

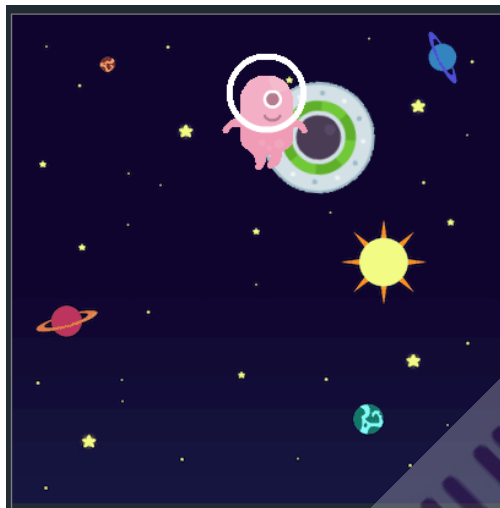
<b>Tópico</b>	<b>ANIMAÇÕES COMPLEXAS</b>	
<b>Class Description</b>	As crianças aprendem a criar animações usando códigos e linguagem javascript construída com blocos, usando palavras-chave mais específicas para a sintaxe do javascript.	
<b>Aula</b>	<b>ADV-C10</b>	
<b>Duração</b>	<b>55 mins</b>	
<b>Objetivo</b> 	<ul style="list-style-type: none"> <li>• Crie uma animação de Alienígena e OVNI no Laboratório de Jogos.</li> </ul>	
<b>Recursos Necessários</b>	<ul style="list-style-type: none"> <li>• Recursos da Professora: <ul style="list-style-type: none"> <li>○ Laptop/PC com Camera</li> <li>○ Code.org</li> <li>○ Fone de ouvido com Microfone</li> </ul> </li> <li>• Recursos do Aluno: <ul style="list-style-type: none"> <li>○ Laptop/PC com Camera</li> <li>○ Code.org</li> <li>○ Fone de ouvido com Microfone</li> </ul> </li> </ul>	
<b>Estrutura da aula</b>	<b>Introdução</b> <b>Atividade Dirigida pela Professora</b> <b>Atividade Dirigida pelo Aluno</b> <b>Fechamento</b> <b>Dicas e sugestões do projeto</b>	<b>5 min</b> <b>10 min</b> <b>30 min</b> <b>5 min</b> <b>5 min</b>
<b>SESSÃO DE INTRODUÇÃO - 5 min</b>		
<p><b>A professora inicia apresentação de slides</b>  <b>dos slides 1 a 21.</b>  Consulte as anotações da oradora e siga as instruções em cada slide.</p>		
<p><b>ATIVIDADE DA PROFESSORA</b>  <b>- 10 min</b></p>		

## A Professora Inicia o Compartilhamento de Tela

### Dizer




### Demonstração




Se você olhar mais de perto a tela do jogo, verá que duas coisas estão acontecendo aqui -



1. O alienígena  quica para trás quando toca nas bordas e sua cor continua mudando.



2. O OVNI  está girando e sua cor muda constantemente.

Forma completa de OVNI: Objeto Voador Não Identificado  
- OVNI é um objeto não identificado detectado pelos humanos e pesquisado pelos cientistas.

Então, vamos criar essa animação legal hoje em um novo laboratório, o laboratório de jogos. Vamos começar!

### Fazer

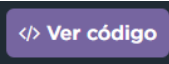



### Execute

[Atividade da Professora 1 - CÓDIGO COMPLETO](#)

Mostre a animação do aplicativo para o aluno.

