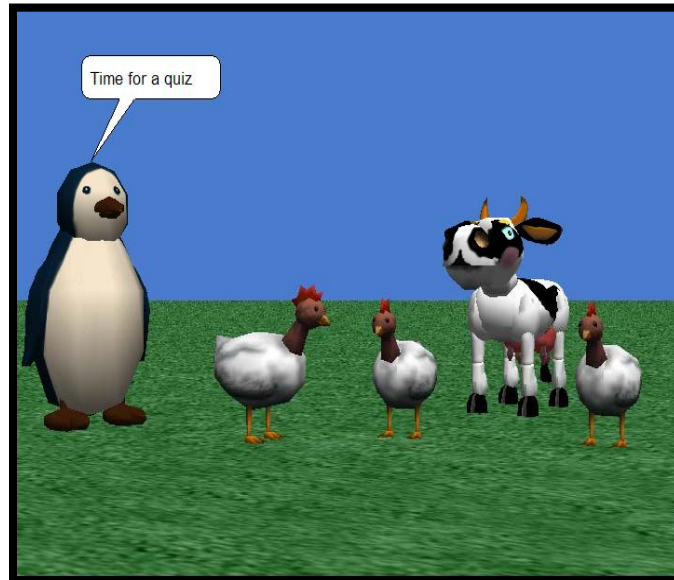


A Simple Quiz:

Ask User Functions.



By Lana Dyck
under the direction of
Professor Susan Rodger
Duke University
June 2009

Introduction and Set-up

This tutorial will demonstrate how to create a simple quiz using the three different kinds of “ask user” functions: ask user for a number, ask user for yes or no and ask user for a string.

Let's get started by setting up the world.

Open a new Alice **grass world**.

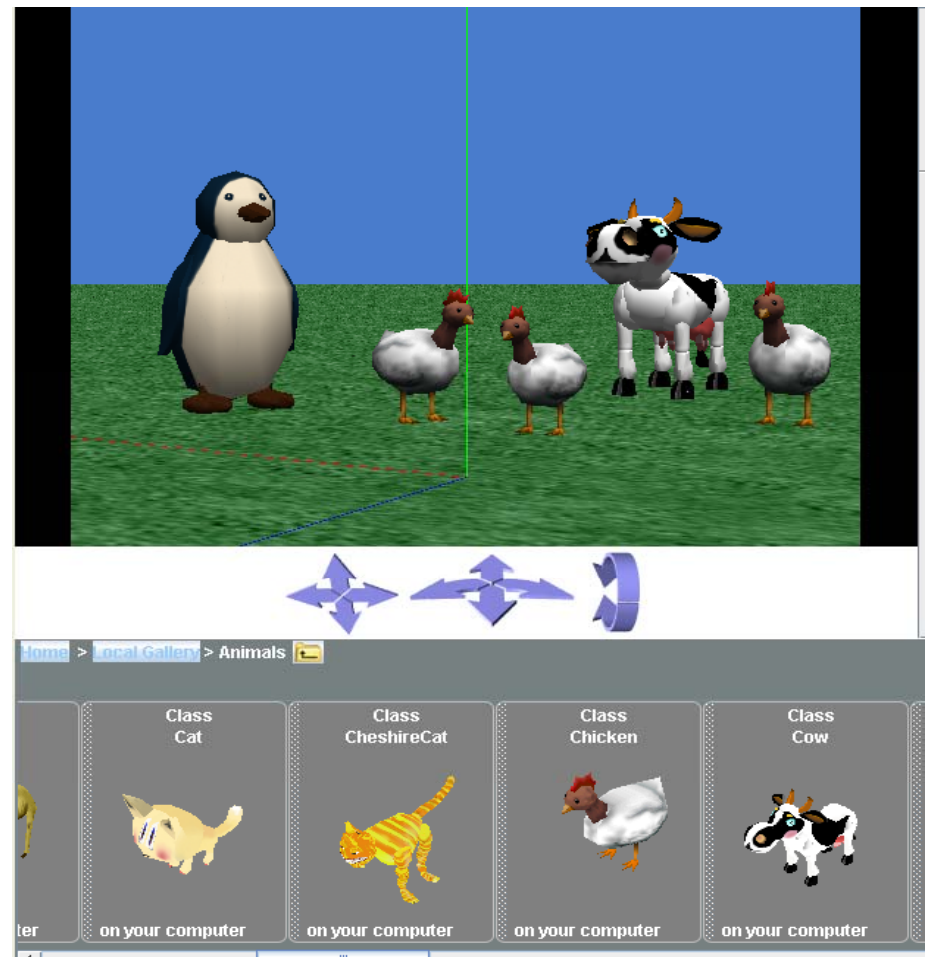
Click on **Add Objects**.

Choose **Local Gallery**, and click **Animals**.

Add one **Penguin**, one **Cow**, and three **Chickens** into your world.

Arrange and resize the objects so they are easy to see.

Click done when finished.



Step 1: Create the Quiz Method

Create a new method that we will use later to hold the instructions for the quiz.



Lets start by creating a quiz method.

Click on **world** in the object tree and create a **new method**.

Name the new method: **quiz**.

Click back on **my first method**.

Step 2: Set Up My First Method

Now we will add the quiz to my first method.

Make sure you are in **my first method**.

Click on **penguin** in the object tree.

Drag the **say** method into the method editor and type in: *Time for a quiz*

Drag in a second **say** method and type in: *That is all folks!*


Click on **world** in the object tree and drag the **quiz** method in between the two penguin say commands

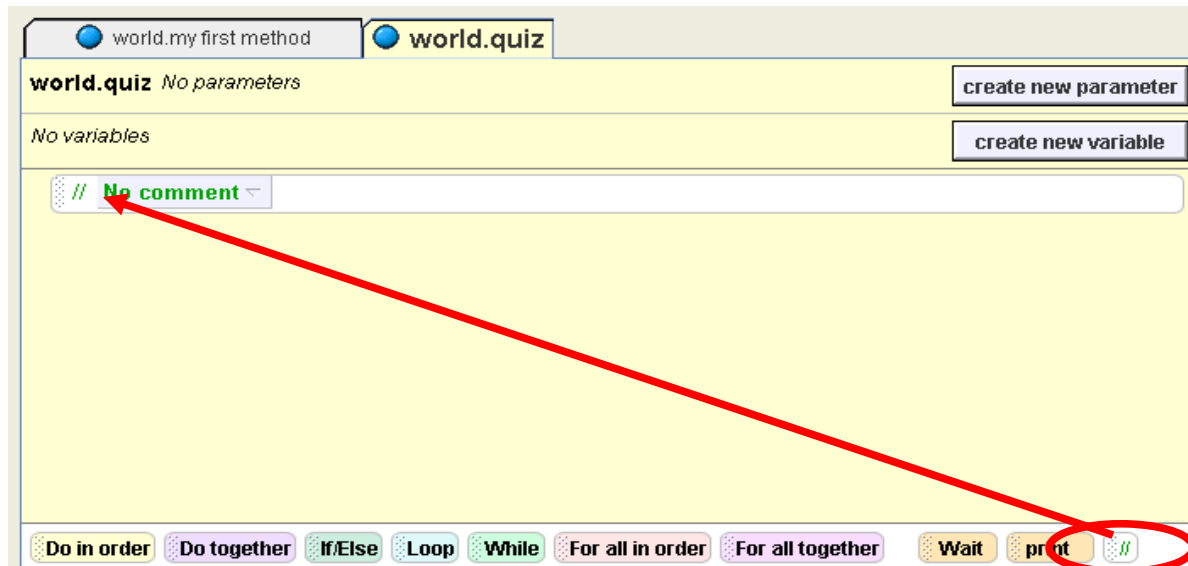
The screenshot shows a programming environment with two main panels. On the left, the 'world's details' panel has tabs for 'properties', 'methods', and 'functions'. Under the 'methods' tab, there are two items: 'my first method' and 'quiz', each with an 'edit' button. A red circle highlights the 'quiz' item, and a red arrow points from it to the right panel. The right panel is titled 'world.my first method' and contains a method editor. At the top, it says 'world.my first method No parameters' with a 'create new parameter' button. Below that, it says 'No variables' with a 'create new variable' button. The editor contains three lines of code: 'penguin say Time for a quiz more...', 'world.quiz', and 'penguin say That is all folks! more...'. The 'world.quiz' line is positioned between the two 'penguin say' lines.

Step 3: Comments

Now we will start writing the quiz method. To make our code easier to read we will add in comments. Comments make it possible to add in notes about the code like, what it will do, or who wrote the code.

First click on the `world.quiz` method.

To add a comment drag and drop the double slash button  into the method



Step 4: Question #1

Click on “No comment” in the method editor and type in: *Question 1*



```
// Question #1
```

Click on **penguin** in the object tree, have it **say**: *How many animals in the world?* and set the duration to 2 seconds.



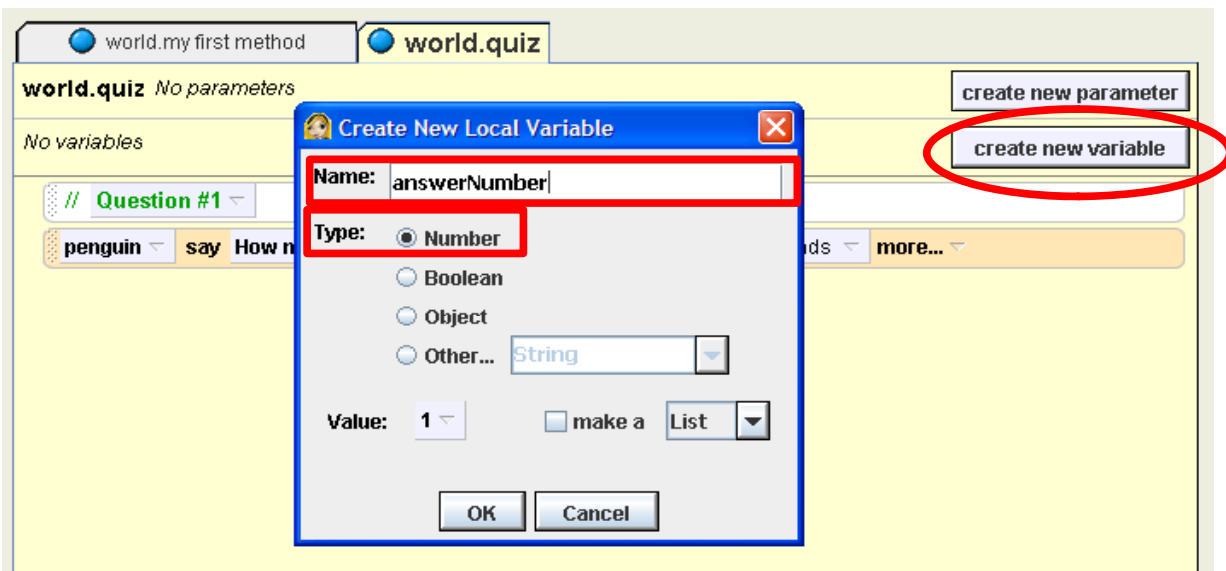
```
penguin say How many animals are in this world? duration = 2 seconds more...
```

We need a variable to store the number information we get from the user.

Click on **create new variable**.

Name it *answerNumber*.

Select **Number** beside Type and click **OK**.



Step 4: Question #1 Continued...

Next we will set the answerNumber value to the number the user will give.

Click and drag the variable **answerNumber** into the method and set the value to **1**

The screenshot shows a quiz editor interface for 'world.quiz'. At the top, there are two tabs: 'world.my first method' and 'world.quiz'. Below the tabs, the text 'world.quiz No parameters' is displayed, along with buttons for 'create new parameter' and 'create new variable'. A variable 'answerNumber' is shown with a value of '1', circled in red. Below this, a question is displayed: 'penguin say How many animals are in this world?' with a duration of '2 seconds'. A red arrow points from the circled 'answerNumber' variable to a 'set value' menu. This menu is open, showing a list of values: '0.25', '0.5', '1', and '2'. The value '1' is highlighted. Below the list are options for 'expressions' and 'other...'. At the bottom of the interface, there is a toolbar with various control buttons: 'Do in order', 'Do together', 'If/Else', 'Loop', 'While', 'For all in order', 'For all together', 'Wait', 'print', and a comment icon.

set value	value
increment world.quiz.answerNumber by 1	0.25
decrement world.quiz.answerNumber by 1	0.5
	1
	2
	expressions
	other...

Step 4: Question #1 Continued...

Click on the **world** in the object tree.

Under the world's details, **functions** find ask user for a number.

Click and drag the **ask user for a number** over the 1 and select **other...**

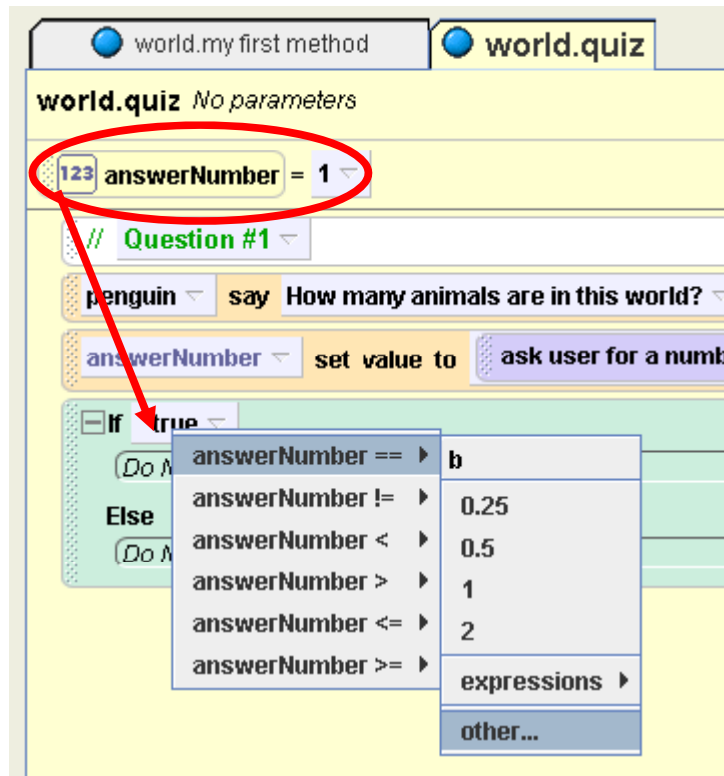
Enter the string: *Enter the number of animals:*

The screenshot shows the Scratch programming environment. On the left, the 'world's details' panel is open to the 'functions' tab, where 'ask user for a number' is circled in red. A red arrow points from this function to the '1' in the 'set value to' block of the 'world.quiz' script. A context menu is open over the '1', with 'other...' selected. A dialog box titled 'Enter a string' is open, showing the text 'Enter the number of animals:' in the input field.

Step 4: Question #1 Continued...

Now we will determine what happens if the answer given is correct or incorrect.

Drag and drop an **If/Else** block from the bottom of the window and set it to **true**.



Drag and drop the **answerNumber** variable over the true.

Select **answerNumber ==** and use **other...** to enter in the value **5**.
5 is the correct answer to this question.

Note: The following is an explanation of all the comparison options

$a == b$	a is equal to b
$a != b$	a does not equal b
$a < b$	a is less than b
$a > b$	a is greater than b
$a <= b$	a is less than or equal to b
$a >= b$	a is greater than or equal to b

Step 4: Question #1 Continued...

For this question when the “If” statement is true, the answer is correct. So we will put our response to the correct answer first.

Click on **penguin** in the object tree.

Under the If:

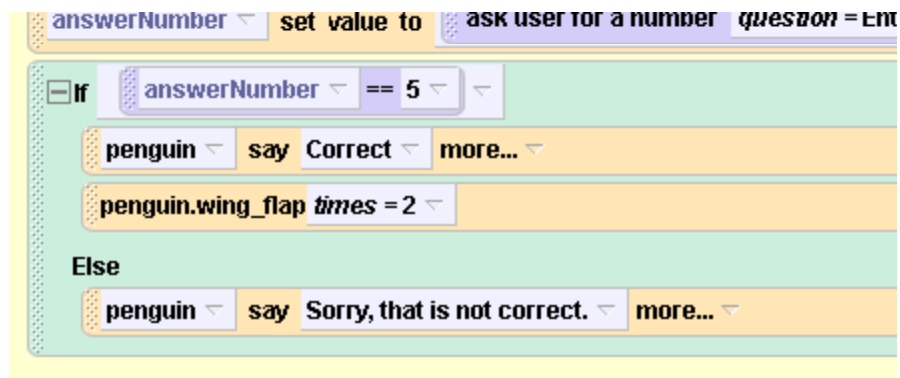
Drag the **say** method and type: *Correct*

Drag the **wing_flap** method and select **2**

When the if statement is false, the answer is incorrect.

Under the Else:

Drag the **say** method and type: *Sorry, that is not correct.*



Play your world to take the one question quiz.

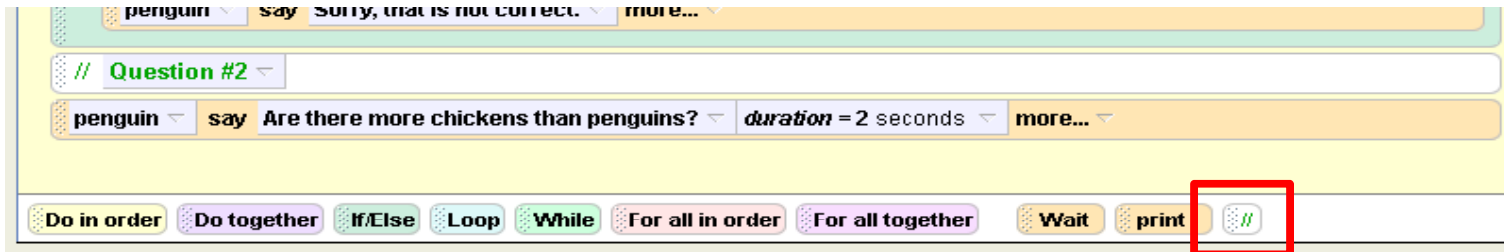


Step 5: Question #2

Now lets make a question that asks the user for a yes or no response.

Drop in a comment and type in: *Question 2*

Click on **penguin** in the object tree, have it **say**: *Are there more chickens than penguins?* and set the duration to 2 seconds.



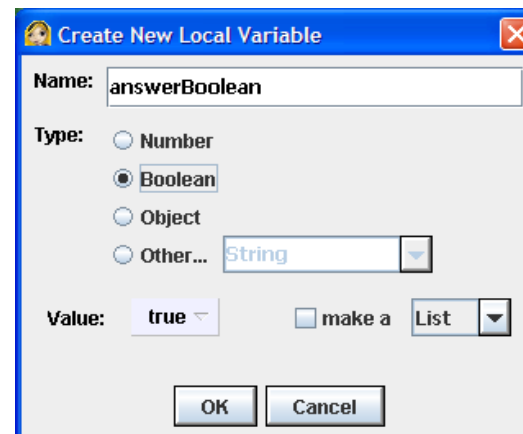
We need a variable to store the boolean information we get from the user.

Click on **create new variable**.

Name it *answerBoolean*.

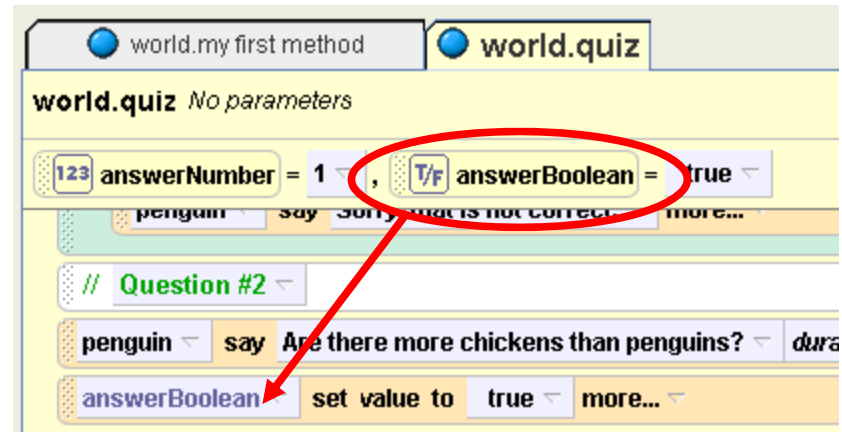
Select **Boolean**

Click **OK**.

