

# CompSci 4

## Chap 4 Sec 2

Sept 18, 2007

Prof. Susan Rodger



# Announcements

- Assignment 3 storyboard due today!
- Assignment 3 worlds due Thursday
  - Turn worlds in on Blackboard
  - Name textfile “README”
  - Put all three files in a folder named  
assign3LOGINID  
where LOGINID is your Duke login

# What we will do today

- Chapter 4 Sec 2 – Parameters
- Classwork

# Overview

- The need for more flexible methods
- Creating methods with parameters
- Passing arguments to methods
- Demo
  - Using Alice – methods with parameters

# A Beetle Band



- Create an animation for a bug band as an ad for their next concert
- Each band member will perform a short solo

# Storyboards

- Each bug member will perform a solo

*Do together*  
*Do in order*  
**georgeBeetle** move up  
**georgeBeetle** move down  
*play sound*

*Do together*  
*Do in order*  
**ringoBeetle** move up  
**ringoBeetle** move down  
*play sound*

*Do together*  
*Do in order*  
**paulBeetle** move up  
**paulBeetle** move down  
*play sound*

*Do together*  
*Do in order*  
**lennonBeetle** move up  
**lennonBeetle** move down  
*play sound*

# Code for georgeBeetle



- We will need four versions of the code, one for each band member
- This code will only work for georgeBeetle

# A Better Solution

- Four versions of similar code is tedious.
- The things that change are
  - The beetle
  - The music the beetle plays
- Better Solution: write a more flexible method

# Parameters

- Built-in methods provide **flexibility** by providing parameters such as distance and duration
- Parameters allow you to pass in values
  - The values are **arguments**
- **Example:**

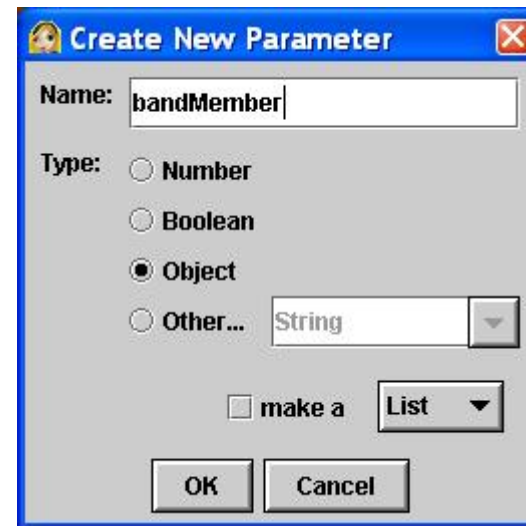


Parameters: distance, duration

Arguments: 0.5 meters, 0.5 seconds

# Types of Parameters

- Alice provides several types of parameters that can be used in your methods.



# The Storyboard

- Write one method and use parameters for
  - Which bandmember to perform
  - Which music to play

Solo:

Parameters: **bandMember**, **music**

Do together

Do in order

**bandMember** move up

**bandMember** move down

play **music**

# Creating a Method Named Solo

- Can be used for any band member
- Will need
  - An Object parameter – which band member is to play a solo
  - A Sound parameter – which music should be played



# World.solo with parameters

The image shows a Scratch script editor window with two tabs: 'World.my first method' and 'World.solo'. The 'World.solo' tab is active. Below the tab, the script is defined with parameters: 'World.solo' followed by an object parameter 'Obj bandMember' and a sound parameter 'music'. Below the parameters, it says 'No variables'. The script area contains a 'Do together' block, which is expanded to show a 'Do in order' block. The 'Do in order' block contains three steps: 1. 'bandMember' moves 'up' by '0.5 meters' with a 'duration = 0.5 seconds'. 2. 'bandMember' moves 'down' by '0.5 meters' with a 'duration = 0.5 seconds'. 3. 'World' plays the sound 'music'.

