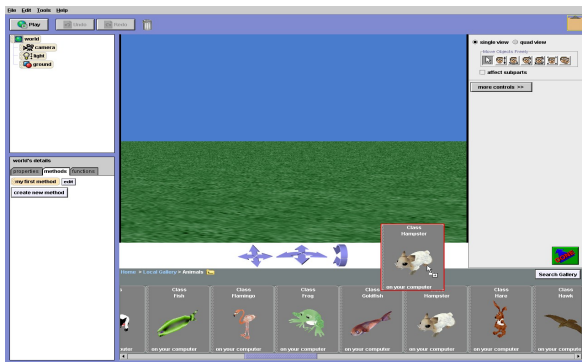


My First ALICE World

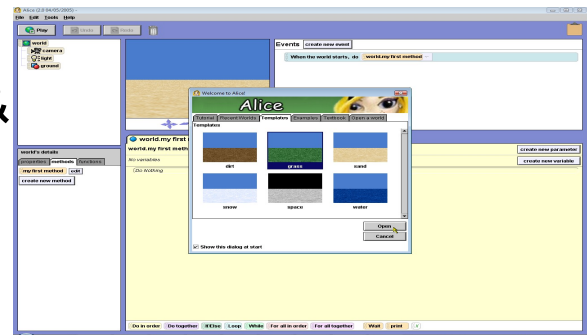
The Hamster Dance

PART 1: Set Up Scene

1. Start a new world. (grass & sky)



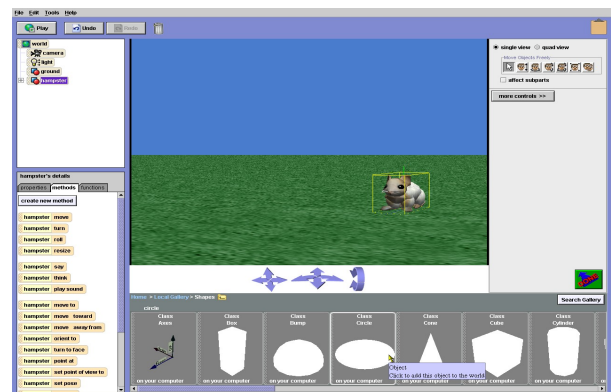
2.



ADD OBJECTS Add a hamster (animals)

3. She's awfully small; let's make her bigger (10X)

4. Add a circle (shapes). This will be a hole. It seems covered by grass, let's move it up a bit, and change it's colour to black so it looks like a hole.




5. If you are asked to save your world, do so now. Pick a name that will make sense to you later, like maybe "HamsterDance"



ALICE Tutorial



Now our scene is set up. Click on the  button.

Now what we want to do is have her dance. To do that we need to pose her into several different positions.



Let's start by “capturing” her original pose. We'll call that “original”.



Next we'll have her move her arms and legs. To do that we'll stand her up (turn backward $\frac{1}{4}$ revolution). Now she's in the ground so we should move her up (which is actually forward). 1 meter is good.

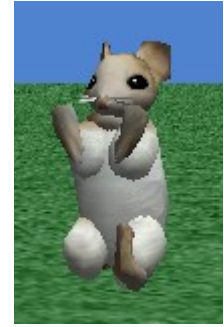
Now we'll move her head (forward $\frac{1}{4}$), arms and legs into a new position and capture that. Remember to select the whole hamster again before capturing the new pose.

Let's try putting her back into her original pose. Notice she doesn't go back to her original POSITION, just her **pose**.



ALICE Tutorial

Now we make up a second pose and capture that.

