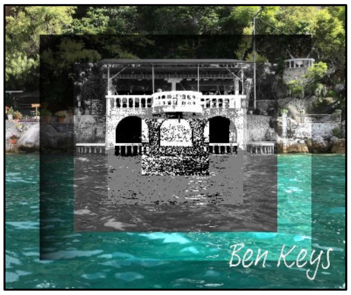
Ben Keys

Completed



Original

```
# CS120 - Project 2
# Ben Keys
# 02/25/2022

def collage():
  # define picture
  pic = makePicture(getMediaPath('balcony.jpeg'))

  # create copy of the picture
  canvas = Picture(pic)
  w = getWidth(canvas); h = getHeight(canvas)

  # defined modifying methods
  methods = [None, darkenToLighten, greyscale, posterize, edgeDetect]

  for i in range(1, len(methods)):
    # crop picture
    ratio = 1 - .2 * i
    newW = int(w * ratio); newH = int(h * ratio)
    startX = int((w - newW) / 2); startY = int((h - newH) / 2)
    croppedPic = crop(pic, startX, startY, newW, newH)

    # Modify picture
    modified = methods[i](croppedPic)

    # Overlay picture
    overlay(modified, canvas, startX, startY)

  overlaySignature(canvas)
  show(canvas)

# Return a copy of the photo cropped to the desired area.
def crop(pic, startX, startY, w, h):
  cropped = makeEmptyPicture(w, h)
  for x in range(w):
    for y in range(h):
      sourcePx = getPixel(pic, x + startX, y + startY)
      getPixel(cropped, x, y).setColorFrom(sourcePx)
  return cropped

# Overlays one photo over another.
def overlay(source, dest, startX, startY):
  for x in range(getWidth(source)):
    for y in range(getHeight(source)):
      sourcePx = getPixel(source, x, y)
      getPixel(dest, x + startX, y + startY).setColorFrom(sourcePx)


# Modifing function 1:
# Darken to top right of the picture and darken the bottom left.
def darkenToLighten(pic):
  pixels = getPixels(pic).tolist()
  min = .2
  max = 1.8
  step = (max - min) / len(pixels)
  multiplier = .2
  pixels.sort(key=pixelValue)
  for px in pixels:
    setColor(px, adjustColorTint(px, multiplier))
    multiplier += step
  return pic

def adjustColorTint(px, multiplier):
  r = getRed(px) * multiplier
  g = getGreen(px) * multiplier
  b = getBlue(px) * multiplier
  return makeColor(r,g,b)

def pixelValue(px):
  return getX(px) * getY(px)


# Modifing function 2:
def greyscale(pic):
  for px in getPixels(pic):
    value = px.getAverage()
    setColor(px, makeColor(value, value, value))
  return pic


# Modifing function 3:
def posterize(pic):
  for px in getPixels(pic):
    luminance = (getRed(px) + getGreen(px) + getBlue(px)) / 3
    if (luminance < 65):
      setColor(px, black)
    elif (6 <= luminance <= 165):
      setColor(px, gray)
    else:
      setColor(px, white)
  return pic


# Modifing function 4:
def edgeDetect(pic):
  threshold = 10
  copy = Picture(pic)
  w = getWidth(pic); h = getHeight(pic)
  for x in range(w):
    for y in range(h):
      px = getPixel(copy, x, y)
      xOffset = 1 if x == 0 else -1
      yOffset = 1 if y == 0 else -1
      thisLum = px.getAverage()
      testLum = getPixel(pic, x + xOffset, y + yOffset).getAverage()
      if abs(testLum - thisLum) > threshold:
        setColor(px, black)
      if abs(testLum - thisLum) <= threshold:
        setColor(px, white)
  return copy


# Overlay signature
def overlaySignature(pic):
  signature = makePicture(getMediaPath('signature.jpg'))
  w = getWidth(signature)
  h = getHeight(signature)
  startX = getWidth(pic) - w - 10 # offset
  startY = getHeight(pic) - h - 15 # offset
  for x in range(w):
    for y in range(h):
      px = getPixel(signature, x, y)
      colorAverage = px.getAverage()
      if (colorAverage < 110):
        destPx = getPixel(pic, x + startX, y + startY)
        colorAverage = 255 - colorAverage
        setColor(destPx, makeColor(colorAverage, colorAverage, colorAverage))
```