

```

let p;
let stemCol = [];
let leaf = [];
let numLeaf = 10;

function setup() {
  createCanvas(windowWidth, windowHeight);
  colorMode(RGB, 255, 255, 255, 1);
  if(width>height)
    p = height/100;
  else
    p = width/100;
  for(let i=0; i<p*24; i++)
    stemCol[i] = random(20);
  for (let i = 0; i<5; i++)
    leaf[i] = [];
  for (let i = 0; i<numLeaf; i++){
    leaf[0][i] = random(-180,180);
    leaf[1][i] = -1*random(height/400,height/100);
    leaf[2][i] = random(1,width/100);
    leaf[3][i] = random(110,200);
    leaf[4][i] = random(10,150);
  }
}

function draw() {
  background(100*mouseX/width-20,100*mouseX/width+50,100*mouseX/width+80);
  noStroke();
  fill(25+40*mouseX/windowWidth,50+40*mouseX/windowWidth,0);
  rect(0, height/1.5, width, height);
  celestial();
  pumpkin(width/2, height/1.4);
  for(let i=0; i<numLeaf; i++)
    fall(i);
}

function fall (i){
  let col = 100*mouseX/windowWidth-50;
  fill(leaf[3][i]+col,leaf[4][i]+col,0);
  stroke(leaf[3][i]+col-20,leaf[4][i]+col-20,0);
  strokeWeight(width/1200);
  let x = cos(leaf[0][i]+0.005)*(width/2)+width/2;
  let y = (leaf[1][i]+0.01)*height/leaf[2][i];
  quad(p*2.5+x,p*5+y, p*1.2+x,p*2.8+y, p*8.8+x,p*2.8+y, p*7.5+x,p*5 +y);
  quad(p*5 +x,p*7+y, p*6.7+x,p*2.8+y, p*5 +x,p*0.6+y, p*3.3+x,p*2.8+y);
}

```

```

quad(p*2.5+x,p*5+y, p*2.3+x,p*5.5+y, p*7.7+x,p*5.5+y, p*7.5+x,p*5 +y);
if ((leaf[1][i]+=0.01)*height/leaf[2][i] - 30 > height){
  leaf[1][i] = -1;
  leaf[2][i] = random(1,width/100);
  leaf[3][i] = random(110,200);
  leaf[4][i] = random(10,150);
}
}

function celestial (){
  let x = map(mouseX, 0, width, p*15, width-p*30, true);
  ellipseMode(CORNER);
  fill(255,254,122,1-mouseX/width);
  ellipse(x, height/10, p*15, p*15);
  fill(100*mouseX/width-20,100*mouseX/width+50,100*mouseX/width+80,1);
  ellipse(x, height/10, p*10, p*10);
  fill(255,205,40,mouseX/width);
  ellipse(x, height/10, p*15, p*15);
}

function pumpkin (x, y) {
  ellipseMode(CENTER);
  x-=p*20;
  y-=p*4;
  cell(95, p*20+x, y, p*40);
  cell(100, p*10+x, p*2+y, p*42);
  cell(100, p*30+x, p*2+y, p*42);
  cell(100, x, p*4+y, p*44);
  cell(100, p*40+x, p*4+y, p*44);
  stem(x-p*20, y-p*36);
  cell(100, p*10+x, p*6+y, p*42);
  cell(100, p*30+x, p*6+y, p*42);
  cell(105, p*20+x, p*8+y, p*40);
}

function stem (x, y) {
  let col = 40*mouseX/windowWidth-20;
  noFill();
  strokeWeight(p/0.8);
  for(let i=1; i<=p*4; i++){
    stroke(stemCol[i]+20+col,stemCol[i]+40+col,0);
    curve(p*14+x,y, p*28+i*3+x,p*22+i*2+y, p*33+i+x,p*6+i/2+y, p*14+x,y);
  }
  for(let i=1; i<=p*4; i++){
    stroke(stemCol[i]+30+col,stemCol[i]+60+col,0);
    curve(p*52+x,y, p*40+i*2+x,p*30-i*2+y, p*37+i+x,p*8-i/2+y, p*52+x,y);
  }
  strokeWeight(p*0.7);
}

```

```

stroke(35+col,60+col,0);
fill(35+col,60+col,0);
quad(p*33+x, p*6+y, p*37+x, p*8+y, p*41+x, p*6+y, p*37+x, p*4+y);
}
function cell(temp, x, y, h) {
  let count=0;
  let col=80*mouseX/width+60;
  noFill();
  strokeWeight(width/200);
  for(let i=0; i<temp; i++){
    stroke(col, col/2, 0);
    ellipse(x, y, (h/1.5)+count, h+count);
    col+=0.5;
    count-=p/3.8;
  }
}
function windowResized() {
  resizeCanvas(windowWidth, windowHeight);
}

```