CompSci 94
Chap 6 Sec 2
Sep 26, 2013

Prof. Susan Rodger

"All your troubles are due to those 'ifs'," declared the Wizard. If you were not a Flutterbudget you wouldn't worry."
- The Emerald City of Oz by Frank Baum

Announcements

- Review for test next time.
  - Hand out Test 1 from last two semesters
    - Should try it before next class
  - Old Quizzes are available on Sakai
  - Study classwork and lecture notes
  - Assignment 3 due today
- Prof. Rodger is out of town next week
  - will answer questions on Piazza
  - Today – Chap 6, Sec 2
  - Execution control – if/else & Boolean functions
  - Relational operators
  - Logical Operators

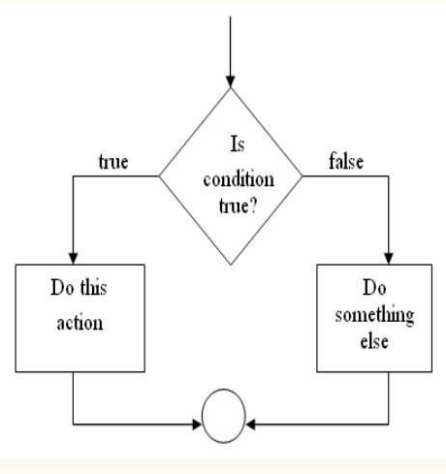
Thinking - More Advanced Worlds

- How do you build animations like simulations and video games?
- Need to write code that involves **decisions**
- Example car-race simulation
  - If the car stays on the road the score increases
  - If the car goes off the road into the stands, the car crashes
  - If the driver gets the car over the finish line, the time is posted and the driver wins!

Logical Expressions

- Decision is made based on current conditions.
- Condition is checked in a logical expression that evaluates to *true* or *false* (Boolean) value.
  - car on road → true
  - car over finish line → false

If/Else



- In Alice, a logical expression is used as the condition in an If/Else control structure
- Decisions (using If/Else) are used in
  - Functions
  - Methods

What this tutorial shows

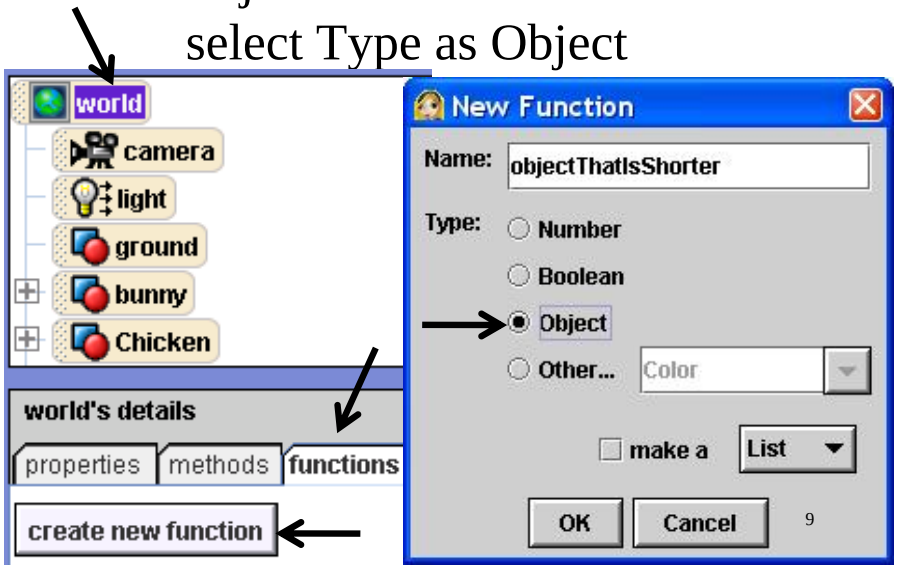
- You've used built-in functions. Now you will write your own functions.
- NOTES: Not all objects can change color. The chicken and snowman can change color, which is why we use them. The bunny cannot change color, the object was just designed that way.

Parts – Build this world with me

- Part 1: Create a "world" function that returns the shorter of two objects – make it a world function because it uses two different objects
- Part 2: Create a "world" function that returns the shorter of three objects – make it a world function because it uses three different objects
- Part 3: Create a chicken function that returns a color (cycling through 4 colors) and use the function to change the color of the chicken

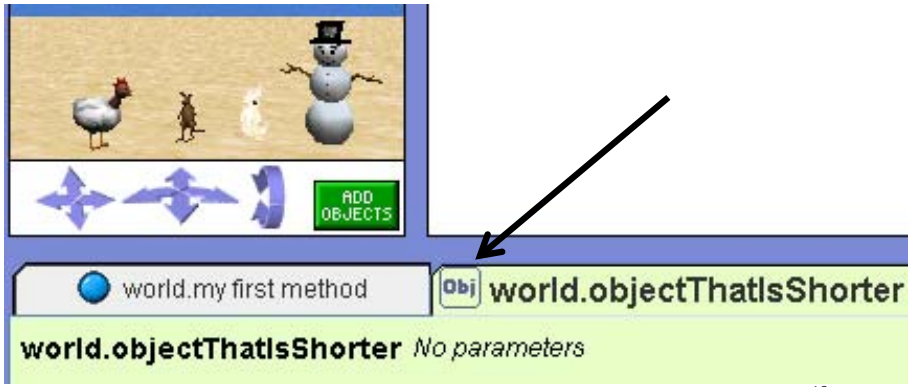
Part 1: Start Alice world with sand and add in chicken, snowman, bunny, and joey

Create World function "ObjectThatIsShorter" – select Type as Object



9

The new function appears – notice the "Obj" – means this function returns an object



10

What this function will do

- A function computes an answer and returns the answer.
- The function ObjectThatIsShorter will compare the chicken and the snowman in height and the answer is the one that is shorter. The function returns the shorter object.
- Since the function type is "object" the function must return an "object"

11

What is the question we will ask?