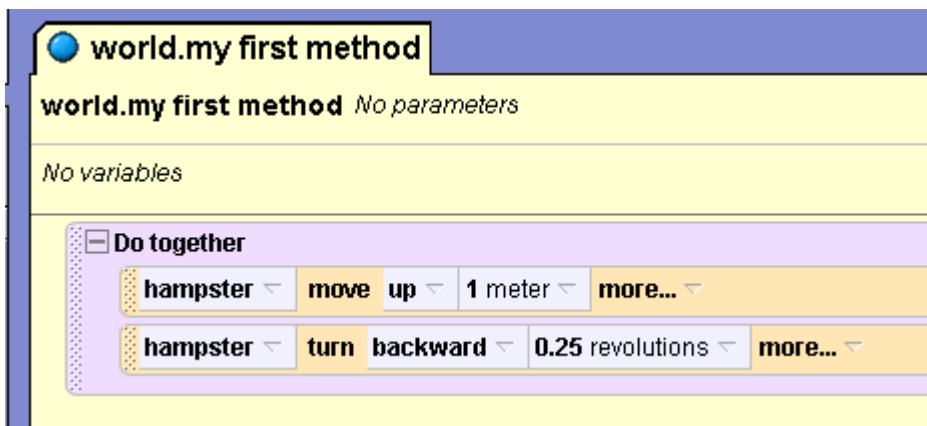


Now we are ready to start programming the movements.

We need to put her back where she was at the start. Set her pose back and move her down again (turn forward $\frac{1}{4}$, then down 1 meter).

Ready.

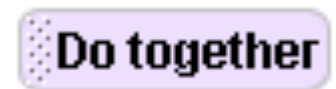
To make her do a dance we will now program some of



the moves we've been doing during the set-up. Drag the turn and move tiles into the *method*.

Let's see what happens if we click on 

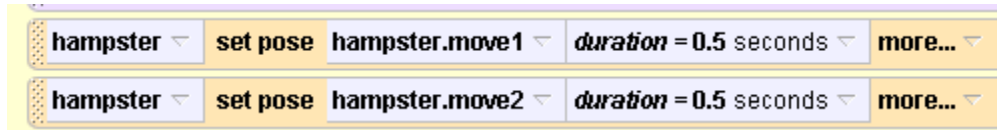
This is OK, but if we make the two things happen *together*, it will look better.





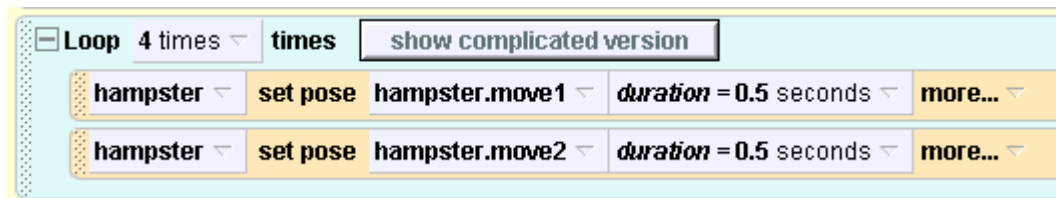
ALICE Tutorial

Next, make her dance! All we need to do is set her poses, by dragging the pose tiles into the method. Try playing her again....



We

could get her to dance for a while by dragging poses into the method repeatedly, but there's an easier way: a **loop**. We'll drag in the loop; drag the set pose tiles into the loop, and see what we've made. Let's have her do it a bunch of times (say, 4). AND then let's have her do it faster.



She ends up in the

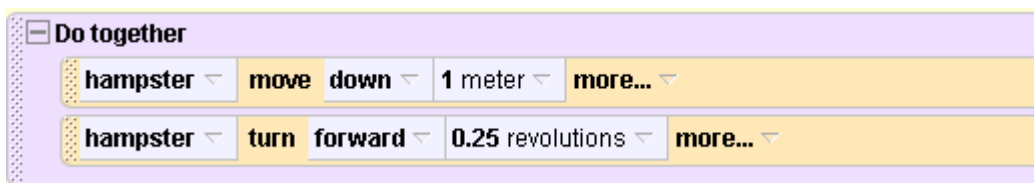
last pose, which looks a little silly, so let's pose her back into the first one after we've done looping.



Great.

Let's put

her back into her original pose and position. This is going to be the opposite of the moves we did at the start, so let's just copy and modify them.





