

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

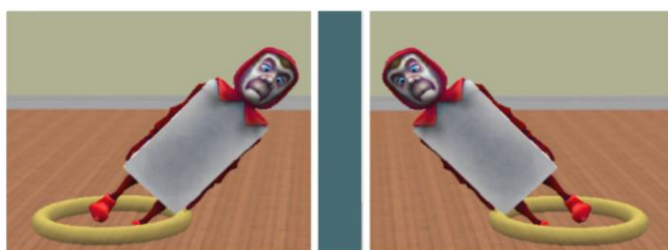
## Academy

# Java Fundamentals

2-4

## Rotação e Randomização

**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:
  - Correlacionar instruções de storyboard com tarefas de execução do programa
  - Adicionar uma instrução de controle ao Code editor
  - Usar números aleatórios para randomizar o movimento



## Desenvolvimento de Programa Usando uma Abordagem de Cima para Baixo

- Da mesma forma que um empreiteiro usa um conjunto de projetos e segue várias etapas, um programador usa um plano e segue uma série de etapas para construir um programa
- Os empreiteiros trabalham com uma finalidade e definem metas específicas quando constroem: os cômodos devem ter eficiência energética, a casa deve atender aos códigos de construção, etc

## Desenvolvimento de Programa Usando uma Abordagem de Cima para Baixo

- Da mesma forma, os programadores também trabalham com uma finalidade e definem metas porque, sem elas, não é possível avaliar o sucesso
- Uma outra condição para a meta ou finalidade de uma animação é um cenário de alto nível: uma história com uma finalidade

## Etapas para Usar uma Abordagem de Cima para Baixo na Programação

- Defina o cenário de alto nível (a finalidade/meta do programa)
- Documente as ações passo a passo em um storyboard textual
- Isso ajuda a adquirir uma compreensão das ações que precisam ser programadas

## Etapas para Usar uma Abordagem de Cima para Baixo na Programação

- Crie uma tabela para alinhar as etapas do storyboard com as instruções de programação
- Revise a tabela durante o desenvolvimento da animação para garantir que a animação atenda às especificações
- Revise a tabela conforme necessário

## Exemplo de Storyboard Textual

- Programe as ações a seguir em ordem:
  - O coelho entra
  - Alice vira-se para olhar para ele
  - O coelho diz: "**Esta é a festa?**"
  - Alice diz: "**Oh, não!**"
  - O coelho vira-se para a esquerda
- Programe as ações a seguir juntas:
  - O coelho sai rapidamente
  - Alice balança a cabeça



Quanto mais exato for o storyboard, mais fácil será de criar a codificação. Certifique-se de que você detalhou cada etapa individual do cenário.



## Alinhar o Storyboard com as Instruções de Programação

Ordem de Instruções	Ação do Storyboard	Instruções de Programação
Do in order	O coelho entra	O coelho move-se dois metros para frente
	Alice vira-se para olhar para ele	Alice vira o rosto para o Coelho
	O coelho diz: "Esta é a festa?"	O coelho diz: "Esta é a festa?"
	Alice diz: "Oh, não!"	Alice diz: "Oh, não!"
	O coelho vira-se para a esquerda	O coelho vira-se 0,25 para a esquerda
Do together	O coelho sai rapidamente Alice balança a cabeça	O coelho move-se quatro metros para frente em um segundo Alice vira a cabeça para a esquerda, para a direita e depois para a esquerda novamente

Este é um exemplo de como as ações definidas no storyboard estão diretamente relacionadas com as instruções de programação.

# Movimento do Objeto

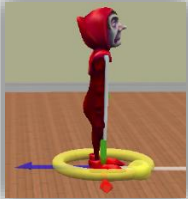
- O movimento do objeto é egocêntrico:
  - Os objetos se movem com base na direção para onde estão direcionados
- Um objeto pode se mover em seis direções:
  - Para cima
  - Para baixo
  - Para frente
  - Para trás
  - Para a direita
  - Para a esquerda

# Exemplos de Procedimentos de Rotação

Procedimento	Descrição
Turn	Gira um objeto para a esquerda, para a direita, para frente ou para trás em seu ponto central Giro para a esquerda ou para a direita no eixo vertical do objeto; giro para frente ou para trás no eixo horizontal do objeto
Roll	Rola um objeto para esquerda ou para a direita em seu ponto central usando o eixo horizontal do objeto

# Comparação entre os Procedimentos Turn e Roll

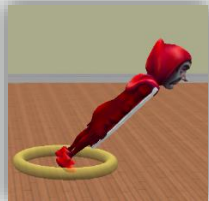
- O objeto **girando para a esquerda e para a direita** em seu ponto central usando um eixo vertical



- O objeto **girando para a esquerda e para a direita** em seu ponto central usando um eixo horizontal



- O objeto **girando para frente** em seu ponto central usando um eixo horizontal



(A carta do baralho já estava de frente para a esquerda  
**Virar para frente** fez com que ela se inclinasse para frente)

- O objeto **girando para trás** em seu ponto central usando um eixo horizontal



(A carta do baralho estava de frente para o lado esquerdo  
**Virar para trás** fez com que ela se inclinasse para trás)

**ORACLE**  
Academy

JF 2-4  
Rotação e Randomização

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.

12

Neste ponto, provavelmente seria útil criar uma animação simples que contivesse um único personagem e testar cada um dos procedimentos roll (de rolagem) e turn (de giro).

## Exemplo de Rotação da Subparte

- Alguns objetos têm subpartes móveis que podem girar e rolar
- O relógio de bolso tem ponteiros móveis de horas e minutos que podem rolar no ponto central do relógio
- O segredo para uma rotação de sucesso é saber qual é o ponto central de um objeto



Fazer com que subpartes de objetos do cenário se movam pode tornar a animação bem mais realista.

## Etapas para Programar um Objeto para Girar

- Selecione a instância do objeto a ser girado
- Arraste um procedimento de giro ou rolagem até o Code editor
- Defina o argumento de direção
- Defina o argumento de quantidade (1,0 = um giro ou uma rolagem completa)

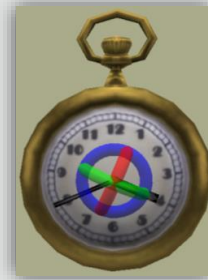
Observação: uma linha de indicador de posição verde aparecerá para ajudar você a alinhar a instrução de programação na posição desejada



Ao girar um objeto, pense em como as subpartes se moveriam como parte da rotação.

## Rotação da Subparte

- Alguns objetos têm subpartes móveis
- Por exemplo, os ponteiros do relógio podem girar e rolar
- A rotação pode ser aplicada a um objeto inteiro ou a subpartes específicas do objeto
- A subparte de um objeto exibe anéis que mostram a área de extensão do movimento



Você pode usar os anéis para mover manualmente as subpartes, a fim de alterar seu estado inicial.

## Etapas para Programar uma Subparte de um Objeto para Girar

- Selecione a instância da subparte do objeto a ser girado
- Arraste um procedimento de giro ou rolagem até o Code editor



- Selecione o valor de direção do argumento de direção
- Defina um valor de rotação do argumento de quantidade (1,0 = um giro ou uma rolagem completa)



## Instruções de Controle

- As instruções de controle definem como as instruções de programação são executadas no programa
- `myFirstMethod` é criado com uma instrução de controle `Do in order` padrão
- Dentro dele, por padrão, todas as instruções de programação são executadas sequencialmente

Essas provavelmente são as instruções de controle do Alice 3 que você usará com mais frequência.

# Instruções de Controle

Procedimento	Descrição
Do in order	Executa em ordem sequencial as instruções contidas na instrução de controle
Do together	Executa simultaneamente as instruções contidas na instrução de controle
Count	Executa um número específico de vezes as instruções de contidas dentro da instrução de controle
While	Executa as instruções contidas dentro da instrução de controle enquanto uma condição especificada é verdadeira

Essas provavelmente são as instruções de controle do Alice 3 que você usará com mais frequência.

# Instruções de Controle no Code Editor

- As instruções de controle estão localizadas na parte inferior do Code editor
- Elas representam blocos de código em que instruções individuais estão agrupadas
- Arraste os mosaicos de controle para o Code editor e, em seguida, adicione instruções a eles



As instruções de controle podem ser adicionadas antes ou depois do código que será inserido com elas.

## Adicionar Procedimentos a uma Instrução de Controle

- Arraste as instruções de programação novas ou existentes até a instruções de Controle
- No exemplo ilustrado abaixo, o objeto Alice se moverá para frente, virará para a esquerda e rolará para a direita em ordem



Vale a pena adicionar comentários que expliquem a finalidade do bloco do código, bem como as linhas individuais de código.

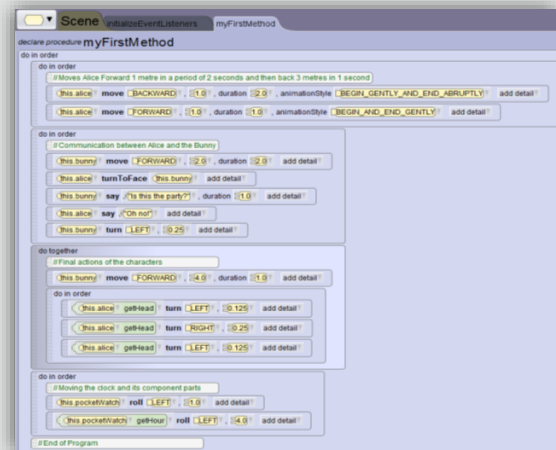
## Dica: Criar Blocos de Programação

- Embora myFirstMethod contenha uma instrução de controle Do in order padrão, você sempre poderá adicionar outros mosaicos Do in order para a área de programação
- Isso permite a você manter instruções de programação juntas em blocos organizados
- Os blocos podem ser facilmente copiados para a área de transferência ou movidos ao redor de um programa grande
- As instruções de controle Do in order aninhadas não interferem no desempenho da animação; elas fazem com que a edição do programa seja mais fácil para o programador

Os blocos de código devem ser criados ao redor de linhas de código que carregam uma finalidade comum. Se você estivesse criando o salto de um gato, poderia inserir todos os comandos desta ação em um bloco de código. Isso tornaria mais fácil desativar o código durante o teste porque bastaria desativar o bloco de código.

# Instruções de Controle Do In Order Aninhadas

- Aqui, três instruções Do in Order e uma instrução Do Together estão aninhadas dentro de uma quarta instrução Do in order
- As instruções aninhadas adicionam uma estrutura visual a um programa, da mesma forma que uma estrutura de tópicos faz em um relatório



O aninhamento é o processo de colocar uma coisa dentro da outra, como os ovos que são colocados lado a lado em uma cesta

# Números Aleatórios

- Os números aleatórios são números gerados pelo computador sem um padrão previsível em sua sequência
- Eles são gerados dentro de um intervalo específico de números
- Os computadores podem precisar de números aleatórios para:
  - Segurança:
    - por exemplo, senhas geradas de maneira aleatória
  - Simulação:
    - por exemplo, modelagem da ciência da terra (isto é, erosão ao longo do tempo)

Quando você usa números aleatórios, é muito normal ver o mesmo número se repetindo. Isso faz parte do comportamento aleatório.

