

# AULA 03

## TIPOS DE JOGOS



# TIPOS DE JOGOS



## Quais tipos de jogos vocês conhecem?

Os videogames são uma das formas de entretenimento mais populares do mundo, abrangendo diversos estilos, mecânicas e objetivos. Eles podem ser classificados de várias maneiras, dependendo de suas características principais, como gênero, plataforma, jogabilidade e modelo de negócio.



# TIPOS DE JOGOS

## Introdução

Os jogos podem ser classificados de diversas maneiras, e uma das principais é com base na sua dimensionalidade: 2D (bidimensional) ou 3D (tridimensional). Essa distinção afeta não apenas a estética, mas também a jogabilidade, as mecânicas e até mesmo os desafios técnicos no desenvolvimento.



# TIPOS DE JOGOS

## O que são jogos 2D?

Jogos 2D são aqueles que possuem gráficos e movimentação limitados a duas dimensões: largura (X) e altura (Y). Os personagens e elementos do cenário não têm profundidade real e se movem em eixos restritos, como lateralmente (esquerda/direita) ou verticalmente (cima/baixo).



# TIPOS DE JOGOS

## O que são jogos 2D?

Características principais:

- ✓ Movimentação restrita a dois eixos (X e Y).
- ✓ Estilo visual pode ser pixel art, vetorial ou animação tradicional.
- ✓ Geralmente mais fáceis e rápidos de desenvolver.
- ✓ Ideal para jogos retrô e plataformas móveis.

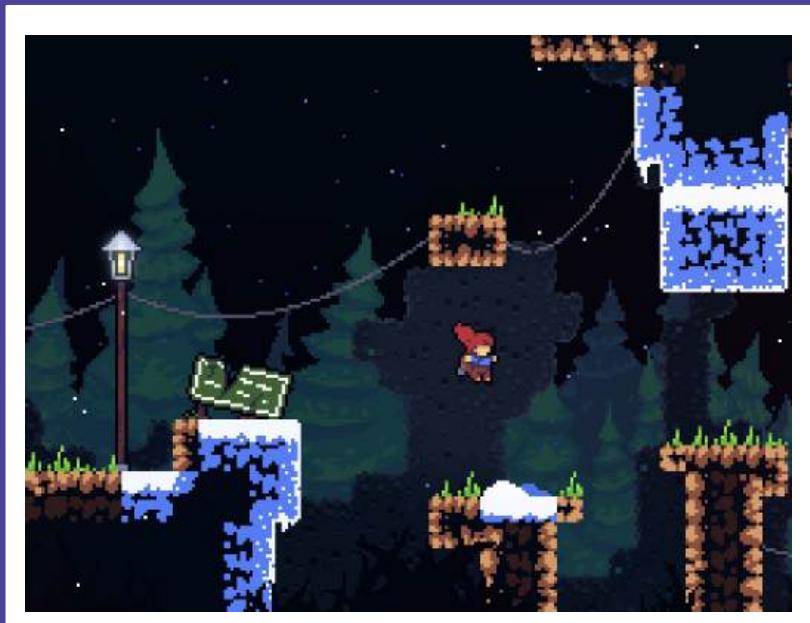


# TIPOS DE JOGOS

## O que são jogos 2D?

Vantagens dos jogos 2D:

- ✓ Desenvolvimento mais simples – Motores como Construct 3, GameMaker e Godot facilitam a criação de jogos 2D.
- ✓ Baixo consumo de recursos – Jogos 2D costumam rodar bem em computadores e dispositivos menos potentes.
- ✓ Fácil aprendizado para iniciantes – Criar jogos 2D exige menos conhecimento técnico do que 3D.



# TIPOS DE JOGOS

## O que são jogos 2D?

Desvantagens dos jogos 2D:

- ✗ Menos imersivos – A falta de profundidade pode limitar a experiência do jogador.
- ✗ Menos realismo – Alguns gêneros, como simulações, são difíceis de representar em 2D.
- ✗ Menos flexibilidade na jogabilidade – Jogos com visão limitada podem restringir possibilidades criativas.

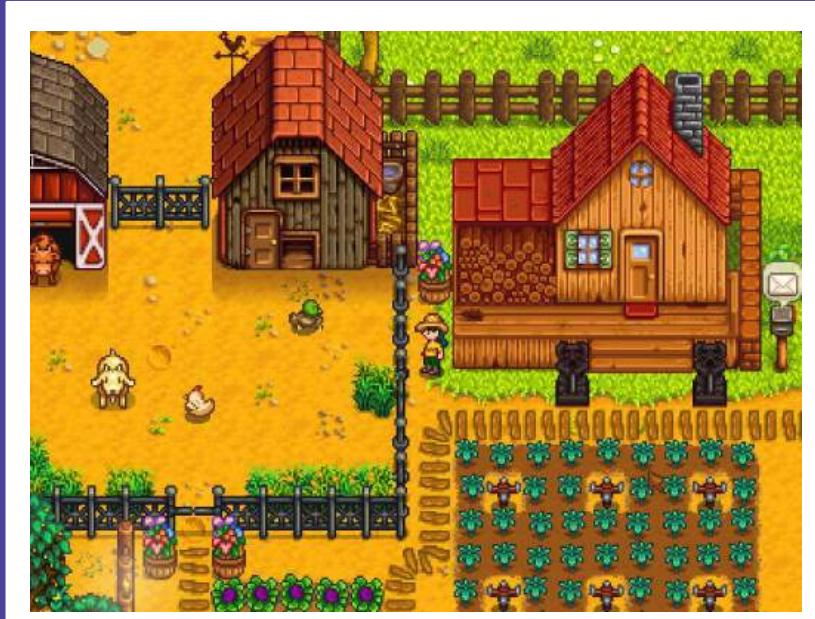


# TIPOS DE JOGOS

## O que são jogos 2D?

Exemplos de jogos 2D famosos:

- 🎮 Super Mario Bros. (Plataforma)
- 🎮 Hollow Knight (Metroidvania)
- 🎮 Celeste (Plataforma)
- 🎮 Undertale (RPG)
- 🎮 Stardew Valley (Simulação)



# TIPOS DE JOGOS

## O que são jogos 3D?

Jogos 3D são aqueles que apresentam gráficos tridimensionais, com profundidade, permitindo a movimentação em três eixos: largura (X), altura (Y) e profundidade (Z). Isso dá maior realismo e liberdade ao jogador.



# TIPOS DE JOGOS

## O que são jogos 3D?

Características principais:

- ✓ Movimentação em todas as direções, incluindo profundidade (Z).
- ✓ Gráficos com modelos tridimensionais e texturas realistas.
- ✓ Maior complexidade na programação e no design de níveis.
- ✓ Requer motores gráficos mais avançados, como Unity e Unreal Engine.



# TIPOS DE JOGOS

## O que são jogos 3D?

Vantagens dos jogos 3D:

- ✓ Maior imersão – O ambiente tridimensional permite experiências mais realistas.
- ✓ Mais liberdade de jogabilidade – Os jogadores podem explorar mundos mais vastos.
- ✓ Gráficos modernos e realistas – A evolução gráfica favorece a criação de visuais impressionantes.



# TIPOS DE JOGOS

## O que são jogos 3D?

Desvantagens dos jogos 3D:

- ✗ Desenvolvimento mais complexo – Requer mais conhecimento técnico e maior tempo de produção.
- ✗ Demanda mais poder de processamento – Jogos 3D precisam de computadores e consoles mais potentes.
- ✗ Dificuldade para iniciantes – Criar jogos 3D exige conhecimento em modelagem, animação e física.

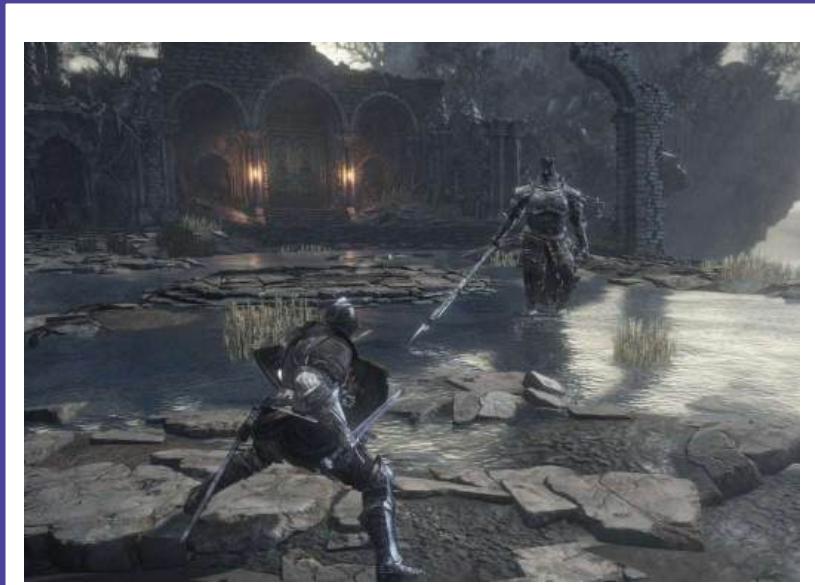


# TIPOS DE JOGOS

## O que são jogos 3D?

Exemplos de jogos 3D famosos:

- 🎮 The Legend of Zelda: Breath of the Wild (Aventura)
- 🎮 Grand Theft Auto V (Mundo Aberto)
- 🎮 The Witcher 3 (RPG)
- 🎮 Minecraft (Sobrevivência)
- 🎮 Dark Souls (Ação)



# TIPOS DE JOGOS



## Comparação Entre Jogos 2D e 3D

Característica	Jogos 2D	Jogos 3D
Gráficos	Simples, estilo pixel art ou vetorial	Modelos tridimensionais com texturas avançadas
Jogabilidade	Movimentação em dois eixos	Movimentação em três eixos
Complexidade	Mais simples de programar	Requer mais conhecimento técnico
Desempenho	Leve, roda em qualquer dispositivo	Exige mais hardware
Imersão	Menor, visão limitada	Maior, sensação de profundidade

# Gêneros de Jogos

Os videogames são classificados em diferentes gêneros de acordo com suas mecânicas, objetivos e estilo de jogabilidade. Cada gênero proporciona uma experiência única, variando desde desafios estratégicos e ação frenética até narrativas envolventes e simulações realistas.



