

The logo for Oracle Academy. The word "ORACLE" is in a bold, orange, sans-serif font. Below it, the word "Academy" is in a smaller, dark gray, sans-serif font. The entire logo is centered on a light gray background, which is framed by dark gray horizontal bars at the top and bottom.

ORACLE

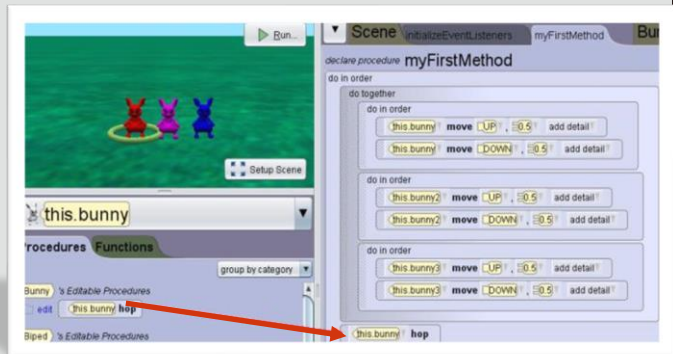
Academy

Java Fundamentals

2-5

Declarar Procedimentos

ORACLE
Academy



Copyright © 2022, Oracle e/ou suas empresas afiliadas. Oracle, Java e MySQL são marcas comerciais registradas da Oracle Corporation e/ou de suas empresas afiliadas. Outros nomes podem ser marcas comerciais de seus respectivos proprietários.

Objetivos

- Esta lição abrange os seguintes objetivos:
 - Comparar e definir uma animação e um cenário
 - Criar um storyboard
 - Criar um fluxograma de um storyboard
 - Descrever a herança e como as características são passadas de superclasses para subclasses
 - Descrever quando implementar uma abstração procedural
 - Demonstrar como criar um procedimento
 - Identificar e usar técnicas de abstração para simplificar o desenvolvimento da animação



Movimento do Objeto

- Os animadores profissionais começam o processo desenvolvendo um cenário — ou uma história — que dê uma finalidade à animação
- Exemplos:
 - Uma história que representa um conflito e uma solução
 - Uma lição que ensina um conceito matemático
 - Um processo para simulação ou demonstração
 - Um jogo com fins de entretenimento ou treinamento

Seu cenário é a ideia principal por trás da animação.

Exemplos de Cenário e Animação

- A definição do cenário e da animação que representa-o é a primeira etapa no processo de programação da animação

Tipo de Cenário	Cenário	Animação
História	Um gato precisa de ajuda para descer de uma árvore	Um bombeiro sobe na árvore e salva o gato
Lição	A memorização dos símbolos químicos é difícil	Um jogo cronometrado faz a correspondência entre os símbolos químicos e as respectivas definições
Processo	Um carro com pneu furado	Uma demonstração mostra como trocar um pneu em um carro virtual
Jogo	Um avião deve evitar a colisão com objetos em sua rota enquanto está voando	Um jogo interativo mostra manobras feitas por um avião ao redor de objetos no céu

O que é um Storyboard?

- Um storyboard identifica as especificações de projeto do cenário de animação:
 - como os objetos aparecem, movem-se, falam, interagem e assim por diante
- Quando um cenário estiver definido, você poderá começar a desenvolver o storyboard da animação
- Há dois tipos de storyboards que são usados para planejar uma animação:
 - Visual: uma série de imagens ilustradas que representam as cenas principais da animação
 - Textual: uma lista detalhada e ordenada de ações que cada objeto executa em cada cena da animação

Os storyboards são usados para definir os estágios pelos quais uma animação passará.

Formatos de Storyboard

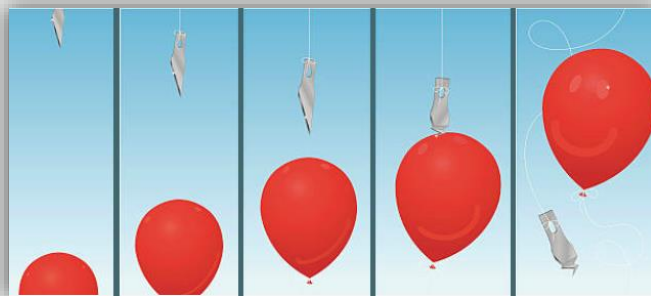
- Os storyboards são criados nos seguintes formatos:
 - Desenhados com papel e lápis
 - Criados usando ferramentas digitais, como um programa de processamento de texto, um programa de desenho ou uma apresentação
 - Criados usando comentários no Code editor do Alice 3



