Conner Lucas

#Project 2 (Version 1.2)
#Conner Lucas
#"Reality-Bending Clash! Battle of Titans!"

def collage():
    # Hello! Welcome to my Project 2! I'm going to do
    # my best to keep the documentation short and sweet,
    # while also being readable and easy to follow. Let's Begin!

    # For this first part, we're grabbing the main image that
    # we'll be manipulating. Then we'll make the canvas that all
    # of our images will be pasted into.
    setMediaPath()
    clashPic = makePicture("PickMeFirst.png")
    canvas = makeEmptyPicture(736, 1000, pink)
    # Here, we have the executions of the different functions.
    # There's quite a few, but they're labeled accordingly.
    mainCroppedPic = bigCrop(clashPic)
    mainCroppedPic2 = bigCrop(clashPic)
    middleCroppedPic = middleCrop(clashPic)
    smallBroly = smallCropBroly(clashPic)
    smallGogeta = smallCropGogeta(clashPic)
    smallBroly2 = smallCropBroly(clashPic)
    smallGogeta2 = smallCropGogeta(clashPic)

    # Here, we'll be manipulating the images themselves
    # with different functions, like negative, more red, etc.
    reverseBroly = reverse(smallBroly2)
    reverseGogeta = reverse(smallGogeta2)
    redBottom = maxRed(mainCroppedPic2)
    maxedBlueB = maxBlue(reverseBroly)
    distortedG = makeDistorted(smallGogeta)
    negativePic = negative(mainCroppedPic)
    negativeFaceG = negative(reverseGogeta)
    grayScaleMiddle = grayScale(middleCroppedPic)
    purpleB = makePurple(smallBroly)
    flippedBottom = reverse(redBottom)

    # Here, we're copying our images into the canvas.
    copyInto(negativePic, canvas, 0, 0)
    copyInto(flippedBottom, canvas, 0, 500)
    copyInto(grayScaleMiddle, canvas, 118, 400)
# Finally, we'll be copying my signature into the canvas, and then showing the final image!
signaturePic = makePicture("signature.png")
targetX = getWidth(canvas)-getWidth(signaturePic)
targetY = getHeight(canvas)-getHeight(signaturePic)
finalImage = chromakeySig(signaturePic, canvas, targetX, targetY)
show (finalImage)

# Below are the crop functions, labeled accordingly...

def bigCrop(srcPic):
    targetPic=makeEmptyPicture(736, 500)
    targetX = 0
    for sourceX in range(50,786):
        targetY = 0
        for sourceY in range(235,735):
            srcPx=getPixelAt(srcPic,sourceX,sourceY)
            srcColor = getColor(srcPx)
            targetPx=getPixelAt(targetPic,targetX,targetY)
            setColor(targetPx, srcColor)
            targetY = targetY + 1
            targetX = targetX + 1
    return targetPic

def middleCrop(srcPic):
    targetPic=makeEmptyPicture(500, 240)
    targetX = 0
    for sourceX in range(155,655):
        targetY = 0
        for sourceY in range(440,680):
            srcPx=getPixelAt(srcPic,sourceX,sourceY)
            srcColor = getColor(srcPx)
            targetPx=getPixelAt(targetPic,targetX,targetY)
            setColor(targetPx, srcColor)
            targetY = targetY + 1
            targetX = targetX + 1
    return targetPic

def smallCropBroly(srcPic):
    targetPic=makeEmptyPicture(90,90)
    targetX = 0
    for sourceX in range(425,515):
        targetY = 0
        for sourceY in range(470,560):
            srcPx=getPixelAt(srcPic,sourceX,sourceY)
            srcColor = getColor(srcPx)
            targetPx=getPixelAt(targetPic,targetX,targetY)
            setColor(targetPx, srcColor)
            targetY = targetY + 1
            targetX = targetX + 1
    return targetPic