- Has to do with chicken's height and snowman's height

- If _____

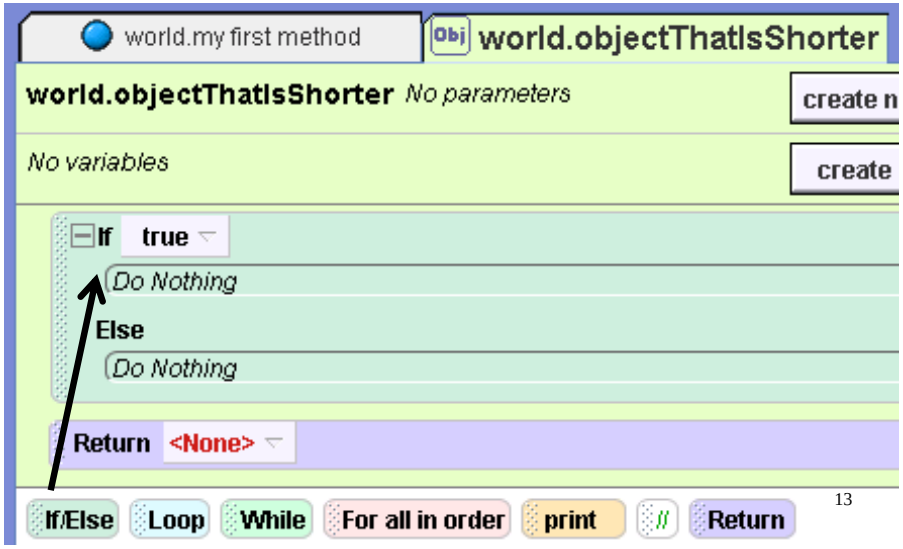
  then we know the answer, so return the answer

Otherwise (else)

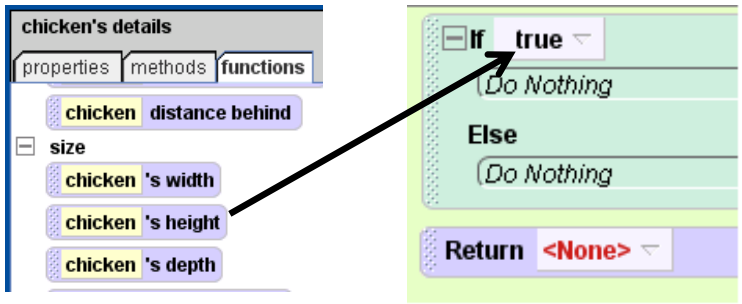
we also know the answer, return the answer

12

First make a decision (ask a question) - Drag up if/else



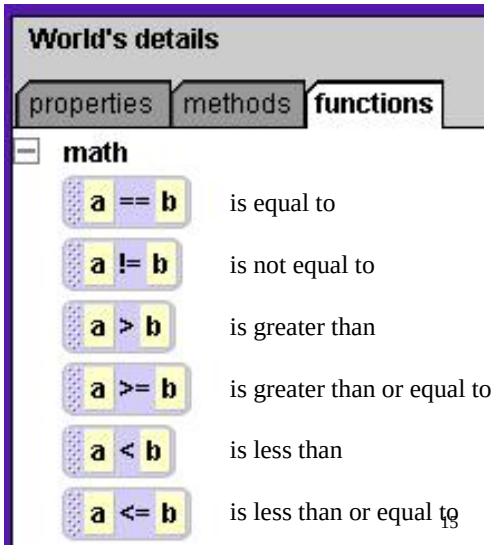
Next drag over and drop in Chicken's height



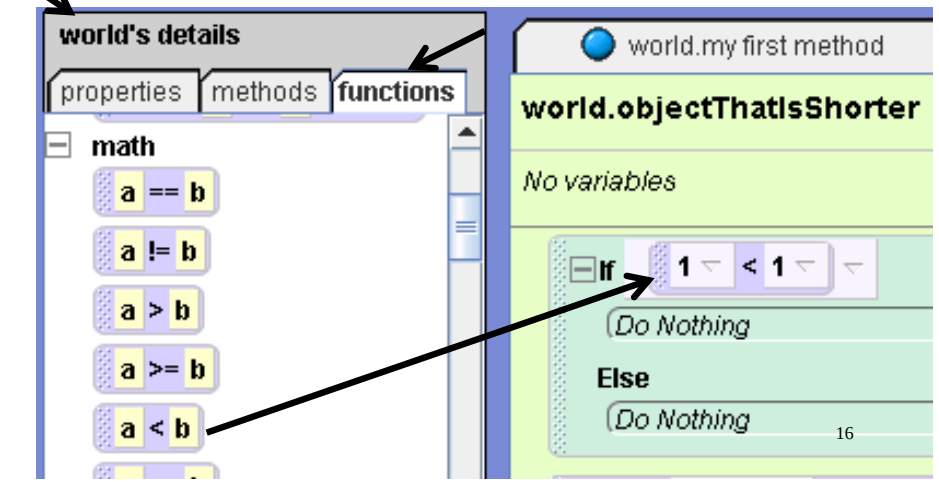
Why can't you drop chicken's height on the true?

Using a Relational Operator

- Use the < relational operator from the World's built-in functions to compare the two heights



Compare chicken and snowman's heights – from world functions, drag over the "a<b" into the "true" and enter 1's



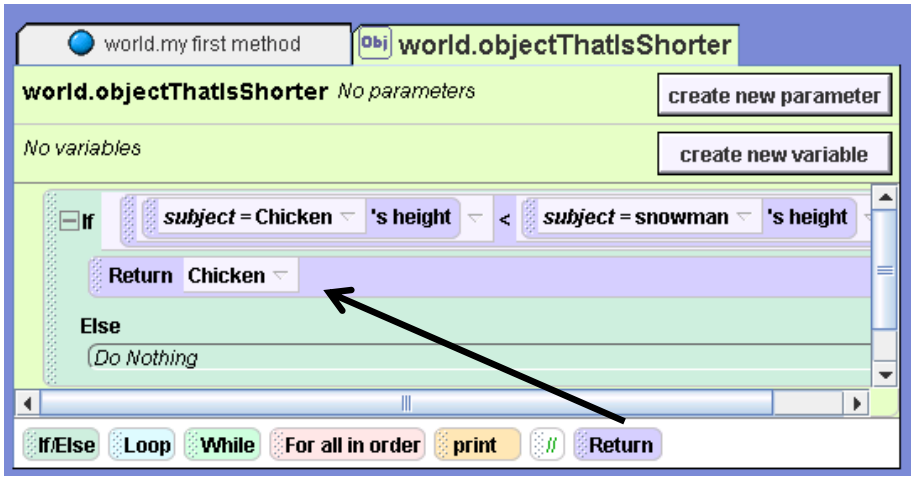
Now click on Chicken, then functions, and drag "Chicken's height" over twice

Click on the second "Chicken" and change to "snowman"

Compute the answer

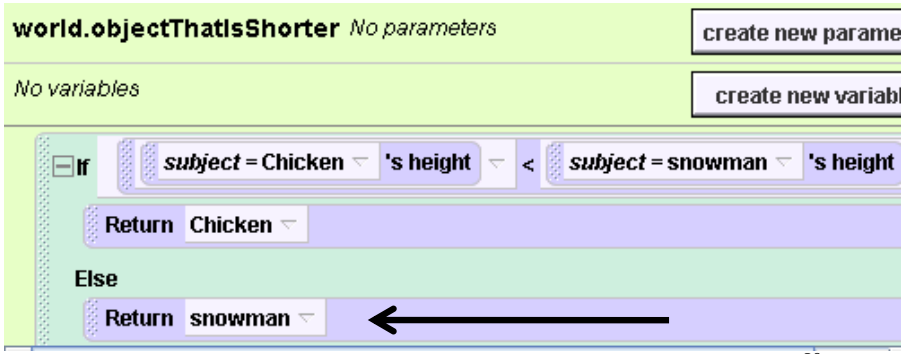
- If the chicken's height is less than the snowman's height, then we know the answer is the Chicken. The "answer" is put in the line after the "if" and before the "else" (see the next slide)
- To return the answer, drag up "Return" from the bottom of the window (see the next slide)

Drag up "return" and select Chicken



If Chicken is taller than snowman

- Then the answer (or shortest object) is snowman
- After the "else" return snowman

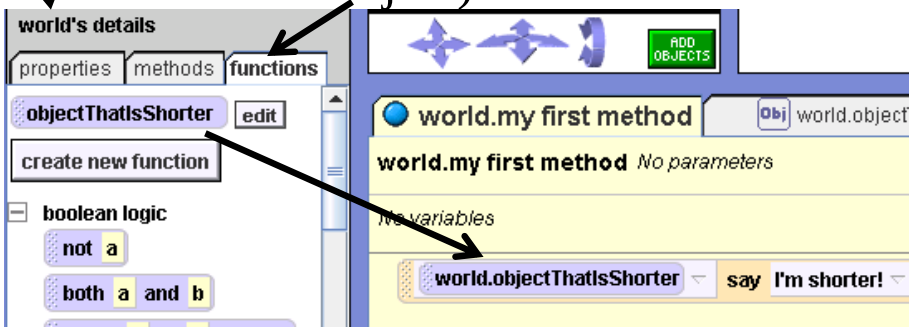


Notes on Functions

- Note in previous example that the answer is either the “chicken” or the “snowman”, but not both. Only one of the two is returned.
- When a function executes“return,” you leave a function and do not execute any more code in the function
- A function must always return an “answer”

Now try out the function, use it where there is an object. In myFirstMethod, put this code…

Replace “chicken” with new function – in World Functions, drag over the function “objectThatIsShorter” (that returns an object) over Chicken.



objectThatIsShorter can be used in place of any object.

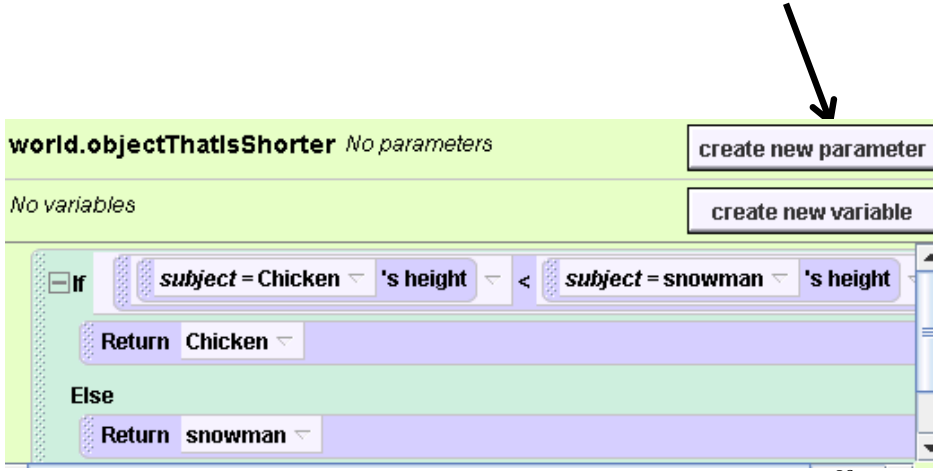
Click Play

- Only the Chicken and snowman’s height are compared and the one that is shorter (the chicken) says “I’m shorter!”
- Does your function really work? Resize the chicken so it is larger than the snowman and then play again. This time the snowman should say “I’m shorter!”

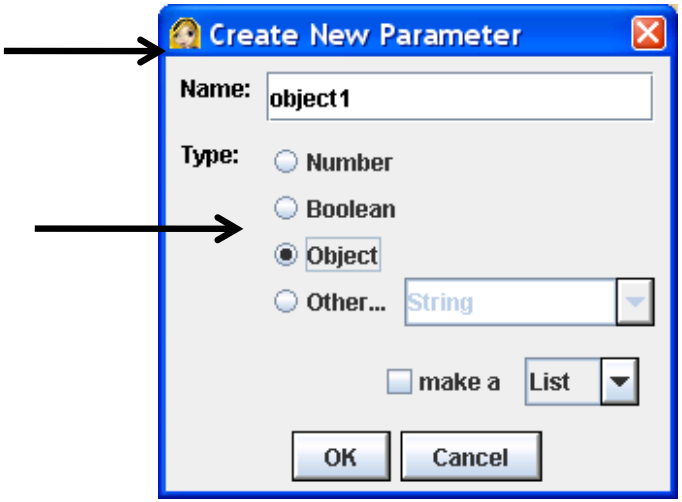
Adding Flexibility

- We wrote the function to compare the chicken and snowman's heights.
- For Example, notice that with the "move" command, it is flexible in that you get to pick the direction and the distance to move.
- To make this function more flexible, we will add two parameters (choices, so you can pick the two objects to compare).

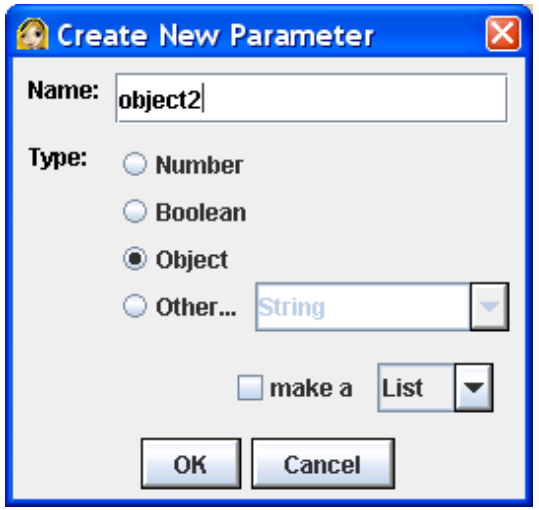
Click on "create new parameter"



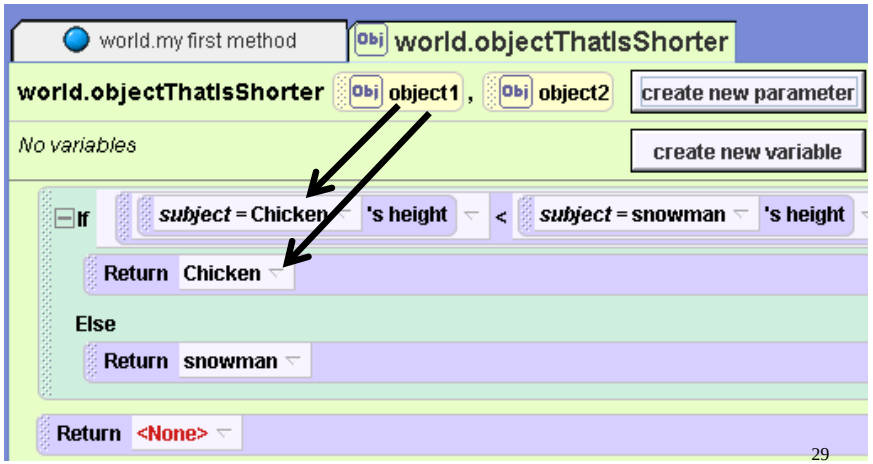
Enter the name "object1" and select type "Object"



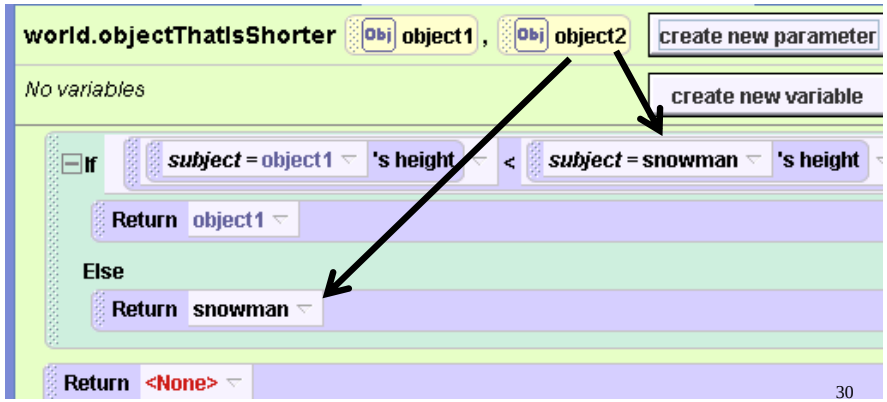
Add another parameter named "object2" of type "Object"