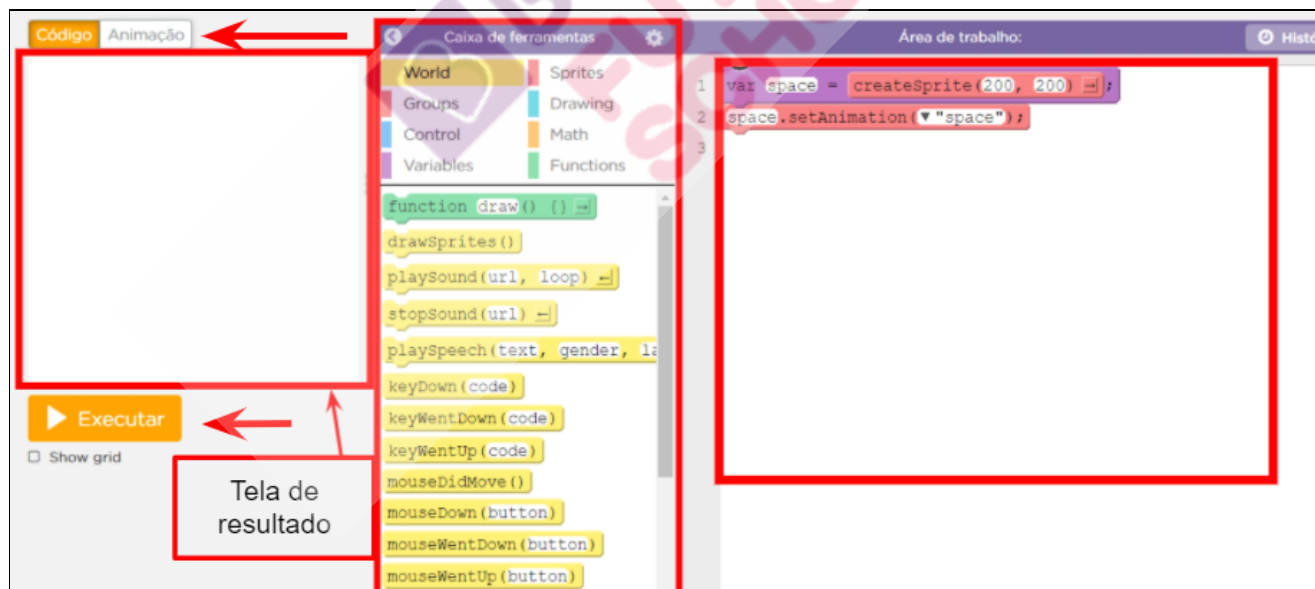
**OBSERVAÇÃO:** Abra a **Atividade-da-Professora-2**. Nela, alguns códigos e recursos estão pré-feitos para você. Você deve adicionar os ativos restantes e explicar o código ao aluno paralelamente.


Quando clicamos no link da atividade, uma página aparece e precisamos clicar em , então a página de código deverá abrir. Em seguida, clicamos no botão  e fazemos o login in no **code.org**, caso não estejamos logados.

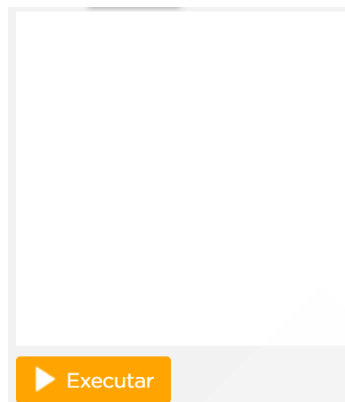
Este projeto foi criado usando minha conta. Então, mais tarde, quando terminarmos a programação, podemos compartilhar isso com nossos amigos!

### Visão geral do Laboratório de Jogos

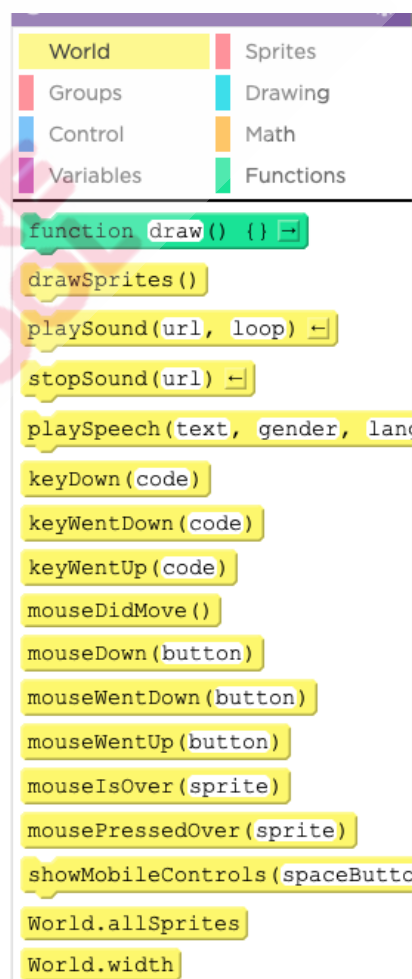
É assim que é o Laboratório de Jogos:



1. No canto superior esquerdo, tenho duas opções: Código e Animação 
  - **Código** é para adicionar um bloco de código
  - **Animação** é para enviar os ativos que poderão ser usados no jogo.



2. Então temos a tela e um botão **Executar**. Assim, quando o botão **Executar** for pressionado, todo o código disponível começará a aparecer na tela.



3. Então temos esta seção de **Caixa de Ferramentas**, onde podem ser encontrados todos os blocos de programação.
4. Em seguida, temos nossa seção de **área de trabalho**, onde podemos colocar todos os nossos códigos.

Área de trabalho:
Histórico de Versões
Mostrar texto

```

1 var espaco = createSprite(200, 200);
2 espaco.setAnimation(▼ "espaco");

```

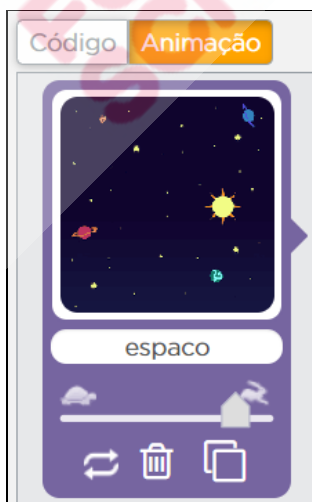
## Começando: como adicionar animações e ativos

### Biblioteca de animação

1. Primeiro, vamos verificar quais ativos já temos em nosso projeto. Para isso, precisamos clicar na guia **Animação** no canto superior esquerdo da tela.

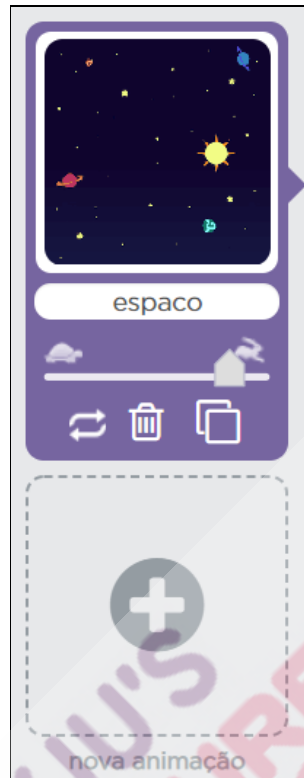
Código Animação

2. Se clicarmos em **Animação**, podemos ver que temos apenas a imagem de fundo.



Isso significa que precisamos adicionar mais dois ativos: Alien e OVNI.

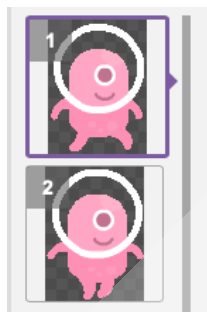
3. **Adicione uma nova animação para o Alienígena:** Então, para adicionar uma nova animação, clique na **nova animação** e procure por imagens de alienígenas.



Aqui devemos obter uma variedade de imagens.



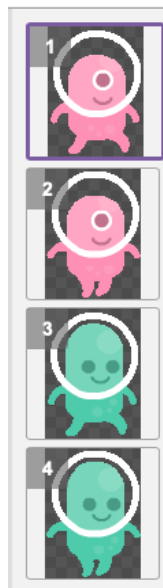
Se passarmos o mouse sobre qualquer imagem, sabemos que algumas das imagens são estáticas (imagens não animadas) e algumas das imagens estão no formato gif (imagens animadas). Então, vamos selecionar uma imagem gif e clicar nela. Agora



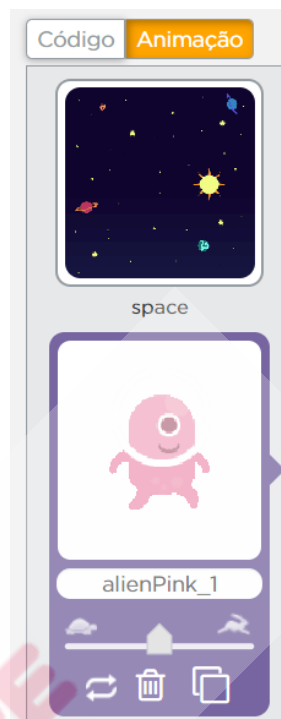
observamos que nosso ativo foi adicionado . Quando eu mostrei a demonstração, você notou que a cor do alienígena ficava mudando, certo? Então, para isso, precisamos adicionar mais 1 imagem de alienígena.



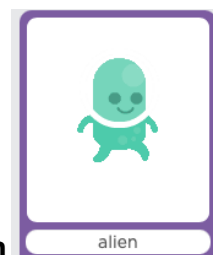
Então clique em **Adicionar novo quadro (Add new frame)** e procure novamente as imagens de alienígenas, em seguida selecione a imagem de outro



alienígena. Todos os quadros adicionados ficarão assim:

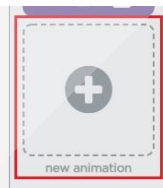


No lado esquerdo, você também pode ver a prévia:

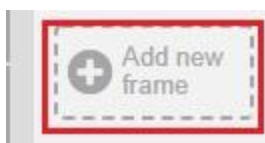


Agora vamos renomear **alienPink\_walk\_1** para **alien**. A razão é que precisamos nos referir a esse nome durante a programação. Seria melhor se o nome não fosse complexo para que se tornasse fácil de lembrar.