def smallCropGogeta(srcPic):
    targetPic=makeEmptyPicture(100, 105)
    targetX = 0
    for sourceX in range(425,515):
        targetY = 0
        for sourceY in range(470,560):
            srcPx=getPixelAt(srcPic,sourceX,sourceY)
            srcColor = getColor(srcPx)
            targetPx=getPixelAt(targetPic,targetX,targetY)
            setColor(targetPx, srcColor)
            targetY = targetY + 1
            targetX = targetX + 1
    return targetPic
for sourceX in range(320,420):
    targetY = 0
for sourceY in range(520,625):
    srcPx = getPixelAt(srcPic, sourceX, sourceY)
    srcColor = getColor(srcPx)
    targetPx = getPixelAt(targetPic, targetX, targetY)
    setColor(targetPx, srcColor)
    targetY = targetY + 1
    targetX = targetX + 1
return targetPic

# Below are the color and image manipulation functions, labeled accordingly...

def negative(pic):
    newPic = duplicatePicture(pic)
    for px in getPixels(newPic):
        r = getRed(px)
        b = getBlue(px)
        g = getGreen(px)
        neg = makeColor(255 - r, 255 - g, 255 - b)
        setColor(px, neg)
    return newPic

def reverse(sourcePic):
    width = getWidth(sourcePic)
    height = getHeight(sourcePic)
    targetPic = makeEmptyPicture(width, height, white)
    targetX = width - 1
    for x in range(0, getWidth(sourcePic)):
        targetY = 0
        for y in range(0, getHeight(sourcePic)):
            pixel = getPixelAt(sourcePic, x, y)
            tx = getPixel(targetPic, targetX, targetY)
            setColor(tx, getColor(pixel))
            targetY = targetY + 1
        targetX = targetX - 1
    return (targetPic)

def maxBlue(pic):
    newPic = duplicatePicture(pic)
    for px in getAllPixels(newPic):
        b = getBlue(px)
        setBlue(px, b * 15)
    return newPic

def maxRed(pic):
    newPic = duplicatePicture(pic)
    for px in getAllPixels(newPic):
        r = getRed(px)
        setRed(px, r * 15)
    return newPic

def makePurple(pic):
    newPic = duplicatePicture(pic)
    for px in getAllPixels(newPic):
        redValue = getRed(px)
        blueValue = getBlue(px)
        setRed(px, redValue * 5)
        setBlue(px, blueValue * 5)
    return newPic
def grayScale(pic):
    newPic = duplicatePicture(pic)
    for px in getAllPixels(newPic):
        redValue = getRed(px)
        greenValue = getGreen(px)
        blueValue = getBlue(px)
        grayValue = (redValue + greenValue + blueValue)/3.0
        myGrayColor = makeColor(grayValue, grayValue, grayValue)
        setColor(px, myGrayColor)
    return newPic

#The below function is one of my own creation. It takes the blues
#in the image and makes them red, while also taking the greens
#and making them black, while also manipulating the light and darkness
#of the image.

def makeDistorted(pic):
    newPic = duplicatePicture(pic)
    for x in range (0, getWidth(newPic)):
        for y in range (0, getHeight(newPic)):
            currentPx = getPixelAt(newPic, x, y)
            currentColor = getColor(currentPx)
            if (distance(currentColor, green) < 225):
                setColor(currentPx, black)
            if (distance(currentColor, blue) < 200):
                setColor(currentPx, red)
    for px in getAllPixels(newPic):
        newCurrentColor = getColor(px)
        darkerNCC = makeDarker(newCurrentColor)
        lighterNCC = makeLighter(darkerNCC)
        setColor(px, lighterNCC)
    return newPic

def chromakeySig(sigPic, canvas, targetX, targetY):
    targetPic = duplicatePicture(canvas)
    for sX in range (0, getWidth(sigPic)):
        for sY in range (0, getHeight(sigPic)):
            sPx = getPixelAt(sigPic, sX, sY)
            sColor = getColor(sPx)
            targetPx = getPixelAt(targetPic, sX + targetX, sY + targetY)
            if distance(black, sColor) < 180:
                setColor(targetPx, black)
    return targetPic

#If you made it this far, you get some chocolate!

"""