# TIPOS DE JOGOS

## 1 – Ação

Os jogos de ação exigem reflexos rápidos, coordenação motora e tomada de decisões instantâneas. São dinâmicos e envolvem combates, desafios físicos e movimentação ágil.

### Subgêneros:

**Plataforma:** O jogador salta entre plataformas e supera obstáculos.  
(Exemplo: Super Mario Bros.).

**Hack and Slash:** Foco em combate corpo a corpo contra múltiplos inimigos.  
(Exemplo: God of War, Streets of Rage)

**Tiro (FPS e TPS):** Jogos de tiro em primeira (FPS) ou terceira pessoa (TPS).  
(Exemplo: Call of Duty, Fortnite)

**Luta:** Dois jogadores se enfrentam em combates diretos.  
(Exemplo: Mortal Kombat, Street Fighter)



# TIPOS DE JOGOS

## 1 – Ação

Principais Características:

- ✓ **Movimentação Ágil** – O jogador deve se mover rapidamente para desviar de obstáculos, atacar ou escapar de inimigos.
- ✓ **Combate Intenso** – Envolve lutas diretas contra inimigos, com uso de ataques rápidos, combos ou armas.
- ✓ **Reflexos Rápidos** – Exige do jogador reações instantâneas para superar desafios e derrotar adversários.
- ✓ **Progressão Constante** – Superar inimigos ou obstáculos resulta em evolução, novas habilidades e recompensas.
- ✓ **Desafios Imersivos** – O jogo oferece um fluxo contínuo de desafios que mantêm o jogador envolvido e focado.



# TIPOS DE JOGOS

## 2 – Aventura

Jogos de aventura focam na exploração e resolução de desafios. Geralmente, possuem histórias ricas e demandam lógica e investigação.

### Subgêneros:

**Point-and-Click** – O jogador clica em objetos e interage com o ambiente para resolver enigmas. Exemplo: Monkey Island, Grim Fandango, The Walking Dead.

**Metroidvania** – Mapas interconectados onde novas áreas são desbloqueadas conforme o jogador ganha habilidades. Exemplo: Hollow Knight, Castlevania: Symphony of the Night, Metroid Dread.

**Survival** – O jogador precisa coletar recursos e sobreviver em um ambiente hostil. Exemplo: Minecraft, ARK: Survival Evolved, Don't Starve.



# TIPOS DE JOGOS

## 2 – Aventura

Principais Características:

- ✓ Exploração – O jogador percorre cenários ricos em detalhes, descobrindo novos locais, itens e segredos.
- ✓ Narrativa Envolvente – A história é central, e o jogador segue um enredo com objetivos claros e escolhas que impactam a trama.
- ✓ Resolução de Quebra-Cabeças – O jogo desafia o jogador a resolver enigmas e encontrar soluções criativas para avançar.
- ✓ Interação com Personagens – O jogador interage com NPCs, que oferecem missões, dicas ou informações importantes.
- ✓ Missões Secundárias – Além da história principal, há atividades extras que enriquecem a experiência e ampliam o mundo do jogo.



# TIPOS DE JOGOS

## 3 – RPG (Role-Playing Game) ✖

O jogador assume o papel de um personagem e evolui ao longo do jogo.

### Subgêneros:

**Action RPG** – Combates em tempo real combinados com progressão de personagem. Exemplo: The Witcher 3, Dark Souls, Elden Ring.

**Turn-Based RPG** – Combates por turnos onde o jogador escolhe ações estrategicamente. Exemplo: Final Fantasy VII, Pokémon, Persona 5.

**MMORPG** – RPGs online massivos onde milhares de jogadores interagem. Exemplo: World of Warcraft, Guild Wars 2, Final Fantasy XIV.



# TIPOS DE JOGOS



## 3 – RPG (Role-Playing Game) ✖

Principais Características:

- ✓ Assunção de Papéis – O jogador controla um ou mais personagens com habilidades, atributos e histórias próprias, assumindo seus papéis no mundo do jogo.
- ✓ Desenvolvimento de Personagens – Os personagens evoluem ao longo do jogo, ganhando novos poderes, habilidades e equipamentos conforme progridem.
- ✓ Escolhas e Consequências – As decisões do jogador afetam o enredo, o destino dos personagens e o mundo ao redor, criando uma experiência única.

# TIPOS DE JOGOS

## 3 – RPG (Role-Playing Game) ✖

Principais Características:

- ✓ **Mundo Imersivo** – Os jogos de RPG geralmente têm um mundo vasto e detalhado, com muitas áreas para explorar e interagir.
- ✓ **Missões Diversificadas** – Além da história principal, o jogador pode realizar missões secundárias, explorar dungeons, combater inimigos e completar tarefas para outros personagens.
- ✓ **Sistema de Combate** – Pode ser baseado em turnos (RPG clássico) ou em tempo real (RPG de ação), com foco em estratégias de combate.



# TIPOS DE JOGOS

## 4 – Estratégia

Jogos que exigem planejamento e pensamento tático.

### Subgêneros:

**RTS (Real-Time Strategy)** – Estratégia em tempo real, onde o jogador gerencia recursos e unidades. Exemplo: Age of Empires, StarCraft, Warcraft III.

**Turn-Based Strategy** – Estratégia baseada em turnos. Exemplo: XCOM, Civilization VI, Fire Emblem.

**Tower Defense** – Proteção de uma base contra ondas de inimigos.

Exemplo: Plants vs. Zombies, Bloons TD.



# TIPOS DE JOGOS

## 4 – Estratégia

Principais Características:

- ✓ **Planejamento e Tática** – O jogador deve desenvolver estratégias detalhadas para vencer, considerando recursos, unidades e adversários.
- ✓ **Tomada de Decisão** – As escolhas do jogador têm grande impacto no resultado, seja no campo de batalha ou no desenvolvimento do jogo.
- ✓ **Gestão de Recursos** – É necessário gerenciar recursos como dinheiro, tropas e tempo para alcançar os objetivos, equilibrando ofensiva e defesa.



# TIPOS DE JOGOS

## 4 – Estratégia

Principais Características:

- ✓ Desafios de Longo Prazo – O foco é o pensamento estratégico de longo prazo, com planos que exigem adaptação e reflexão durante todo o jogo.
- ✓ Interação com a IA ou Jogadores – Pode ser jogado contra a inteligência artificial (IA) ou contra outros jogadores, o que aumenta a complexidade e o desafio.
- ✓ Mapas e Cenários Dinâmicos – Os mapas e o cenário do jogo oferecem variabilidade, exigindo que o jogador se ajuste constantemente às novas condições.



# TIPOS DE JOGOS

## 5 – Simulação 🎮

Simulam atividades da vida real ou experiências específicas.

### Subgêneros:

**Simulação de Vida:** Controle de personagens em rotinas diárias. (Exemplo: The Sims)

**Simulação de Construção e Gestão:** Criar e gerenciar cidades, empresas ou parques. (Exemplo: SimCity, RollerCoaster Tycoon)

**Simulação de Veículos:** Condução realista de veículos. (Exemplo: Flight Simulator, Euro Truck Simulator)



# TIPOS DE JOGOS

## 5 – Simulação 🚗

Principais Características:

- ✓ Recriação Realista – O jogo tenta simular atividades, ambientes ou sistemas reais de forma precisa, como dirigir, construir ou administrar.
- ✓ Controle de Sistemas Complexos – O jogador interage com sistemas detalhados, como gerenciar uma fazenda, pilotar aviões ou construir cidades.
- ✓ Liberdade de Ação – Frequentemente, o jogador tem liberdade para tomar decisões e ver como elas afetam o ambiente simulado. única a cada jogada.



# TIPOS DE JOGOS



## 5 – Simulação 🚗

Principais Características:

- ✓ **Objetivos Realistas** – Ao invés de desafios fictícios, os jogos de simulação muitas vezes têm objetivos baseados em tarefas reais ou responsabilidades cotidianas.
- ✓ **Aprendizado Prático** – Pode ajudar na compreensão de processos ou habilidades do mundo real, como condução, construção ou administração de empresas.
- ✓ **Ambiente Dinâmico** – O cenário e os recursos disponíveis mudam conforme o jogador interage, criando uma experiência única a cada jogada.

# TIPOS DE JOGOS

## 6 – Esportes e Corrida

Simulam esportes reais ou fictícios, podendo ter um tom mais realista ou arcade.

### Subgêneros:

**Esportes:** Jogos baseados em modalidades esportivas.

(Exemplo: FIFA, NBA 2K)

**Corrida Arcade:** Foco na diversão e jogabilidade casual.

(Exemplo: Mario Kart, Need for Speed)

**Corrida Realista:** Física e direção detalhada.

(Exemplo: Gran Turismo, Forza Horizon)



# TIPOS DE JOGOS

## 6 – Esportes e Corrida

Principais Características:

- ✓ **Simulação de Esportes Reais** – Os jogos de esportes e corrida recriam de maneira fiel esportes reais, como futebol, basquete, corrida de carros, futebol americano, entre outros.
- ✓ **Competição e Desempenho** – O objetivo principal é competir contra outros jogadores ou a IA, vencendo por meio de habilidade, estratégia e desempenho.
- ✓ **Mecânicas Realistas** – Físicas realistas, controle preciso e regras do esporte verdadeiro são recriados para aumentar a imersão.



# TIPOS DE JOGOS



## 6 – Esportes e Corrida

Principais Características:

- ✓ **Modos de Jogo Variados** – Além das competições padrão, esses jogos frequentemente oferecem modos de torneios, carreiras e desafios.
- ✓ **Multiplayer** – A maioria dos jogos de esportes e corrida permite interação com outros jogadores, seja localmente ou online, proporcionando uma experiência social.
- ✓ **Personalização de Equipamentos e Personagens** – Muitos jogos permitem a customização de personagens, equipes, veículos ou equipamentos, adicionando mais profundidade ao jogo.

# TIPOS DE JOGOS

## 7 – Puzzle e Quebra-Cabeça 🎮

Jogos que desafiam a lógica e o raciocínio.