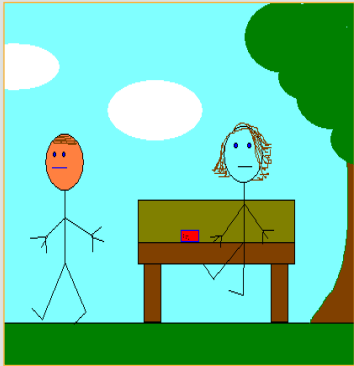
Não importa como o storyboard é criado. Se ele informar detalhes do que está acontecendo a cada estágio para que você possa codificá-lo, então o storyboard estará cumprindo seu papel.

Storyboards Visuais

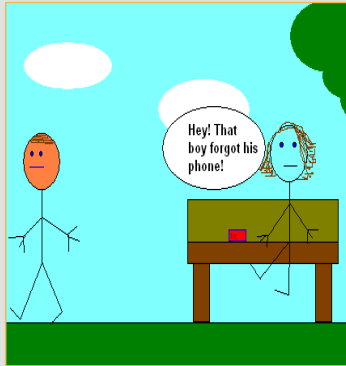
- Um storyboard visual ajuda o leitor a compreender:
 - Os componentes de uma cena
 - Como a cena inicial será configurada
 - Os objetos móveis e estáticos em uma cena
 - As ações que ocorrerão
 - As interações com o usuário que ocorrerão durante a execução da animação



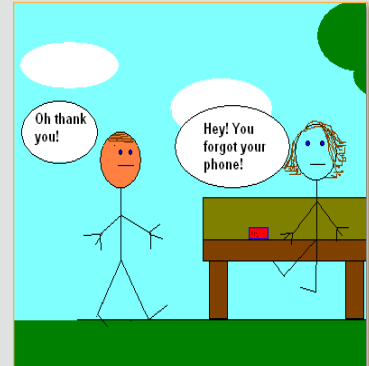
Exemplo de um Storyboard Visual



Um menino e uma menina estão sentados no banco de um parque. O menino se afasta e esquece o telefone.



A menina observa o celular. E pensa: "Ih! Aquele menino esqueceu o telefone"



A menina grita: "Ei! Você esqueceu seu telefone!" O menino se vira e retorna até o banco. Ele diz: "Poxa! Muito Obrigada!"

ORACLE
Academy

JF 2-5
Declarar Procedimentos

Copyright © 2022, Oracle e/ou suas empresas afiliadas. Oracle, Java e MySQL são marcas comerciais registradas da Oracle Corporation e/ou de suas empresas afiliadas. Outros nomes podem ser marcas comerciais de seus respectivos proprietários.

9

Um storyboard visual não é um teste da sua capacidade de desenho. A questão aqui não é ser um bom artista. Se você é capaz de desenhar um boneco palito, então é capaz de desenhar um storyboard visual.

Storyboards Textuais

- Um storyboard textual ajuda o leitor a compreender as ações que ocorrerão durante a animação
- Os objetos móveis e estáticos podem ser facilmente identificados dentro das instruções de ação, mas talvez seja necessária uma descrição mais detalhada se também houver mais programadores envolvidos na implementação de qualquer cena

Um algoritmo é uma lista de ações para executar uma tarefa ou solucionar um problema. Na computação, um storyboard textual é um algoritmo.

A parte importante na criação de um storyboard textual é assegurar que você simplificou todas as tarefas para partes de componentes individuais. Para codificação no Alice 3, um storyboard que descrevesse "ficar de pé" não seria refinado o suficiente. Você precisa descrever as ações de todas as articulações que estão envolvidas no movimento de ficar de pé.

Exemplo 1 de Storyboard Textual

- Programe as ações a seguir em ordem:
 1. Um menino e uma menina estão sentados no banco de um parque
 2. O menino se levanta e sai andando, deixando o celular no banco do parque
 3. A menina vira-se para olhar para o telefone
 4. A menina pensa: **"Ih! Aquele menino esqueceu o telefone!"**
 5. A menina grita: **"Ei! Você esqueceu seu telefone!"**
 6. O menino para e volta
 7. O menino volta até o banco do parque e diz: **"Poxa! Muito obrigado!"**

Este é um storyboard de nível superior. Você precisaria refiná-lo ainda mais para tornar sua animação real no Alice 3.

