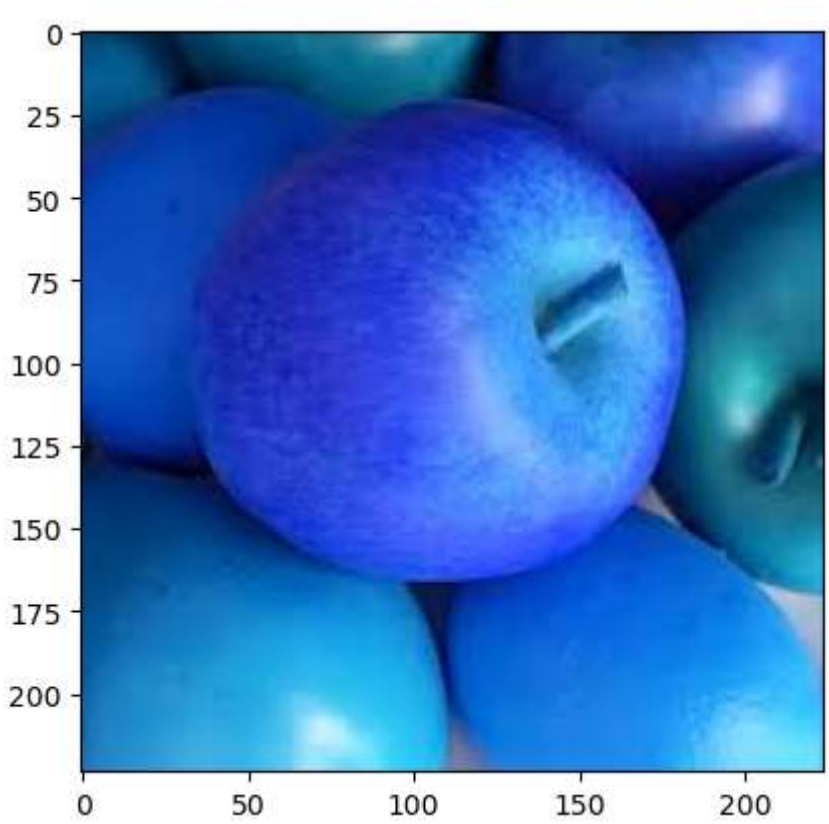
In [1]:
```
!pip install opencv-python
```

```
Requirement already satisfied: opencv-python in c:\users\admin\anaconda3\lib\site-pac
kages (4.7.0.72)
Requirement already satisfied: numpy>=1.17.0 in c:\users\admin\anaconda3\lib\site-pac
kages (from opencv-python) (1.23.5)
```

In [2]:
```
import cv2
import matplotlib.pyplot as plt
import numpy as np
%matplotlib inline
```

In [3]:
```
img = cv2.imread('apple.jpg', cv2.IMREAD_UNCHANGED)
plt.imshow(img)
```

Out[3]:
```
<matplotlib.image.AxesImage at 0x1e92c5d58a0>
```



In [4]:
```
print(img.shape)
```

```
(224, 224, 3)
```

In [5]:
```
img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
plt.imshow(img)
```

Out[5]:
```
<matplotlib.image.AxesImage at 0x1e92c6fb7c0>
```

In [6]:
```
pixel_values = img.reshape(-1,3)
pixel_values = np.float32(pixel_values)
```

