

The Essentials of Alice (Bunny)



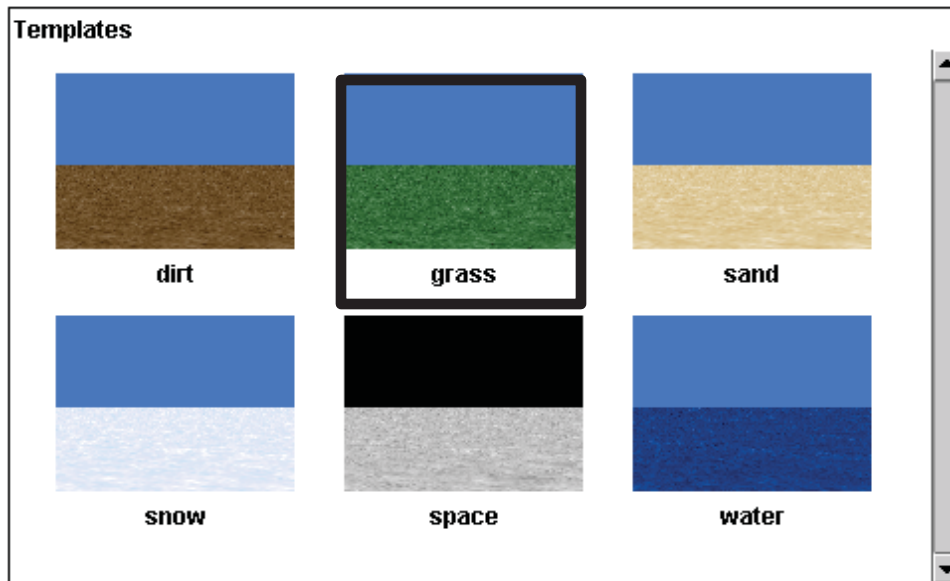
By Jenna Hayes
under the direction of
Professor Susan Rodger
Duke University
July 2008

*This tutorial will teach you how to create a short animation in an Alice world.
Follow the steps and use the pictures to help find things on your screen.*

Step 1: Choosing a Background

When you open Alice, the first thing you must do is choose a background for your animation. You have six choices.

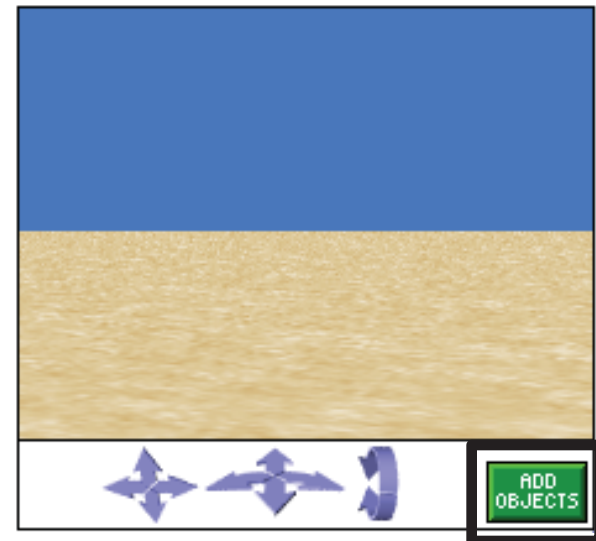
For this tutorial, choose the **grass** background, and then click **Open**.