Exemplo 2 de Storyboard Textual

- Este exemplo mostra como você pode desenvolver o storyboard inserindo primeiro comentários no Code editor do programa
- Depois, você pode começar a desenvolver a animação diretamente no storyboard



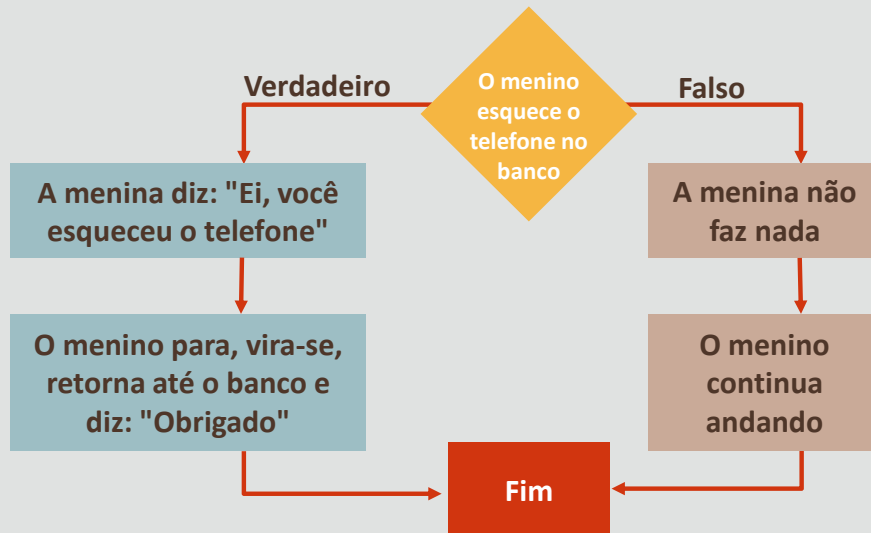
Esta é uma técnica muito útil para garantir que todas as partes do cenário sejam codificadas.

Componentes do Storyboard Textual

Componente	Definição	Exemplos
Cena	O local (ou "mundo" no Alice 3) em que a história acontece	Parque, biblioteca, escola, casa
Objetos	Personagens móveis ou estáticos que você programa para se moverem e realizarem ações	Animais, carros, pessoas, árvores
Ações	Instruções sobre como cada objeto deve agir na cena	Andar dois metros, virar para a esquerda e falar "Oi!"
Interações com o Usuário	Maneiras como o usuário que está vendo a animação pode manipular objetos na animação	Comandos do teclado ou cliques do mouse para fazerem com que os objetos se movam
Especificações de Projeto	Qual deve ser a aparência dos objetos e do cenário na animação	O tamanho, a posição, a localização, a cor

Criando um Fluxograma de um Storyboard

- Criar um fluxograma de um storyboard ajuda a organizar o fluxo das ações e condições da animação



Um fluxograma permite que você planeje o fluxo do código mostrando os processos e o que acontece como resultado de cada decisão. Este é um exemplo muito simples. Um fluxograma pode tornar-se bastante complexo à medida que os programas ficam maiores.

Usando Comentários para Organizar o Programa

- Os storyboards textuais podem ser usados para gerar instruções de comentários do programa e organizar o desenvolvimento do programa
- Os storyboards também podem ajudar os programadores a identificarem ações repetitivas de um objeto e ações idênticas que podem ser executadas por vários objetos

Características herdadas

- Vamos analisar como um Dálmata herda as respectivas características:
 - As características da classe Cão (a classe pai ou "superclasse") inclui quatro patas, dois olhos, pelo e a capacidade de latir
 - As características da classe da raça Dálmata (a classe filha ou "subclasse" que é um subconjunto da classe cão) incluem pelo branco, marcas pretas e outras características



Classe Cão (Superclasse)

Classe Dálmata
(Subclasse)

Herança da Classe

- Assim como acontece com os animais no mundo real, os objetos no mundo da programação herdam as características de sua classe, inclusive todos os métodos (procedimentos e funções) da classe
- Por exemplo, todos os objetos dentro de uma classe Quadrúpede no Alice herdam as características de um quadrúpede de quatro patas, uma cabeça, um corpo, etc
- Dentro da superclasse Quadrúpede, existem subclasses para cães, gatos, lobos, leões, vacas, etc
- Cada subclasse adiciona características que identificam mais especificamente os objetos dentro dela

As subclasses podem herdar de sua superclasse, mas o oposto não pode ocorrer. Esse é um processo unilateral. Todos os tipos diferentes de subclasse terão acesso ao código que existe no nível da superclasse. Cada tipo específico de cão tem acesso ao código que foi gravado na classe cão. Isso significa que todas as características que são genéricas dos cães seriam codificadas no nível cão, enquanto as características que tornam determinado cão único seriam codificadas no nível da subclasse.

