In [1]:

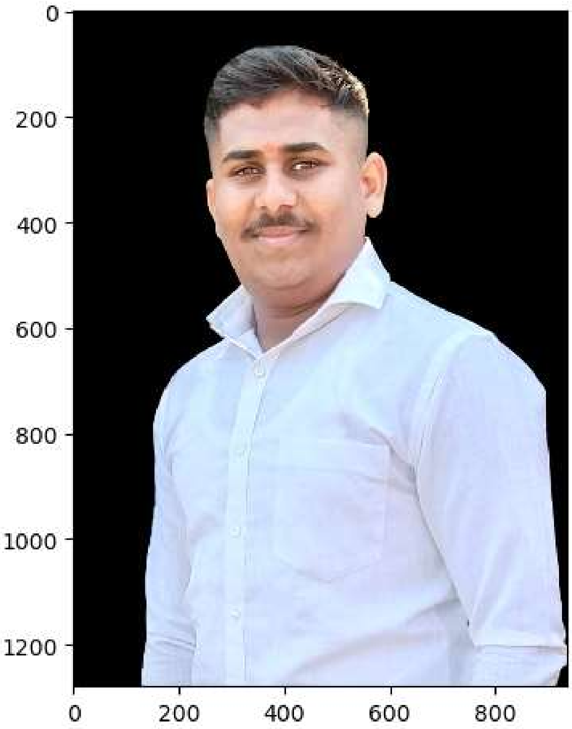
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
from skimage.io import imread, imshow
```

In [2]:

```
image = imread('C:/Users/samir/OneDrive/Desktop/Sameer1.png')
imshow(image)
```

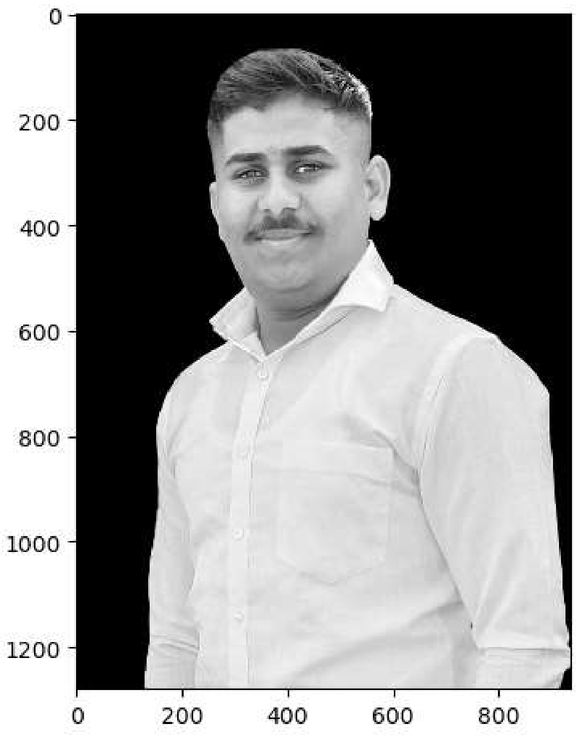
Out[2]:

```
<matplotlib.image.AxesImage at 0x27ffb0c0a90>
```



In [3]:
```
image1 = imread('C:/Users/samir/OneDrive/Desktop/Sameer1.png', as_gray=True)
imshow(image1)
```

Out[3]:
```
<matplotlib.image.AxesImage at 0x27ffb1d4490>
```



In [4]:
```
print(image.shape)
print(image1.shape)
```

```
(1280, 937, 3)
(1280, 937)
```

In [5]:
```
print(image.size)
print(image1.size)
```

```
3598080
1199360
```

In [6]:
```
pixel_feat1 = np.reshape(image1, (1280 * 937))
pixel_feat1
```

Out[6]:
```
array([0., 0., 0., ..., 0., 0., 0.])
```

In [7]:
```
pixel_feat2 = np.reshape(image, (1280 * 937 * 3))
pixel_feat2
```

Out[7]:
```
array([0, 0, 0, ..., 0, 0, 0], dtype=uint8)
```

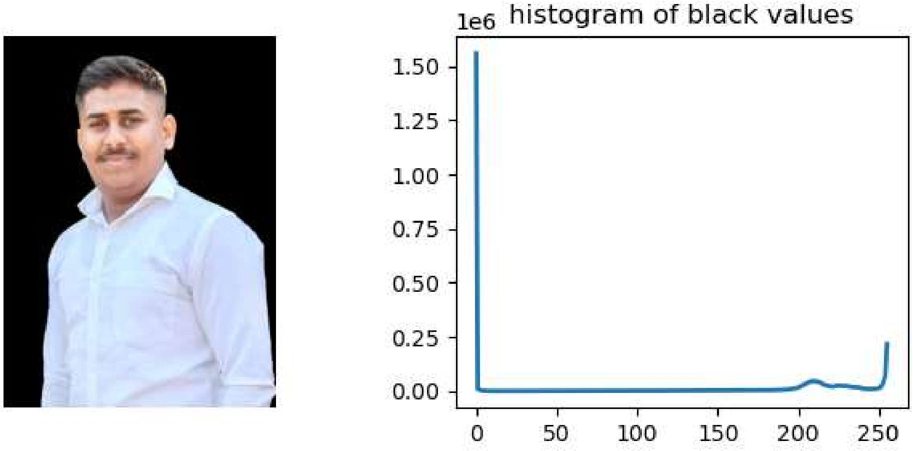
In [8]:
```
from skimage import filters
```

In [10]:
```
from skimage.exposure import histogram
hist, hist_centers = histogram(image)

#Plotting the Image and the Histogram of gray values
fig, axes = plt.subplots(1, 2, figsize=(8, 3))
axes[0].imshow(image, cmap=plt.cm.gray)
axes[0].axis('off')
axes[1].plot(hist_centers, hist, lw=2)
axes[1].set_title('histogram of black values')
```

```
C:\Users\samir\anaconda3\Lib\site-packages\skimage\_shared\utils.py:326: UserWarning: This might be a
color image. The histogram will be computed on the flattened image. You can instead apply this functi
on to each color channel, or set channel_axis.
  return func(*args, **kwargs)
```

Out[10]:
```
Text(0.5, 1.0, 'histogram of black values')
```



In [ ]: