

# Animações na Unity

Movimento para suas cenas

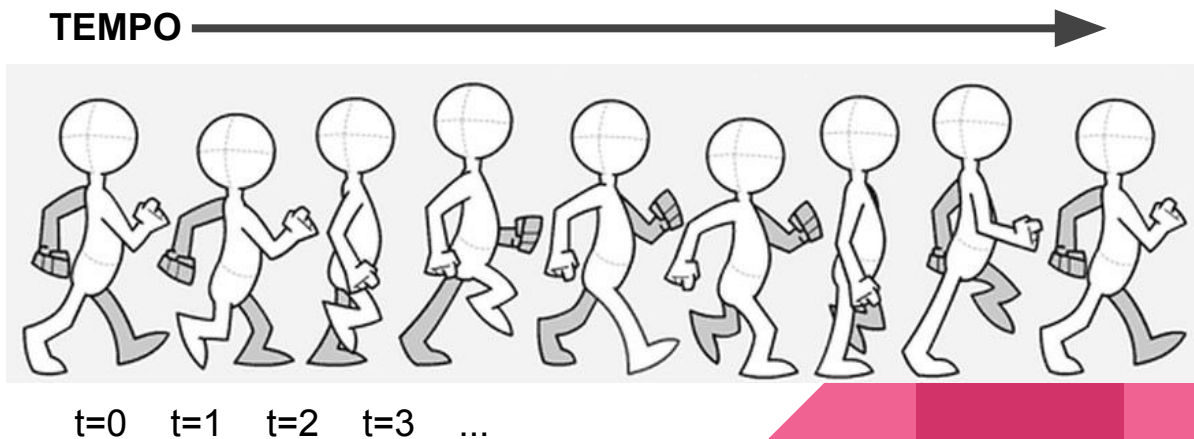
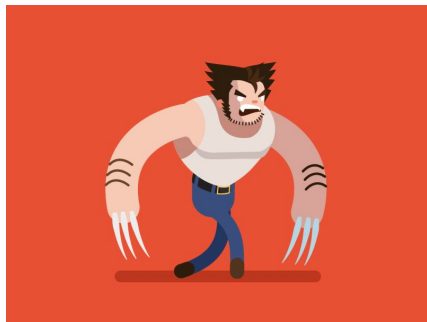
# Topicos

- Estrutura de uma animação
- Objetos AnimationClip e AnimatorController
- Sistema Mecanim
  - Transições, parâmetros e condicionais
- Controle de animação interativa



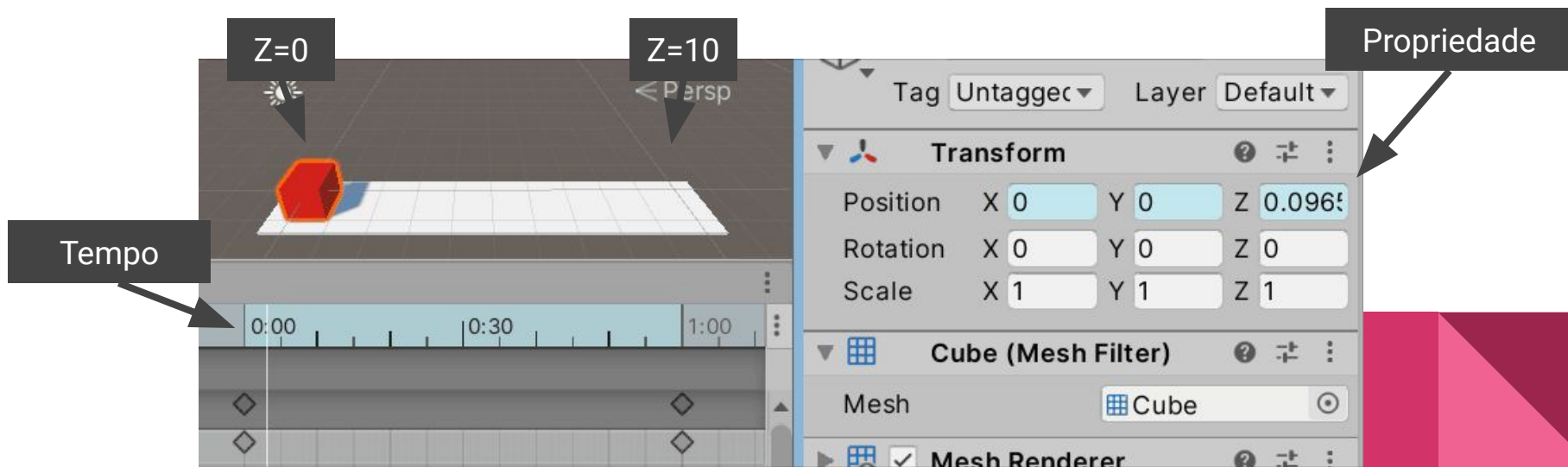
# Animação

Uma animação são estados (propriedades, parâmetros, valores) que mudam ao longo do tempo. Por exemplo, valores de posição e rotação de um determinado Transform (mãos, pés, cabeça, etc)



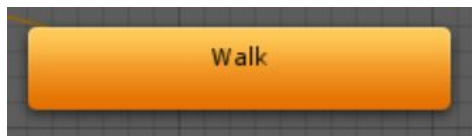
# Animação : Propriedades

No exemplo abaixo, vemos a propriedade Z de um cubo mudando de 0 até 10 ao longo de 1 segundo



# Animation Clip

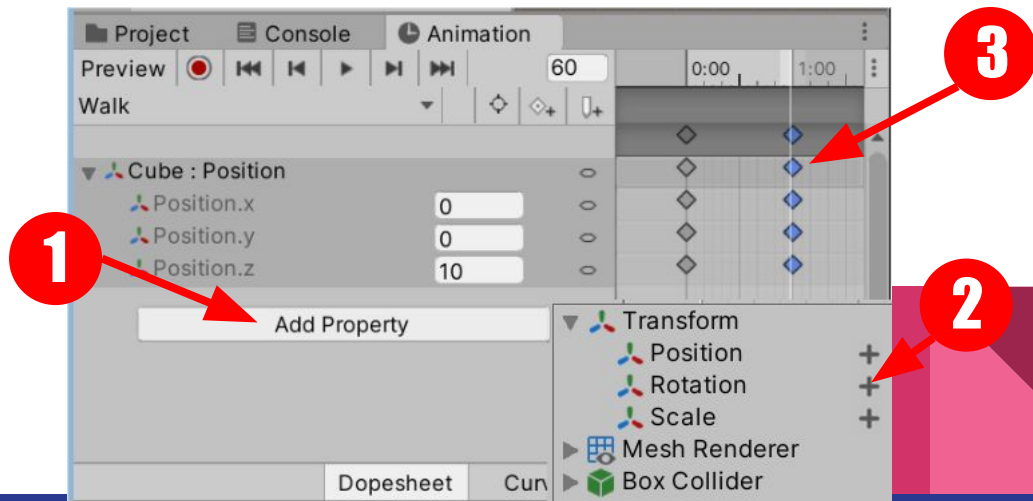
Na Unity, a mudança de propriedades ao longo de um tempo é armazenada em um Asset chamado “Animation Clip”. Ele é visualizado na aba “Animation” que é composta por dois campos: Propriedades e Timeline. Um clipe é uma coleção de propriedades de um objeto, com valores x tempo.