Identificando Comportamentos Repetitivos no Storyboard

- Entender que as subclasses pertencem a superclasses ajuda você, programador, a identificar comportamentos repetitivos no storyboard
- Por exemplo:
 - Se você planeja programar um cão, um gato ou um leão para andar, é necessário programar o comportamento repetitivo de andar na superclasse ou no nível quadrúpede
 - Assim, todas as subclasses (cão, gato, leão) podem usar a característica herdada de andar e você não precisará programar o comportamento repetitivo de andar para cada objeto individualmente

Com isso, é possível escrever um código bem menor, o que facilita a leitura e a compreensão dos programas.

Herança

- Quando um objeto Dálmata é criado, ele herda procedimentos, funções e propriedades da classe Quadrúpede e da subclasse Dálmata que você pode ver no Code editor

A herança significa que cada objeto da subclasse herda os métodos e as propriedades da respectiva superclasse



Criar Métodos Herdados

- Além dos métodos predefinidos, você pode criar seus próprios métodos e fazer com que sejam exibidos, ou fiquem disponíveis, para qualquer objeto da subclasse
- Os métodos herdados sempre aparecerão no início da lista de métodos predefinidos quando eles são criados

Guia myFirstMethod

- Por padrão, a guia myFirstMethod é exibida quando o Code editor é aberto



Hierarquia de Classes

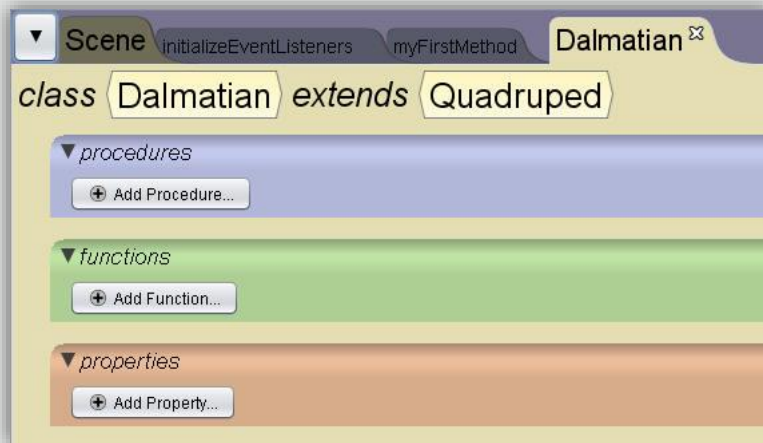
- Clique no menu drop-down da hierarquia de classes à esquerda da guia myFirstMethod (indicado por uma seta que aponta para baixo) para exibir a lista de classes e subclasses na animação



Quanto mais alta for a lista de classes com que você puder codificar seu procedimento, melhor. Isso significa que mais objetos terão acesso a ele.

Exibir os Métodos de Classe

- Selecione uma super ou subclasse para exibir os procedimentos, as funções e as propriedades definidos para a classe selecionada



Abstração Procedural

- Verifique o código existente ou o storyboard textual para identificar e planejar os métodos que precisam ser declarados no programa
- Talvez seja necessário implementar a abstração procedural se um objeto no Alice precisar executar uma ação, mas não houver um procedimento herdado que realize essa ação
- Identifique um comportamento repetitivo e crie um método para ele:
 - Simplifica o código, tornando-o mais fácil de ser lido
 - Permite que muitos objetos de uma classe usem o mesmo método
 - Permite que subclasses de uma superclasse usem o método

Se você tiver vários objetos que estejam realizando as mesmas ações, a criação de um procedimento separado que contenha essa ação diminuirá consideravelmente o tamanho do seu código. Isso também garante que todos os objetos comportem-se da mesma maneira porque é uma parte única de código que está controlando-os.

