Processing CribSheet

Code	Examples

Datatypes:
`int irk;`
`float fred;`
`boolean bob;`
`String sasquatch;`
`color crass;`
	`int h, dread = -56, fish = 2;`
	`boolean babble = true;`
	`String y = "Why???", darp;`
	`color my_red = color(255,0,0);`

Comparisons and Conjunctions
`<  less than`
`<= less than or equal to`
`== equal`
`!= not equal`
`>= greater than or equal to`
`>  greater than`
 	
`&&  and`
`||  or`
`!   not`
	`boolean harry = true;`
	`int x = 63;`
	
	`if (harry == true) {body}`
	`if (harry) {body}`
	`if (!harry) {body}`
	
	`boolean grat = x > 56;`
	
	`if (x > 56 || harry) {body}`
	
	`if (x > 56 && x < 90 && x % 2 == 0) {body}`

If:
`if (test) {body of statements}`
 	
`if (test) single_statement;`
 	
`if (test) {body_1}`
`else {body_2}`
 	
`if (test_1) {body_1}`
`else if (test_2) {body_2}`
`else if (test_3) {body_3}`
`else {body_4}`
	`if (B > 12)`
	`  println("It's a big B");`
	`else if (B < 0 || B % 3 == 1) {`
	`  C = 12;`
	`  println("Oh wow!");`
	`} else {`
	`  C = 0;`
	`  println("That's all.");`
	`}`

Loop/Iteration
`while (test) {body}`
 	
`for (init; test; update) {body}`
 	
`break;  // immediately terminates loop`
 	
`continue;  // immediately goes back to the while(test) clause or the for() iterator`
	`int k;`
	
	`k = 0;`
	`while (k < 5){`
	`  println("k is",k);`
	`  k += 1;`
	`}`
	
	`for (k = 0; k < 5; ++k)`
	`  println("k is",k);`
	
	`for (int f = 0; f < 5; ++f){`
	`  if (f % 4 == 0)`
	`  println("f is divisible by 4");`
	`  else`
	`  println("f is not divisible by 4");`
	`}`
	
	`k = 0;`
	`while (true) {`
	`  println("k is",k);`
	`  k += 1;`
	`  if (k > 5) break;`
	`}`

Conversions and Operators

`int/float to String:`
`String str(int n)`
`String str(float f)`
	`str(45) -> "45"`
	`str(45.3) -> "45.3"`
	`str(99999) -> "99999"`

`String to int: int(String s)`	`int("45") -> 45`
	`int("fred") -> 0`
	`int("45a") -> 0`

`String to float: float(String s)`	`float("45.3") -> 45.3`
	`float("45") -> 45.0`
	`float("a45") -> NaN  (Not a Number)`

`String methods:  assume String s;`
`int s.length()`
`char s.charAt(int position)`
`string s.substring(int begin, int past_end)`	

Program and Function Structure

```
Global variable declarations

void setup() {
   local variable declarations
   size(____, ____);  // these must be literal integers (not variables)

   // code to set things up, or to draw a static image
}

void draw() {
  local variable declarations
  // code to redraw image for each frame
}
```

Function declaration:

```
datatype function_name (datatype arg1, datatype arg2 …) {
    statements;
}
```

Drawing

```
line(x1,y1,x2,y2)

rect(x,y,width,height)  // (x,y) is upper-left corner

circle(x_center,y_center,diameter)

Color possibilities for fill(), stroke(), background() (and others):
(gray)
(red,green,blue)
(red,green,blue,alpha)

stroke(color possibilities)  // color of the following lines or outlines

fill(color possibilities)   // color of the following filled areas
```

Objects

```
class Box {
   // instance variables
      float x, y, w, h;

   // a constructor:
      Box(float ax, float ay, float aw, float ah) {
          x = ax;
          y = ay;
          w = aw;
          h = ah;
      }
   // an "overloaded" constructor calling another constructor to do work:
      Box(float bx, float by) {
          this(bx, by, random(2,10), random(3, 18));
      }
   // a method:
      void display() { rect(x,y,w,h); }

   // a method calculating how far this Box is from another
      float howFarAway(Box fred) { return dist(x, y, fred.x, fred.y); }
}

// Now for some code using Box class and its instances

Box a;   // no actual box yet… just a Box variable
Box b = new Box(4., 5.);   // b is an actual box
a = new Box(7., 8.5, 34., 5.75);  // now a is also an actual box
a.display();  // now a is drawn
a.x += 8.;    // now a is "moved" to the right (though just its x variable is changed)
a.display();  // now a second rectangle is drawn, further to the right.
Println(b.howFarAway(a));
```