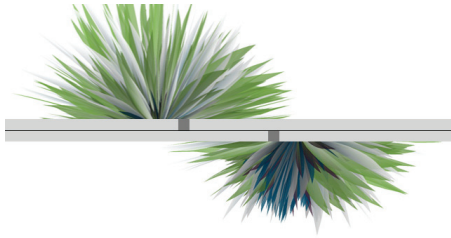


## Scrollbar

### CODE



```
/**
 * Scrollbar.
 *
 * Move the scrollbars left and right to
 * change the positions of the images.
 */

// The next line is needed if running in JavaScript Mode with
// Processing.js
/* @pjs preload="seedTop.jpg,seedBottom.jpg"; */

HScrollbar hs1, hs2; // Two scrollbars
PImage img1, img2; // Two images to load

void setup() {
  size(640, 360);
  noStroke();

  hs1 = new HScrollbar(0, height/2-8, width, 16, 16);
  hs2 = new HScrollbar(0, height/2+8, width, 16, 16);

  // Load images
  img1 = loadImage("seedTop.jpg");
  img2 = loadImage("seedBottom.jpg");
}

void draw() {
  background(255);

  // Get the position of the img1 scrollbar
  // and convert to a value to display the img1 image
  float img1Pos = hs1.getPos()-width/2;
  fill(255);
  image(img1, width/2-img1.width/2 + img1Pos*1.5, 0);

  // Get the position of the img2 scrollbar
  // and convert to a value to display the img2 image
  float img2Pos = hs2.getPos()-width/2;
  fill(255);
  image(img2, width/2-img2.width/2 + img2Pos*1.5, height/2);

  hs1.update();
  hs2.update();
  hs1.display();
  hs2.display();

  stroke(0);
  line(0, height/2, width, height/2);
}

class HScrollbar {
  int swidth, sheight; // width and height of bar
  float xpos, ypos; // x and y position of bar
  float spos, newspos; // x position of slider
  float sposMin, sposMax; // max and min values of slider
  int loose; // how loose/heavy
  boolean over; // is the mouse over the slider?
  boolean locked;
```

```

float ratio;

HScrollbar (float xp, float yp, int sw, int sh, int l) {
    swidth = sw;
    sheight = sh;
    int widthtoheight = sw - sh;
    ratio = (float)sw / (float)widthtoheight;
    xpos = xp;
    ypos = yp-sheight/2;
    spos = xpos + swidth/2 - sheight/2;
    newspos = spos;
    sposMin = xpos;
    sposMax = xpos + swidth - sheight;
    loose = l;
}

void update() {
    if (overEvent()) {
        over = true;
    } else {
        over = false;
    }
    if (mousePressed && over) {
        locked = true;
    }
    if (!mousePressed) {
        locked = false;
    }
    if (locked) {
        newspos = constrain(mouseX-sheight/2, sposMin, sposMax);
    }
    if (abs(newspos - spos) > 1) {
        spos = spos + (newspos-spos)/loose;
    }
}

float constrain(float val, float minv, float maxv) {
    return min(max(val, minv), maxv);
}

boolean overEvent() {
    if (mouseX > xpos && mouseX < xpos+swidth &&
        mouseY > ypos && mouseY < ypos+sheight) {
        return true;
    } else {
        return false;
    }
}

void display() {
    noStroke();
    fill(204);
    rect(xpos, ypos, swidth, sheight);
    if (over || locked) {
        fill(0, 0, 0);
    } else {
        fill(102, 102, 102);
    }
    rect(spos, ypos, sheight, sheight);
}

float getPos() {
    // Convert spos to be values between
    // 0 and the total width of the scrollbar
    return spos * ratio;
}
}

```