In [7]:
```
print(pixel_values.shape)
```

```
(50176, 3)
```

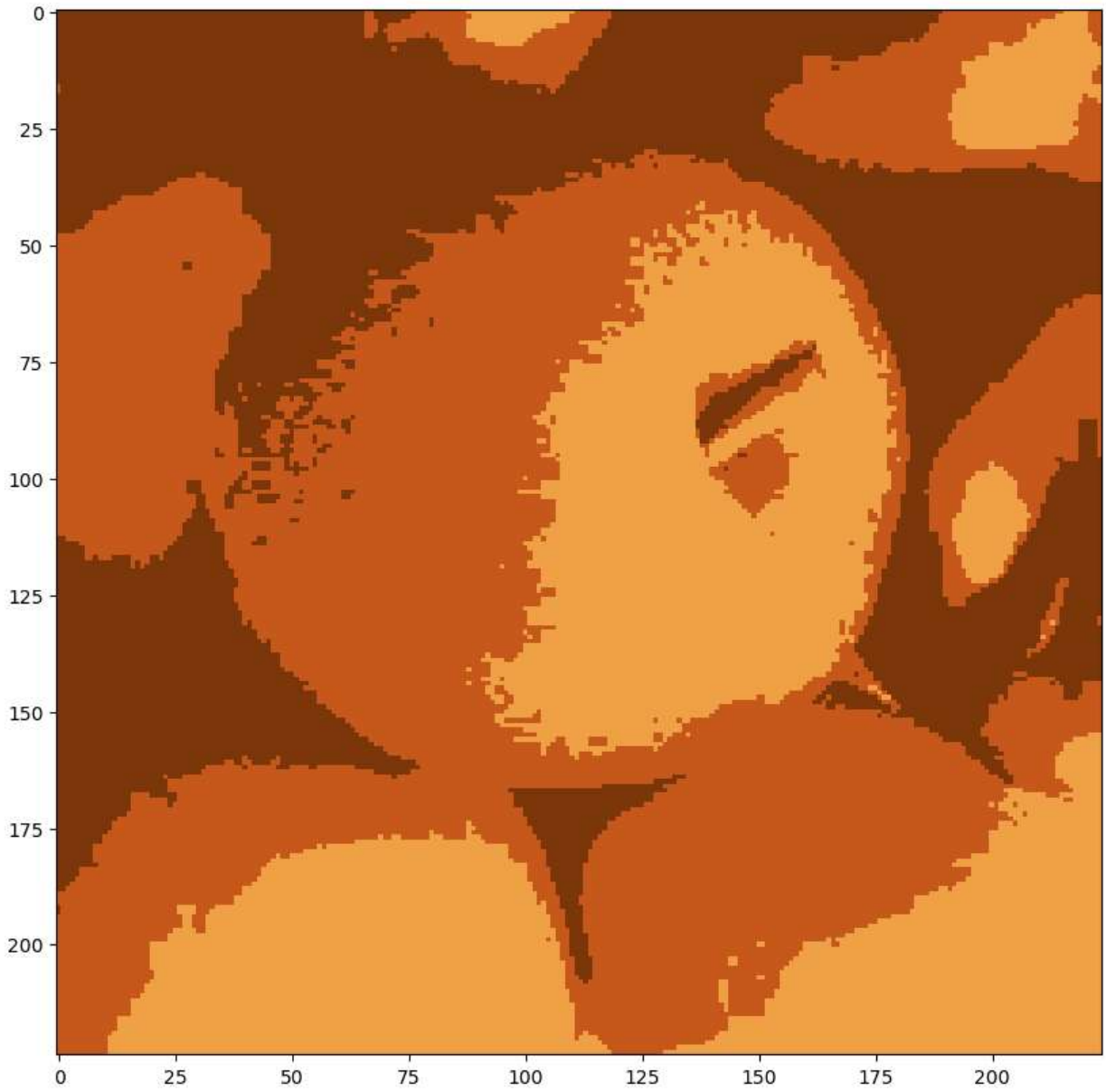
In [8]:
```
criteria = (cv2.TERM_CRITERIA_EPS + cv2.TERM_CRITERIA_MAX_ITER, 100, 0.2)
```

In [11]:
```
k = 3;
_, labels, (centers) = cv2.kmeans(pixel_values, k, None, criteria, 10, cv2.KMEANS_RAND
```

In [12]:
```
centers = np.uint8(centers)
labels = labels.flatten()
```

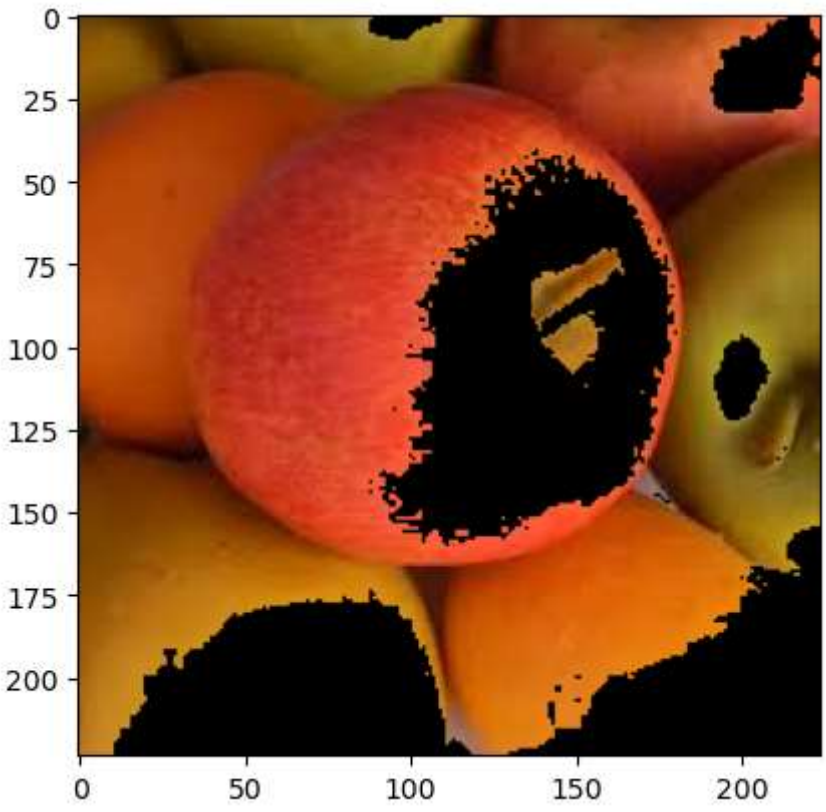
In [15]:
```
segmented_image = centers[labels.flatten()]
```

In [16]:
```
segmented_image = segmented_image.reshape(img.shape)
plt.figure(figsize=(12,10))
plt.imshow(segmented_image)
plt.show()
```

