Repetition

• We first illustrate simple looping using “count,” when you know how many times you want to repeat a section of code.

• Then we examine looping based on a condition. The loop continues repeating as long as a condition is true, and stops when the condition is no longer true.

• We also write some methods.
Add in Objects

• Create a new world and use the Grass template
• Click Setup Scene
• Add a GoldenMonkey from the Biped class
• Add three bananas from the Props class

Position Objects

• The bananas will be halfway in the ground. With the default handle style and while holding SHIFT down, drag the bananas up so that they are on the ground. Then move them so that your world looks like this:
Create new procedure: biped.hop

• Since our monkey is a biped, we can teach it, and all other bipeds, to hop with just one method!

• Click Edit Code, then:

Name the procedure: “hop”

• You should see the Biped hop tab open now:
Write the procedure “hop”

• Drag up “do together” from the bottom of the pane and put a “do in order” inside it
• In the “do in order” drag in a “this.move” block. Select “up” and 0.5 meter
• Select “add detail” and set the duration to 0.25 sec

Write the procedure “hop” (cont.)

• Make a copy of the “move” command and put it under the first one. Change “up” to “down” in the second one.
Testing Hop

• Click on the `initializeEvent Listeners` tab

• On the dropdown, select “this.goldenMonkey”

Testing Hop (cont.)

• In the “do in order,” drag in a “this.goldenMonkey.hop” block above “this.myFirstMethod” and run the world
Write the procedure “hop” (cont.)

• The monkey moves up, but not forward
• Add into the “do together” – “this.move” select forward 0.3 meters and set the duration to 0.5 sec

Let’s get the monkey to hop in our story

• Click on the “myFirstMethod” tab
• On the dropdown, select “this.goldenMonkey”
• Drag the “hop” procedure into myFirstMethod
Don’t Forget

• Click on *initialize EventListeners*
• Delete the call to hop

Repeating the hop - count

• Click “run” and the monkey hops once
• We’d like to have the monkey hop 4 times
• A “count” allows you to repeat commands a specified number of times
• Drag “count” into “myFirstMethod”, select “custom” and type 4
Monkey hop 4 times

• Drag the hop method inside the count loop and click “run.” The monkey should hop 4 times.

• We could add more items to the loop. Drag in “this.goldenMonkey.say” and type “hop,” select “add detail” and pick duration = 0.5 sec

Count

• The number of times the monkey hops is exact.
• We can change the number to another number, say 3, by clicking on the 4 and changing it to a 3
Now let’s teach the monkey to eat

• Now we will teach the monkey to eat one of the bananas.
• We don’t know how far the banana is, so we will need the monkey to hop over to the banana repeatedly until the monkey is close to it.

Create new procedure “eat”

• Just as before, we will create a biped procedure
Create new procedure “eat” (cont.)

• Type in “eat” as the name
• You should see the “biped.eat” tab

Write “eat”

• We will need a parameter to represent the banana we want the monkey to eat.
• Click “Add Parameter...” On the “value type,” select “Gallery Class...”
Write “eat” (cont.)

• In the middle window, select “Prop”
• This will include the bananas
• Click “OK”
• Name it “food”

Write “eat” (cont.)

• You should see the parameter available for use in the eat tab
Write “eat” (cont.)

• Drag in a “this.turnToFace” block and select our parameter “food”

• To repeat code based on a condition, we will need a “while” loop. Drag one in and select “true”

How a While Loop Works

while <condition>
  <code>

• If the condition is true, the code is executed. If the condition is still true, the code is executed again. This repeats until the condition is false, then the code in the loop is no longer executed
Write “eat” (cont.)

• We want the monkey to keep hopping toward the banana as long as the distance between them is greater than 1 m.

• We will need to change the condition of the while loop.

• Click on “true” and change to “Relational (Decimal Number)” and then greater than. Select both 1s as placeholders.
Write “eat” (cont.)

• Click on the “functions” tab under the biped
• Drag in “this.getDistanceTo” over the first 1 in the while loop condition and select our parameter “food”

Write “eat” (cont.)

• From the “procedures” tab, drag in the “hop” method into the while loop.
Write “eat” (cont.)

• Now let’s add some commands to bend over and eat the banana.

• Add after the while loop, commands for the monkey to turn forward .125, the banana to disappear (set opacity to 0) and the monkey to turn backward.

Biped.eat

• Here is the code now for biped.eat
Now finish the story

• Click on the “myFirstMethod” tab
• After the “count,” put in a “do in order”
• Drag in a “this.goldenMonkey.eat” block 3 times, selecting banana, banana2, and banana3

If you know about lists…

• If you have done the list tutorial, then look at the rest of this tutorial. Otherwise, you are done.
• You could make a list of bananas for the monkey to eat.
Make a List

• Add a scene property
• Make sure to click on the “Scene” tab first

Make a List (cont.)

• Name the list “food”
• Set type to “Prop” and check “is array”
Make a List (cont.)

• Click “custom array” and add the bananas
• Click OK

Now use the list

• In the “myFirstMethod” delete the “do in order”
• Drag up from the bottom “for each in” and select item type “Prop,” name the item “item,” and set the array to “this.food”
• Drag in a “goldenMonkey.eat” block into the for loop and select “item”
EXTRAS

• You could add more bananas (or other props) and add them to the list

• You could have the monkey move other parts when it hops, such as its feet or head