```
 //pixelaion
 //Lucian Novosel - 2014
 //Open sourcerer on the internet of things
 // WWW: luciannovosel.com GH: github.com/luciannovo

PImage img;
import processing.pdf.*;
PGraphics pdf;
String inputPath  = "inputImages/blowup/";
String outputPath = "outputImages/blowup/";
String imageName;
int increment = 5;
int pixel = 0;
int horz_pixels;
int vert_pixels;
int lastPixel;

void setup(){

  for(int i=0; i<39;i++){

    imageName = i + ".png";

    //Load the image and define the resolution
    img = loadImage(inputPath + imageName);
    horz_pixels = (img.width  - img.width % increment)/increment;
    vert_pixels = (img.height - img.height % increment)/increment;
    lastPixel   = img.height * img.width; //the last pixel from the image tha
    println("Max hoizontal pixels(rendered): ", horz_pixels);
    println("Max vertical pixels(rendered):", vert_pixels);
    size(horz_pixels*increment, vert_pixels*increment); //canvas is the size
    background(255);
    smooth();

    //Create the scalable PDF
    pdf = createGraphics(horz_pixels*increment, vert_pixels*increment, PDF, o
    pdf.beginDraw();
    pdf.background(255);

    //
    int firstPixel = int(increment/2 - 1);
    println("The first sampling pixel is(picture): ", firstPixel); //
    for(int vp = 0; vp < vert_pixels; vp = vp + 1){
      pixel = firstPixel + ((vp *increment) * img.width);
      for(int hp = 0; hp < horz_pixels; hp = hp + 1){
```

```
        loadPixels();
        println("Horizontal Pixels: ", hp, "Vertical Pixels: ", vp );
        println("Pixel Location is thus: ", pixel);
        float r = red(img.pixels[pixel]);
        float g = green(img.pixels[pixel]);
        float b = blue(img.pixels[pixel]);
        renderPixel( hp, vp, r, g, b); //
        pixel = pixel + increment;//update pixel here
      }
    }

    String imgPath = ("../../generatedImagesEX4/" + frame.getTitle() + ".png"
    saveFrame(imgPath);
    pdf.dispose();
    pdf.endDraw();
  }
}

void renderPixel( int horizontal_pixel, int vertical_pixel, float inpR, float
   //write to the pdf and sketch
   noStroke();
   pdf.noStroke();
   fill(inpR,inpG,inpB, 255);
   pdf.fill(inpR,inpG,inpB, 255);
   rect(horizontal_pixel*increment, vertical_pixel*increment,  increment, inc
   pdf.rect(horizontal_pixel*increment, vertical_pixel*increment,  increment,
}
```