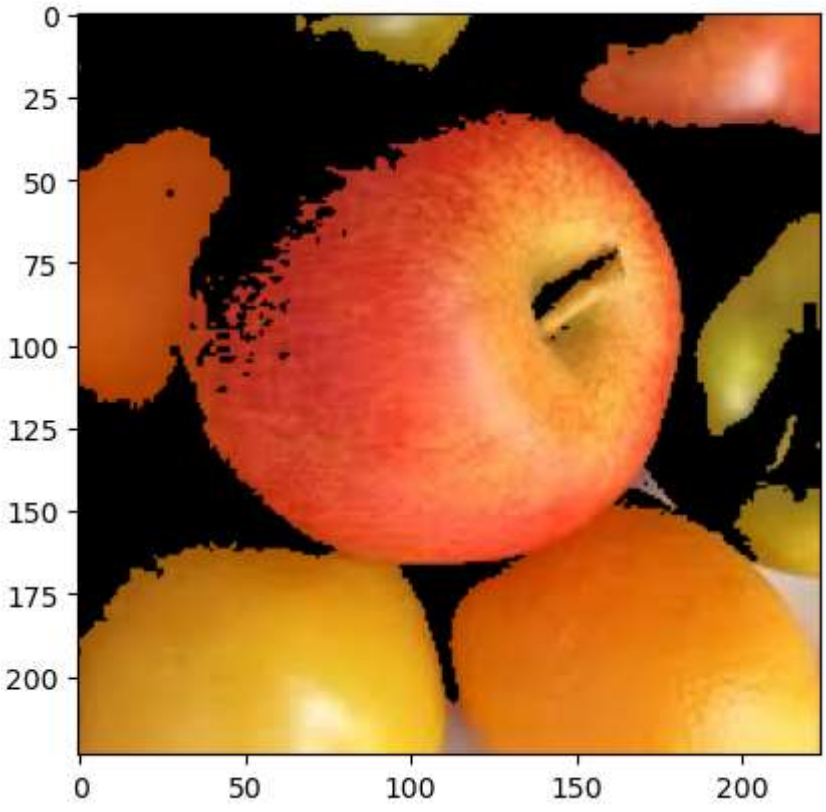


In [18]:
```
masked_image0 = np.copy(img)
masked_image1 = np.copy(img)
masked_image2 = np.copy(img)

masked_image0 = masked_image0.reshape((-1, 3))
masked_image1 = masked_image1.reshape((-1, 3))
masked_image2 = masked_image2.reshape((-1, 3))

cluster = 0
masked_image0[labels == cluster] = [0, 0, 0]

cluster = 1
masked_image1[labels == cluster] = [0,0,0]

cluster = 2
masked_image2[labels == cluster] = [0,0,0]

masked_image0 = masked_image0.reshape(img.shape)
masked_image1 = masked_image1.reshape(img.shape)
masked_image2 = masked_image2.reshape(img.shape)
```

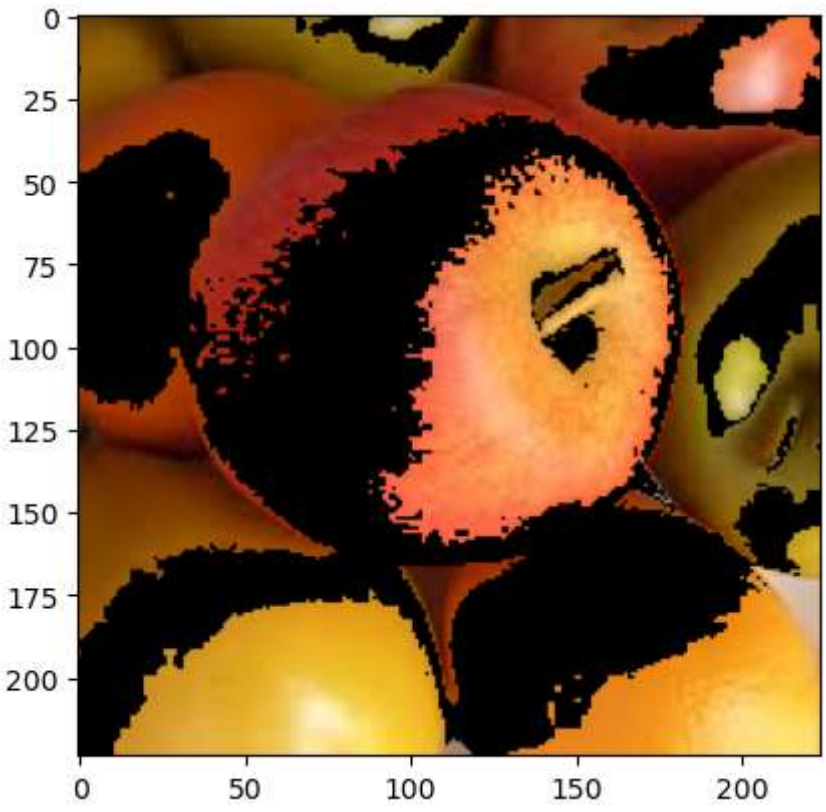
In [19]:
```
plt.imshow(masked_image0)
plt.show()
```



In [20]:
```
plt.imshow(masked_image1)
plt.show()
```



In [21]:
```
plt.imshow(masked_image2)
plt.show()
```



In [ ]: