Working with image files and resizing

Install Pillow

```
C:\Users\Som\AppData\Local\Programs\Python\Python38\Scripts>python -m pip instal
l pillow --user
Collecting pillow
  Downloading Pillow-8.2.0-cp38-cp38-win_amd64.whl (2.2 MB)
     |████████████████████████████████| 2.2 MB 211 kB/s
Installing collected packages: pillow
Successfully installed pillow-8.2.0

C:\Users\Som\AppData\Local\Programs\Python\Python38\Scripts>
```

```
>>> from PIL import Image
>>> import os
>>> os.getcwd()
'c:\\py01'
# you can place the image file in the working directory ..
>>> img=Image.open(r"SPC00.png")
>>> img.show()
```

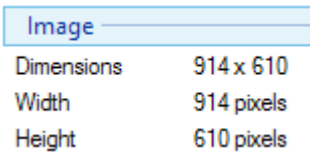
Want to rotate the image ?

```
>>> a=img.rotate(90)
>>> a.show()
```

What is the size of the image in pixels ?

```
>>> img.size
(914, 610)
```

If you directly check the image property by right clicking it's the same



Want to check the image format ?

```
>>> img.mode
'RGBA'
```

So we can see that the image is a 32 bit image with 4 channels Red, Green, Blue and Alpha transparency masking each 8 bits

How to resize an image and display ?

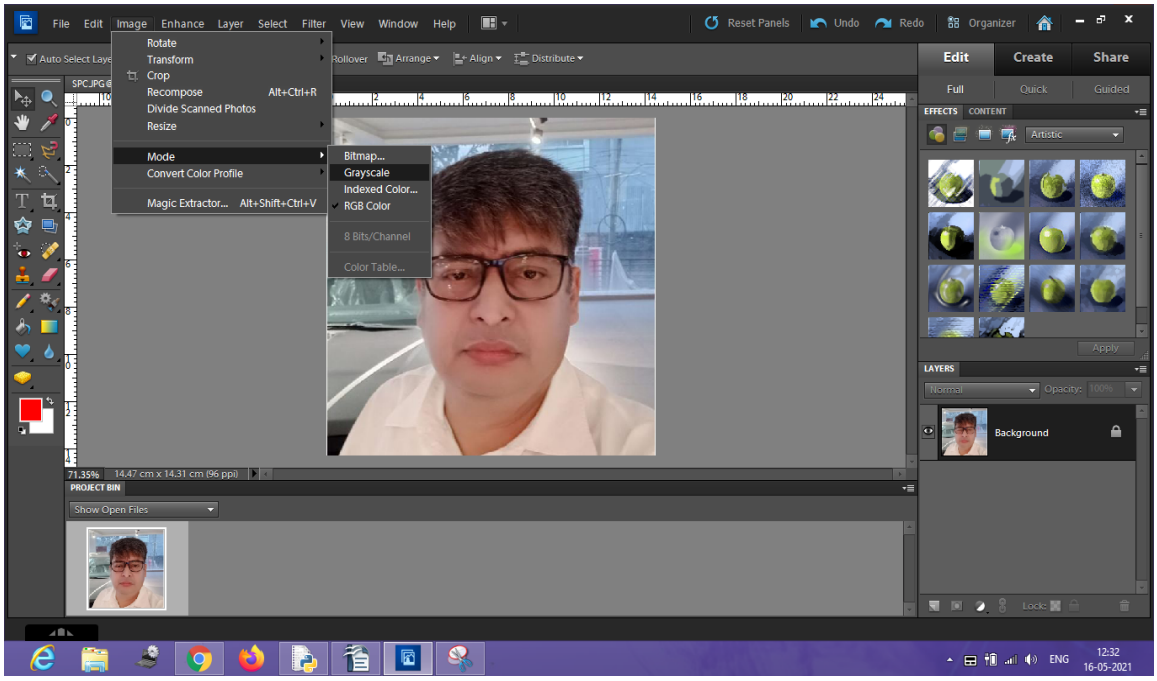
size=(182, 122)