Step 2: Object Library

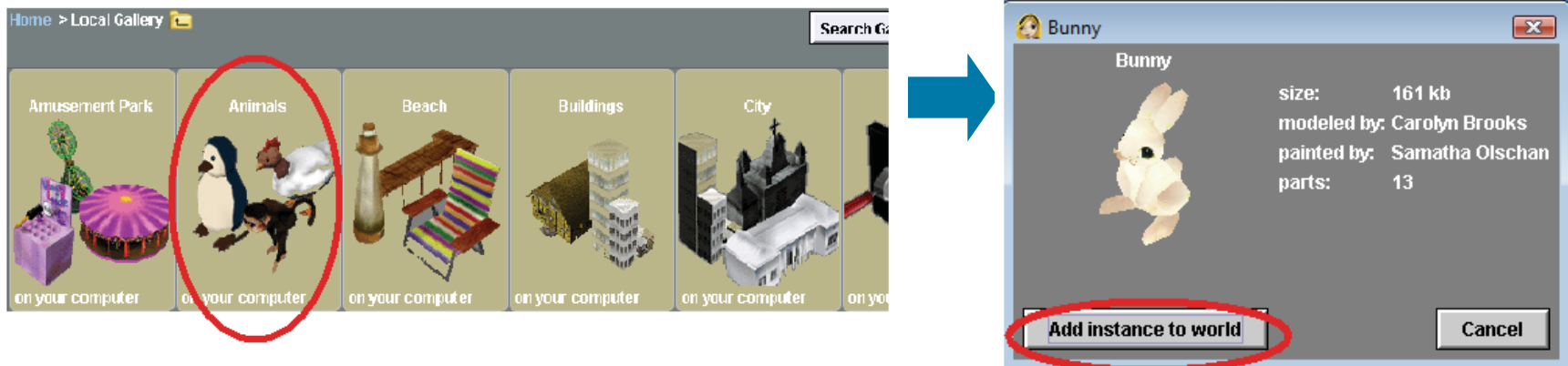
Next, add an **object** to your Alice World. Alice is full of different kinds of **objects** to add to your world.

Click on the green **Add Objects** button:



Step 3: Adding the Object

Click on the **animals** folder of objects. Find the **bunny** and click on it. Then click **Add Instance to World**. The **bunny object** will appear in your world.

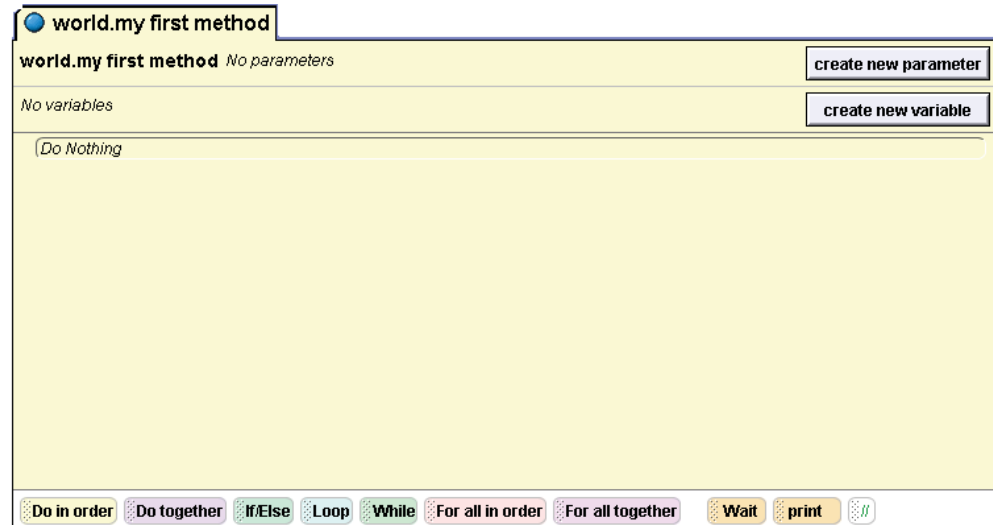


Click the green **Done** button to the right to begin working with your new bunny.



Step 4: Finding the methods

On your Alice screen, you should see your **method editor**. This is the area where you will be telling your bunny what to do. It looks like this:



A **method** is a command you can tell to your bunny. Your bunny already knows certain commands.

Find the **object tree** that lists the objects in your world, in the left hand corner of your screen, and click on **bunny**.

Step 5: Adding Methods

Below the **object tree**, the methods, or commands, that your bunny already knows will pop up:

The screenshot shows a software interface for creating methods for an object named 'bunny'. On the left, a panel titled 'bunny's details' has tabs for 'properties', 'methods', and 'functions'. Below these tabs is a 'create new method' button and a list of methods: 'bunny move', 'bunny turn', 'bunny roll', 'bunny resize', 'bunny say', 'bunny think', 'bunny play sound', 'bunny move to', and 'bunny move toward'. On the right, a method editor window titled 'world.my first method' is open. It shows the method name 'world.my first method' with 'No parameters' and 'No variables'. The main area contains a 'Do Nothing' block. A 'Do in order' block is being dragged from a palette at the bottom into the editor. The palette includes 'Do in order', 'Do together', 'If/Else', 'Loop', 'While', 'For all in order', 'For all together', 'Wait', 'print', and 'If'.

Find the **Do in order** button at the bottom of the method editor, and drag and drop it into the method editor. Now, when we drag and drop methods inside this **Do in order**, they will be carried out one by one, in the order that they are listed.

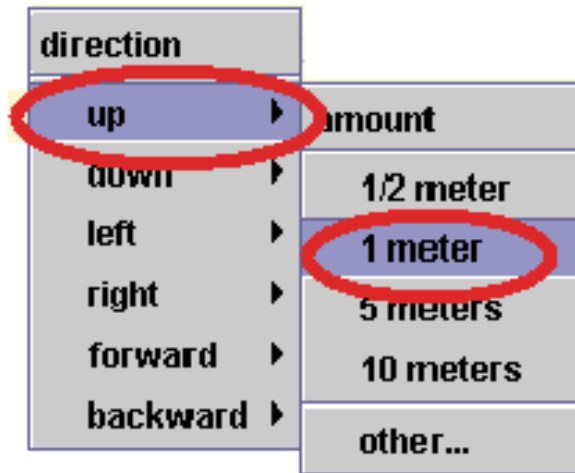
Step 5 (continued): Adding Methods

Click on the **move** method in the list of **methods** under the **object tree**, drag it across to the method editor, and drop it on top of your **Do in order** command:

The image shows a software interface for creating a method. On the left, a panel titled "bunny's details" has tabs for "properties", "methods", and "functions". The "methods" tab is active, showing a list of methods: "bunny move", "bunny turn", "bunny roll", "bunny resize", "bunny say", "bunny think", "bunny play sound", "bunny move to", and "bunny move toward". A "create new method" button is also present. A green arrow points from the "bunny move" block to the right. On the right, a "world.my first method" editor is shown. It has a title bar "world.my first method" and a subtitle "world.my first method No parameters". Below this, it says "No variables". There are buttons for "create new parameter" and "create new variable". The main area contains a "Do in order" block with a sub-block "bunny move". At the bottom, there is a palette of 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: Method Specifics

When you release the **move** method, you should see some options for your method--what **direction** you want the bunny to move in, and how far you want it to move.

