importing library files for image processing

```
In [1]: import skimage               # to install this package the command is:      conda install -c anaconda scikit-image
        import matplotlib.pyplot as plt
```

```
In [2]: %matplotlib inline
```

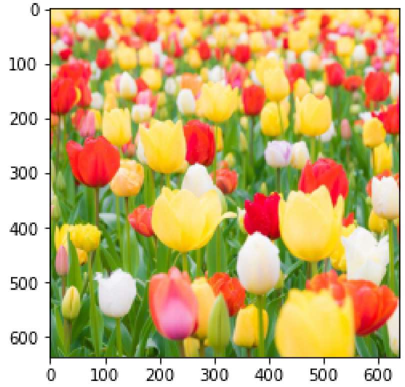
```
In [3]: from skimage import io                          # to read input and output
```

```
In [4]: from skimage import data                        # set of images in data " https://scikit-image.org/docs/dev/auto_examples/index.html"
```

```
In [5]: img = io.imread('flowers.jpg')     # Load a image from system
```

```
In [6]: plt.imshow(img)                  #display the image with x and y axis
```

```
Out[6]: <matplotlib.image.AxesImage at 0x2b6fbacb850>
```



```
In [7]: img                                # to see the array of the image
```

```
Out[7]: array([[[255, 226, 156],
                [255, 226, 156],
                [255, 227, 153],
                ...,
                [250, 186,  86],
                [247, 182,  80],
                [243, 178,  74]],

               [[255, 226, 156],
                [255, 227, 154],
                [255, 227, 153],
                ...,
                [255, 197,  95],
                [255, 192,  89],
                [251, 188,  85]],

               [[254, 225, 155],
                [254, 226, 153],
                [255, 225, 152],
                ...,
                [255, 207, 106],
                [255, 202, 102],
                [255, 199,  97]],

               ...,

               [[ 46, 110,  23],
                [ 46, 111,  21],
                [ 47, 113,  16],
                ...,
                [120, 162,  96],
                [ 98, 146,  86],
                [ 87, 138,  81]],

               [[ 45, 111,  23],
                [ 46, 111,  21],
                [ 47, 113,  16],
                ...,
                [115, 161,  96],
                [ 97, 147,  88],
                [ 88, 141,  87]],

               [[ 44, 110,  22],
                [ 47, 112,  22],
                [ 48, 114,  17],
                ...,
                [113, 164,  98],
                [ 98, 149,  93],
                [ 90, 143,  91]]], dtype=uint8)
```

In [8]:
```
type(img)          # to see the type
```

Out[8]:
```
numpy.ndarray
```

In [9]:
```
img.shape    # to check how many rows and columns in image (rows, Columns, Channels)
```

Out[9]:
```
(638, 640, 3)
```

In [10]:
```
camera = data.camera()     # importing a image from data
```

In [11]:
```
camera                     # display the array matrix
```

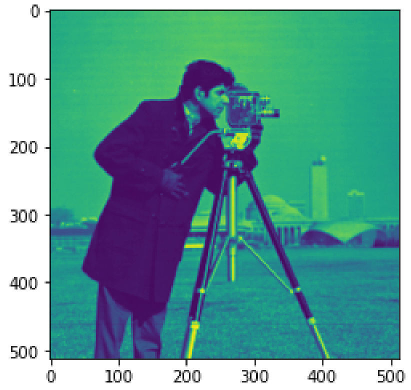
Out[11]:
```
array([[156, 157, 160, ..., 152, 152, 152],
       [156, 157, 159, ..., 152, 152, 152],
       [158, 157, 156, ..., 152, 152, 152],
       ...,
       [121, 123, 126, ..., 121, 113, 111],
       [121, 123, 126, ..., 121, 113, 111],
       [121, 123, 126, ..., 121, 113, 111]], dtype=uint8)
```

In [12]:
```
type(camera)         # to see the type
```

Out[12]:
```
numpy.ndarray
```

In [13]:
```
plt.imshow(camera)
camera.shape             # to check how many rows and columns in image (rows, Columns, Channels)
```