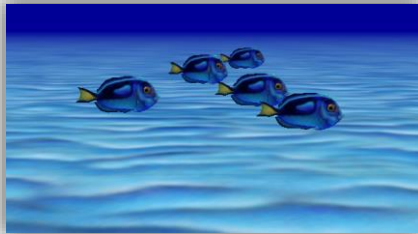
Abstração Procedural Definida

- Um exemplo de abstração procedural é remover uma instrução de programação repetitiva ou longa de `myFirstMethod` e inseri-la em seu próprio método para que vários objetos ou classes tenham mais facilidade de lerem, entenderem ou reutilizarem o código

A abstração procedural é o processo de analisar o código de programação, identificar instruções de programação repetitivas e extrair delas seus próprios métodos tornando, assim, o código mais fácil de ser compreendido e reutilizado

Exemplo 1 de Abstração Procedural

- Um ou mais objetos podem executar os mesmos movimentos repetitivos
- A animação de um peixe nadando requer a criação de vários procedimentos repetitivos que deverão, então, ser repetidos para cada peixe na escola, ocupando muito espaço no myFirstMethod



ORACLE
Academy

JF 2-5
Declarar Procedimentos

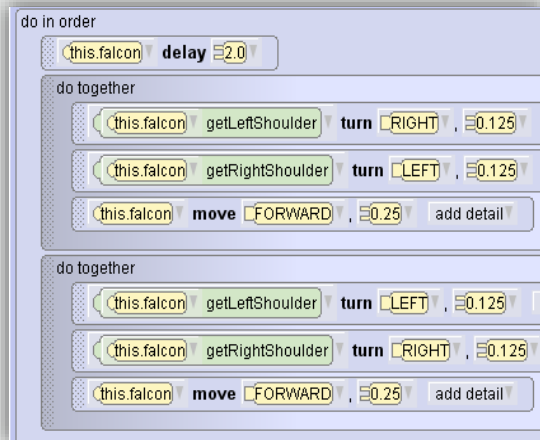


Copyright © 2022, Oracle e/ou suas empresas afiliadas. Oracle, Java e MySQL são marcas comerciais registradas da Oracle Corporation e/ou de suas empresas afiliadas. Outros nomes podem ser marcas comerciais de seus respectivos proprietários.

26

Exemplo 2 de Abstração Procedural

- Às vezes, um procedimento que é necessário para executar uma ação não está disponível por padrão
- Por exemplo, um pássaro precisa voar, mas não existe um procedimento de voar disponível para objetos pássaros



ORACLE
Academy

JF 2-5
Declarar Procedimentos

Copyright © 2022, Oracle e/ou suas empresas afiliadas. Oracle, Java e MySQL são marcas comerciais registradas da Oracle Corporation e/ou de suas empresas afiliadas. Outros nomes podem ser marcas comerciais de seus respectivos proprietários.

27

Antes de realizar a abstração procedural de um movimento como este, é útil criar um storyboard que defina o movimento de todas as articulações do objeto. Você deverá projetar e, em seguida, criar um procedimento que realize um único movimento completo que retorne o objeto à sua posição original. Dessa forma, o código poderá ser chamado várias vezes para obter o resultado desejado.

Quando Ocorre uma Abstração Procedural

- Uma abstração procedural pode ocorrer antes ou depois de as instruções de programação serem criadas
- No entanto, desenvolvendo primeiro o storyboard, o programador conseguirá identificar mais facilmente os procedimentos que serão necessários antes de começar a programação

Exemplos de Quando Declarar um Procedimento

- Declare um procedimento quando:
 - Os movimentos não tiverem um procedimento padrão, como um pássaro voando
 - Os movimentos precisarem ser usados para vários objetos ou classes, como todos os quadrúpedes saltitando para cima e para baixo
 - Movimentos individuais exigirem várias instruções de programação, como uma pessoa movendo partes do corpo para caminhar

Um dos principais benefícios de fazer isso no Alice 3 é que, depois de criar um procedimento complexo, você poderá transferi-lo facilmente de uma animação para outra. Você aprenderá mais sobre esse assunto mais adiante neste curso.