# Animation Clip - Properties

Na janela Animation, ao clicar em “Add Property” (1) é possível selecionar as propriedades dos componentes passíveis de animação (2). Os valores são armazenados ao longo da Timeline em “Keyframes” (3) representados por losangos

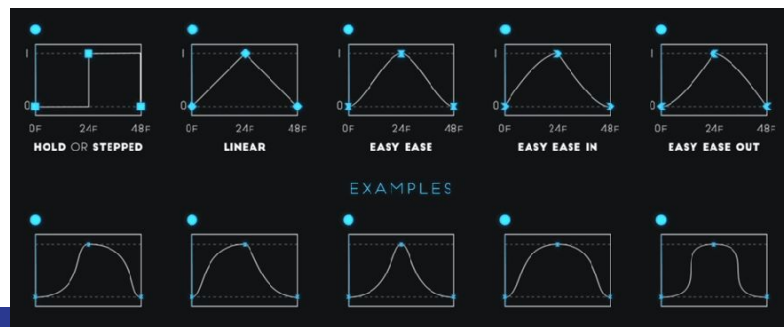
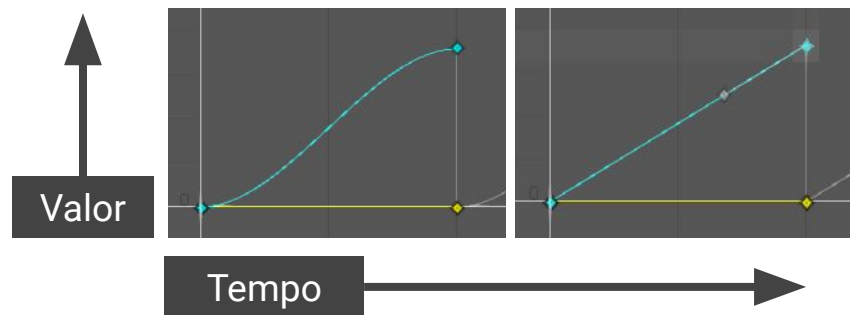
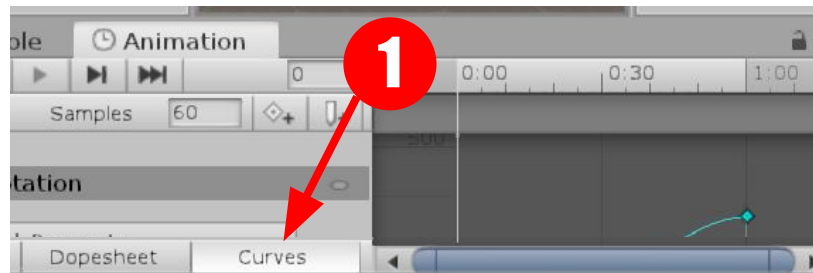
Note que não é preciso declarar os valores em todos os frames da timeline, apenas os principais valores. O software **interpola** os valores entre cada keyframe de modo automático.



# Curva de Interpolação

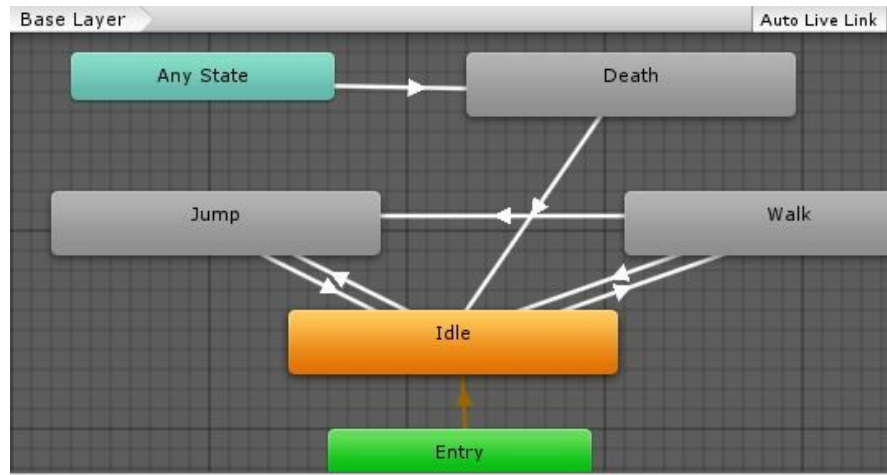
A interpolação entre dois KeyFrames é representada por uma curva e pode ser visualizada ao clicar em “Curves” (1)

Trata-se de um gráfico **Valor x Tempo**, e é um princípio básico da animação, para descrever diferentes modos de interpolação.



# Animator Controller

O animator é um componente que armazena uma ou mais animações. Ele também é responsável por gerenciar toda lógica de troca de animações.





# Atividade: Asset Ventilador



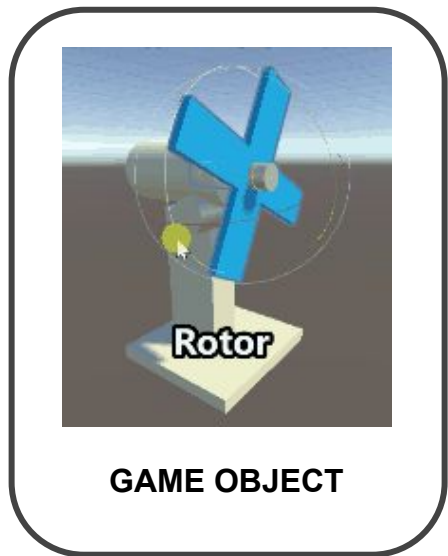
**PLANEJANDO ANIMAÇÕES:**

ESTADOS, MOVIMENTOS,

MUDANÇAS DE PROPRIEDADES

Dúvidas? --> Aula 3 e 5

# Game Object > Animation Clips



(Rotor)

BAIXA

MÉDIA

ALTA

(OSCILAÇÃO)

ATIVADA

DESATIVADA

DESLIGADO

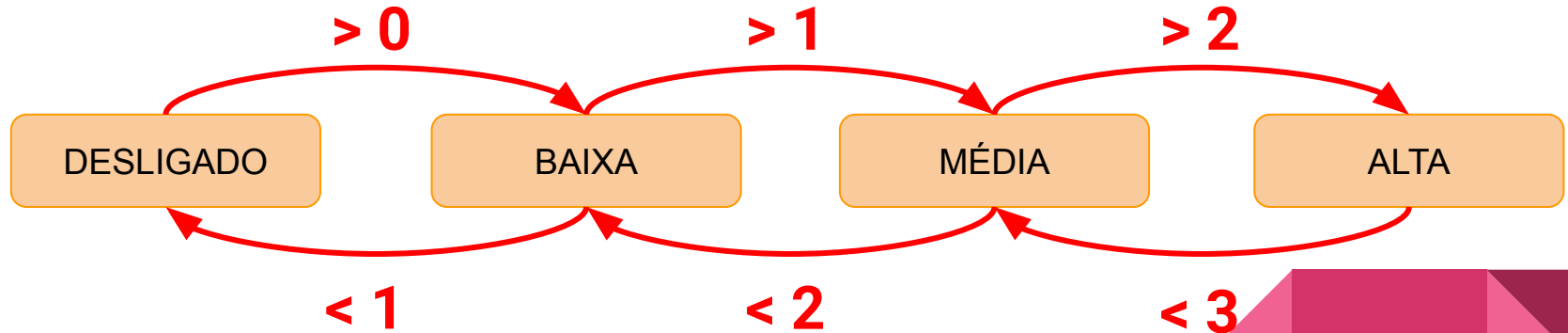
Movimento controlado  
pelo usuário  
(Circular Drive)

Movimentos  
pré-determinados  
(Animator Controller)

INCLINAÇÃO  
(tilt)

# Animator Controller e Lógica de Transição

O parâmetro "FAN CONTROL" será um valor inteiro de 0 a 3 e irá definir a velocidade do rotor



# Criando os Assets

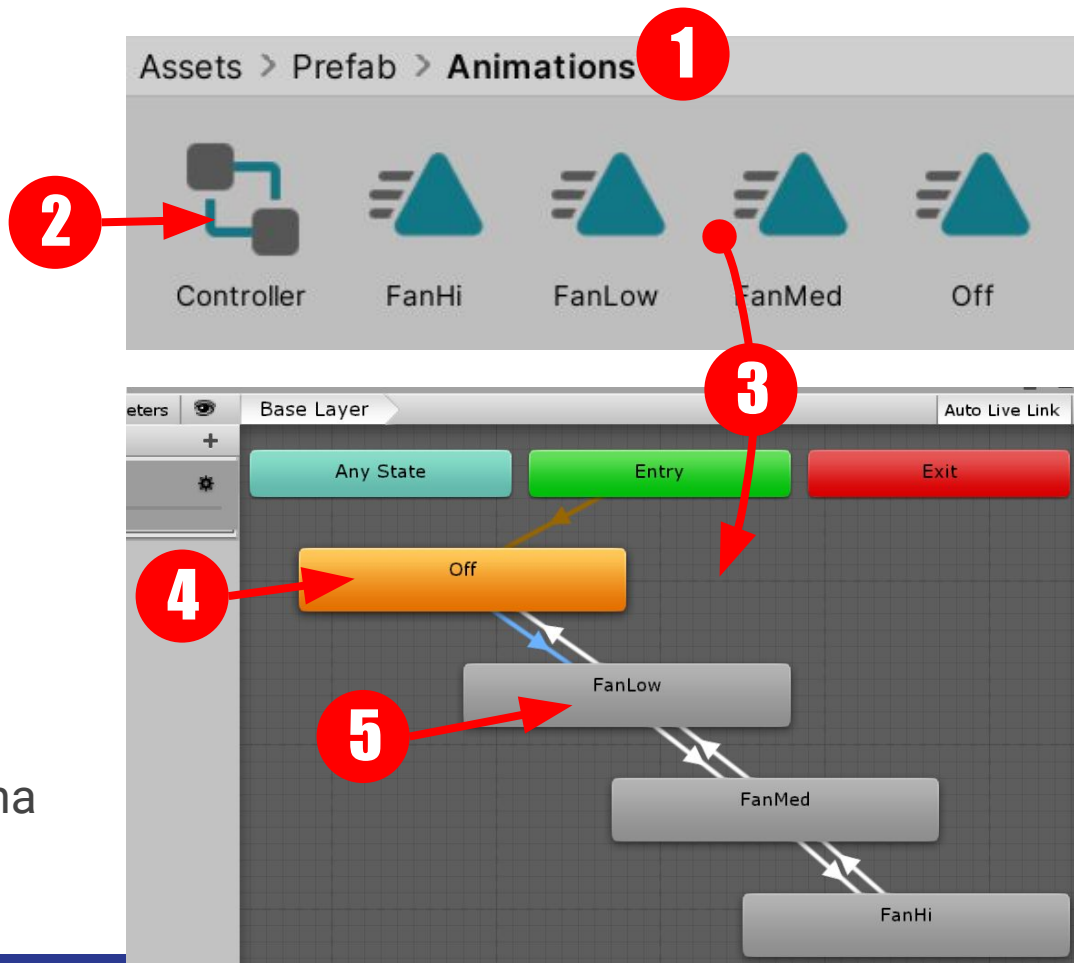
Criar os assets no Project (1)

Clicar 2x em Controller (2)

Arrastar os Animation Clips para a janela Animator Controller (3)

RMB > Make Transition (4)

Clique em outro Clip para criar a transição (5) representada por uma setinha



# Transições e Parâmetros

Clicar na aba Parameters (1)