ALICE Tutorial

To keep our main method tidy and small, we're going to put what we've done so far into a new method we'll call "doTheDance". Now instead of all that code, we replace it with a **call** to our new method.

The screenshot shows the ALICE IDE interface for the 'world.doTheDance' method. The editor title is 'world.doTheDance No parameters'. Below the title, it says 'No variables'. The code is organized into several blocks:

- Do together** (purple background):
 - hampster move up 1 meter more...
 - hampster turn backward 0.25 revolutions more...
- Loop 4 times** (cyan background):
 - hampster set pose hampster.move1 duration = 0.5 seconds more...
 - hampster set pose hampster.move2 duration = 0.5 seconds more...
- Do in order** (yellow background):
 - hampster set pose hampster.move1 more...
 - hampster set pose hampster.original more...
- Do together** (purple background):
 - hampster move down 1 meter more...
 - hampster turn forward 0.25 revolutions more...

The screenshot shows the ALICE IDE interface for the 'world.my first method' editor. The editor title is 'world.my first method No parameters'. Below the title, it says 'No variables'. The code block contains a call to the 'world.doTheDance' method.

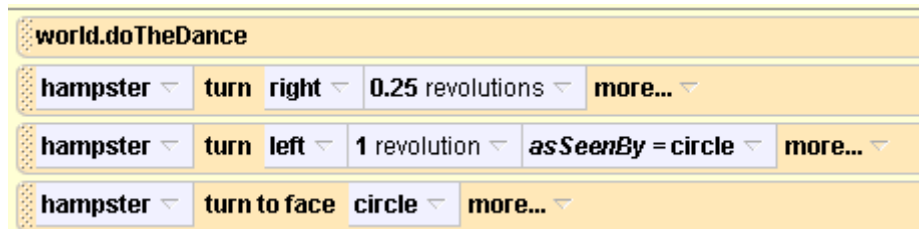


ALICE Tutorial

Now for the finale:

For her finale, she is going to 'run' around the hole and then jump into it. Here's how we do that:

Getting her to go around the hole involves moving forward AND turning at the same time. We *could* do some calculations to figure out a trajectory, but there's an easier way – we simply turn her from the perspective of the hole using **as seen by**. It works, but she is facing wrong so we need to fix that. Let's try **turn right $\frac{1}{4}$** .



That seems to work.

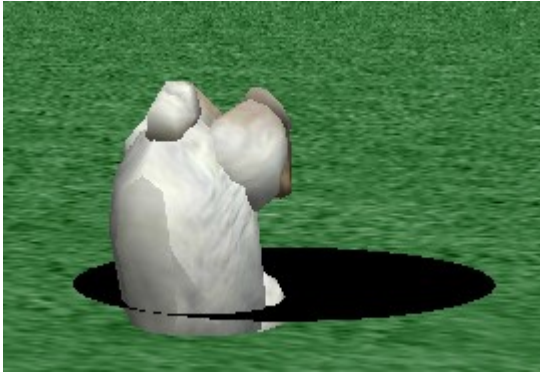
Last: She needs to face the hole and jump in. We'll do this in several moves:

1. **turn to face** the hole
2. Make her jump in the air (turn, then move and turn up, then move and turn down)
3. To get her into the hole, she will have to move down farther than she moved up.



ALICE Tutorial

Not bad, but she didn't go all the way down and the hole seems a little off center. Let's fix that.



world.my first method world.doTheDance

world.my first method *No parameters*

No variables

world.doTheDance

- hamster turn right 0.25 revolutions more...
- hamster turn left 1 revolution asSeenBy = circle more...
- hamster turn to face circle more...
- // jump
- Do together
 - hamster turn backward 0.25 revolutions duration = 0.25 seconds more...
 - hamster move forward 1 meter duration = 0.25 seconds more...
- Do together
 - hamster turn forward 0.5 revolutions duration = 0.25 seconds more...
 - hamster move forward 5 meters duration = 0.25 seconds more...
- hamster move forward 5 meters duration = 0.25 seconds more...



ALICE Tutorial

TA-DA!!!