Declarando um Procedimento - Exemplo 1

- O pássaro voa girando os ombros e movendo-se para frente simultaneamente
- Esse movimento repetitivo pode ser extraído em um procedimento próprio de voar

Do together:

- O pássaro vira o ombro direito para trás
- O pássaro vira o ombro esquerdo para trás
- O pássaro se move para frente

Do together:

- O pássaro vira o ombro direito para frente
- O pássaro vira o ombro esquerdo para frente
- O pássaro se move para frente

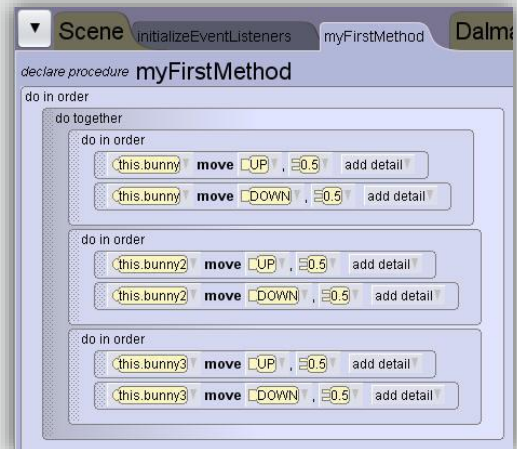


Do together:

- Moscas de pássaros
- O pássaro se move para frente

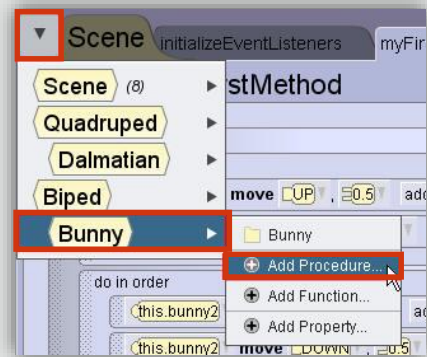
Declarando um Procedimento - Exemplo 2

- Cada coelho move-se para cima e para baixo, a fim de simular um movimento de salto
- Este movimento repetitivo usado por todos os objetos coelho poderia ser extraído em um procedimento próprio de salto



Etapas para Salvar um Procedimento

- Na hierarquia de classes, selecione a classe que deve herdar o procedimento
- Todas as subclasses herdarão, então, o procedimento também (Elas estão recuadas dentro das respectivas superclasses)
- Clique no botão Add Procedure...



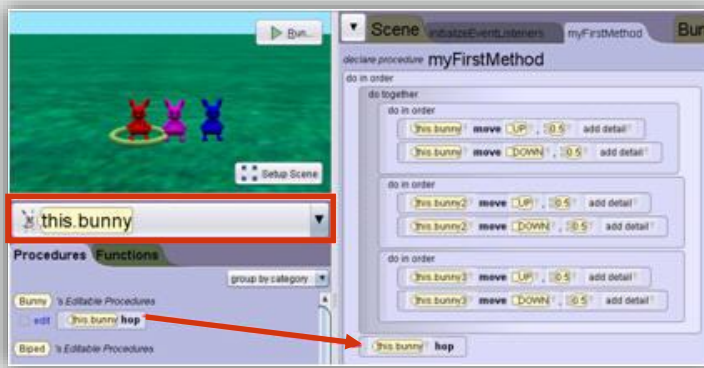
Etapas para Salvar um Procedimento

- Especifique um nome para o procedimento e clique em OK
- Uma nova guia com o nome do procedimento será aberta
- Escreva o código do procedimento
- Adicione o novo procedimento a `myFirstMethod` agora mesmo Assim, quando testar a animação em `myFirstMethod`, você também estará testando a animação no novo procedimento

Quando você atribuir um nome ao procedimento, defina um nome descritivo para ele. Esse nome deve informar o que o procedimento faz, mas sem vinculá-lo a um tipo específico de objeto porque outras subclasses poderão usá-lo.