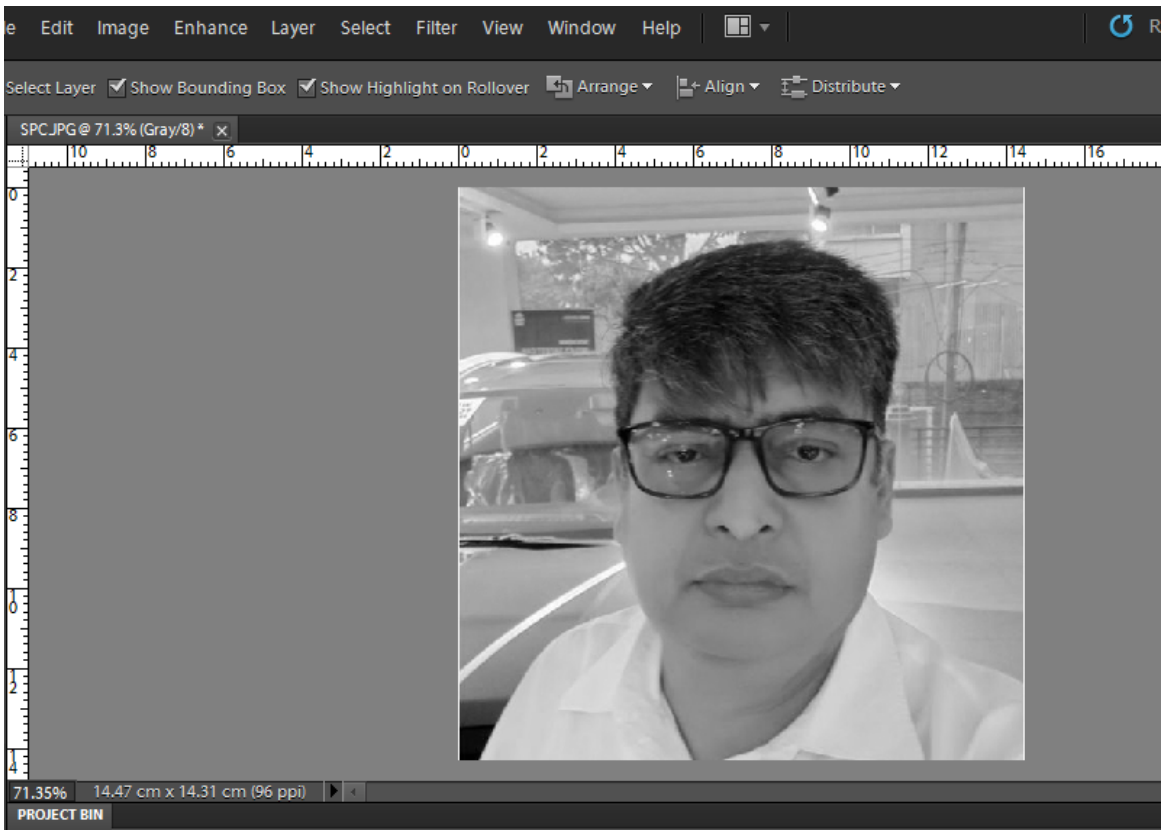
```
>>> from PIL import Image
>>> img=Image.open(r'SPC00.png')
>>> size = (182, 122)
>>> newimg=img.resize(size)
>>> newimg.show()
```

How about saving a resized image ?

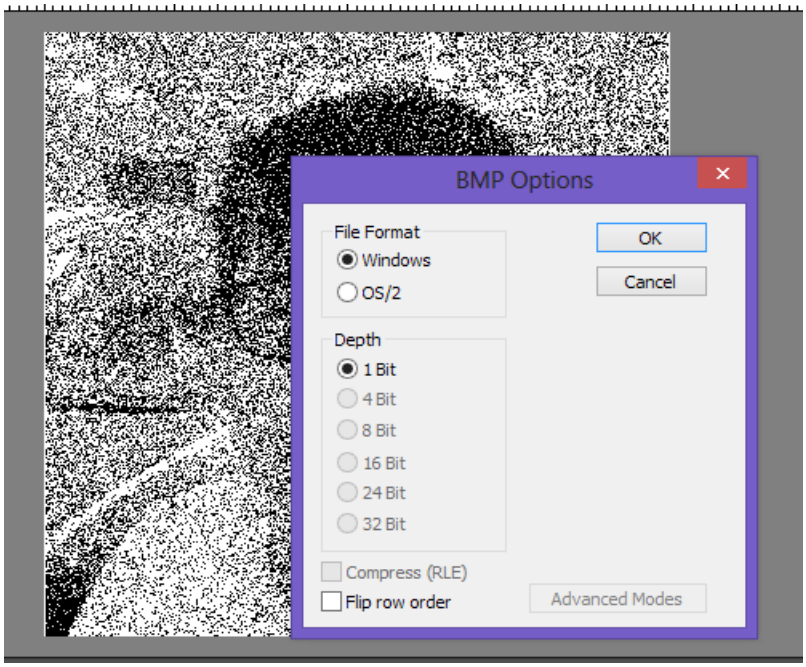
```
>>> newimg.save(r'resized00.png')
>>> newimg=img.resize(size, resample=Image.BILINEAR)
>>> newimg.save(r'resized00.png')
>>> nimg=Image.open(r'resized00.png')
>>> nimg.size
(182, 122)
>>> nimg.show()
```

let's try to open a Grey Scale 8 bits and 1 bit BMP image after necessary modifications in Photoshop

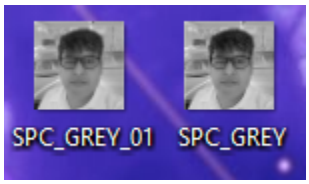




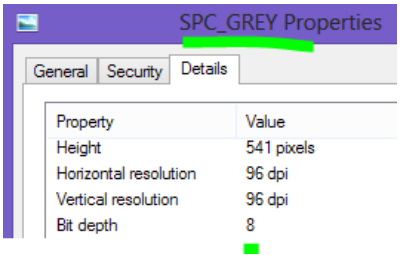
And an one bit BMP



The files are named as SPC_GREY.JPG (8 bits) and SPC_GREY_01.BMP (1 bit)




```
>>> img=Image.open(r'SPC_GREY.jpg')
>>> img.mode
'L'
```



L indicates an 8 bit pixels, GreyScale image

```
>>> img=Image.open(r'SPC_GREY_01.bmp')
>>> img.show()
>>> img.mode
'P'
```