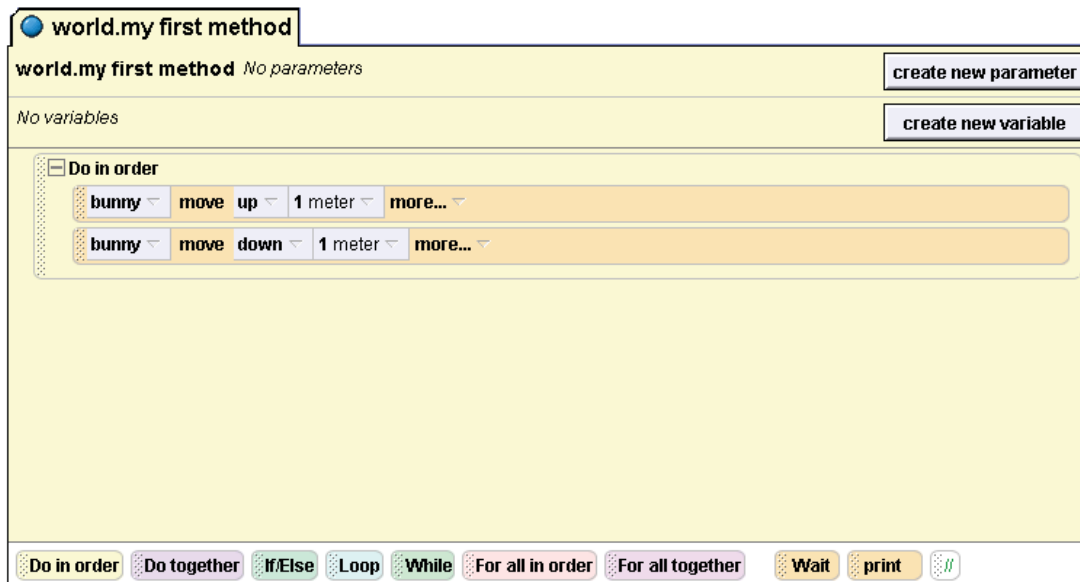


Since we want to make the bunny move **up** first, move your mouse over the **up** direction, and then click on **1 meter**:

Your first command to your bunny is now complete.

Step 6: Finishing the Method

To make your bunny **move** back **down** from its jump, right click on your **move** command and select **copy**. Your **move** command will be copied right beneath where you dropped it. Click on the small down arrow next to the second **up** change it to **down**. Your method editor should look like this:



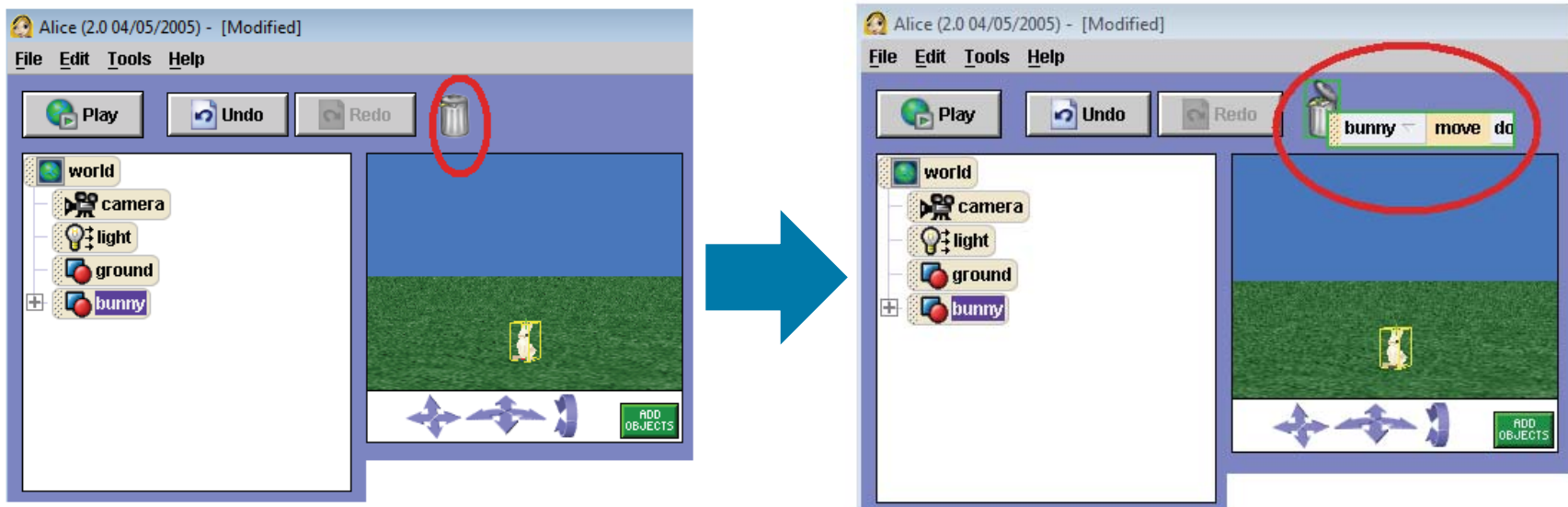
Find the **play** button in the upper left hand corner of your screen, and click it to watch your bunny jump!