Etapas para Adicionar um Procedimento à Guia myFirstMethod antes da Programação

- Clique na guia myFirstMethod
- Selecione a instância para a qual você está codificando o procedimento no menu da instância
- Na guia Procedures, localize o procedimento declarado e arraste-o para myFirstMethod



Etapas para Adicionar um Procedimento à Guia myFirstMethod antes da Programação

- Clique na guia na parte superior do Code editor com o nome do procedimento para retornar ao procedimento



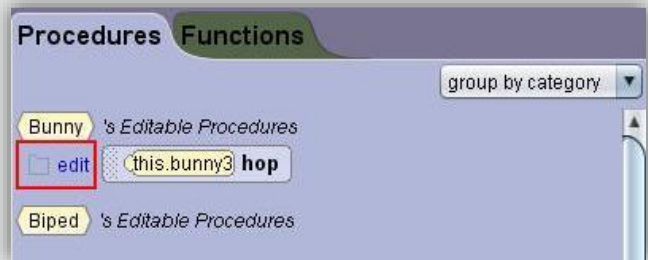
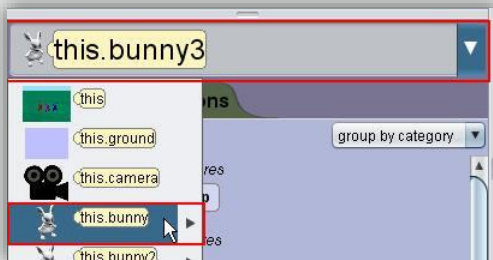
Etapas para Adicionar um Procedimento ao Código

- Selecione a instância do objeto a que deseja adicionar o procedimento declarado
- Encontre o procedimento declarado listado na guia Procedures no painel Methods
- Os procedimentos declarados estão disponíveis na guia Procedure de todos os objetos que herdam esses procedimentos
- Arraste o procedimento declarado até o código



Etapas para Acessar e Editar Procedimentos Declarados

- No menu da instância, selecione a instância em que o procedimento foi declarado
- Clique em Editar à esquerda do nome do procedimento



Etapas para Acessar e Editar Procedimentos Declarados

- Crie ou edite instruções de programação para o procedimento
- Clique no botão Run para testar o procedimento e depurar, caso seja necessário
- Quando terminar a edição do procedimento, clique na guia myFirstMethod para retornar

Identificar Oportunidades para Procedimentos Declarados

- Quando estiver programando, sempre identifique oportunidades para declarar procedimentos usando técnicas de abstração procedural
- Por exemplo, em myFirstMethod, o peixe nada lentamente para cima e para baixo repetidamente
- Como resultado, um movimento "nadar" separado deverá ser declarado

```
declare procedure myFirstMethod
do in order
  this.pajamaFish move FORWARD , 0.5 add detail
  this.pajamaFish move UP , 0.25 add detail
  this.pajamaFish move DOWN , 0.25 add detail
  this.pajamaFish move UP , 0.25 add detail
  this.pajamaFish move DOWN , 0.25 add detail
```



```
declare procedure bob
do in order
  this move FORWARD , 0.5 add detail
  this move UP , 0.25 add detail
  this move DOWN , 0.25 add detail
  this move UP , 0.25 add detail
  this move DOWN , 0.25 add detail
```

Sempre verifique se há um código repetitivo dentro do seu programa. Se você estiver escrevendo o mesmo código duas ou mais vezes, provavelmente poderá abstrair-lo.

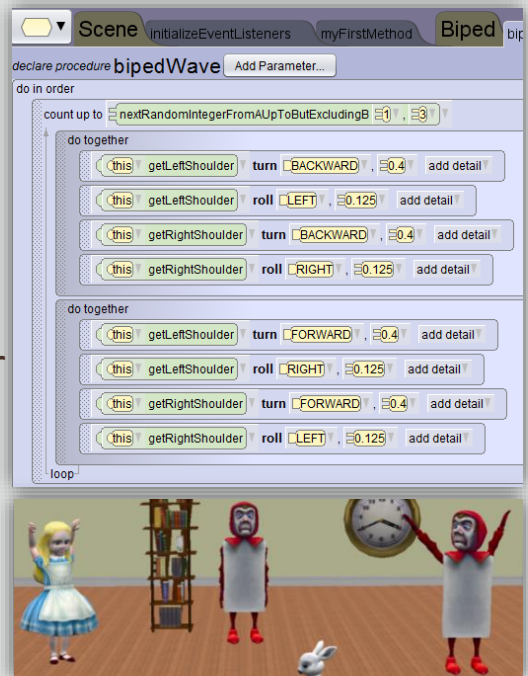
Abstração Procedural e Uso da Área de Transferência

- Depois que tiver criado várias instruções de programação na guia myFirstMethod, você poderá determinar se o código atenderia melhor ao programa se estivesse em um procedimento declarado
- Para economizar tempo, arraste as instruções de programação até o ícone da área de transferência
- Então, depois de criar o procedimento declarado, você poderá arrastar as instruções de programação da área de transferência até o procedimento declarado

Ao arrastar um bloco de código para um procedimento por meio da área de transferência, você precisará alterar o nome do objeto para "this" para que ele possa ser acessado por vários objetos.