#### 4. Adicionar nova animação para o OVNI: Para isso, clique em **Nova Animação**



**(new animation)**, procure imagens de OVNI's (UFOs) e selecione uma imagem. Como você viu na demonstração, a cor do OVNI ficava mudando, então, para fazer isso, vamos adicionar mais um ativo. Clique em **Add new frame**





novamente e procure as imagens de OVNI's.





Biblioteca de animação


ufo


  
ufo


  
sticker\_25

  
ufoBlue


  
ufoGreen


  
ufoRed

  
ufoYellow

  
ufo\_02

Adicione mais uma imagem de OVNI. A combinação dessas duas imagens de OVNI's criará um efeito de animação para o OVNI.

  
ufoYellow\_1

  
ovni

Agora renomeie **ufoYellow\_1** para **ovni**.

Com isso, concluímos a adição de ativos.

## Programação e explicação

Código Animação

Agora vamos começar a programar. Para isso, clique no botão **Código** no canto superior esquerdo da tela.

Código Animação

- 1. Adicionar ativos à tela:** Temos que criar **sprites** para cada um dos ativos (espaço, ufo e alien).

O que é um **sprite**?

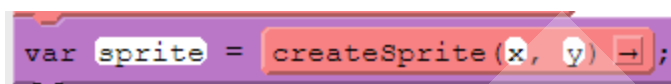
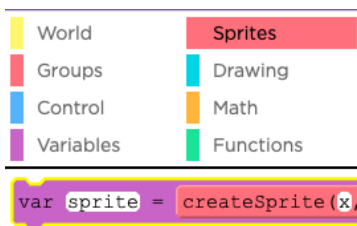
Um sprite é o nome dado a um personagem ou objeto que é basicamente uma imagem exibida na tela. Agora carregamos as imagens e as nomeamos na seção Animação. Esses são os recursos do jogo mas, para usar esses recursos, temos que escrever o código. E para usar esses ativos, primeiro temos que criar um objeto sprite. É como se criássemos um componente para o sprite e depois colocássemos a imagem desejada nele. Da mesma forma, primeiro criamos um componente de sprite e, em seguida, definimos uma animação específica (ativo) para esse sprite.

→ **Criando um sprite para o 'espaço':** Você pode ver na tela que o código para um sprite já foi criado, o que está adicionando um sprite espacial na tela.

```
var espaco = createSprite(200, 200);
espaco.setAnimation("espaco");
```

- Criação de um elemento sprite:** Um elemento sprite é criado em uma determinada posição e é armazenado em uma variável. Você encontrará o código para criar um sprite e armazená-lo em uma

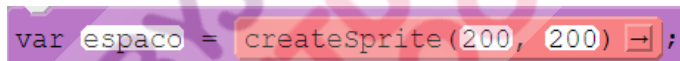
variável `var sprite = createSprite(200, 200);` na seção



`var nome_da_variavel = createSprite( x , y )`

A função `createSprite()` cria um sprite em uma posição especificada e o armazena dentro de uma variável. Aqui:

- O **nome\_da\_variável** é **espaco**.
- O **x** representa o valor da coordenada X onde este sprite será posicionado. Aqui, a posição x é 200.
- O **y** representa o valor da coordenada Y onde o sprite será posicionado. Aqui, a posição y é 200.



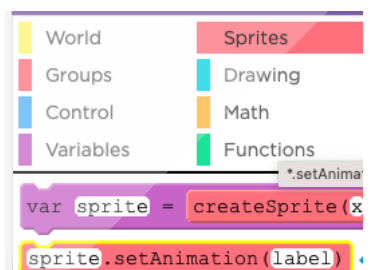
- **Atribuir os ativos ao sprite:** Agora, precisamos atribuir os ativos (as animações que criamos) ao sprite e armazená-lo em uma variável **espaco**.

Você encontrará o código para atribuir o ativo à variável

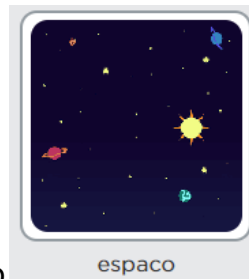
`sprite.setAnimation("animation_1");`

na

seção



`nome_da_variavel.setAnimation(' nome_da_animação '`



O **espaco** é o nome da animação portanto, selecionaremos o **espaco** da lista suspensa que está dentro da função **setAnimation()**.

```
espaco.setAnimation(▼ "espaco");
```

Agora, vamos criar mais 2 sprites para as outras 2 animações: OVNI e Alienígena. Faremos exatamente da mesma forma que criamos a animação do espaço.