Match-3 – O jogador combina peças semelhantes.

Exemplo: Candy Crush, Bejeweled.

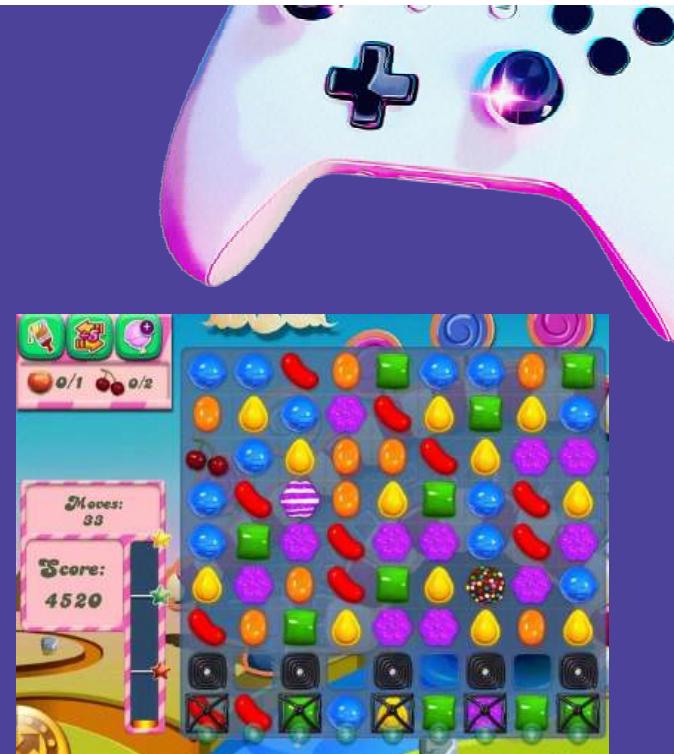
Jogos de Lógica – Resolver enigmas e desafios matemáticos.

Exemplo: Sudoku, Tetris, Portal 2

## 8 - Horror e Sobrevivência 🎃

Criam uma atmosfera de suspense e terror, desafiando o jogador a sobreviver em condições extremas.

Exemplos: Resident Evil, Silent Hill, The Forest.



# TIPOS DE JOGOS

## 7 – Puzzle e Quebra-Cabeça 🎮

Principais Características:

- ✓ Desafios Cognitivos – O foco é resolver enigmas, quebra-cabeças ou problemas lógicos para avançar no jogo.
- ✓ Pensamento Crítico – O jogador precisa usar raciocínio lógico e estratégias para superar obstáculos e encontrar soluções criativas.
- ✓ Objetivos de Curto Prazo – A maioria das missões consiste em resolver desafios específicos, seja para completar uma fase ou desbloquear um novo nível.



# TIPOS DE JOGOS



## 7 – Puzzle e Quebra-Cabeça 🎮

Principais Características:

- ✓ **Progressão Gradual** – Os quebra-cabeças geralmente se tornam mais difíceis à medida que o jogo avança, exigindo maior habilidade e raciocínio do jogador.
- ✓ **Simples Mecânicas** – Muitas vezes, os jogos de puzzle têm mecânicas simples, mas a complexidade surge nas soluções criativas necessárias para resolver os enigmas.
- ✓ **Estímulo Mental** – Proporcionam um exercício mental, ajudando o jogador a melhorar habilidades como resolução de problemas, concentração e memória.

# TIPOS DE JOGOS



## 8 - Horror e Sobrevivência 😱

Principais Características:

- ✓ Ambiente Tenso e Imersivo – Criam uma atmosfera de medo e suspense, com cenários sombrios, sons aterrorizantes e clima de constante perigo.
- ✓ Elementos de Sobrevivência – O jogador deve gerenciar recursos limitados, como comida, armas e munição, enquanto luta pela sobrevivência em um ambiente hostil.
- ✓ Enfrentando Inimigos Horríveis – Os jogadores enfrentam criaturas monstruosas, zumbis ou outras ameaças sobrenaturais, geralmente com mecânicas de combate e evasão.

# TIPOS DE JOGOS



## 8 - Horror e Sobrevivência 😱

Principais Características:

- ✓ **História Envolvente** – A narrativa muitas vezes explora mistérios, segredos sombrios e situações de vida ou morte, aumentando a tensão emocional.
- ✓ **Gestão de Recursos e Estresse** – O gerenciamento de recursos é crucial, com o jogador tendo que equilibrar o uso de itens para garantir sua sobrevivência a longo prazo.
- ✓ **Jogabilidade Desafiadora** – A dificuldade geralmente é alta, com os jogadores enfrentando desafios que exigem raciocínio, habilidade e coragem para superar.
- ✓ **Medo e Ansiedade** – Elementos como perseguições, sustos repentinos e a constante sensação de estar sendo caçado são usados para gerar medo psicológico.

# TIPOS DE JOGOS

## 9 – Jogos de Mundo Aberto (Open World) 🌎

São jogos que oferecem liberdade para explorar vastos ambientes sem restrições lineares, permitindo que o jogador siga sua própria jornada.

Exemplos: GTA V, Red Dead Redemption 2, Minecraft.



# TIPOS DE JOGOS



## 9 – Jogos de Mundo Aberto (Open World) 🌎

Principais Características:

- ✓ **Liberdade de Exploração** – O jogador pode viajar e explorar o vasto mundo do jogo sem restrições lineares, decidindo para onde ir e o que fazer.
- ✓ **Missões Não Lineares** – A história e os objetivos podem ser completados em diferentes ordens, permitindo que o jogador escolha seu caminho.
- ✓ **Ambiente Dinâmico** – Elementos como clima, ciclo dia/noite, e NPCs interativos tornam o mundo mais imersivo e reativo às ações do jogador. e personagens que reagem às escolhas do jogador.

# TIPOS DE JOGOS



## 9 – Jogos de Mundo Aberto (Open World) 🌎

Principais Características:

- ✓ Atividades Secundárias – Além da missão principal, há uma variedade de atividades extras, como caça, desafios, comércio e mini jogos.
- ✓ Interação Profunda com o Mundo – O jogador pode interagir com diversos elementos do ambiente, como veículos, construções, e personagens, afetando diretamente o mundo do jogo.
- ✓ Narrativas Complexas e Ricas – A história geralmente é complexa e imersiva, com múltiplos arcos e personagens que reagem às escolhas do jogador.

# EXERCICIO 3 – GAME 1 CONTRA 100



## Tutorial:

Jogo de ação 2D onde o jogador deve derrotar 100 inimigos antes que sua barra de vida acabe. Enfrente ondas de inimigos, teste suas habilidades e sobreviva até eliminar todos os 100 adversários.

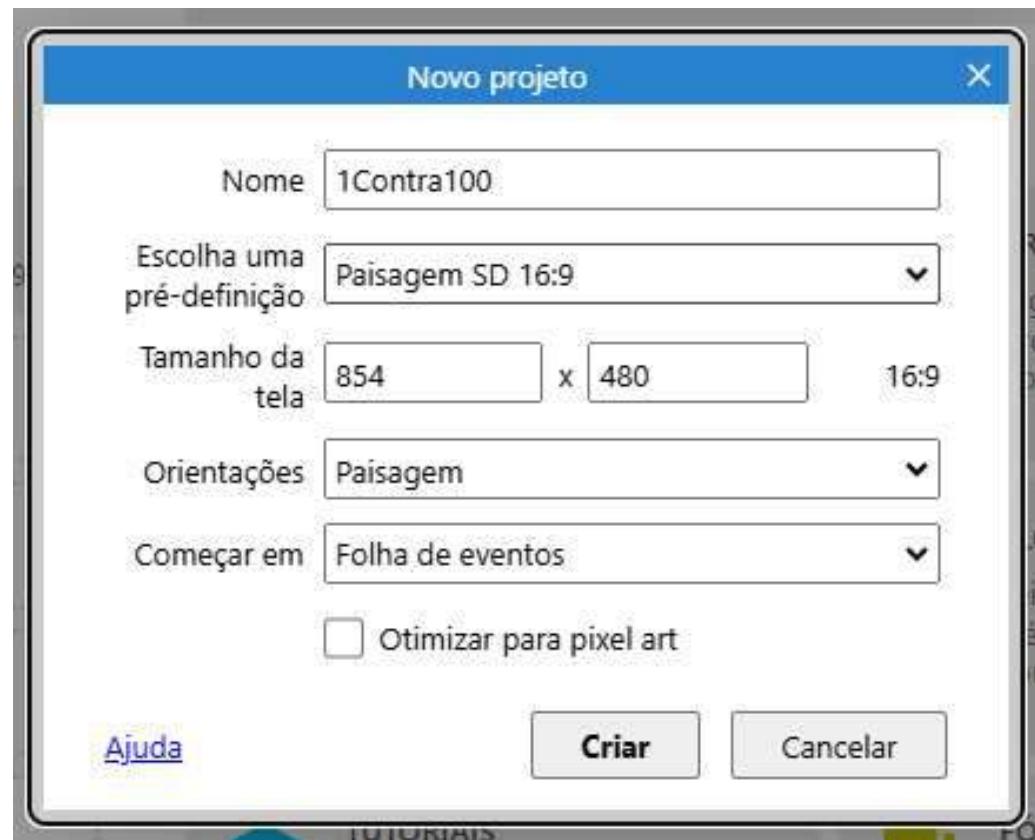
1. Cenário
2. Player
3. Inimigos
4. Mecânica
5. HUD
6. Reinício



# EXERCICIO 3 - GAME 1 CONTRA 100



**1 – Novo projeto e  
deixe do jeito do  
modelo e em  
seguida criar!**



# EXERCICIO 3 - GAME 1 CONTRA 100



2 – Agora vamos criar 2 camadas. Chamadas: **jogo** e **hud**.

3 – Marque a camada **jogo** e vamos adicionar os elementos do jogo. Em seguida Clique em **Layout1**.

4 – Altere o Tamanho do Layout1 para: **4000x4000**.



# EXERCICIO 3 - GAME 1 CONTRA 100



Criamos um mapa Grande para da uma impressão de quase infinito.