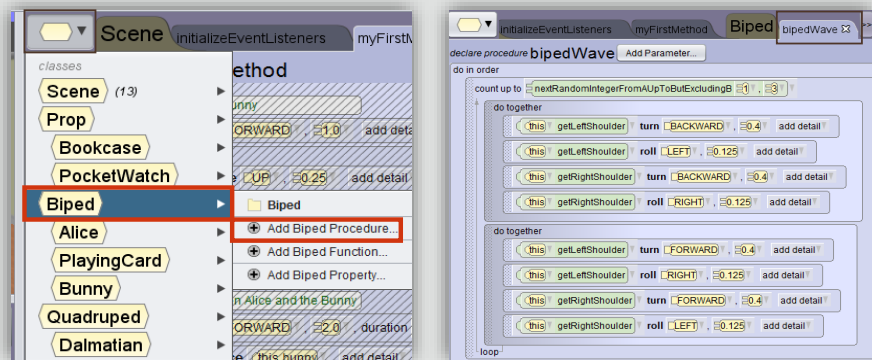
Usar Procedimentos Herdados

- Os procedimentos declarados no nível da superclasse estão disponíveis para outros objetos nesta classe
- Por exemplo, um procedimento "bipedWave" criado para playingCard mexer seus braços no ar poderia ser usado para fazer com que Alice mexesse os braços no ar também



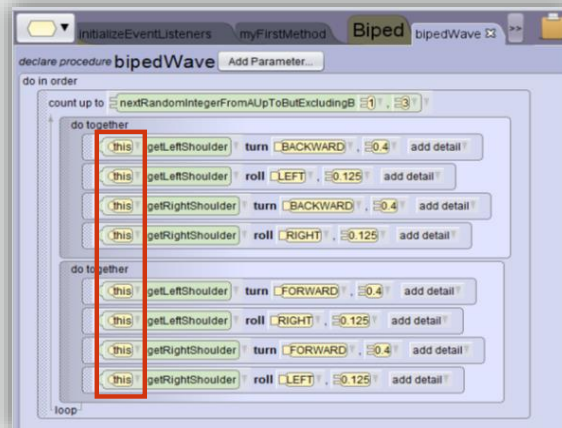
Declarar um Procedimento no Nível da Superclasse

- Declare o procedimento "bipedWave" no nível da superclasse para que playingCards e Alice possam usá-lo
- Qualquer outro bípede que seja adicionado posteriormente poderá ter acesso também a este procedimento



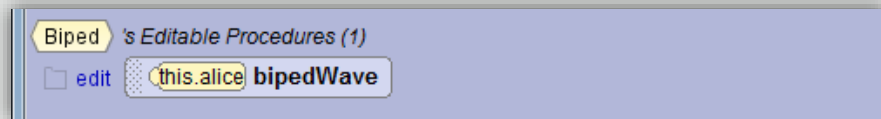
Identificador do Objeto this

- Quando um procedimento declarado é criado, o identificador do objeto "this" é usado para indicar que a instância que está chamando o procedimento é "this"



Exemplo do Identificador do Objeto this

- Se Alice estiver selecionada no menu da instância, o procedimento bipedWave será adicionado a myFirstMethod; "this" no procedimento declarado refere-se a Alice



- Se playingCard estiver selecionada no menu da instância, o procedimento bipedWave será adicionado a myFirstMethod; agora "this" no procedimento declarado refere-se a playingCard
- Na essência, "this" sempre se refere à instância da classe que está chamando o procedimento

Terminologia

- Estes são os principais termos usados nesta lição:
 - Algoritmo
 - Procedimentos declarados
 - Herança
 - Abstração procedural
 - Cenário
 - Storyboard

Resumo

- Nesta lição, você deverá ter aprendido a:
 - Comparar e definir uma animação e um cenário
 - Criar um storyboard
 - Criar um fluxograma de um storyboard
 - Descrever a herança e como as características são passadas de superclasses para subclasses
 - Descrever quando implementar uma abstração procedural
 - Demonstrar como criar um procedimento
 - Identificar e usar técnicas de abstração para simplificar o desenvolvimento da animação