Step 7: Editing your methods

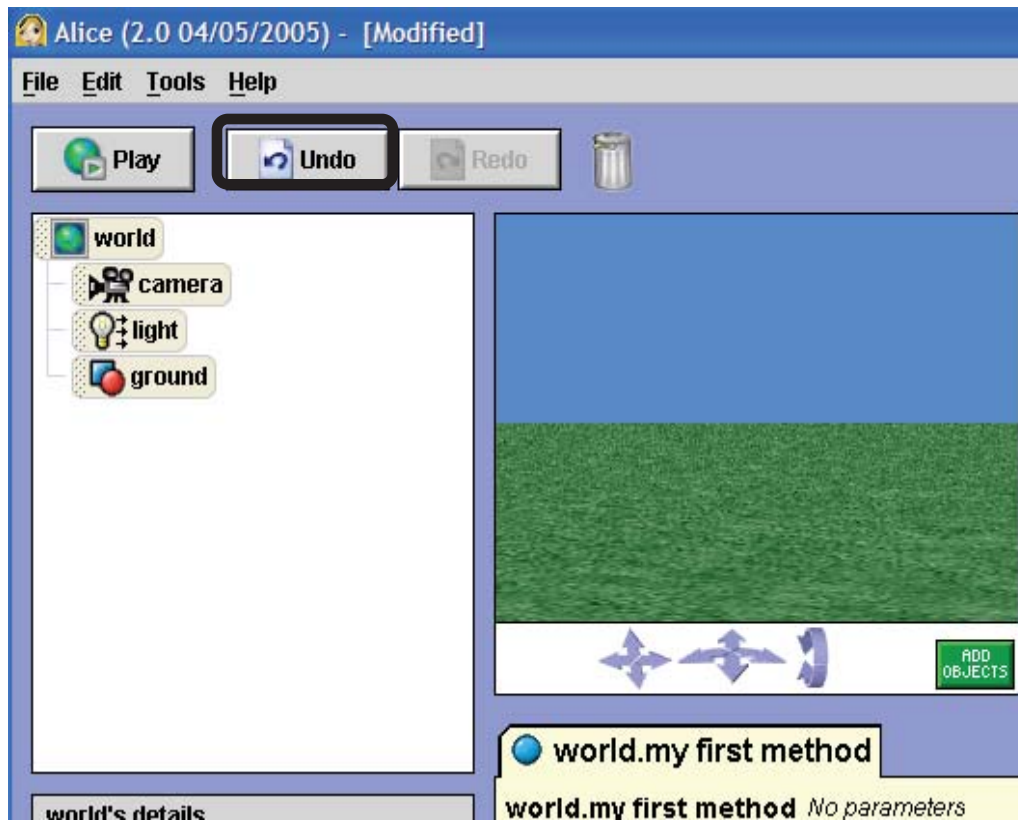
Now you'll learn how to get rid of methods you don't want anymore. Let's say you don't want the bunny to move down anymore.

Click on the word **move** on your **bunny move down** command and drag the command up to the **trash can** in the upper left part of your window. When the outline around the trash can and the command turns green, you can drop the command in the trash can to delete it.



Step 7 (continued): Editing your methods

If you have deleted something or done something wrong, and you want to erase what you have just done, you can click the **Undo** button in the upper left hand corner of your screen. This button is a lifesaver in many situations.



Try clicking it to get back the **bunny move** command you just deleted.

Step 8: Doing two methods at once

If you want to make your **bunny** do more than one thing at once, you use the **Do Together** button. Let's say we want to make the bunny wag it's ears and say " I love Alice!" at the same time.

Find the **Do together** button at the bottom of your method editor and drag and drop it under your **Do in order** that's already there.

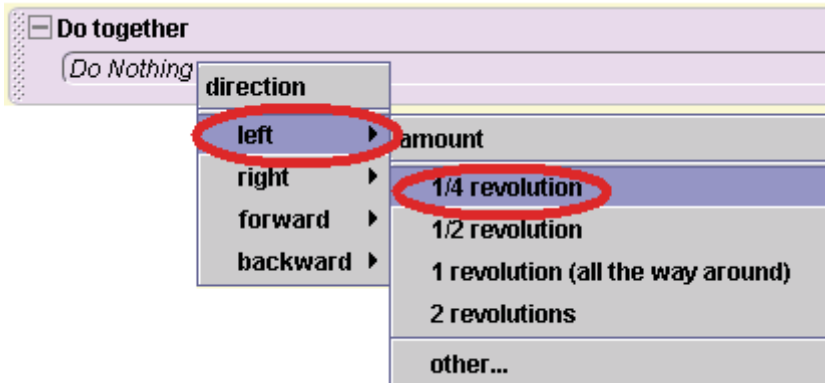
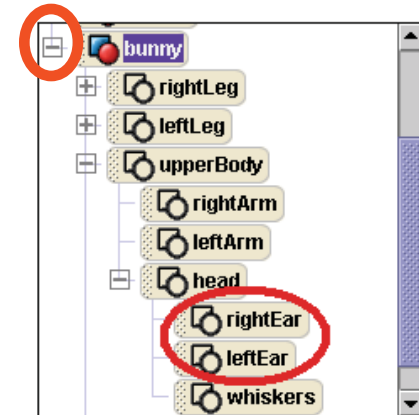
The screenshot shows a Scratch method editor for a method named "world.my first method". The editor has a yellow background and contains the following elements:

- Method name: world.my first method (No parameters)
- Buttons: create new parameter, create new variable
- Variables: No variables
- Block structure:
 - A "Do in order" block containing two "bunny" blocks:
 - Block 1: bunny move up 1 meter more...
 - Block 2: bunny move down 1 meter more...

The bottom toolbar contains the following buttons: Do in order, Do together (circled in red), If/Else, Loop, While, For all in order, For all together, Wait, print, and a comment icon. A red arrow points from the "Do together" button to the "Do in order" block.

Step 9: Moving a Specific Part

To command just the bunny's ears to move, click the plus sign next to **bunny** on the object tree. You should see more parts of the bunny. Then click on the plus sign next to **upperBody**, and then again next to **head**. Now you should see the bunny's ears.



To make the bunny wag its ears, we'll use the **turn** method, and tell the ears to turn. To see the methods that the bunny's ears know, click on **rightEar** and they'll show up in the bottom left corner of your screen. Choose the **turn** method, and drag it into your **Do Together** command in the method editor. To make the ear turn out, select **left** for the direction, and then select **1/4 Revolution** to make the ear turn $\frac{1}{4}$ of a complete circle to the left.

Step 9 (continued): Moving a Specific Part

Now click on the **leftEar**, and repeat the process, except instead of asking the ear to turn **left**, ask it to turn **right**.

The image shows a Scratch code editor window for a method named "world.my first method". The method has no parameters and no variables. The code is organized into two main blocks:

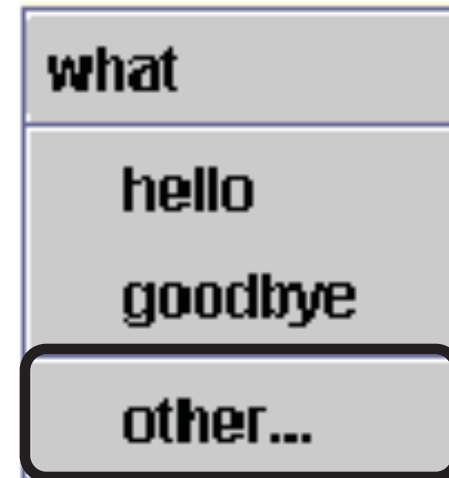
- Do in order:** Contains two "move" blocks. The first block moves "bunny" up by 1 meter. The second block moves "bunny" down by 1 meter.
- Do together:** Contains two "turn" blocks. The first block turns "bunny.upperBody.head.rightEar" left by 0.25 revolutions. The second block turns "bunny.upperBody.head.leftEar" right by 0.25 revolutions.

At the bottom of the editor, there is a palette of code blocks including "Do in order", "Do together", "If/Else", "Loop", "While", "For all in order", "For all together", "Wait", "print", and "//".

