

Grace Sawicki

Completed

Originals



```
# Grace Sawicki, Project 2, October 17 2022
```

```
def collage():
    # setting variables needed
    setMediaPath("/Users/gracesawicki/Public/CS 120 Project 2- Grace Sawicki")
    picture = makePicture(getMediaPath("castle.png"))
    canvas = makeEmptyPicture(800,651)
    canvas2 = makeEmptyPicture(765,617)

    # scaling down and cropping main picture
    scaleDown(picture,canvas,5)
    copyFun(canvas,canvas2,17,782,16,633)

    # setting variable for chomakey picture
    setMediaPath("/Users/gracesawicki/Public/CS 120 Project 2- Grace Sawicki")
    waterPic = makePicture(getMediaPath("canalpainting.png"))
    waterCanvas = makeEmptyPicture(800,640)
    waterCanvas2 = makeEmptyPicture(780,617)

    # scaling and cropping water picture for chromakey
    scaleDown(waterPic,waterCanvas,5)
    copyFun(waterCanvas,waterCanvas2,10,790,11,628)

    # mirroring picture so water is at the top
    mirrorPic(waterCanvas2)
    explore(waterCanvas2)

    # lighten the water
    lighten(waterCanvas2)

    # chromakey the grass into water from another picture
    chromakey(canvas2,waterCanvas2,0,80,0,270,100,255)
    chromakey(canvas2,waterCanvas2,80,553,0,130,100,255)
    chromakey(canvas2,waterCanvas2,553,764,0,270,100,255)
    chromakey(canvas2,waterCanvas2,290,338,309,349,110,180)

    # grayscale on a diagonal
    grayScale(canvas2)

    # edge detect section on diagonal
    edgeDetect(canvas2,15)
```

```

# invert colors in a square
invert(canvas2,308,498,35,205)
invert(canvas2,455,530,259,334)
invert(canvas2,410,450,369,409)

# mirror bar from bottom to top
mirrorBar(canvas2,312,606,707)

# add edge detect boader to other areas with 5 pixel boarder
edgeDetectStraight(canvas2,15,303,308,30,210,black,white)
edgeDetectStraight(canvas2,15,498,503,30,210,black,white)
edgeDetectStraight(canvas2,15,303,503,30,35,black,white)
edgeDetectStraight(canvas2,15,303,503,205,210,black,white)

# add edge detect to second object with 5 pixel boarder
edgeDetectStraight(canvas2,15,601,606,0,278,white,black)
edgeDetectStraight(canvas2,15,707,712,0,278,white,black)
edgeDetectStraight(canvas2,15,601,712,273,278,white,black)

# add edge detect to third object with 5 pixel boarder
edgeDetectStraight(canvas2,15,450,455,254,339,black,white)
edgeDetectStraight(canvas2,15,530,535,254,339,black,white)
edgeDetectStraight(canvas2,15,450,535,254,259,black,white)
edgeDetectStraight(canvas2,15,450,535,334,339,black,white)

# add edge detect to fourth object with 5 pixel boarder
edgeDetectStraight(canvas2,15,405,410,364,414,black,white)
edgeDetectStraight(canvas2,15,450,455,364,414,black,white)
edgeDetectStraight(canvas2,15,405,455,364,369,black,white)
edgeDetectStraight(canvas2,15,405,455,409,414,black,white)

#add signature
# setting variables for signature
setMediaPath("/Users/gracesawicki/Public/CS 120 Project 2- Grace Sawicki")
pic2 = makePicture(getMediaPath("IMG_6219.jpeg"))
signature = makeEmptyPicture(100,100)

# scales down and adds sigature to canvas2
scaleDown(pic2,signature,6)
addSignature(canvas2,signature,625,543,white)

# explore the final image
explore(canvas2)

def scaleDown(picture_in,picture_out,scale):
# scales down picture
in_x = 0
for out_x in range(0,int(getWidth(picture_in)/ scale)):
    in_y = 0
    if (out_x <= (getWidth(picture_out) - 1)):
        for out_y in range(0,int(getHeight(picture_in)/ scale)):
            if (out_y <= (getHeight(picture_out) - 1)):
                color = getColor(getPixel(picture_in,in_x,in_y))
                setColor(getPixel(picture_out,out_x,out_y), color)
                in_y = in_y + scale
            in_x = in_x + scale

def copyFun(picture_in,picture_out,startX,endX,startY,endY):
# crops pictures to remove boarder
newX = 0

