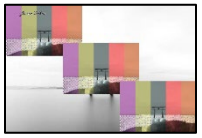
Jovan Dukes

Completed



Originals

```
from jes4py import*

#Jovan Dukes
#10/21/2024
#CS 120

picture1 = makePicture(getMediaPath("japan.webp"))
picture = makePicture(getMediaPath("japan.webp"))
picture2 = makePicture(getMediaPath("nasa-rTZW4f02zY8-unsplash.jpg"))
picture3 = makePicture(getMediaPath("jeremy-thomas-E0AHdsENmDg-unsplash.jpg"))
picture4 = makePicture(getMediaPath("matteo-bernardis-QpIayO5KIRE-unsplash.jpg"))
picture5 = makePicture(getMediaPath("signature.png"))

def Mainfunction(picture, picture1, picture2, picture3, picture4, picture5):
    pic = greenScreen(picture1)
    water = merg_pics(picture4, picture3)
    result = chromakey1(pic, water)
    result = chromakey2(result, picture2)
    result = changeTints(result)

    result = SingleScale(result, 4/10)
    result = blur(result)
    result1 = copy_Picture_to_picture(picture, result, 0, 0)

    x_dest = (getWidth(picture) - getWidth(result)) // 2
    y_dest = (getHeight(picture) - getHeight(result)) // 2
    result2 = copy_Picture_to_picture(result1, result, x_dest, y_dest)

    x_dest = (getWidth(picture) - getWidth(result))
    y_dest = (getHeight(picture) - getHeight(result))
    result3 = copy_Picture_to_picture(result2, result, x_dest, y_dest)
    signature = SingleScale(picture5, 1/3)
    signed_photo = reverseChromakey(signature, result3)
    explore(signed_photo)
```

```
def blur(picture):
    blur_picture = duplicatePicture(picture)
    for x in range(1, getWidth(picture) - 1):
        for y in range(1, getHeight(picture) - 1):
            center = getPixel(blur_picture, x, y)
            left = getPixel(picture, x - 1, y)
            right = getPixel(picture, x + 1, y)
            top = getPixel(picture, x, y - 1)
            bottom = getPixel(picture, x, y + 1)
            new_red = (getRed(center) + getRed(left) + getRed(right) + getRed(top) + getRed(bottom)) // 5
            new_green = (getGreen(center) + getGreen(left) + getGreen(right) + getGreen(top)+getGreen(bottom))// 5
            new_blue = (getBlue(center) + getBlue(left) + getBlue(right) + getBlue(top) + getBlue(bottom)) // 5
            setColor(center, makeColor(new_red, new_green, new_blue))
    return blur_picture

def SingleScale(source_picture, scale_factor):
    new_width = int(getWidth(source_picture) * scale_factor)
    new_height = int(getHeight(source_picture) * scale_factor)
    new_picture = makeEmptyPicture(new_width, new_height)
    source_x = 0
    for new_x in range(0, getWidth(new_picture)):
        source_y = 0
        for new_y in range(0, getHeight(new_picture)):
            color = getColor(getPixel(source_picture, source_x, source_y))
            setColor(getPixel(new_picture, new_x, new_y), color)
            source_y = int(new_y / scale_factor)
        source_x = int(new_x / scale_factor)
    return new_picture

def copy_Picture_to_picture(canvas, picture_out, target_x, target_y):
    original_target_y = target_y
    for source_x in range(0, getWidth(picture_out)):
        target_y = original_target_y
        for source_y in range(0, getHeight(picture_out)):
            source_pixel = getPixel(picture_out, source_x, source_y)
            color = getColor(source_pixel)
            target_pixel = getPixel(canvas, target_x, target_y)
            setColor(target_pixel, color)
            target_y += 1
        target_x += 1
    return(canvas)
```

```
def applyTint(picture, x_start, x_end, red_tint, green_tint, blue_tint):
    for x in range(x_start, x_end):
        for y in range(getHeight(picture)):
            pixel = getPixel(picture, x, y)
            setRed(pixel, min(255, getRed(pixel) * red_tint))
            setGreen(pixel, min(255, getGreen(pixel) * green_tint))
            setBlue(pixel, min(255, getBlue(pixel) * blue_tint))
    return picture

def changeTints(picture):
    width = getWidth(picture)
    bar_width = width // 5
    picture = grayScale(picture)
    picture = darken(picture)
    picture = applyTint(picture, 0, bar_width, 1.5, 1, 1.5 )
    picture = applyTint(picture, bar_width, bar_width * 2, 1.5, 1.5, 1)
    picture = applyTint(picture, bar_width * 2, bar_width * 3, 1, 1.1, 1.1)
    picture = applyTint(picture, bar_width * 3, bar_width * 4, 2, 1, 1)
    picture = applyTint(picture, bar_width * 4, width, 1.8, 1.2, 0.9)
    return picture

def grayScale(picture):
    for pixel in getPixels(picture):
        intensity = (getRed(pixel) + getGreen(pixel) + getBlue(pixel)) / 3
        setColor(pixel, makeColor(intensity, intensity, intensity))
    return picture

def greenScreen(picture):
    for x in range(getWidth(picture)):
        for y in range(265, getHeight(picture)):
            pixel = getPixel(picture, x, y)
            red = getRed(pixel)
            blue = getBlue(pixel)
            green = getGreen(pixel)
            if red >= 190 and blue >= 190 and green >= 190:
                setRed(pixel, 255)
                setBlue(pixel, 0)
                setGreen(pixel, 255)
            elif (red >= 100 and red < 190) and (blue >= 100 and blue < 190) and (green >= 100 and green < 190):
                setRed(pixel, 0)
                setBlue(pixel, 0)
                setGreen(pixel, 255)
    return(picture)
```

```
def darken(picture):
    for x in range(getWidth(picture)):
        for y in range(0, 265):
            pixel = getPixel(picture, x, y)
            red = getRed(pixel)
            blue = getBlue(pixel)
            green = getGreen(pixel)
            if red >= 190 and blue >= 190 and green >= 190:
                color = getColor(pixel)
                color = makeDarker(makeDarker(color))
                setColor(pixel, color)
    return picture

def chromakey1(source, background):
    for source_pixel in getPixels(source):
        x = getX(source_pixel)
        y = getY(source_pixel)
        if (getRed(source_pixel) > 100 and getGreen(source_pixel) > 125 and getBlue(source_pixel) < 126 ):
            background_pixel = getPixel(background, x, y)
            background_color = getColor(background_pixel)
            setColor(source_pixel, background_color)
    return source

def chromakey2(source, background):
    for source_pixel in getPixels(source):
        x = getX(source_pixel)
        y = getY(source_pixel)
        if (getRed(source_pixel) == 0 and getGreen(source_pixel) == 255 and getBlue(source_pixel) == 0 ):
            background_pixel = getPixel(background, x, y)
            background_color = getColor(background_pixel)
            setColor(source_pixel, background_color)
    return source

def reverseChromakey(source, background):
    for source_pixel in getPixels(source):
        x = getX(source_pixel)
        y = getY(source_pixel)
        if not (getRed(source_pixel) == 255 and getGreen(source_pixel) == 255 and getBlue(source_pixel) == 255):
            background_pixel = getPixel(background, x, y)
            setColor(background_pixel, getColor(source_pixel))
    return background

def mergepic(source, destination, start_point):
    width = getWidth(source)
    height = getHeight(source)
```

```
    for x in range(start_point, width,2):
        for y in range(start_point, height, 2):
            color = getColor(getPixel(source,x,y))
            setColor(getPixel(destination, x, y), color)
    return destination

def merg_pics(source, destination):
    width = getWidth(source)
    height = getHeight(source)
    result = makeEmptyPicture(width, height)
    result = mergepic(source, result, 0)
    result = mergepic(destination, result, 1)
    return result
```