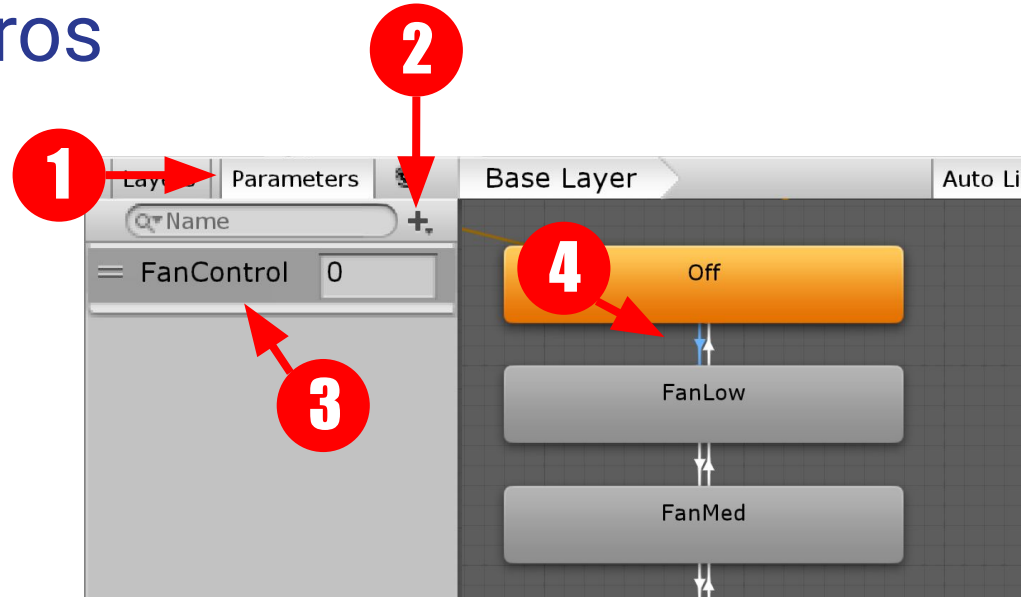
Clicar + e escolher INT (2)

Nomear "FanControl" (3)

Selecionar a transição (4)

Na Janela Inspector: em Conditions,  
Clicar "+" (5)

Configurar Todas Condições (6)  
segundo a lógica descrita no slide 10

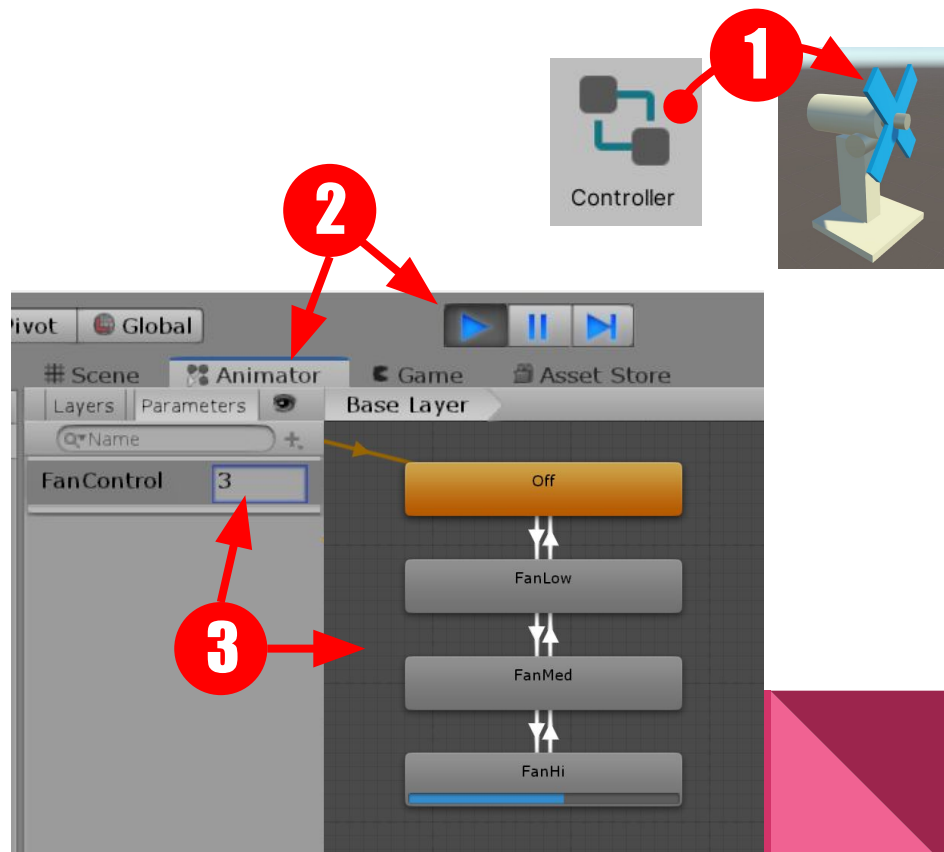


# Testes de Transição

Adicionar Componente Animator Controller no objeto ventilador (1)

Clicar em Play, abrir aba Animator e selecionar o objeto ventilador (2)

Alterar a variável FanControl e observar as transições (3)



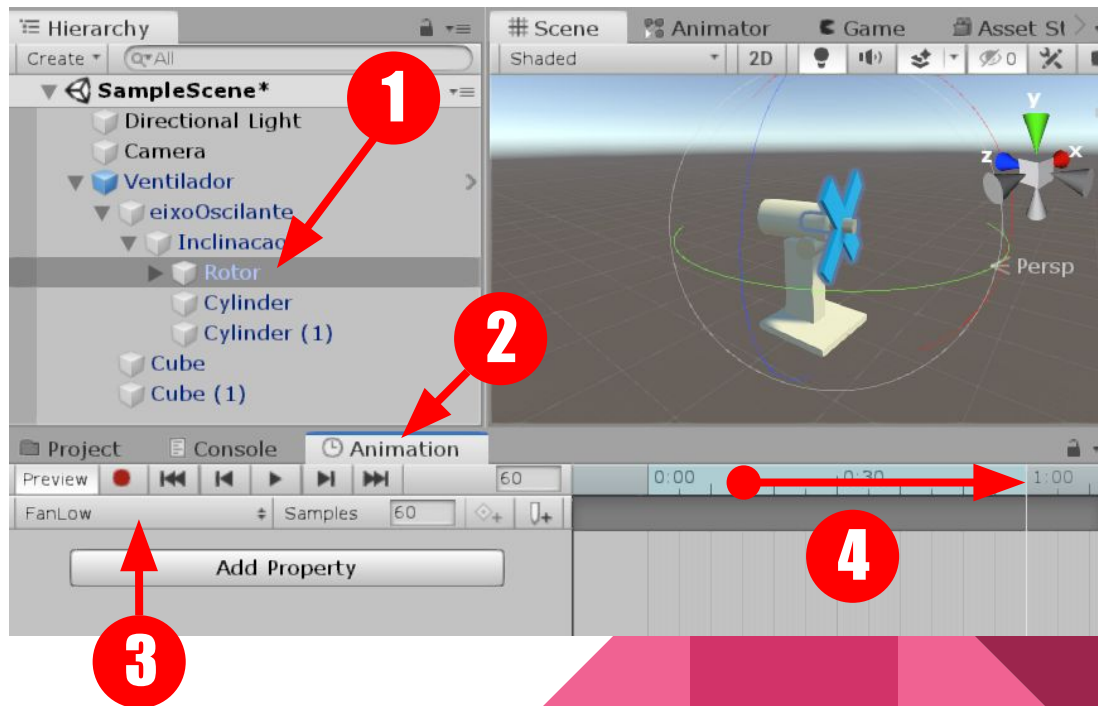
# Animação Rotor 1/2

Selecionar o objeto que rotaciona hélices (1)

