Announcements

• Read Chapter 9, Sec 2 for next time
• Assignment 6 due Thursday, Nov 4
• Today
  – Chapter 9, Section 1 – Lists
Collections

• In some animations, several objects must perform the same actions
  – Example: marching band marching
• It is convenient to collect all objects into a group (collection)
  – Major benefit – write code for all the objects in the group (rather than separate code for each object)
List

• A list - one way to organize objects into a collection
  – You may use lists to organize
    • Shopping list
    • Todo list

• In programming, a list is a collection of objects or information. We call an organizing structure a **data structure**.
Creating Lists

• In Alice, a list can be a list of numbers, or a list of objects, or a list of colors, etc.

• Let’s create a list of skeletons

![Skeletons Image]
Create List (cont)

- Type in name
- Select type
- Select “make a list”
- Add skeletons to list (click “new item” 4 times)
- Result is:
Programming with a List

• Can “iterate through a list”
  – Do something to each item in the list
    • In order (use “For all in order”)
    • All together (use “For all together”)
Example/Demo: Iteration in Order

For each skeleton in order
  skeleton says “Boo”
For each skeleton in order
  skeleton turns its head around
Applying a Part of an object

- Drag in skeleton turn
- Select part
- Drag over part
- Drag in item
- Type in part (must spell correctly!)
Example/Demo: Iteration Together

For all skeleton together
skeleton says “Boo”
For all skeleton in together
skeleton turns its head and neck around
Add in a girl

- Her parts are almost the same to the head.

- Rename “body” part to “upperBody”

- Add her to the list and click “Play”
Add in MadScientist

• His parts are not the same structure as the skeleton, cannot be renamed to match.

• What happens if we put him in the list and Play?

• Note: Alice bug – cannot delete from a list
Selecting an item from a list

• You can select a random item or a specific item from a list.
• This code makes a particular skeleton jump

• Suppose we want to have one item from our list (chosen randomly) to jump up and down. How would we do that?
List Questions

• What are differences between *For all in order* and *For all together*?
• Why is the list a world variable?
• When would you want to use each of them?
• What can you put in a list?
• When can you refer to a part of an object in a list?
• What type of method can you not put in a *For all in order* or *For all together*?
Below, the same skeleton jumps up and flips, can't do for each skeleton

For all `world.scarythings`, every `Obj item_from_scarythings` together
- `skeleton.jumpUpAndFlip`

Can pass each object as a parameter

For all `world.scarythings`, every `Obj item_from_scarythings` together
- `world.jumpUpAndFlip scaryThing = item_from_scarythings`
Classwork today

- Create a list of players
- Make them do several things (See handout).
Solution to having a random object from a list jump

- Add a variable called jumper
- Drag over and set the variable “jumper” to skeleton temporarily
Solution (cont)

• Drag over the list scaryThings and drop it on skeleton. You can choose which item in the list. Select “random item from list”
Solution (cont)

• Then change “skeleton” in the move commands to “world.jumper”

• Play a random object from your list will jump each time this code executes.