- **Criando um sprite para o ovni:** Agora, vamos criar um sprite para o ovni e definir sua posição. Você pode escolher qualquer posição que desejar. Para obter a posição, mova o mouse sobre a tela de jogo e você obterá as coordenadas x e y.



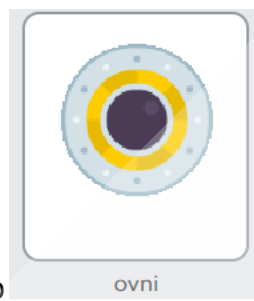
- 1) Definimos a posição do sprite em  $x = 250$  e  $y = 100$  e o armazenamos em uma variável chamada **ovni**. Você pode definir quaisquer valores x e y de acordo com sua escolha.

```
var ovni = createSprite(250, 100);
```

- 2) Portanto, mude o nome da variável de **sprite** para **ovni**.

- 3) Agora, use este bloco `sprite.setAnimation(label)`, mude o nome da variável e o ativo para aquele que criamos, como este:

```
ovni.setAnimation(▼"ovni");
```



O **ovni** é o nome da animação portanto, precisamos escrever este nome dentro da função `setAnimation()`.

- **Criando um sprite para o alienígena:** da mesma forma, precisamos criar outro sprite para o alien e definir o nome da animação como **alien**.

```
var alien = createSprite(200, 200);
alien.setAnimation(▼"alien");
```



Porque **alien** é o nome da animação que nós criamos.

- 2. Defina a velocidade do alienígena:** definir a velocidade do alienígena é feito através da definição das velocidades nas direções **X** e **Y**.

**A velocidade (Velocity)** representa a velocidade com que o objeto se move em uma direção específica. Quanto maior a velocidade, maior a rapidez da animação.

**VelocityX** representa a velocidade com que o objeto se move no eixo X, que é a direção horizontal

Para definir a velocidade X e Y de qualquer sprite, precisamos usar este bloco de código:

```
sprite.velocityX
sprite.velocityY
```

Você encontrará este bloco de código na seção

Sprites

- 1) Então, vamos arrastar este bloco `sprite.velocityX` para nosso código.

```
var espaco = createSprite(200, 200);
espaco.setAnimation(▼ "espaco");

var ovni = createSprite(250, 100);
ovni.setAnimation(▼ "ovni");

var alien = createSprite(200, 200);
alien.setAnimation(▼ "alien");

sprite.velocityX = 0;
```

- 2) Como armazenamos o sprite do alien dentro de uma variável **alien**, vamos atualizar o nome da variável de **sprite** para **alien**.

```
alien.velocityX
```

- 3) Então, vamos definir a velocidade do alien na direção **X** para **2**.

```
alien.velocityX = 2;
```

Isso fará com que o alienígena se mova na direção X a 2 pixels por quadro. Se você definir a velocidade X para um número maior, o alienígena se moverá na direção X mais rápido.

- 4) Da mesma forma, a fim de definir a velocidade para Y, precisamos definir a velocidade alienígena na direção Y para **3**.

**VelocityY** representa a velocidade com que o objeto se move no eixo Y, que é a direção vertical

```
alien.velocityY = 3;
```

Isso fará com que o alienígena se mova na direção Y a 2 pixels por quadro. Se você definir a velocidade Y para um número maior, o alienígena se moverá na direção Y mais rápido.

- 3. Faça os sprites ficarem dentro da tela do jogo:** Agora, chamaremos a função `createEdgeSprites()`:

```
12 createEdgeSprites();
```

Isso irá preservar as bordas da tela do jogo. Eventualmente, isso nos ajudará a evitar que os sprites saiam da tela do jogo. Esta função cria uma variável chamada **bordas** para trancar as bordas. Usaremos essa variável mais tarde.