Abrir aba animation (2)

Selecionar Clip da Lista (3)

Clicar na Timeline e posicionar agulha (linha branca) no tempo 1:00s (4)



# Animação Rotor 2/2

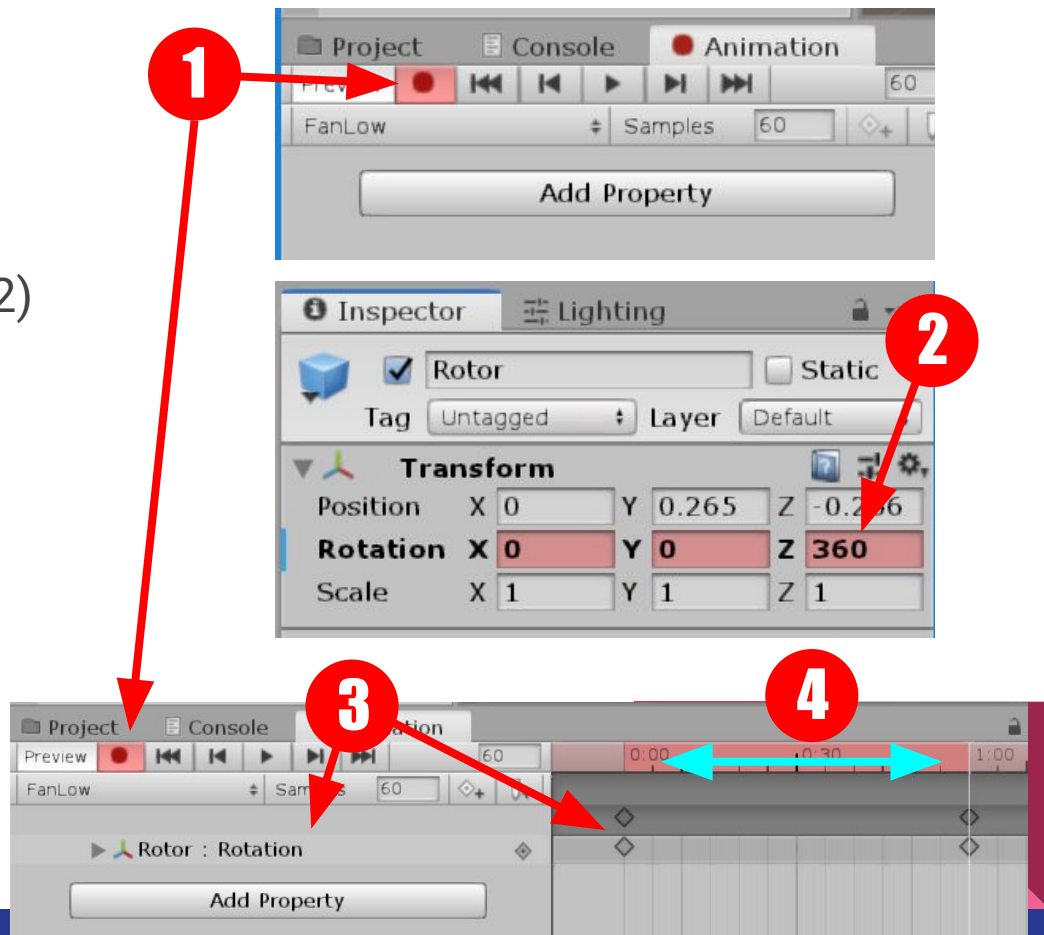
Clique em REC (1)

Altere a rotação do objeto para 360 (2)

Observe a criação dos frames e a inclusão da propriedade animada (3)

Arraste a agulha pela Timeline e observe que o movimento do objeto é interpolado entre os frames (4)

Desative REC ao terminar



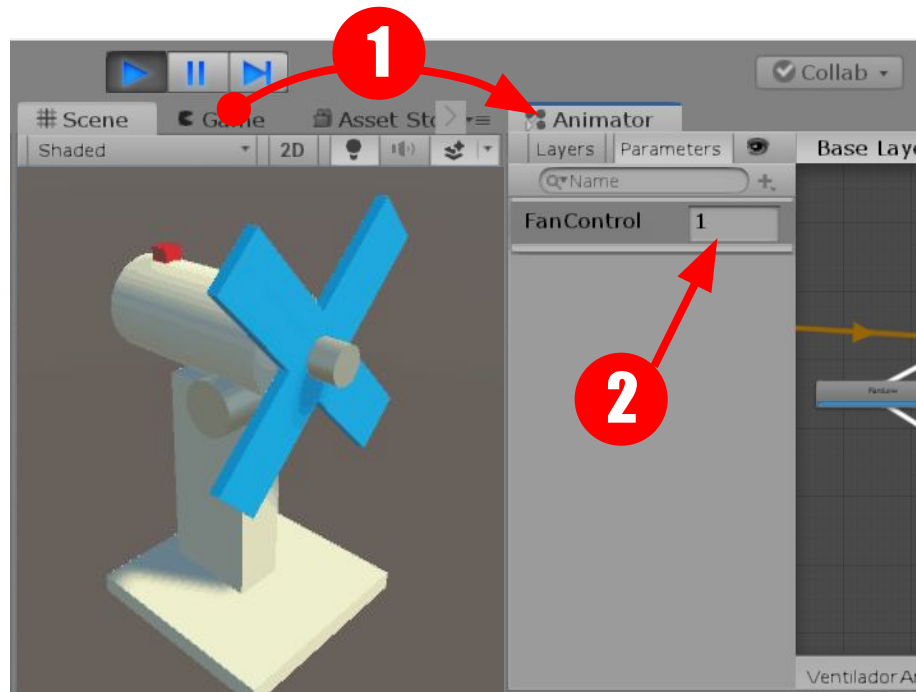


# Teste Animação

Arraste a aba Animator para o lado da Scene (1) e Clique em Play

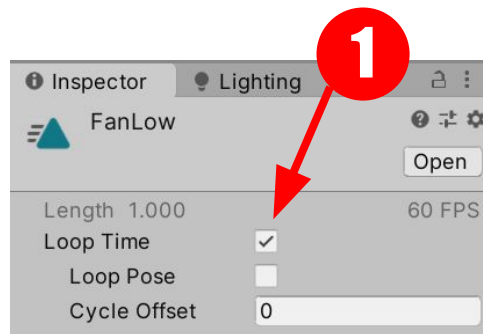
Altere o parâmetro FanControl (2)

Observe que a hélice roda uma única vez e pára. Além disso, acelera e desacelera.