Esse quadrado pontilhado é a visão da nossa câmera! Cada vez que a gente se move só podemos ver esse pedaço, que por sua vez se move junto com o player.



5 – Agora vamos baixar o material da aula e extraír os arquivos. Link:  
[https://github.com/ProfaMarcia/Material\\_1contra100/archive/refs/heads/main.zip](https://github.com/ProfaMarcia/Material_1contra100/archive/refs/heads/main.zip)

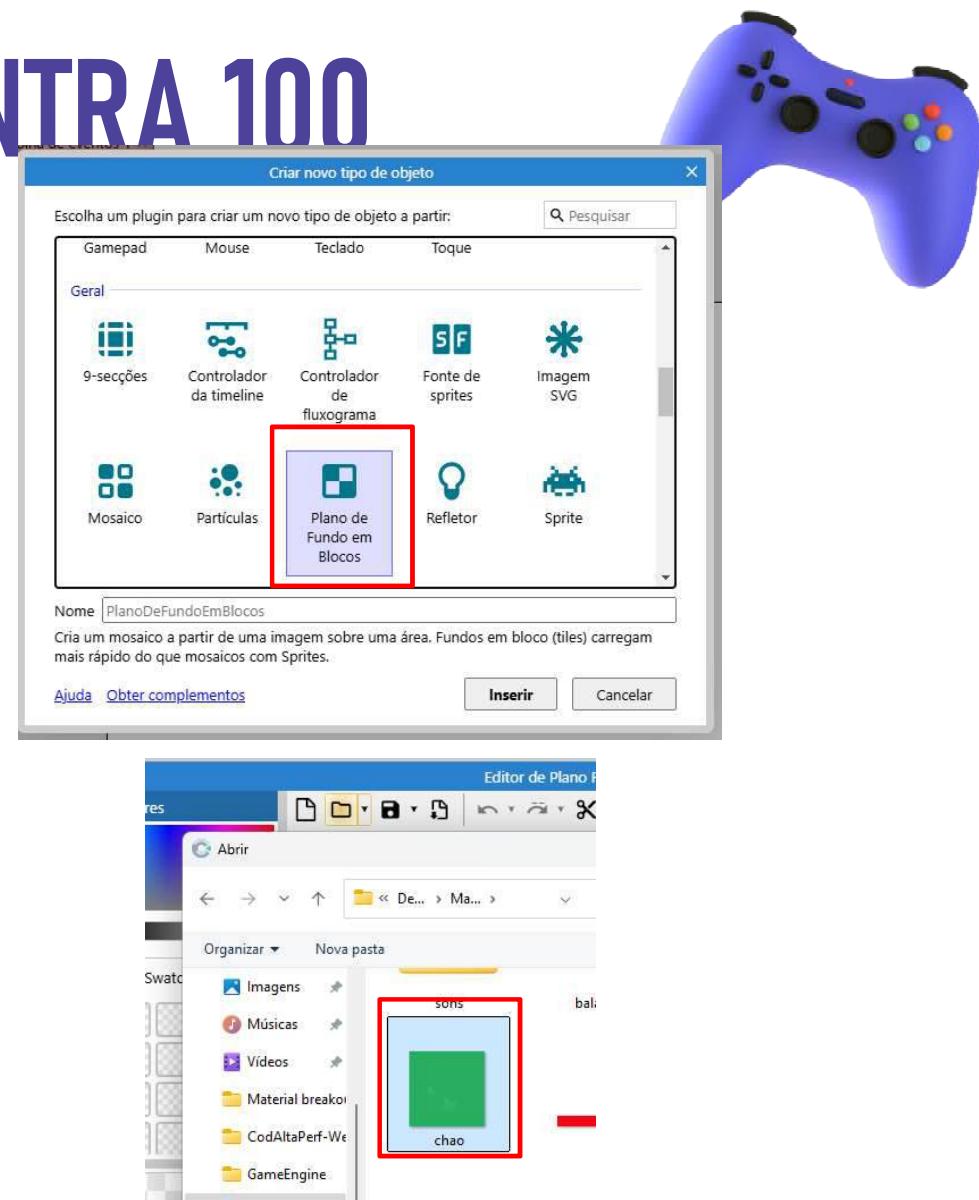


# EXERCICIO 3 - GAME 1 CONTRA 100

6 – Agora vamos criar o chão e os outros objetos para o nosso jogo. Clique 2x com botão esquerdo e em criar objeto, clique em **Plano de Fundo em Blocos**.

Usamos o Plano de Fundo em Blocos para que a imagem não fique desfocada ou esticada, pois fica com uma aparência feia.

7 – Vá em **Carregar imagem do arquivo > Escolha: chao**.

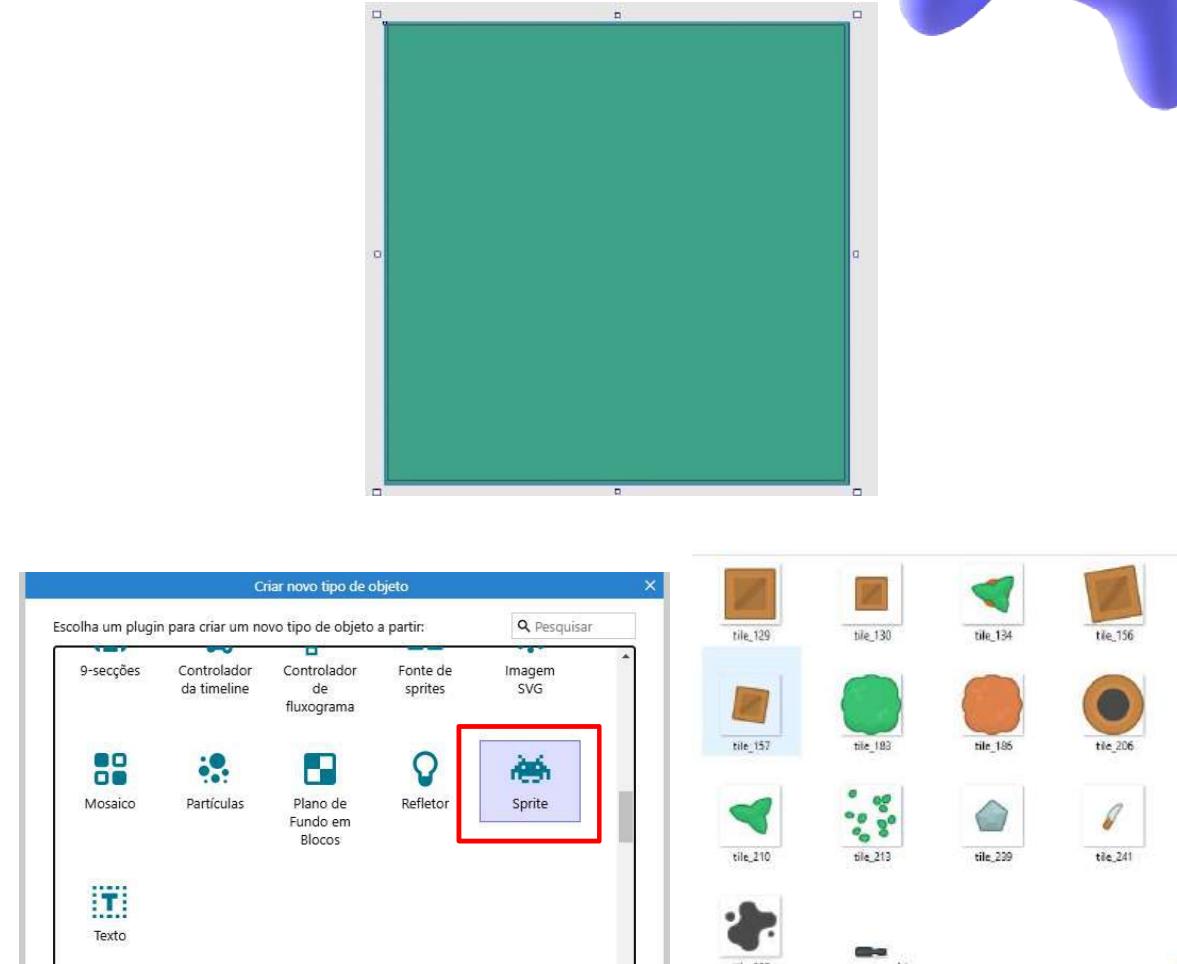


# EXERCICIO 3 - GAME 1 CONTRA 100



8 – Vamos abrir o **chao** toda a tela e deixar sobrar um pouco nas bordas.

9 – Clique 2x fora tela preenchida e add **Sprite** e comece a usar os objetos da pasta objetos do Material baixado.



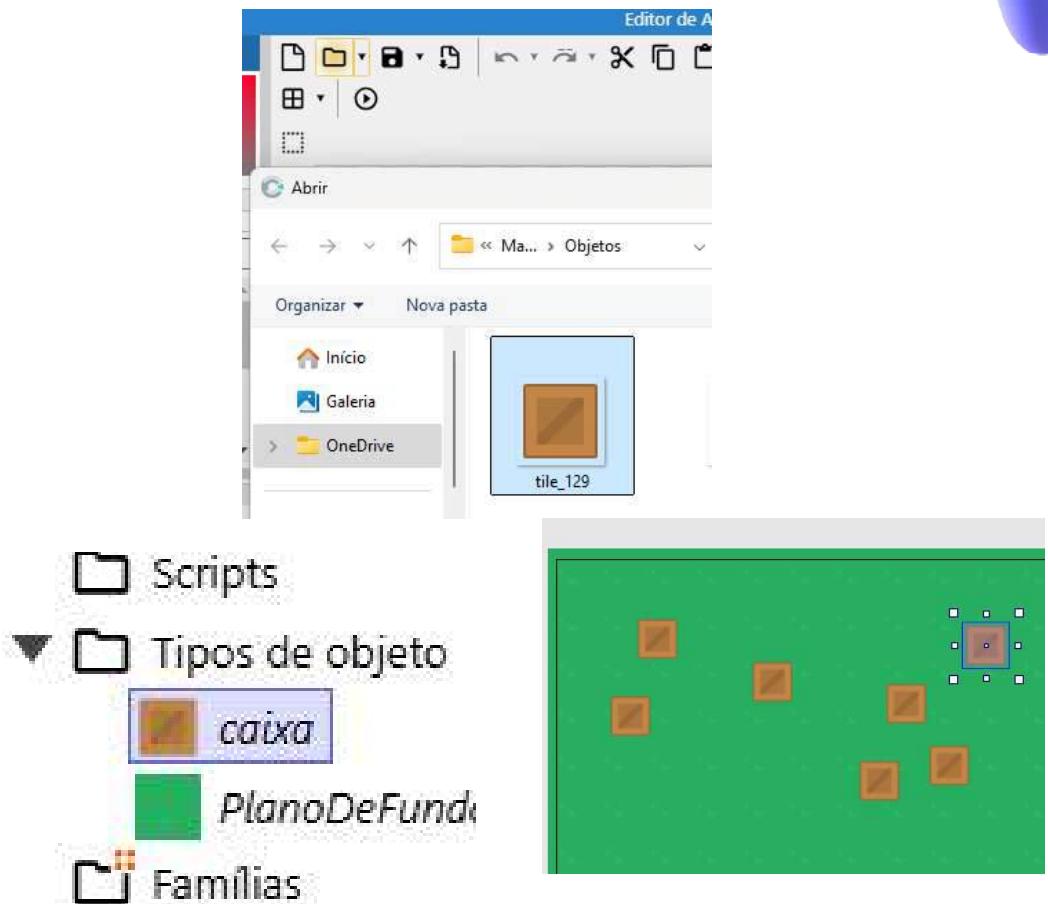
# EXERCICIO 3 - GAME 1 CONTRA 100



10 – Sprite > Carregar imagem do arquivo > tile\_129 > Abrir.

11 – Renomeei para caixa, você pode clicar e arrastar ou copiar e colar para ter várias caixas.

12 – Agora vamos espalhar os objetos para preencher o cenário, use sua criatividade e preencha do jeito que quiser.



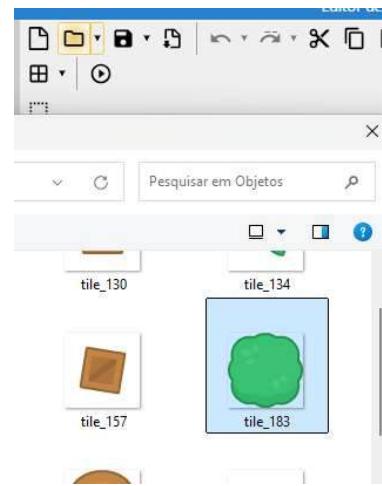
# EXERCICIO 3 - GAME 1 CONTRA 100



13 – **Sprite > Carregar imagem do arquivo > tile\_183 > Abrir.**

14 – Renomeei para moita, você pode clicar e arrastar ou copiar e colar para ter várias moitas.

15 – Agora espalhe os objetos para preencher o cenário, use sua criatividade e preencha do jeito que quiser.



# EXERCICIO 3 - GAME 1 CONTRA 100

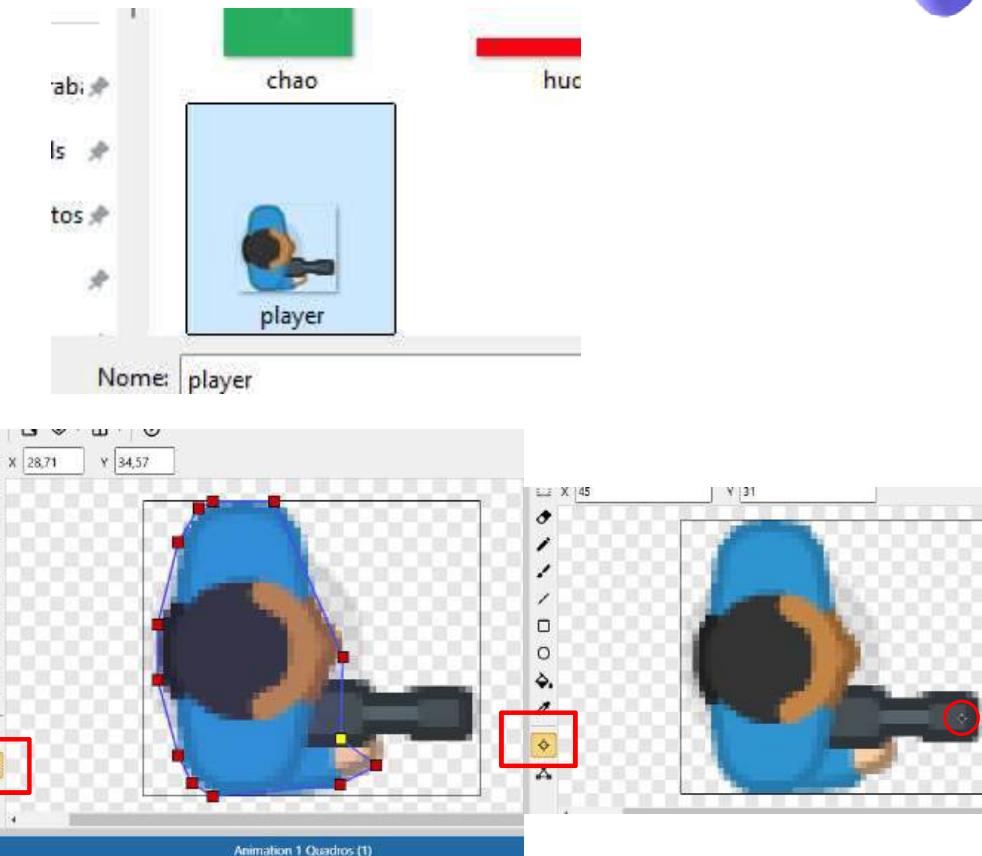


16 – Agora Vamos criar nosso player.

17 – **Sprite > Carregar imagem do arquivo > player > Abrir.**

18 – Ajuste a colisão em **Editar Polígono de Colisão** como mostrado na figura ao lado. Após ajustes renomeei para player. Ajuste também o ponto de origem no cano da arma.

A colisão do player é onde a bala do inimigo vai acertar e vai matar o player, caso acerte na arma não irá matá-lo. Por esse motivo os pontos de colisão é apenas no corpo do player.



# EXERCICIO 3 - GAME 1 CONTRA 100



19 – Add os comportamentos no player. São eles: **CentrarEm**, **Piscar**, **8Direções** e **RestritoAoLayout**.

**CentrarEm:** Faz com que a câmera siga o player.

**Piscar:** Sempre que o player pega um dano ele vai piscar.

**8Direções:** Faz o player andar em 8 direções.

**RestritoAoLayout:** Impede que o player saia do layout.

player comportamentos		
	Nome	Tipo
	CentrarEm	Centrar em
	Piscar	Piscar
	8Direções	8 Direções
	RestritoAoLayout	Restrito ao layout



20 - Agora teste para ver se está funcionando.

# EXERCICIO 3 - GAME 1 CONTRA 100



21 – Add o Objeto: **Mouse**.

22 – Vá em **Folha de Eventos1 > Adicionar evento > Sistema > A cada tick.**

23 – Add ação > Player > Definir Ângulo > **Ângulo: angle(player.X, player.Y, Mouse.X, Mouse.Y).**

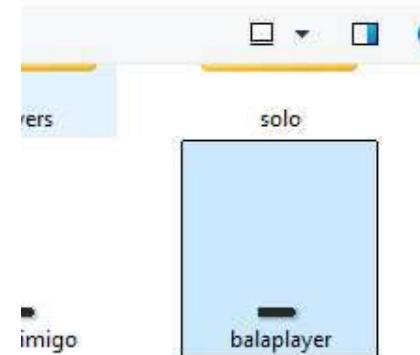
A função `angle`, faz que sempre o primeiro objeto: `player.X, player.Y`, tenha sempre o ângulo do segundo objeto `Mouse.X, Mouse.Y`, ou seja, o ângulo do player vai ser alterado pelo mouse.

The screenshot shows the RoboWork software interface. On the left, there is a library with a green icon labeled 'Mouse'. Below the library, the project tree shows a folder 'Folha de eventos' containing a sub-item 'Folha de eventos 1', which is highlighted. Underneath are 'Scripts' and other items. In the center, a dialog box titled 'Parâmetros para player: Definir ângulo' is open. It contains a description 'Ângulo: Novo ângulo do objeto, em graus.' and a text input field with the expression 'Ângulo angle(player.X, player.Y, Mouse.X, Mouse.Y)'. At the bottom of the dialog are buttons for 'Cancelar', 'Ajuda Encontrar Expressões', 'Voltar', and 'Pronto'.

# EXERCICIO 3 - GAME 1 CONTRA 100



24 – Add o Objeto: Áudio.



25 – Insira a Sprite > balaplayer.

26 – Add comportamento: Projétil e DestruirForaDoLayout.

**Projétil:** Move o objeto para frente.

**DestruirForaDoLayout:** Automaticamente destrói o objeto se ele sai da área de layout.

balaplayer comportamentos	
Nome	Tipo
Projétil	Projétil
DestruirForaDoLayout	Destruir fora do layout
<a href="#">Adicionar novo comportamento</a>	

# EXERCICIO 3 - GAME 1 CONTRA 100



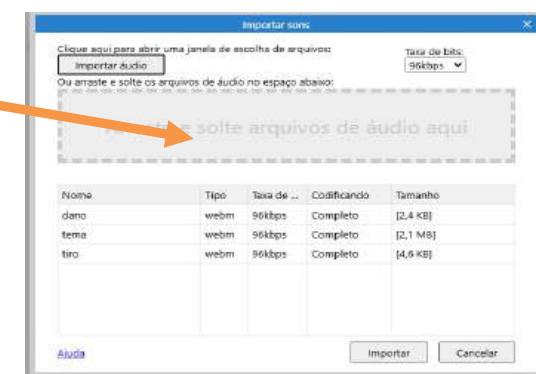
27 – Vá em folha de Layout1 > Adicionar Evento > Mouse > Ao clicar:



28 – Add ação > Player > Dar Origem a outro Objeto > balaplayer:



29 – Agora vamos importar os sons > vá na pasta sons > Clique com direito em Importar Sons > Importar áudio:



# EXERCICIO 3 - GAME 1 CONTRA 100



Vamos configurar os Inimigos:

30 – Vamos criar uma variável global >  
Nome: **vidas** > Valor inicial: **300**.

31 – Crie um Sprite > Na pasta Players:  
selecione todas as imagens.

32 – Em animação: Mude a velocidade  
para: **0**.

Adicionar variável global

Nome	vidas
Tipo	Número
Valor inicial	300
Descrição (opcional)	
Estático	<input checked="" type="checkbox"/>



Propriedades da Animação

Nome	Animation 1
Velocidade	0
Repetir	<input type="checkbox"/>
Contagem de Repetições	1

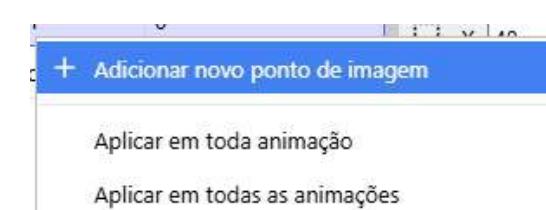
# EXERCICIO 3 - GAME 1 CONTRA 100

33 – Coloque o primeiro **ponto de origem** no braço do Inimigo.



34 – Aplicar esses pontos em todas animações.

35 – Adicione novo **ponto de origem** na ponta da arma.



36 – Aplicar esses pontos em todas animações.



# EXERCICIO 3 - GAME 1 CONTRA 100



37 – Adicione o comportamento do inimigo para Piscar e ExploradorDeRotas.

Vamos criar uma variável de instância, para que cada cópia de um inimigo tenta 100 vidas.

38 – Clicar em Variáveis de instância > Editar varáveis de Instância > Nome: vidas, Tipo: Número, Valor inicial: 100.

The image displays three screenshots of the Construct 3 editor:

- Top Screenshot:** A list titled "inimigo comportamentos" (Enemy Behaviors) showing two entries:

	Nome	Tipo
	Piscar	Piscar
	ExploradorDeRotas	Explorador de rotas

A blue button at the bottom right says "Adicionar novo comportamento" (Add new behavior).
- Middle Screenshot:** The "Variáveis de instância" (Instance Variables) panel for an enemy object (UID 231). It shows a dropdown menu for behaviors, currently set to "Piscar". Below it is a dropdown for "ExploradorDeRotas". A table shows a variable named "vidas" with a value of 30.

	Nome	Tamanho da célula
vidas	vidas	30
- Bottom Screenshot:** An open dialog titled "Editar variável da instância" (Edit Instance Variable) for the "vidas" variable. It contains fields for Name (vidas), Type (Número), Value (100), Description (optional), and a checked checkbox for "Mostrar na Barra de Propriedades" (Show in Properties Bar). Buttons for "OK" and "Cancelar" are at the bottom.

# EXERCICIO 3 - GAME 1 CONTRA 100



39 – Inimigo vai atirar também, crie uma Sprite > balainimigo.

40 – Add o comportamento Projétil e DestruirForaDoLayout.

Agora vamos fazer o inimigo atirar.

41 – Vá para Folha de Eventos 1 > Adicionar um evento > A cada X segundos > random(0.8, 4).

42 – Adicionar Ação > Inimigo > Dar origem a outro Objeto > balainimigo.

balainimigo comportamentos	
Nome	Tipo
Projétil	Projétil
DestruirForaDoLayout	Destruir fora do layout
<a href="#">Adicionar novo comportamento</a>	

← Parâmetros para System: A cada

**Intervalo (segundos):** Especifica o tempo, em segundos, entre as ações. Intervalo é limitado pela taxa de quadros (não é executadas mais de uma vez por tick).

Intervalo (segundos) `random(0.8, 4)`

← Parâmetros para inimigo: Dar origem a outro objeto

Objeto: Escolha o tipo de objeto ao qual será dado origem.

Objeto: balainimigo  
Camada: 0  
Ponto de imagem: 1  
Criar hierarquia:   
Nome do:   
Cancelar Ajuda Encontrar Expressões Voltar Pronto

# EXERCICIO 3 - GAME 1 CONTRA 100



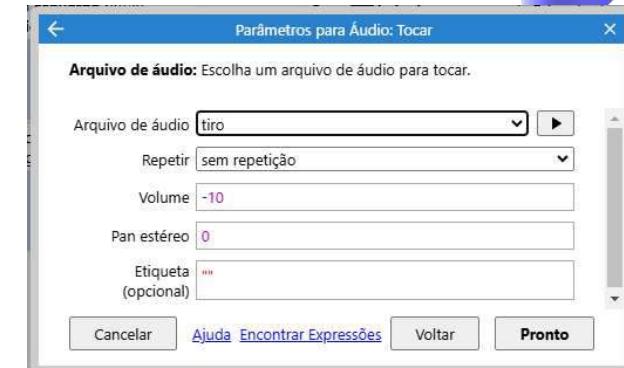
43 – Adicionar Ação > Áudio > Tocar > Tiro.

Agora vamos fazer a IA dos inimigos

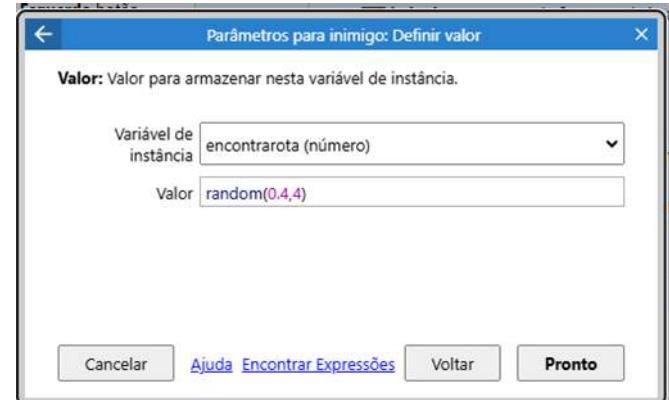
44 – Vá em Layout 1 > Variáveis de Instância > Nome: **encontrarota**, Valor: **0**.

45 – Volte para Folha de Eventos 1 > Add eventos > Inimigo > Ao criar:

46 – Add Ação > Inimigo > Definir Valor > valor: **random(0.4,4)**.



inimigo variáveis de instância				
Nome	Tipo	Valor inicial	Most...	Descriç...
vidas	Número	100	<input checked="" type="radio"/>	
encontrarota	Número	0	<input checked="" type="radio"/>	
<a href="#">Adicionar nova variável de instância</a>				

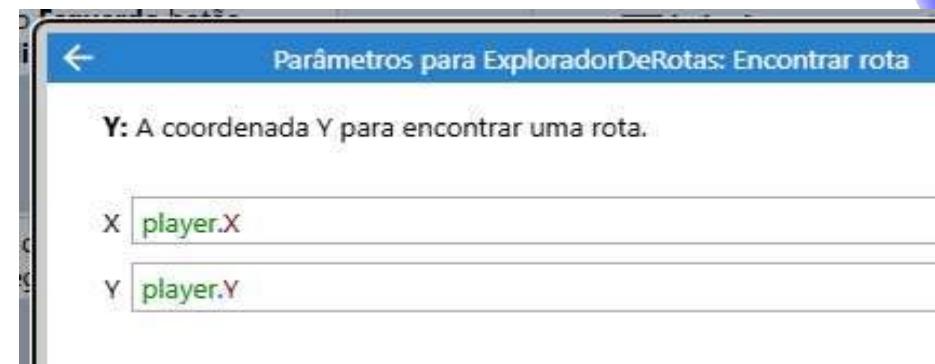


# EXERCICIO 3 - GAME 1 CONTRA 100

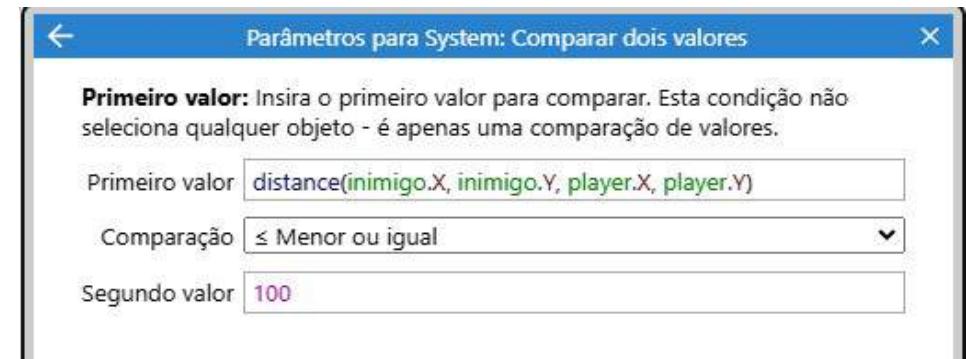


47 – Adicionar Evento > Sistema > A cada X segundos> Intervalo: **inimigo.encontrarota** .

48 – Adicionar Ação > Inimigo > **Encontrar Rota** > X: **player.X**, y: **player.Y**.



49 – Add Eventos > Sistema > Comparar dois valores: **distance(inimigo.X, inimigo.Y, player.X, player.Y)**. Comparação: **<= Menor ou igual**, Segundo valor: **100**.



# EXERCICIO 3 – GAME 1 CONTRA 100



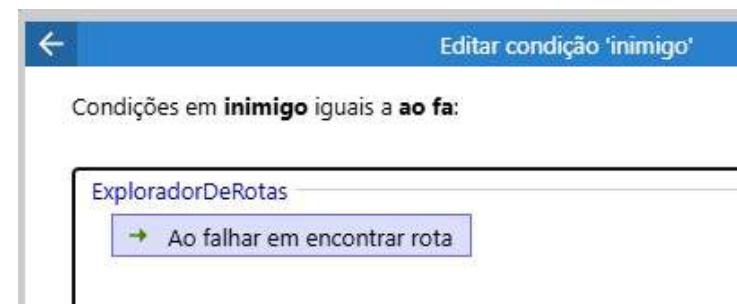
50 – Add Ação > Inimigo > ExploradorDeRotas: Parar.

