Unreal, etc.**
- ▶ **A atividade pode ser desenvolvida em grupo de até 5 integrantes.**
 - ▶ **Veja que, apesar de terem sido escolhidos algumas tecnologias de implementação para serem apresentadas em nossas aulas. Vocês estão livres para utilizarem outras correlatas, tal como citado acima.**

Vídeo Aula

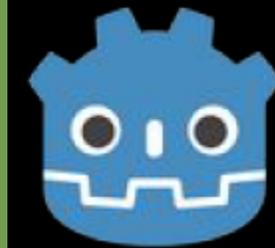
- ▶ https://animaeducacao.zoom.us/rec/play/VovU8CJES4pHtTmnlCO2_yKDVnxUsTu60fJkeJP1j7fvkJF2Vl0I_TAnP-q7Au4kAaioEmqHrzThY_FG.DRT3vUed_L0EqE-N?canPlayFromShare=true&from=share_recording_detail&continueMode=true&componentName=rec-play&originRequestUrl=https%3A%2F%2Fanimaeducacao.zoom.us%2Frec%2Fshare%2FiXVHlpiLvDFcC5mbCkvdt0Wq4vqBE0dEwLuwK1FIZd09H95CbBmP9_szk3gL_7c.2q5_n6-MgkqUzs2-



GODOT



GODOT



- ▶ Motor de jogo de código aberto
 - ▶ Desenvolvimento iniciado por Juan Linietsky e Ariel Manzur em conjunto com a OKAM Studio em 2001 na Argentina.
 - ▶ Fevereiro de 2014: código fonte foi publicado no GitHub sob a licença MIT
 - ▶ 15 de dezembro de 2014: Primeira versão estável
 - ▶ Atualmente com a versão 4...
 - ▶ Visa fornecer um ambiente de desenvolvimento de jogos totalmente integrado
 - ▶ Apenas um binário de execução bastante leve

GODOT



- ▶ Permite que os desenvolvedores criem um jogo, sem precisar de outras ferramentas
 - ▶ além das usadas para a criação de conteúdo (recursos visuais, música, etc.)
- ▶ A arquitetura do motor é construída em torno do conceito de uma árvore de "nós".
 - ▶ Os nós são organizados dentro de "cenas", que são grupos de nós reutilizáveis, instáveis, herdáveis e aninhados.
 - ▶ Todos os recursos do jogo, incluindo scripts e ativos gráficos, são salvos como parte do sistema de arquivos do computador
 - ▶ Esta solução de armazenamento visa facilitar a colaboração entre equipes de desenvolvimento de jogos usando sistemas de controle de versão de software

GODOT



- ▶ Trabalha com quatro tipos de linguagens:

- ▶ GDScript

- ▶ Linguagem principal do programa.

- ▶ Sintaxe semelhante ao Python.

- ▶ C#

- ▶ Adicionado na versão 3.0.

- ▶ Visa atender o padrão da indústria.

- ▶ VisualScript:

- ▶ Programação via interface gráfica para pessoas que não sabem programar ou programadores que desejam expor comportamentos do jogo para projetistas, artistas, etc.

- ▶ GDNative:

- ▶ Semelhante ao C++.

- ▶ utilizada para modificação da própria engine, criar extensões e novas funções.

GODOT



- ▶ Permite a especificação de compressão de textura e configurações de resolução para cada plataforma.
- ▶ Executa no Linux, macOS e Microsoft Windows, mas permite outras plataformas.
- ▶ Oferece suporte a games para multi plataformas:
 - ▶ Mobile Android, iOS
 - ▶ Desktop Linux, macOS, Microsoft Windows, BSD
 - ▶ Web HTML5, WebAssembly
 - ▶ Realidade virtual / estendida HTC Vive, Valve Index, Oculus Rift, Oculus Go, Oculus Quest, todos os fones de ouvido Microsoft MR, ARKit da Apple, etc.

GODOT



- ▶ Sistema de animação sofisticado com a capacidade de
 - ▶ esqueletos (skeletons)
 - ▶ inversão cinética (inverse kinematic)
 - ▶ deformação (morphing).
- ▶ Sistema de física integrado para 2D e 3D que suporta
 - ▶ colisões
 - ▶ objetos rígidos
 - ▶ cinética de corpos, etc.



► <https://wccfttech.com/steam-deck-wont-have-any-exclusive-games-valve-confirms/>

Steam Deck Won't Have Any Exclusive Games, Valve Confirms

Alessio Palumbo · Dec 1, 2021 08:00 AM EST · Copy Shortlink

57

ADVERTISEMENT

Ad



GODOT



Vídeo Comparativo

- ▶ **Unity VS Godot: How Do They Compare?**
 - ▶ <https://www.youtube.com/watch?v=kahPeM0scpM>

Vídeos

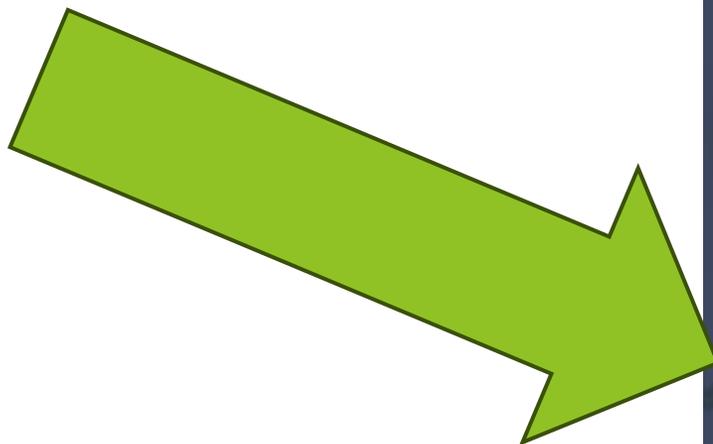
- ▶ [The Godot 2022 Showreels](#)

- ▶ **<https://godotengine.org/article/announcing-godot-2022-showreels/>**

GODOT

▶ Baixe o GODOT

▶ <https://godotengine.org/>

The banner features a dark blue background with a pattern of small, colorful, cartoonish alien-like creatures. The main text is in a large, white, sans-serif font. Below the main text is a smaller line of white text. At the bottom, there is a white button with the text 'Download Latest' and a blue button with the version number '4.2.2'.

**Your free, open-source
game engine.**

Develop your 2D & 3D games, cross-platform projects, or even XR ideas!

Download Latest 4.2.2

GODOT

Download Godot 4 for Windows



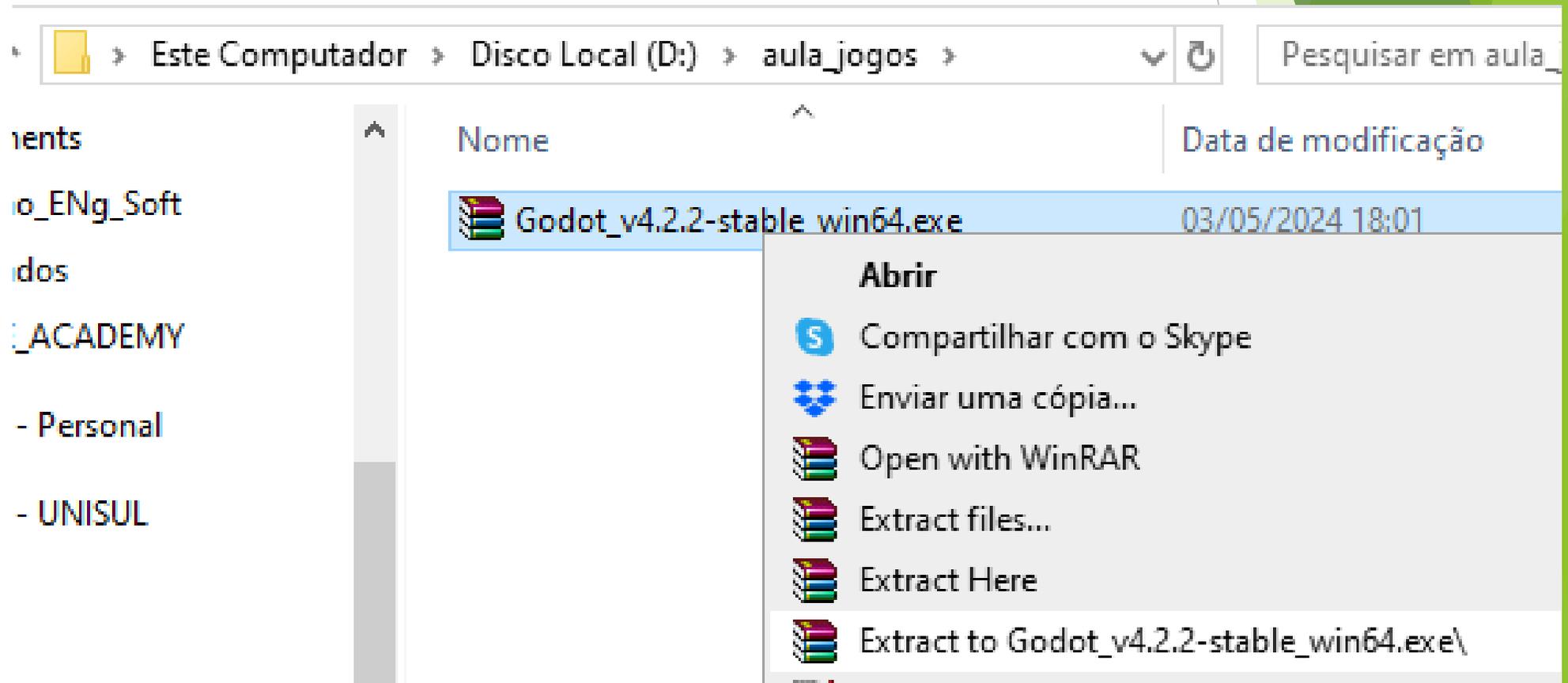
Godot Engine

4.2.2

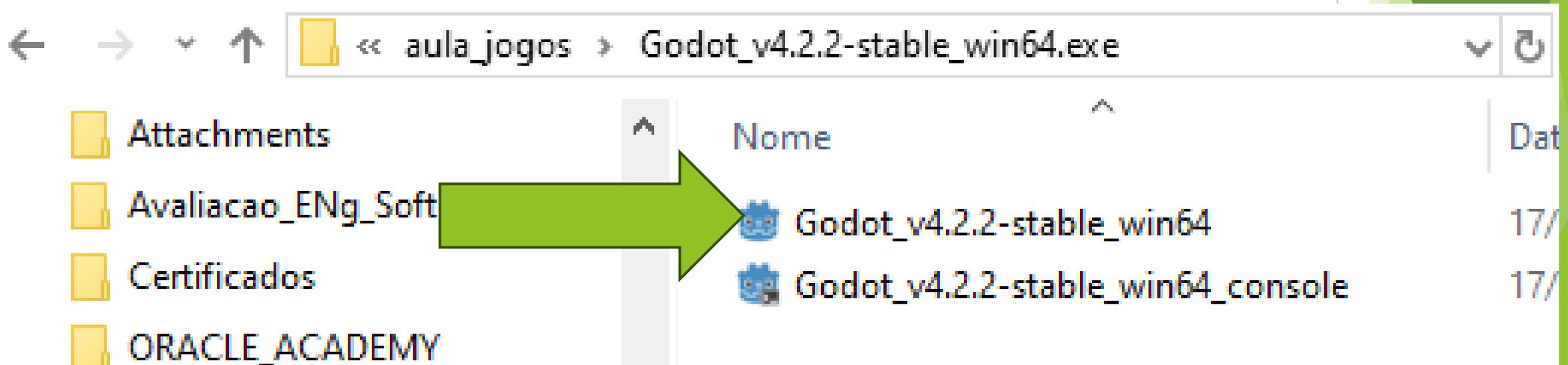
64 bit - 17 April 2024

GODOT

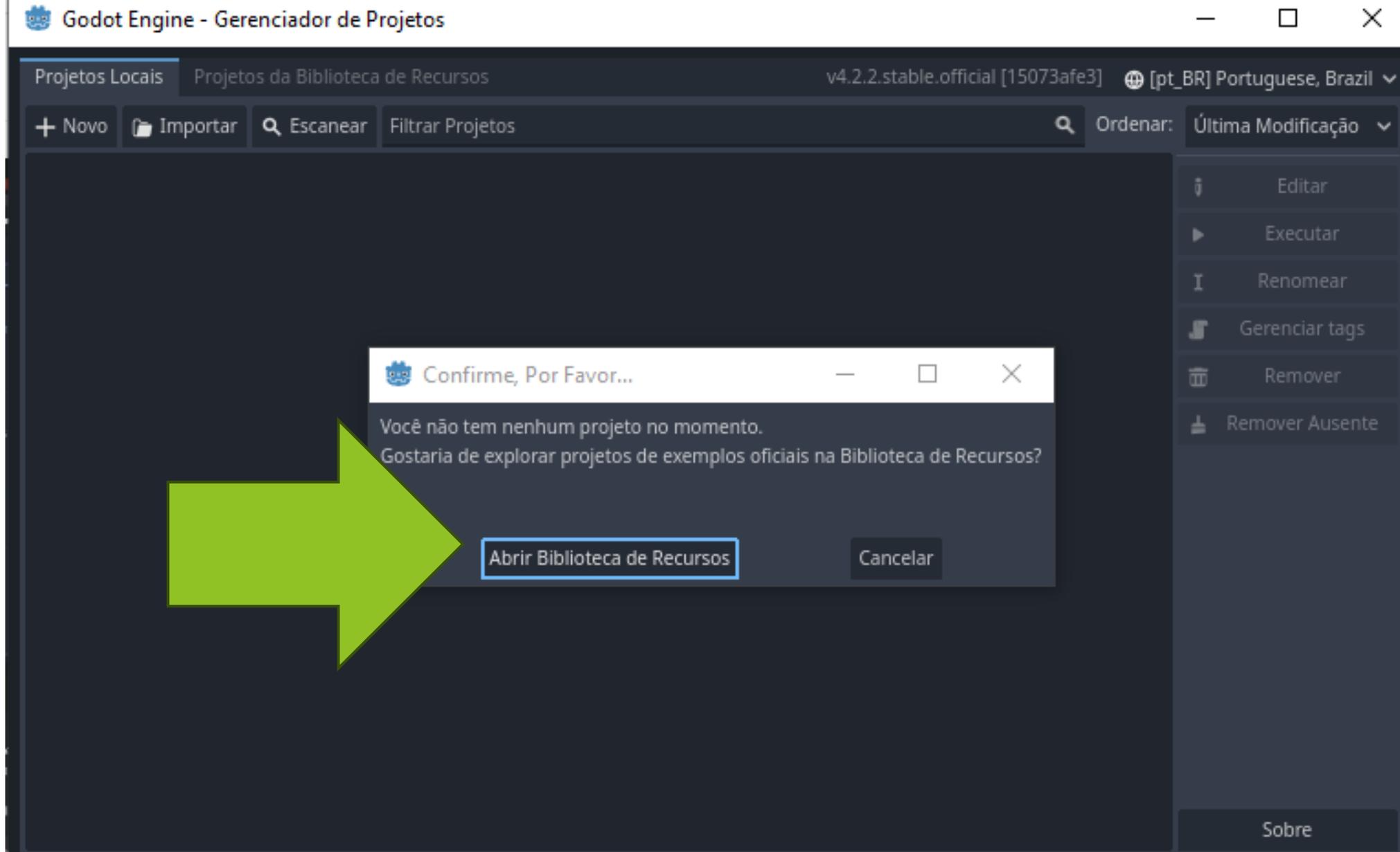
- ▶ Descompacte o zip

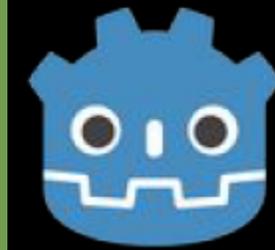


GODOT



GODOT





Godot Engine: Hello Game World!

- ▶ Esse exemplo foi originalmente feito para o Godot 3, mas foi modificada para funcionar no Godot Engine versão 4.
- ▶ Videoaula:
<https://www.youtube.com/watch?v=1wxJ7JTGvaY> (Essa videoaula foi gravada usando o Godot 3).

Pesquisar Modelos, Projetos, e Demonstrações

Ordenar: **Atualizado Recentemente**

Categoria: **Todos**

Site: **godotengine.**

Primeiro Anterior 1 2 3 Próximo Último



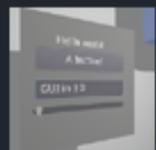
Joypads (Gamepads) Demo

Demos
Godot Engine
MIT



XR Character Body Centric Movement Demo

Demos
Godot Engine
MIT



GUI in 3D Viewport Demo

Demos
Godot Engine
MIT



3D Resolution Scaling Demo

Demos



XR Origin Centric Movement Demo

Demos
Godot Engine
MIT



Screen Capture Demo

Demos
Godot Engine
MIT



Dynamic Split Screen Demo

Demos
Godot Engine
MIT



3D in 2D Viewport demo

Demos

Defina a
linguagem

[pt_BR] Portuguese, Brazil

- [en] English
- [ar] Arabic
- [bg] Bulgarian
- [ca] Catalan
- [cs] Czech
- [de] German
- [el] Greek
- [eo] Esperanto
- [es] Spanish
- [es_AR] Spanish, Argentina
- [et] Estonian
- [fa] Persian
- [fi] Finnish
- [fr] French
- [gl] Galician
- [he] Hebrew
- [hu] Hungarian
- [id] Indonesian

Você pode ver os projetos já prontos

Pesquisar Modelos, Projetos, e Demonstrações

Ordenar: **Atualizado Recentemente**

Categoria: **Todos**

Site: **godotengine.**

Primeiro Anterior 1 2 3 Próximo Último



Joypads (Gamepads) Demo
Demos
Godot Engine
MIT



XR Origin Centric Movement Demo
Demos
Godot Engine
MIT



XR Character Body Centric Movement Demo
Demos
Godot Engine
MIT



Screen Capture Demo
Demos
Godot Engine
MIT



GUI in 3D Viewport Demo
Demos
Godot Engine
MIT



Dynamic Split Screen Demo
Demos
Godot Engine
MIT



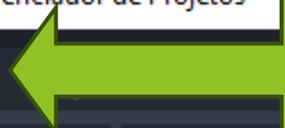
3D Resolution Scaling Demo
Demos



3D in 2D Viewport demo
Demos

- [pt_BR] Portuguese, Brazil
- [en] English
- [ar] Arabic
- [bg] Bulgarian
- [ca] Catalan
- [cs] Czech
- [de] German
- [el] Greek
- [eo] Esperanto
- [es] Spanish
- [es_AR] Spanish, Argentina
- [et] Estonian
- [fa] Persian
- [fi] Finnish
- [fr] French
- [gl] Galician
- [he] Hebrew
- [hu] Hungarian
- [id] Indonesian

Vamos Criar um projeto novo



Projetos Locais

v4.2.2.stable.official [15073afe3] [pt_BR] Portuguese, Brazil

+ Novo

📁 Importar

🔍 Escanear



Ordenar:

Última Modificaçã

- 🔗 Editar
- ▶ Executar
- I Renomear
- 🏷 Gerenciar tags
- 🗑 Remover
- 👤 Remover Ausente

Sobre

Projetos Locais

Projetos da Biblioteca de Recursos

v4.2.2.stable.official [15073afe3] [pt_BR] Portuguese, Brazil

+ Novo

Importar

Escanear

Filtrar Pro



Ordenar:

Última Modificaçã

Novo Projeto (Ctrl+N)

Clicar em Novo
E depois em Novo
Projeto

- Editar
- Executar
- Renomear
- Gerenciar tags
- Remover
- Remover Ausente

Sobre

Projetos Locais

Projetos da Biblioteca de Recursos

+ Novo

Importar

Escanear

Filtrar Projetos

🔍

Ordenar:

Última Modificaçã

Criar Novo Projeto

Nome do Projeto:

Novo Projeto de Jogo

Caminho do Projeto:

C:/Users/richard/Documents



Navegar

Você não pode salvar um projeto no caminho selecionado. Crie uma nova pasta ou escolha outro caminho.

Renderizador:

 Avançado+ Mobile Compatibilidade

- Suporta apenas plataformas de desktop.
- Gráficos 3D avançados disponíveis.
- Pode escalar para cenas grandes e complexas.
- Suporta dispositivos de renderização atuais.
- Renderização mais lenta de cenas simples.

O renderizador pode ser alterado posteriormente, mas as cenas podem precisar de ajustes.

Controle de Versão:

Git



Criar & Editar

Cancelar

Sobre

Vamos selecionar a pasta do Projeto

Caminho: D: \ /UNISUL/Jogos

Favoritos: Diretórios & Arquivos:

No meu caso será
D:\Unisul\Jogos

Recentes:

Selecionar a Pasta Atual

Cancelar

Nome do Projeto:

Hello

Criar Pasta

Caminho do Projeto:

✓ Navegar

O nome do projeto será Hello
Depois clicar em “Criar Pasta”

Renderizador:

Avançado+

Mobile

Compatibilidade

- Suporta apenas plataformas de desktop.
- Gráficos 3D avançados disponíveis.
- Pode escalar para cenas grandes e complexas.
- Suporta dispositivos de renderização atuais.
- Renderização mais lenta de cenas simples.

O renderizador pode ser alterado posteriormente, mas as cenas podem precisar de ajustes.

Controle de Versão: Git

Criar & Editar

Cancelar

Nome do Projeto

Hello

Veja que adicionou a pasta da projeto

Criar Pasta

Caminho do Projeto:

D:/UNISUL/Jogos/Hello



Navegar

Renderizador:

Avançado+

Mobile

Compatibilidade

- Suporta apenas plataformas de desktop.
- Gráficos 3D avançados disponíveis.
- Pode escalar para cenas grandes e complexas.
- Suporta dispositivos de renderização atuais.
- Renderização mais lenta de cenas simples.

O renderizador pode ser alterado posteriormente, mas as cenas podem precisar de ajustes.

Controle de Versão:

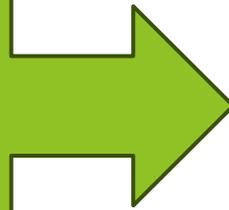
Git



Criar & Editar

Cancelar

Dependendo do projeto você pode escolher o renderizador



Renderizador:

- Avançado+
- Mobile
- Compatibilidade

- Suporta apenas plataformas de desktop.
- Gráficos 3D avançados disponíveis.
- Pode escalar para cenas grandes e complexas.
- Suporta dispositivos de renderização atuais.
- Renderização mais lenta de cenas simples.

Renderizador:

- Avançado+
- Mobile
- Compatibilidade

- Suporta desktop + plataformas mobile.
- Gráficos 3D menos avançados.
- Menos escalável para cenas complexas.
- Suporta dispositivos de renderização atuais.
- Renderização rápida de cenas simples.

Renderizador:

- Avançado+
- Mobile
- Compatibilidade

- Suporta desktop, plataformas mobile + web.
- Gráficos 3D menos avançados (em desenvolvimento).
- Destinado a dispositivos modestos/antigos.
- Usa renderizador OpenGL 3 (OpenGL 3.3/ES 3.0/WebGL2).
- Renderização mais rápida de cenas simples.

Nome do Projeto:

Hello

Criar Pasta

Caminho do Projeto:

D:/UI

Selecione Avançado



Navegar

Renderizador:

Avançado+

Mobile

Compatibilidade

- Suporta apenas plataformas de desktop.
- Gráficos 3D avançados disponíveis.
- Pode escalar para cenas grandes e complexas.
- Suporta dispositivos de renderização atuais.
- Renderização mais lenta de cenas simples.

O renderizador pode ser alterado posteriormente, mas as cenas podem precisar de ajustes.

Controle de Versão:

Git



Criar & Editar

Cancelar

Nome do Projeto:

Hello

Criar Pasta

Caminho do Projeto:

D:/UNISUL/Jogos/Hello



Navegar

Renderizador:

Avançado+

- Suporta apenas plataformas de desktop.
- Gráficos 3D avançados disponíveis.
- Pode escalar para cenas grandes e complexas.

Mobile

Compatibilidade

Se já tem conta no Git pode usar o Git

O renderizador pode ser alterado posteriormente, mas as cenas podem precisar de ajustes.

Controle de Versão:

Git

Nenhum

Git

Cancelar

Nome do Projeto:

Hello

Criar Pasta

Caminho do Projeto:

D:/UNISUL/Jogos/Hello



Navegar

Renderizador:

Avançado+

Mobile

Compatibilidade

- Suporta apenas plataformas de desktop.
- Gráficos 3D avançados disponíveis.
- Pode escalar para cenas grandes e complexas.
- Suporta dispositivos de renderização atuais.
- Renderização mais lenta de cenas simples.

O renderizador p

Depois clique em Criar & Editar

mente, mas as cenas podem precisar de ajustes.

Controle de Versão: Git

Criar & Editar

Cancelar

Cena Importar

Filtro: nome, t:tipo, g

Criar Nó Raiz:

- Cena 2D
- Cena 3D
- Interface de Usuário
- Outro Nó

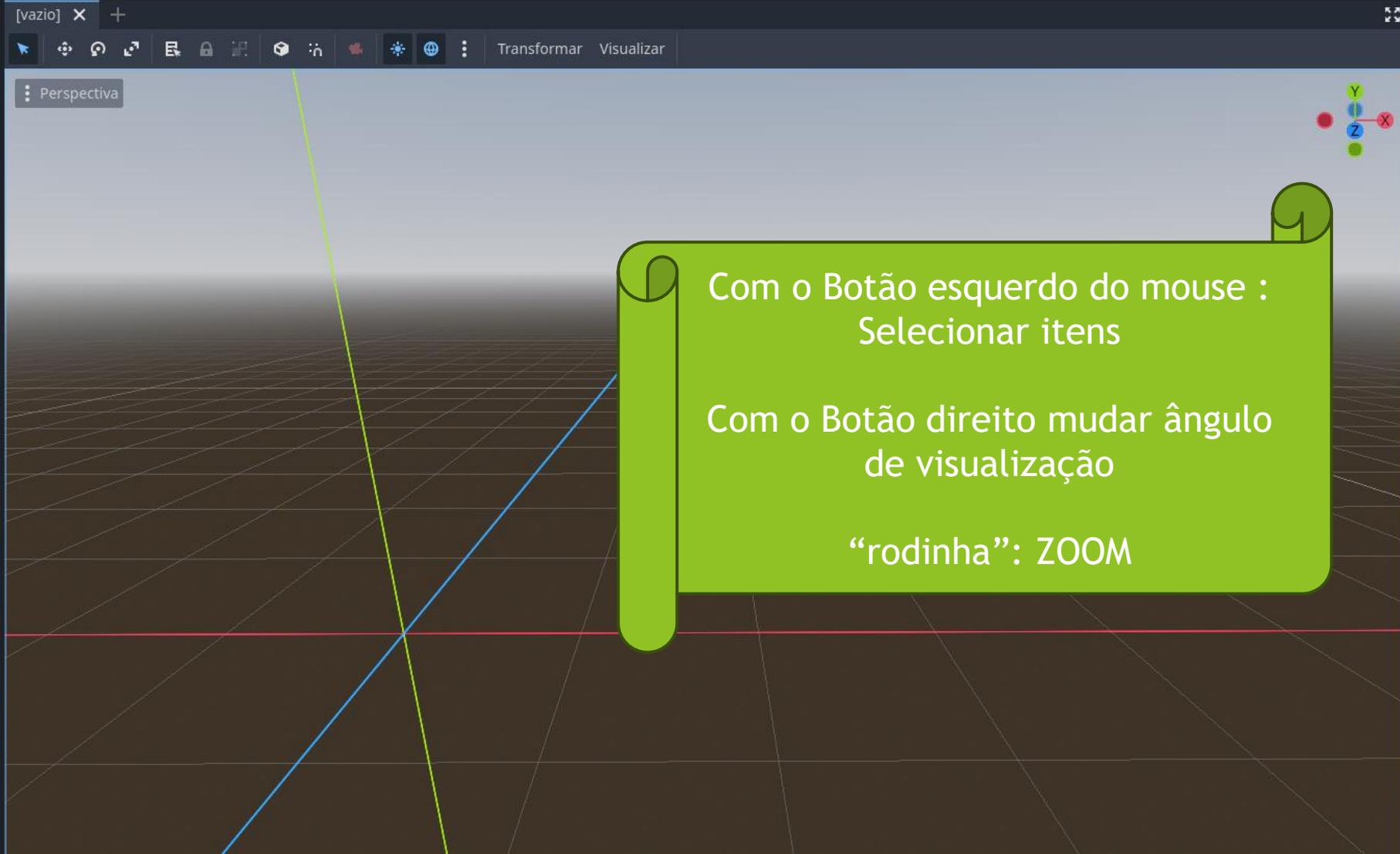
Arquivos

res://

Filtrar Arquivos

Favoritos:

- res://
- icon.svg



Com o Botão esquerdo do mouse :
Selecionar itens

Com o Botão direito mudar ângulo
de visualização

“rodinha”: ZOOM

Inspeção Nó Histórico

Filtrar Propriedades

Importar

Filtro: nome, t:tipo, g

Criar Nó Raiz:

- Cena 2D
- Cena 3D
- Interface de Usuário
- Outro Nó

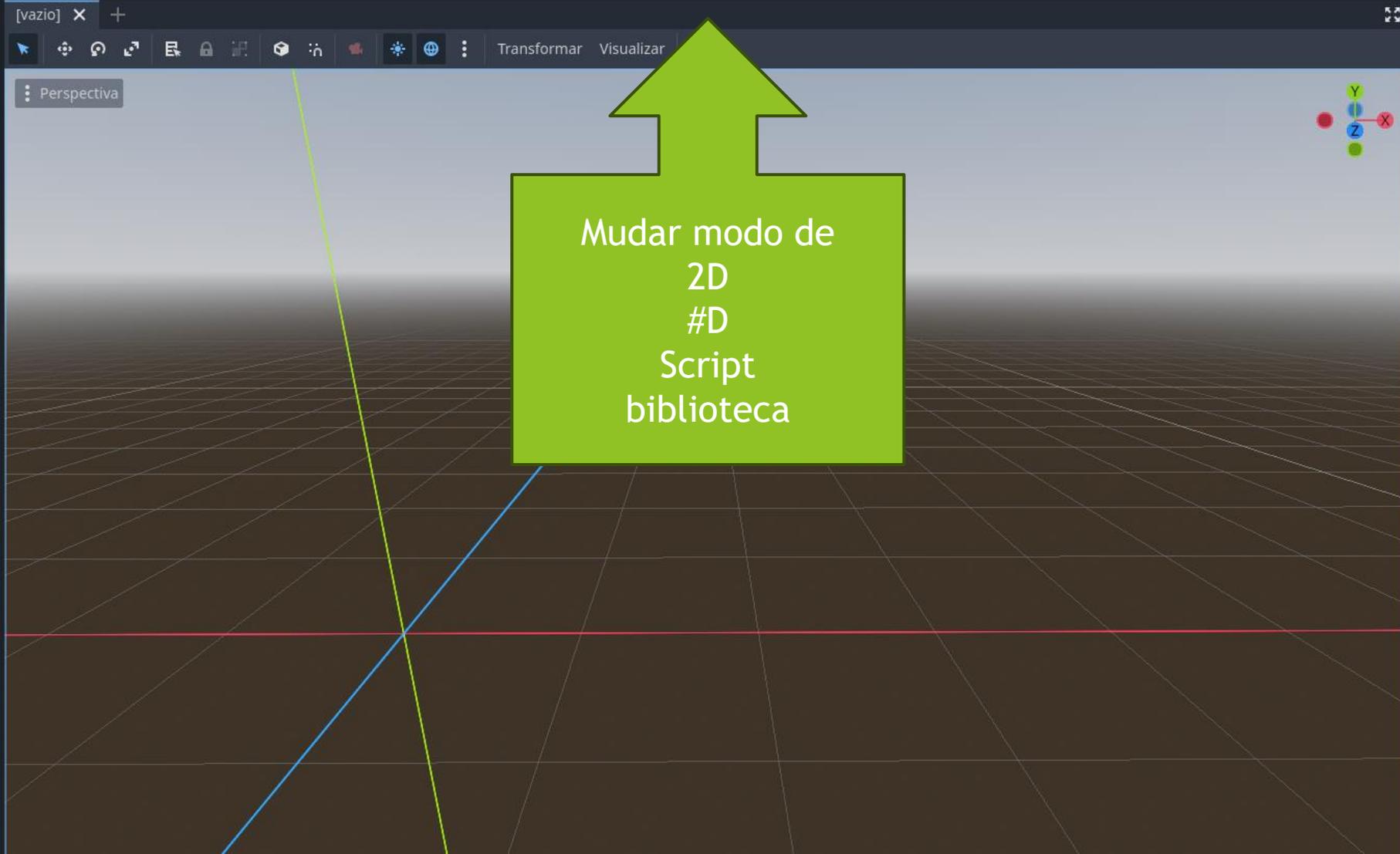
Arquivos

res://

Filtrar Arquivos

Favoritos:

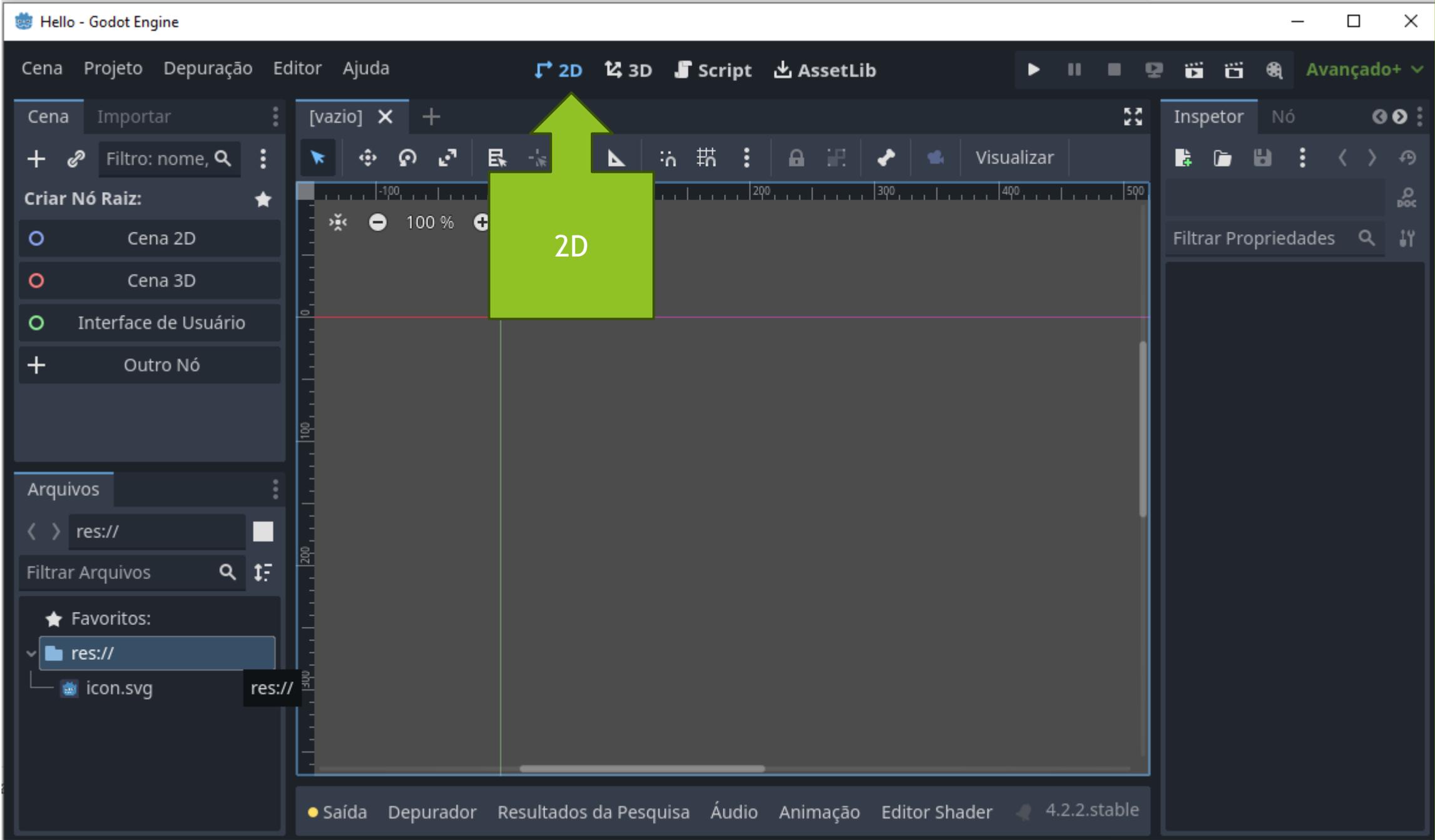
- res://
- icon.svg



Mudar modo de
2D
#D
Script
biblioteca

Inspetor Nó Histórico

Filtrar Propriedades



Script

Cena Importar

+ 🔗 Filtro: nome, 🔍

Criar Nó Raiz: ★

- Cena 2D
- Cena 3D
- Interface de Usuário
- + Outro Nó

Arquivos

< > res://

Filtrar Arquivos 🔍 ⚙️

★ Favoritos:

- res://
- icon.svg

Arquivo Pesquisar Depuração Documentação Online 📄 Pesquisar Ajuda 🔍

Filtrar Script 🔍

Inspetor Nó

Filtrar Propriedades 🔍 ⚙️

Bibliotecas
AssetLib



Cena Importar

+ 🔗 Filtro: n 🔍

Criar Nó Raiz: ★

- Cena 2D
- Cena 3D
- Interface de Usuário
- + Outro Nó

[vazio] ✕ +

Pesquisar Recursos (Excluindo Modelos, Projetos e Demonstrações) 🔍 Importar... Plugins...

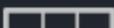
Ordenar: Atualizado Rece ▾ Categoria: Todos ▾ Site: godotengine.or ▾ Suporte

Primeiro Anterior 1 2 3 4 5 6 7 8 9 10 Próximo Último

 Vertex Color Import Swapper
Ferramentas
Invertex
MIT

 Godot Rollback Netcode (Godot 4)
Scripts
dsnopek
MIT

 Spatial Gardener (Godot 4.x)
3D Tools
dreadpon
MIT

 Config Table Manager

Arquivos

< > res://

Filtrar Arquivos 🔍 ⚙️

★ Favoritos:

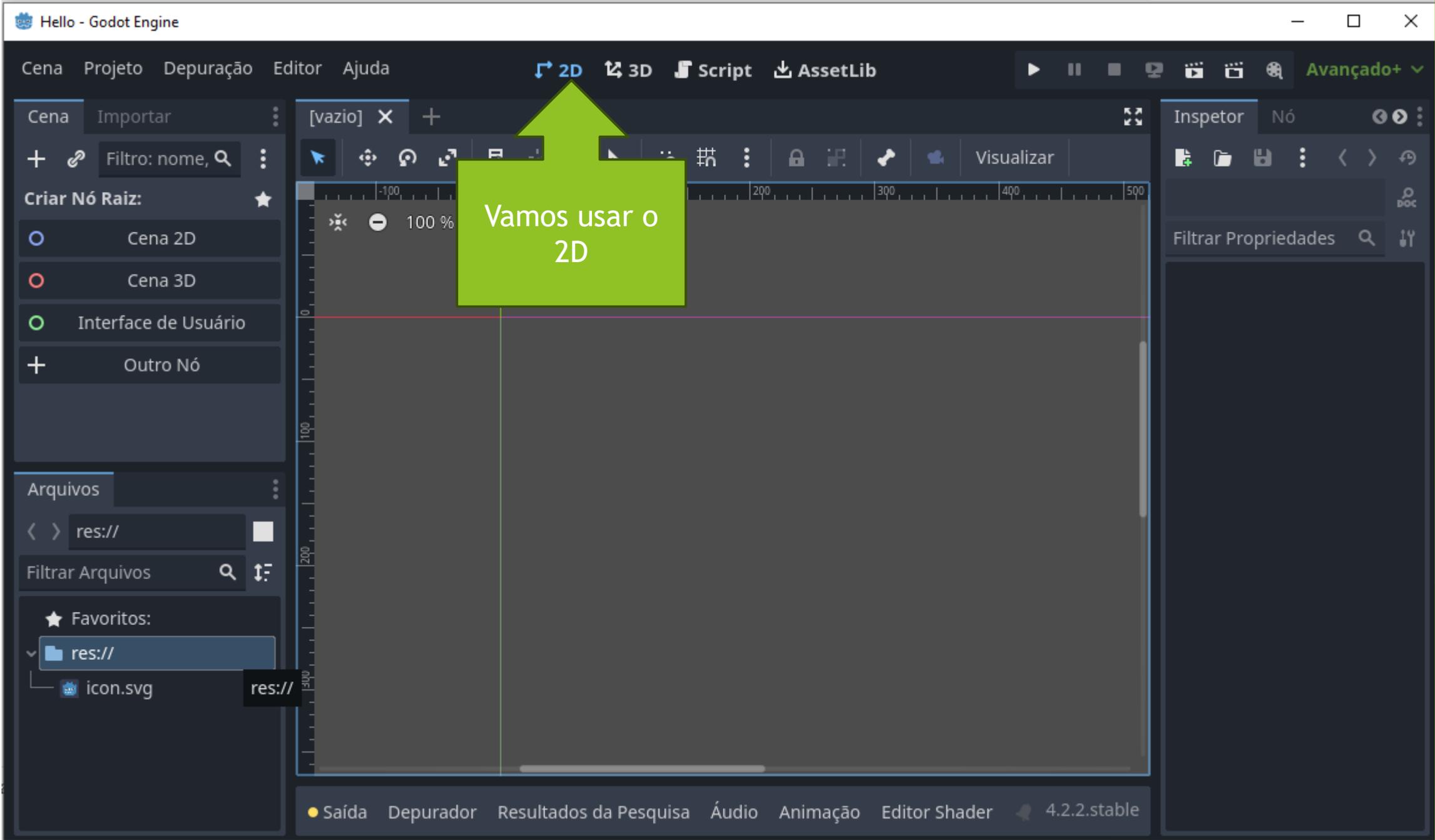
- res://
- icon.svg

Inspetor Nó

📄 📄 📄 📄 📄 📄 📄 📄

DOC

Filtrar Propriedades 🔍 ⚙️



Vamos usar o 2D

Importar

Filtro: nome, t:tipo, g

Criar Nó Raiz:

- Cena 2D
- Cena 3D
- Interface de Usuário
- Outro Nó

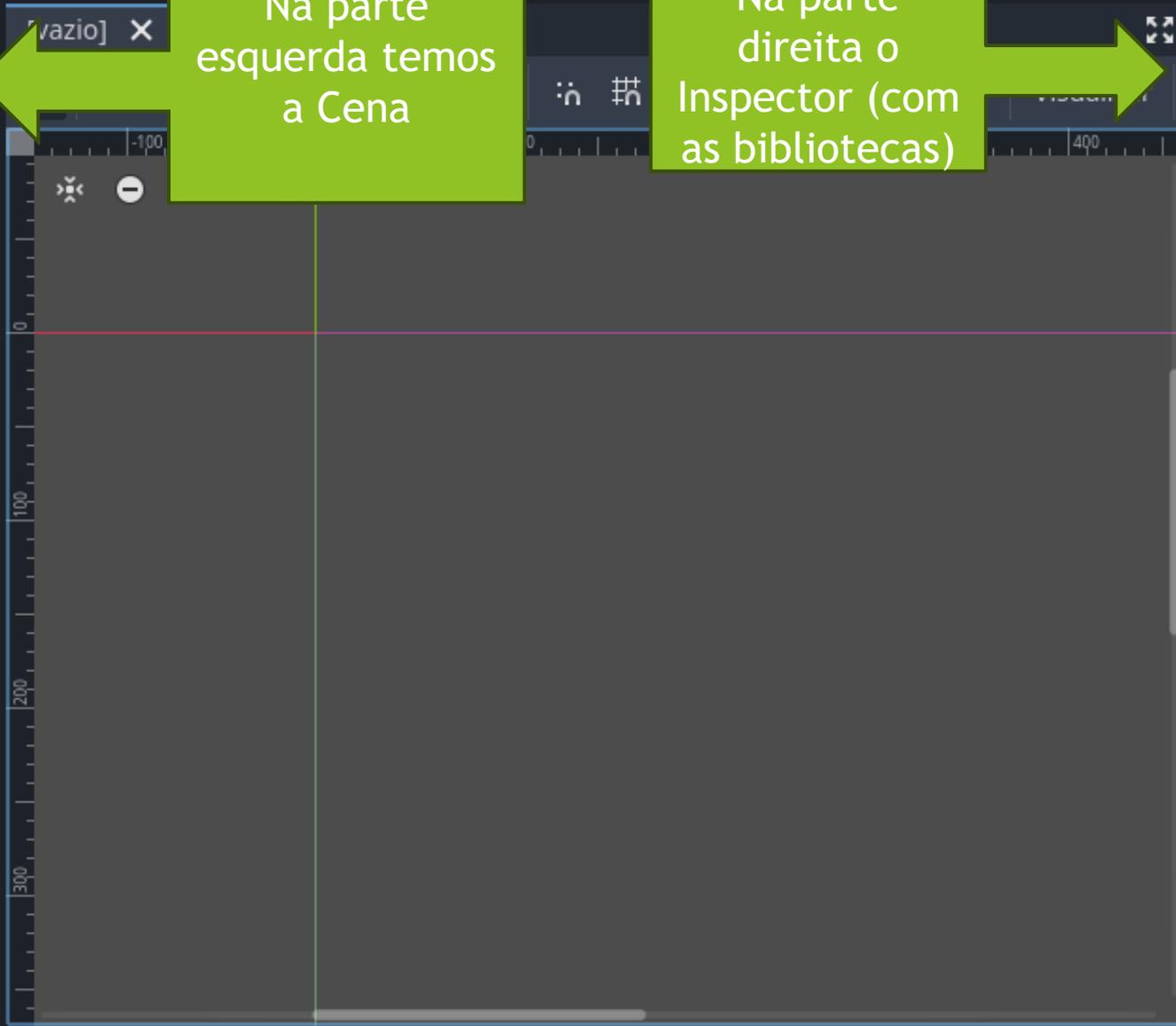
Arquivos

res://

Filtrar Arquivos

Favoritos:

- res://
- icon.svg



Na parte esquerda temos a Cena

Na parte direita o Inspector (com as bibliotecas)

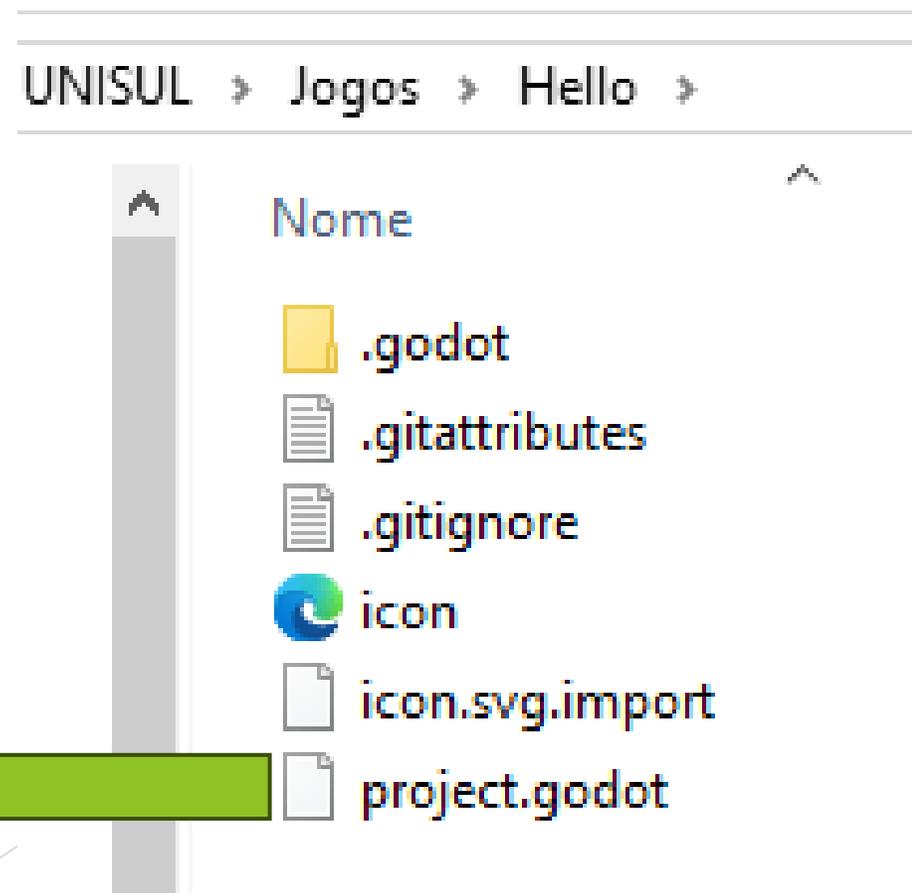
Inspector Nó Histórico

Filtrar Propriedades

GODOT

- ▶ Note que o projeto criado, basicamente são arquivos texto

```
project.godot x
1 ; Engine configuration file.
2 ; It's best edited using the editor UI and not directly,
3 ; since the parameters that go here are not all obvious.
4 ;
5 ; Format:
6 ; [section] ; section goes between []
7 ; param=value ; assign values to parameters
8
9 config_version=5
10
11 [application]
12
13 config/name="Hello"
14 config/features=PackedStringArray("4.2", "Forward Plus")
15 config/icon="res://icon.svg"
16
```



GODOT

- ▶ O projeto Hello está na página wiki do prof. Saulo
 - ▶ https://pati.arisa.com.br/wiki/index.php/Godot_Engine:_Hello_Game_World!

Godot Engine: Hello Game World!



- ▶ Vamos precisar dos Assets

- ▶ Assets

 - ▶ https://arisa.com.br/~saulo/aulas/unisul/games/godot/walking_assets.zip

 - ▶ São recursos

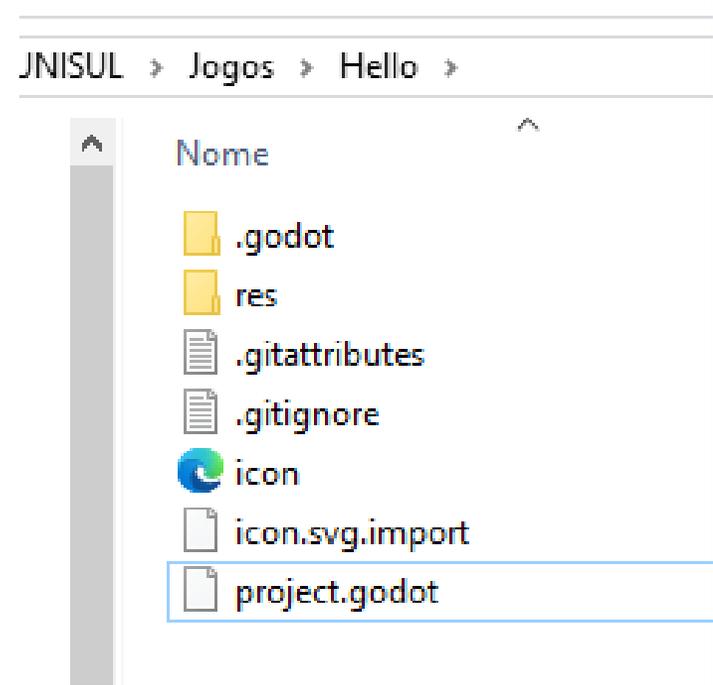
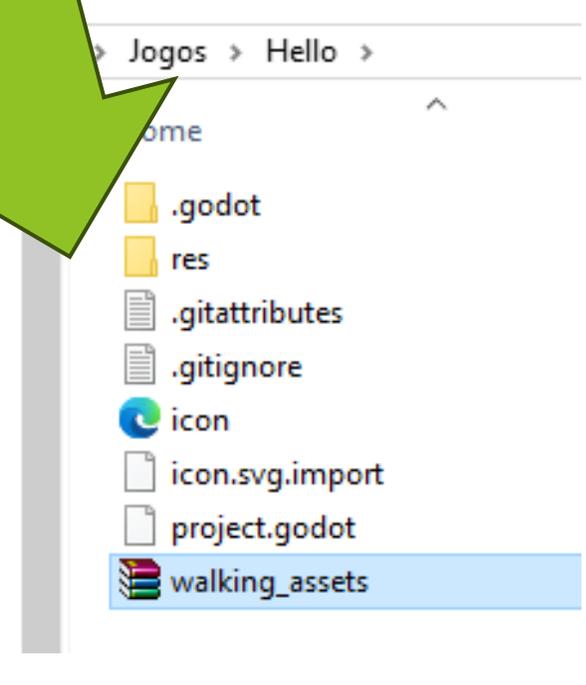
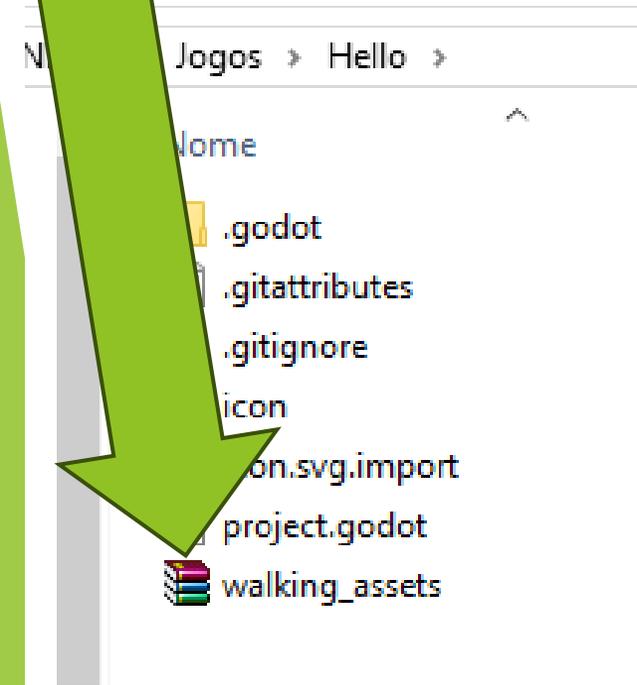
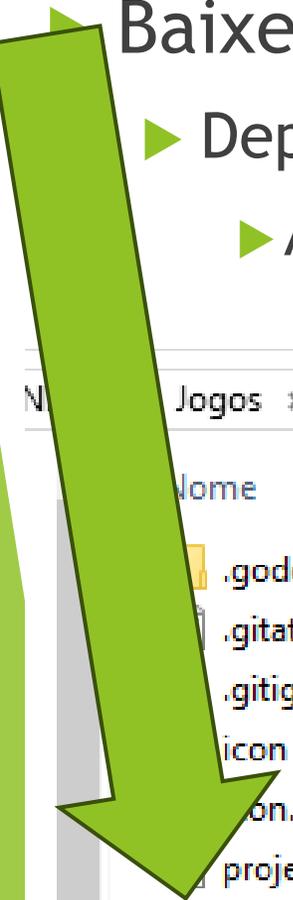
 - ▶ Imagens, sons etc...

Godot Engine: Hello Game World!



Baixe os Assets dentro da pasta do Projeto.

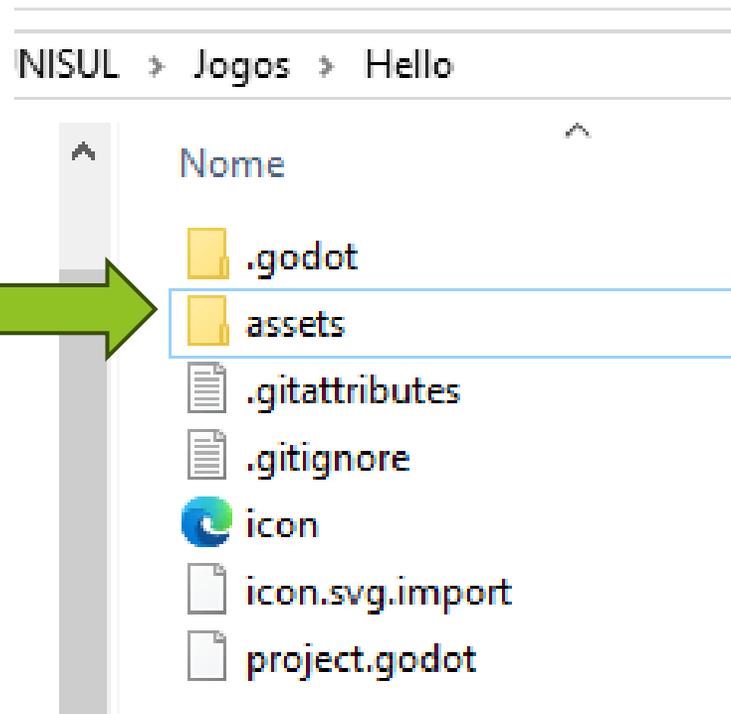
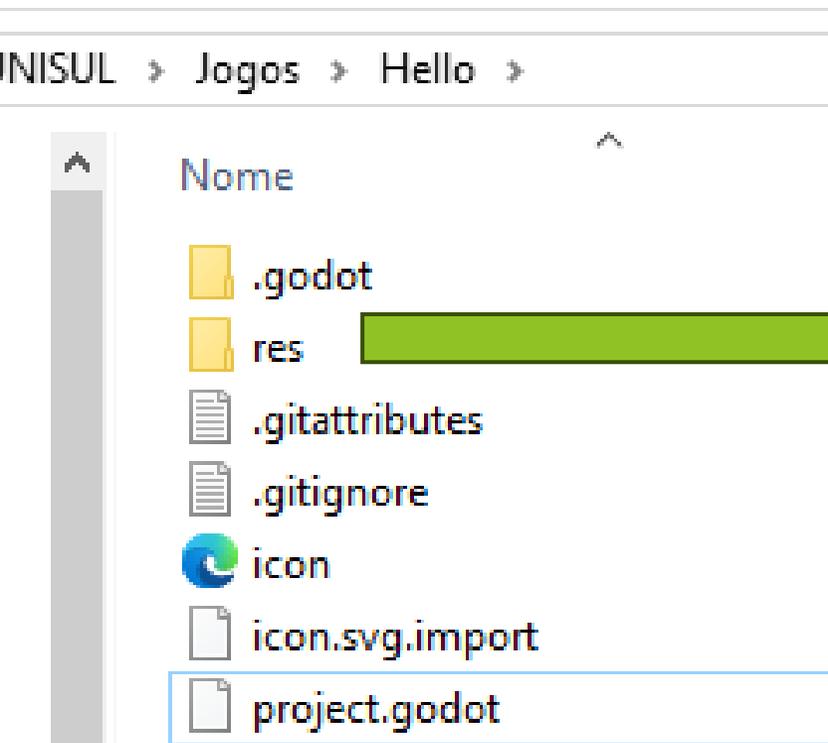
- ▶ Deposite Extraia a pasta “res”
- ▶ Apague o zip





Godot Engine: Hello Game World!

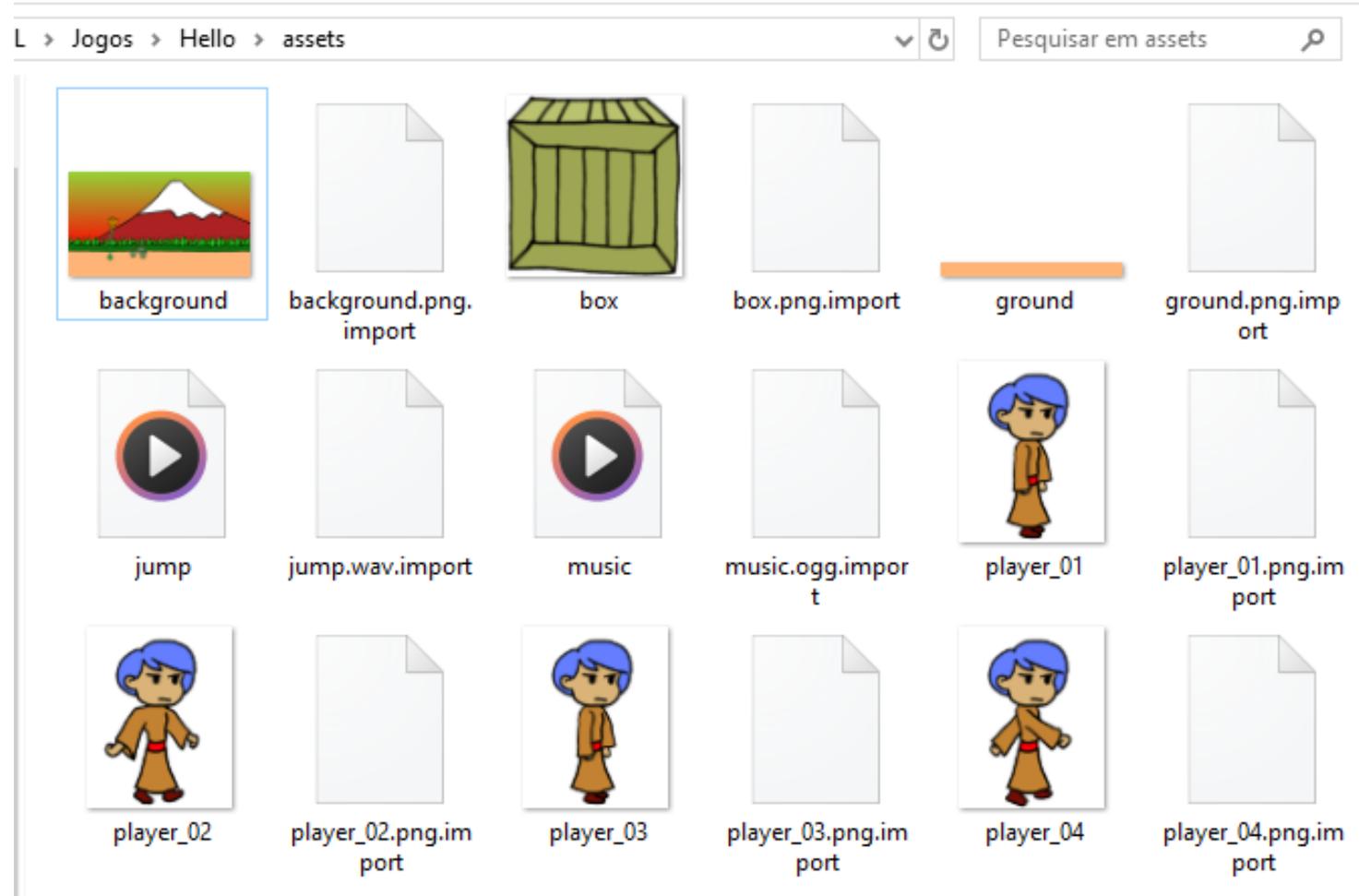
- Renomeia a pasta res para assets



Godot Engine: Hello Game World!



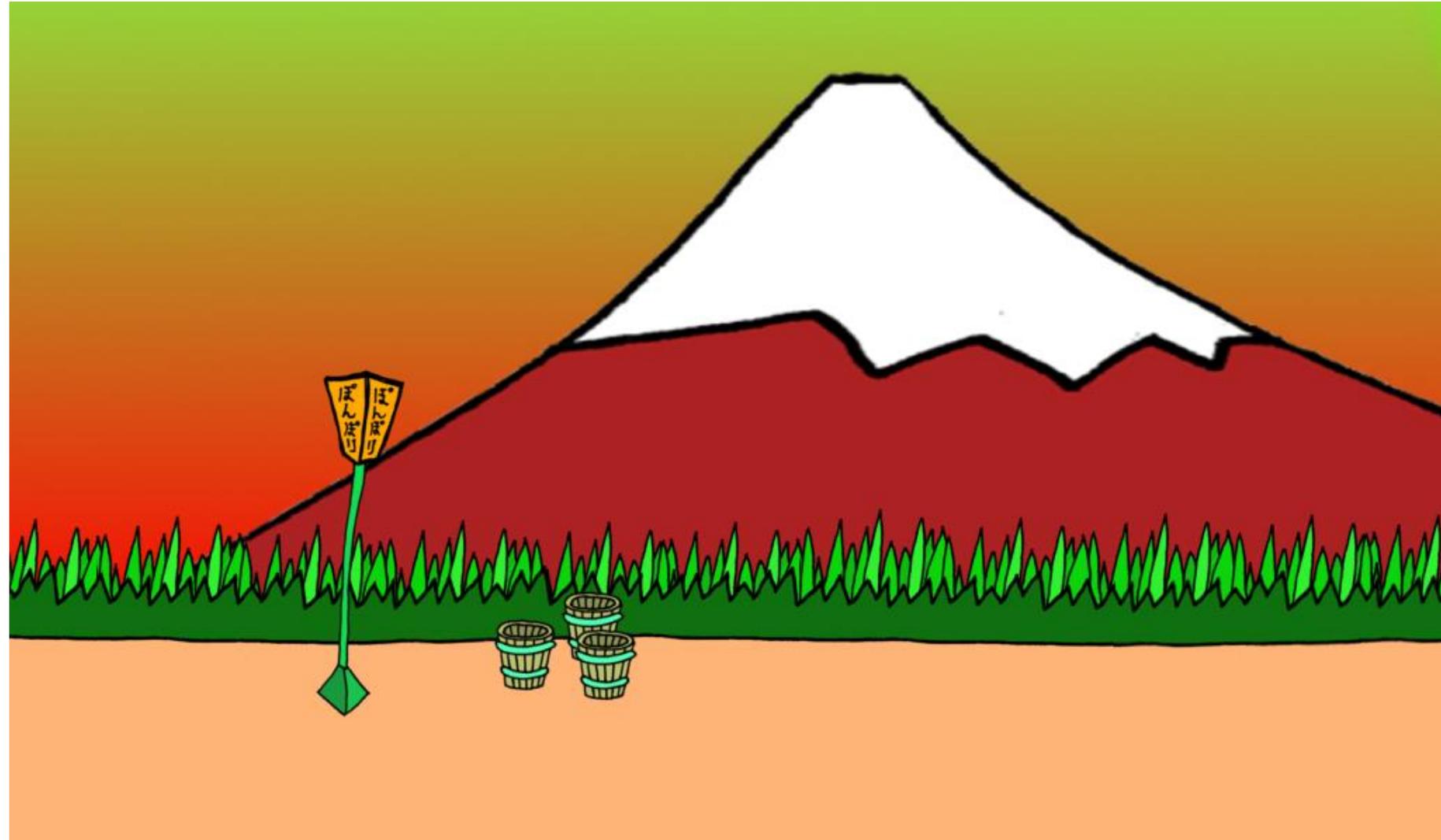
► O que temos nos assets.

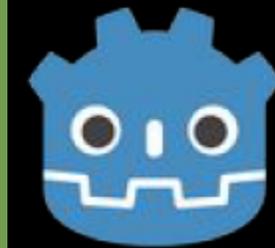




Godot Engine: Hello Game World!

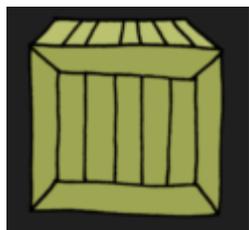
► Figura de fundo.





Godot Engine: Hello Game World!

- ▶ Tem uma caixa.



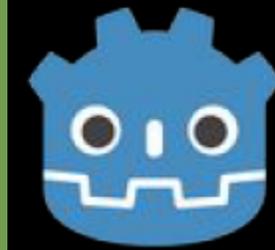
- ▶ Tem o chão.



- ▶ Tem os bonecos.

- ▶ Simular movimento





Godot Engine: Hello Game World!

▶ Som de pulo

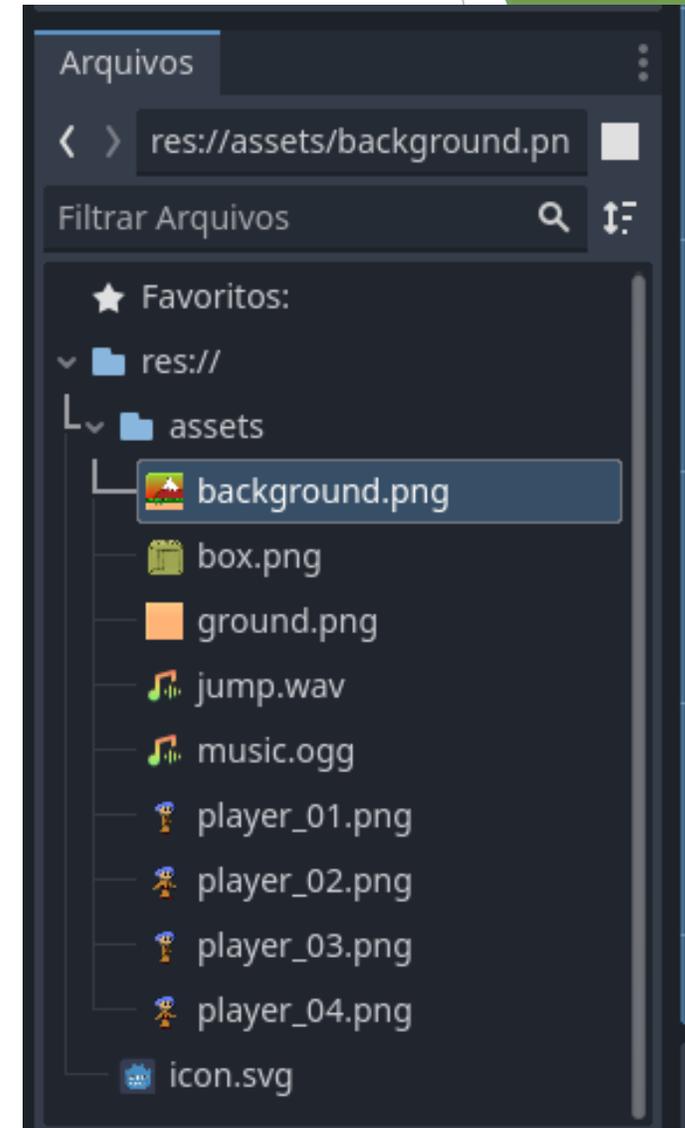
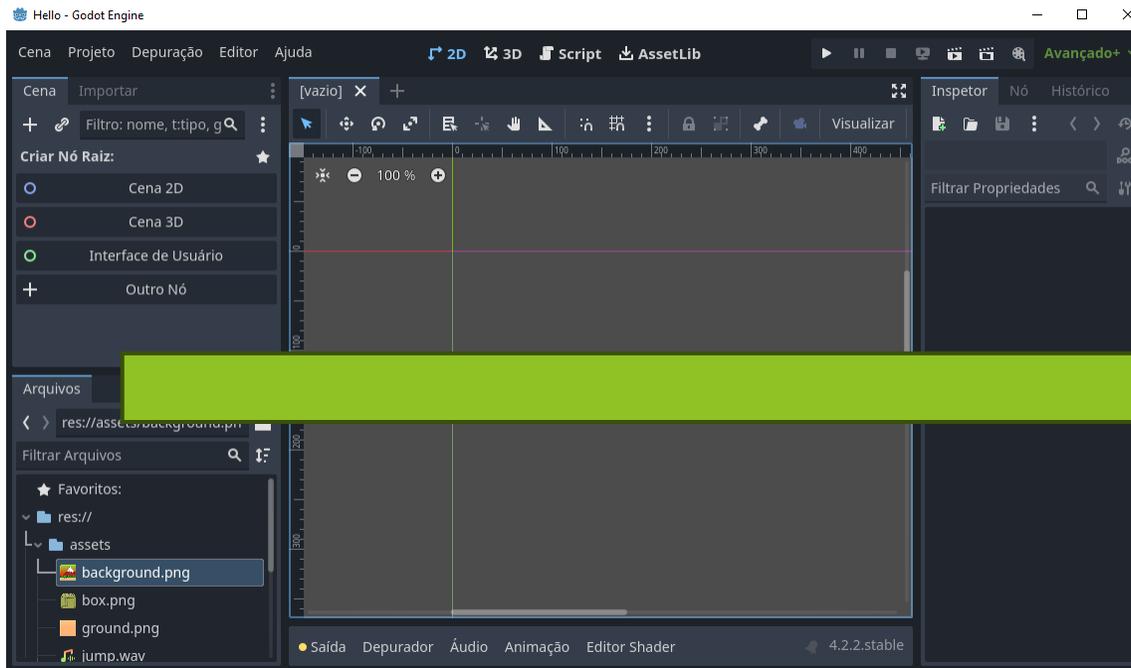
▶ Musica





Godot Engine: Hello Game World!

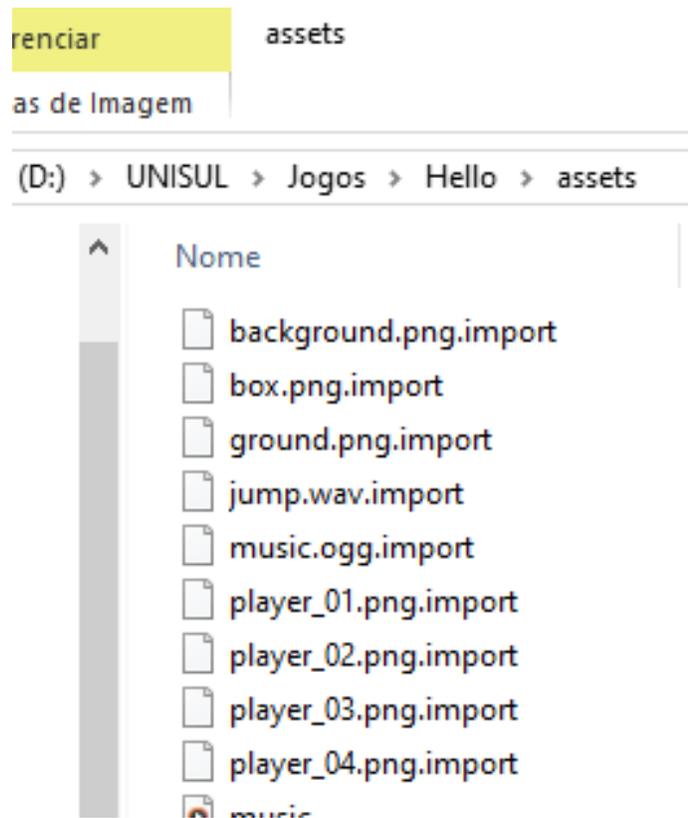
- ▶ Voltando ao GODOT
 - ▶ Os Assests aparecem na guia Arquivos.





Godot Engine: Hello Game World!

- ▶ O GODOT cria automaticamente os arquivos de importação.



```
background.png.import x
1  [remap]
2
3  importer="texture"
4  type="CompressedTexture2D"
5  uid="uid://dvqs4isguo5qq"
6  path="res://.godot/imported/backgro
7  metadata={
8    "vram_texture": false
9  }
10
11  [deps]
12
13  source_file="res://assets/backgroun
14  dest_files=["res://.godot/imported/
15
16  [params]
17
18  compress/mode=0
19  compress/high quality=false
```

Cena Importar

+

Clique no “+”
Dentro da guia Cena

Criar Nó Raiz:

- Cena 2D
- Cena 3D
- Interface de Usuário
- + Outro Nó

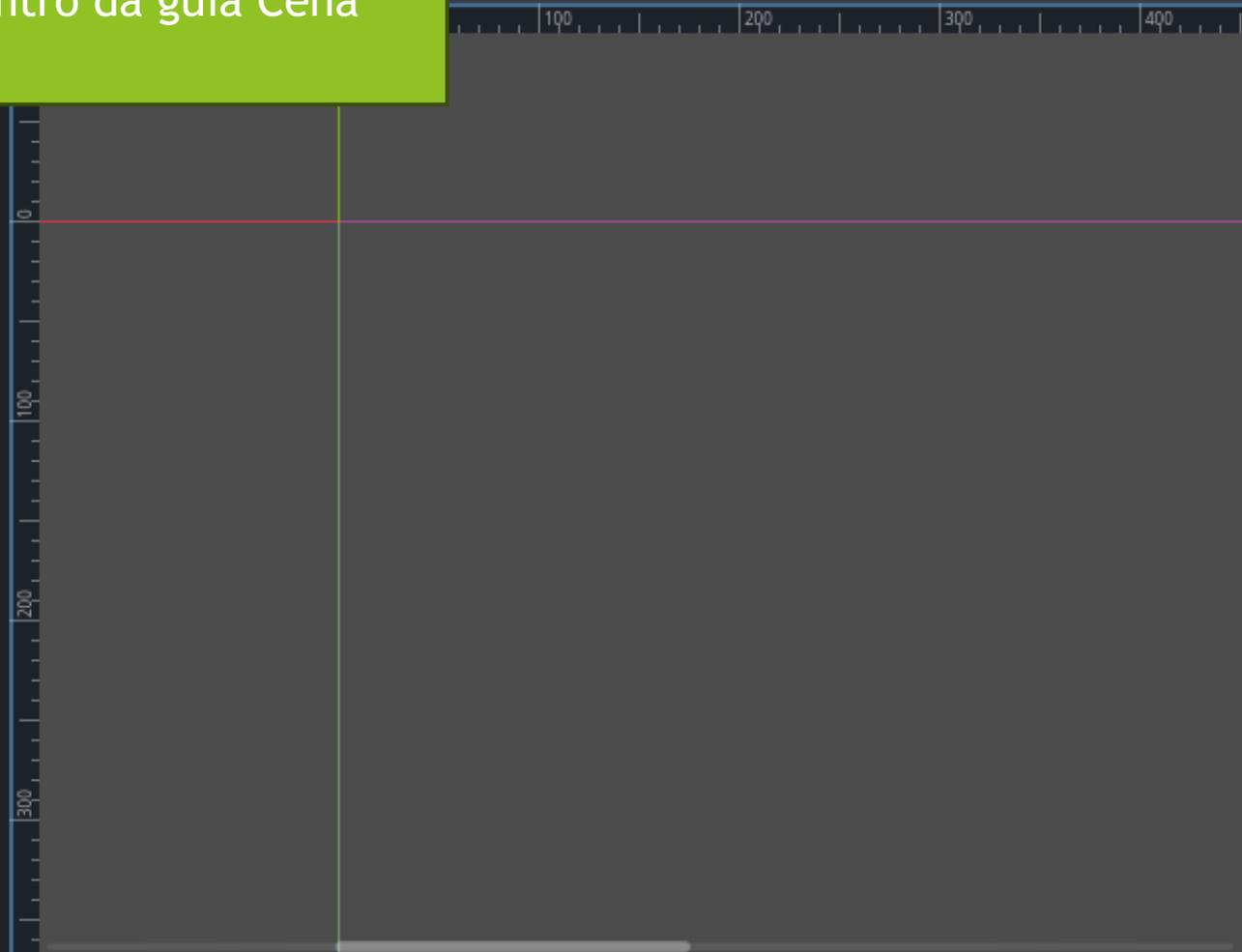
Arquivos

< > res://assets/background.pn

Filtrar Arquivos 🔍 ⚙️

★ Favoritos:

- ▼ res://
 - └─ assets
 - 🖼️ background.png
 - 📦 box.png
 - 🟠 ground.png
 - 🎵 jump.wav



Inspetor Nó Histórico

📄 📄 📄 🔍 ⏪ ⏩ ↺

Filtrar Propriedades 🔍 🛠️

Procure por staticBody2D

Favoritos:

Pesquisar:

staticBody

Correspondências:

- Node
 - CanvasItem
 - Node2D
 - CollisionObject2D
 - PhysicsBody2D
 - StaticBody2D**
- Node3D
 - CollisionObject3D
 - PhysicsBody3D
 - StaticBody3D

Recentes:

Clique em criar

Descrição:

StaticBody physics body that can't be moved by external forces. When moved manually, it doesn't affect other bodies in its path.

Criar

Cancelar

Cena Importar

[não salvo](*)

Filtro: nom

StaticBody2D

Renomeie para Box

Arquivos

res://assets/background

Filtrar Arquivos

Favoritos:

res://

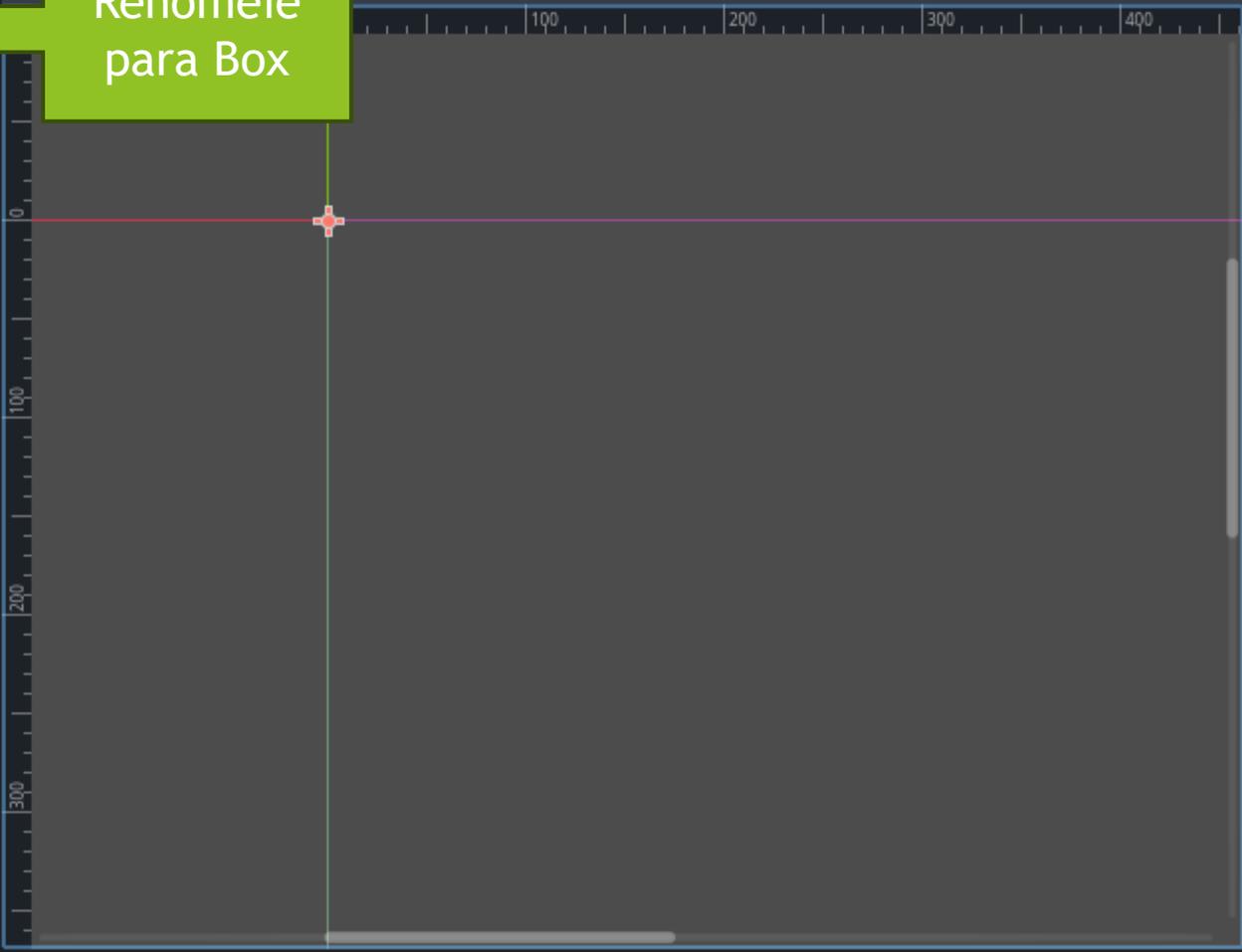
assets

background.png

box.png

ground.png

jump.wav



Inspetor Nó Histórico

StaticBody2D

Filtrar Propriedades

StaticBody2D

Physics Mate... [vazio]

Constant Lin... x 0 px/s

y 0 px/s

Constant Ang... 0 °/s

CollisionObject2D

Disable Mode Remove

Collision

Input

Node2D

Transform

CanvasItem

Visibility

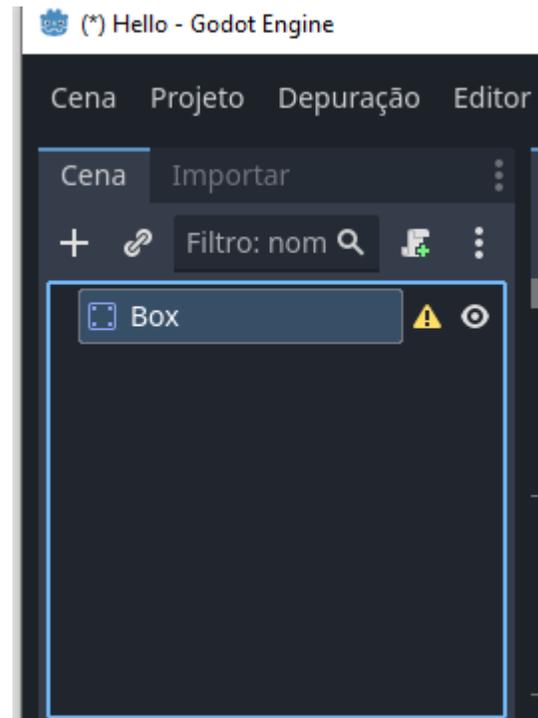
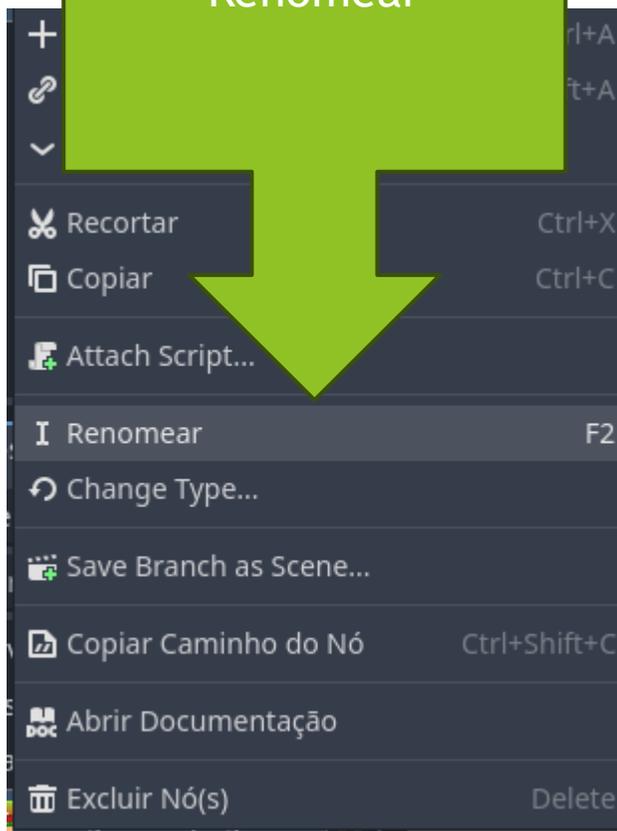
Ordering

Texture

Material

Node

Botão Direito
E selecione
Renomear

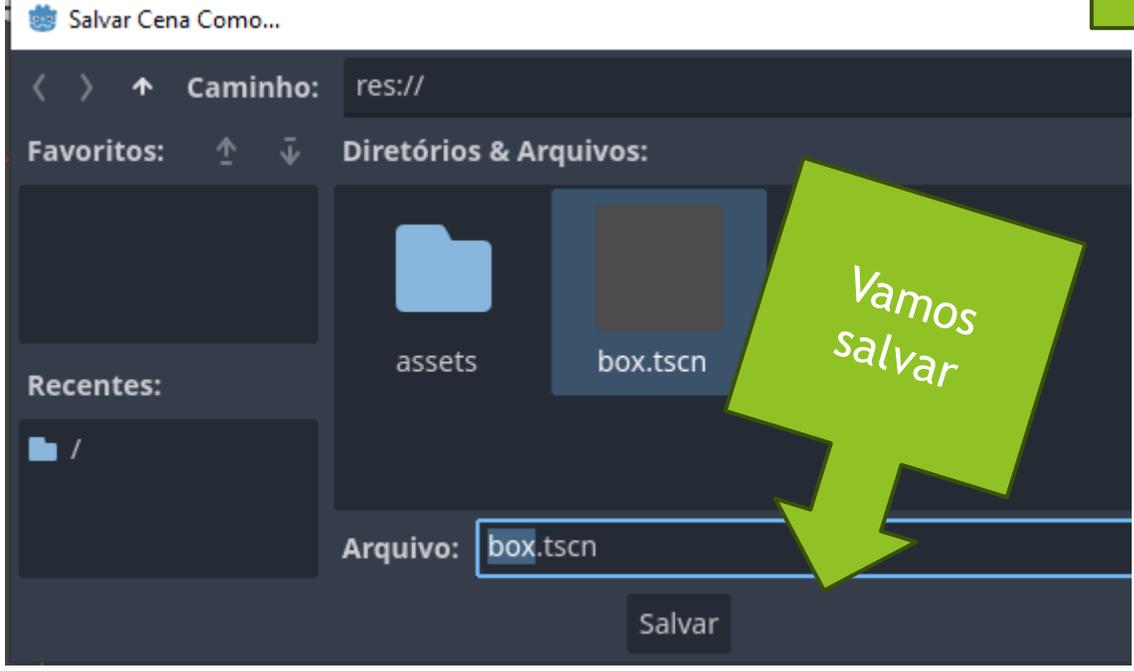


box.tscn - Hello - Godot Engine



The image shows the top menu bar of the Godot Engine interface. The 'Cena' menu is open, and the 'Salvar Cena' option is highlighted. A green callout box with the text 'Vamos salvar' and a downward-pointing arrow is positioned over the 'Salvar Cena' option.

Menu Item	Shortcut
Nova Cena	Ctrl+N
Nova Cena H...	Ctrl+Shift+N
Abrir Cena...	Ctrl+O
Reabrir Cena Fechad...	Ctrl+Shift+T
Abrir Recentes	→
Salvar Cena	Ctrl+S
Salvar Cena Como...	Ctrl+Shift+S
Salvar Todas as Cenas	Ctrl+Shift+Alt+S



The image shows the 'Salvar Cena Como...' dialog box in Godot Engine. The file name 'box.tscn' is entered in the 'Arquivo:' field and is highlighted. A green callout box with the text 'Vamos salvar' and a downward-pointing arrow is positioned over the 'Arquivo:' field. The 'Salvar' button is visible at the bottom right.

Salvar Cena Como...

Caminho: res://

Diretórios & Arquivos:

- assets
- box.tscn

Recentes:

- /

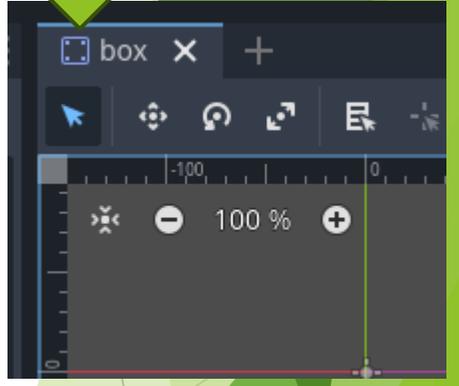
Arquivo: **box.tscn**

Salvar

cena



A green callout box containing the text 'cena' and a downward-pointing arrow.

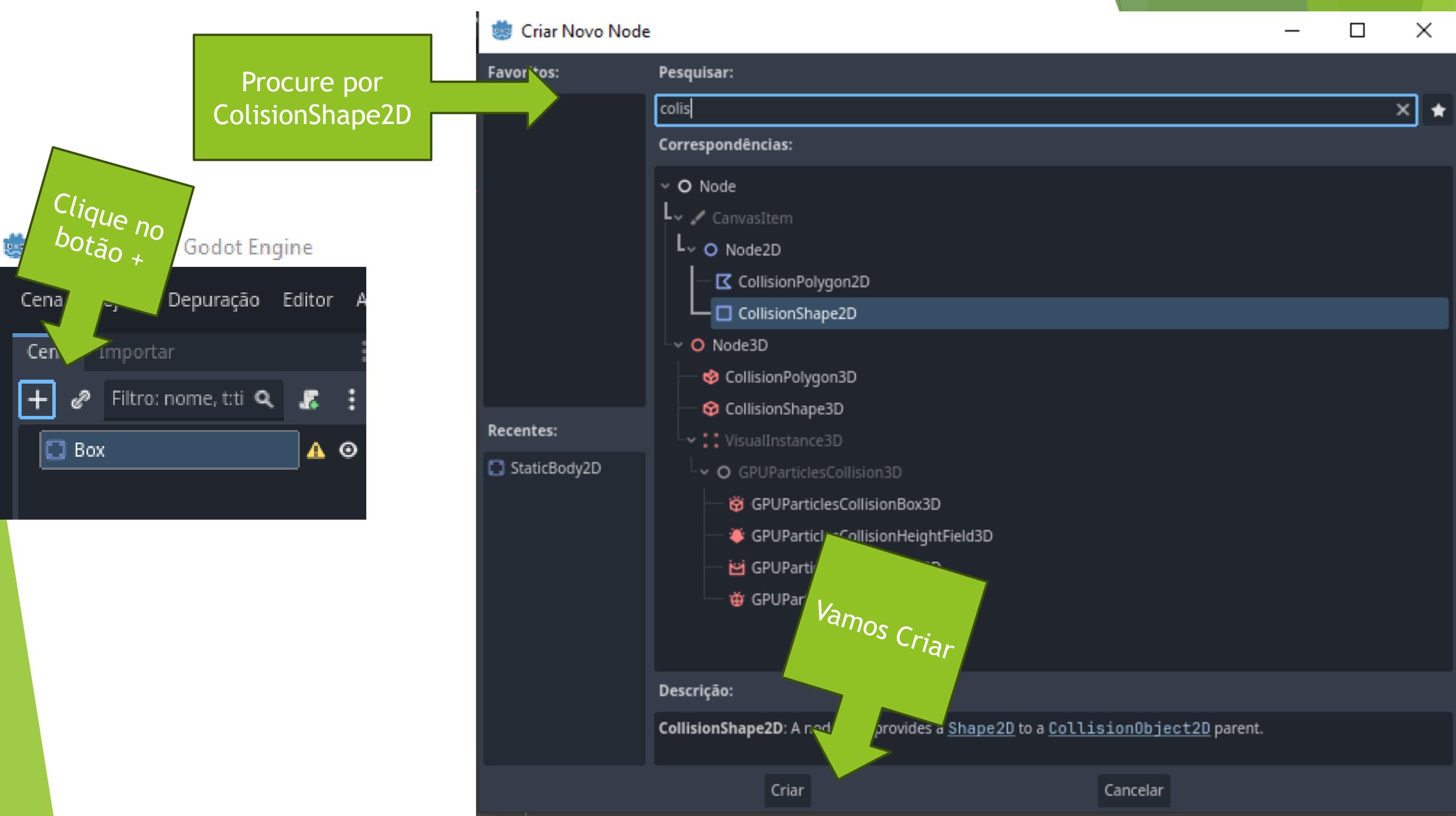


The image shows a Godot Engine viewport window titled 'box'. The scene is displayed in a 2D perspective view. A green callout box with the text 'Vamos salvar' and a downward-pointing arrow is positioned over the scene. The viewport shows a simple scene with a floor and walls, and a camera view.

The image shows a screenshot of a game engine's interface. At the top, there are tabs for 'Cena' and 'Importar'. Below them is a search filter 'Filtro: nome, t:tipo, g' and a toolbar with various icons for navigation and editing. A central toolbar contains icons for selection, rotation, translation, and scaling. Below the toolbar is a ruler with markings at -100, 0, 100, 200, 300, and 400. On the left, a panel shows a 'Box' object with a yellow warning icon and a visibility icon. A dark overlay box contains the following text:

Aviso de configuração de nó:

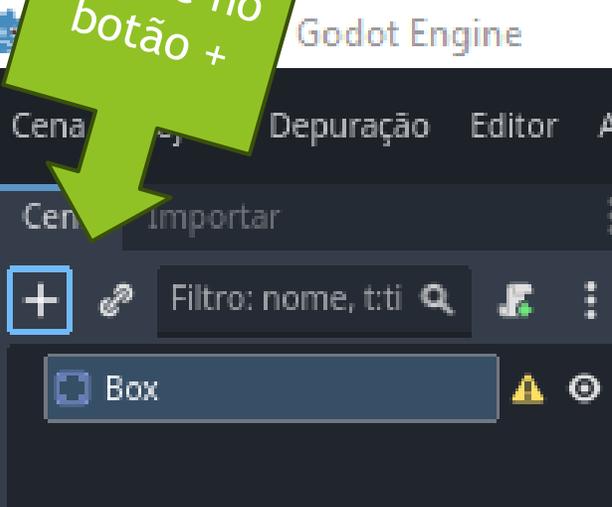
- Este nó não tem forma, por isso não pode colidir ou interagir com outros objetos. Considere adicionar um CollisionShape2D ou CollisionPolygon2D como filho para definir sua forma.



Procure por ColisionShape2D

Clique no botão +

Vamos Criar



Criar Novo Node

Favoritos:

Pesquisar:

colis

Correspondências:

Node

CanvasItem

Node2D

CollisionPolygon2D

CollisionShape2D

Node3D

CollisionPolygon3D

CollisionShape3D

VisualInstance3D

GPUParticlesCollision3D

GPUParticlesCollisionBox3D

GPUParticlesCollisionHeightField3D

GPUParticlesCollisionSphere3D

GPUParticlesCollisionTriangleMesh3D

Descrição:

CollisionShape2D: A node provides a Shape2D to a CollisionObject2D parent.

Criar

Cancelar

(*) box.tscn - h2 - Godot Engine

Cena Projeto Depuração Editor Ajuda

2D 3D Script

Cena Importar

+ 🔗 Filtro: nome, tti 🔍 ⚙️

Box
CollisionShape2D ⚠️

box(*) X +

🖱️ 📏 🔄 🗑️ 📐 📏 📏 📏 🔒 📄 🔗 🌐 Visualizar

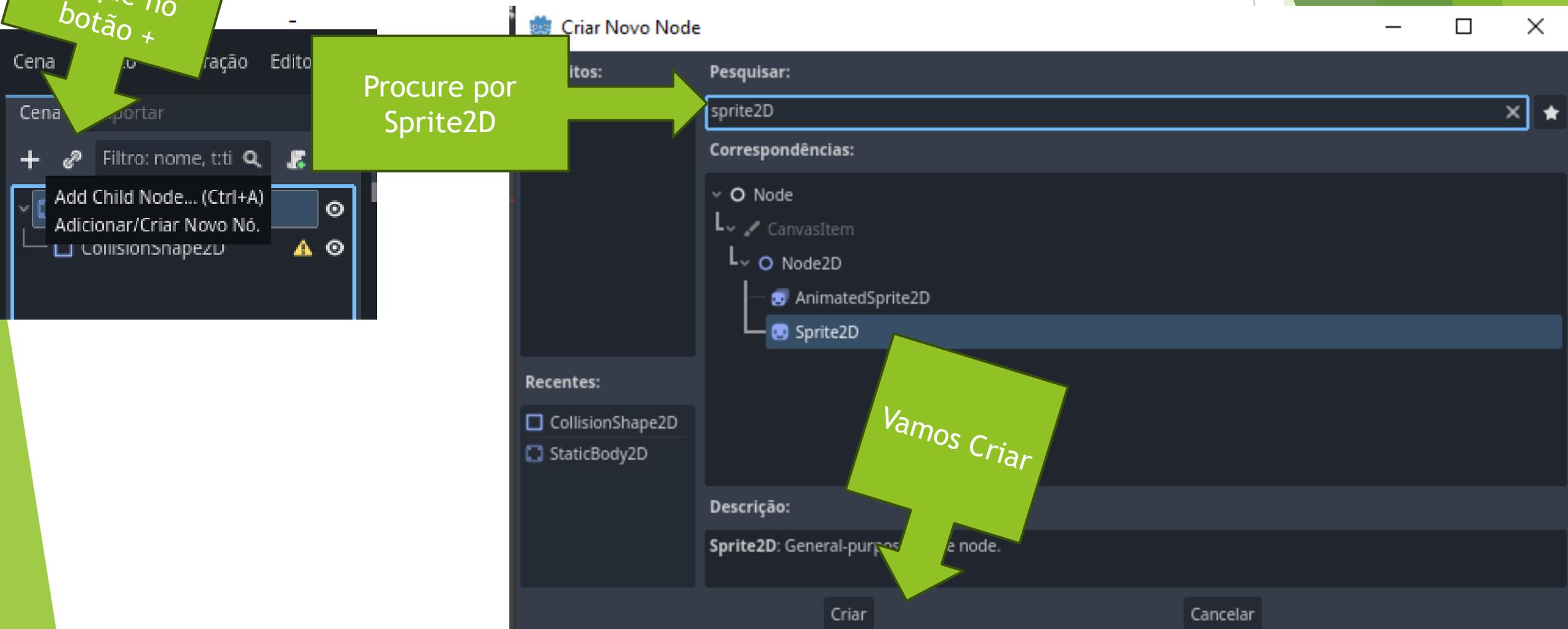
-100 0 100 200 300
📏 - 100% +

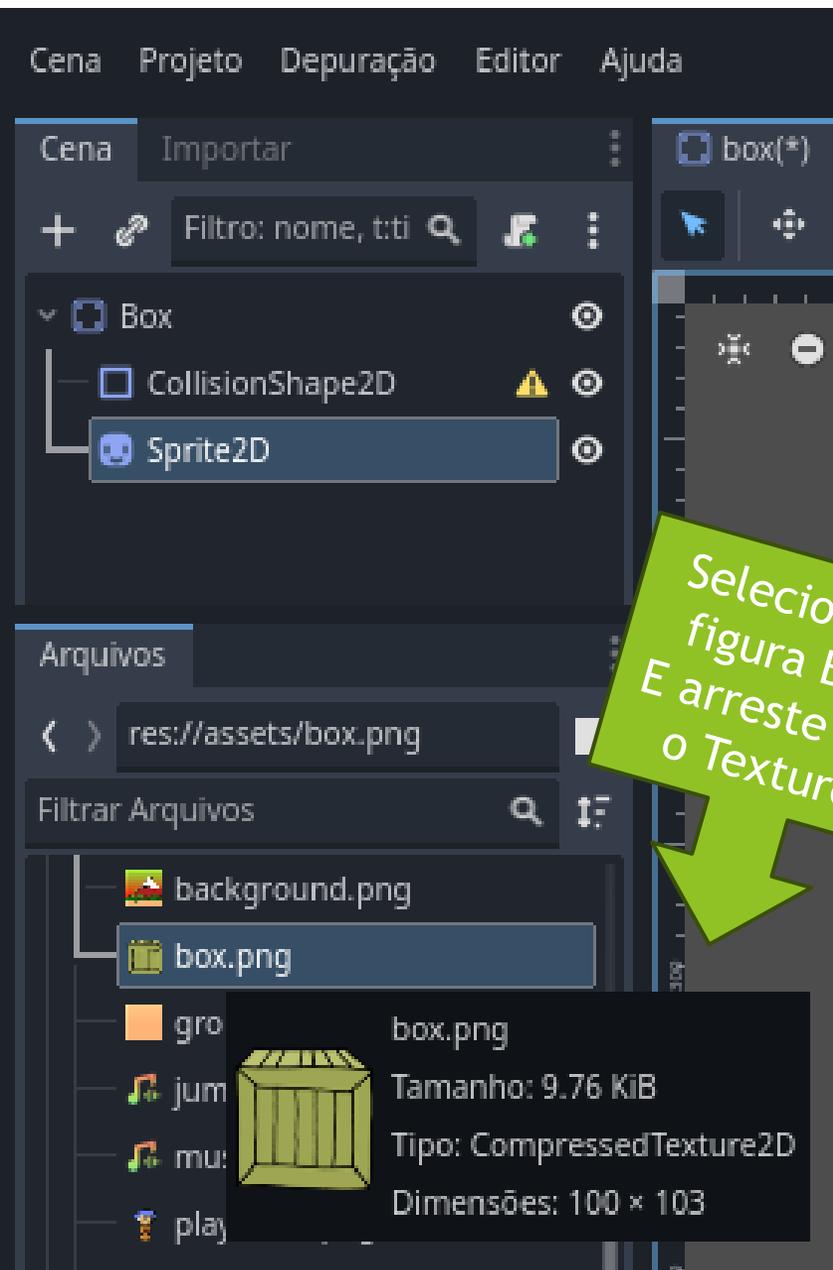
Aviso de configuração de nó:
• Uma forma deve ser fornecida para que CollisionShape2D funcione. Crie um recurso de forma para isso!

