Names: _____

Animation Questions

- what changes from one frame to the next? → one variable for each thing
- for each variable –
  - what kind of value is it? (whole number, number with a decimal point, true or false) → type for declaration (at the very beginning of the sketch, before `setup()`)
  - where does the value start? → initialization (in `setup()`)
  - how does the value change from one frame to the next? → update (in `draw()`, usually at the end)

Variable Recap

- declaring variables

```
type name;            // where type is the type of the variable and
                      // name is its name
```

- types

```
int                   // whole number
float                 // numbers with decimal points
boolean               // true or false
```

- assignment statements – for initialization and update

```
name = value;         // where name is the name of the variable whose
                      // value is to be set, and value is the new value
                      //  (value can be a literal or an expression)
```

Program Structure Recap (Active Sketch)

```
// declare variables

void setup () {         // …stuff to do once at the beginning goes here…
  // open window
  // initialize variables
}

void draw () {          // …stuff to do for each frame goes here…
  // draw frame
  // update variables
}
```

At the End of Class

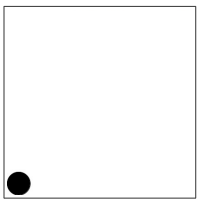
Hand in whatever you have done during class, even if a sketch is incomplete.

- Make sure each sketch is named as directed and has a comment with the names of your group. Also be sure to save your sketches! (This should be in your sketchbook **~/cs120/sketchbook**)
- Copy the entire directory for each sketch (not only the `.pde` file) into your handin directory (**/classes/cs120/handin/*username***). You only need to hand in one copy for the group.

Exercises

For each exercise, identify the variables needed and fill in the table from Monday's in-class exercises before writing code.

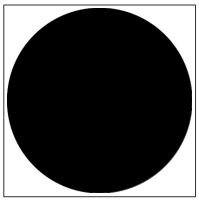
1. Create a sketch named **sketch_250917a** where a small black circle (width and height 50) starts in the lower left corner of the window as shown and moves diagonally up and right. The drawing window is 400x400.



2. Modify the previous sketch so it works no matter what size the drawing window is – that is, the circle always starts in the lower left corner without having to change more than the size of the window. You will need one or both of the following system variables:
   - `width`, `height` – dimensions of the drawing window (only valid after `size(…,…)` has been executed)

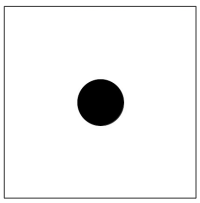
3. Create a sketch named **sketch_250917b** where a circle starts just filling the drawing window (as shown) and shrinks. The drawing window is 400x400.

   Use `width` and/or `height` as appropriate rather than hardcoding specific values related to the size of the window (other than in `size(…,…)`).



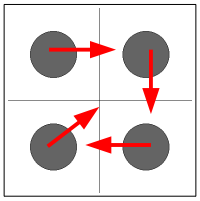
4. Create a sketch named **sketch_250917c** where a circle (width and height 100) centered in the window goes from black (as shown) to red. The drawing window is 400x400.

   Use `width` and/or `height` as appropriate rather than hardcoding specific values related to the size of the window (other than in `size(…,…)`).



5. Create a sketch named **sketch_250917d** where four circles start in the positions shown and move as indicated by the red arrows. (You do not need to draw the arrows, the lines, or the box around the outside – those are just to help you understand how the sketch should work.) The drawing window is 600x600.

   Use `width` and/or `height` as appropriate rather than hardcoding specific values related to the size of the window (other than in `size(…,…)`).

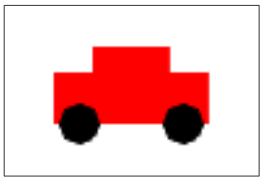


If you have time, try the following –

- Save a copy of your sketch from #4 as **sketch_250917e**, then modify it so the circle starts filled in white and then becomes redder.

- The statements on the right draw the car shown. You want to animate the car so it drives off towards the right side of the screen. What variable(s) do you need? Can you do it with just one variable?

  Fill in the table below (the car should start on the left side of the window), then create a new sketch named **sketch_250917f** where the car starts on the left side of the window and drives towards the right side.



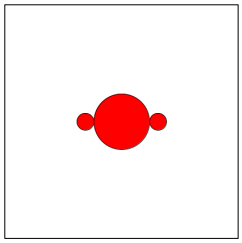
```
rectMode(CENTER);
ellipseMode(CENTER);

fill(255,0,0);
stroke(255,0,0);
rect(100,285,60,20);
rect(100,270,30,10);
fill(0);
stroke(0);
ellipse(80,295,15,15);
ellipse(120,295,15,15);
```

what changes?	what kind of value?	what's the starting point?	how does the value change?

Names: _____

If you still have time, try the following –



- Create a new sketch called **sketch_250917g** which contains a three-circle pattern as shown – the pattern should be centered in the drawing window (write the sketch so that it works with any size window) and the center circle should start with a width and height of 1 and then grow over time, with the other two circles staying the same size and always just touching the center circle.

  Start by identifying the animation variables needed – fill in the table below.

what changes?	what kind of value?	what's the starting point?	how does the value change?

- The reading introduced the `random()` function which can be used to generate random values: `random(n)` returns a value between 0 and n (including n). You can substitute `random(n)` (with something substituted for n) anywhere a number (specifically, a float value) is expected. For example

  ```
  fill(random(255));
  ```

  sets the fill color to a random shade of gray between 0 and 255.

  - Create a new sketch called **sketch_250917h** which has a single circle centered in the drawing window. (Write your sketch so that it works with any size window.) Use a variation on the `fill` example given above to choose a random color (rather than a random shade of gray) for the ellipse – that is, the red and the green and the blue components of the color should each be separate random numbers. Run the sketch – what do you see?
  - Modify the sketch so that a random color is chosen for the circle each time the sketch is run, but then it stays a fixed color – no flickering. Hint: this is still animation because something changes over time – in this case, the circle's color changes from "not yet initialized" (the circle's color goes from "not yet chosen" to a particular color at the very beginning of the sketch even though it doesn't change again after that) – and animation calls for variables!

what changes?	what kind of value?	what's the starting point?	how does the value change?