Out[13]:
```
(512, 512)
```

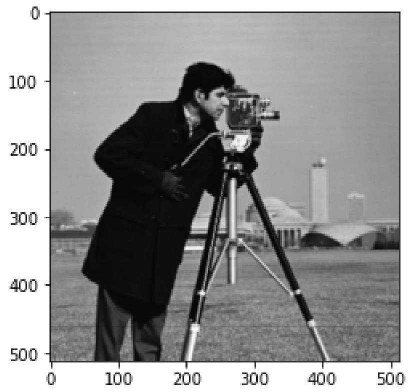


In [14]:
```
camera.size            #size of the image
```

Out[14]:
```
262144
```

```
In [15]: plt.imshow(camera,'gray')            # display the image in gray
```

```
Out[15]: <matplotlib.image.AxesImage at 0x2b6fbd2e1f0>
```



```
In [16]: camera.max()                        # maximam intensity or pixel value
```

```
Out[16]: 255
```

```
In [17]: camera.min()                        # minimum intensity or pixel value
```

```
Out[17]: 0
```

```
In [18]: img.max()
```

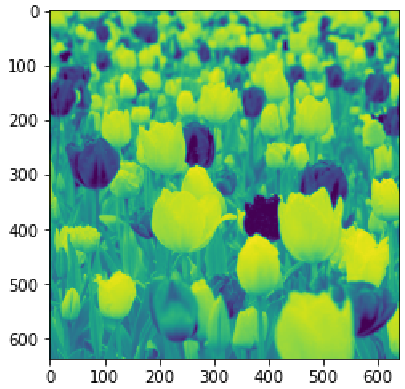
```
Out[18]: 255
```

```
In [19]: img.min()
```

```
Out[19]: 0
```

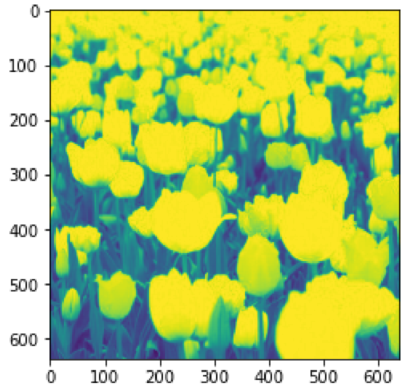
In [20]:
```
plt.imshow(img[:,:,1])         # Channel ( 0 - Red , 1- Green, 2 - Blue)
```

Out[20]:
```
<matplotlib.image.AxesImage at 0x2b6fbecda60>
```

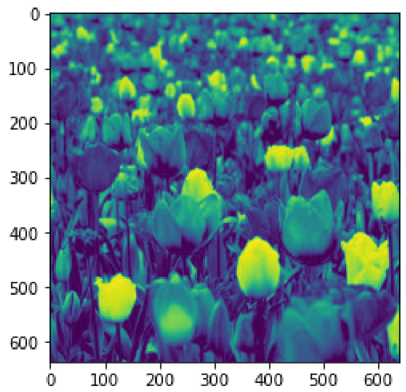


In [21]:
```
plt.imshow(img[:,:,0])          # Channel ( 0 - Red , 1- Green, 2 - Blue)
```

Out[21]:
```
<matplotlib.image.AxesImage at 0x2b6fbf22b20>
```



```
In [22]: plt.imshow(img[:,:,2])          # Channel ( 0 - Red , 1- Green, 2 - Blue)

Out[22]: <matplotlib.image.AxesImage at 0x2b6fbf79b50>
```

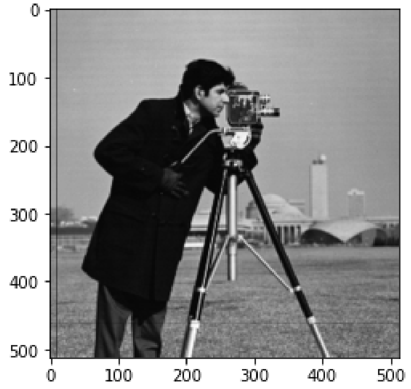


```
In [23]: img[20,30]                   # checking the particular pixel value, in this img is a coloured image

Out[23]: array([240, 195,  80], dtype=uint8)
```

```
In [24]: camera = data.camera()          # loading a  new image named camera and intensity value of particular pixel is displayed
         camera[10,20]

Out[24]: 153
```

```
In [25]: camera[:,10] = 0                  # Assign value for all rows and 10 th column
```