Click and drag “object1” over both Chickens

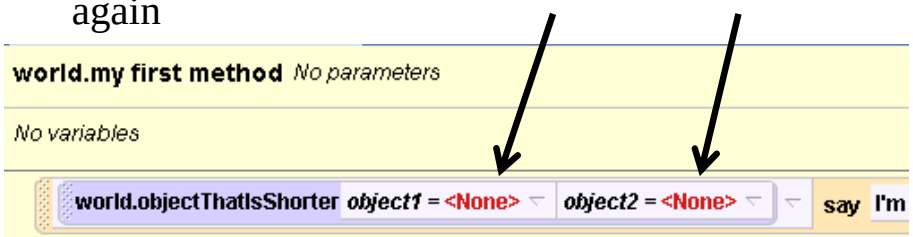


Click and drag object2 on top of both of the snowman’s



Now let’s Test the method

- Back in myfirstmethod, note that now you have to choose two objects to compare in the function objectThatIsShorter.
- Select two animals and Play.
- Then select two different animals and play again

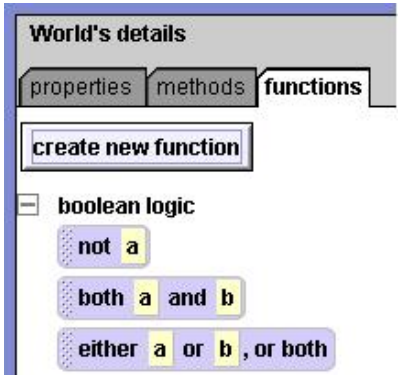


Resize objects to make some taller, and click Play again

Part 2: Using Boolean Logic

- Suppose we want the shorter of three objects.
- How do we find the shorter of the Chicken, the joey and the bunny?

Logical Operators



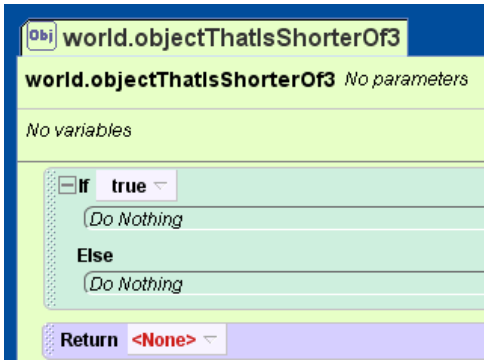
- Use Boolean logic operators to check more than one condition

What do we ask?

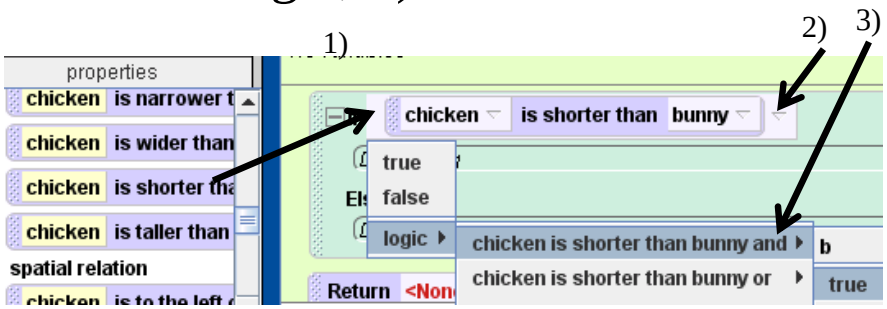
- If the chicken is shorter than the bunny AND the chicken is shorter than the joey then do we know which of the three is the shortest?
- Note: Both must be true for the whole statement to be true!

Create another World function called objectThatIsShorterOf3

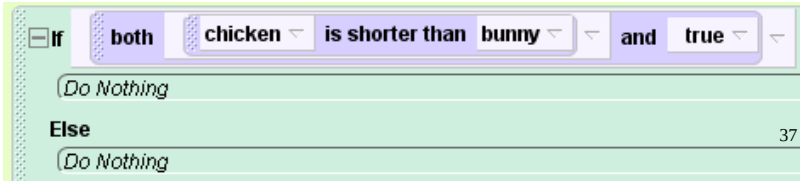
- Return value should be type Object
- Drag up an if statement



1) Drag over shorter than, 2) select logic, 3) select “and” true



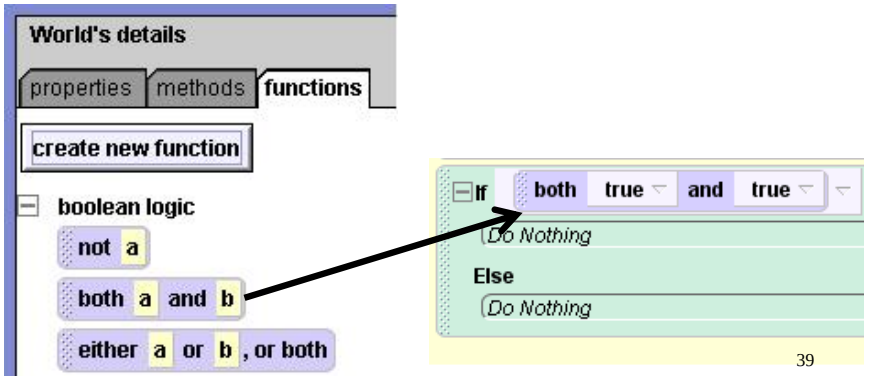
Result is:



Put in the second case over the true

Another way to get the AND in:

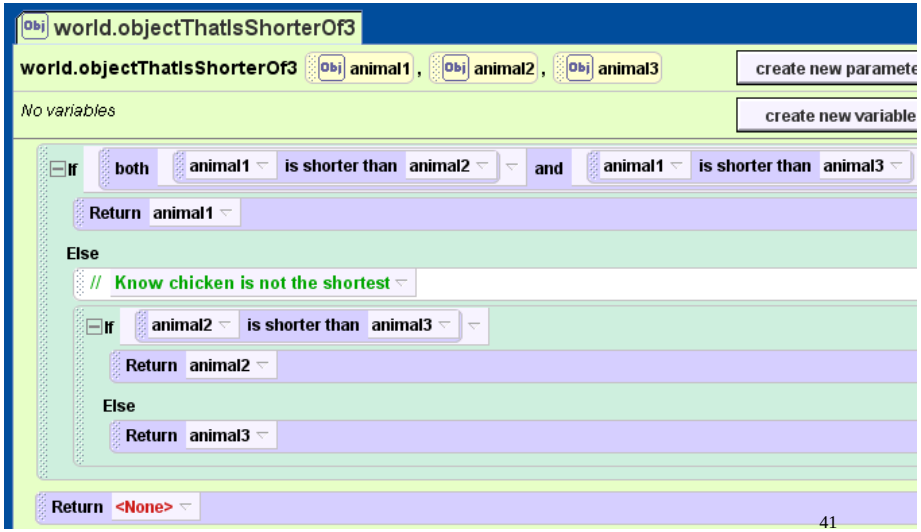
- Could drag the AND over the true and then fill in both parts



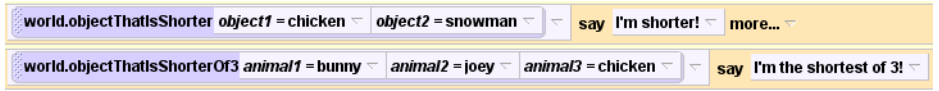
Complete the function

- Can that last Return <None> be executed?

Add in three object parameters



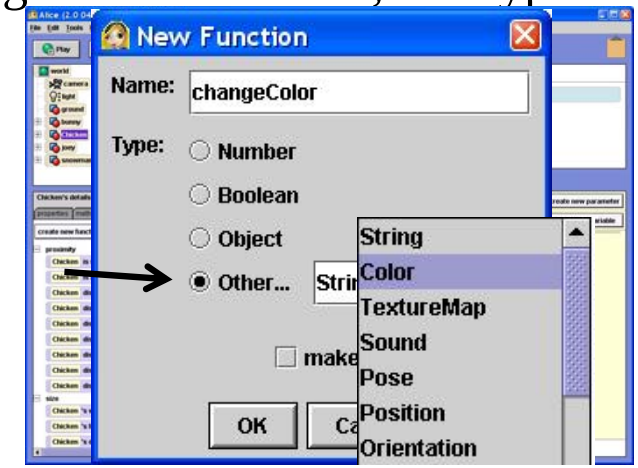
Test out the function



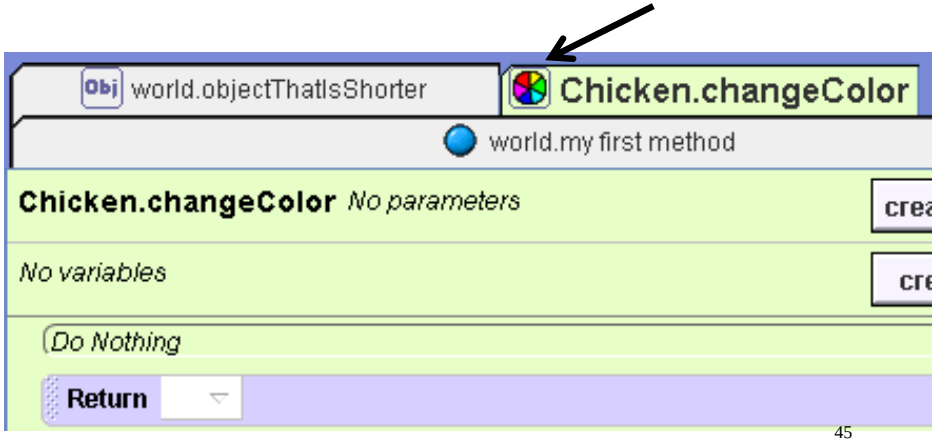
Part 3: Change the color of the chicken from “no color” to “blue” to “red” to “green” and cycle through again

- In this part we want to have the chicken cycle through a set of colors from white(no color) to blue to red to green to white, etc.
- We will write a function to check what the current color of the chicken is and from that compute what the next color in the cycle should be.
- The function only is about the chicken so it should be a chicken function

Create Chicken Function “changeColor” Click on Chicken, then functions, then “create new function”, type “changeColor” as name, and type “Color”



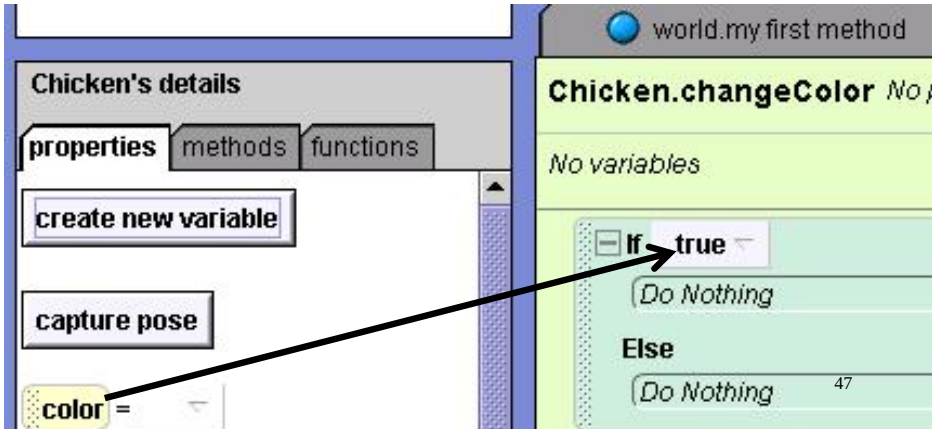
The new function appears –
notice the color wheel –
this function returns a new color



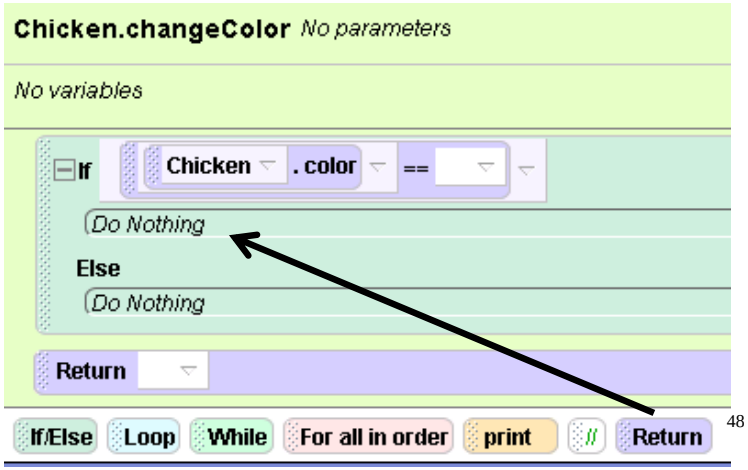
Returning new Chicken color - Idea

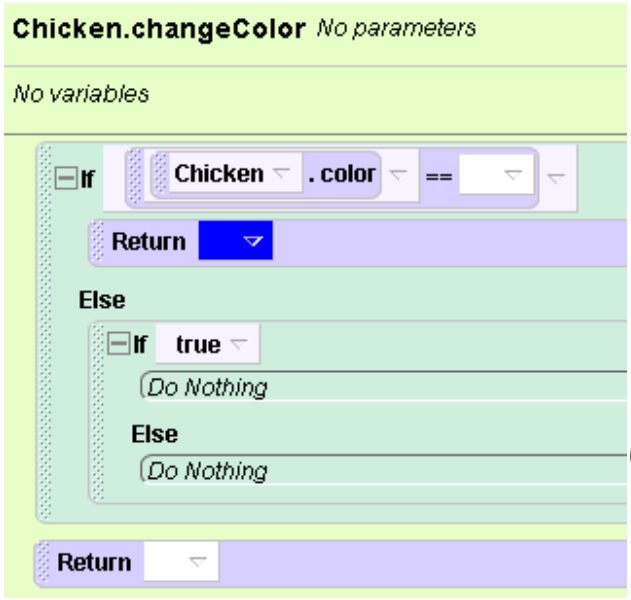
- If the chicken color is “no color” then we want to return the new color “blue”
- Else, if the chicken color is “blue”, then we want to return the new color “red”
- Else, if the chicken color is “red”, then we want to return the new color “green”
- Else if the chicken color is “green”, then we want to return the new color “no color”

First Drag up the if/else. Then select “Chicken”, “properties”. Then drag over the color property into “true” and select “Chicken.color == no color”



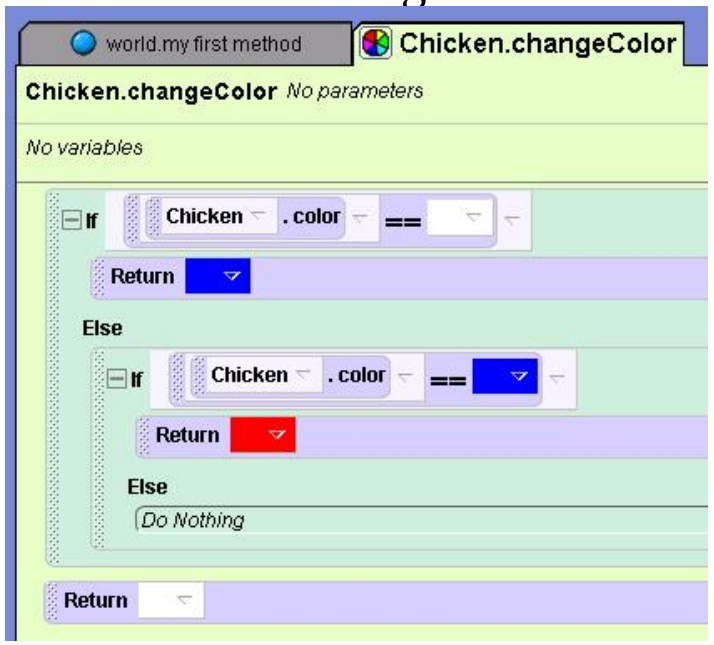
The “no color” when dropped looks like white. Now drag up “return”, drop in after the if, and select “Blue”