```

```

for x in range(startX,endX):
    newY = 0
    for y in range(startY,endY):
        pixel = getPixel(picture_in,x,y)
        newPixel = getPixel(picture_out,newX,newY)
        setColor(newPixel,getColor(pixel))
        newY = newY + 1
    newX = newX + 1

def chromakey(mainPic,bg,startX,endX,startY,endY,blueStart,blueEnd):
    # replaces a color spectrum with the water image
    for p in getPixels(mainPic):
        x = getX(p)
        y = getY(p)
        if ((startX <= x <= endX) and (startY <= y <= endY)):
            if (blueStart <= getBlue(p) <= blueEnd):
                bgpx = getPixel(bg,x,y)
                bgColor = getColor(bgpx)
                setColor(p,bgColor)

def mirrorPic(picture):
    # mirrors the water in the canal image to be at the top of the image
    height = getHeight(picture)
    mirrorPoint = height / 2
    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(bottomPixel)
            setColor(topPixel,color)

def lighten(picture):
    # lightens 1 time
    for x in range(0,getWidth(picture)):
        for y in range(0,getHeight(picture)):
            pixel = getPixel(picture,x,y)
            setColor(pixel,makeLighter(getColor(pixel)))

def grayScale(picture):
    # gray scales an image
    for x in range(0,getWidth(picture)):
        for y in range(0,getHeight(picture)):
            if (y >= ((2 * (x- 100)) + 74)):
                if (y >= (((2.3) * (x-100)) + 58)):
                    p = getPixel(picture,x,y)
                    intensity = (getRed(p)+getGreen(p)+getBlue(p))/3
                    setColor(p,makeColor(intensity,intensity,intensity))

def luminance(pixel):
    # used in edge detect function
    r = getRed(pixel)
    g = getGreen(pixel)
    b = getBlue(pixel)
    return (r+g+b)/3

def edgeDetect(source,threshold):
    # edge detects on the diagonal line
    for x in range(0,getWidth(source)-1):
        for y in range(0,getHeight(source)-1):
            number2 = (((2.3) * (x - 100)) + 58)

```

```

number1 = ((2 * (x - 100)) + 74)
if (y <= number2):
    number2 = (((2.3) * (x - 100)) + 58)
    number1 = ((2 * (x - 100)) + 74)
    if (y >= number1):
        botrt = getPixel(source,x+1,y+1)
        thislum = luminance(getPixel(source,x,y))
        brlum = luminance(botrt)
        if abs(brlum-thislum) > threshold:
            setColor(getPixel(source,x,y),black)
        if abs(brlum-thislum) <= threshold:
            setColor(getPixel(source,x,y),white)
for x in range(0,getWidth(source)-1):
    for y in range(0,getHeight(source)-1):
        number2 = (((2.3) * (x - 100)) + 58)
        number1 = ((2 * (x - 100)) + 74)
        if (y <= number1):
            number2 = (((2.3) * (x - 100)) + 58)
            number1 = ((2 * (x - 100)) + 74)
            if (y >= number2):
                botrt = getPixel(source,x+1,y+1)
                thislum = luminance(getPixel(source,x,y))
                brlum = luminance(botrt)
                if abs(brlum-thislum) > threshold:
                    setColor(getPixel(source,x,y),white)
                if abs(brlum-thislum) <= threshold:
                    setColor(getPixel(source,x,y),black)

def invert(picture,startX,endX,startY,endY):
    # inverts the colors in an image
    for x in range(startX,endX):
        for y in range(startY,endY):
            p = getPixel(picture,x,y)
            red = getRed(p)
            green = getGreen(p)
            blue = getBlue(p)
            negColor = makeColor(255-red, 255-green, 255-blue)
            setColor(p,negColor)

def mirrorBar(picture,yCord,startX,endX):
    # mirrors part of an image while making cutting it off before the mirror point
    mirrorPoint = yCord
    for x in range(startX,endX):
        for y in range(0,273):
            pbottom = getPixel(picture,x,y)
            ptop = getPixel(picture,x,mirrorPoint + (mirrorPoint-7) - 1 - y)
            intensity = (getRed(ptop)+getGreen(ptop)+getBlue(ptop))/3
            setColor(pbottom,makeColor(intensity,intensity,intensity))
    return picture

def edgeDetectStraight(source,threshold,startX,endX,startY,endY,option1,option2):
    # edge detects in straight lines
    for x in range(startX,endX):
        for y in range(startY,endY):
            p = getPixel(source,x,y)
            botrt = getPixel(source,x+1,y+1)
            thislum = luminance(p)
            brlum = luminance(botrt)
            if abs(brlum-thislum) > threshold:
                setColor(p,option1)

```

```
    if abs(brlum-thislum) <+ threshold:
        setColor(p,option2)

def addSignature(target,signature,toX,toY,color):
    # adds the signature from a white background onto the image
    toYStart = toY
    for x in range(0, getWidth(signature)):
        toY = toYStart
        for y in range(0,getHeight(signature)):
            p = getPixel(signature,x,y)
            if (getRed(p) < 225 and getGreen(p) < 225 and getBlue(p) < 225):
                setColor(getPixel(target, toX, toY), color)
            toY = toY + 1
        toX = toX + 1
    return target
```