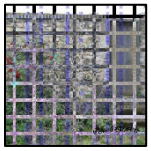
# Sam Pettrone

Completed



Original



```
#Sam Pettrone
#Started on 10/23/2026

from mediaComp import *

def collage():
    setMediaFolder()
    pic=makePicture(getMediaFolder("gardenwall.jpg"))
    signature=makePicture(getMediaFolder("signature.jpg"))
    newPic=makeEmptyPicture(714,714)
    #I wanted the collage to be a square so I made it the largest square size allowed that was divisible by 42.
    #42 was chosen kind of arbitrarily, I just wanted a decent sized grid.
    #And 17 is being used in the functions because 714/42=17 so each square in the grid is 17x17 pixels.
    for gridColumn in range(42):
        for gridRow in range(42):
            #These if statements use modulo, also known as the remainder function. I kind of picked the numbers
            #arbitrarily. They provide the "tic-tac-toe board" pattern on the grid.
            if(gridColumn%4==2 or gridRow%9==3):
                rotate(pic,newPic,17*gridColumn,17*gridRow,17*(gridColumn+1),17*(gridRow+1),
                            17*gridColumn,17*gridRow)
            else:
                copyOver(pic,newPic,17*gridColumn,17*gridRow,17*(gridColumn+1),17*(gridRow+1),
                            17*gridColumn,17*gridRow)
            if(gridColumn%6==2 or gridRow%5==3):
                invert(newPic,17*gridColumn,17*gridRow,17*(gridColumn+1),17*(gridRow+1),17*gridColumn,17*gridRow)
            elif(gridColumn%5==2 and gridRow%8!=2):
                grayscale(newPic,17*gridColumn,17*gridRow,17*(gridColumn+1),17*(gridRow+1),
                            17*gridColumn,17*gridRow)
            elif(gridColumn%4==2 and gridRow%11!=1):
                cyanotype(newPic,17*gridColumn,17*gridRow,17*(gridColumn+1),17*(gridRow+1),
                            17*gridColumn,17*gridRow)
    reverseChromakey(signature,newPic,0,0,300,144,414,570)
    pictureTool(newPic)
```

```
#Below are all the image modification functions. They may all look VERy similar (as I literally copied and pasted
#the same code 5 times and then applied modifications as needed),
#but I couldn't condense them into one function. They're all different enough that it would be difficult to
#condense them, and if I did, it would make the code harder to read.

#Made the rotate function myself
def rotate(originalPic, newPic, startX, startY, endX, endY, targetX, targetY):
    width=endX-startX
    height=endY-startY
    for x in range(width):
        for y in range(height):
            target=getPixelAt(newPic, targetX+y, targetY+x)
            current=getPixelAt(originalPic,x+startX,y+startY)
            setColor(target,getColor(current))

def copyOver(originalPic, newPic, startX, startY, endX, endY, targetX, targetY):
    width=endX-startX
    height=endY-startY
    for x in range(width):
        for y in range(height):
            target=getPixelAt(newPic, targetX+x, targetY+y)
            current=getPixelAt(originalPic,x+startX,y+startY)
            setColor(target,getColor(current))

def invert(pic, startX, startY, endX, endY, targetX, targetY):
    width=endX-startX
    height=endY-startY
    for x in range(width):
        for y in range(height):
            target=getPixelAt(pic, targetX+x, targetY+y)
            setRed(target,255-getRed(target))
            setGreen(target,255-getGreen(target))
            setBlue(target,255-getBlue(target))

def grayscale(pic, startX, startY, endX, endY, targetX, targetY):
    width=endX-startX
    height=endY-startY
    for x in range(width):
        for y in range(height):
            target=getPixelAt(pic, targetX+x, targetY+y)
            color=(getRed(target)+getGreen(target)+getBlue(target))//3
            setColor(target,makeColor(color))
```

```
def cyanotype(pic, startX, startY, endX, endY, targetX, targetY):
    width=endX-startX
    height=endY-startY
    for x in range(width):
        for y in range(height):
            target=getPixelAt(pic, targetX+x, targetY+y)
            color=(getRed(target)+getGreen(target)+getBlue(target))//3
            setColor(target,makeColor(color))
            setRed(target,color*.75)
            setGreen(target,color*.75)
            blue_value=color
            if blue_value < 63:
                blue_value = blue_value * 2
            elif blue_value < 192:
                blue_value = blue_value * 1.3
            else:
                blue_value = blue_value * 1.2
            if blue_value > 255:
                blue_value = 255
            setBlue(target,blue_value)

def reverseChromakey(originalPic, newPic, startX, startY, endX, endY, targetX, targetY):
    width=endX-startX
    height=endY-startY
    for x in range(width):
        for y in range(height):
            target=getPixelAt(newPic, targetX+x, targetY+y)
            current=getPixelAt(originalPic,x+startX,y+startY)
            if(distance(getColor(current),white)>200):
                setColor(target,getColor(current))
```