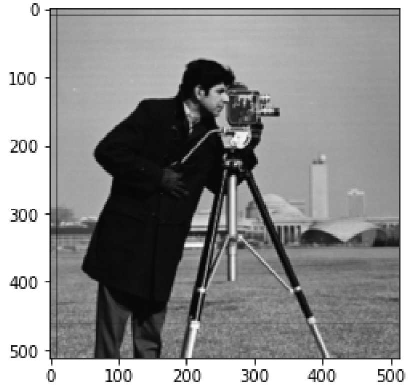
```
In [26]: plt.imshow(camera, 'gray')         # display the image

Out[26]: <matplotlib.image.AxesImage at 0x2b6fcfa1a90>
```



```
In [27]: camera[10,:] = 0                        # Assign value for 10th row and all columns
```

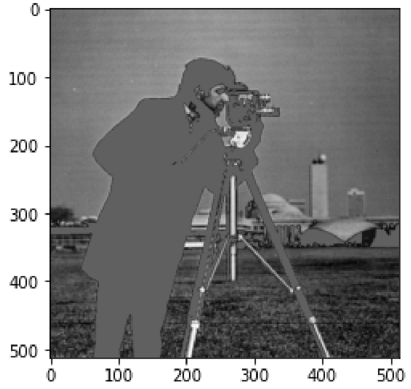
```
In [28]: plt.imshow(camera,'gray')
```

```
Out[28]: <matplotlib.image.AxesImage at 0x2b6fcffb0a0>
```

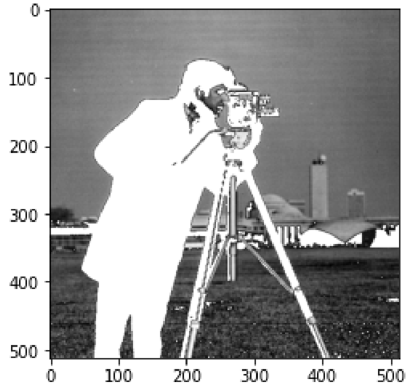


```
In [29]: #Threshold
         camera = data.camera()
         mask = camera<87                # in image the pixel values which ever lesser than 87 will be assigned 150
         camera[mask] = 150
         plt.imshow(camera,'gray')
```

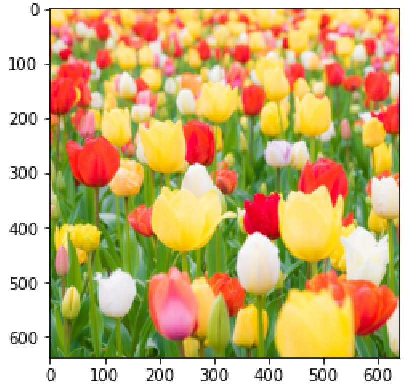
```
Out[29]: <matplotlib.image.AxesImage at 0x2b6fcf67370>
```



```
In [30]: #Threshold now i changed the assigned value to 255
         camera = data.camera()
         mask = camera<87                 # in image the pixel values which ever lesser than 87 will be assigned 150
         camera[mask] = 255
         plt.imshow(camera,'gray')

Out[30]: <matplotlib.image.AxesImage at 0x2b6fb9f61c0>
```



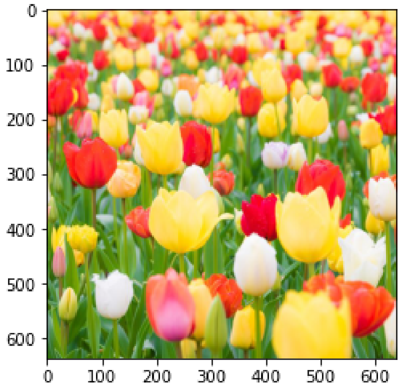
```
In [32]: img1 = io.imread('flowers.jpg')

In [33]: plt.imshow(img1)

Out[33]: <matplotlib.image.AxesImage at 0x2b6fd1392e0>
```



```
In [35]: img1.shape

Out[35]: (638, 640, 3)
```

```
In [36]: img1[50,60]

Out[36]: array([204, 188, 126], dtype=uint8)
```

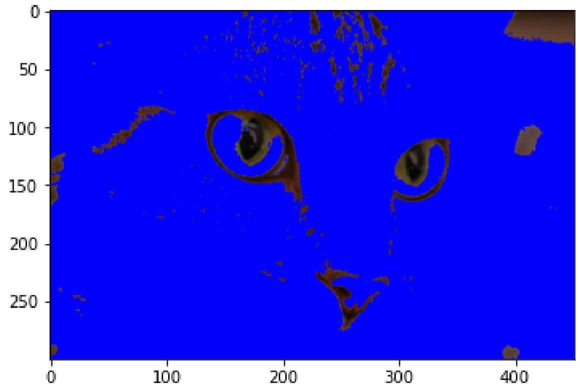
```
In [37]: img1[50,60] = [0,255,0]      # it is a coloured image so [R,G,B] assigned to a particular pixel
         plt.imshow(img1)

Out[37]: <matplotlib.image.AxesImage at 0x2b6fd2bc580>
```



```
In [38]: cat = data.chelsea()
```

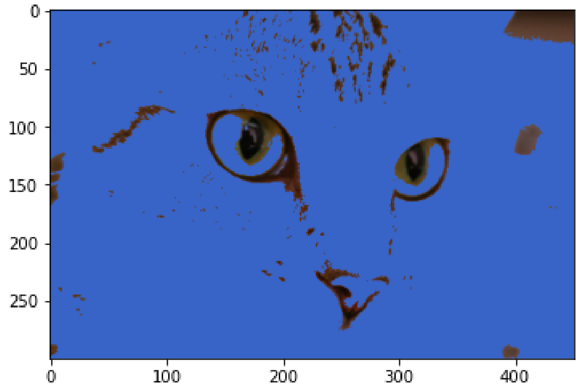
In [40]:
```
red_cat = cat.copy()
mask = cat[:,:,0]>100
red_cat[mask] = [0,0,255]            # change the values so that you can get R, G & B  you will get image [R,G,B]
plt.imshow(red_cat)
```

Out[40]:
```
<matplotlib.image.AxesImage at 0x2b6fd312a00>
```

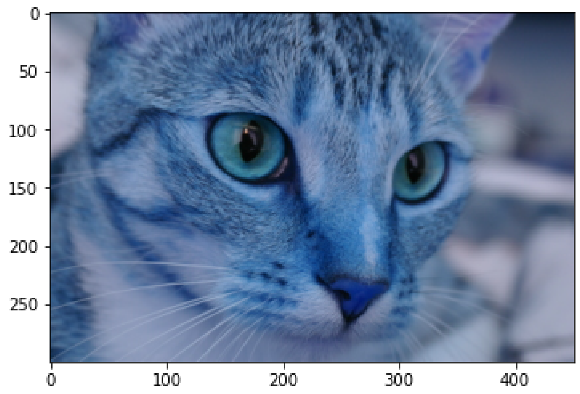


In [41]:
```
red_cat = cat.copy()
mask = cat[:,:,0]>100
red_cat[mask] = [56,100,200]            # change the values so that you can get R, G & B  you will get image [R,G,B]
plt.imshow(red_cat)
```

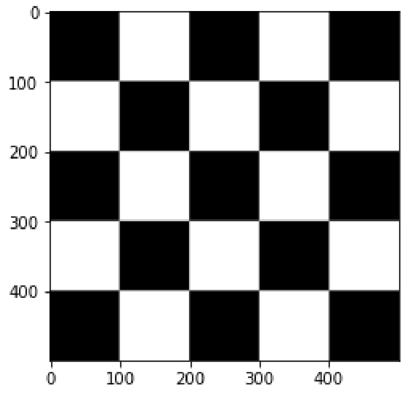
Out[41]:
```
<matplotlib.image.AxesImage at 0x2b6fd361ac0>
```



```
In [42]: BGR_cat = cat[:,:,::-1]              # to assing R value - B; G value - G and B value - R
         plt.imshow(BGR_cat)

Out[42]: <matplotlib.image.AxesImage at 0x2b6fd3b77f0>
```



```
In [43]: img4 = io.imread('C:/Users/hemac/Documents/my computer vision projects/cb.png')  # import the image even from anyother location

In [44]: plt.imshow(img4)

Out[44]: <matplotlib.image.AxesImage at 0x2b6fd6b8580>
```


0
100
200
300
400
0
100
200
300
400