Selecione
Box , depois
Clique no
botão +

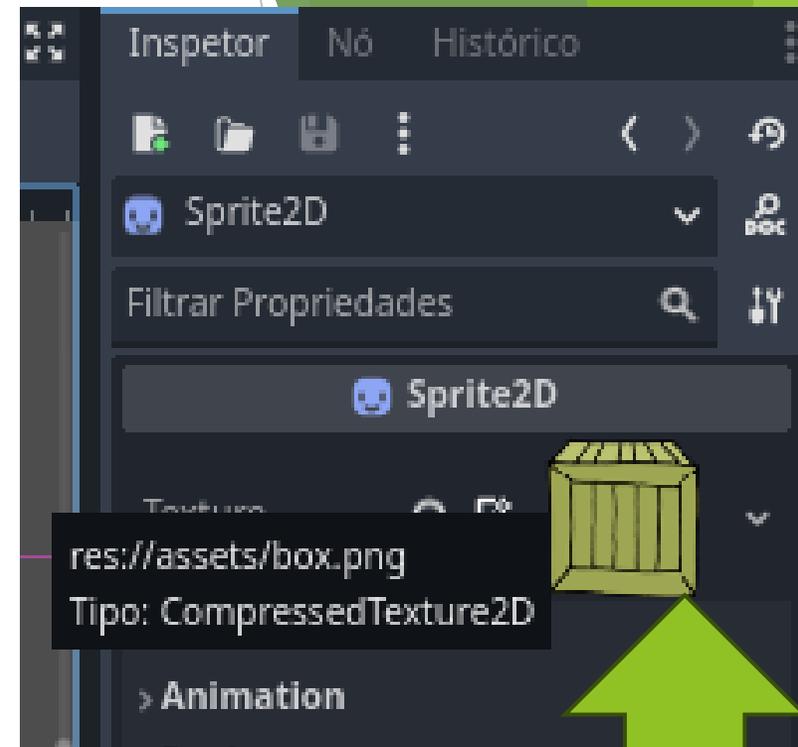
Procure por
Sprite2D

Vamos Criar



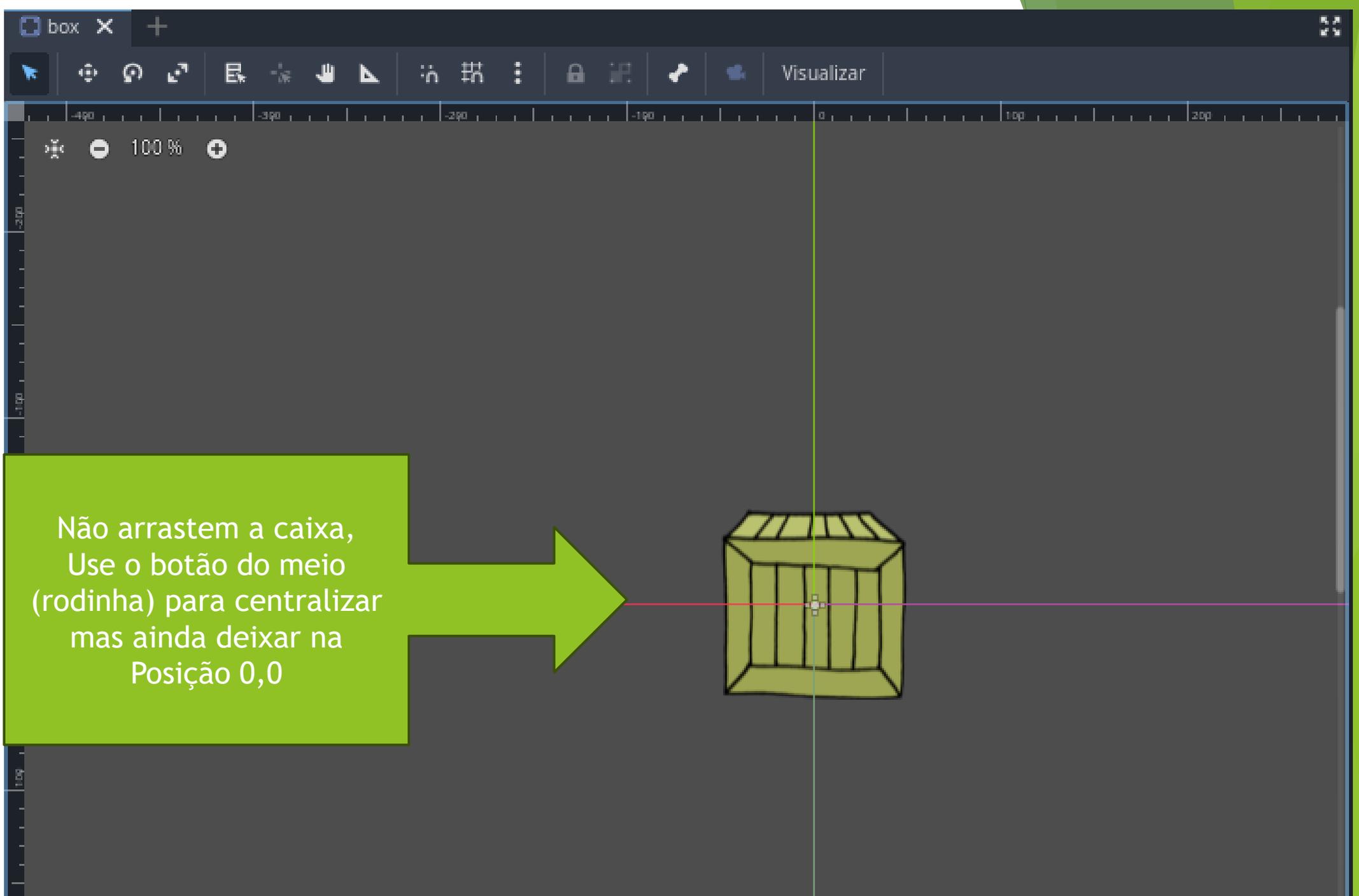


Selecione a figura Box
E arraste até o Texture

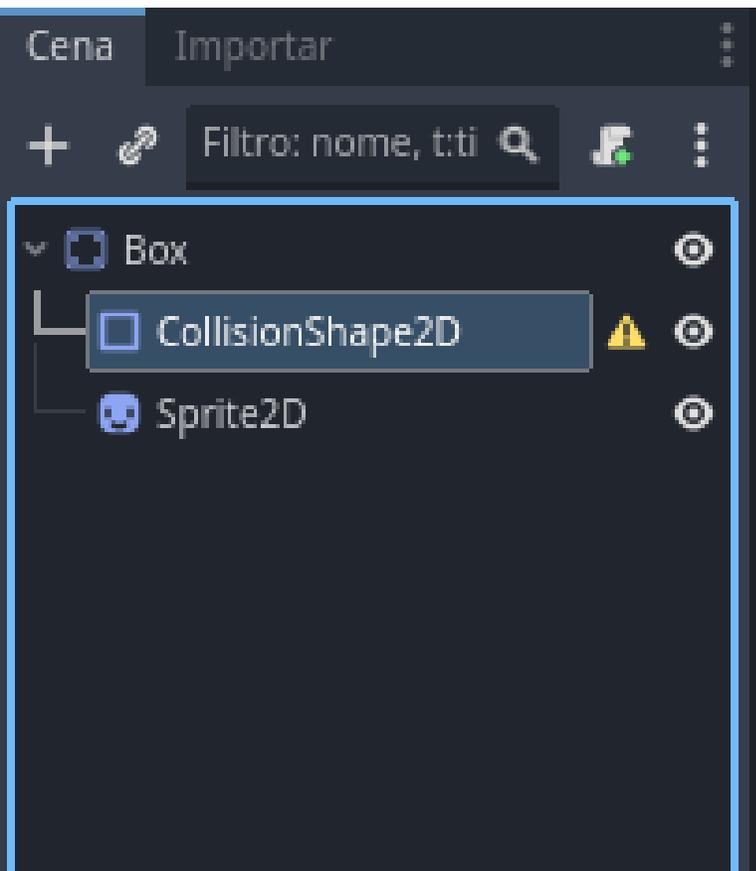




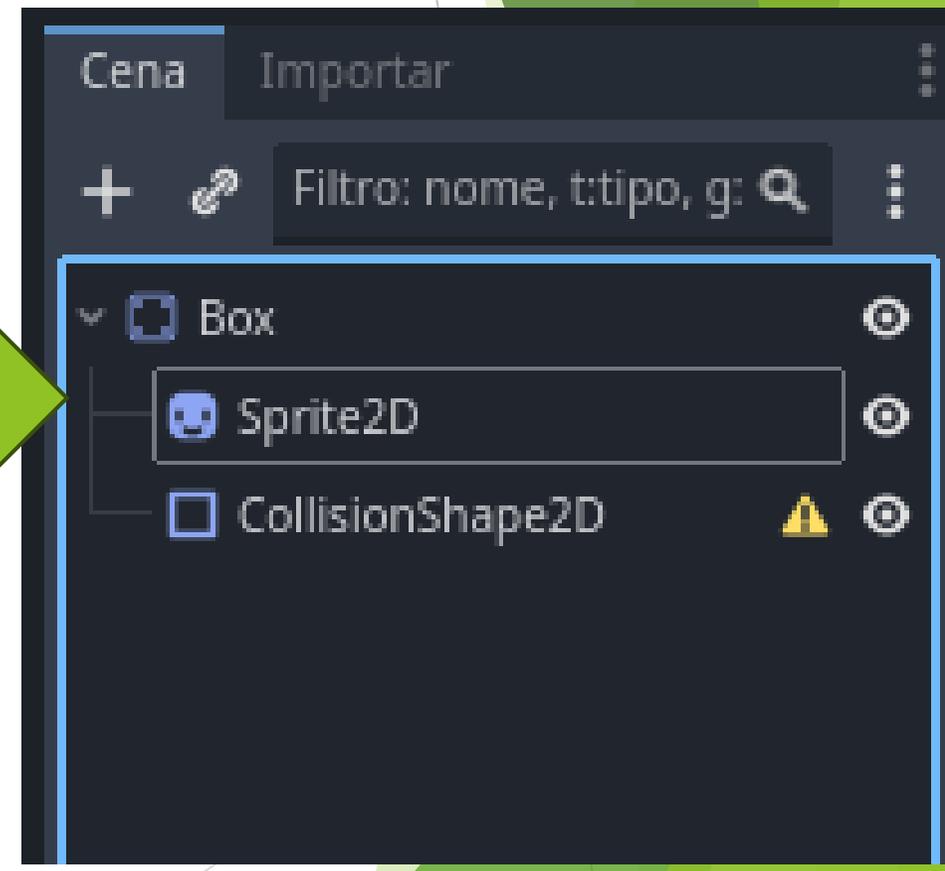
Veja o Resultado

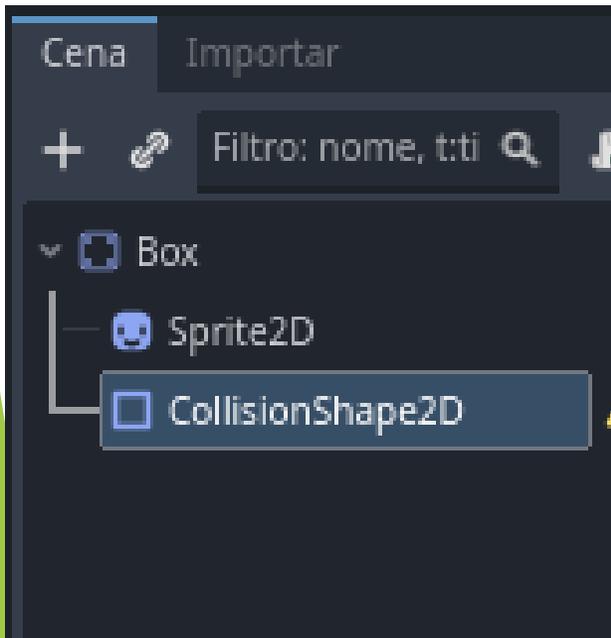


Não arrastem a caixa,
Use o botão do meio
(rodinha) para centralizar
mas ainda deixar na
Posição 0,0

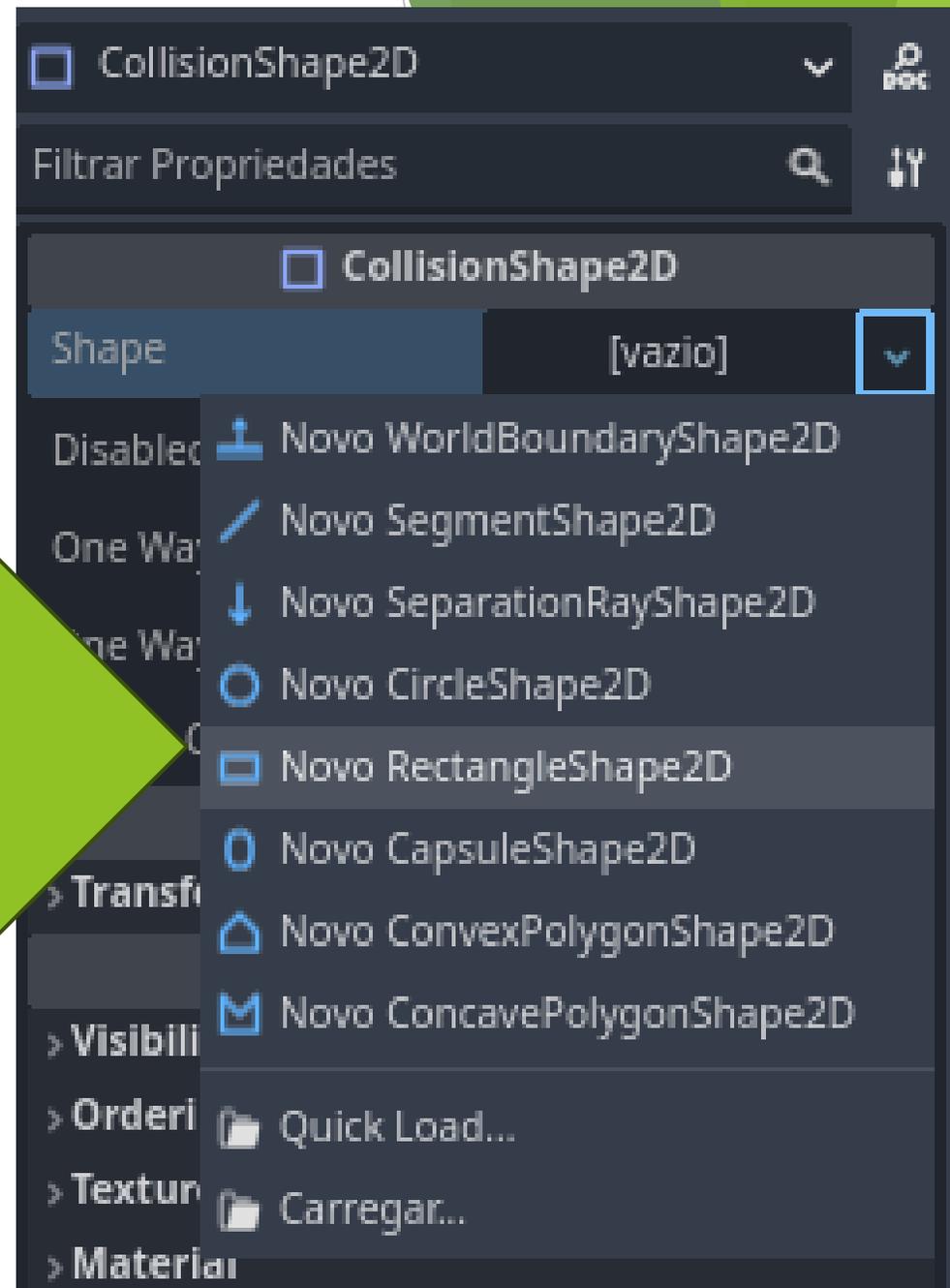


Coloque o Collision
para baixo
Para mudar a ordem
de desenho
Cuidado para que os
dois elementos sejam
filhos de Box

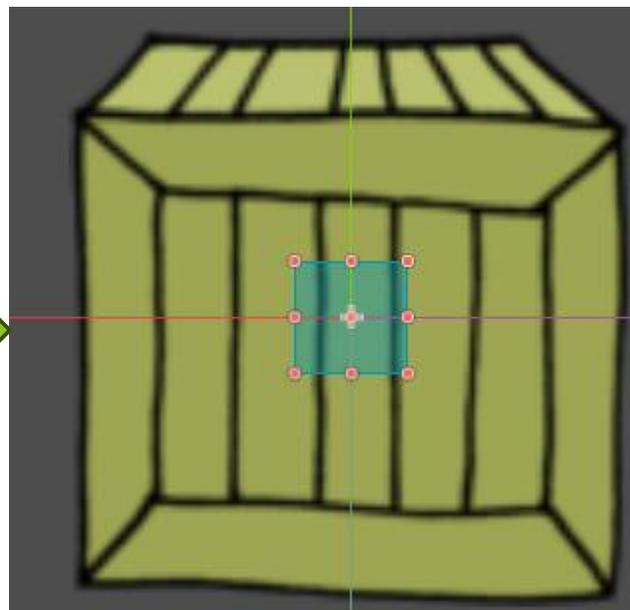




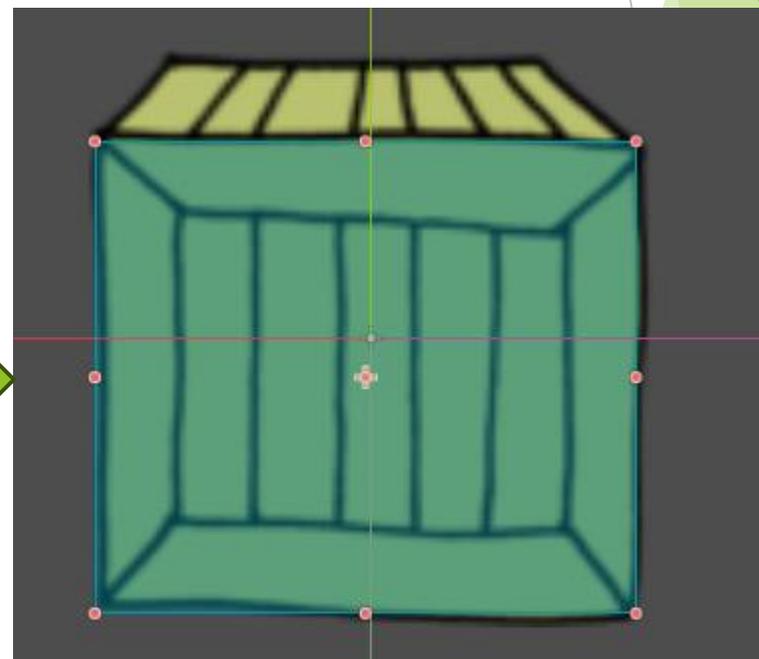
Com o CollisionShaoe2D
selecionado
Vá em Shape, clique na
seta para baixo e
selecione
“Novo
RectanguloShape2D”

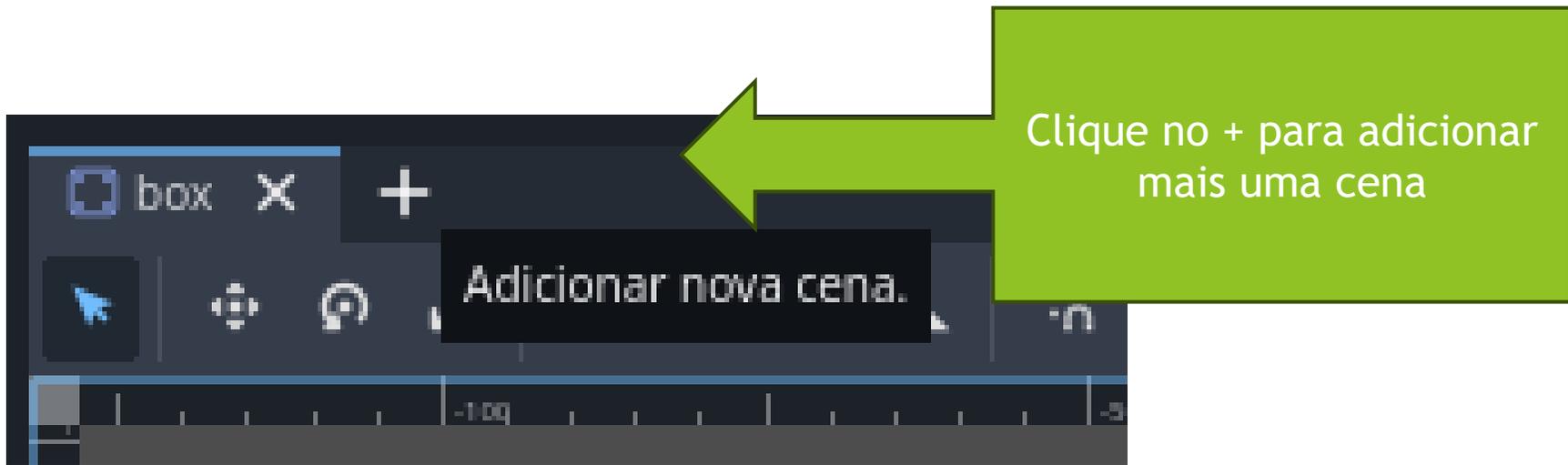


Apareceu o Retângulo

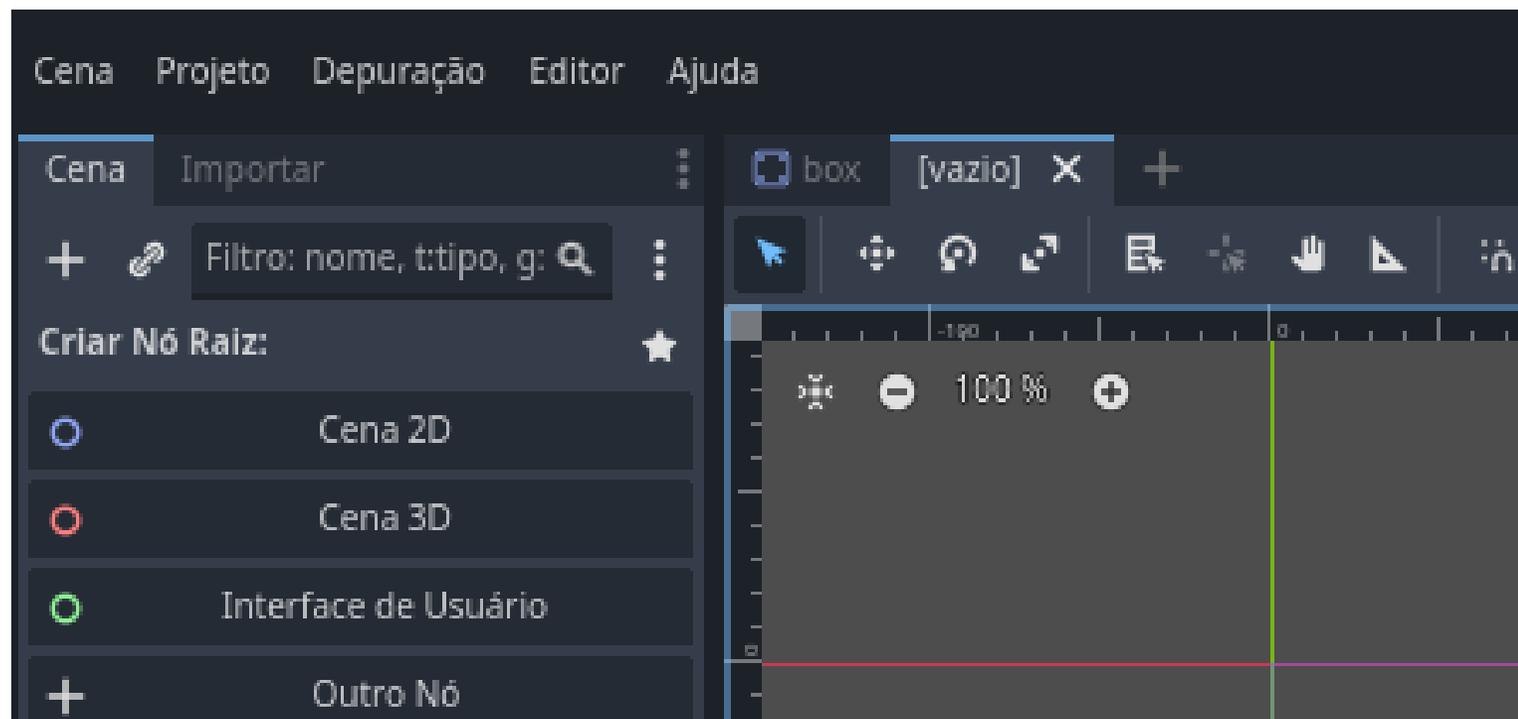


Ajuste os pontos de colisão





h2 - Godot Engine



h2 - Godot Engine

Cena Projeto Depuração Editor Ajuda

Cena Importar



Filtro: n

Cria Add Child Node... (Ctrl+A)
Adicionar/Criar Novo Nó.



Cena 2D



Cena 3D



Interface de Usuário



Outro Nó

Clique no + para adicionar um novo elemento

 Criar Novo Node

CharacterBody2D

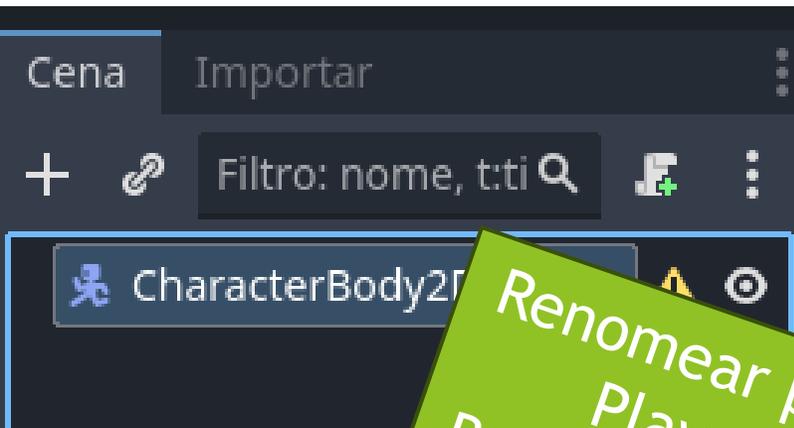
Favoritos:

Pesquisar:

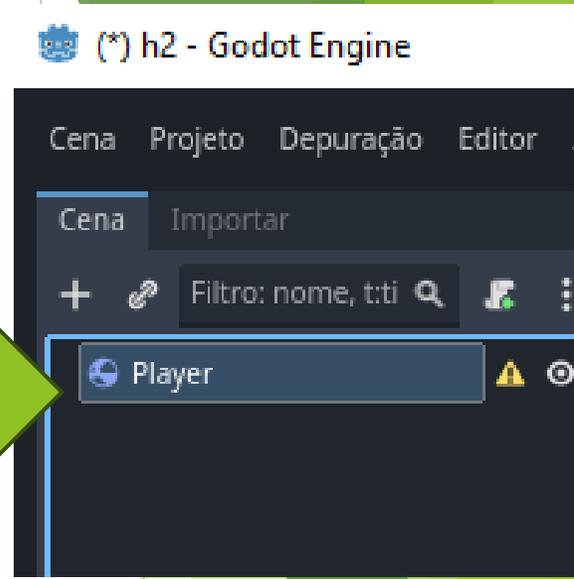
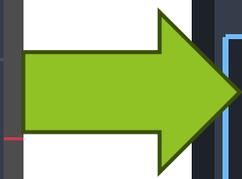
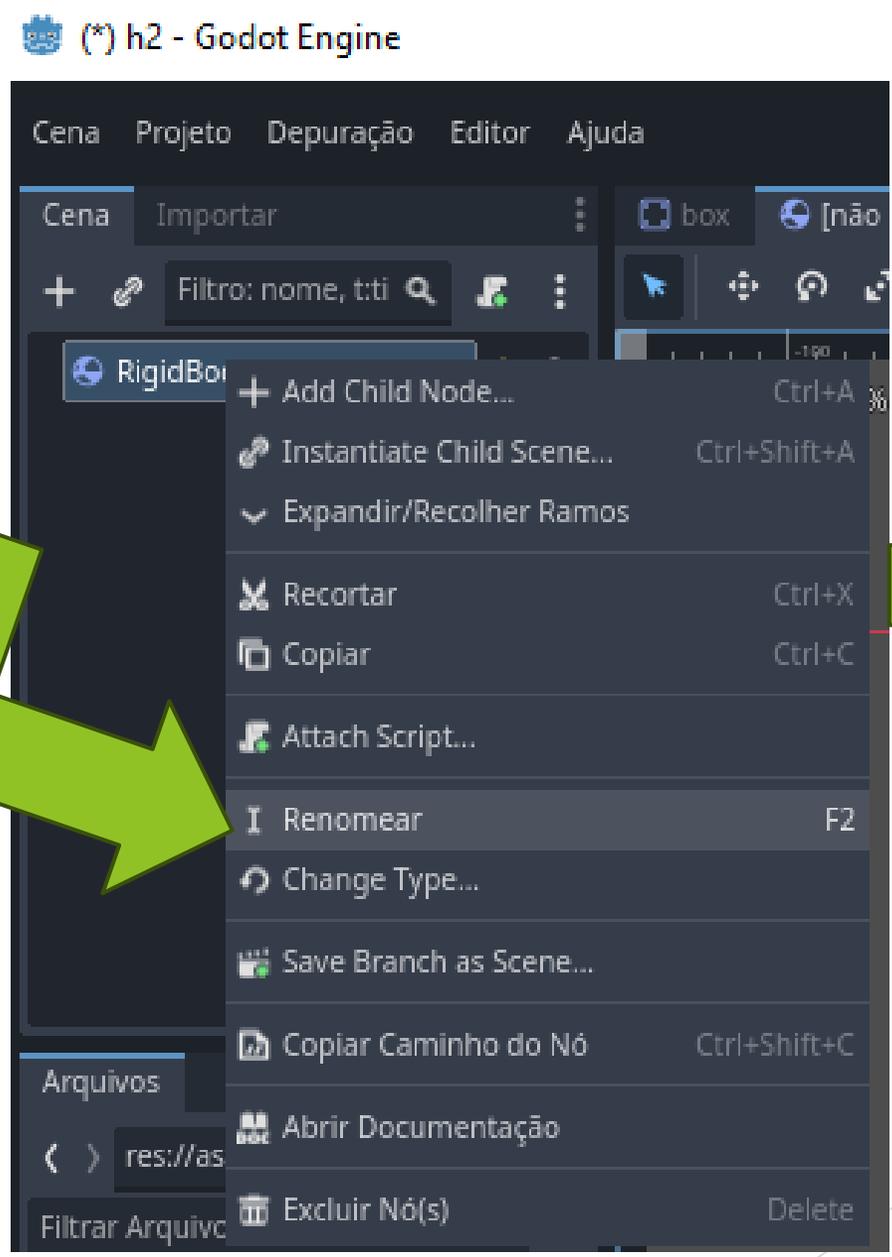
Char

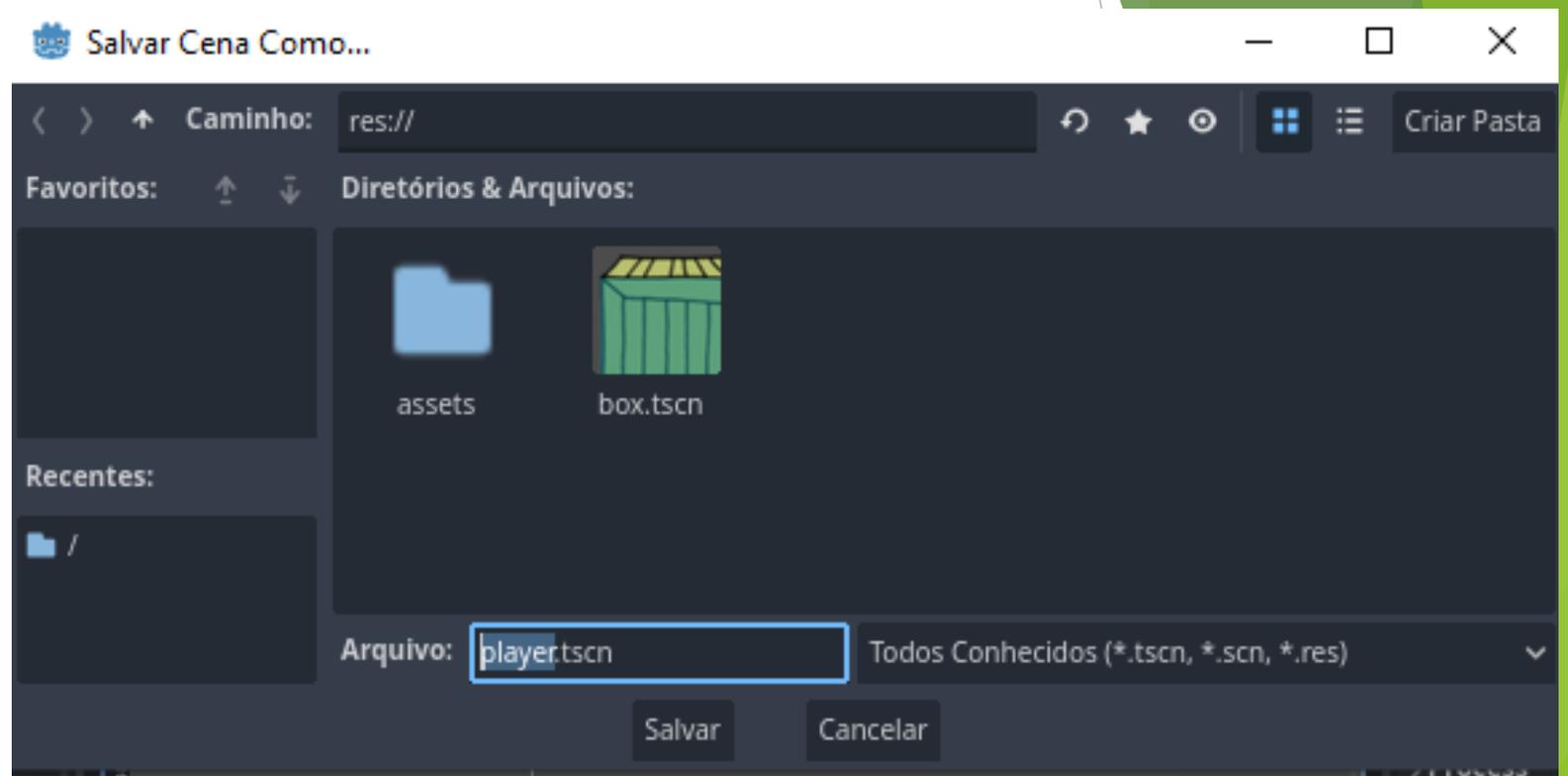
Correspondências:

- Node
 - CanvasItem
 - Node2D
 - CollisionObject2D
 - PhysicsBody2D
 - CharacterBody2D**
- Node3D



Renomear para Player
Botão direito →
Renomear





Salve a Cena

Cena Importar

Filtro: nome, t:ti

+  Add Child Node... (Ctrl+A)
Adicionar/Criar Novo Nó.