World.my first method   World.solo

World.solo   Obj bandMember ,   music

No variables

☒ Do together

☒ Do in order

bandMember   move   up   0.5 meters   duration = 0.5 seconds   more...

bandMember   move   down   0.5 meters   duration = 0.5 seconds   more...

World   play sound   music   more...

# Calling the solo method

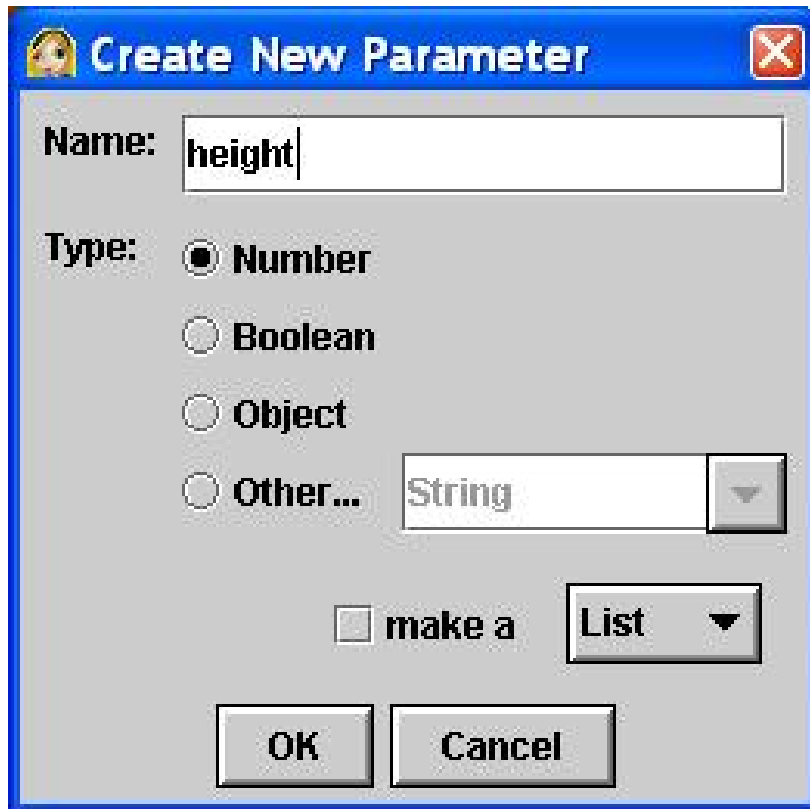
The screenshot shows a NetLogo interface with two tabs at the top: 'World.my first method' (selected) and 'World.solo'. Below the tabs, the text 'World.my first method No parameters' is displayed. Underneath, it says 'No variables'. A section titled 'Do in order' contains four rows of code blocks, each representing a call to the 'World.solo' method with two arguments: 'bandMember' and 'music'.

bandMember	music
georgeBeetle	World.bassGuitar
lennonBeetle	World.guitarSolo
ringoBeetle	World.DRUM
paulBeetle	World.saxophone

- Note that in each call, arguments must be given for both parameters

# A Number Parameter

- Add a number parameter for **height** the **bandMember** jumps up and down



A screenshot of a 'Create New Parameter' dialog box. The dialog has a blue title bar with a small icon on the left and a red 'X' close button on the right. The main area is light gray. It contains a 'Name:' label followed by a text box with the word 'height' inside. Below this is a 'Type:' label followed by four radio button options: 'Number' (which is selected), 'Boolean', 'Object', and 'Other...'. To the right of the 'Other...' option is a text box containing the word 'String' and a small downward arrow. Below these options is a checkbox labeled 'make a' followed by a text box containing the word 'List' and a small downward arrow. At the bottom of the dialog are two buttons: 'OK' and 'Cancel'.

Create New Parameter

Name: height

Type: ☒ Number  
☐ Boolean  
☐ Object  
☐ Other... String

☐ make a List

OK Cancel

Demo

# Classwork Today

- Create worlds
  - Beetle band duet
  - Frog Escape

