

# Art 382: Introduction to Interactive Media

[mcdo@umbc.edu](mailto:mcdo@umbc.edu)

Lecture 20, Ap. 23

# Schedule

- Take Roll
- All about "foamy.fla"

# Review

- Variable declarations
  - "var a: int;"
  - makes a variable-- a place to store a number
- assignments
  - a = 2;
  - copies a number from one place to another
- expressions
  - a = b\*2;
  - do arithmetic; generally part of an assignment

# more review

- timeline control
  - gotoAndPlay(20);
  - sets which frame Flash plays next
- conditional statements
  - if (a==20) { c=b; } else { c=d; }
  - depending on the "condition", different statements are executed

# for loops

```
for (initializer; condition; increment) {  
    statements  
}
```

initializer-- sets a counter variable

condition-- tests the counter; if true, loop again

increment-- modifies the counter (usu. by adding a "step" value)

# for examples

- `for (i=0; i<10; i=i+1) { bang(i); }`
  - calls "bang" 10 times with values
  - 0, 1, 2, 3, 4, 5, 6, 7, 8, 9
- `for (j=10; j>0; j=j-1) { bang(j); }`
  - calls "bang" with 10,9,8,7,6,5,4,3,2,1
- `for (mop=0; mop<200; mop+=20) { bang(mop); }`
  - "bang" with 0,20,40,60,80,100,120,...180

# Arrays

- lists of variables, referred to by name and "index"
- must be declared: "var ar:Array;"
- must be "initialized": "ar= new Array(size);"
- used via bracket notation: "name[i] = x;"
- work great with for loops!
- for (i=0; i<6; ++i) { ar[i]=i\*2 }
  - sets ar[0]=0, ar[1]=2, ar[2]=4, ar[3]=6, ar[4]=8,ar[5]=10

# mouse input

- `addEventListener(MouseEvent.CLICK, functionName);`
- `stage.addEventListener(MouseEvent.MOUSE_MOVE,`
- generate events in response to mouse input
- "CLICK" you know; many others
- `MOUSE_DOWN`, `MOUSE_UP`,  
`MOUSE_OVER`, `DOUBLECLICK`
- different functions for each one, maybe!

# MouseEvents are complicated!

```
function makeFoam(e:MouseEvent) {  
    foam[foamCounter].x = e.stageX;  
    foam[foamCounter].y = e.stageY;  
}
```

- e is a "MouseEvent" object
  - contains lots of data
  - "e.stageX" = x-coordinate of the mouse location

# "foamy" is kinda complicated

- arrays of symbols, velocities (x, y, and rotation)
- use of many symbols
- eventListener attached to the stage (no button)
- eventListener makes symbols appear in response to MOUSE\_MOVE
- animation of position, rotation, alpha