

Lesson 16: Animating a Simple Story

Shapes review: Open a new file. Using Logo commands and repeats, draw the following:

- a square
- a bigger square
- a rectangle
- an octagon
- a triangle
- a circle
- a bigger circle.

Animations review: This is how an animation works: first set your shape, then move forward, then wait. Repeat that aloud three times.

Create your first animation line of code using this formula and the shape inch1.

```
setshape "inch1 fd 3 wait 500
```

Now make two more lines, for the shapes inch2 and inch3. Create a procedure **to inch** that holds this animation code.

Let's set up the animation. We need to put the first shape on the animal, lock the shape, and set its heading to 90. Then pull the pen up. Now repeat the procedure **inch** a number of times. Make this into a procedure called **inchwalk**. Test it all by typing **inchwalk** into the Listener window.

Work on it for 10 minutes. You can do it! Check the previous chapter if you need to.

A Story

Here's our assignment for today. We are going to animate a simple story. We want a tiny dog to walk up to a giant moose and say "Bark bark!" Then the silly moose turns around and runs away.

One tool we need is use of **turtletext**, which we can shorten to **tt**. That is where we can write what the characters say, sort of like the balloons over the heads of cartoon characters. As long as the pen is down, this command stamps whatever the input word or words are at the location of the turtle.

Let's try this in the Listener window:

pd tt [Bark Bark]



Great! We can tell the turtle to use different fonts and sizes, but we don't want to worry about that right now. This will do. If I drag the turtle away, the text stays there. That's just what I want.

Now "tell" the second turtle to show itself, using **st** or show turtle:

tell 1 st

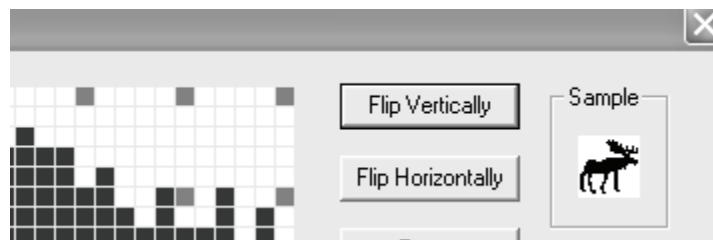
(Remember, the first turtle is Turtle **0**.)

What shapes will we need? We'll need two dog shapes facing to the right, walking up to the moose. We'll need one moose shape facing left, facing the dog. Then when the moose turns around and runs, we will need three moose shapes facing to the right.

As it happens, all we need to do is take the extra moose shape, called simply "moose," and flip it around so it faces left. First, drag the shape called "moose" over to Turtle **0**. How do you know if it is Turtle **0**? Right-click on it and check its properties. Now open the Shape Editor window by clicking on the shape editor icon in the row across the top

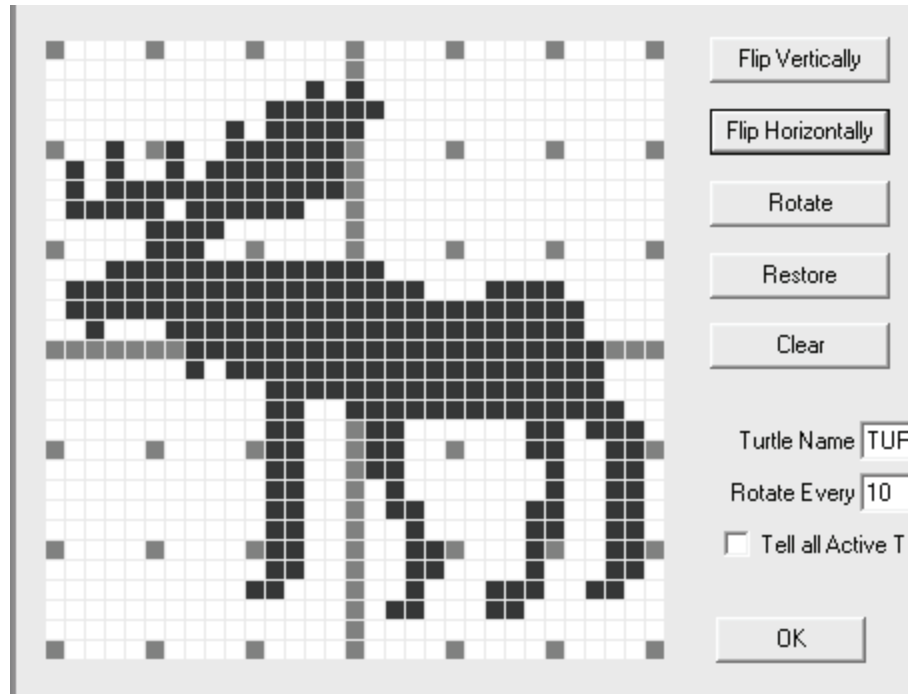


If the wrong turtle shows up in the editor, go back to the Listener window and type **tell 0**. That makes Turtle **0** the active turtle.



Do we flip horizontally or vertically to make it face the other way? Let's try vertically. Seems like that ought to work. Click on Flip Vertically.

Oops! That wasn't it. Click the same button again to set it right. Now try to Flip Horizontally.



That's better. Click OK.

I'd like to get rid of "bark bark" in the middle of my screen right now. I want it to appear only at the proper time. This is a good time to execute `cs` in the Listener window, to clear the screen.

Now set up your screen. Put the moose in the center of the screen, facing left. Drag Turtle **1** to the bottom, left of center.

We need to increase the size of the moose, Turtle **0**, using `setturtle` or `setts`.

In the Listener window, create a command to enlarge Turtle **0**:

```
tell 0 setts 5
```

We are addressing Turtle **0** and setting its turtle size to 5, or five times bigger.



It's time to code! Remember, to animate, first set your shape, then move forward, then wait.

You will need to:

1. Create an animation procedure called **dogwalk** that tells the turtle to: set its shape to dog1, take a few steps, wait a bit; then set its shape to dog2, take a few steps, wait a bit.
2. Create an animation procedure called **mooserun** that tells the turtle to: set its shape to moose1, take a number of steps, wait just a little, and then repeat that for the next two shapes.

Now, in the Listener window, you will need to:

1. **tell 1** to put on dog1 shape, lock the shape, set the heading to 90, pen up, and walk forward using **dogwalk** till it gets close to the moose
2. **tell 1** to put its pen down and print "bark bark"
3. **tell 0** to put on moose1 shape, lock the shape, set the heading to 90, pen up, and run forward using **mooserun**.



Do your best. Only look it up in the answer key if you are stuck for at least 10 minutes. Get it working in the Listener window. After that, paste all the code into a new procedure called **to story**. Test it. Save your environment!