public void pixarincorparated(int numRowPixels) {

Pixel[][] pixels = this.getPixels2D();

for (int mRow=0; mRow < numRowPixels; mRow ++) {

for (int mCol=0; mCol < numRowPixels; mCol ++) {

// for user input

int pixelCount = 0;

int avgerageRed = 0;

int avgerageGreen = 0;

int avgerageBlue = 0;

int pixelHeight = pixels.length / numRowPixels;

int pixelWidth = pixels[mRow].length / numRowPixels;

int startRow = mRow \* pixelHeight;

int startCol = mCol \* pixelWidth;

for (int row=0; row < pixelHeight; row ++) {

for (int col=0; col < pixelWidth; col ++) {

Pixel curPixel = pixels[row + startRow][col + startCol];

averageRed += curPixel.getRed();

averageGreen += curPixel.getGreen();

averageBlue += curPixel.getBlue();

pixelCount += 1;

}

}

if (pixels.length < numRowPixels || pixels[0].length < numRowPixels) {

System.out.println("Try a smaller size.");

return;

}

// calculate amount to pixelate

avgerageRed /= pixelCount;

avgerageGreen /= pixelCount;

avgerageBlue /= pixelCount;

// amount to pixelate

Color avgPixelColor = new Color(averageRed, avgerageGreen, avgerageBlue);

for (int row=0; row < pixelHeight; row ++) {

for (int col=0; col < pixelWidth; col ++) {

pixels[row + startRow][col + startCol].setColor(avgPixelColor);

}

}

}

}

}