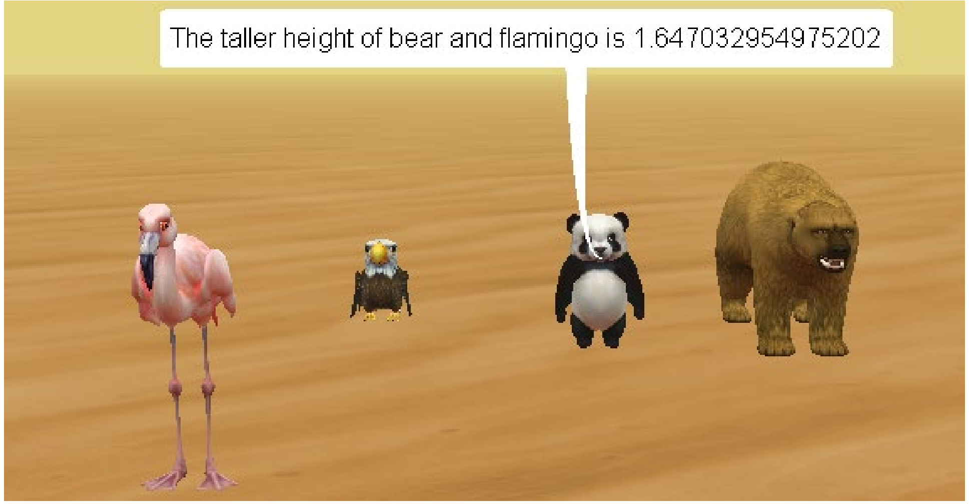
CompSci 94
Writing your own Functions
October 21, 2021



Prof. Susan Rodger

Announcements

- Sakai QZ and videos for Tuesday
- Assignment 4 due Oct 28

Function vs Procedure

- What is the difference between a function and a procedure?

Function vs Procedure

- What is the difference between a function and a procedure?
  - Procedure is something to do – turn, move, dance
  - Function is a calculated value – a number, an object, a direction
  - A function by itself is not very useful, a function has to be used in some way based on the type of value it calculates

Write a function called tallerHeight to compute the height of the tallest of two objects.

- What type of function should it be? Where do you create it?

- What is the return type?

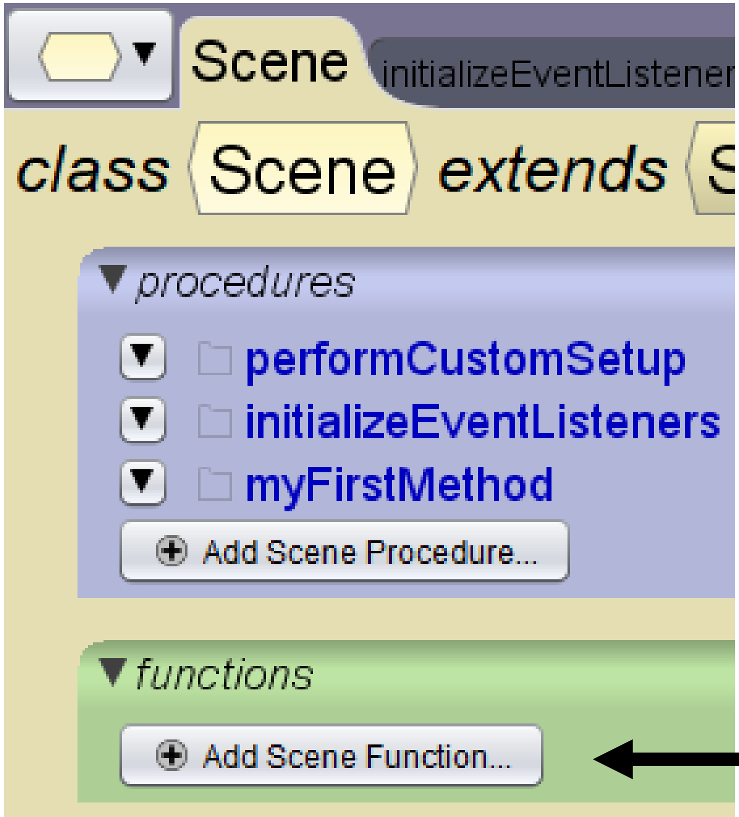
- Need two parameters, what are their types?

Write a function called tallerHeight to compute the height of the tallest of two objects

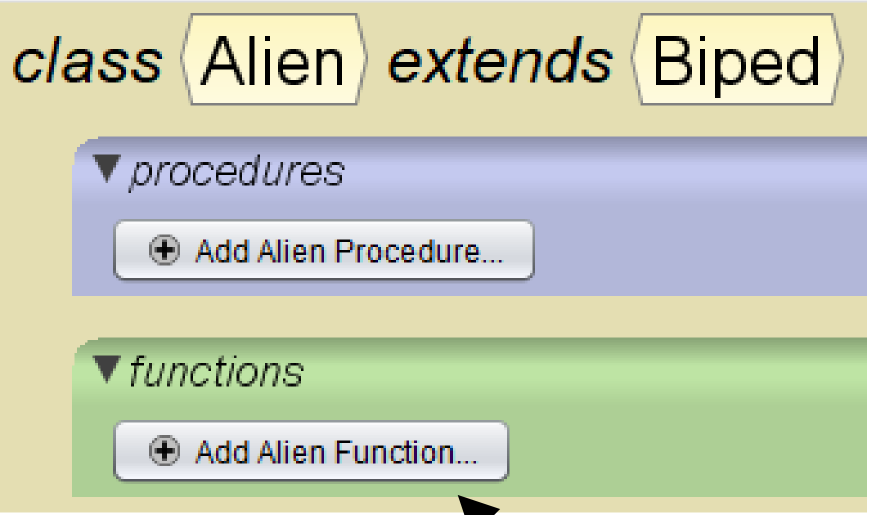
- What type of function should it be? Where do you create it?
  - Scene function
    - Like to be able to use it for any two objects
- What is the return type?
  - DecimalNumber
- Need two parameters, what are their types?
  - SJointedModel
    - Then works for any creatures

Can write your own functions

Function for Scene OR Function for character



```
class Scene extends S
```

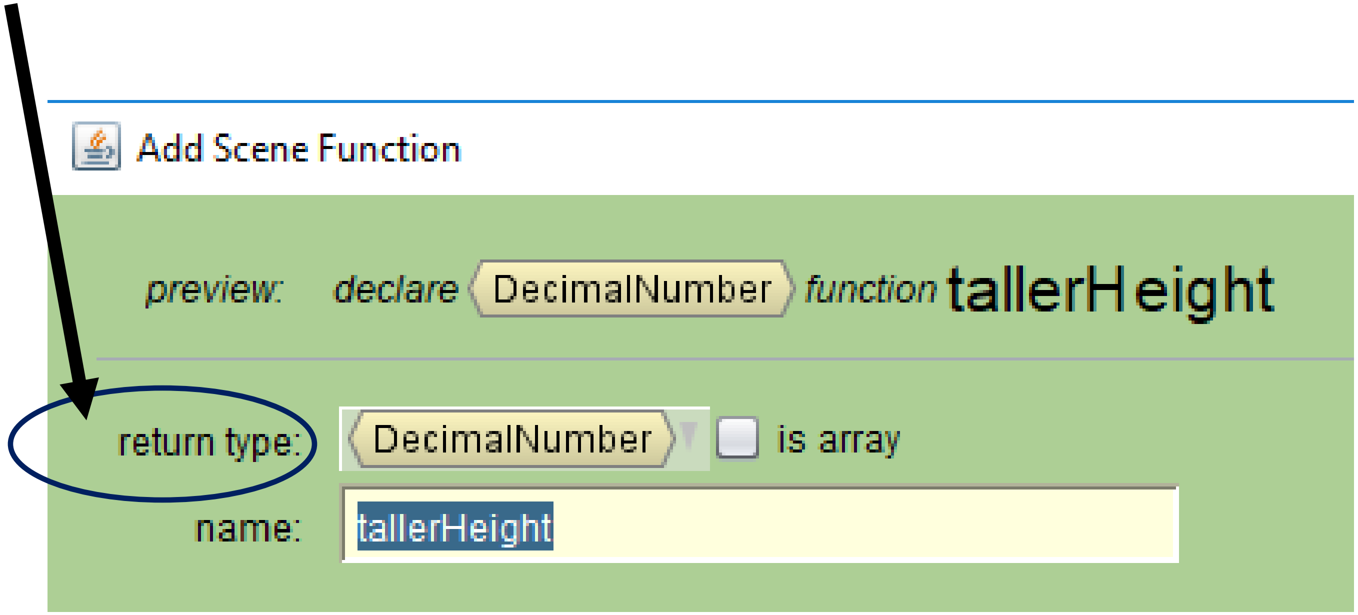


```
class Alien extends Biped
```

Use scene function if it involves multiple objects

Create Scene function tallerHeight

- Inputs: two objects
- Output (return value): the height of the taller object
- Return type: decimalNumber



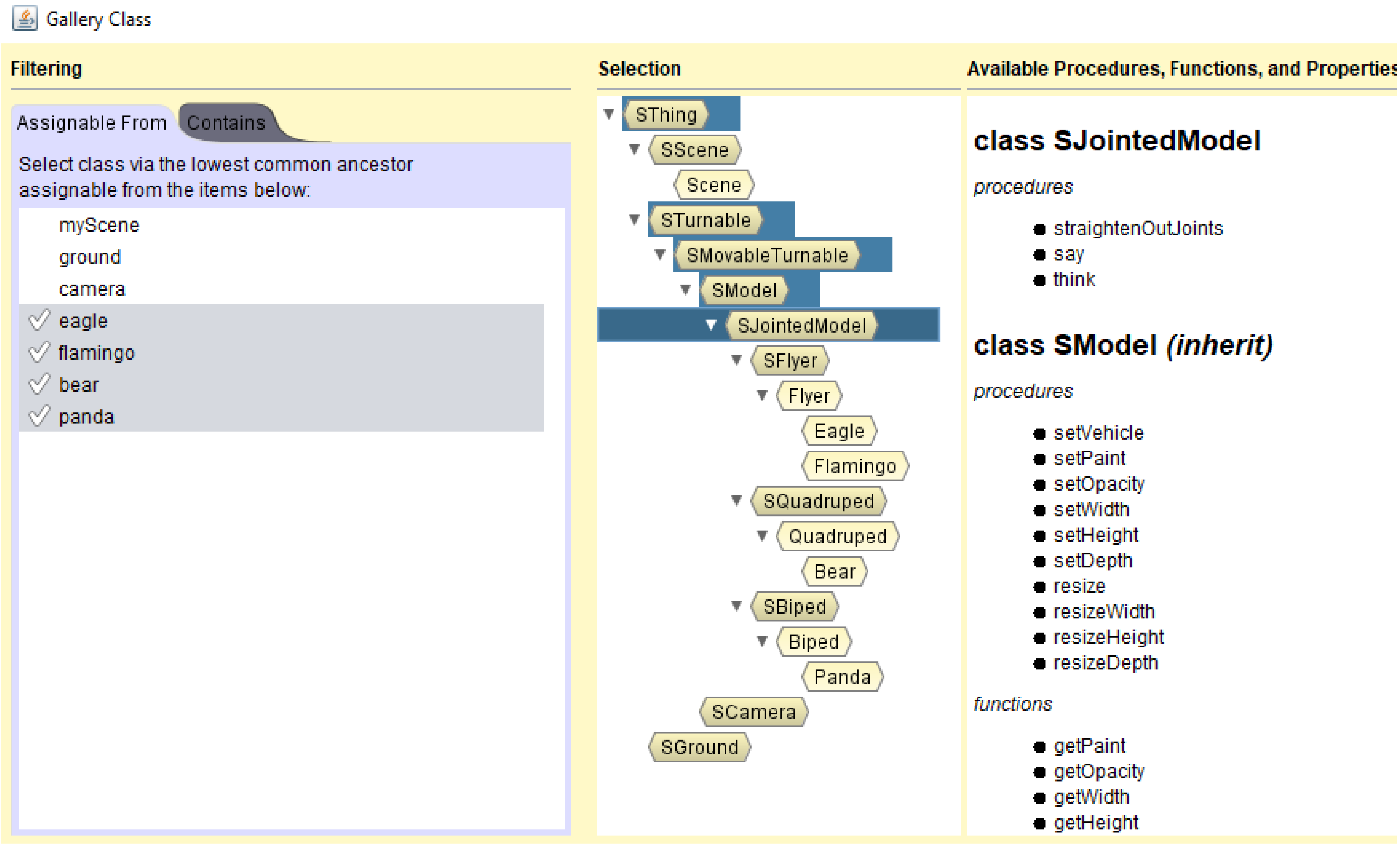
Add Scene Function

preview: declare DecimalNumber function tallerHeight

return type: DecimalNumber ☐ is array

name: tallerHeight

Parameters - SJointedModel



Gallery Class

Filtering

Assignable From | Contains

Select class via the lowest common ancestor assignable from the items below:

- myScene
- ground
- camera
- ✓ eagle
- ✓ flamingo
- ✓ bear
- ✓ panda

Selection

- ▼ SThing
  - ▼ SScene
    - Scene
  - ▼ STurnable
    - ▼ SMovableTurnable
      - ▼ SModel
        - ▼ SJointedModel
          - ▼ SFlyer
            - ▼ Flyer
              - Eagle
              - Flamingo
          - ▼ SQuadruped
            - ▼ Quadruped
              - Bear
          - ▼ SBiped
            - ▼ Biped
              - Panda
        - SCamera
  - SGround

Available Procedures, Functions, and Properties

class SJointedModel

procedures

- straightenOutJoints
- say
- think

class SModel *(inherit)*

procedures

- setVehicle
- setPaint
- setOpacity
- setWidth
- setHeight
- setDepth
- resize
- resizeWidth
- resizeHeight
- resizeDepth

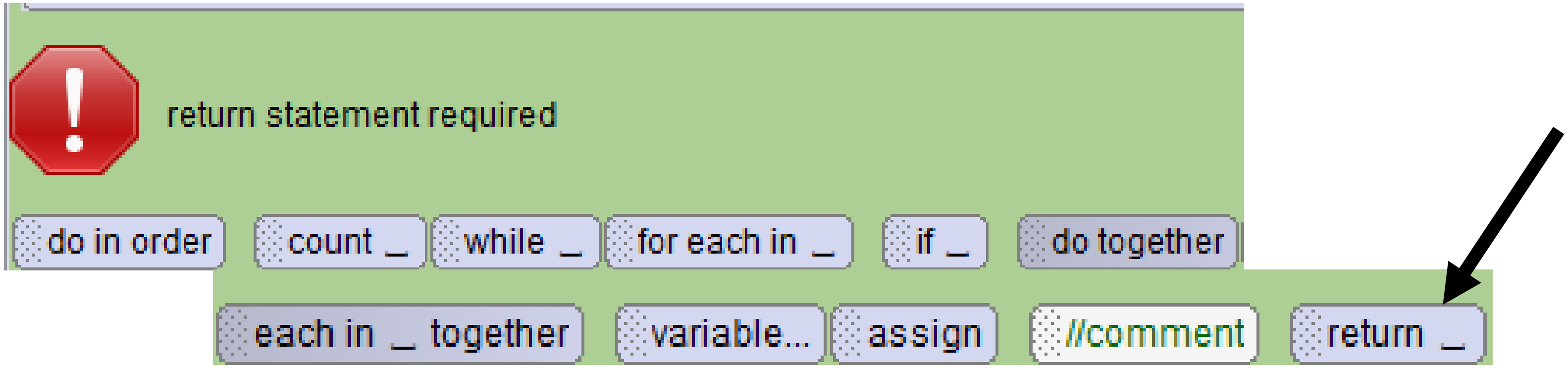
functions

- getPaint
- getOpacity
- getWidth
- getHeight

Q1. What line of code do we have to put in every function?

Q1. What line of code do we have to put in every function?

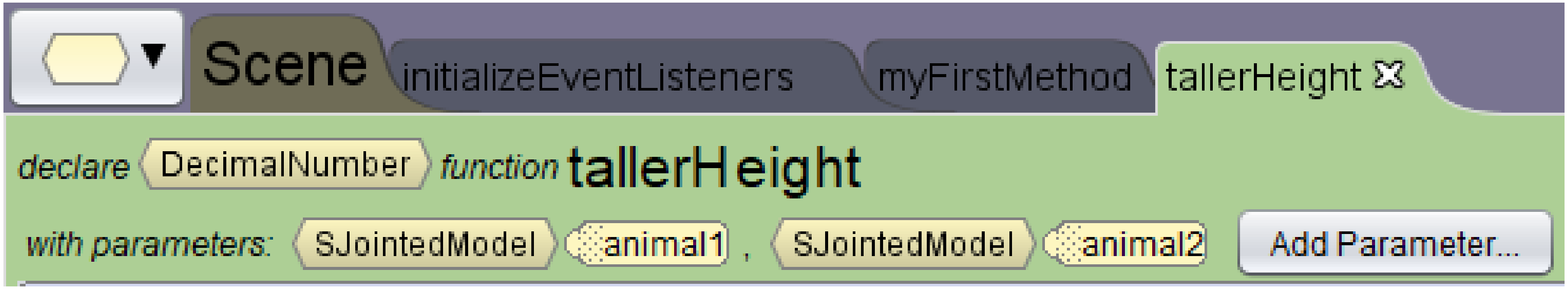
- Return statement!
  - Must return the same type as the specified return value.



return statement required

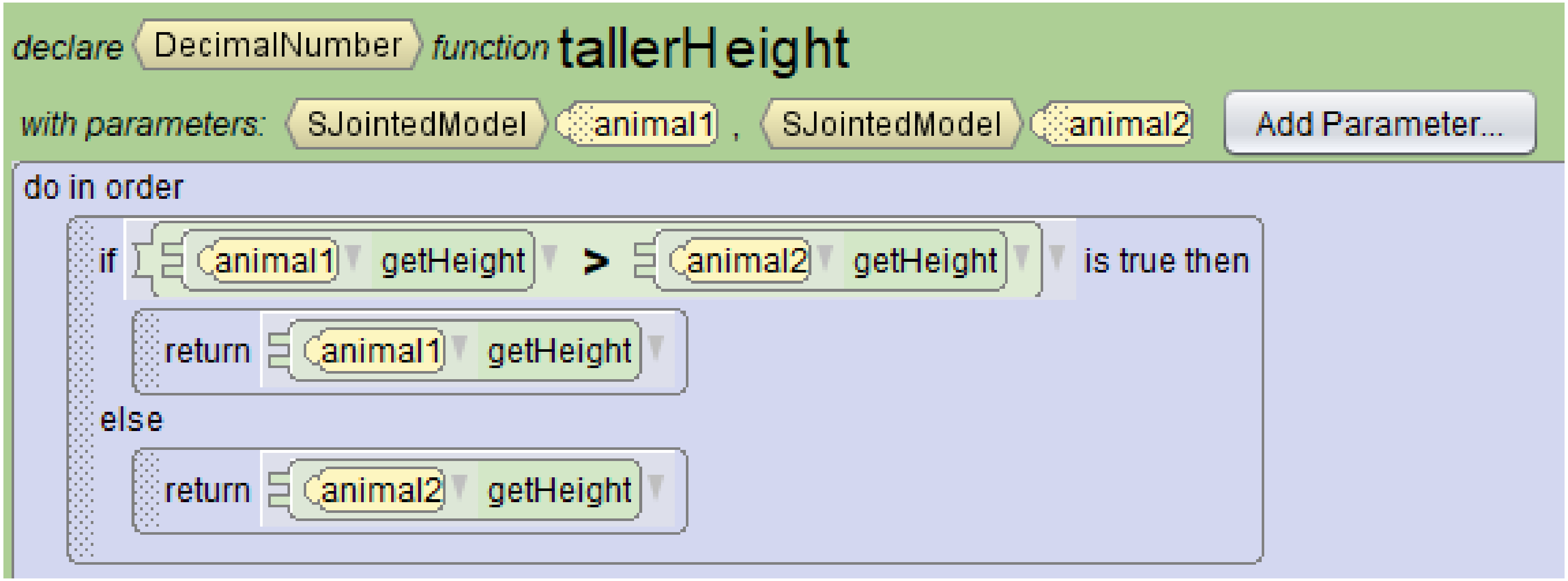
Q2 What is the code for tallerHeight?

Q2 What is the code for tallerHeight?



```
declare DecimalNumber function tallerHeight
with parameters: SJointedModel animal1 , SJointedModel animal2
```

Q2 What is the code for tallerHeight?



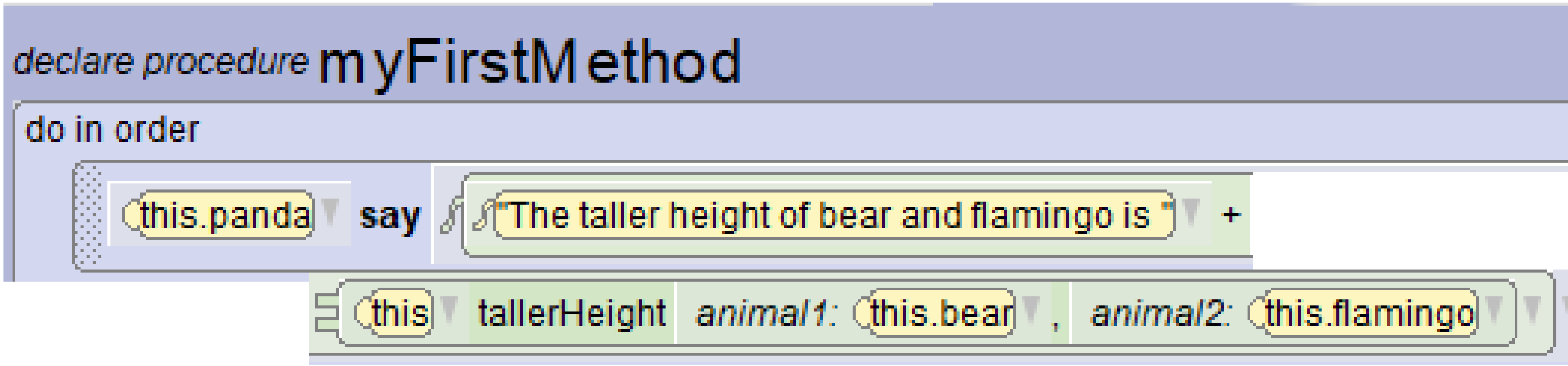
```
declare DecimalNumber function tallerHeight
with parameters: SJointedModel animal1 , SJointedModel animal2   Add Parameter...
do in order
    if animal1 getHeight > animal2 getHeight is true then
        return animal1 getHeight
    else
        return animal2 getHeight
```

Q3 Given a bear and a flamingo, how does one use the function tallerHeight?

- Have panda say what the taller height is of the bear and flamingo.

Q3 Given a bear and a flamingo, how does one use the function tallerHeight?

