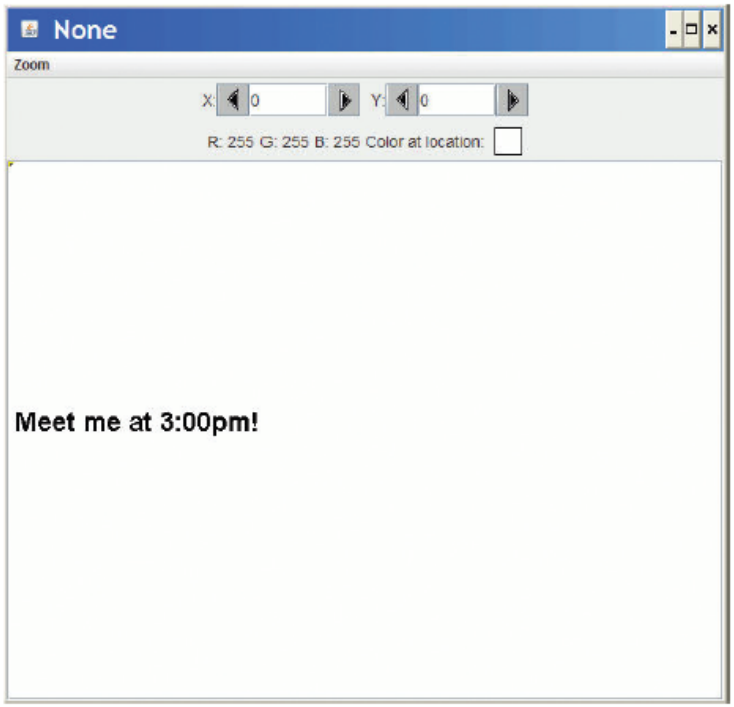
Hiding Text in a Picture

- **Steganography** is hiding information in ways that can't be easily detected.
- One form of steganography is hiding text information of a picture.

Our Algorithm for Hiding Text

- We'll draw our message in black pixels on a message picture.
- We'll hide our message in a picture of the same size.
- First: Make sure that all red values are **even**.
- Second: For every pixel where the message picture is black, add one to the red value at the corresponding x,y.

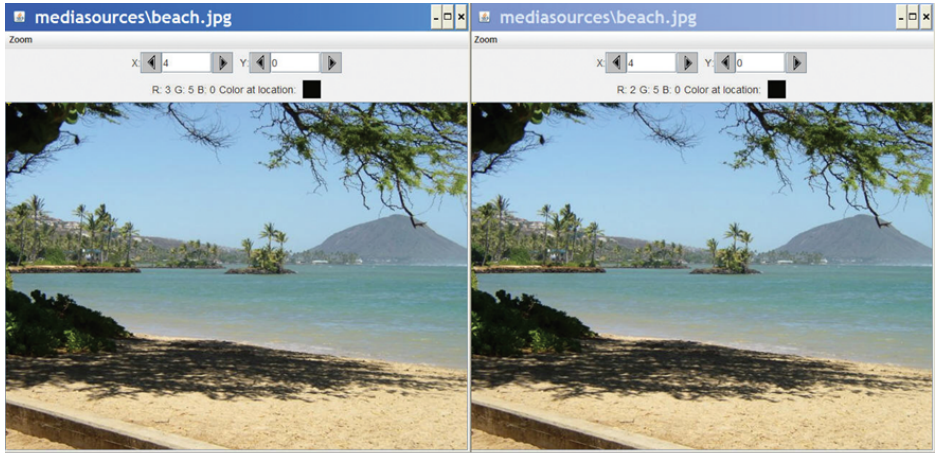


Function to Encode the Message

```
def encode(msgPic ,original ):
    # Assume msgPic and original have same dimensions
    # First , make all red pixels even
    for pxl in getPixels(original ):
            # Using modulo operator to test oddness
            if (getRed(pxl) % 2) == 1:
                    setRed(pxl , getRed(pxl) - 1)
    # Second , wherever there ' s black in msgPic
    # make odd the red in the corresponding original pixel
    for x in range(0, getWidth(original )):
            for y in range(0, getHeight(original )):
                    msgPxl = getPixel(msgPic ,x,y)
                    origPxl = getPixel(original ,x,y)
                    if (distance(getColor(msgPxl),black) < 100.0):
                            # It' s a message pixel! Make the red value odd.
                            setRed(origPxl , getRed(origPxl )+1)
```

Doing the Encoding

```
>>> beach = makePicture(getMediaPath("beach.jpg"))
>>> explore(beach)
>>> msg = makePicture(getMediaPath("msg.jpg"))
>>> encode(msg,beach)
>>> explore(beach)
>>> writePictureTo(beach,getMediaPath("beachHidden.png"))
```



Original Encoded

It's really important to save the message as .PNG or .BMP, **not** JPEG. JPEG is **lossy** so pixel color values might change. PNG and BMP are lossless formats.

Decoding: Getting the Message Back

- Create a new “message” picture of same size as the encoded image.
- For each pixel, if the red value is **odd**, make the pixel in the message at the same x,y **black.**

```
def decode(encodedImg):
  # Takes in an encoded image. Return the original message
  message = makeEmptyPicture(getWidth(encodedImg),getHeight(encodedImg))
  for x in range(0,getWidth(encodedImg)):
    for y in range(0,getHeight(encodedImg)):
      encPxl = getPixel(encodedImg,x,y)
      msgPxl = getPixel(message,x,y)
      if (getRed(encPxl) % 2) == 1:
        setColor(msgPxl,black)
  return message
```