51 – Adicionar Evento > Inimigo > Ao encontrar rota > Add Ação > Inimigo > Mover ao Longo da Rota.



52 – Adicionar Evento > Inimigo > Ao falhar em encontrar rota.

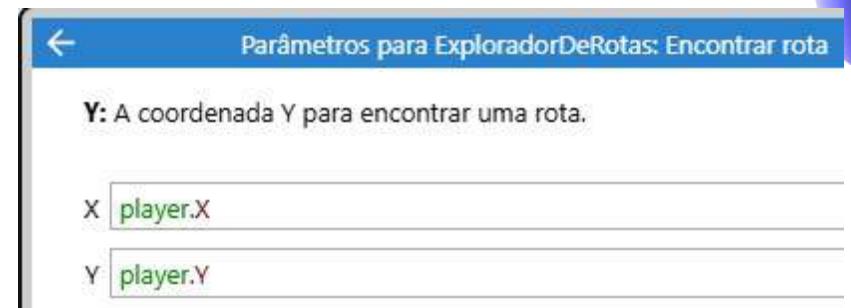
Caso falhe ao Encontrar a Rota



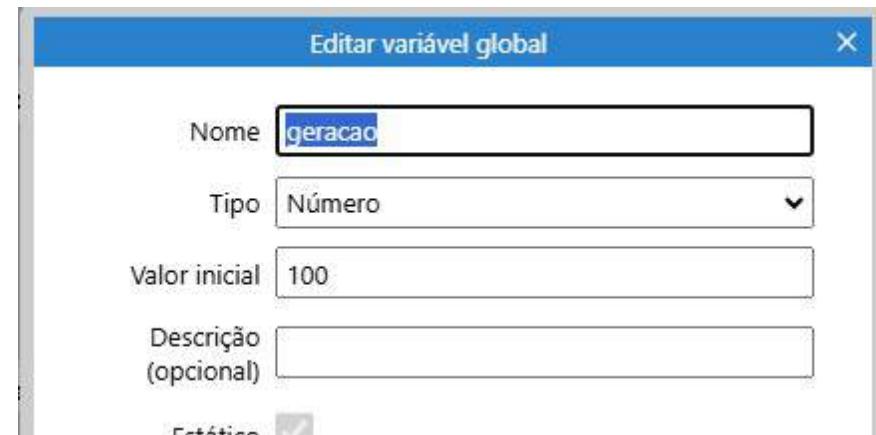
# EXERCICIO 3 - GAME 1 CONTRA 100



53 – Add Ação > Inimigo > Encontrar Rota:



54 – Clique como botão direito na parte em branco da folha de eventos 1 > Nome: geracao, Tipo: Número, Valor Inicial: 100.



# EXERCICIO 3 - GAME 1 CONTRA 100



55 – Add Eventos > Sistema > Comparar variável > Variável: geracao, Comparaçao: >Maior que.

Parâmetros para System: Comparar variável

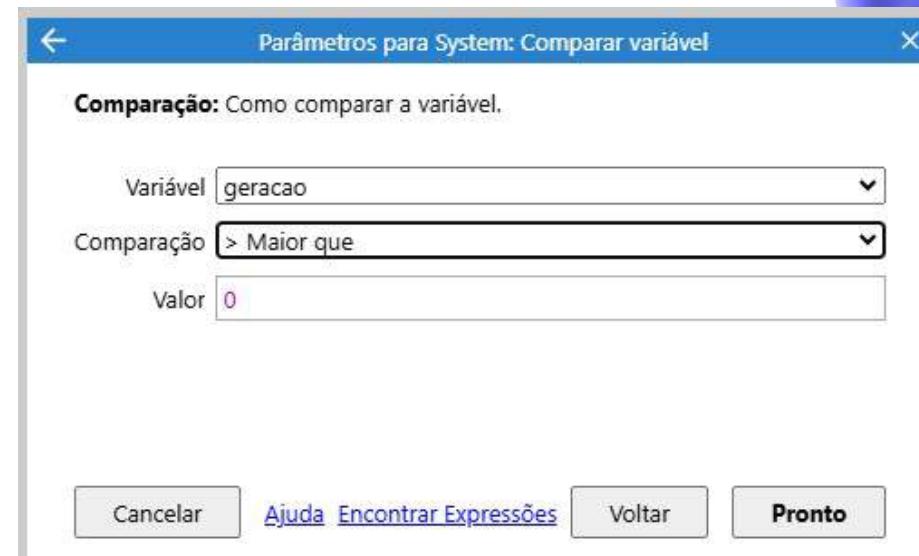
Comparação: Como comparar a variável.

Variável: geracao

Comparação: > Maior que

Valor: 0

Cancelar Ajuda Encontrar Expressões Voltar Pronto

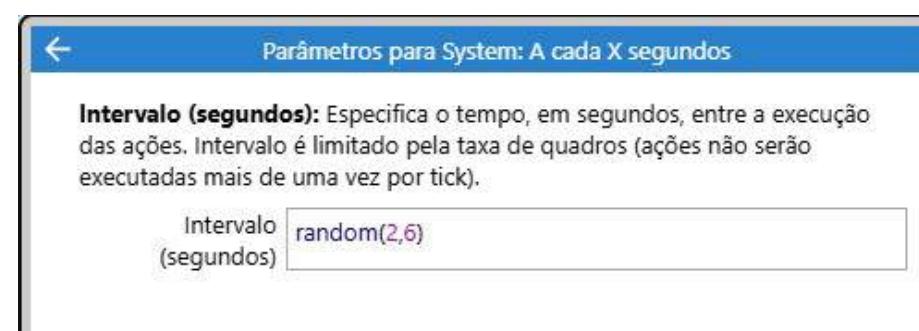


56 – Clique com botão direito em cima do Evento: geracao > Add outra condição > sistema > A cada X segundos: random(2, 6).

Parâmetros para System: A cada X segundos

Intervalo (segundos): Especifica o tempo, em segundos, entre a execução das ações. Intervalo é limitado pela taxa de quadros (ações não serão executadas mais de uma vez por tick).

Intervalo (segundos): random(2,6)



# EXERCICIO 3 - GAME 1 CONTRA 100



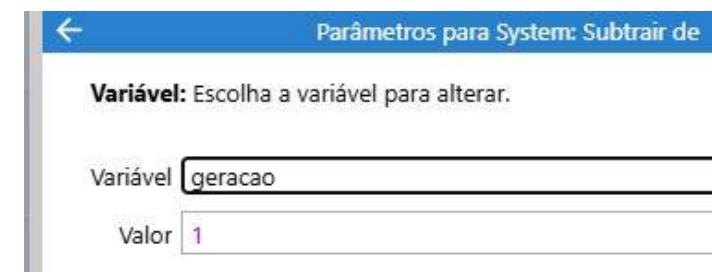
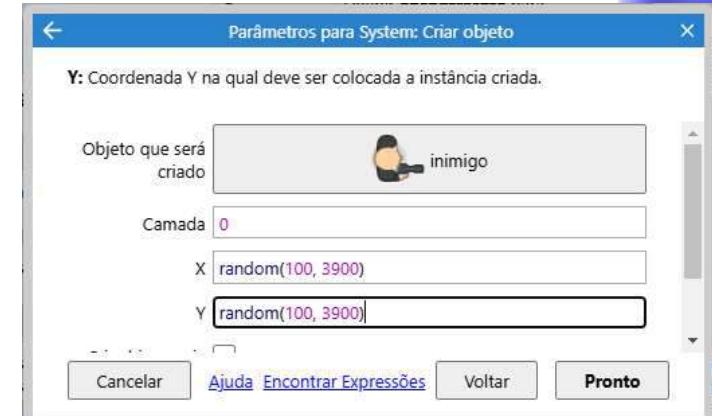
57 – Add Ação > Sistema > Criar Objeto > X: random(100, 3900), Y: random(100, 3900).

58 – Add Ação > Sistema > Subtrair de >  
Valor: 1.

Toda vez que o jogo gerar um inimigo vai subtrair  
da variável geração.

59 – Add Ação > Inimigo > Definir quadro >  
Quadro: choose(0,1,2,3,4,5,6).

Definir quadro, o jogo vai escolher as imagens dos  
personagens.



- 
- Adicionar/remover quadro de animação
  - Definir quadro
  - Definir velocidade

# EXERCICIO 3 - GAME 1 CONTRA 100



60 – Add Eventos > **balaplayer** > **Ao colidir com outro objeto** > **inimigo**.

61 – Add Ação > **Inimigo** > **Subtrair de** > **Valor: 25**. Add Ação > **Inimigo** > **Piscar**.

Add Ação > **Inimigo** > **Tocar: dano**.

The image shows a Scratch script editor window. On the left, there's a script for the 'balaplayer' sprite with a 'when green flag clicked' hat and a 'when [collide with enemy v]' event. The 'when green flag clicked' script contains an 'add [1] to [score v]' action. The 'when [collide with enemy v]' script contains three actions:

- 'Subtrair 25 de [vidas v]' (with a lightning bolt icon)
- 'Piscar: Piscar 0.1 visível 0.1 apagado por 1.0 segundos' (with a lightbulb icon)
- 'Tocar [dano v] sem repetição com o volume de 0 dB (pan estéreo 0, tag "")' (with a speaker icon)

At the bottom of the script area, there are buttons for 'Adicionar ação' and 'Adicionar...'.

62 – Criar uma variável : matou = 0.

# EXERCICIO 3 - GAME 1 CONTRA 100



63 – Add Eventos > Inimigo > Comparar variável de Instância > Variável: vidas, Comparação:  $\leq$  menor ou igual a 0.

Parâmetros para inimigo: Comparar variável de instância

Variável de instância: Escolha a variável da instância para comparar.

Variável de instância: vidas (número)

Comparação:  $\leq$  Menor ou igual

Valor: 0

64 – Add Ação > Inimigo > Destruir.

65 – Add Ação > Sistema > Adicionar a > matou: 1.

← Parâmetros para System: Adicionar a

Variável: Escolha a variável para alterar.

Variável: matou

Valor: 1

# EXERCICIO 3 - GAME 1 CONTRA 100

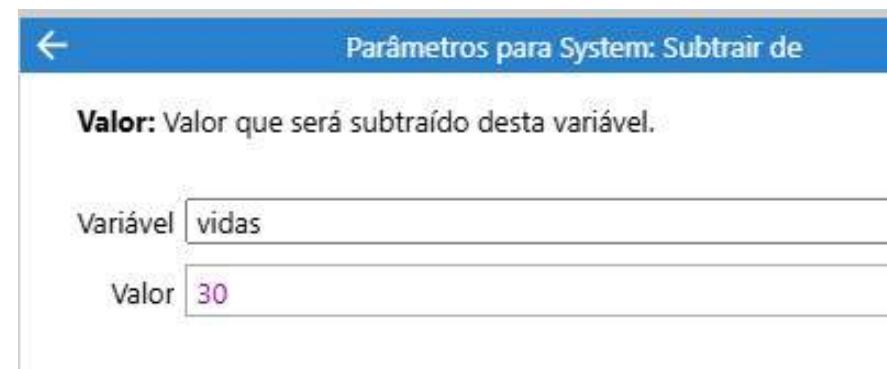


66 – Add Eventos > **balainimigo** > **Ao Colidir com Outro Objeto** > **Player**.

67 – Add Ação > **Subtrair de** > **Variável: vidas, Valor: 30**.

68 – Add Ação > **Player** > **Piscar**.

69 – Add Ação > **Tocar** > **Dano**.



balainimigo	Ao colidir com player	Sistema	Subtrair 30 de vidas
		player	Piscar: Piscar 0.1 visível 0.1 apagado por 1.0 segundos
		Áudio	Tocar dano sem repetição com o volume de 0 dB (pan estéreo 0, tag "")
Adicionar ação			Adicionar...

# EXERCICIO 3 - GAME 1 CONTRA 100



70 – Agora volte ao Layout e tranque a **Camada 0**. Crie a camada **hud** se ainda não tiver feito.

71 – Clique na camada **hud** > Insira os **Sprite** > **hud0**. Depois insira **hud1**.

72– Coloque **hud1** em cima de **hud0**, e renomeei as sprites com ambos com os nomes **hud0** e **hud1**.



# **EXERCICIO 3 - GAME 1 CONTRA 100**

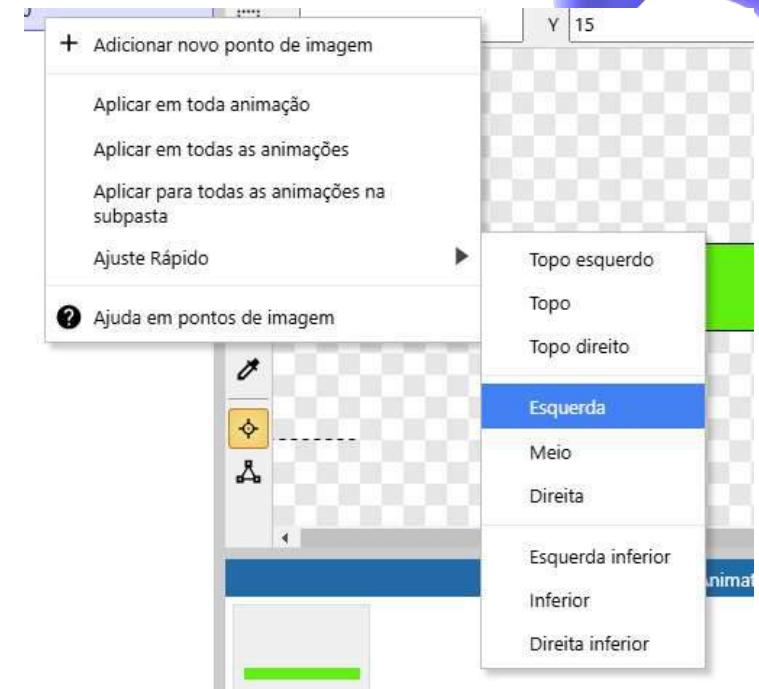


**NÃO ESQUEÇA DE ATRIBUIR O COMPORTAMENTO  
ANCORAR! Para que o Hud fique fixo na tela.**

**73 – Clique novamente na Sprite de hud1 e edite > Em origem > Botão direito > Ajuste Rápido > Esquerdo.**

74 – Inserir **Texto** > edite para textoqtde. E modifique a.

75 – Volte para folha de eventos 1 > Em eventos a cada tick > Add Ação > hud1 > definir largura > vidas.



# EXERCICIO 3 - GAME 1 CONTRA 100



NÃO ESQUEÇA DE ATRIBUIR O COMPORTAMENTO  
ANCORAR! Para que o texto fique fixo na tela.

76 – Em eventos a cada tick > Add Ação >  
**Textoqtde** > definir texto > "Restam :" & 100  
- matou.

77 – Insira um texto: Você venceu! > Add  
Evento > Sistema > Comparar variável.  
Variável: **matou**, Comparaçāo: = igual a,  
Valor: **100**. Em propriedade desmarque:  
**Visibilidade Inicial**.

78 – Add Ação > **Texto** > **Definir  
visibilidade** > **Visível**.

← Parâmetros para System: Comparar variável

**Valor:** Valor para comparar com a variável.

Variável: matou

Comparação: = Igual a

Valor: 100

Adicionar ação 'Texto'

**Definir visibilidade:** Defina se o objeto está oculto ou visível.

Aparência

Definir visibilidade

# EXERCICIO 3 - GAME 1 CONTRA 100



79 – Add Ação > Sistema > aguarde > 3 segundos.

The screenshot shows a sequence of actions in the RoboBoa software:

- Sistema: matou = 100
- Texto
- Definir visibilidade de Visível
- Sistema: [Wait 3 seconds (use time scale: Verdadeiro)]
- Sistema: Reiniciar layout

Buttons at the bottom: Adicionar ação, Adicionar...

80 - Add Ação > Sistema > Reiniciar Layout.

The screenshot shows the parameters for the "Comparar variável" (Compare variable) action:

Parâmetros para System: Comparar variável

Valor: Valor para comparar com a variável.

Variável: vidas

Comparação: ≤ Menor ou igual

Valor: 0

Buttons at the bottom: Cancelar, Ajuda, Encontrar Expressões, Voltar, Pronto

81 – Faça a copia de Sistema matou = 100.

Edite: Variável: vidas, Comparaçao: Menor ou Igual, valor: 0.

82 – Edite os demais, apague Ação Texto, e add Ação > Player > Destruir no lugar.

The screenshot shows a sequence of actions in the RoboBoa software:

- Sistema: vidas ≤ 0
- player: Destruir
- Sistema: [Wait 3 seconds (use time scale: Verdadeiro)]
- Sistema: Reiniciar layout

Buttons at the bottom: Adicionar ação, Adicionar...

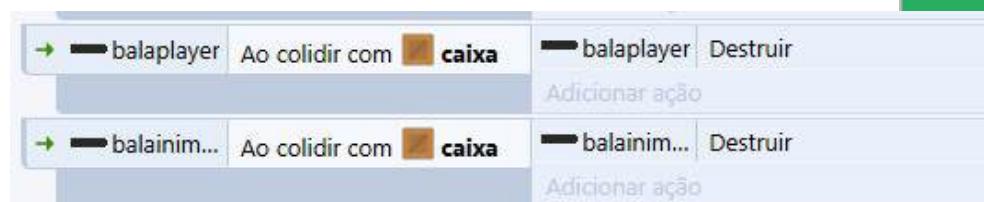
# EXERCICIO 3 - GAME 1 CONTRA 100



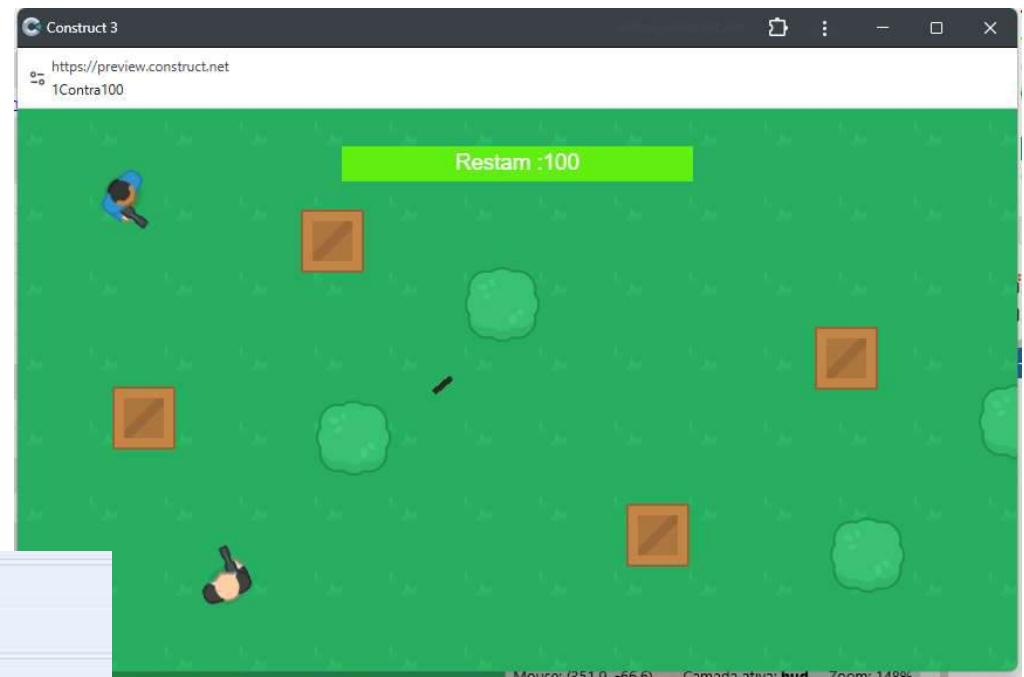
83 – Add Eventos > Sistema > Ao iniciar o Layout > Add Ação > Sistema > Redefinir variáveis globais.

84 – Add Ação > Áudio > Parar tudo. Add Ação > Tocar > Tema.

85 -



Acabooooouuu!



**mission passed!**  
**respect + 99**

**ATÉ A PRÓXIMA AULA!**