Você encontrará o bloco `12 createEdgeSprites();` na seção **Sprites**.

```
var espacio = createSprite(200, 200);
espacio.setAnimation(▼ "espaco");

var ovni = createSprite(250, 100);
ovni.setAnimation(▼ "ovni");

var alien = createSprite(200, 200);
alien.setAnimation(▼ "alien");

alien.velocityX = 2;
alien.velocityY = 3;
createEdgeSprites();
```

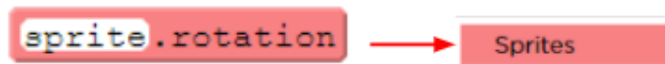
- 4. Defina uma função draw():** Agora, vamos definir uma função **draw()**. Se houver um bloco de código que você deseja que seja executado continuamente, ele deve ser escrito dentro de uma função. Assim, o código escrito dentro de uma função se torna reutilizável e você não precisa escrever o mesmo código várias vezes. A função **draw()** é executada continuamente até que o programa seja interrompido. Portanto, qualquer código escrito dentro desse bloco de código é executado continuamente. Portanto, podemos fazer nossa animação ganhar vida com esta função.

```
17 function draw() {
18
19 }
```

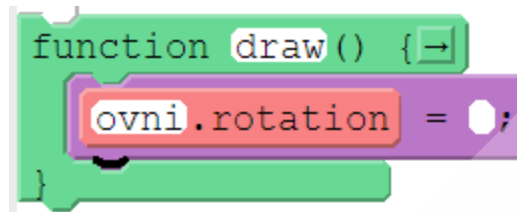
function draw() {} → World

- 5. Gire o OVNI continuamente:** Queremos que o sprite do OVNI gire continuamente. Portanto, precisamos colocar o código para girar o **ovni** continuamente dentro da função **draw()** enquanto a função draw é executada continuamente.





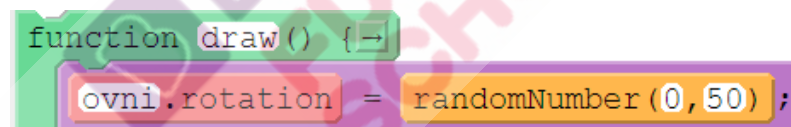
- 1) Mude o nome da variável de sprite para **ovni**, pois queremos girar o sprite do ovni.



- 2) Vamos girar o sprite do ovni em uma velocidade aleatória.

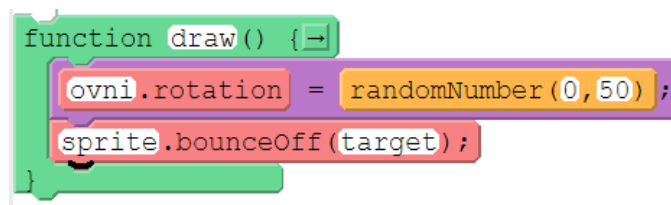


O bloco de código acima irá gerar números aleatórios entre 1 a 10. Se você aumentar ou diminuir o número, aumentará ou diminuirá a velocidade de rotação, respectivamente. Aqui, vamos gerar números aleatórios entre 0 e 50. Você pode optar por gerar quaisquer números aleatórios que desejar.



6. **Executar a ação:** agora precisamos definir a ação a ser executada quando o alienígena tocar nas bordas da tela do jogo. Queremos que nosso alienígena quique sempre que tocar na borda da tela.

- 1) Precisamos usar o bloco **sprite.bounceOff(target)** que você encontrará na seção **Sprites**. Adicione-o dentro da função draw.



- 2) Atualize o nome da variável de sprite para **alien**, porque precisamos adicionar esta funcionalidade ao alien.

```
function draw() {  
  ovní.rotation = randomNumber(0,50);  
  alien.bounceOff(target);  
}
```

- 3) Digite **edges (bordas)** dentro da função **bounceOff** e pressione **Enter**.

```
alien.bounceOff(edges);  
  
function draw() {  
  ovní.rotation = randomNumber(0,50);  
  alien.bounceOff(edges);  
}
```

**Observação:** esta variável só pode ser usada porque especificamos o

```
createEdgeSprites();
```

, que bloqueia todas as quatro bordas da tela.

Agora, sempre que o sprite alienígena tocar na borda da tela, ele quicará.

7. **Exibir todos os sprites:** agora, precisamos chamar a função **drawSprites()**. Esta função exibe todos os sprites na tela do laboratório de jogo. Se não mencionarmos isso, nenhum dos sprites criados aparecerá na tela.

```
function draw() {  
  ovní.rotation = randomNumber(0, 50);  
  alien.bounceOff(edges);  
  drawSprites();  
}
```

drawSprites() → World

**Código Completo:**

```
1  var espaco = createSprite(200, 200) ;
2  espaco.setAnimation(▼ "espaco") ;
3
4  var ovni = createSprite(250, 100) ;
5  ovni.setAnimation(▼ "ovni") ;
6
7  var alien = createSprite(200, 200) ;
8  alien.setAnimation(▼ "alien") ;
9
10 alien.velocityX = 2;
11 alien.velocityY = 3;
12 createEdgeSprites();
13
14 function draw() {
15   ovni.rotation = randomNumber(0, 50) ;
16   alien.bounceOff(edges);
17   drawSprites();
18 }
19
```