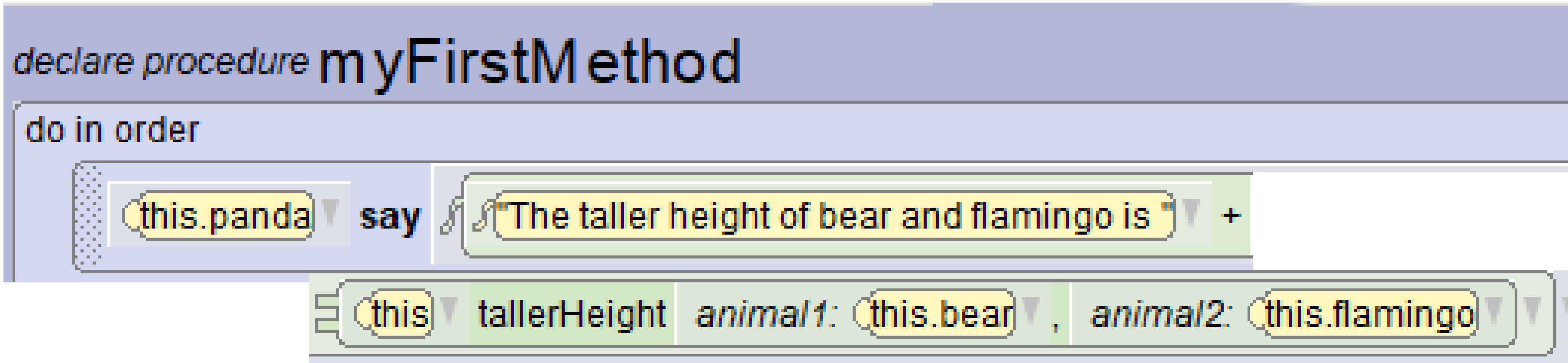
- Have panda say what the taller height is of the bear and flamingo.



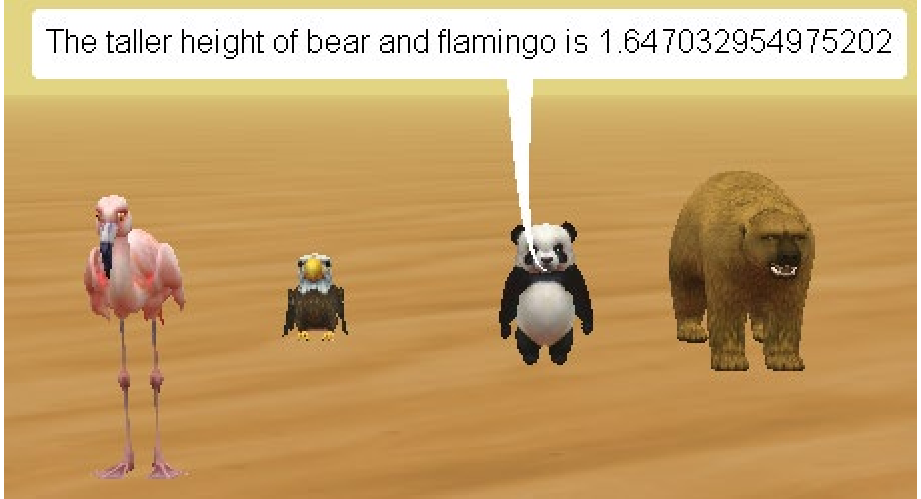
```
declare procedure myFirstMethod
do in order
    this.panda say "The taller height of bear and flamingo is " +
        this tallerHeight animal1: this.bear , animal2: this.flamingo
```

Q3 Given a bear and a flamingo, how does one use the function tallerHeight?

- Have panda say what the taller height is of the bear and flamingo.



```
declare procedure myFirstMethod
do in order
    this.panda say "The taller height of bear and flamingo is " + this tallerHeight animal1: this.bear , animal2: this.flamingo
```



```
The taller height of bear and flamingo is 1.647032954975202
```

Q4. Write a function called tallerObject to return the object who is taller of two objects.

- What type of function should it be? Where do you create it?

- What is the return type?

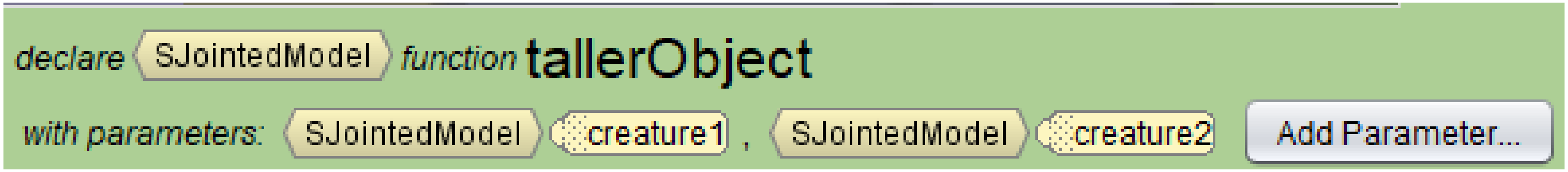
- Need two parameters, what are their types?

Q4. Write a function called tallerObject to return the object who is taller of two objects.

- What type of function should it be? Where do you create it?
  - Scene function
    - Like to be able to use it for any two objects
- What is the return type?
  - SJointedModel
- Need two parameters, what are their types?
  - SJointedModel
    - Then works for any creatures

