

# Tobi Lott

Completed



Originals



```
def collage():
    # written by: Tobi Lott
    #written: October 18, 2021
    picture = makeEmptyPicture(640, 904, black)
    #creates border, pictures, and centerpictures
    border(picture)
    mural = makePicture(getMediaPath("Mural.jpg"))
    floral = makePicture(getMediaPath("Flowerfence.jpg"))
    newMural = makeEmptyPicture(getWidth(mural)/2, getHeight(mural)/2)
    newFloral = makeEmptyPicture(getWidth(floral)/2, getHeight(floral)/2)
    combinedPicture = makeEmptyPicture(getWidth(newFloral), getHeight(newFloral))
    newCombinedPicture = makeEmptyPicture(384, 670)
    centerPictures(mural, floral, newMural, newFloral, combinedPicture, newCombinedPicture)
    #copies all elements into one picture
    for x in range(0, 384):
        for y in range(0, 640):
            color = getColor(getPixel(newCombinedPicture, x, y))
            setColor(getPixel(picture, x + 128, y + 132), color)
    signature = makePicture(getMediaPath("signature.jpeg"))
    sign(signature, picture)
    explore(picture)

def centerPictures(mural, floral, newMural, newFloral, combinedPicture, newCombinedPicture):
    scale(mural, newMural, 2)
    scale(floral, newFloral, 2)
    wF = getWidth(newFloral)
    hF = getHeight(newFloral)
```

```

wM = getWidth(newMural)
hM = getHeight(newMural)
barWF = wF / 5
barWM = wM / 5
merge(newFloral, combinedPicture, wF/2, wF, 0, hF/2, 0)
merge(newFloral, combinedPicture, 0, wF/2, hF/2, hF, 0)
merge(newMural, combinedPicture, 0, wF/2, 0, hF/2, 0)
merge(newMural, combinedPicture, wM - wF/2, wM, hF/2, hF, 38)
grayScale(combinedPicture, 0, wF/2, 0, hF/2)
redScale(combinedPicture, wF/2, wF, 0, hF/2)
greenScale(combinedPicture, 0, wF/2, hF/2, hF)
blueScale(combinedPicture, wF/2, wF, hF/2, hF )
copy(combinedPicture, newCombinedPicture, 48, 432, 0, 640, -48, 0)
return newCombinedPicture

```

```

def border(border):
    #scales the butterfly picture
    butterfly = makePicture(getMediaPath("butterfly2.jpg"))
    wB = getWidth(butterfly)
    hB = getHeight(butterfly)
    newButterfly = makeEmptyPicture(wB/15, int(hB/15))
    nWB = getWidth(newButterfly)
    nHB = getHeight(newButterfly)
    scale(butterfly, newButterfly, 15)
    background = makePicture(getMediaPath("Flowerfence.jpg"))
    copy(newButterfly, border, 0, nWB, 0, nHB, 0, nHB*3)
    #changes the background color of the butterfly
    changeColor(newButterfly, 200, 200, 200, makeColor(0, 100, 89), black, 20, 20, 20)
    copy(newButterfly, border, 0, nWB, 0, nHB, nWB, 0)
    #resets butterfly to the original colors and changes the butterfly to the new color
    changeColor(newButterfly, 0, 100, 89, white, black, 20, 20, 20)
    changeColor(newButterfly, 200, 200, 200, white, makeColor(0, 100, 89), 20, 20, 20)
    copy(newButterfly, border, 0, nWB, 0, nHB, nWB*2, 0)
    #resets butterfly to original colors
    changeColor(newButterfly, 0, 100, 89, white, black, 0, 100, 89)
    copy(newButterfly, border, 0, nWB, 0, nHB, 0, nHB)
    #edits the colors in the butterfly picture to blue scale
    blueScale(newButterfly, 0, nWB, 0, nHB)
    copy(newButterfly, border, 0, nWB, 0, nHB, 0, nHB*2)
    #resets butterfly picture back to original picture and puts the floral picture as the butterfly
    changeColor(newButterfly, 170, 170, 255, white, black, 20, 20, 20)
    chromakey(newButterfly, background, 323, 555)
    copy(newButterfly, border, 0, nWB, 0, nHB, 0, 0)
    #mirrors the border all around
    mirrorVertical(border)