Criar Novo Node

Favoritos:

Recentes:

-  RigidBody2D
-  Sprite2D
-  CollisionShape2D
-  StaticBody2D

Pesquisar: animeted

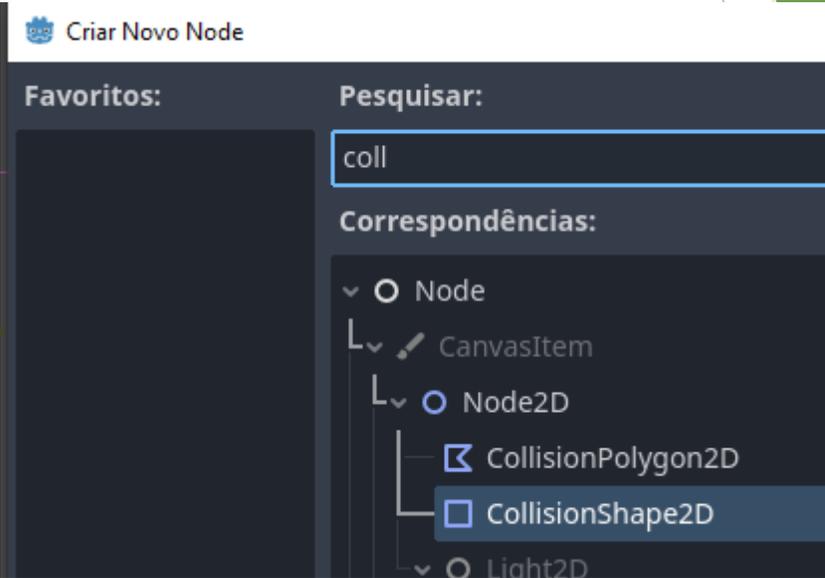
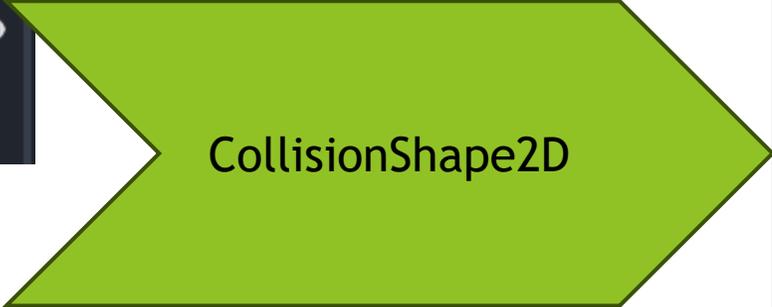
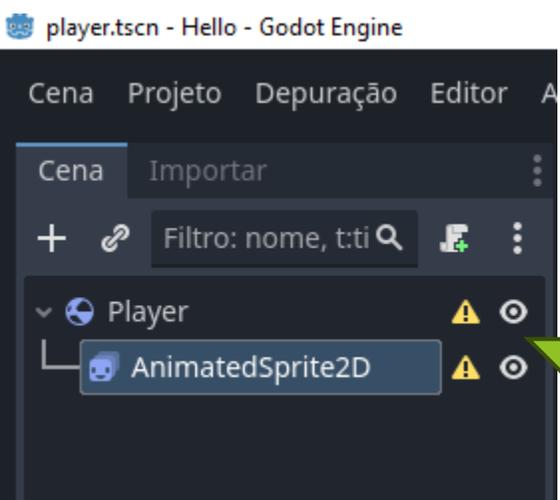
Correspondências:

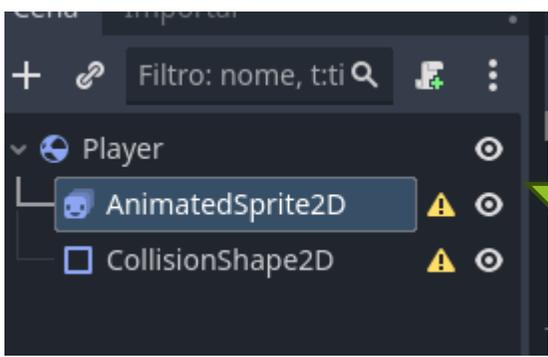
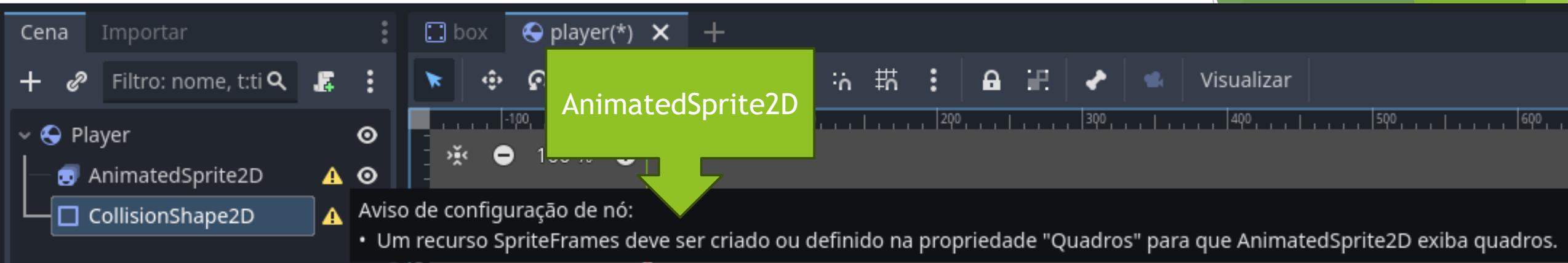
- Node
 - CanvasItem
 - Node2D
 - AnimatedSprite2D**
- Node3D
 - VisualInstance3D
 - GeometryInstance3D
 - SpriteBase3D
 - AnimatedSprite3D

Descrição:

AnimatedSprite2D: Sprite node that contains multiple textures as frames to play for animation.

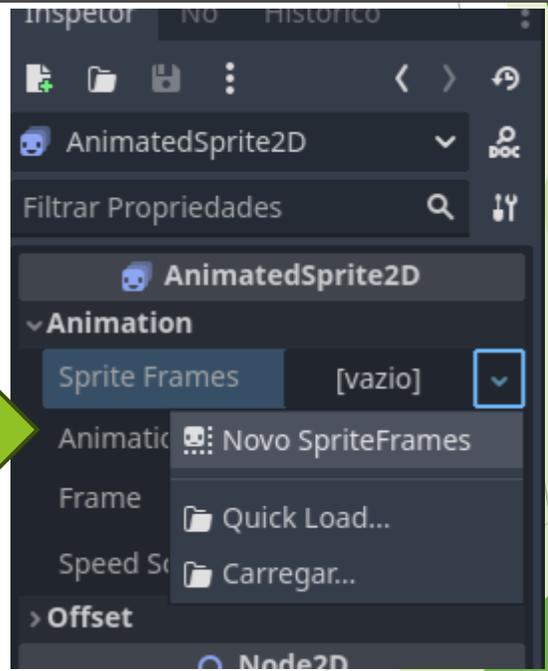
Criar **Cancelar**





Sumiu o warning
Do
animatedSprite
2D

Selecione
AnimatedSprite2D
E em propriedades
No campo Sprite
Frames selecione
"Novo SpriteFrames"



Clique nos Frames
para aparecer o
editor de quadros

Inspetor No Historico

AnimatedSprite2D

Filtrar Propriedades

AnimatedSprite2D

Animation

Sprite Frames SpriteFra

Animation default

Frame 0

Speed Scale 1

Offset

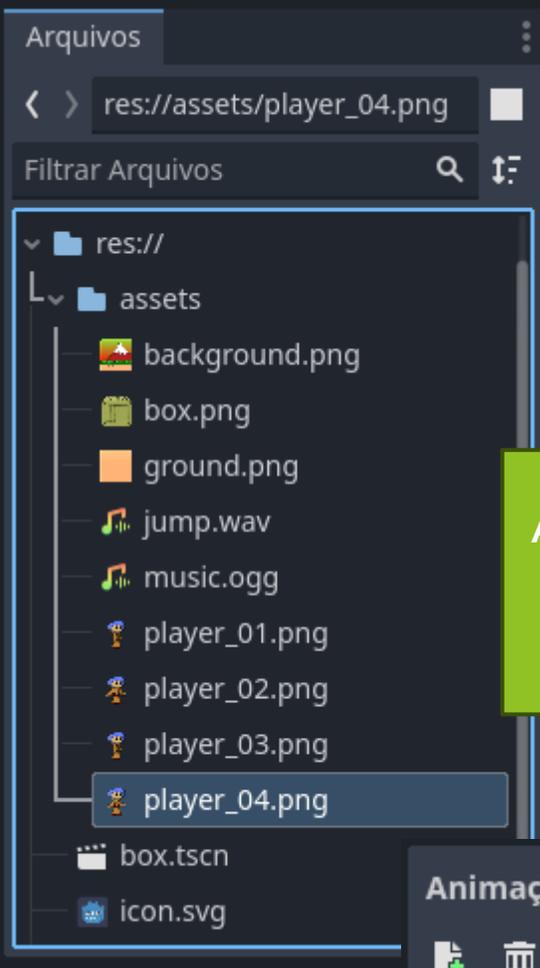
Animações: Quadros da Animação:

5 FPS

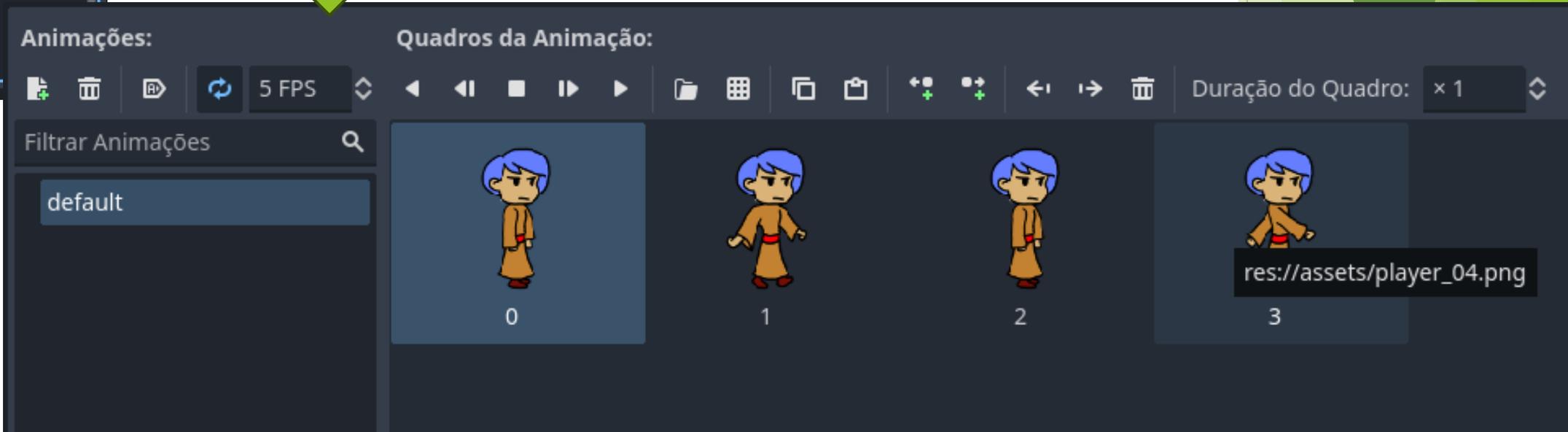
Duração do Quadro: x 1

Filtrar Animações

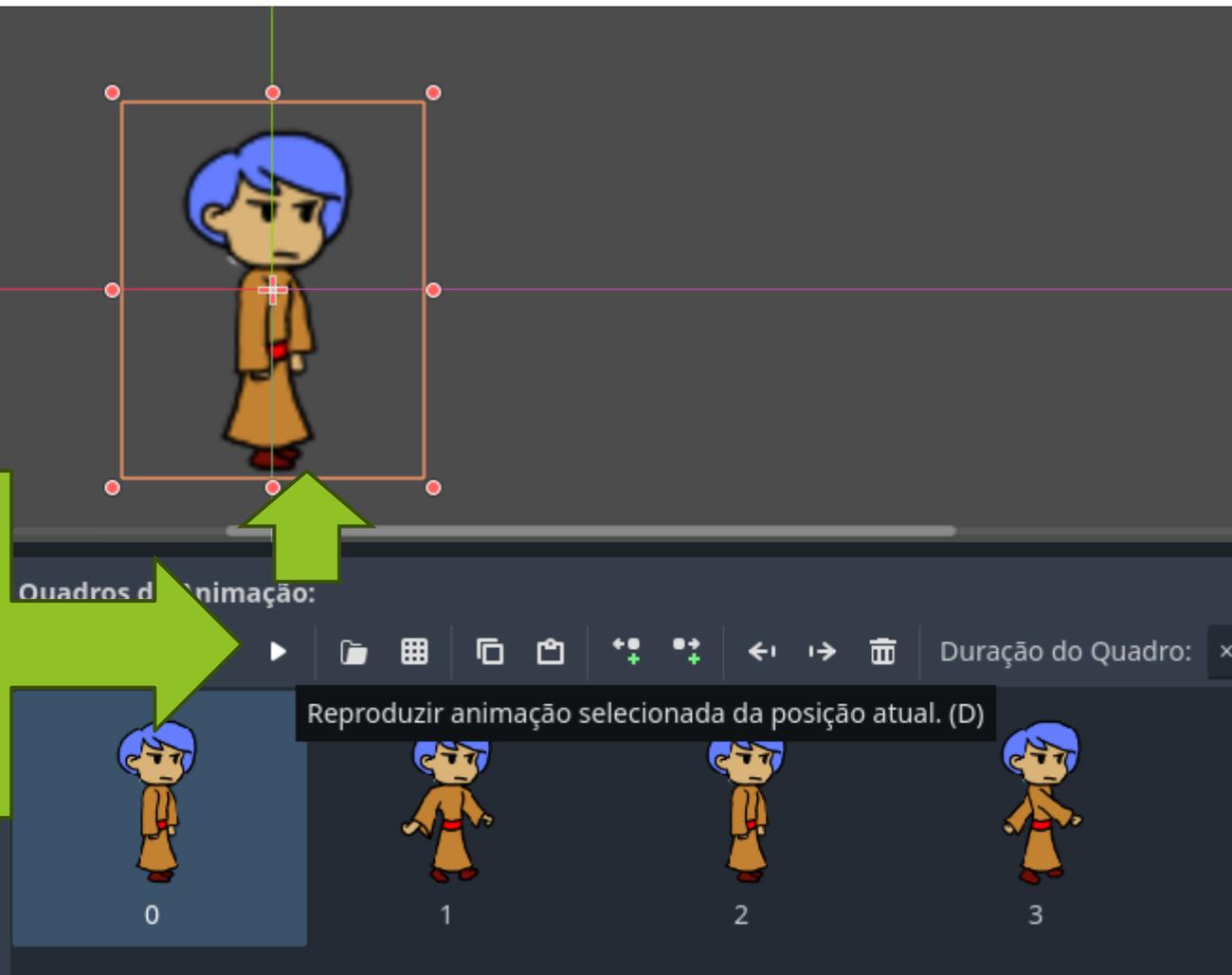
default



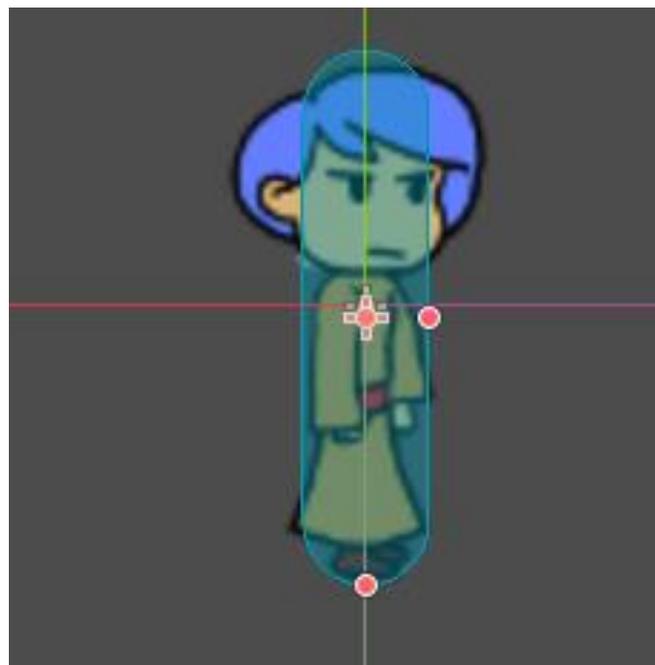
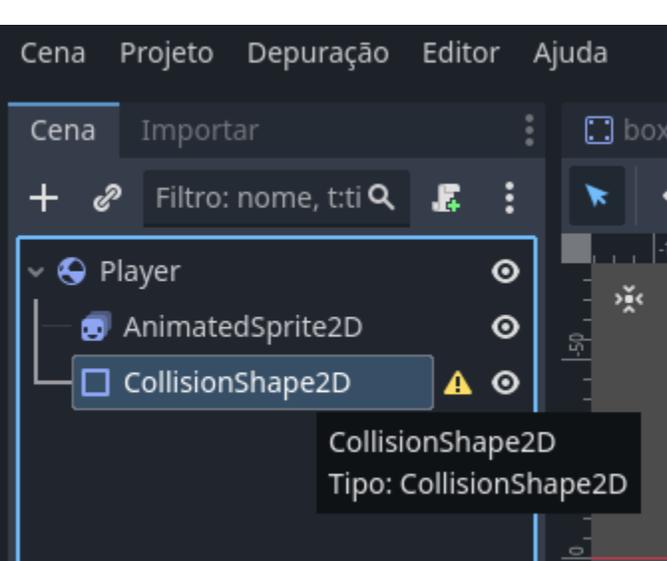
Arraste as imagens do boneca na sequência abaixo para formar a nossa animação



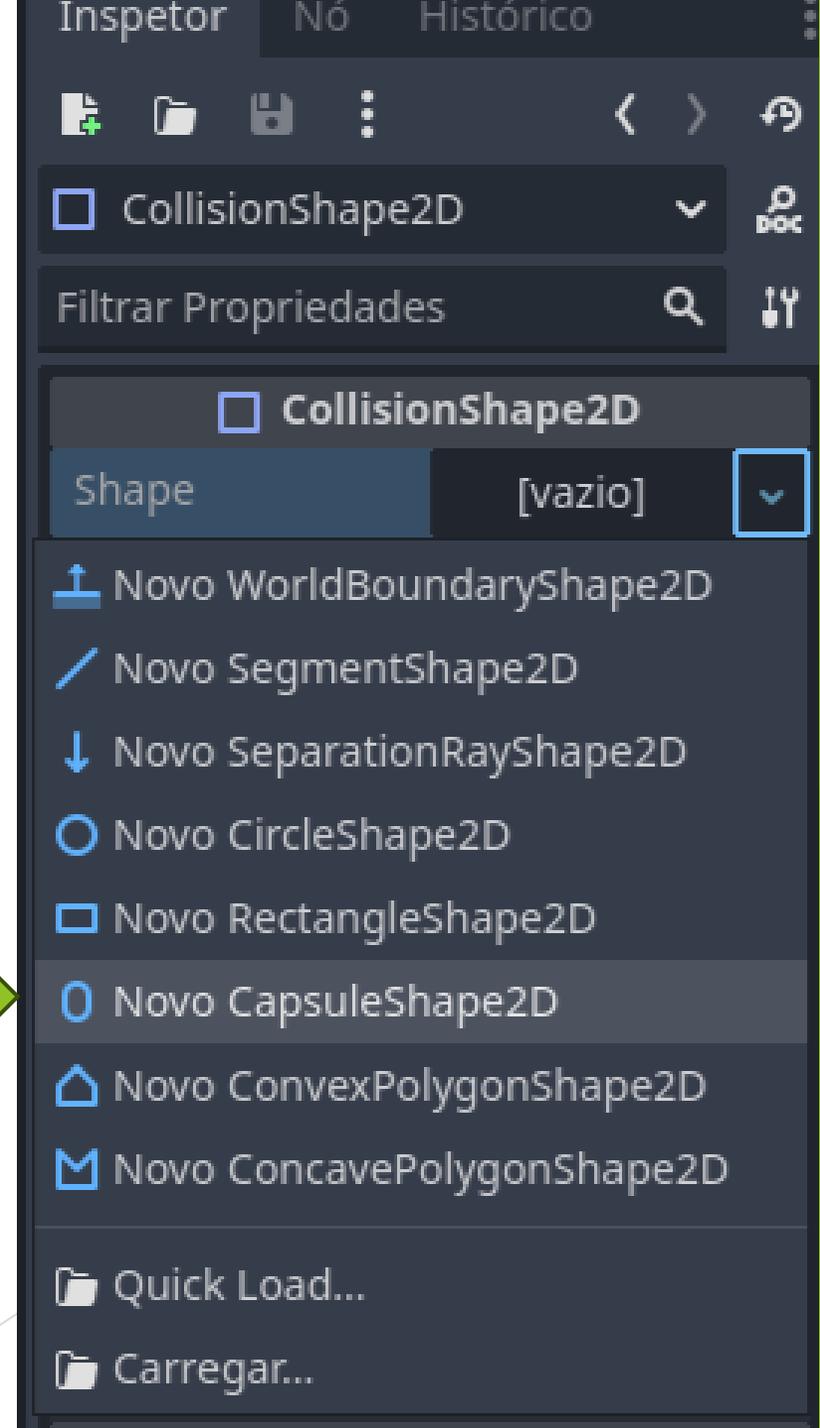
Clique em Play
para ver a
animação

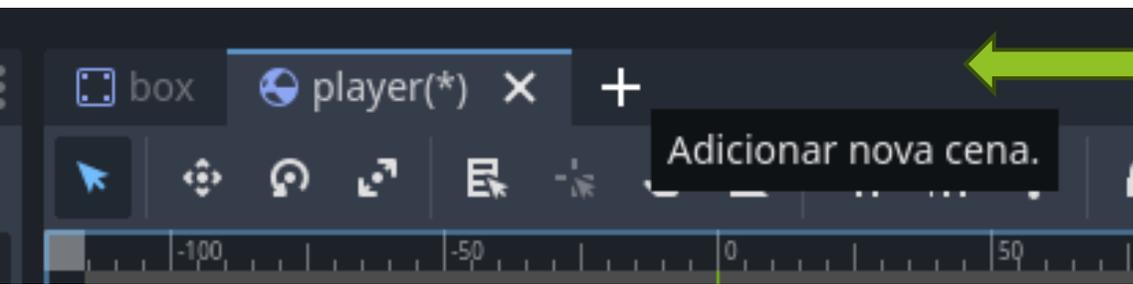


Reproduzir animação selecionada da posição atual. (D)

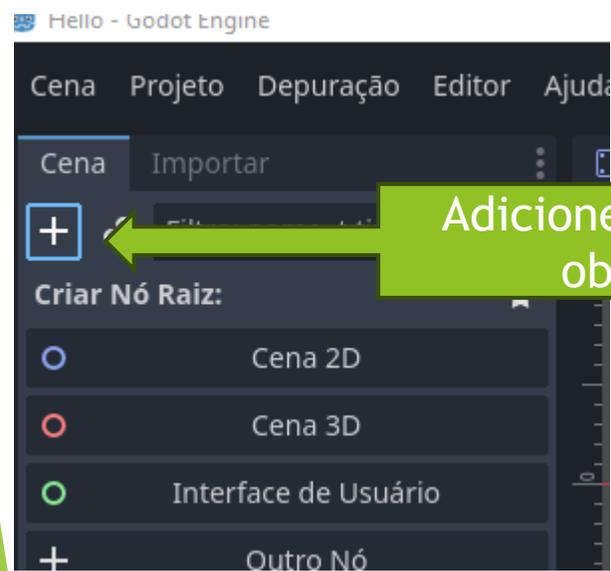


Selecione o
CollisionShape2D
Depois vá em
Shape
E adicione Novo
CapsuleShape2d



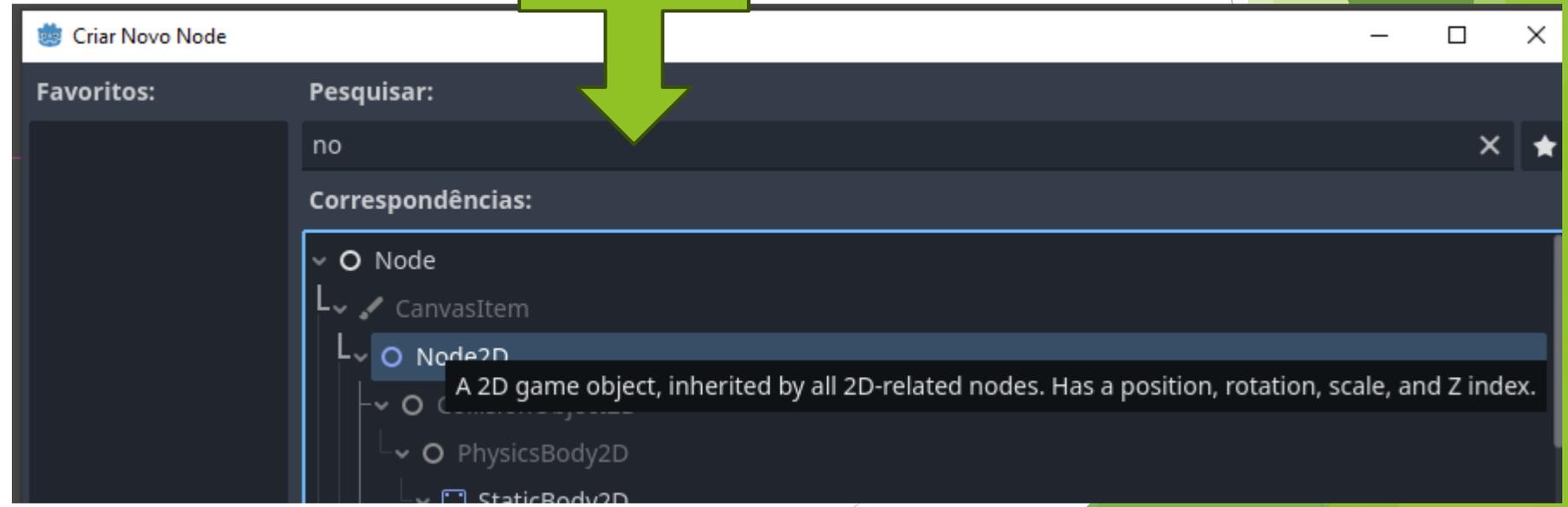


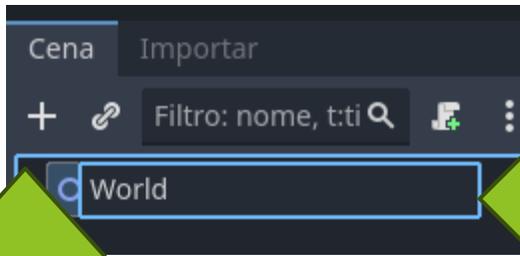
Adicione uma nova Cena



Adicione um novo objeto

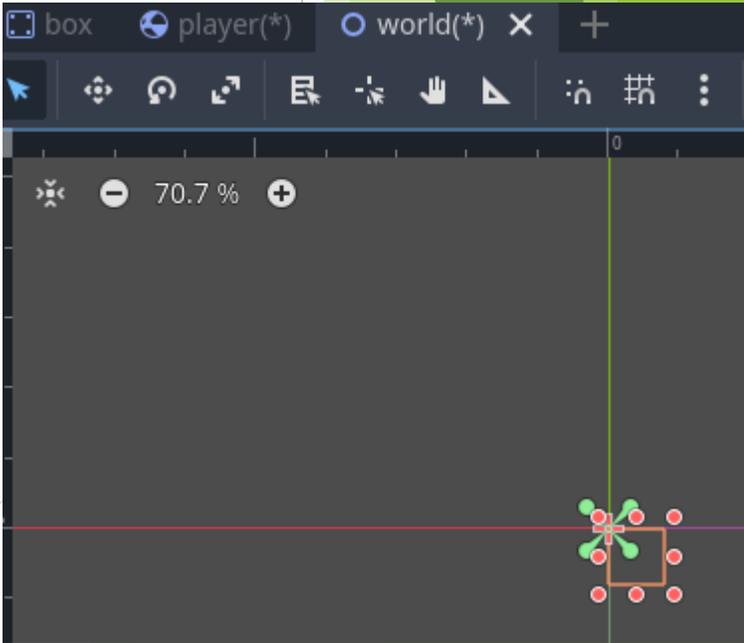
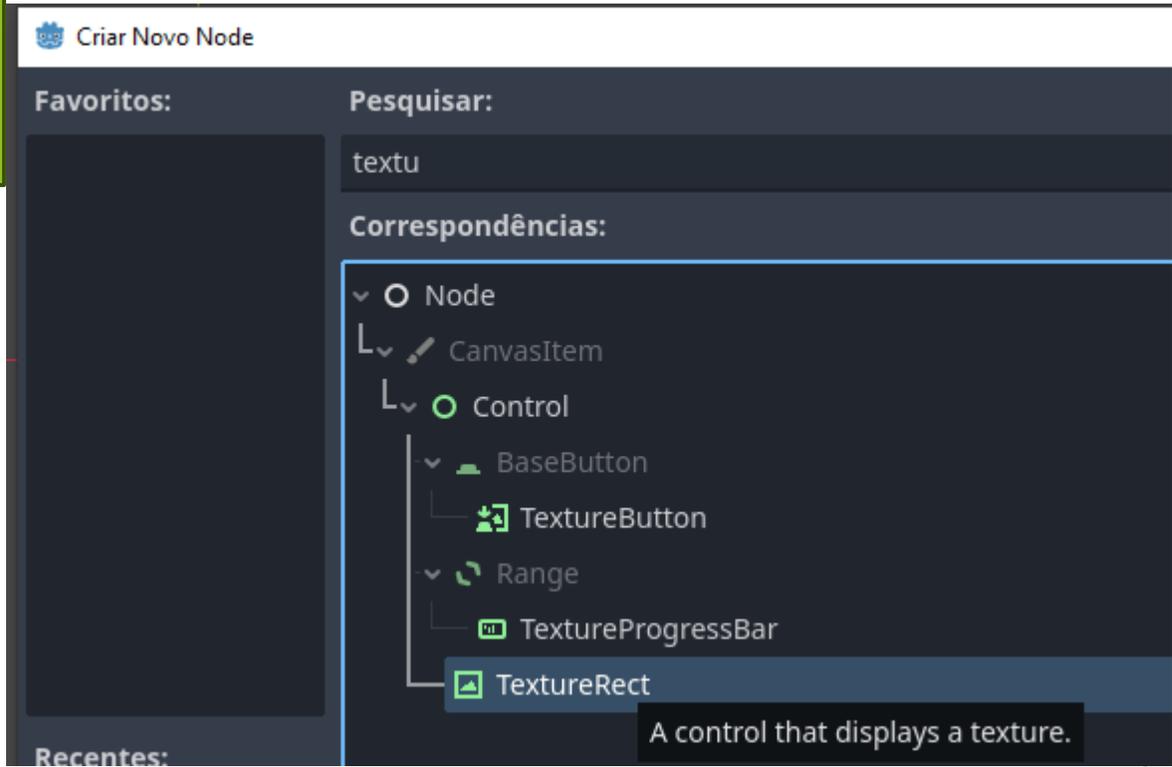
node2D