**Resultado:**



## A Professora Para o Compartilhamento de Tela

**ATIVIDADE DO ALUNO**  - 30 mins

### O Aluno Inicia o Compartilhamento de Tela

- Peça ao aluno que pressione a tecla **ESC** para voltar ao painel.
- Oriente o aluno a iniciar o compartilhamento de tela.
- A professora entra em tela cheia.

**Dizer**  


**Fazer**  


Agora é sua vez.

Agora que você sabe como criar animações e sprites, crie as funcionalidades para os sprites para que nossa atividade seja concluída.

**Observação: o aluno recebeu o código do espaço, da animação do ovni e do sprite espaço. Peça ao aluno para realizar a seguinte tarefa.**

Agora faça o seguinte:

1. Crie sprites de OVNI e de alienígenas.
2. Defina a velocidade X e Y do alienígena e chame a função `createEdgeSprites` para bloquear as bordas

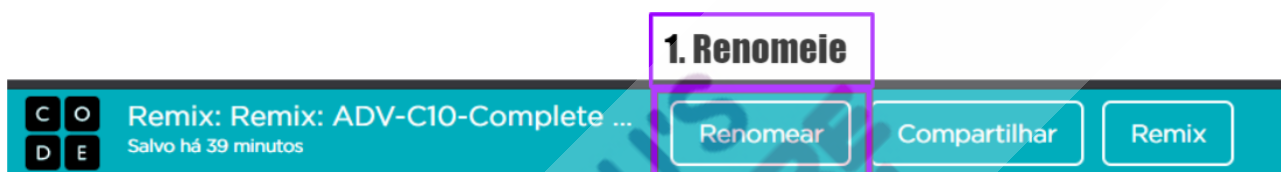
[Atividade da Professora 1 - LABORATÓRIO DE JOGOS](#)

da tela do jogo.

3. Crie a função de desenho (*draw*) e, dentro dela, adicione o seguinte código:
  - a. Código para girar o sprite do ovni em uma velocidade aleatória.
  - b. Código para quicar o alienígena de volta quando ele tocar na borda da tela do jogo.
  - c. Chamar a função `drawSprites` para exibir todos os sprites na tela do jogo.

### Para compartilhar -

1. Clique no botão **Renomear**.



2. Digite o nome do projeto.

 A screenshot of a text input field with a teal border. Inside the field, the text 'ADV-C10' is entered. A large, faint 'BYJU'S FUTURE SCHOOL' watermark is visible in the background.


3. Agora clique em **Salvar** para salvar o nome do projeto.
4. Clique em **Compartilhar** para compartilhar o jogo com seus amigos.

### 2. Compartilhe



 A screenshot of a form titled 'Submit Project URL'. It has a light gray background. Below the title, there is a text input field with the placeholder 'Enter Your Address'. To the right of the input field is a teal button with the text 'Submit URL'.

**A Professora Orienta o Aluno a Parar o Compartilhamento de Tela**

**SESSÃO DE FECHAMENTO - 5 mins**

**A professora inicia apresentação de slides dos slides 22 a 23.**  
Consulte as anotações da oradora e siga as instruções em cada slide.



**DICAS E SUGESTÕES PARA O PROJETO - 5 mins**

**A professora inicia apresentação de slides do slide 24.**  
Consulte as anotações da oradora e siga as instruções em cada slide.



**A Professora Inicia o Compartilhamento de Tela**

**Dizer**



**Fazer**



**CORRIDA DE CARROS**

**Objetivo do projeto:**

Hoje você aprendeu sobre sprites e como criar animações e sprites no laboratório de jogos.

Neste projeto, você terá que fazer um jogo de corrida de carros que é jogado automaticamente sem qualquer entrada. O jogo não deve precisar de nenhuma intervenção. Faça o carro azul ganhar a corrida.

**História:**

Vimos muitos jogos de corrida de carros. Nesses jogos, todos os pilotos de carros dão o melhor de si para vencer a corrida. Mas apenas um vence.

Criaremos um jogo de corrida de carros em que os carros não são controlados pelo usuário, mas pelos valores que você fornece ao programar o jogo. Quanto mais baixos forem os valores, mais rápido o carro deve se mover na direção superior.