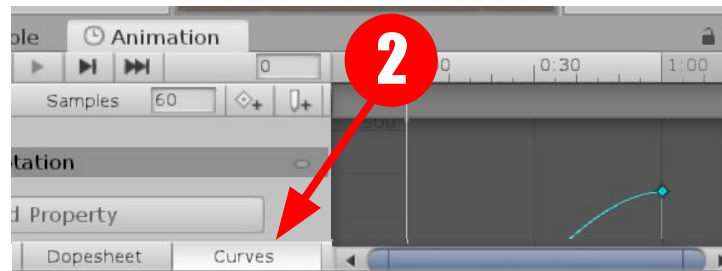


# Loop e Curvas de Interpolação

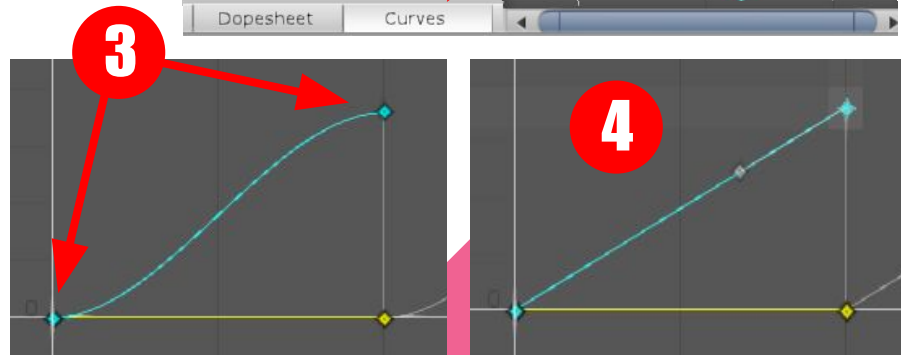
Em Projeto, selecione o clipe e no Inspector, marque opção Loop Time (1)



Em Animation, clique em Curves (2)

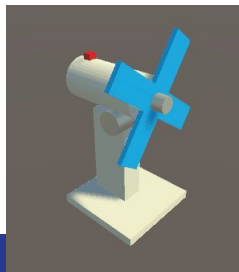


RMB nas pontas da curva e selecione a opção AUTO (3)



Observe a curva constante (4) - sem aceleração e desaceleração

Teste novamente a animação



# Animações Simultâneas

Um ventilador pode ligar ou desligar o modo oscilação sem interromper o funcionamento das hélices. São portanto, **movimentos simultâneos porém independentes**.

Na Unity, resolvemos isso com **“Animation Layers”**.

É um modo que o Animator Controller apresenta para “mixar” diferentes redes de animações. É a mesma lógica que permite que um personagem execute simultaneamente duas animações distintas, como correr e recarregar arma, por exemplo.



# Animation Layers

Para criar um animation Layer, selecione o objeto onde está o Animator.

Em Animator, aba Layers (1), clique no sinal + e nomear como "Oscilacao" (2)

Clicar na engrenagem (3) e setar Weight = 1

Em Project, criar um Clip chamado Oscilacao, marcar Loop Time e arrastar para Animator (4)



# Animation Clip : Oscilação

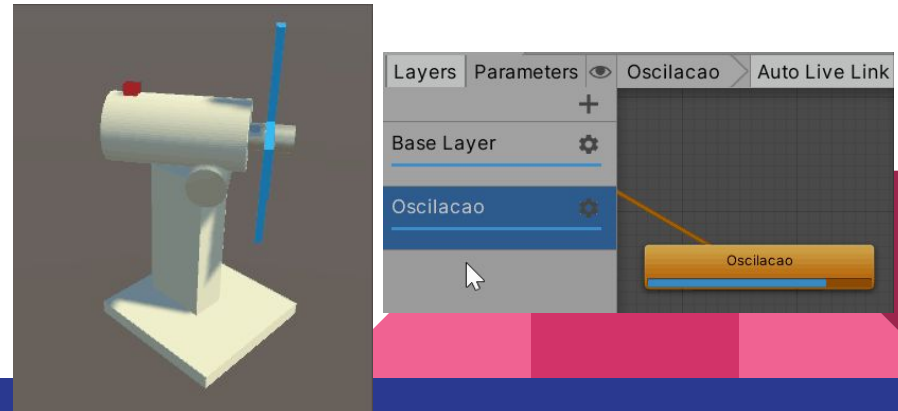
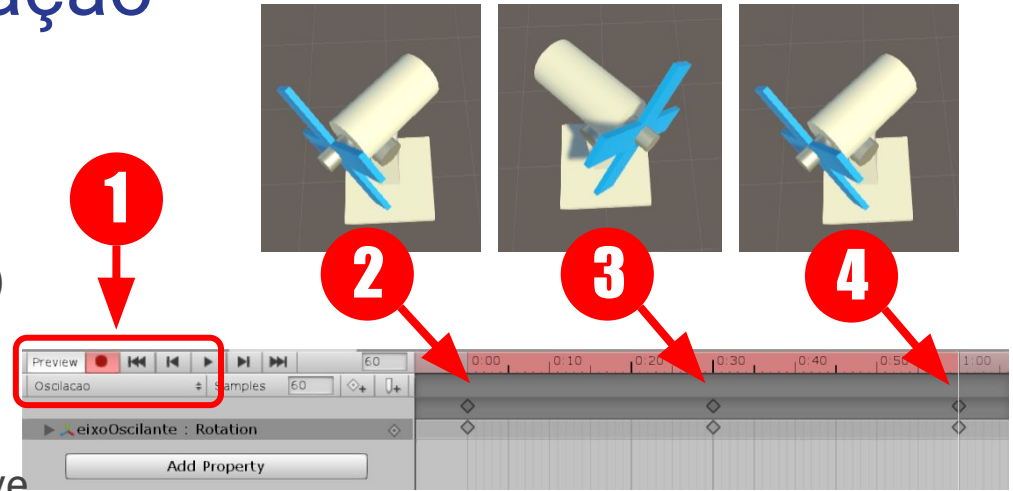
Em animation, selecione Clip Oscilacao, pressione REC (1)

No tempo 0:00 rotacione o rotor a 45° (2)

No tempo 0:30 rotacione -45° (3)

No tempo 1:00 retorne a 45° (4) e desative REC

Teste e observe que as animações ocorrem simultaneamente nos diferentes layers. Lembre-se que o parâmetro FanControl tem que ser maior que 1 para o rotor girar.