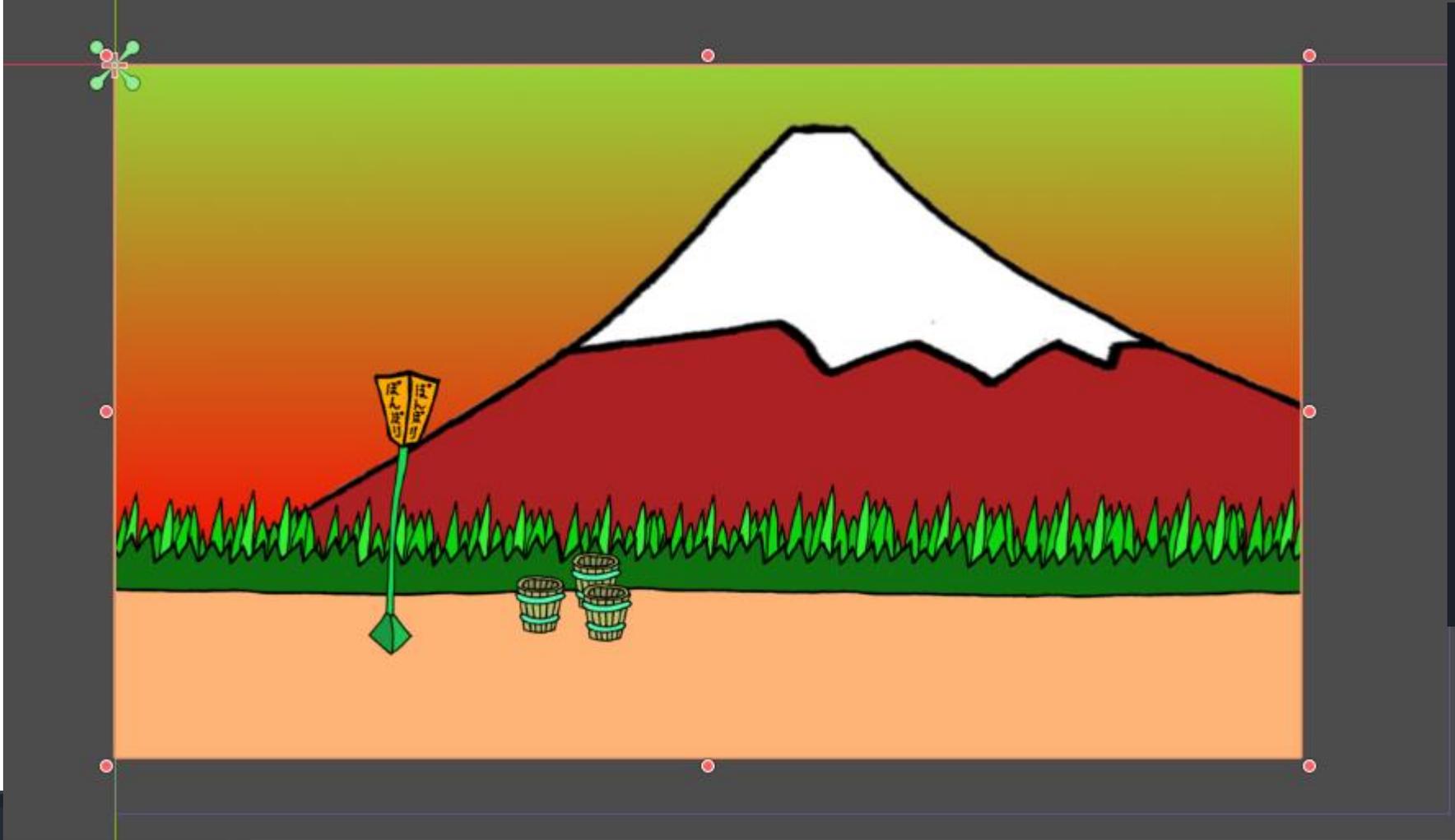




Renomeie para World e salve

Clique no + E adicione um TextureRect





Inspetor Nó Histórico

TextureRect

Filtrar Propriedades

TextureRect

Texture 

Expand Mode Keep Size

Stretch Mode Scale

Flip H Ativo

Flip V Ativo

Control

Arquivos

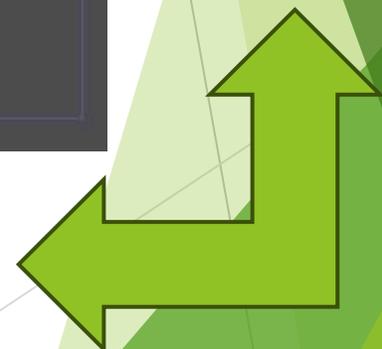
res://assets/background.pn

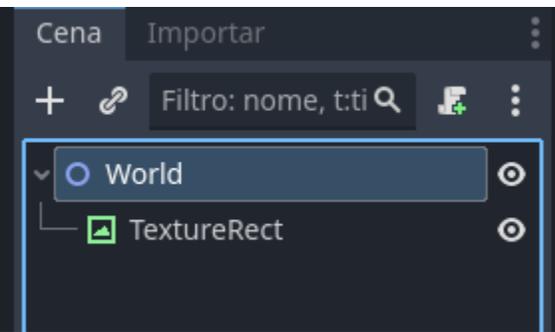
Filtrar Arquivos

- res://
 - assets
 - background.png
 - box.png

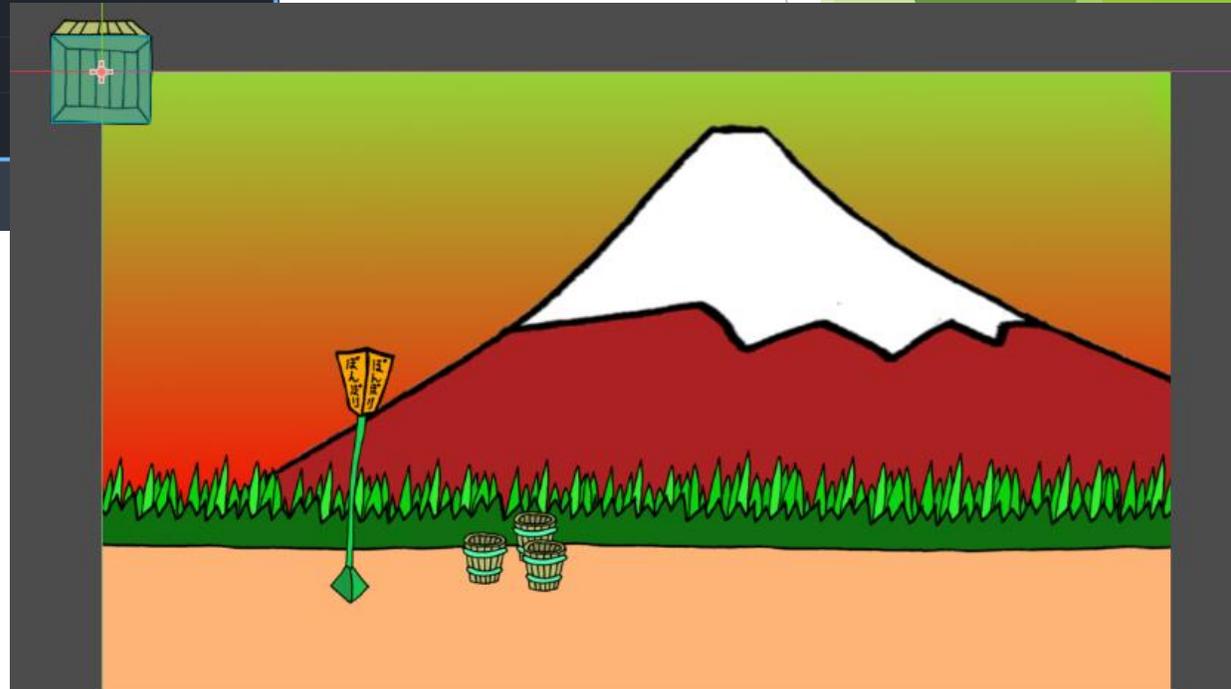
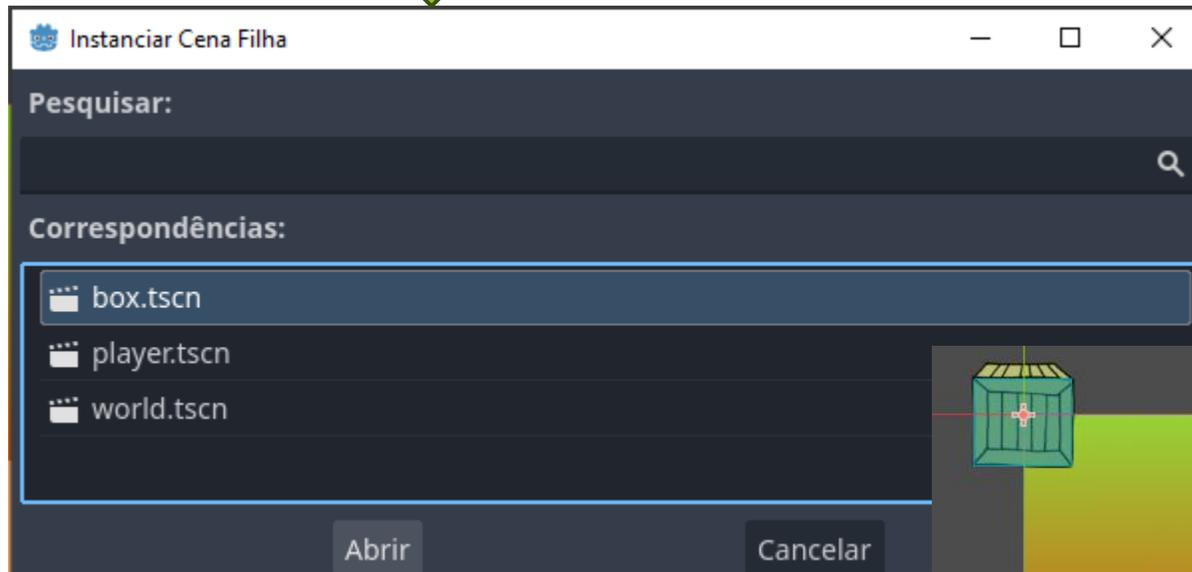
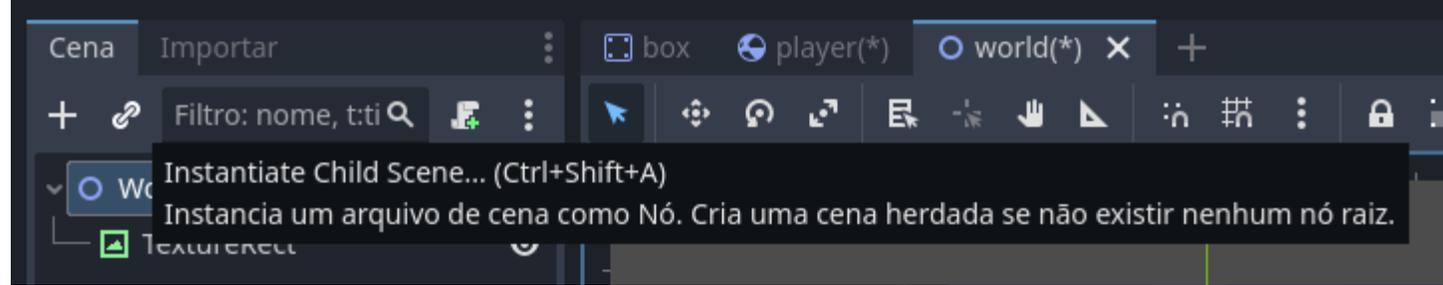


Arraste o background para Texture

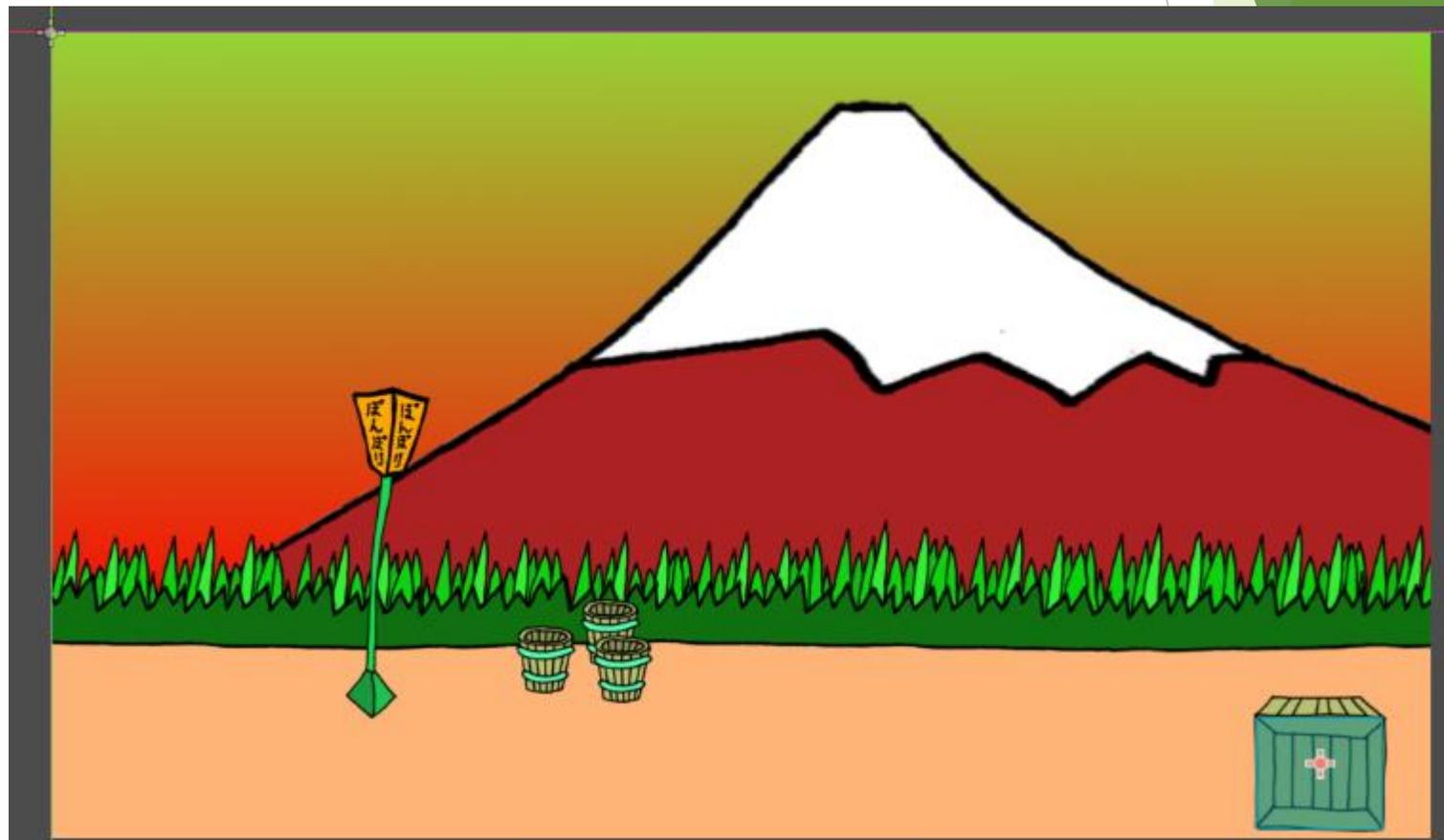




Selecione o World
Clique nas correntes
Selecione box

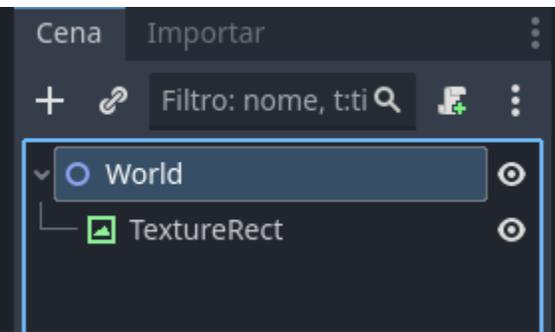


Mova a caixa no
local que deseja
colocar na
cenário
Use Ctrl+D para
duplicar

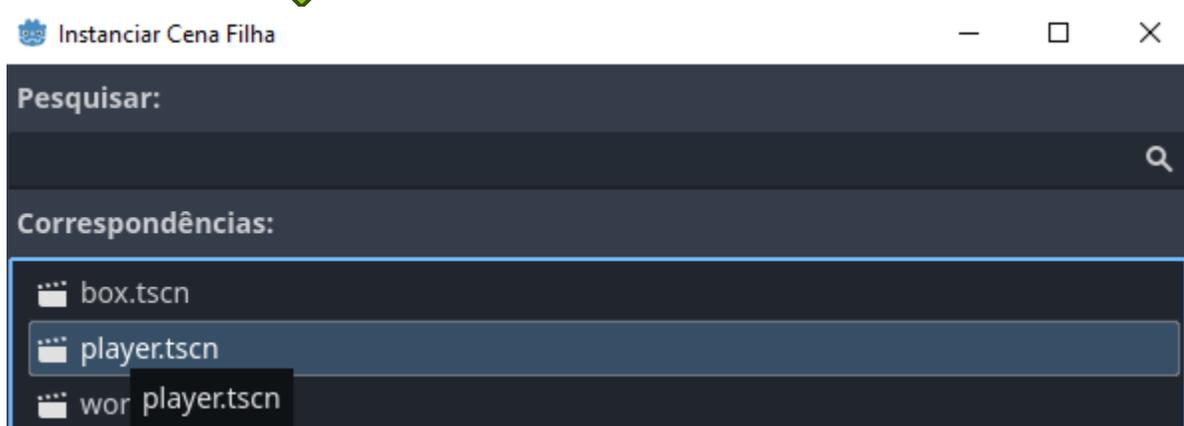
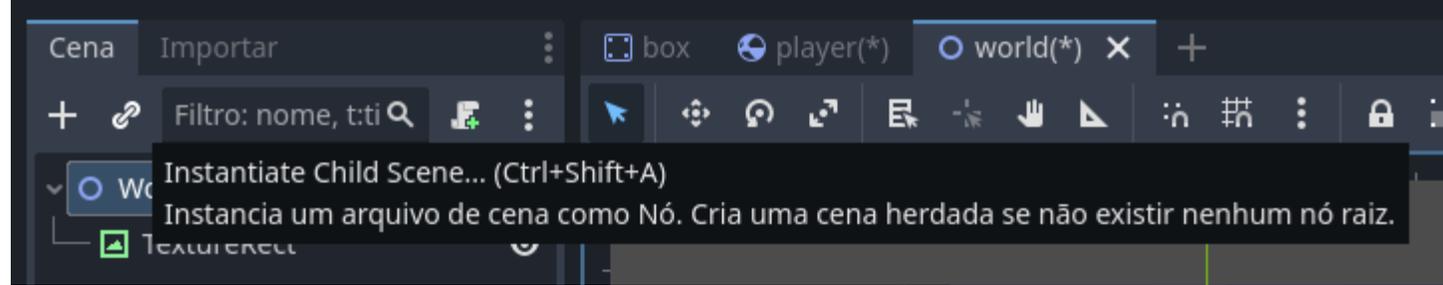


Cada cópia (Ctrl+d) fica em cima da anterior, então é só arrastar e colocar no lugar correto





Selecione o World
Clique nas correntes
Selecione Player





Confirme, Por Favor...



A cena principal não foi definida, selecionar uma?
Você pode alterá-la mais tarde nas "Configurações do Projeto" na categoria 'Aplicação'.

Selecionar

Cancelar

Selecionar Atual

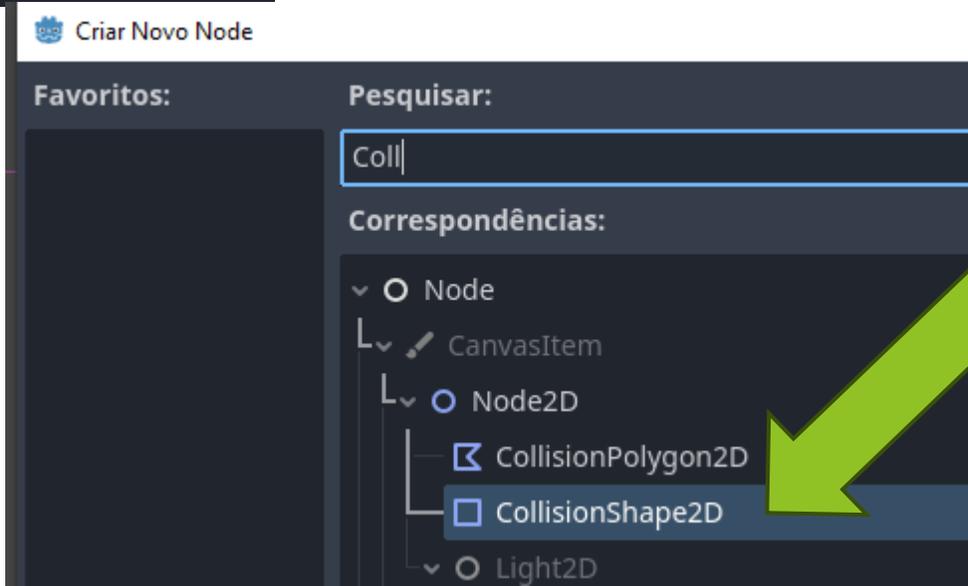
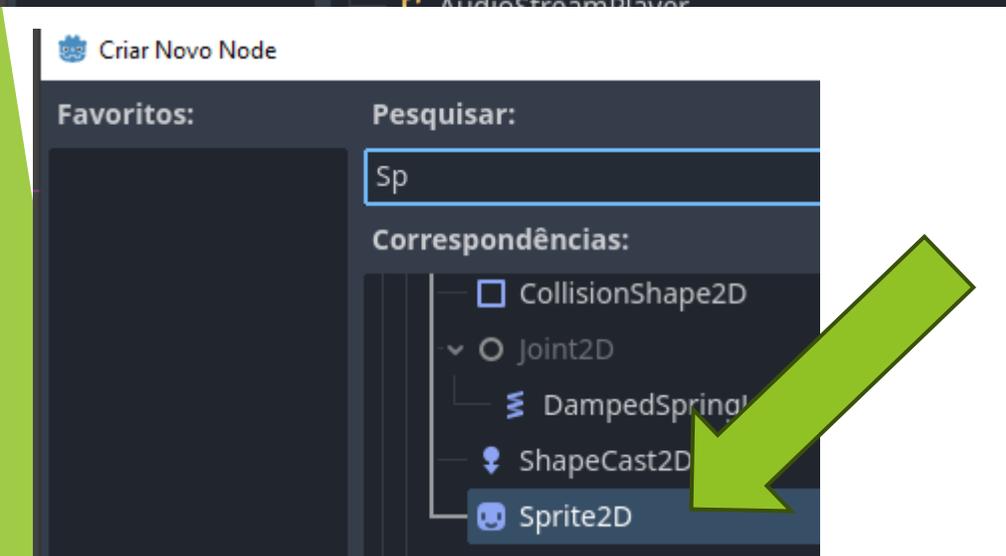
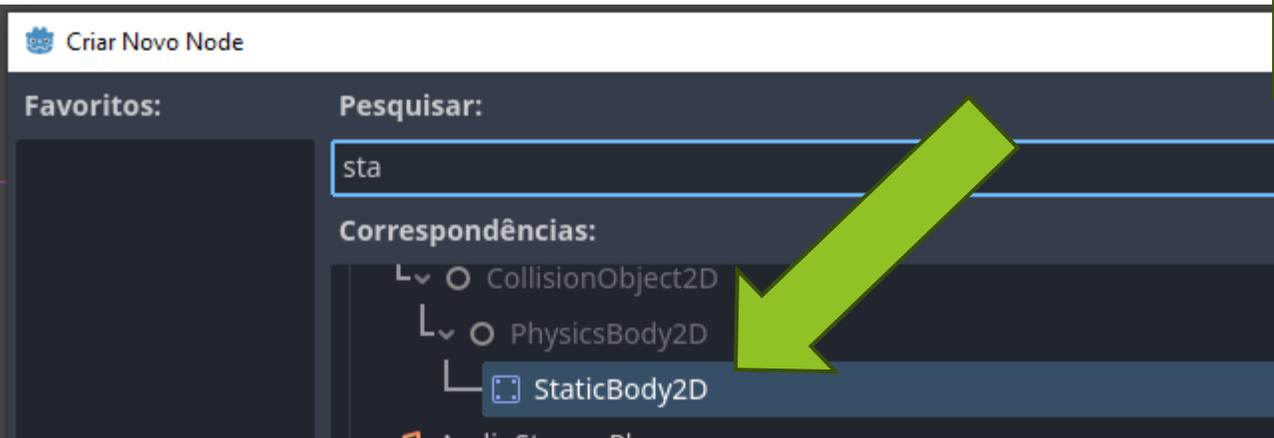
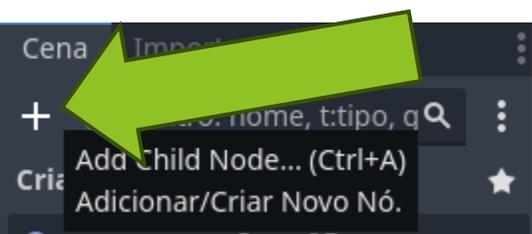
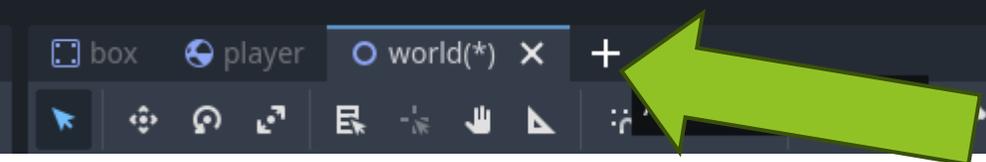


Executar Projeto (F5)
Roda o projeto.

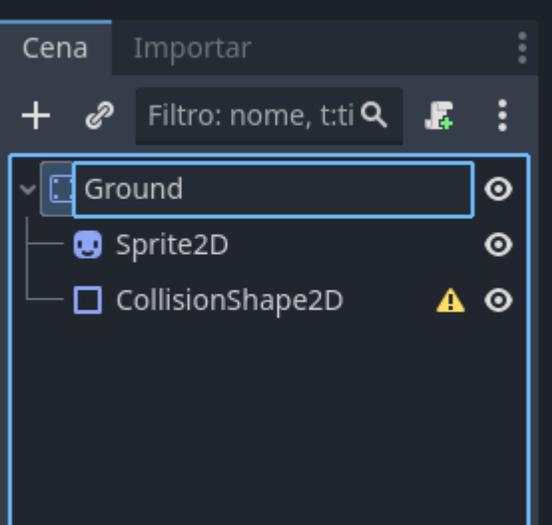
Histór

O personagem simplesmente cai
Porque não tem nada segurando ele

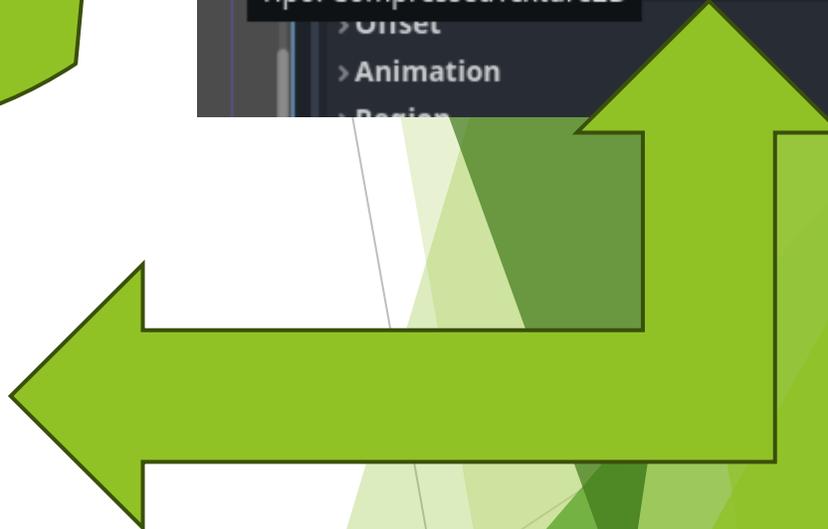
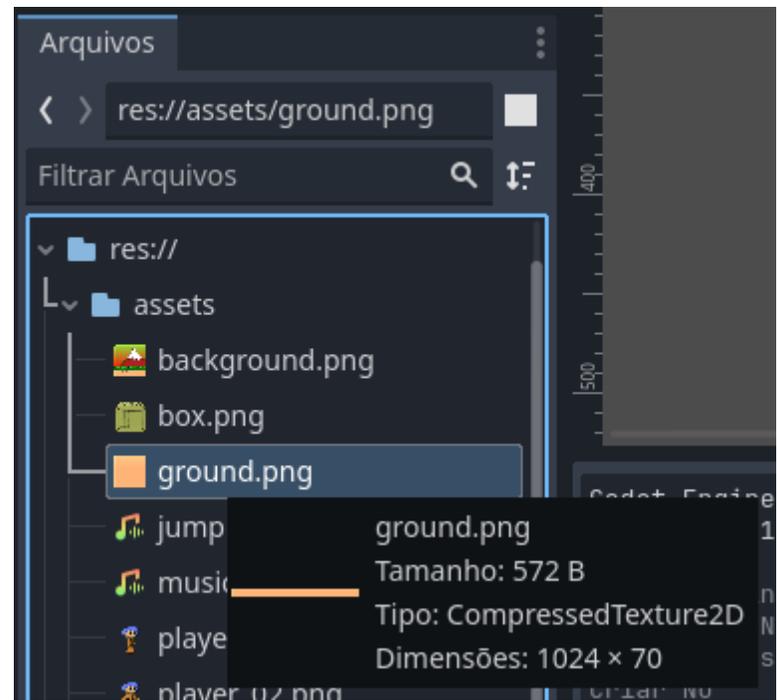
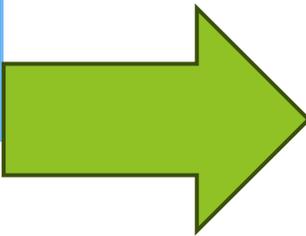
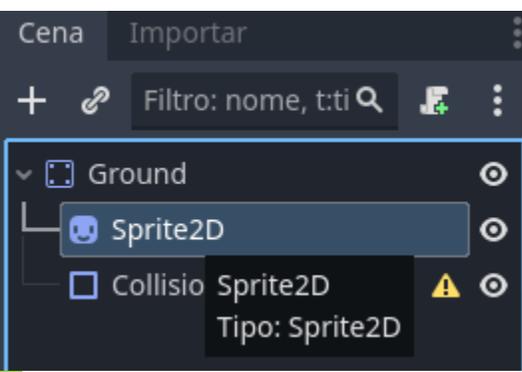
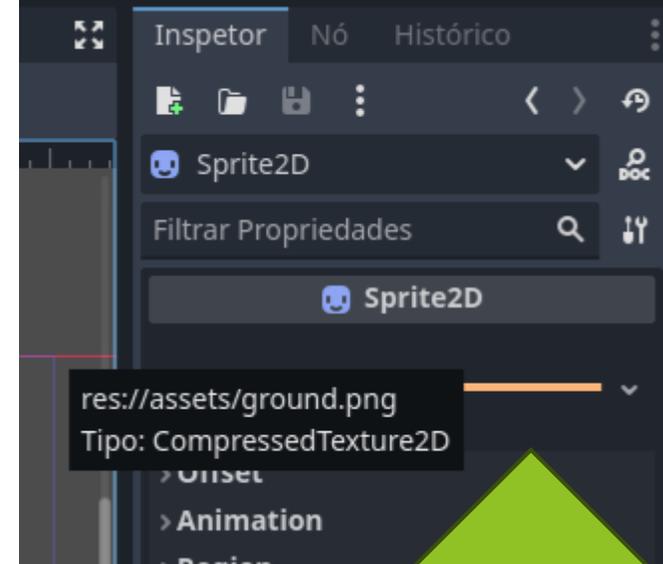


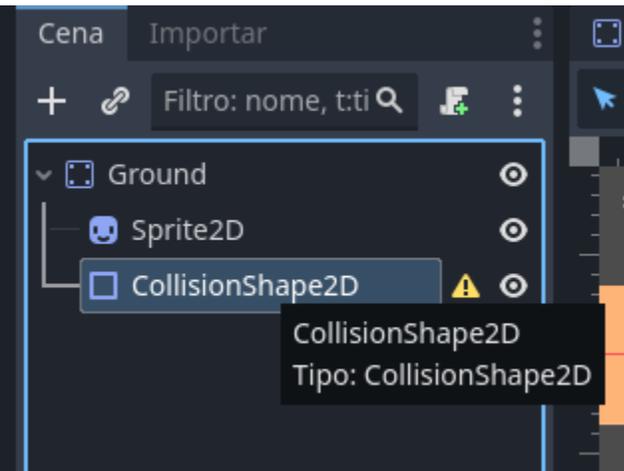
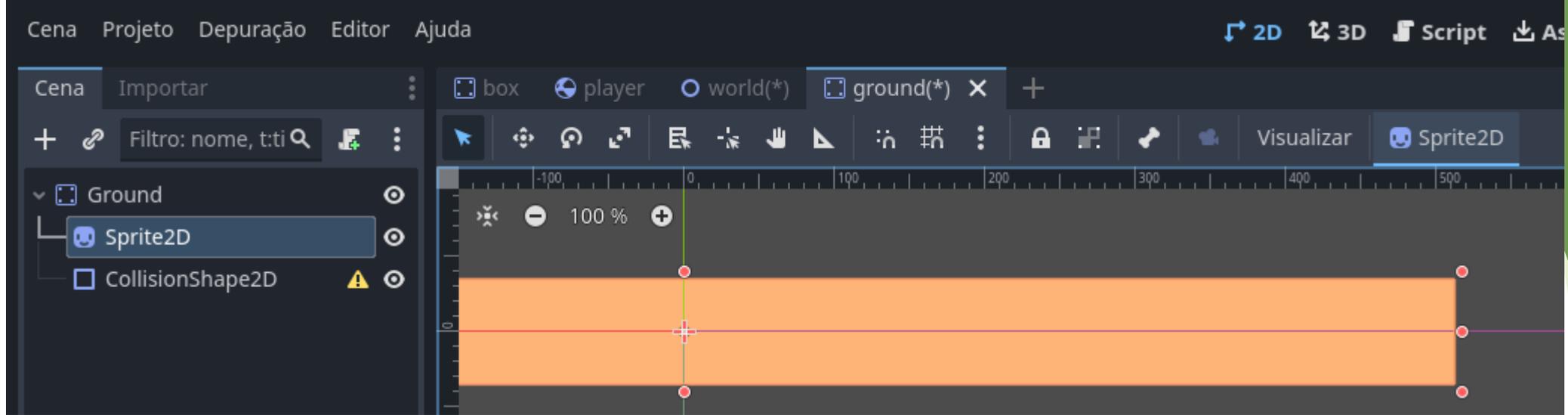


Crie um cena
Crie os filhos
StaticBody2D
Sprite2D
CollisionShape2D

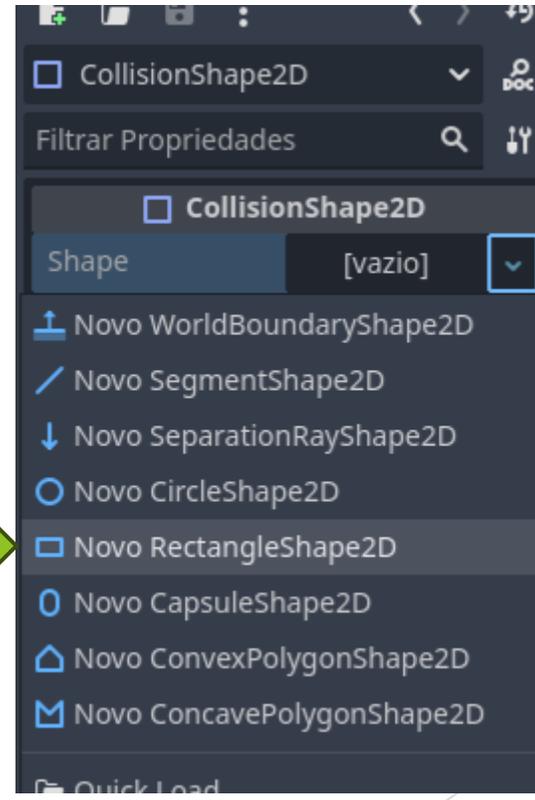


Mude o nome de StaticBody para Ground e Salve
Selecione o Sprite
Arraste a figura ground.png para a propriedade Texture