Under the "else" part, drag in another if/else, and repeat the second if (if the color is blue, return red)

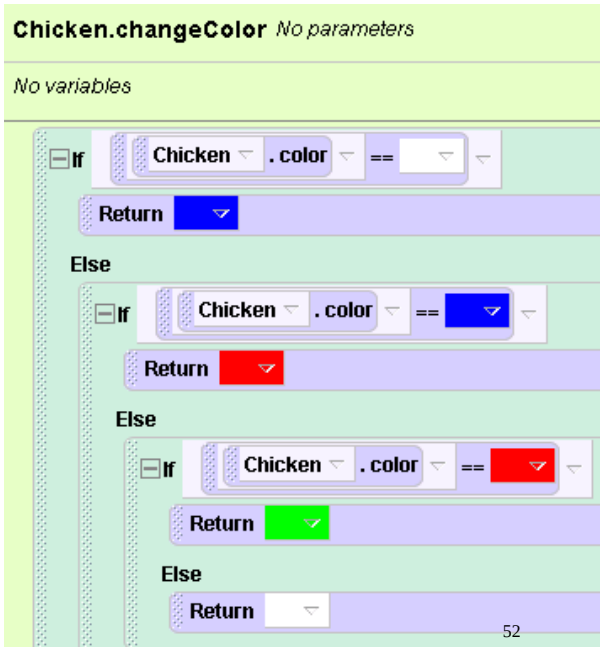
Resulting in:



Now continue with two more colors

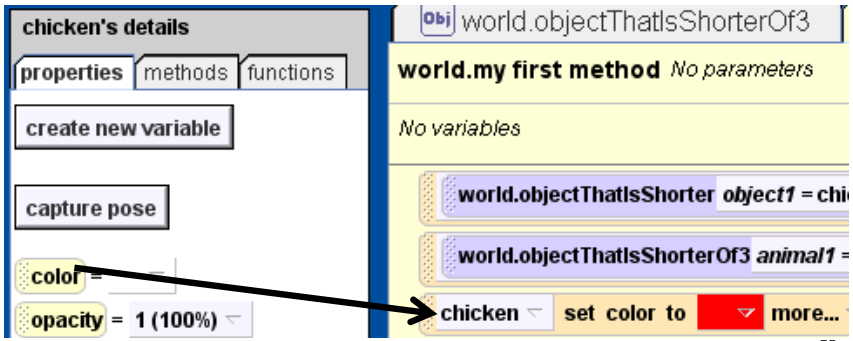
- If the chicken color is red, return green
- Else, return "no color"

Here is the final code… with three nested if's



Now let's use this new function to change the color of the chicken

- In myFirstMethod, drag over the chicken's color property and set it to red

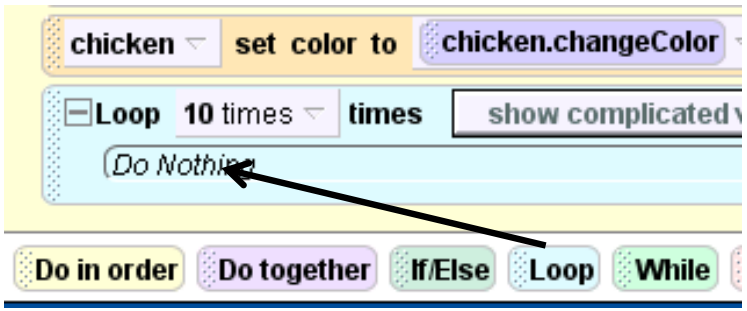


Now use the changeColor function

- Click on "Chicken" then "function" tab and drag the new function and drop it over the color red

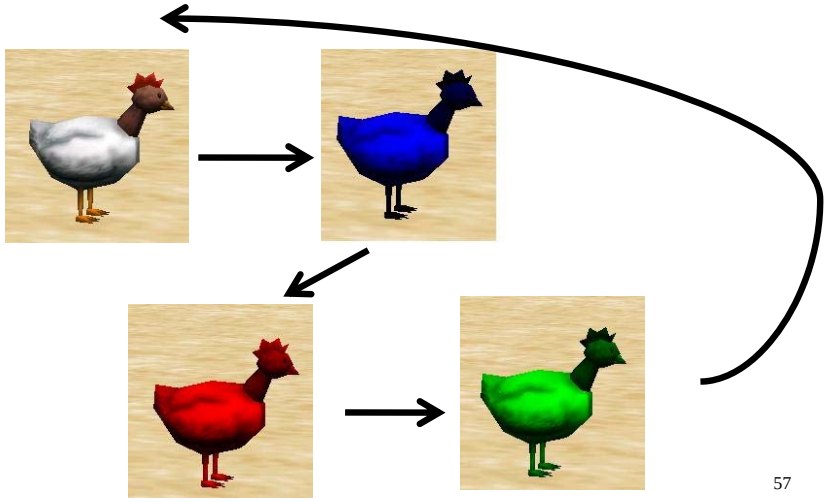
Have it repeat using loop

- Drag up the "loop" tab and select 10

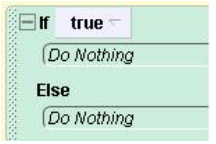


Then move the set color to inside the loop and Play!

Watch the chicken change colors



Check



- Where do you get the if?

- Do you have to fill all the parts of the if?
- Where do you find the relational operators?
- Where do you find the logical operators?

- What are the differences between a method and a function?

Random Numbers

- Skip, We will cover this later

Classwork today

- Write functions and methods with if/else