Abra o link da **Resolução do Projeto** e faça uma demonstração do projeto para o aluno

Estou muito animada para ver a solução do seu projeto e sei que você se sairá muito bem.

Até breve!

**A Professora Clica em**

**✕ Terminar Aula**

**A professora inicia apresentação de slides**  **dos slides 25 a 26.**  
Consulte as anotações da oradora e siga as instruções em cada slide.

### **Atividades Adicionais**

#### **ATIVIDADES ADICIONAIS DO ALUNO**



#### **O Aluno Inicia o Compartilhamento de Tela**

#### **Atividade Adicional:**

**Criar mais um alienígena e movê-lo apenas para cima e para baixo.**

1. Criar mais uma animação de alienígena e renomeá-la como **alien2**.



2. Então, crie um novo sprite e defina a animação recém-criada para este sprite.



```
var alien2 = createSprite(50, 300);
alien2.setAnimation("alien2");
```

3. E depois, defina a velocidade na direção Y para qualquer valor que deseje. Quanto mais alto for o número, mais rápido o alienígena se moverá. Quanto mais baixo for o número, mais lento o sprite irá mover-se.

```
alien2.velocityY = 1;
```

Precisamos definir a velocidade apenas na direção Y, pois queremos que o alienígena se mova apenas para cima e para baixo.

4. Então, para se certificar de que o alienígena não ultrapasse as bordas da tela, faça o alien2 saltar para trás sempre que tocar na borda. Para isso, escreva o código dentro da função draw(), uma vez que o código dentro da função draw() é executado continuamente e, dessa forma, podemos verificar constantemente se o alienígena tocou na borda ou não. Se sim, faça com que ele quique na borda.

```
function draw() {
  ovni.rotation = randomNumber(0, 50);
  alien.bounceOff(edges);
  alien2.bounceOff(edges);
  drawSprites();
}
```

Número da Atividade	Nome da atividade	Links
Atividade da Professora 1	JOGO COMPLETO	<a href="https://studio.code.org/projects/gamelab/VD662M1x7dRQhvasm_Av5ovGmUFz_yQJQ9Mp22YuR9M">https://studio.code.org/projects/gamelab/VD662M1x7dRQhvasm_Av5ovGmUFz_yQJQ9Mp22YuR9M</a>
Atividade da Professora 1	LABORATÓRIO DE JOGOS	<a href="https://studio.code.org/projects/gamelab/eg5vaBG T-Jk-blvw9vPqCcH9eGikj-t8xf0x5SooJY0">https://studio.code.org/projects/gamelab/eg5vaBG T-Jk-blvw9vPqCcH9eGikj-t8xf0x5SooJY0</a>
Atividade do Aluno 1	LABORATÓRIO DE JOGOS	<a href="https://studio.code.org/projects/gamelab/mxgDgFr L-6RwTQb8t6oGPS8MsVctgMRd0DPJz2IfohM">https://studio.code.org/projects/gamelab/mxgDgFr L-6RwTQb8t6oGPS8MsVctgMRd0DPJz2IfohM</a>
Resolução do Projeto	CORRIDA DE CARROS	<a href="https://studio.code.org/projects/gamelab/thHCRIK PoEi118jP7RwglzhLu_XT73p_Mu04Wbi2aDE">https://studio.code.org/projects/gamelab/thHCRIK PoEi118jP7RwglzhLu_XT73p_Mu04Wbi2aDE</a>
Link do recurso visual de referência para a professora	Link do recurso visual	<a href="https://s3-whjr-curriculum-uploads.whjr.online/7557cffa-e28e-48f5-b745-ad7bd78e1094.html">https://s3-whjr-curriculum-uploads.whjr.online/7557cffa-e28e-48f5-b745-ad7bd78e1094.html</a>
Referência da Professora: Questionário de aula	Questionário de aula	<a href="https://s3-whjr-curriculum-uploads.whjr.online/302009b7-0129-4a4e-b6fb-4e4d34ec0473.pdf">https://s3-whjr-curriculum-uploads.whjr.online/302009b7-0129-4a4e-b6fb-4e4d34ec0473.pdf</a>
Referência da Professora - NÃO SE DEVE ADICIONAR AO PAINEL DE LINKS DE ATIVIDADES		
Link do Recurso Visual para Referência da Professora	Recurso visual	<a href="https://s3-whjr-curriculum-uploads.whjr.online/4aed090f-dfe3-4499-8f31-f4fb8bef9384.html">https://s3-whjr-curriculum-uploads.whjr.online/4aed090f-dfe3-4499-8f31-f4fb8bef9384.html</a>