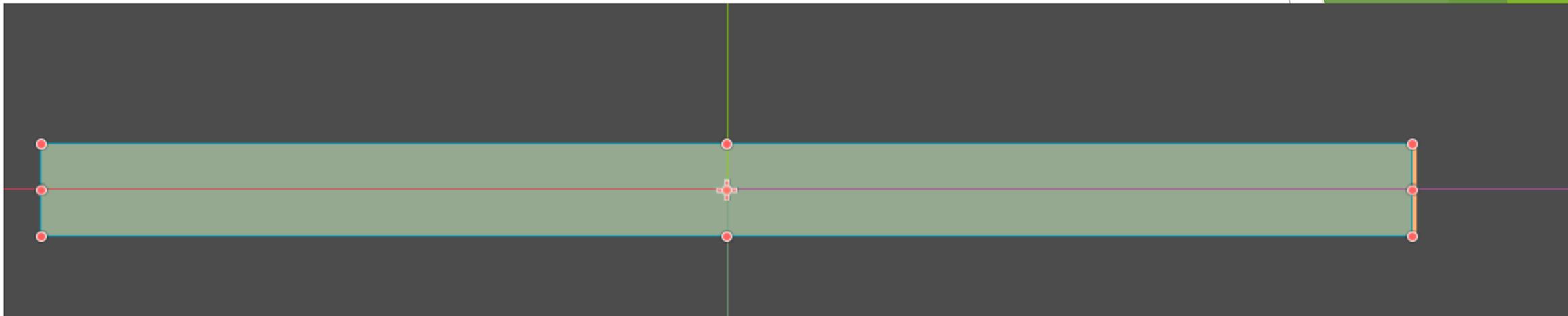


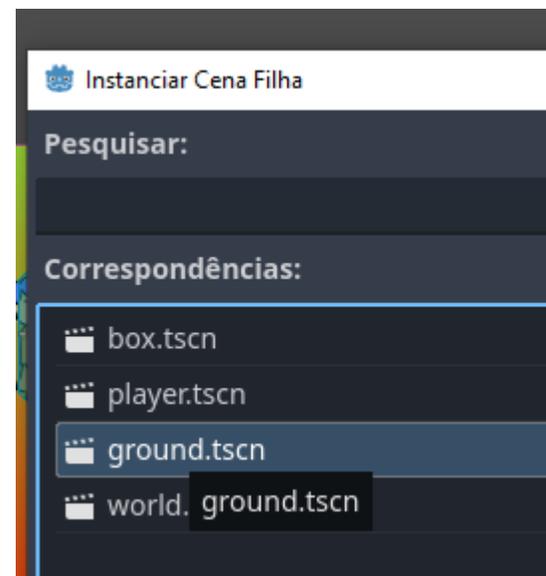
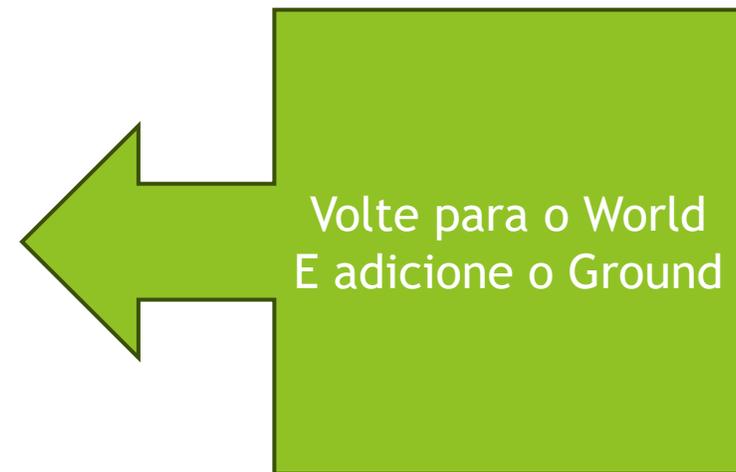
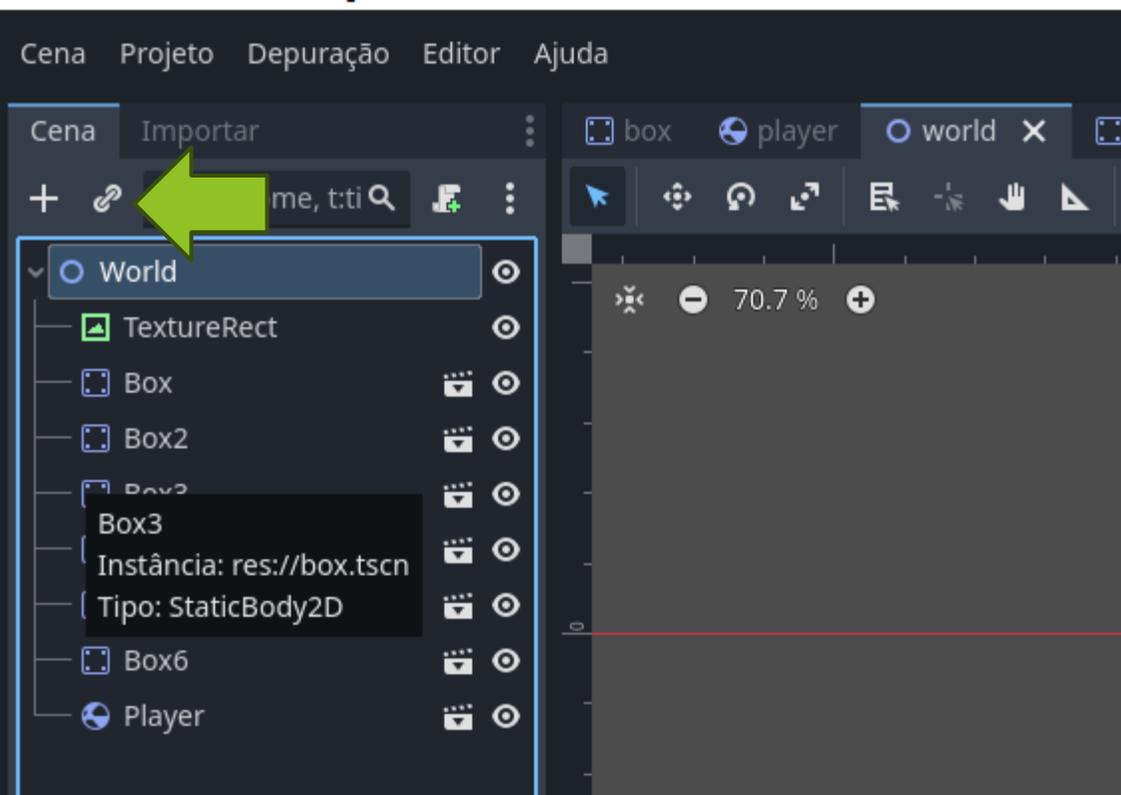


Crie Novo RectangleShape2D



Ajuste o CollisionShape de acordo com a figura
Salve





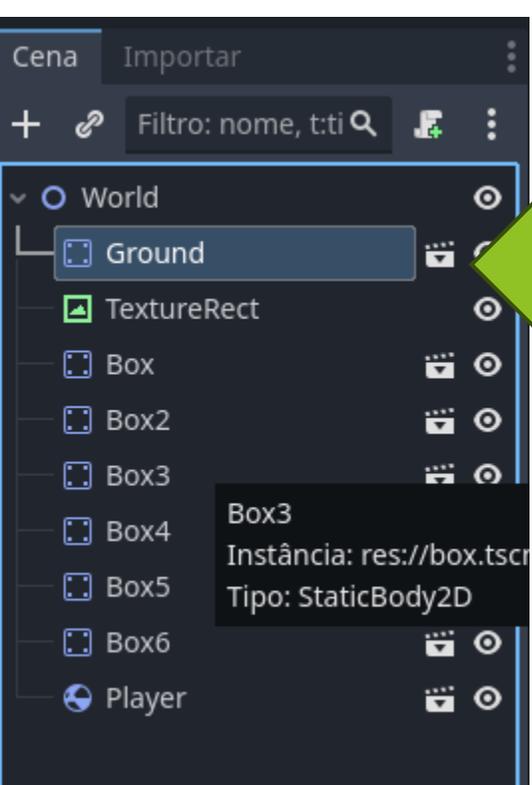


Posicionamos o chão

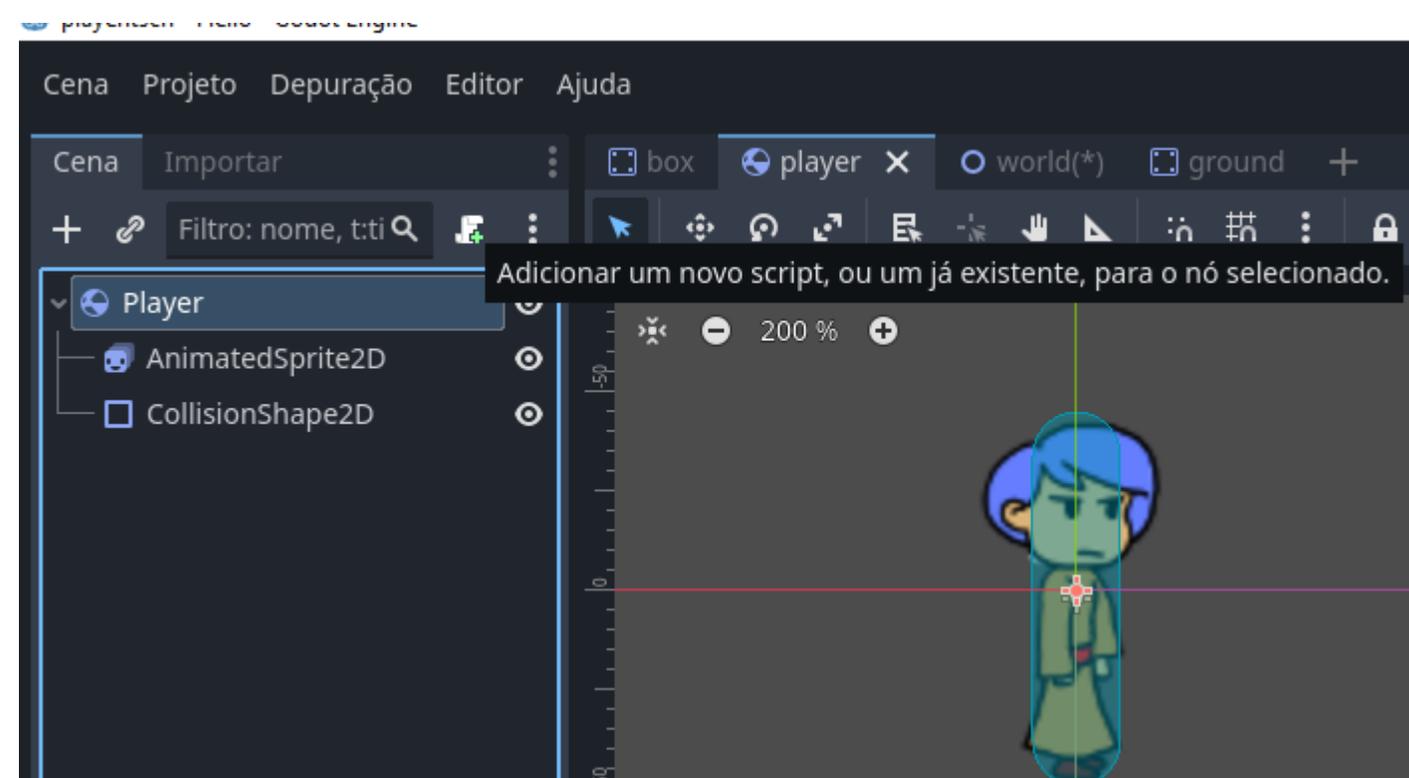


Agora o boneco para no chão

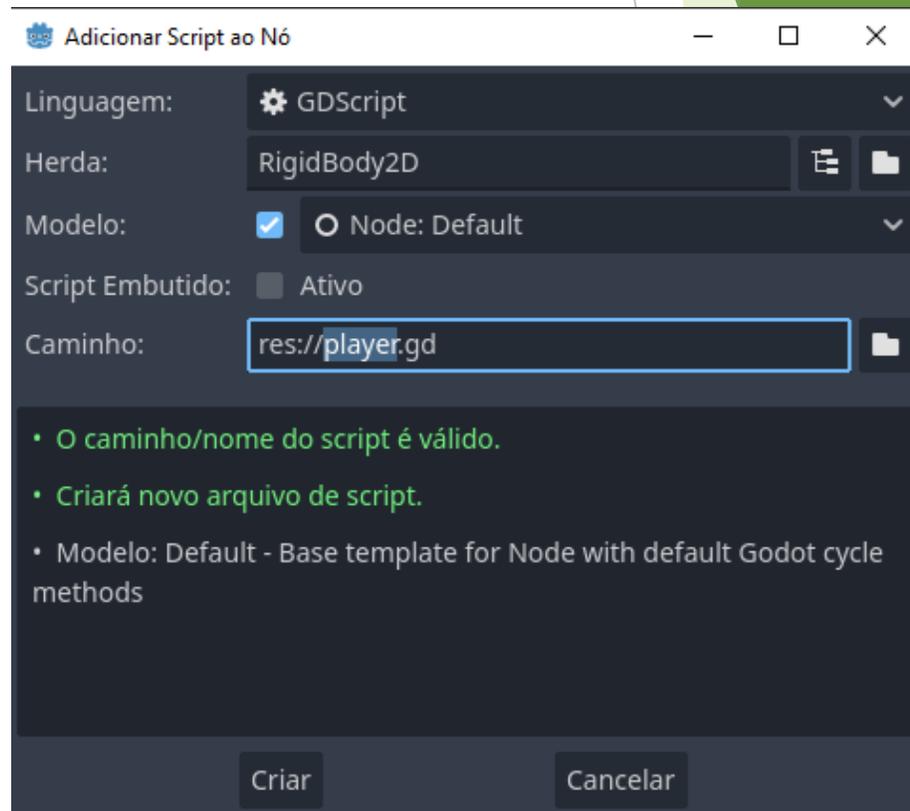




Mude o Ground para primeiro (para não desaparecer o pé do personagem)



Selecione a cena Player
E depois o objeto Palyer
E vamos adicionar um
Script



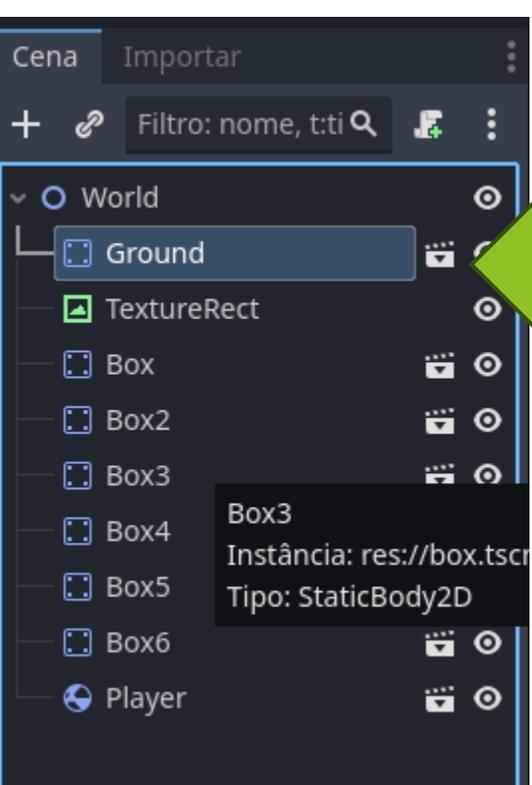
box player(*) world(*) ground +

Arquivo Editar Pesquisar Ir Para Depuração

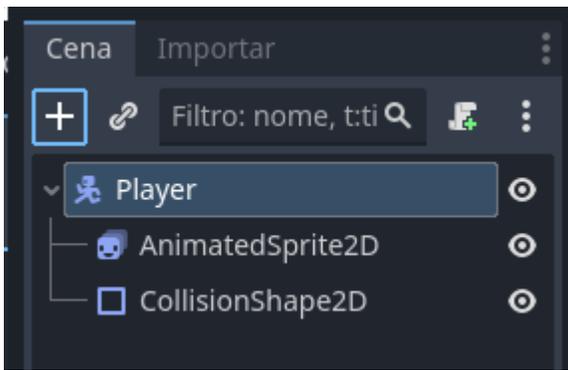
Filtrar Script 🔍

⚙️ player.gd

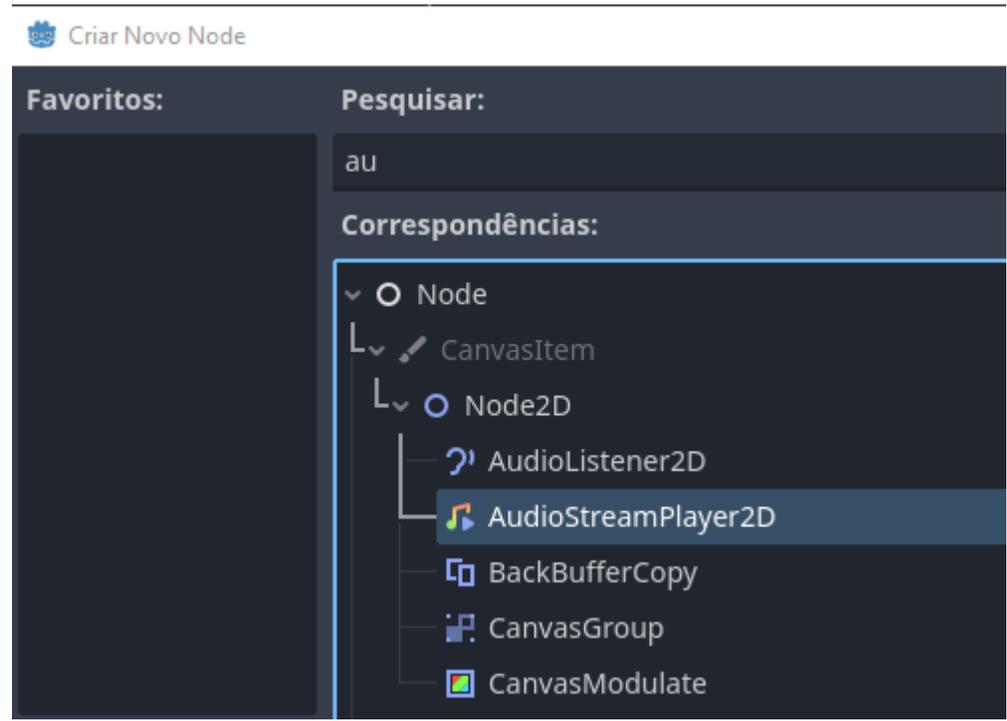
```
1 extends RigidBody2D
2
3
4 # Called when the node enters the scene tree for the first time.
5 func _ready():
6     >| pass # Replace with function logic to run when script starts
7
8
9 # Called every frame. 'delta' is the time between frames.
10 func _process(delta):
11     >| pass
12
```



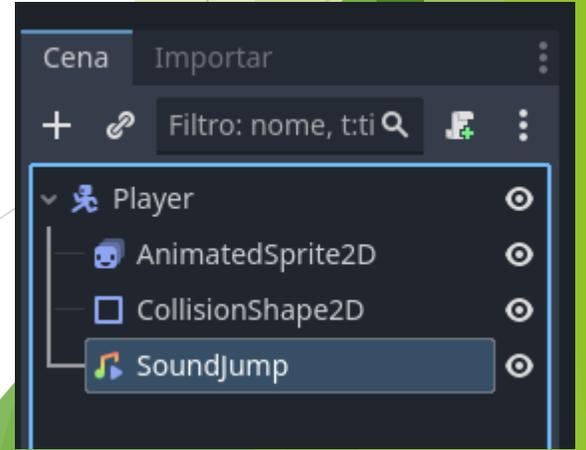
Mude o Ground para primeiro (para não desaparecer o pé do personagem)

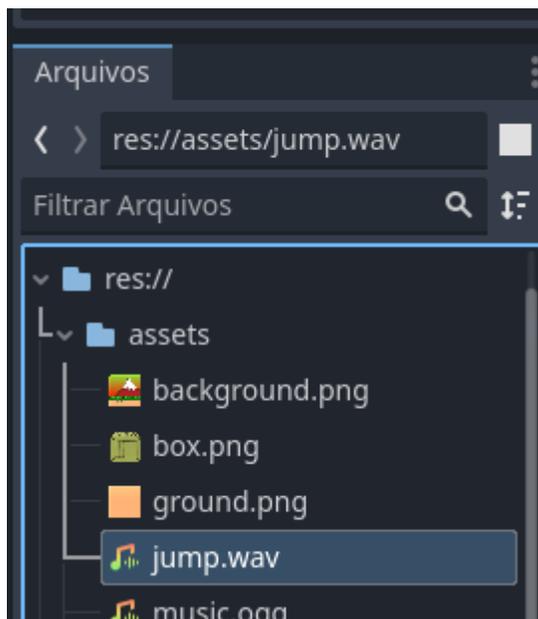


Adicionar o
AudioStream
mPlayer2D

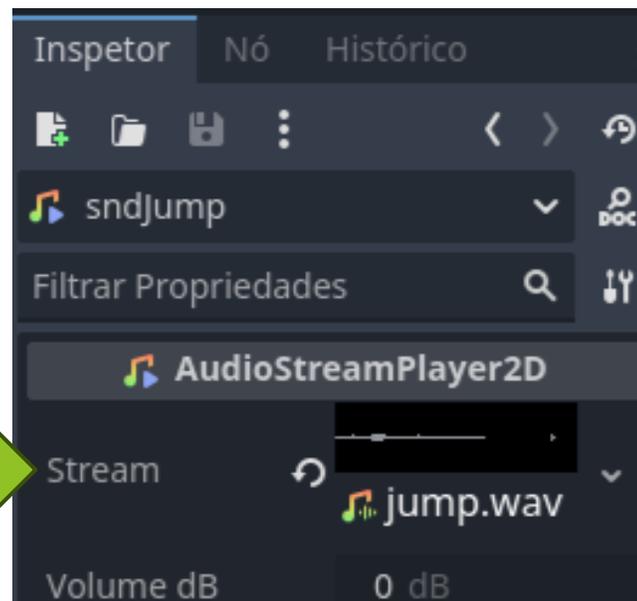


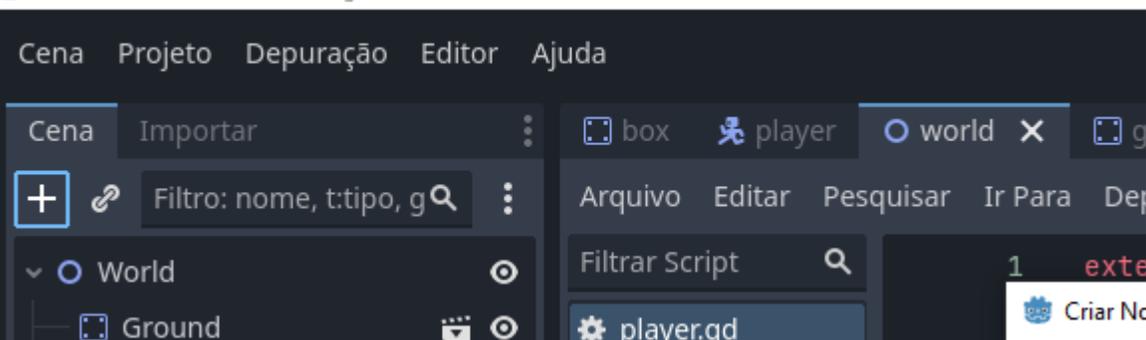
Mude o
nome para
SoundJump



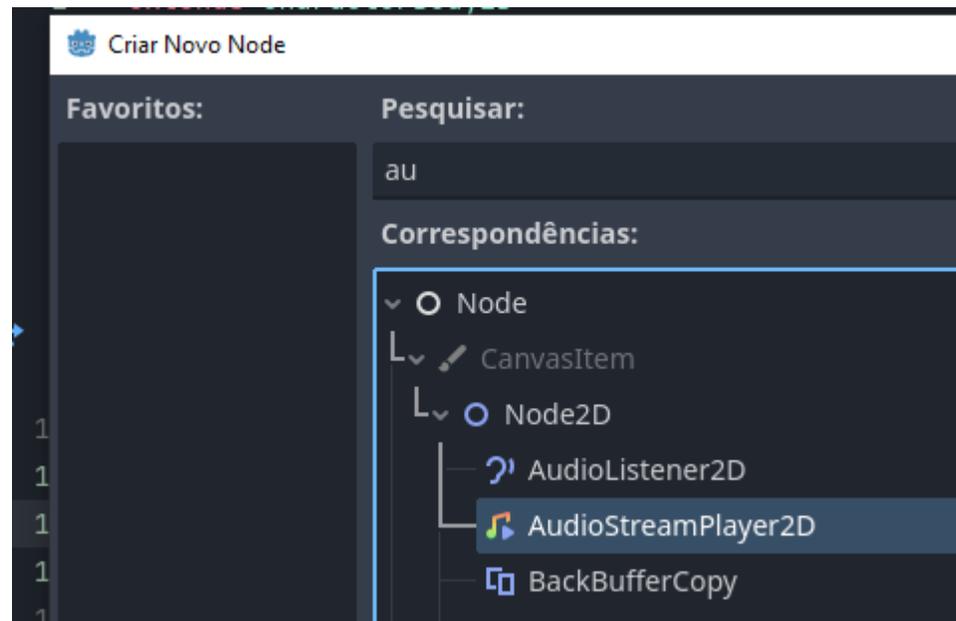


Arraste o som
jump.wav para o
Stream

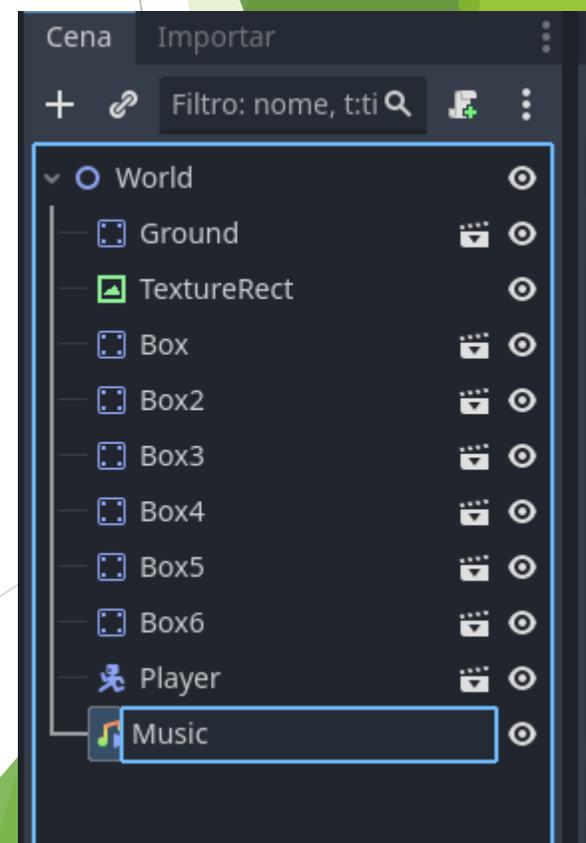


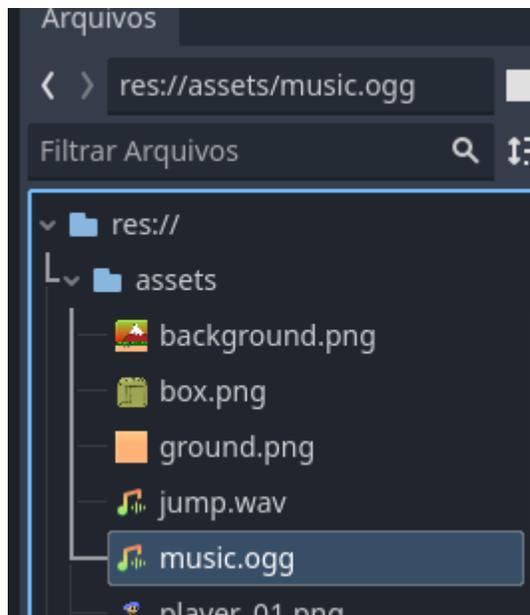


Adicionar o
AudioStreamP
layer2D
Na cena
World

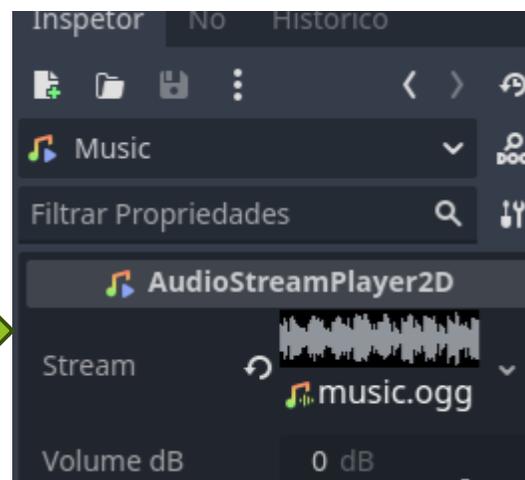


Mude o
nome para
Music

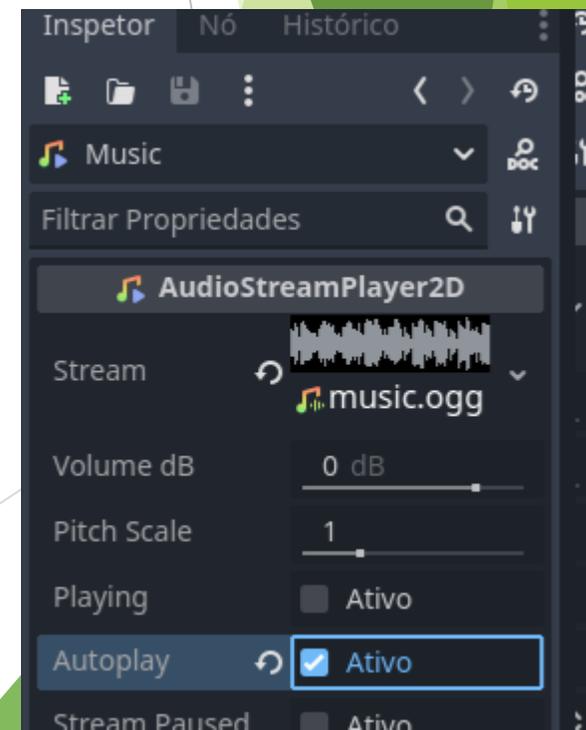


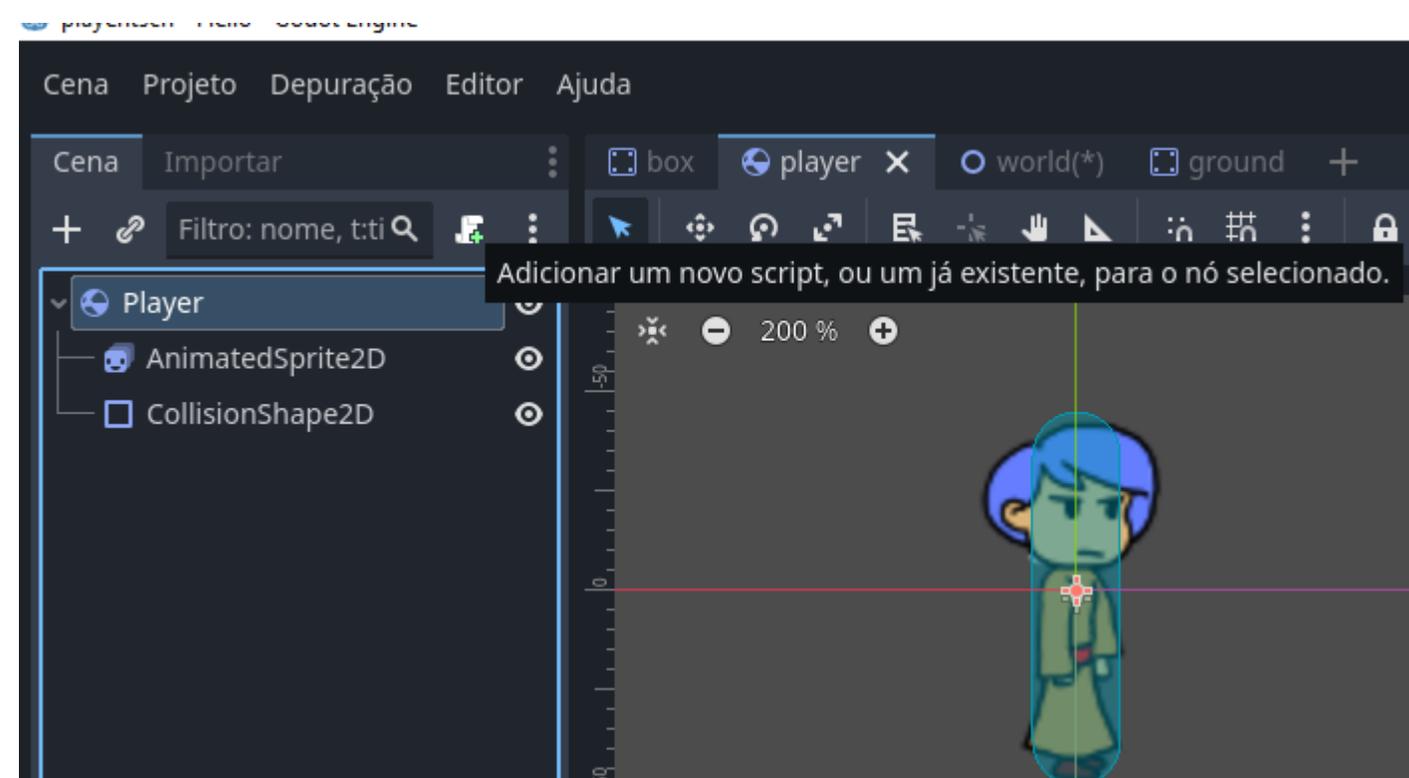


Arraste o som
music.ogg para o
Stream

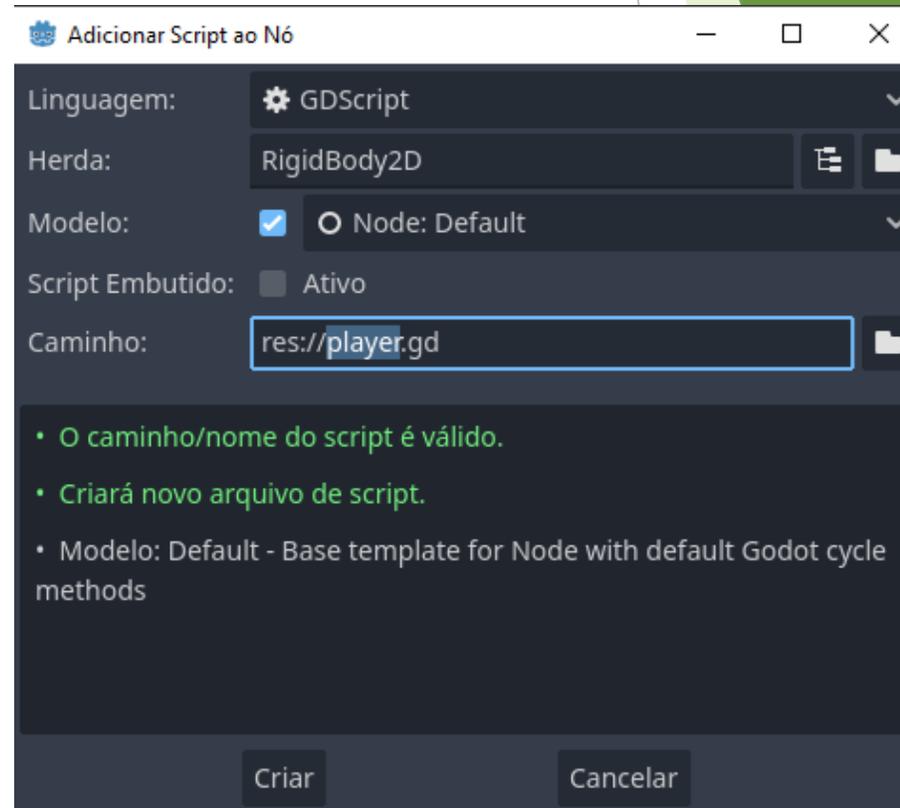


Marque o
autoPlay





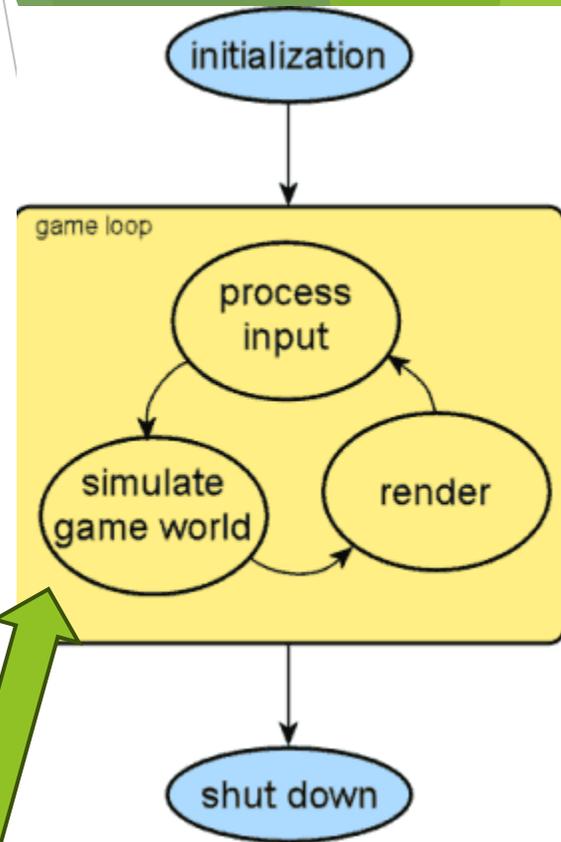
Selecione a cena Player
E depois o objeto Palyer
E vamos adicionar um
Script



```
1 extends Rigidbody2D
2
3
4 # Called when the node enters the scene tree for the first time.
5 func _ready():
6     >| pass # Replace this with initialization logic
7
8
9 # Called every frame. 'delta' is the time between frames.
10 func _process(delta):
11     >| pass
12
```