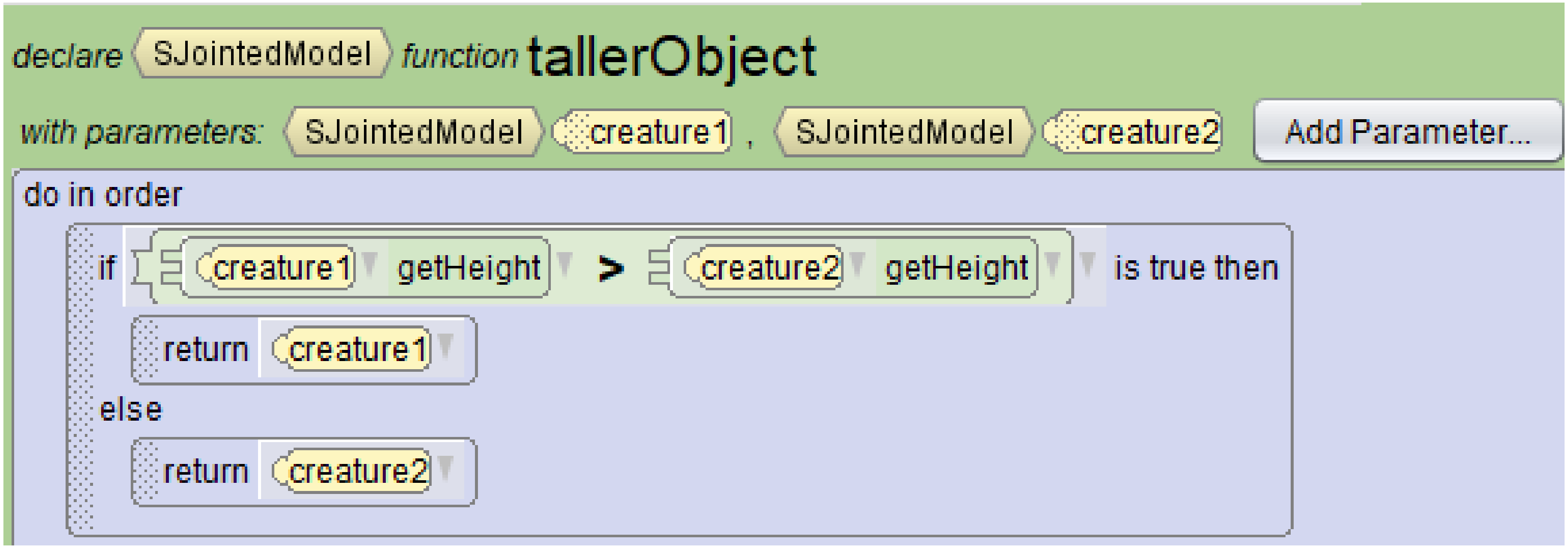
Q5 What is the code for tallerObject?

Q5 What is the code for tallerObject?



```
declare SJointedModel function tallerObject
with parameters: SJointedModel creature1 , SJointedModel creature2
```

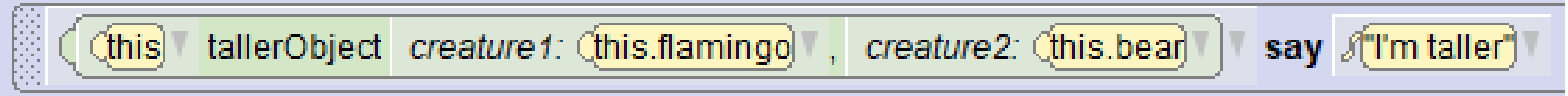
Q5 What is the code for tallerObject?



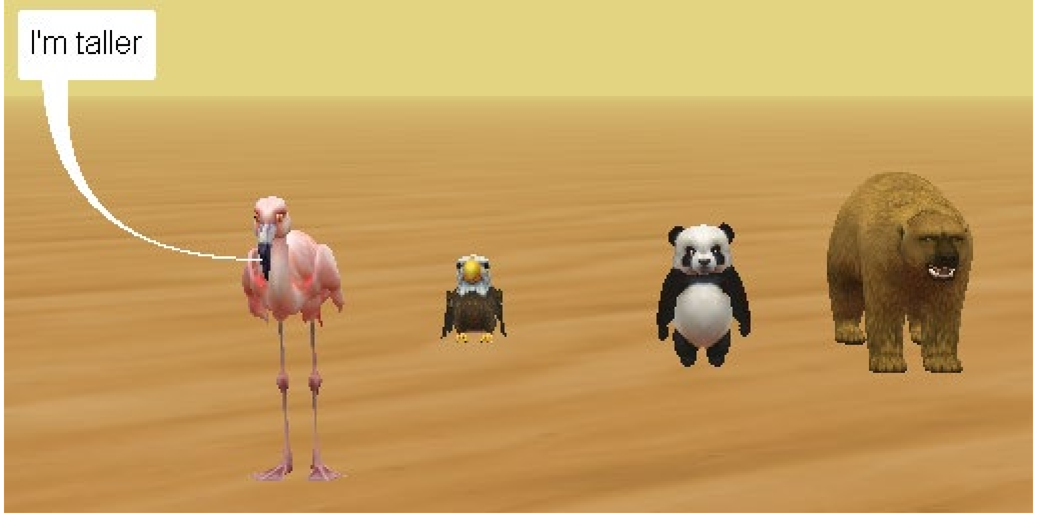
```
declare SJointedModel function tallerObject
with parameters: SJointedModel creature1 , SJointedModel creature2
do in order
    if creature1 getHeight > creature2 getHeight is true then
        return creature1
    else
        return creature2
```

Q6 How do you get the taller of the bear and flamingo to say they are taller?

Q6 How do you get the taller of the bear and flamingo to say they are taller?



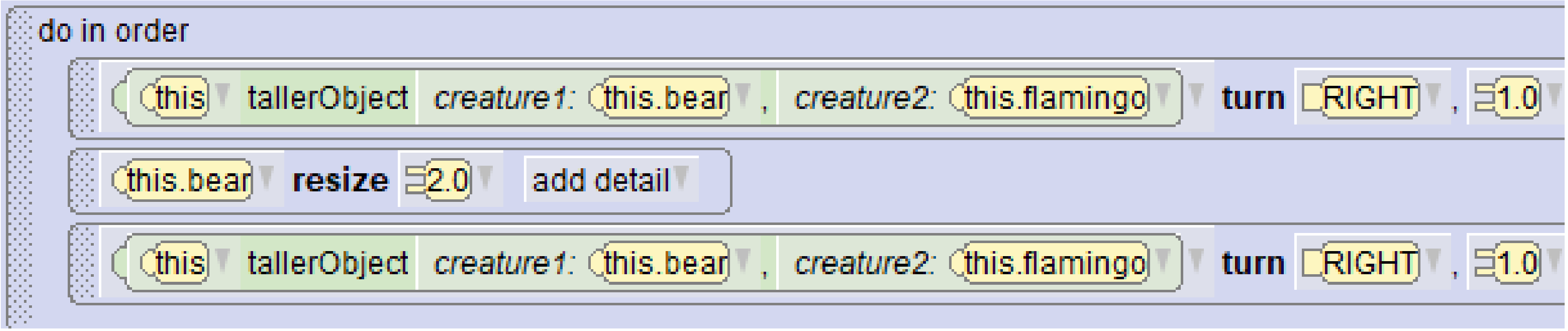
```
this tallerObject creature1: this.flamingo , creature2: this.bear say "I'm taller"
```



Q7 How do you write code for ?

- The taller of the bear and flamingo to turn around once
- The bear to double in size (so it is taller)
- The taller of the bear and flamingo to turn around once.

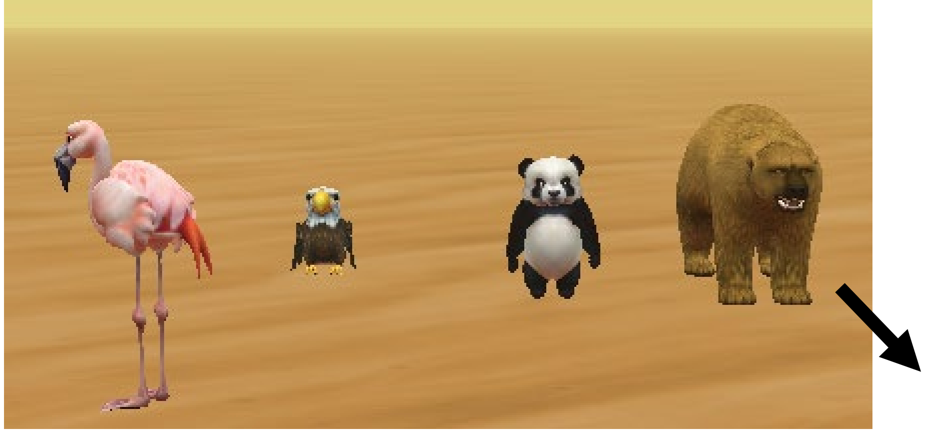
Q7 How do you write code for ?

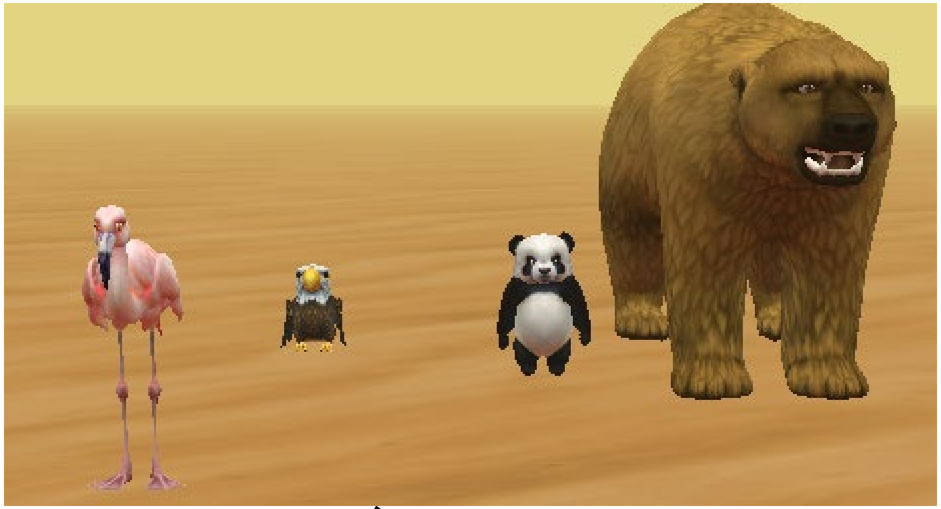


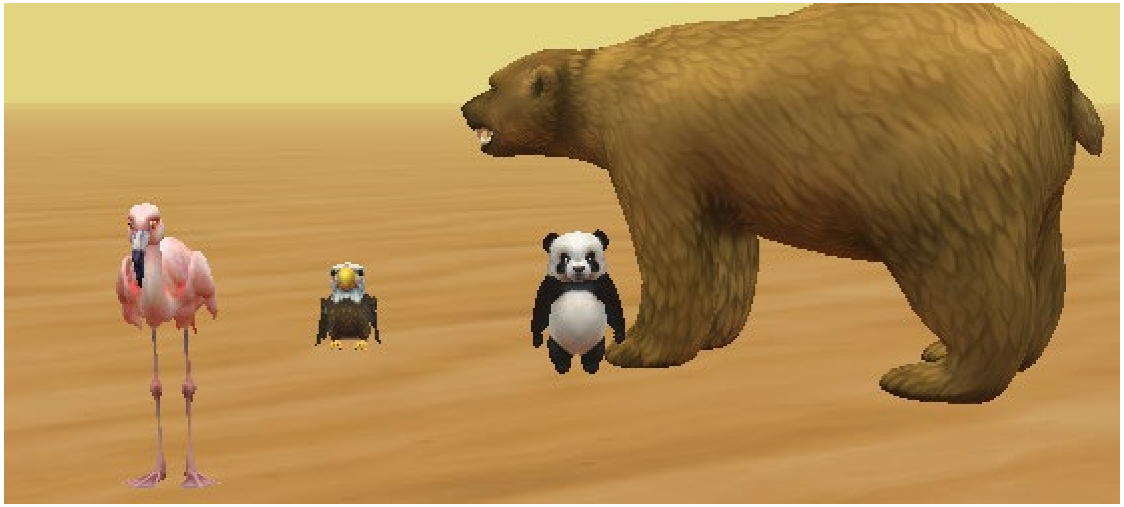
```
do in order
    (this tallerObject creature1: this.bear , creature2: this.flamingo) turn RIGHT , 1.0
    this.bear resize 2.0 add detail
    (this tallerObject creature1: this.bear , creature2: this.flamingo) turn RIGHT , 1.0
```

Use tallerObject function in place of an object.

Q7 When code runs…



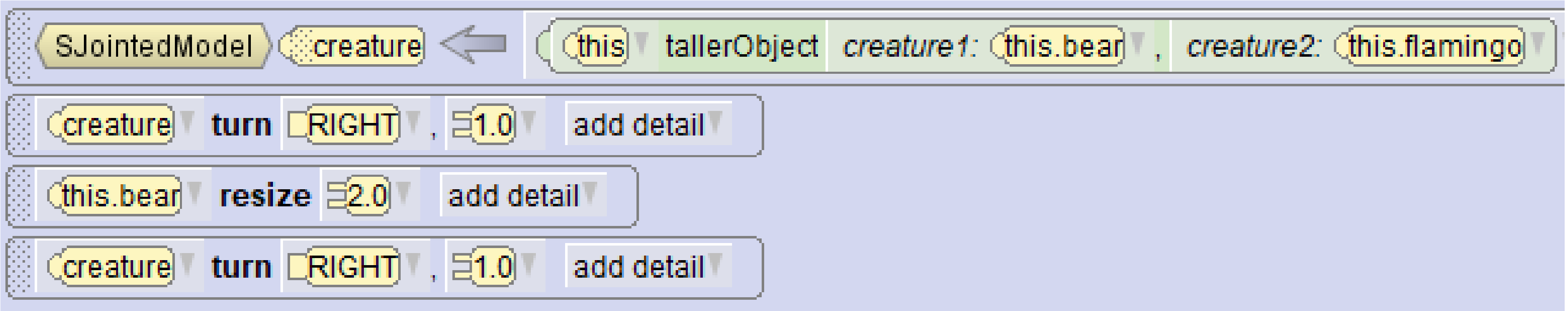




1. Flamingo turns
2. Bear gets bigger
3. Bear turns

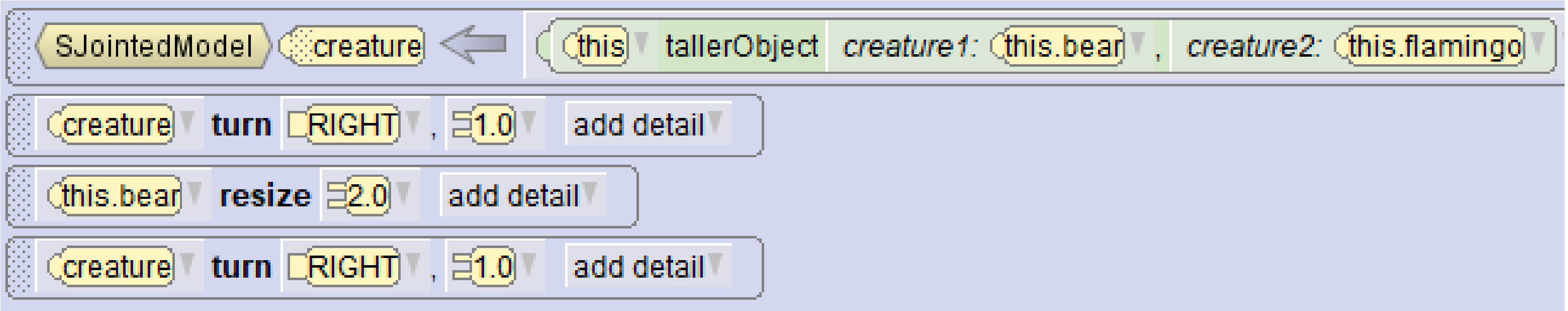
One more Question

What does this code do?



```
SJointedModel creature ← this tallerObject creature1: this.bear , creature2: this.flamingo
creature turn RIGHT , 1.0 add detail
this.bear resize 2.0 add detail
creature turn RIGHT , 1.0 add detail
```

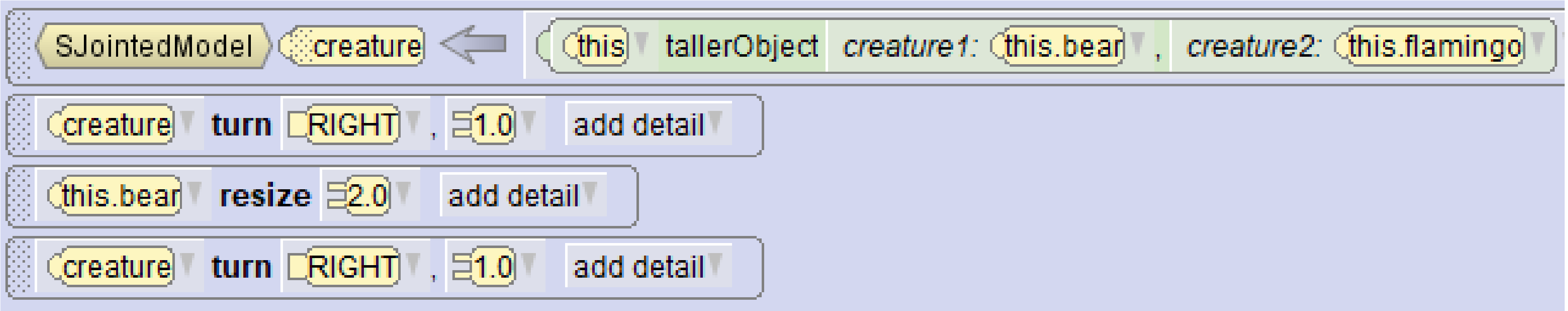
What does this code do?



```
SJointedModel creature ⇐ this tallerObject creature1: this.bear , creature2: this.flamingo
creature turn RIGHT , 1.0 add detail
this.bear resize 2.0 add detail
creature turn RIGHT , 1.0 add detail
```

- The taller animal (flamingo) is stored in variable creature.
- Flamingo turns around, then **bear** gets bigger.
- Then Flamingo turns around again!

What does this code do?



```
SJointedModel creature ← this tallerObject creature1: this.bear , creature2: this.flamingo
creature turn RIGHT , 1.0 add detail
this.bear resize 2.0 add detail
creature turn RIGHT , 1.0 add detail
```

- In the last line if we want the taller of the two to turn around, we MUST call the function again to recalculate the taller one, since the bear changed its height

Class Today

- Jumping cat calculating how high and how far to jump, and other things…