```

```

mirrorHorizontal(border)
return border

def sign(picture, combinedPicture):
    #adds the signature to the bottom
    sign0 = makeEmptyPicture(1000, 1000)
    rotate(picture, sign0)
    sign1 = makeEmptyPicture(200, 200)
    scale(sign0, sign1, 5)
    sign2 = makeEmptyPicture(640, 904)
    copy(sign1, sign2, 0, 200, 0, 100, 440, 804)
    chromakey2(sign2, combinedPicture)
    return combinedPicture

def scale(picture, canvas, scale):
    sourceX = 0
    for x in range(0, getWidth(picture)/scale):
        sourceY = 0
        for y in range(0, int(getHeight(picture)/scale)):
            px = getPixel(picture, sourceX, sourceY)
            color = getColor(px)
            setColor(getPixel(canvas, x, y), color)
            sourceY = sourceY + scale
            sourceX = sourceX + scale
        return canvas

def grayScale(picture, widthStart, widthStop, heightStart, heightStop):
    for x in range(widthStart, widthStop):
        for y in range(heightStart, heightStop):
            p = getPixel(picture, x, y)
            intensity = (getRed(p) + getGreen(p) + getBlue(p)) / 3
            setColor(p, makeColor(intensity, intensity, intensity))
    return picture

def redScale(picture, widthStart, widthStop, heightStart, heightStop):
    #lowers the intensity of all the colors except red
    for x in range(widthStart, widthStop):
        for y in range(heightStart, heightStop):
            p = getPixel(picture, x, y)
            intensity = (getGreen(p) + getBlue(p)) / 3
            setColor(p, makeColor(getRed(p), intensity, intensity))
    return picture

def greenScale(picture, widthStart, widthStop, heightStart, heightStop):

```

```

#lowers the intensity of all the colors except green
for x in range(widthStart, widthStop):
    for y in range(heightStart, heightStop):
        p = getPixel (picture, x, y)
        intensity = (getRed(p) + getBlue(p)) / 3
        setColor(p, makeColor(intensity, getGreen(p), intensity))
return picture

def blueScale (picture, widthStart, widthStop, heightStart, heightStop):
    #lowers the intensity of all the colors except blue
    for x in range(widthStart, widthStop):
        for y in range(heightStart, heightStop):
            p = getPixel (picture, x, y)
            intensity = (getRed(p) + getGreen(p)) / 3
            setColor(p, makeColor(intensity, intensity, getBlue(p)))
    return picture

def merge (picture, newPicture, widthStart, widthStop, heightStart, heightStop, offset):
    for x in range(widthStart, widthStop):
        for y in range(heightStart, heightStop):
            px = getPixel (picture, x, y)
            color = getColor (px)
            px2 = getPixel (newPicture, x + offset, y)
            setColor (px2, color)
    return newPicture

def copy (picture, newPicture, widthStart, widthStop, heightStart, heightStop, xOffset, yOffset):
    for x in range(widthStart, widthStop):
        for y in range(heightStart, heightStop):
            color = getColor (getPixel (picture, x, y))
            setColor (getPixel (newPicture, x + xOffset, y + yOffset), color)
    return border

def mirrorVertical (picture):
    mirrorPoint = getWidth (picture) / 2
    width = getWidth (picture)
    for y in range (0, getHeight (picture)):
        for x in range (0, mirrorPoint):
            leftPixel = getPixel (picture, x, y)
            rightPixel = getPixel (picture, width - x - 1, y)
            color = getColor (leftPixel)
            setColor (rightPixel, color)
    return picture

def mirrorHorizontal (picture):

```

```

mirrorPoint = getHeight(picture) / 2
height = getHeight(picture)
for x in range(0, getWidth(picture)):
    for y in range(0, mirrorPoint):
        topPixel = getPixel(picture, x, y)
        bottomPixel = getPixel(picture, x, height - y - 1)
        color = getColor(topPixel)
        setColor(bottomPixel, color)
return picture

def chromakey(picture, background, xoffset, yoffset):
    #moves background to picture
    for x in range(0, getWidth(picture)):
        for y in range(0, getHeight(picture)):
            p = getPixel(picture, x, y)
            r = getRed(p)
            b = getBlue(p)
            g = getGreen(p)
            if ((r < 10) and (b < 10) and (g < 10)):
                bgpx = getPixel(background, x + xoffset, y + yoffset)
                bgcol = getColor(bgpx)
                setColor(p, bgcol)
    return picture

def chromakey2(picture, background):
    #moves picture to background
    for x in range(0, getWidth(picture)):
        for y in range(0, getHeight(picture)):
            p = getPixel(picture, x, y)
            r = getRed(p)
            b = getBlue(p)
            g = getGreen(p)
            if ((r < 10) and (b < 10) and (g < 10)):
                bgpx = getPixel(background, x, y)
                color = getColor(p)
                setColor(bgpx, color)
    return background

def changeColor(picture, rSet, gSet, bSet, color, color2, rSet2, gSet2, bSet2):
    for x in range(0, getWidth(picture)):
        for y in range(0, getHeight(picture)):
            p = getPixel(picture, x, y)
            r = getRed(p)
            b = getBlue(p)
            g = getGreen(p)

```

```
    if (r >= rSet) and (b >= bSet) and (g >= gSet):
        setColor(p, color)
    if (r <= rSet2) and (b <= bSet2) and (g <= gSet2):
        setColor(p, color2)
return picture

def rotate(picture, canvas):
    targetX = 0
    for x in range(0, getWidth(picture)):
        targetY = 0
        for y in range(0, getHeight(picture)):
            color = getColor(getPixel(picture, x, y))
            setColor(getPixel(canvas, targetY, getWidth(picture) - targetX - 1), color)
            targetY = targetY + 1
        targetX = targetX + 1
    return canvas
```