Executa quando cria o objeto

É o Simulate game world do Game Loop

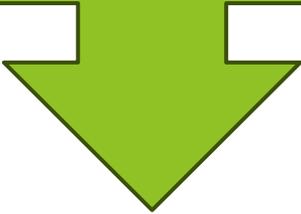


Mais informações sobre game loop:
https://pati.arisa.com.br/wiki/index.php/Jogos_Digitais

Adicione o script

```
extends CharacterBody2D
var speed = 300.0
var jump_speed = -500.0
# Pega a gravidade das configuracoes do projeto
var gravity = ProjectSettings.get_setting("physics/2d/default_gravity")
func _physics_process(delta):
    # Add the gravity.
    velocity.y += gravity * delta
    # Handle Jump.
    if Input.is_action_just_pressed("ui_up") and is_on_floor():
        $SoundJump.play()
        velocity.y = jump_speed
    # Get the input direction.
    var direction = Input.get_axis("ui_left", "ui_right")
    velocity.x = direction * speed
    if direction > 0:
        $AnimatedSprite2D.flip_h = false
        $AnimatedSprite2D.play()
    elif direction < 0:
        $AnimatedSprite2D.flip_h = true
        $AnimatedSprite2D.play()
    else:
        $AnimatedSprite2D.stop()
    move_and_slide()
```

Script para movimentar o personagem
extends deve ter o tipo do objeto que será movido
speed será nossa variável para indicar o quanto o personagem deve se mover
Jump_speed será nossa variável para indicar a altura do salto
gravity definir como vai funcionar a física no mundo
<https://docs.godotengine.org/en/stable/about/introduction.html>



```
1  extends CharacterBody2D
2
3  var speed = 300.0
4  var jump_speed = -500.0
5  # Pega a gravidade das configurações do projeto
6  var gravity = ProjectSettings.get_setting("physics/2d/default_gravity")
7
```

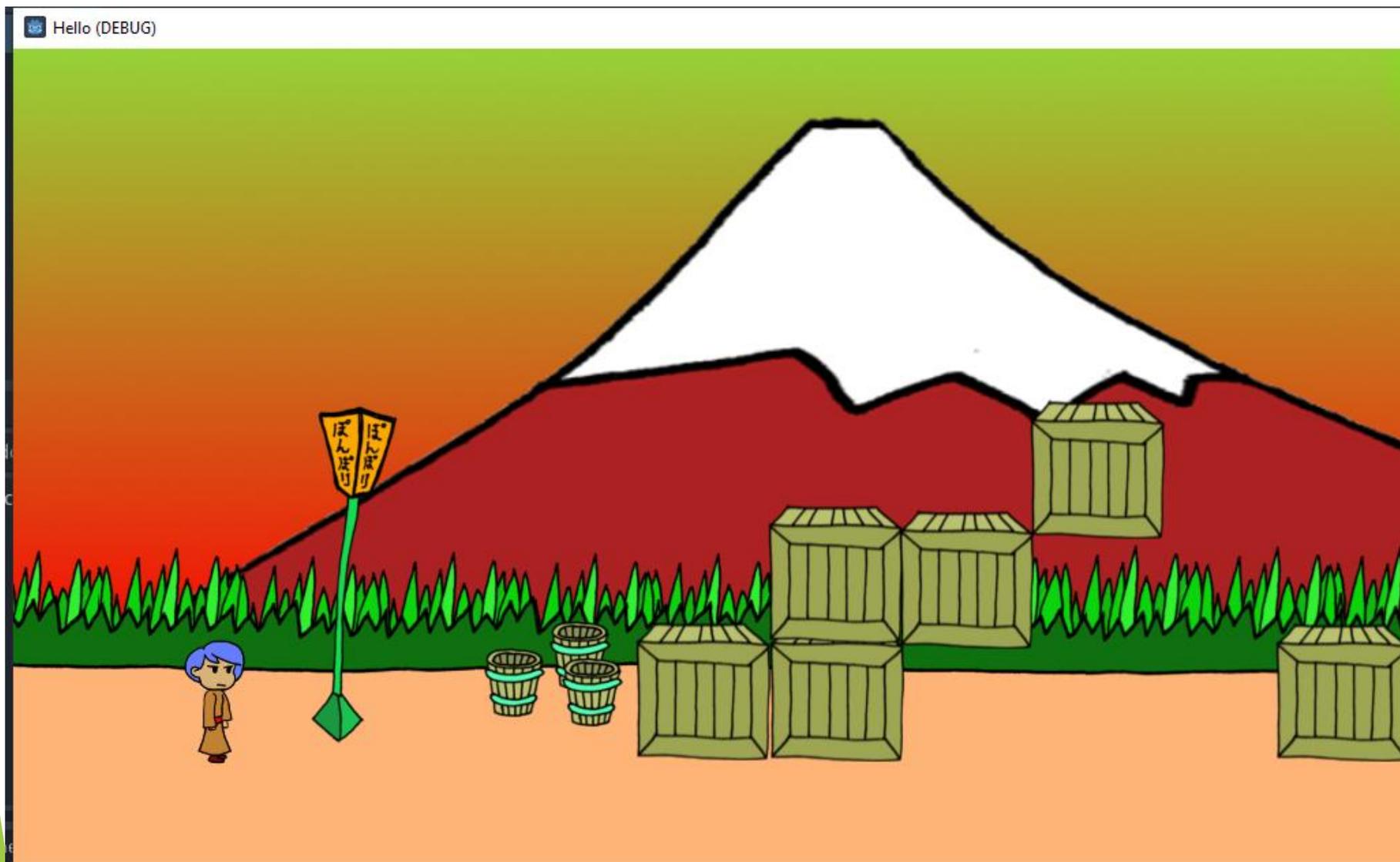
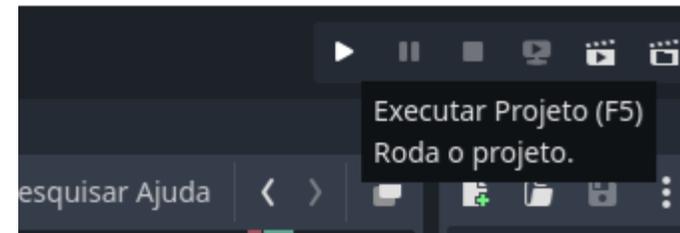
Quando clicar fazer algo
Toda vez que acionar o teclado adiciona a velocidade
Se a tecla foi a up (seta para cima) o personagem pula e aciona o Som de pular

```
func _physics_process(delta):  
    # Add the gravity.  
    velocity.y += gravity * delta  
  
    # Handle Jump.  
    if Input.is_action_just_pressed("ui_up") and is_on_floor():  
        $SoundJump.play()  
        velocity.y = jump_speed
```

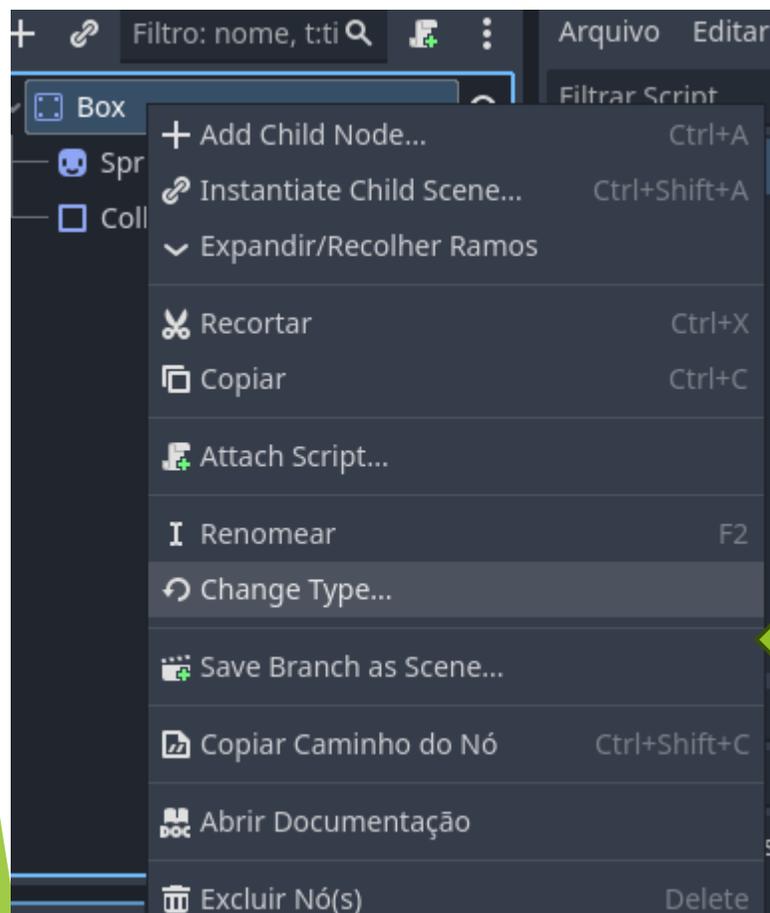
```
# Get the input direction.  
var direction = Input.get_axis("ui_left", "ui_right")  
velocity.x = direction * speed  
if direction > 0:  
> $AnimatedSprite2D.flip_h = false  
> $AnimatedSprite2D.play()  
elif direction < 0:  
> $AnimatedSprite2D.flip_h = true  
> $AnimatedSprite2D.play()  
else:  
> $AnimatedSprite2D.stop()  
  
move_and_slide()
```

Direction indica se tecla pressionada é a seta da esquerda ou da direita.
Adiciona o quanto movimentar em velocity
Se for seta da direita o personagem vai para direita
E o flip false mantém a imagem original
Se for seta da esquerda O boneco vira a imagem e (flip =true)
E o personagem vai para a esquerda
E quando parar de pressionar a tecla ele para.
E por fim executa move_and_slide()

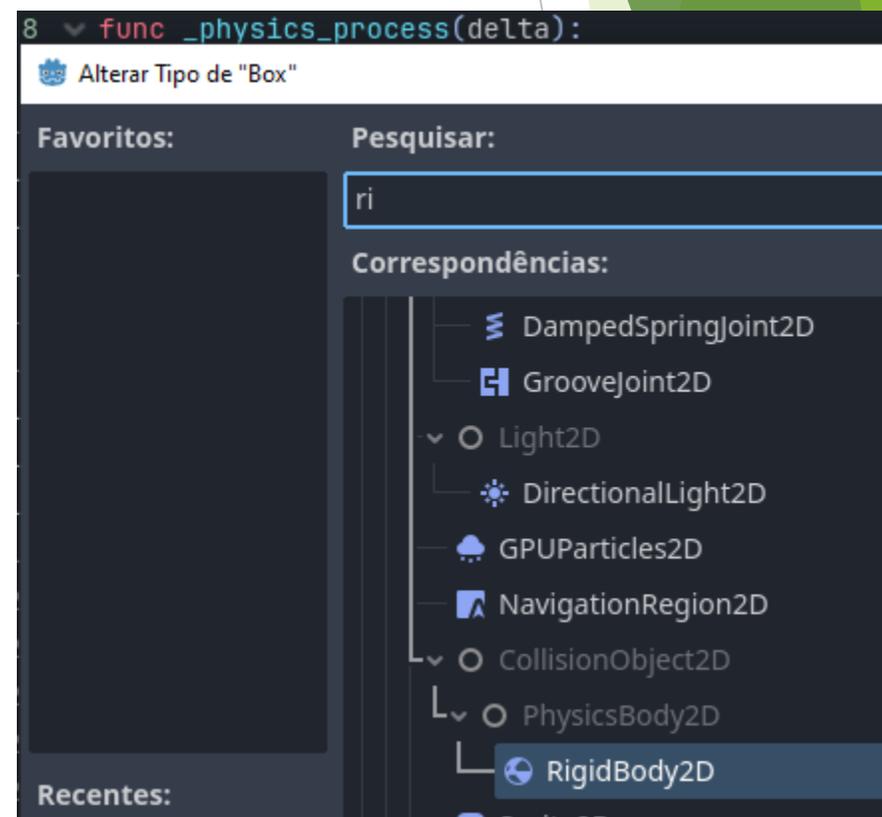
Execute o projeto



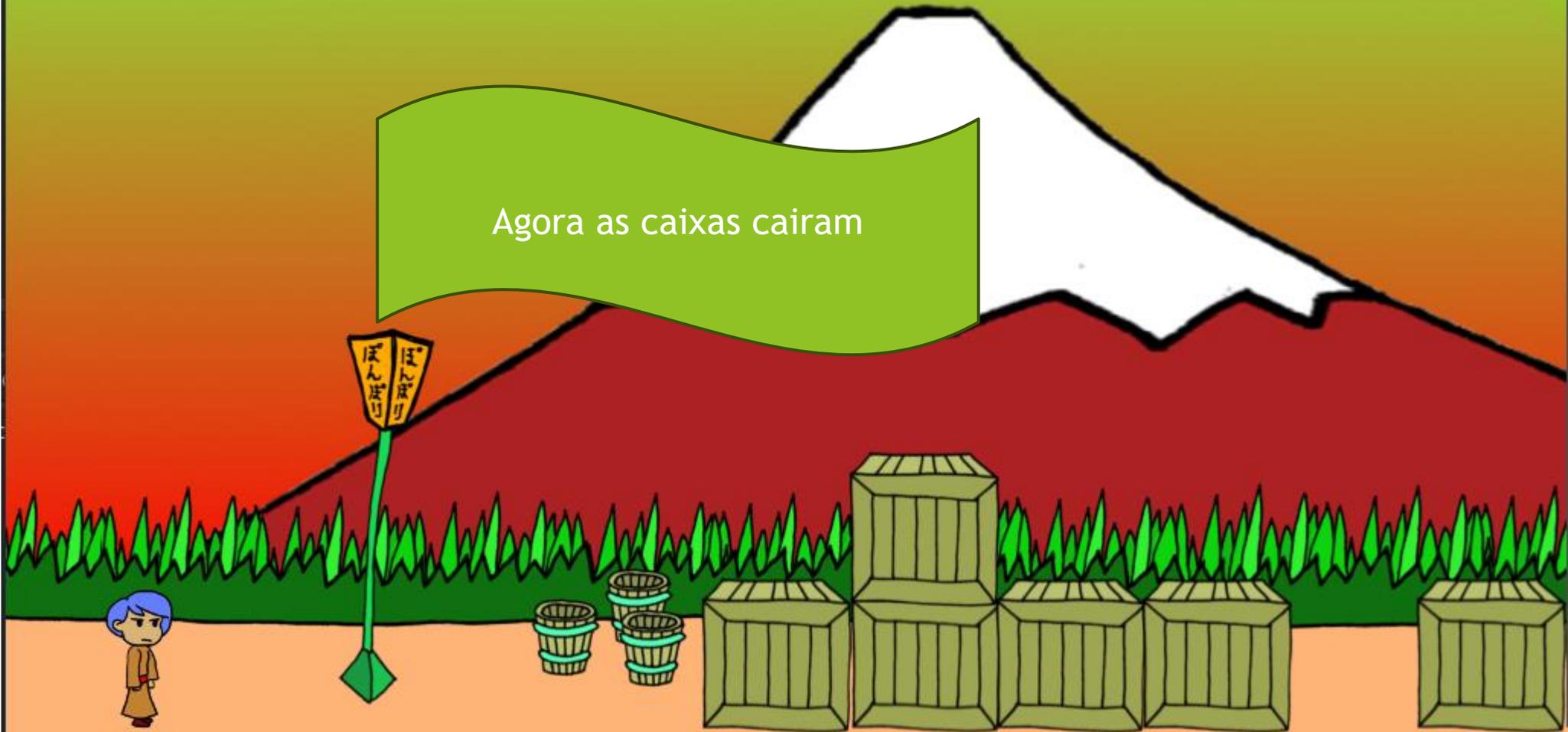
Se quiser Mover as caixas
Temos que mudar o tipo



Change Type
Para
RigidBody2D



Agora as caixas caíram



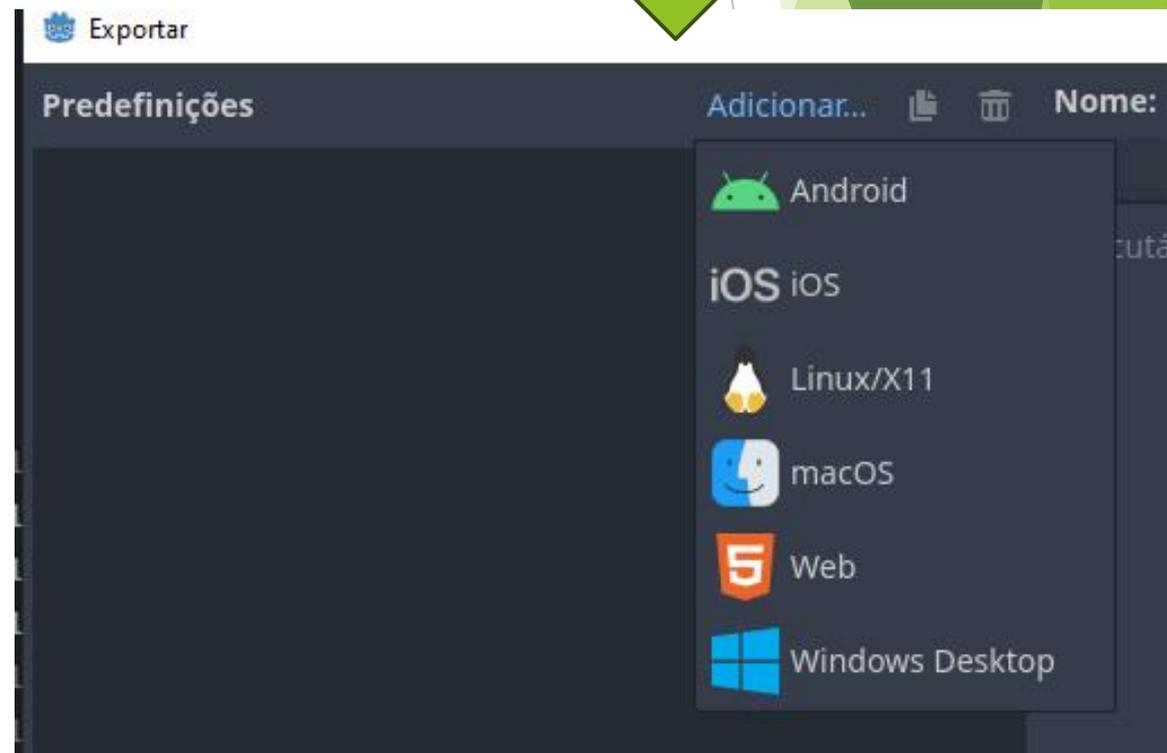
Resumo da estrutura

- Configurações do Projeto:
 - Exibição... Janela...
 - Largura da Viewport: 1024
 - Altura da Viewport: 600
 - Player (CharacterBody2D)
 - AnimatedSprite2D
 - CollisionShape2D
 - SoundJump (AudioStreamPlayer2D)
 - Box e Ground (StaticBody2D)
 - Sprite2D
 - CollisionShape2D
 - World (Node2D)
 - Fundo (TextureRect)
 - Music (AudioStreamPlayer)
 - Linkar
 - Ground
 - Player
 - Box (algumas)
- Pra música ficar em loop, clicar no music.ogg e em cima, ao lado de Cena, clicar em importar. Selecionar Repetir e reimportar.



Vamos Exportar

Adicione a plataforma



BPTC	<input checked="" type="checkbox"/> Ativo
S3TC	<input checked="" type="checkbox"/> Ativo
ETC	<input type="checkbox"/> Ativo
ETC2	<input type="checkbox"/> Ativo

- Nenhum modelo de exportação encontrado no caminho esperado:

- C:/Users/richard/AppData/Roaming/Godot/export_templates/4.2.2.stable/windows_debug_x86_64.exe

- Nenhum modelo de exportação encontrado no caminho esperado:

- C:/Users/richard/AppData/Roaming/Godot/export_templates/4.2.2.stable/windows_release_x86_64.exe

- A ferramenta rcredit deve ser configurada nas Configurações do Editor (Exportar > Windows > rcredit) para alterar o ícone ou os dados de informações do aplicativo.

Os seguintes modelos de exportação para esta plataforma não foram encontrados: [Gerenciar Modelos de Exportação](#)

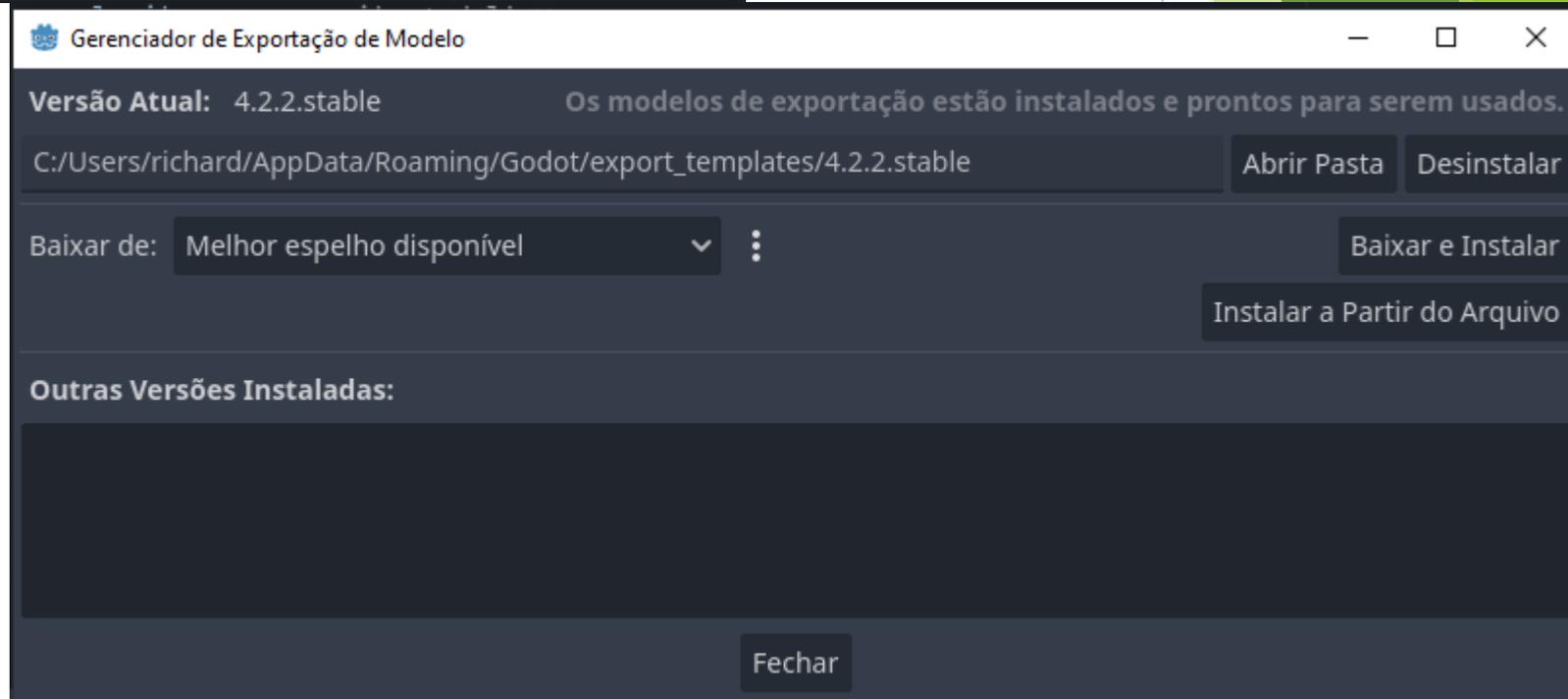
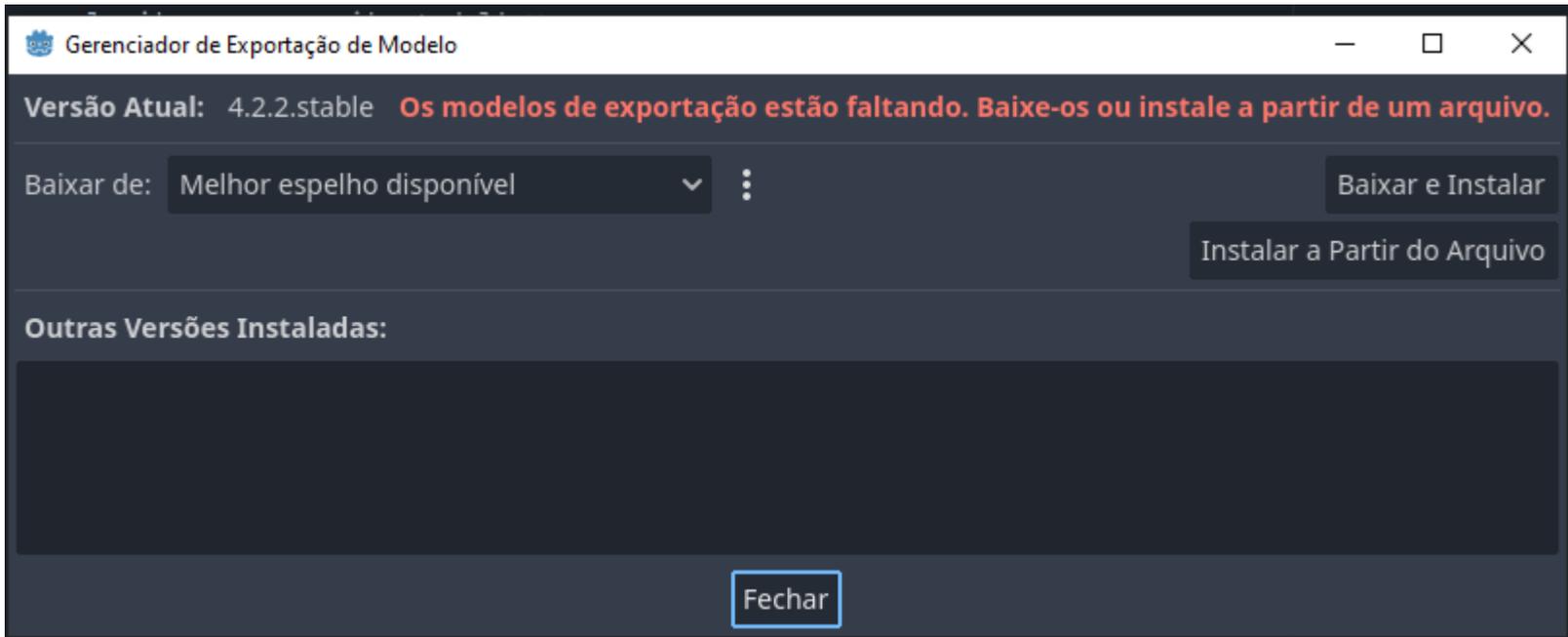
Exportar Tudo...

Exportar Projeto...

Exportar como Zip...

Fechar

Clique para baixar um modelo





Vamos Exportar

Para mudar o ícone

- A ferramenta rcredit deve ser configurada nas Configurações do Editor (Exportar > Windows > rcredit) para alterar o ícone ou os dados de informações do aplicativo.

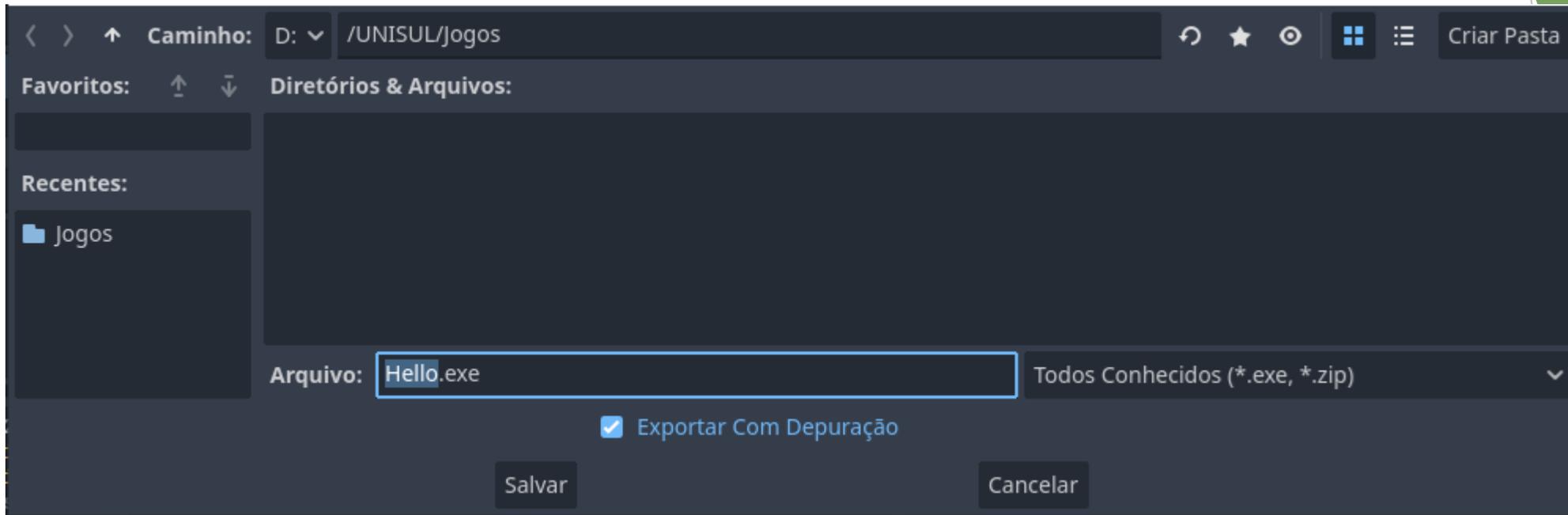
Exportar Tudo...

Exportar Projeto...

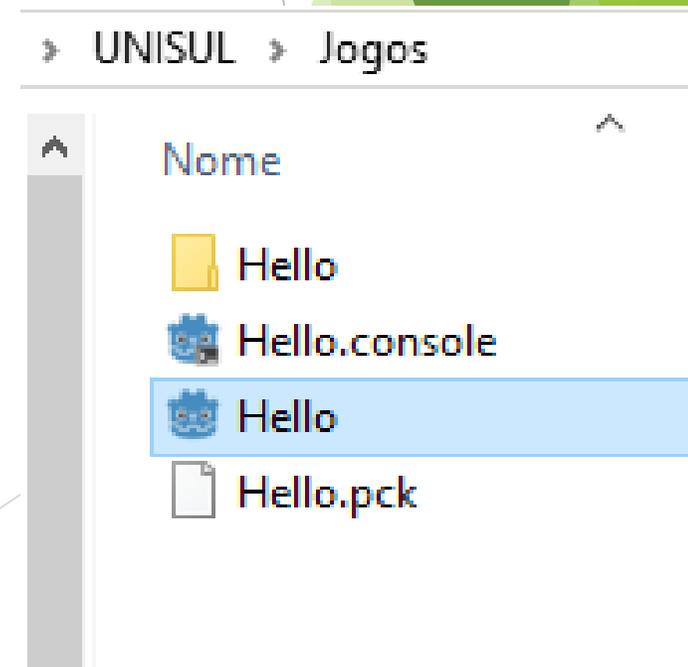
Exportar PCK/Zip...

Fechar

Clique em exportar



Salve
E veja o exe



Bom Estudo !!!