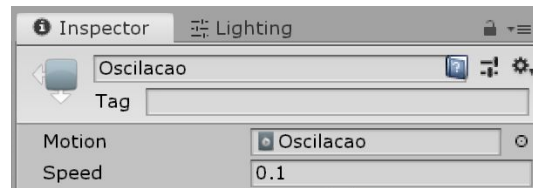


# Parâmetro x Velocidade

Outra característica da função oscilação de um ventilador: Ela sempre prossegue o movimento a partir do ponto onde foi interrompida (não retorna ao ponto inicial quando desligado)

Para “pausar” um animation clip podemos recorrer ao parâmetro Speed. Ele define a velocidade da animação em execução. Assim, Speed 0 pausa, Speed 0.5 executa em 50%, Speed 1.0 em 100%, Speed 2.0 em 200%, etc...

O Animator Controller também fornece um modo muito fácil de alterar o Speed dinamicamente, com o uso de um **multiplicador**. Assim, **posso reduzir ou aumentar a velocidade de um clipe bastando alterar seu multiplicador**.



# Usando Speed Multiplier

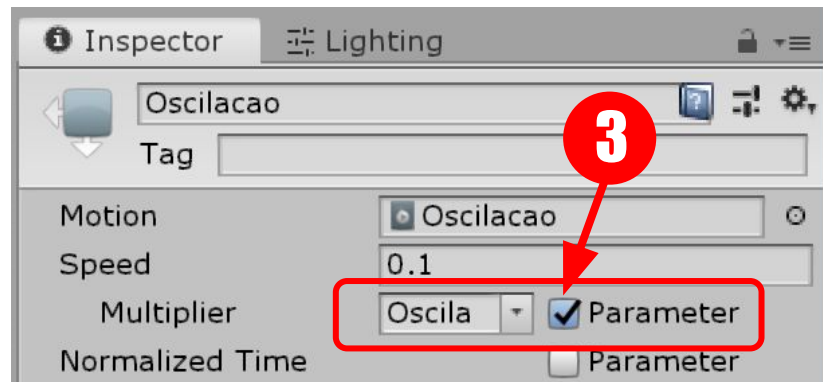
Criar parametro tipo Float e nomear como “Oscila” (1). Este será a nossa variável de controle.

Selecionar a animação Oscilacao (2)

Em Inspector: Speed>Multiplier

Marcar opção Parameter e selecionar Oscila (3)

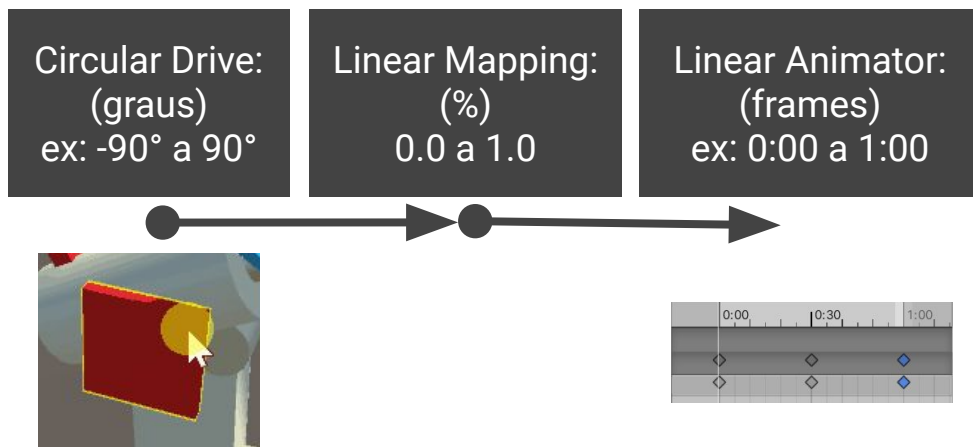
Teste: altere o parametro Oscila de 0.0 para 1.0 e observe o resultado



# Controle Interativo

Para o controle de inclinação, vamos utilizar o movimento circular de um objeto para determinar a posição de um clipe de animação.

Para isso, converteremos valores de uma extensão de rotação em uma unidade genérica (porcentagem) e então convertremos de volta em uma extensão de animação.



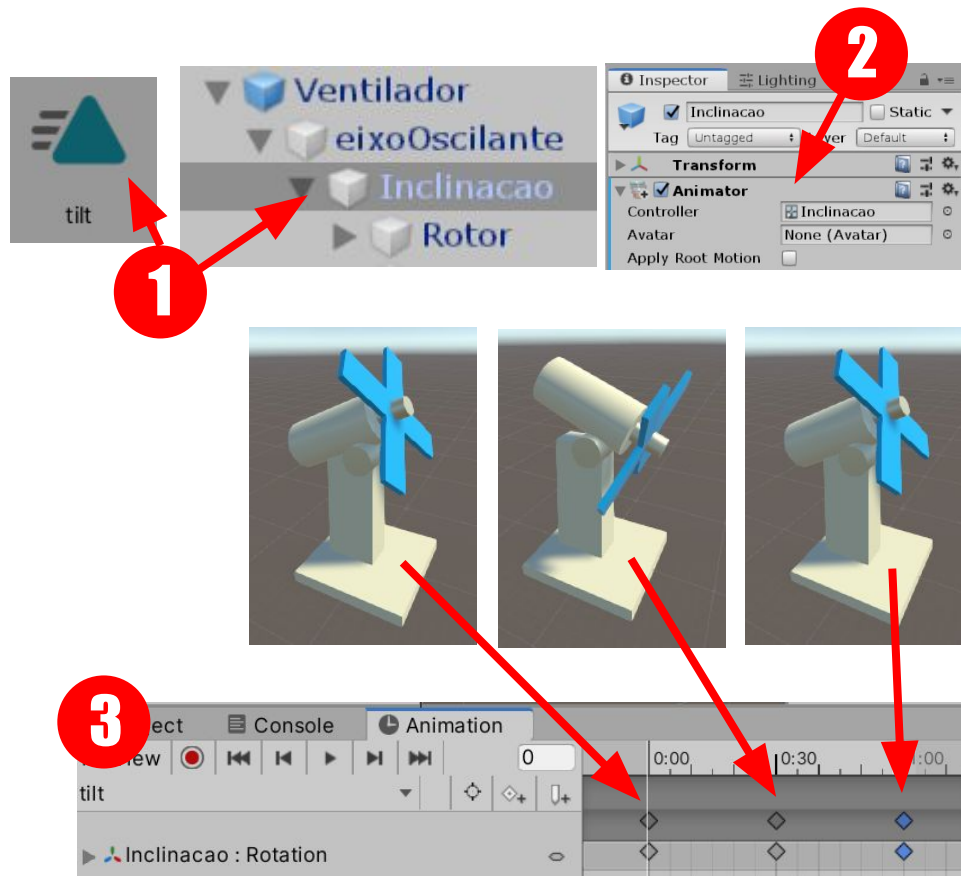


# Animation : Inclinação

Crie o Clip “Tilt” e arraste para o objeto que faz a inclinação do rotor (1)