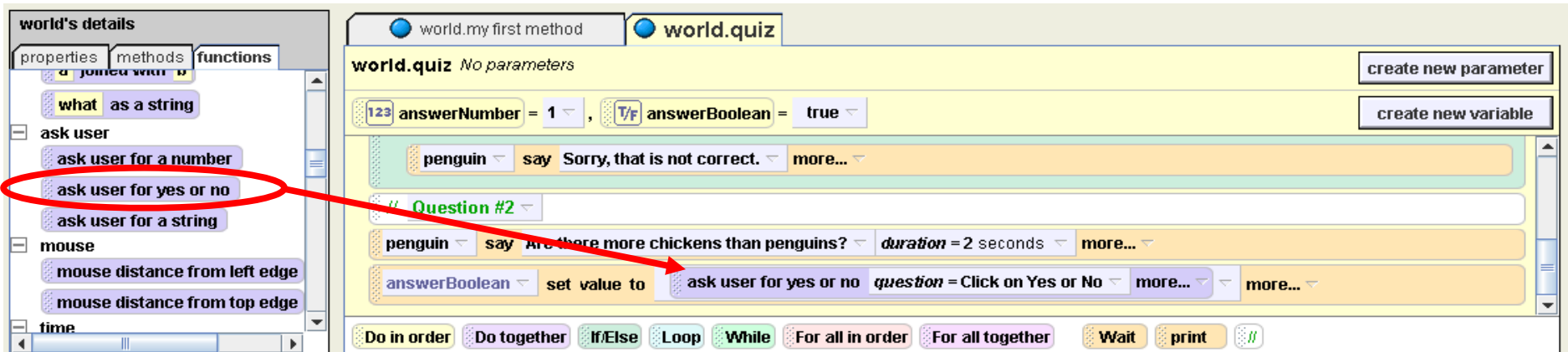


Step 5: Question #2 Continued...

Drag and drop the variable **answerBoolean** into the method and set the value to **true**.



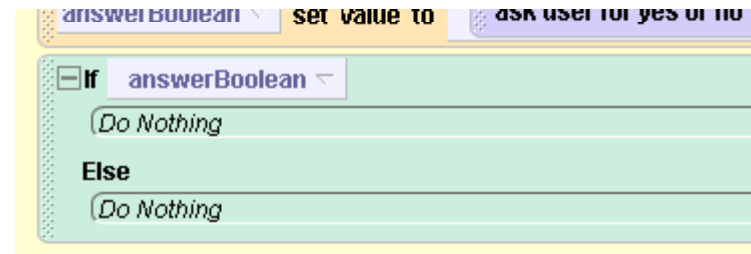
In the world's functions, drag and drop the **ask user for yes or no** over the **true**.
Type in: *Click on Yes or No*.



Step 5: Question #2 Continued...

Now we will add in our responses to the answer in another If/Else statement.

Drag and drop an **If/Else** statement and select **true**.



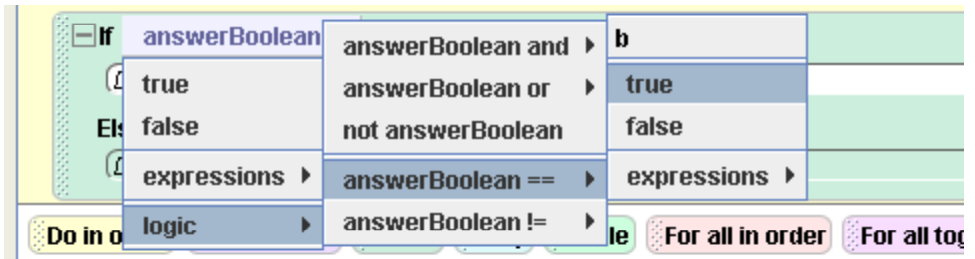
Drag and drop the variable **answerBoolean** over the true.

The If question accepts just the variable because answerBoolean is a boolean type. To make the question explicit we will use the boolean logic functions.

Click on answerBoolean in the If block

Select logic

answerBoolean ==
true



Step 5: Question #2 Continued...

Now complete the question #2 by adding in the methods as shown below:



The image shows a Scratch code editor with an if-else block. The if condition is 'answerBoolean == true'. The if block contains two actions: 'penguin say Correct' and 'penguin turn left 1 revolution'. The else block contains one action: 'penguin say Sorry, that is not correct.'.

```
if (answerBoolean == true) {
  penguin say Correct
  penguin turn left 1 revolution
} else {
  penguin say Sorry, that is not correct.
}
```

Play your world to take the
two question quiz.



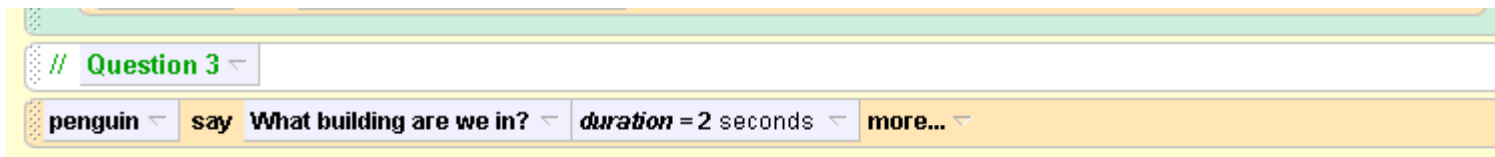
Step 6: Question #3

For our final question we will ask the user to enter a string.

Drop in a comment and type in: *Question 3*

Click on **penguin** in the object tree, have it say: *What building are we in?* and set the duration to 2 seconds.

Note: A string is a set of letters or characters. A space is considered a character so be careful if you have a space at the end of your word or sentence.



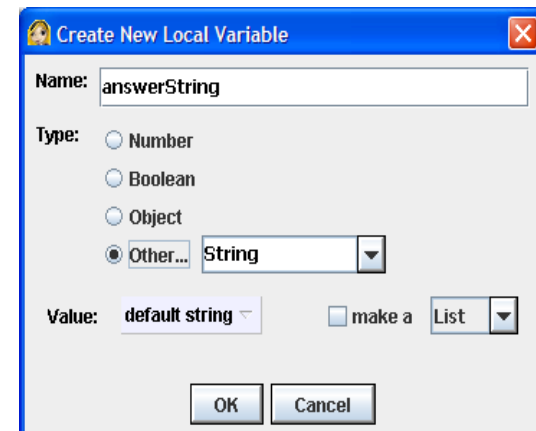
We need a variable to store the string information we get from the user.

Click on **create new variable**.

Name it *answerString*.

Select **Other...** and **String**

Click **OK**.



Step 6: Question #3 Continued...

Drag and drop the variable `answerString` into the method and set the value to `default string`.

In the world's functions, drag and drop the `ask user for a string` over the `default string`.
Type in: *Enter acronym*.

The screenshot shows a programming environment with a left sidebar titled "world's details" containing "properties", "methods", and "functions" tabs. The "functions" tab is active, showing a list of functions including "what as a string", "ask user", and "mouse". The "ask user" function is expanded, showing sub-functions like "ask user for a number", "ask user for yes or no", and "ask user for a string".

The main workspace shows the "world.quiz" function being edited. The function has no parameters. At the top, there are three variable declarations: `answerNumber = 1`, `answerBoolean = true`, and `answerString = default string`. Below these, there is a comment `// Question 3`. The main body of the function contains a "say" block with the text "What building are we in?" and a duration of 2 seconds. Below the "say" block is a "set value to" block where the variable `answerString` is set to the value of the "ask user for a string" block, with the question text "Enter acronym".

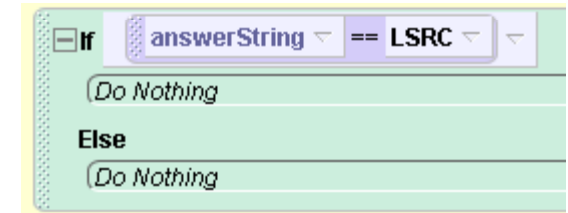
At the bottom of the workspace, there are several control blocks: "Do in order", "Do together", "If/Else", "Loop", "While", "For all in order", "For all together", "Wait", "print", and a comment block.

Step 6: Question #3 Continued...

Now we will add in our responses to the answer in another If/Else statement.

Drag and drop an **If/Else** statement and select **true**.

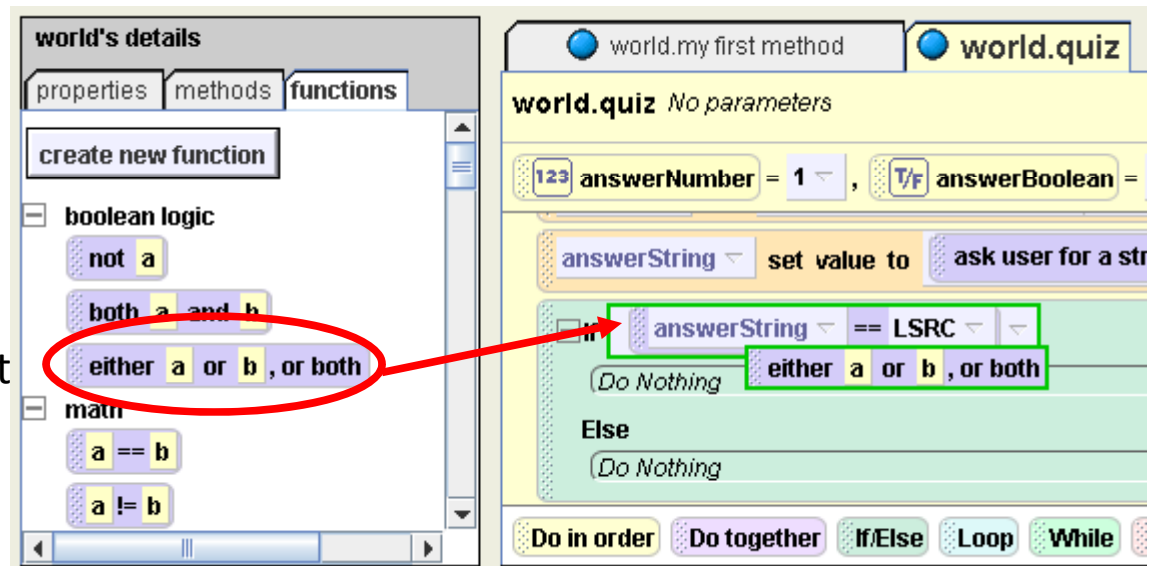
Drag and drop the variable **answerString** over the **true**.
Select **answerString==** then use **other...** to type in: **LSRC**



But wait, what if the user enters in "lsrc"? Alice is case sensitive so we need to allow for the user to enter LSRC or lsrc as the correct answer.

Click on **world** in the object tree and find the boolean logic functions

Drag and drop **either a or b, or both**,
or both onto the If statement
and select **true**



Step 6: Question #3 Continued...

Now drag the variable answerString onto the true
Choose answerString== and other...

The screenshot shows a Scratch script editor for a quiz. The script is titled "world.quiz" and has no parameters. It contains the following code:

```
answerNumber = 1, answerBoolean = true, answerString  
answerString set value to ask user for a string question = Enter acronym  
If either answerString == LSRC or true or both  
  Do Nothing  
Else  
  Do Nothing
```

A red circle highlights the "answerString" variable in the script. A red arrow points from this circle to the "answerString == LSRC" condition in the "If" block. A dropdown menu is open, showing the options "answerString ==", "answerString !=", and "other...".

Type in: lsrc

The dialog box is titled "Enter a string" and has a close button (X) in the top right corner. It contains a label "Enter a string:" followed by a text input field containing the text "lsrc". Below the input field are two buttons: "OK" and "Cancel".

Step 6: Question #3 Continued...

Click on penguin in the object tree and add in the responses for when the answer is correct or incorrect as shown below.

The screenshot displays a code editor with the following logic:

```
if either answerString == LSRC or answerString == lsrc, or both  
  penguin say Correct more...  
  penguin roll left 1 revolution more...  
else  
  penguin say Sorry, that is not correct more...
```

The code is organized into a light green background area. The 'if' statement is on a purple bar, and the 'else' statement is on a light green bar. The actions for the 'if' branch are on orange bars, and the action for the 'else' branch is on a light green bar.