Animation

Active Sketches: Interaction vs Animation

In an *active sketch*, things change (or have the potential to change) over time.

In *interaction*, the changes over time are driven by user actions.

- mouse or keyboard actions

In *animation*, the changes over time are driven (only) by the passage of time.

A sketch can include both interaction and animation at the same time.

Variables

In a static sketch, the position, size, and color of each shape is hardcoded.

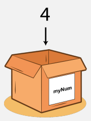
```
fill(255,0,0);
rect(100,100,100,100);
```

- changing the value means changing the code itself and running the program again

For values that change over time, we need *variables*.

Variables

A *variable* is a name which has an associated value.



- when a statement containing the name of a variable is executed, the computer looks up and substitutes in the current value of the variable

```
// draw a rectangle whose upper left corner
// is at the mouse's current position
rectMode(CORNER);
rect(mouseX,mouseY,50,50);
```

- *system variables* are defined and maintained by the system
- ☆ other variables must be defined and maintained by the programmer

https://miro.medium.com/v2/resize:fit:1400/1*QT3V91g69EUI7Uv0GY1nsg.png

Animation Questions

What changes (or has the potential to change) from one frame to the next?

→ one variable for each thing identified

For each variable –

- what kind of value is it?
  - whole number
  - numbers with a decimal point
  - true or false
- what's the starting point?
- how does the value change from one frame to the next?

Animation Example

for a rectangle moving to the right…

What changes (or has the potential to change) from one frame to the next?

→ one variable for each thing identified

x position of the rectangle

For each variable –

- what kind of value is it?
  - whole number
  - number with a decimal point
  - true or false

number – either whole number or number with decimal point

- what's the starting point?

on the left side – x is 0 (CORNER mode)

- how does the value change from one frame to the next?

x gets bigger → add 1

(Non-System) Variables

Four steps –

1. *declare* – "hey computer, this name is going to mean something!"
   - variables must be declared before initialization
2. *initialize* – give it a starting value
   - variables must be initialized before use or update
3. *use* – to draw (or whatever)
4. *update* – change its value

variable declaration

```
type name;  // description
```

- *type* specifies what kind of value the variable stores
  - `int` – whole numbers
  - `float` – numbers with decimal points
  - `boolean` – true or false
- *name* should be brief but descriptive
  - convention is to start with lowercase letter and separate words with capital or _
  - case-sensitive
- *description* provides info not apparent from the declaration itself

```
name = value;
```

assignment statement

- *value* can be a literal or an expression
  - as an expression, it can reference the variable *name*
- the value of *value* is computed and then stored in *name*, replacing any value already there

Structural Pattern – Animation

Where does stuff go?

Animation variables are usually –

- declared at the very beginning of the sketch, before `setup()` and `draw()`
- initialized in `setup()`
- used and updated in `draw()`

Animation Example

What changes (or has the potential to change) from one frame to the next?

→ one variable for each thing identified

For each variable –

- what kind of value is it? → type, for the declaration
  - whole numbers → `int`
  - numbers with decimal points → `float`
  - true or false → `boolean`
- what's the starting point? → initialization
- how does the value change from one frame to the next? → update

for a rectangle moving to the right…

x position of the rectangle

number – either whole number or number with decimal point

on the left side – x is 0 (CORNER mode)

x gets bigger – add 1

```
// rectangle moves to the
//  right

int x; // x coordinate of
       //  the left side

void setup () {
  // open window
  size(800, 400);

  // initialize variables
  x = 0;
}

void draw () {
  // draw one frame
  background(255);
  rectMode(CORNER);
  stroke(0);
  fill(255, 0, 0);
  rect(x, 200, 60, 20);

  // update variables
  x = x+1;
}
```

Pattern Summary

- nothing changes (static picture)

  → static mode sketch

```
size(200,200);
background(255);
rectMode(CENTER);
fill(255,0,0);
rect(100,100,100,100);
```

- something changes based on the mouse position

  → active mode sketch (`setup()`, `draw()`) using `mouseX`, `mouseY`

```
void setup () {
  size(400, 400);
}
void draw () {
  background(255);
  rectMode(CORNER);
  fill(255,0,0);
  rect(mouseX,mouseY,50,50);
}
```

- something changes over time

  → active mode sketch (`setup()`, `draw()`) with animation variables

```
int x;
void setup () {
  size(400, 400);
  x = 0;
}
void draw () {
  background(255);
  rectMode(CORNER);
  fill(255,0,0);
  rect(x,100,50,50);
  x = x+1;
}
```

- do something when the mouse is clicked (or pressed or released) or a key is pressed

  → active mode sketch (`setup()`, `draw()`) with event handlers