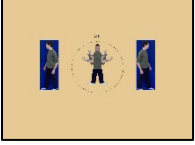


# Beckett Neal

Completed



Originals



```
from jes4py import *
from math import radians,cos,sin
```

```
# Made By Beckett Neal
# 10/21/24
# Project2
# Titled: The Vitruvian Collage
```

```
def collage():
    setMediaPath()
    torso = makePicture(getMediaPath("mLeft1.jpg"))
    facearmleg = makePicture(getMediaPath("mScary.jpg"))
    sign = makePicture(getMediaPath("Sign.png"))
    specialcolor = makeColor(231,201,147)
    art = makeEmptyPicture(1000, 736,specialcolor)
    facemake(facearmleg,art)
    armsmake(facearmleg,art)
    armsmake2(facearmleg,art)
    torsomake(torso, art)
    legsmake(torso,art)
    finishtouch(art,torso)
    mirror(art)
    makecircle(art)
    signed = signature(art, specialcolor, sign, offset_x=480, offset_y=180)
    explore(art)
```

```
def signature(art, specialcolor, sign, offset_x=0, offset_y=0):
    #ReverseChromaKey signature
    sign_width = getWidth(sign)
    sign_height = getHeight(sign)
    for x in range(sign_width):
        for y in range(sign_height):
            signpix = getPixel(sign, x, y)
            artpix = getPixel(art, x + offset_x, y + offset_y)
            r=getRed(signpix)
            g=getGreen(signpix)
            b=getBlue(signpix)
            if r == 255:
                if g == 255:
                    if b ==255:
                        setColor(artpix,specialcolor)
            else:
                setColor(artpix,black)
    return art
```

```
def makecircle(art):
    #utilizing the math module and the equation to make a circle,
    #I was able to calculate an orgin point, utilize a radius,
    #then using my knowledge of said circle equation, make a function
```

```

#that utilizes the math cosine and sine to find the positioning of a pixel
#needed to make a circle in the required given field.
centerheight = getHeight(art)//2 -10 #358
centerwidth = getWidth(art)//2 #500
radius = 135
for i in range (0,360):
    theta = radians(i)
    x = round(centerwidth + radius *cos(theta))
    y = round(centerheight + radius *sin(theta))
    pixel = getPixel(art,x,y)
    setColor(pixel,black)

def finishtouch(art,torso):
    #this function is just making sure to clean up unwanted pixels from the crops,
    #and to also add the original image into the collage.
    height = getHeight(art)
    width = getWidth(art)
    specialcolor = makeColor(231,201,147)
    for x in range(0,width):
        for y in range(0,height):
            pixel = getPixel(art,x,y)
            bluecolor = int(getBlue(pixel))
            greencolor = int(getGreen(pixel))
            redcolor = int(getRed(pixel))
            if bluecolor > 60:
                if redcolor <= 45:
                    if greencolor <= 60:
                        setColor(pixel,specialcolor)
    startx = getWidth(art)//2 - 500 +200
    starty = 215
    for x in range(0,getWidth(torso)):
        for y in range(0,getHeight(torso)):
            pixel = getPixel(torso,x,y)
            color = getColor(pixel)
            newx = startx + (x -0)
            newy = starty + (y - 0)
            copypix = getPixel(art,newx,newy)
            setColor(copypix,color)

def mirror(picture):
    #this was our mirror function in an old assignment utilized for the entire
    #collage.
    width = getWidth(picture)
    mirror_point = width // 2
    for x in range(0, mirror_point):
        for y in range(0, getHeight(picture)):
            left_pixel = getPixel(picture, x, y)
            right_pixel = getPixel(picture, width - x - 1, y)
            color = getColor(left_pixel)
            setColor(right_pixel, color)

def facemake(pic, art):
    #this function replicates the face from an image and puts it into collage.
    facewidth = 45-33
    faceheight = 40-0
    art_center_x = getWidth(art) // 2
    art_center_y = getHeight(art) // 2

    startx = art_center_x - facewidth
    starty = art_center_y - faceheight//2 -116

```

```

for x in range(33,45):
    for y in range(0,40):
        pixel = getPixel(pic,x,y)
        color = getColor(pixel)
        newx = startx + (x - 33)
        newy = starty + (y - 0)
        copypix = getPixel(art,newx,newy)
        setColor(copypix,color)

def torsomake(pic, art):
    #this function takes half of a torso and places it in the center offset of
    #the art canvas.
    torsowidth = 60 - 40
    torsoheight = 148 - 55
    art_center_x = getWidth(art) // 2
    art_center_y = getHeight(art) // 2
    #align torso
    startx = art_center_x - torsowidth
    starty = art_center_y - torsoheight // 2 - 40

    for x in range(40, 60):
        for y in range(55, 148):
            pixel = getPixel(pic,x,y)
            color = getColor(pixel)
            newx = startx + (x - 40)
            newy = starty + (y - 55)
            copypix = getPixel(art,newx,newy)
            setColor(copypix,color)

def armsmake(pic, art):
    #this function takes the arms of said picture, and places them into the canvas.
    armwidth = 54 - 6
    armheight = 100 - 40
    art_center_x = getWidth(art) // 2
    art_center_y = getHeight(art) // 2
    #align arms
    startx = art_center_x - armwidth
    starty = art_center_y - armheight

    for x in range(6, 54):
        for y in range(40, 100):
            pixel = getPixel(pic,x,y)
            color = getColor(pixel)
            newx = startx + (x - 6)
            newy = starty + (y - 40) -40
            copypix = getPixel(art,newx,newy)
            setColor(copypix,color)

def armsmake2(pic,art):
    #this one was tough. I took the same arm function, except this time I needed
    #to straighten the arms out. My first thought was to draw a 3x3 of pixels on my
    #board and trying to figure out how to adjust the x and y accordingly. I
    #struggled. I realized however I could use similar logic of the circle function
    #onto the arms, except changing the radians needed for the degree needed.
    armwidth = 54 - 6
    armheight = 100 - 40
    art_center_x = getWidth(art) // 2
    art_center_y = getHeight(art) // 2
    startx = art_center_x - armwidth

```

```

starty = art_center_y - armheight

theta = radians(-30) #radians for arms

#center of "armbox"
x_c = (6 + 54) / 2
y_c = (40 + 100) / 2

for x in range(6, 54):
    for y in range(40, 100):
        pixel = getPixel(pic, x, y)
        color = getColor(pixel)
        x_translated = x - x_c #math distance between x cord, and center of arm
        y_translated = y - y_c #vice versa with y
        #utilize sin and cos to find position of new cord
        x_rotated = (x_translated * cos(theta)) - (y_translated * sin(theta))
        y_rotated = (x_translated * sin(theta)) + (y_translated * cos(theta)) #^
        newx = round(startx + x_rotated)
        newy = round(starty + y_rotated) +15
        copypix = getPixel(art, newx, newy)
        setColor(copypix, color)

def legsmake(pic, art):
    #function to create legs
    legswidth = 54 - 0
    legsheight = 260 - 184
    art_center_x = getWidth(art) // 2
    art_center_y = getHeight(art) // 2
    #leg align
    startx = art_center_x - legswidth
    starty = art_center_y - legsheight // 2

    for x in range(0, 54):
        for y in range(184, 260):
            pixel = getPixel(pic,x,y)
            color = getColor(pixel)
            newx = startx + (x - 0)
            newy = starty + (y - 184) +40
            copypix = getPixel(art,newx,newy)
            setColor(copypix,color)

```