



Custom Functions

```
1 stop();
2 //draggable variables to snap draggables to original positions
3 var startX:Number;
4 var startY:Number;
5 //response variables for drag/drop activities
6 var invalidResponseText:String;
7 var incorrectResponseText:String;
8 //variable which will gray down a correctly used draggable
9 var activeAlpha:Number = .5;
10 //alpha variable which will bring alpha to 100% on a previously used draggable, if
    clicked
11 var activeAlpha:Number = 1;
12 //dragBounds variables which set the parameters in which draggables can move on the
    screen
13 var dragBounds:Rectangle = new Rectangle(235,63,322,297);
14 //variable identifying drop targets
15 var validResponseStart:String = "dropTarget";
16
17 // transform to control volume.
18 var genericTransform:SoundTransform = new SoundTransform();
19 var myVolume:Number = 1;
20 var myStartVolume:Number = 0;
21 var volumeOn:Number = 1;
22 var volumeOff:Number = 0;
23
24 //function to update the soundTransform.
25 function updateTransform() {
```

Custom Functions

- We've done this before (button functions)
- Only new concept: Passing arguments (data) we actually want to use.
- Why?
 - To be lazy, to make changes easier and more widespread, all the typical stuff
- Syntax (should be familiar)
(*variableName:variableType, nextName:nextType*)
- Example:
(*evt:MouseEvent*)

Once you have the arguments

- Use them inside function just like a variable.

- For example:

```
function showScore(nCorrect:int, nWrong:int) {
```

```
    var score:Number;
```

```
    score = nCorrect / (nCorrect + nWrong);
```

```
    trace(score);
```

```
}
```

- Function call:

```
showScore(25, 75);
```

Variable “scope”

- Passed to a function or created inside a function
 - Short time to live
- Functions can also “pull” variables from outside the function.
- Order: check inside first, then outside

Returning data

- Functions can give you a “result”
- Needs a type
- Uses keyword return
- For example:

```
function showScore(nCorrect:int, nWrong:int):String {  
    var score:Number;  
    score = nCorrect / (nCorrect + nWrong);  
    return String(score);  
}
```

- Function call:
resultsText.text = showScore(25, 75);

Functions and pathing

- It's not about where you make the function call from
- It's about where the function "lives"

Example (Kendra Hall)

- Functions to do custom start drag
- Functions to do custom stop drag
- A function to update a progress gauge