Note que um novo Animator é criado automaticamente (2)

Em Animation, faça uma animação com o rotor inclinado para cima, para baixo, e novamente para cima (3)



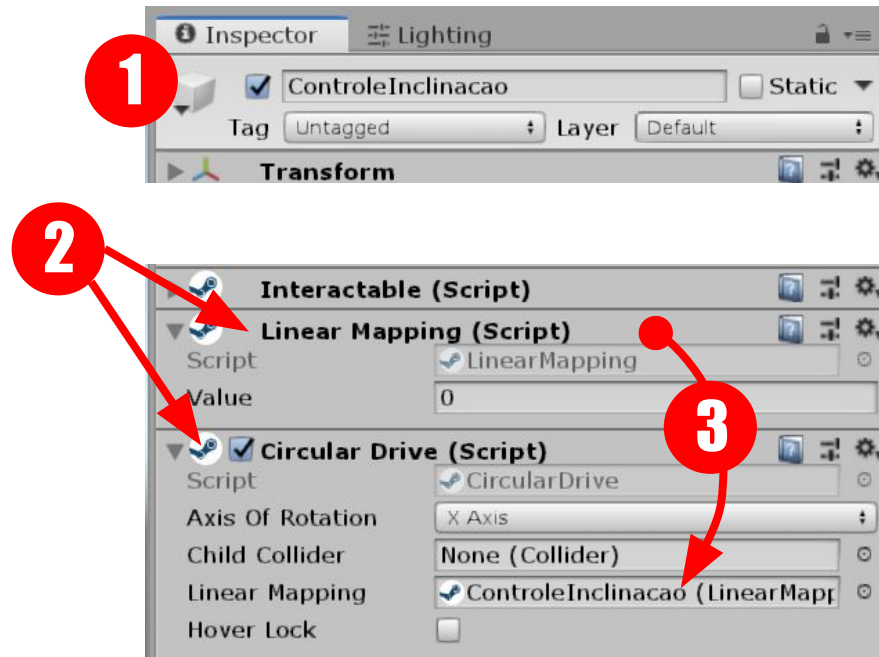
# Controle Interativo 1/2

Vamos utilizar o controle de rosca (com circular drive) feito na aula anterior (1).

Adicione o Componente “LinearMapping” (2)

Arrastar componente LinearMapping para campo “LinearMapping” (3) do Circular Drive.

Com o LinearMapping podemos converter um determinado movimento do Drive (ex valor de rotação) em um valor percentual (de 0.0 a 1.0)



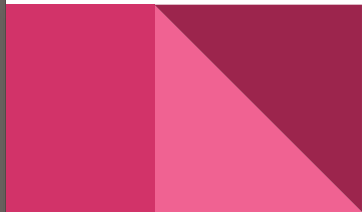
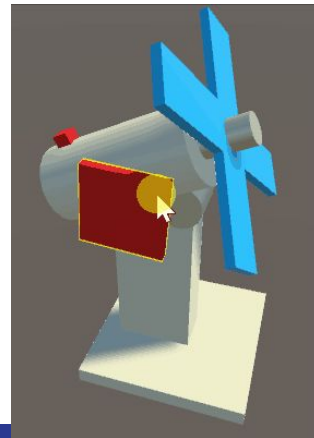
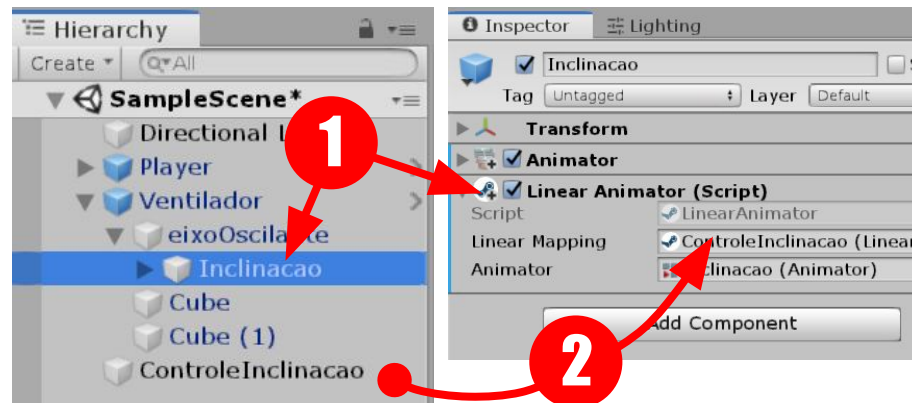
# Controle Interativo 2/2

Selecione objeto Inclinação e adicionar componente “LinearAnimator” (1)

Arrastar objeto ControleInclinacao para campo LinearMapping (2)

Testar: Play e Girar o controle de inclinação

Com o “LinearAnimator” convertemos o valor percentual do “LinearMapping” em uma posição (frame) de um Animation Clip



# Recapitulando...

Essa aula foi densa !

Animações são usadas para descrever qualquer tipo de mudança de propriedade ao longo do tempo, não apenas movimento. Elas podem ser controladas tanto via Animator Controller, quanto via interação direta.

Já vimos como a simulação física pode enriquecer o realismo. A utilização das curvas de animação é uma forma poderosa de criar movimentos complexos.

Um clipe de animação é um asset contendo uma curva matemática que rege propriedade(s) de objeto(s). Podem ser criados usando a funcionalidade de gravação (REC) e ajustando a curva na aba Animation.

# Recapitulando...

O Animator Controller é um componente que gerencia a troca entre animações por meio de fluxos e variáveis determinadas pelo usuário. Determinamos as transições entre animações usando fluxos. “FanControl” e “Oscila” foram variáveis criadas por nós para controlar nossas animações. Já o Speed é um parâmetro interno do Animator.

Mais uma vez temos um exemplo de programação visual. Portanto é importante refletir o que foi feito e a relação entre as diversas entidades, pois vamos começar na próxima aula a ‘brincar’ com código.



# Recapitulando...

Por fim usamos objetos interativos criados em aulas anteriores para controlar nossas animações. O componente Circular Drive permitiu criar uma interação rotativa em um objeto A. Ao adicionar o componente Linear Mapping no objeto e vinculá-lo ao Circular Drive, ele passou a armazenar um valor referente ao mapeamento do movimento rotativo (de 0 a 1).

Já o Linear Animator foi colocado em um objeto B para controlar a animação de inclinação, vinculando o valor do mapeamento linear (da rotação) do objeto A. Portanto, os vínculos entre objetos são ações fundamentais na criação de lógicas e mecânicas interativas.