## Exemplos de Números Aleatórios

- Os números aleatórios podem ser gerados dentro um amplo intervalo de números, tanto positivos quanto negativos, e podem incluir números inteiros ou partes fracionais de um número
- Exemplos de números aleatórios:
  - 15674
  - 6934022
  - 0.371
  - 89.763



# Números Aleatórios em Animações

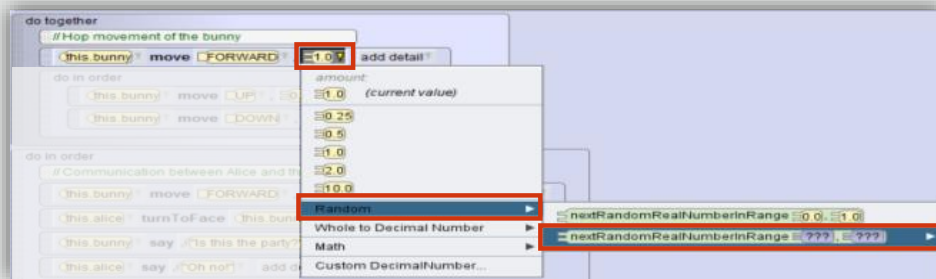
- Os animais não se movem em linhas geométricas retas
- Eles mudam levemente de direção à medida que andam, nadam e voam
- Os números aleatórios podem ser utilizados no argumento de distância de um método para que o objeto mova-se de maneira menos previsível e mais semelhante à vida real
- Em geral, os números aleatórios são usados ao criar jogos que exigem um comportamento imprevisível



O comportamento aleatório prende a atenção do usuário porque faz com que os personagens se comportem de maneira diferente em cada etapa da animação.

## Etapas para Usar Números Aleatórios

- Para qualquer argumento numérico, um das opções é um número aleatório
- Selecione o argumento na lista drop-down
  - Selecione Random
  - Selecione nextRandomRealNumberInRange
  - Selecione os valores baixo e alto

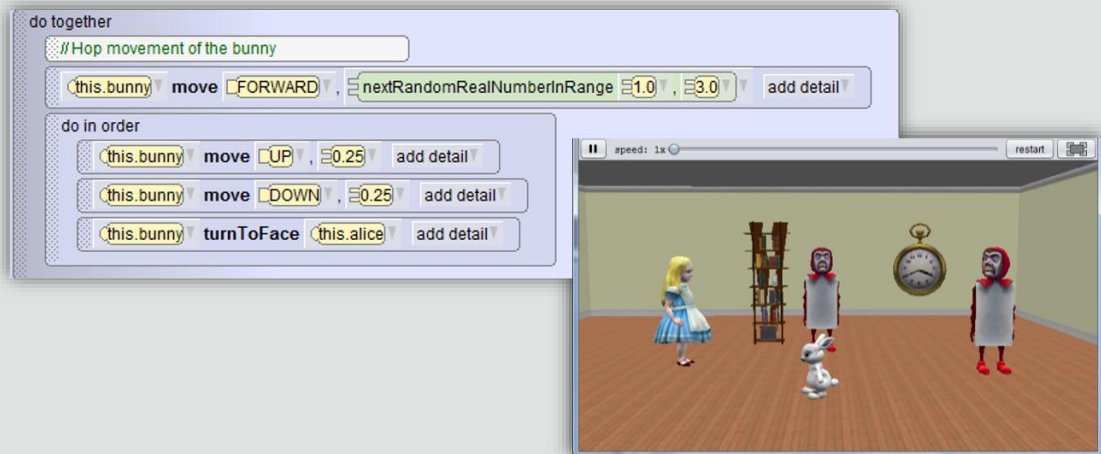


## Etapas para Usar Números Aleatórios

- Execute a animação
- O Alice 3 gerará aleatoriamente um valor dentro do intervalo selecionado quando a instrução for executada

## Exemplo de Animação com Número Aleatório

- O coelho move-se para frente em uma distância aleatória, deslocando-se para cima e para baixo antes de se virar em direção a Alice no run-time



Pode ser útil trocar os valores do intervalo aleatório e testar o programa novamente.

# Terminologia

- Estes são os principais termos usados nesta lição:
  - Instruções de controle
  - Aninhamento
  - Números aleatórios
  - Storyboard textual

# Resumo

- Nesta lição, você deverá ter aprendido a:
  - Correlacionar instruções de storyboard com tarefas de execução do programa
  - Adicionar uma instrução de controle ao Code editor
  - Usar números aleatórios para randomizar o movimento



