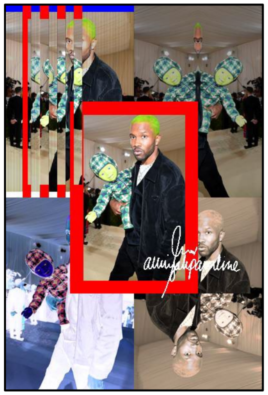
Alliyah Martin

Completed



Originals

```
#alliyah October 17
def collage () :
  setMediaPath()
  original = makePicture (getMediaPath ("frank.jpeg"))
  mod1 = makePicture (getMediaPath ("frank.jpeg"))
  mod2 = makePicture (getMediaPath ("frank.jpeg"))
  mod3 = makePicture (getMediaPath ("frank.jpeg"))
  mod4 = makePicture (getMediaPath ("frank.jpeg"))
  mod5 = makePicture (getMediaPath ("frank.jpeg"))

  color = makeColor (34, 145, 230)
  width = getWidth (original)
  canvas = makeEmptyPicture (getWidth (original) * 2, getHeight (original), color)
  copySlice (mod5, 0, int(width * 1/15), canvas, 0, 0)
  for i in range (15):
    copySlice (mod5, int(width * i/15), int(width * (i + 1) / 15), canvas, int (width * (i * 2)/15) ,0)


  canvas = makeEmptyPicture (getWidth (original)*2, getHeight(original)*2)
  sig = makePicture (getMediaPath ("sig.jpg"))


  contrast (mod1)
  redBoarder (mod5)
  darken (mod2)
  mirrorVertical(mod2)

  negative (mod3)
  increaseBlue (mod3)

  sepia (mod4)
  mirror (mod4)

  greek (mod1)


  copy (mod1, canvas, 0, 0)
  copy (mod2, canvas, getWidth(mod2), 0)
  copy (mod3, canvas, 0, getHeight (mod3))
  copy (mod4, canvas, getWidth(mod4), getHeight (mod4))
  copy (mod5,canvas,int(getWidth(canvas)/2-getWidth(original)/2),int(getHeight(canvas)/2-getHeight(original)/2))


  copySlice (mod5, int(width * 1/15), int (width * 2/15), canvas, int(width * 2/15), 0)
  copySlice (mod5, int(width * 2/15), int (width * 3/15), canvas, int(width * 4/15), 0)
  copySlice (mod5, int(width * 3/15), int (width * 4/15), canvas, int(width * 6/15), 0)
  copySlice (mod5, int(width * 4/15), int (width * 5/15), canvas, int(width * 8/15), 0)
  chromaSig (sig, canvas, 199, 304)
  explore (canvas)
  writePictureTo (canvas, "alliyah_pauline.jpg")
  show(canvas)



def copy(pic, target, targX, targY):
  targetX = targX
  for x in range(0, getWidth(pic)):
    targetY = targY
    for y in range(getHeight(pic)):
      px = getPixel(pic, x, y)
      tx = getPixel(target,targetX,targetY)
      setColor(tx, getColor(px))
      targetY = targetY + 1
    targetX = targetX + 1

def copySlice (pic, sliceStart, sliceEnd, target, targX, targY):
  targetX = targX
  for x in range (sliceStart, sliceEnd):
    targetY = targY
    for y in range (getHeight (pic)):
      pixel = getPixel (pic, x, y)
      tx = getPixel (target, targetX, targetY)
      setColor (tx, getColor(pixel))
      targetY = targetY + 1
    targetX = targetX + 1


#mod5
def contrast(pic):
  pixels=getPixels(pic)
  for i in range(0,len(pixels)):
    px=pixels[i]
    color=getColor(px)
    if i % 2==0:
      color=makeLighter(color)
    if i % 2==1:
      color=makeDarker(color)
    setColor(px,color)

def redBoarder(pic):
  bottom=getHeight(pic)-20
  rightSide=getWidth(pic)-20
  for pix in getPixels(pic):
    y=getY(pix)
    x=getX(pix)
    if x>=rightSide:
      setColor(pix,red)
    if x<21:
      setColor(pix,red)
    if y>=bottom:
      setColor(pix,red)
    if y<21:
      setColor(pix,red)


#mod1
def greek(pic):
  bottom=getHeight(pic)-10
  for pix in getPixels(pic):
    y=getY(pix)
    if y<10:
      setColor(pix,blue)
    if y>bottom:
      setColor(pix,white)
```

```
#mod2
def darken(pic):
  for px in getPixels(pic):
     color = getColor(px)
     darkerColor = makeDarker(color)
     setColor(px, darkerColor)

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

#mod3
def negative(pic):
  for px in getPixels(pic):
    red=getRed(px)
    green=getGreen(px)
    blue=getBlue(px)
    negColor=makeColor(255-red, 255-green, 255-blue)
    setColor(px, negColor)

def increaseBlue(picture):
  for p in getPixels(picture):
    value = getBlue(p)
    setBlue (p, value * 1.2)




#mod4
def grayScaleNew(picture):
  for px in getPixels(picture):
    newRed = getRed(px) * 0.299
    newGreen = getGreen(px) * 0.587
    newBlue = getBlue(px) * 0.114
    luminance = newRed + newGreen + newBlue
    setColor (px, makeColor(luminance, luminance, luminance))
```

```
def sepia (picture):
  grayScaleNew(picture)
  for p in getPixels(picture):
    red = getRed(p)
    blue = getBlue(p)
    #tint shadows
    if (red < 63):
      red = red*1.1
      blue = blue*0.9
   #tint midtones
    elif (red > 62 and red < 192):
      red = red*1.15
      blue = blue*0.85
    #tint highlights
    if (red > 191):
      red = red*1.08
    if (red > 255):
       red = 255
       blue = blue*0.93
    #set the new color values
    setBlue(p, blue)
    setRed(p, red)


def mirror(picture):
 pixels = getPixels(picture)
 target = len(pixels) - 1
 for index in range(0,len(pixels)/2):
   pixel1 = pixels[index]
   color1 = getColor(pixel1)
   pixel2 = pixels[target]
   setColor(pixel2,color1)
   target = target - 1


def chromaSig (source, target, targetX, targetY):
  for x in range (0, getWidth (source)):
    for y in range (0, getHeight (source)):
      px = getPixel (source, x, y)
      color = getColor (px)
      targ = getPixel (target, x + targetX, y + targetY)
      if distance (color, black) < 200:
        setColor (targ, white)
```

```
def scale (source,canvas,factor):
  height = getHeight (source) - 3
  width = getWidth  (source)
  targetX = 0
  for sourceX in range (0, width, factor):
    targetY = 0
    for sourceY in range(0, height, factor):
      color = getColor(getPixel(source,int(sourceX),int(sourceY)))
      setColor(getPixel(canvas,targetX,targetY),color)
      sourceY = sourceY + 1
    sourceX = sourceX + 1





collage ()
```