Step 10: Talking

Now that your bunny has proper ear wagging action, make it talk. Click on **bunny** in the object tree to display the bunny's methods. Then click on **bunny Say** and drag it into your **Do Together** command under your other methods. This small menu will appear:

Click on **other** to enter in your own text. In the text box that pops up, type **I love Alice!**



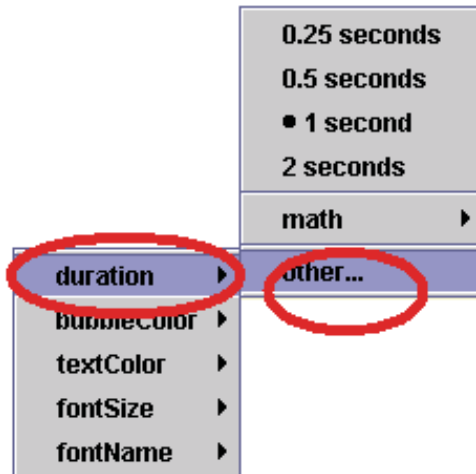
Press the **play** button to see what your world looks like now.

Step 11: Adjusting the Timing

You may have noticed when you played your world that the bunny's speech bubble appears and then disappears very quickly, almost too quickly to read. There is a way to fix this!



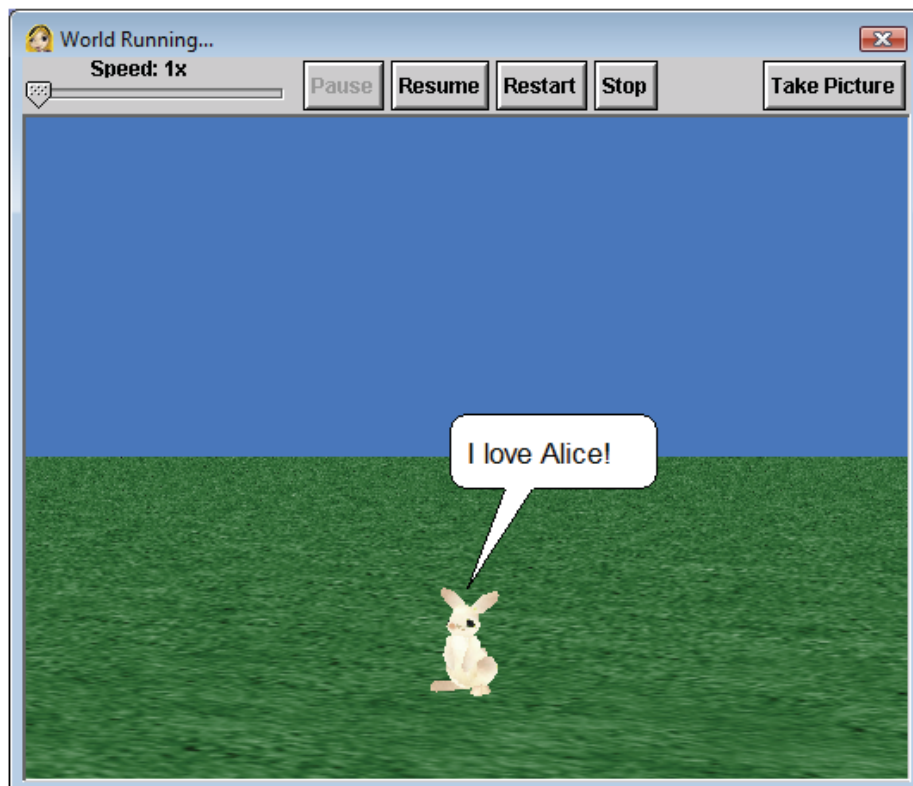
Look at the line in your method editor that commands your bunny to speak. On that line, click the word **more...** next to the command.



Click **duration** on the small menu that appears. You can see that **1 second** is already selected, which is how long the speech bubble appears for. We want to make it longer, so click on **other**, and then enter in **3** on the calculator that appears.

Step 12: Your Turn!

Now **play** your Alice world. You can actually read the speech bubble now! Congratulations on finishing your first Alice world!



These are only the very basics of what you can do with Alice. Try out your bunny's other methods, and see what you can make him do!