



CPSC203 – Introduction to Problem Solving and Using Application Software

Winter 2010

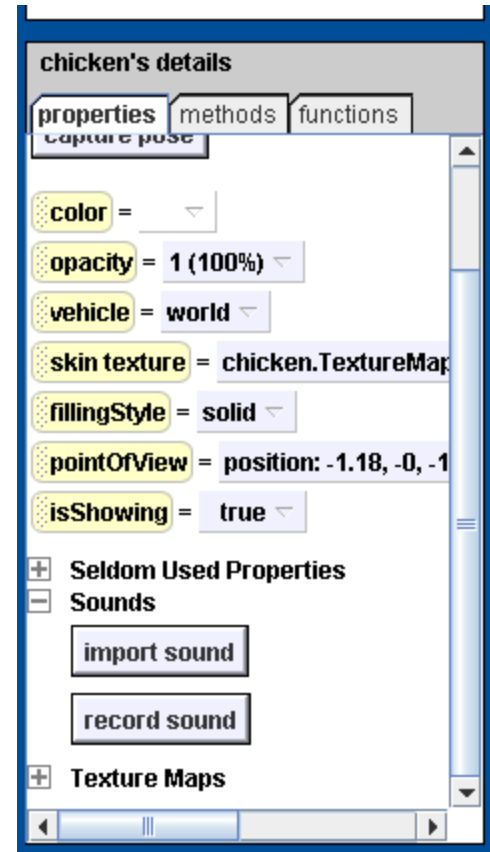
Tutorial 8: Mehrdad Nurolahzade

Introduction

- Sounds in Alice
- part named
- Lists
- For all in order and For all together

Adding Sound Files to Alice

- To add a sound file to an Alice object, select the object, go to properties and expand the sound section.
- We now have two options: we can either import a sound file, or record one ourselves.



Playing a Sound

- To play a sound file, use an object's *play sound* method and select one of the files imported/recorded as the *sound* parameter.
- If a sound file is too long, the *duration* parameter can be used to limit the length of file played.

The Farm

- Grab the Alice program file from:
<http://pages.cpsc.ucalgary.ca/~kawash/peeking/tutorials/lab6.a2w>
- The world contains three animals, and the fist method calls a class-level method:
horse.neigh.
- This method turns the mouth of the horse downwards then upwards.
- What is lacking here is the actual sound of the horse neighing.

Exercise

- Create a class-level method for the cow class called mooing. In it, the cow should turn its lower mouth in the same way the horse does.
- Notice that the cow class already has a built-in “moo” sound file so there is no need to import one.

Exercise

- Do the same thing for the chicken class. Notice that a chicken sound is available, but it does not belong to the chicken class, but to the world.
- Fix the durations so that the chicken sound does not go on after the chicken's mouth movement is complete.

Exercise

- Create a new world-level method named *introduction*. This method takes as parameters the animal part that needs to be turned (mouth/lower mouth) and the sound the animal needs to make.
- Call this method three times, so that the horse, cow and chicken all get the chance to introduce themselves in their own language.

Finding Sound Effects Online

- <http://www.soundboard.com>



- <http://www.soundsnap.com>



Function: part named

- Grab the Alice program file from:
<http://pages.cpsc.ucalgary.ca/~kawash/peeking/tutorials/lab6lists.a2w>
- Complete the *world.lookToRight* method so that it makes the *aCow* object passed as a parameter to it turn its head to right and then back to the original position.
- How do you ask Alice to return the head part of the cow object?

Function: part named

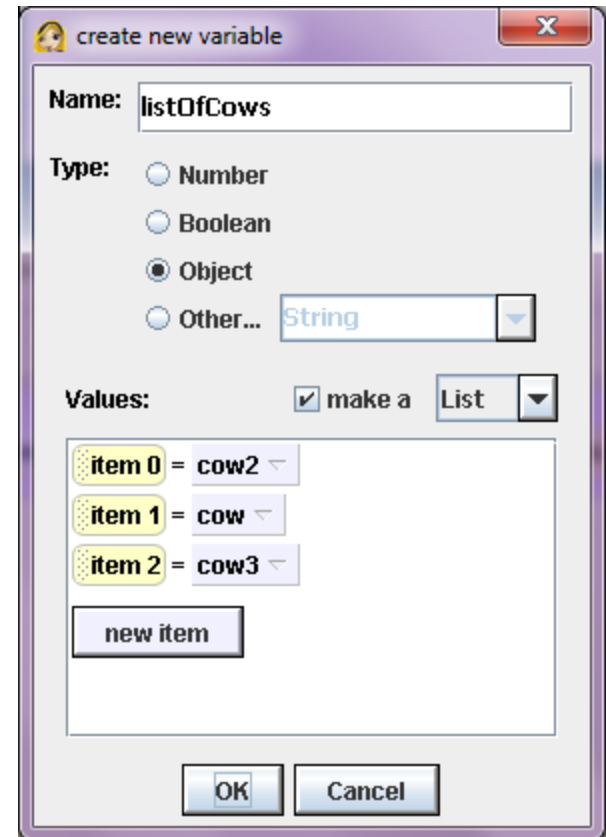
- The answer is using the cow function “part named” which gives you access to the named part of the cow object.
- For example, **cow part named *body.frontRightTigh.leg*** gives the front right leg while **cow part named *neck.head*** would give you the head.

Exercise

- Modify the program so that all the cows look to their right at the same time.

Lists

- Since all the objects we are moving are of the same type and use the same methods and body parts, we can simply add them all to a list.



listOfCows

- click on *world* in the object tree, then select “create new variable” in the *world*’s properties.
- Name the variable “listOfCows”.
- Select the “make a List” checkbox .
- Add the three cows (*cow*, *cow2*, and *cow3*) as items in the list.

“For all in order” and “For all together”

- The two control structures that are used with lists are named “For all in order” and “For all together”.
- **For all in order** performs operations on lists sequentially, i.e. one list item at a time.
- **For all together** performs operations on all list items simultaneously.

Exercise

- Modify your program so that the cows turn their heads one at a time using the **For all in order** control structure.