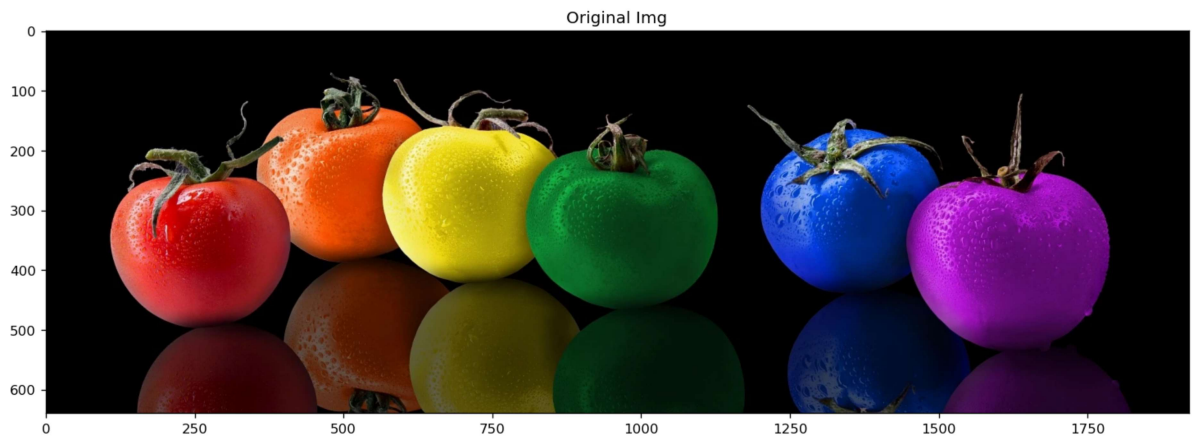


## Assgn 03

```
In [1]: import numpy as np
import cv2
import matplotlib.pyplot as plt
from matplotlib.pyplot import figure
figure(figsize=(15, 15), dpi=120)

image = cv2.imread("./images/practice_img/hsv1.webp", cv2.IMREAD_COLOR)

plt.imshow(cv2.cvtColor(image, cv2.COLOR_BGR2RGB)),plt.title('Original Img')
plt.show()
```

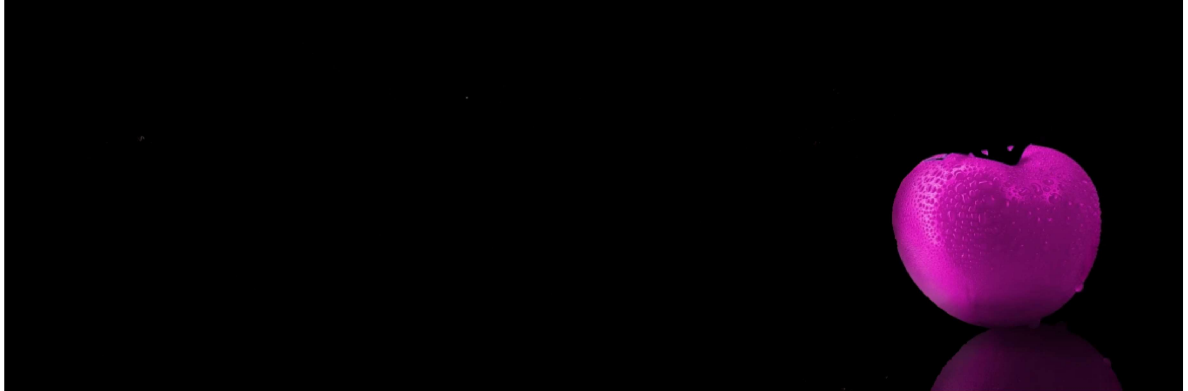


### 1. Find the HSV values of the below 'Assgn\_03 Questioning Img' image ;

- (hMin = , sMin = , vMin = ), (hMax = , sMax = , vMax = )
- Then, generate the same as the answer image again with the found value to show that the result is correct

```
In [23]: import matplotlib.pyplot as plt
from matplotlib.pyplot import figure
figure(figsize=(15, 15), dpi=100)
image = cv2.imread("./results/assgn3.png", cv2.IMREAD_COLOR)
plt.imshow(image),plt.axis('off'),plt.title('Assgn_03 Questioning Img')
plt.show()
```

Assgn\_03 Questioning Img



HMin: 125, SMin: 102, VMin: 0, HMax: 168, SMax: 225, VMax: 255

```
In [43]: import numpy as np
import cv2
import matplotlib.pyplot as plt
from matplotlib.pyplot import figure

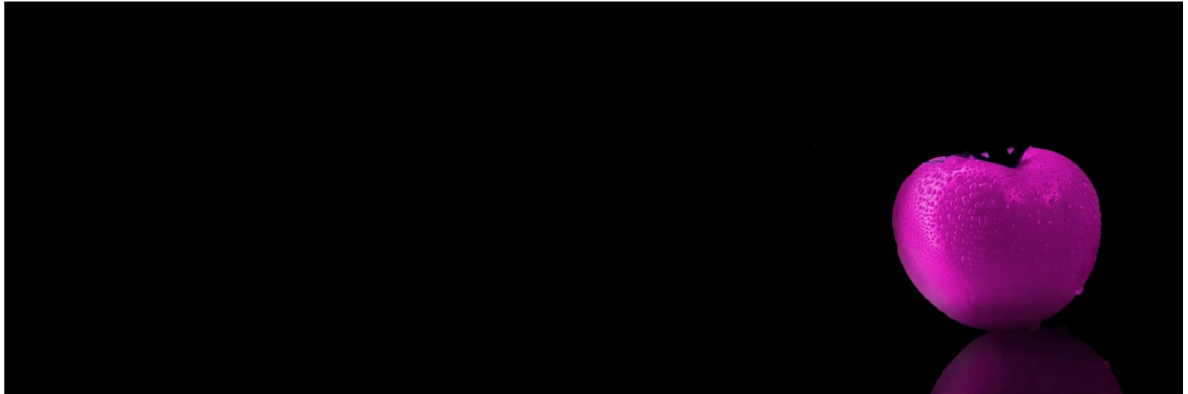
image = cv2.imread("./images/practice_img/hsv1.webp", cv2.IMREAD_COLOR)

hsv = cv2.cvtColor(image, cv2.COLOR_BGR2HSV)
lower = np.array([140, 118, 0])
upper = np.array([167, 255, 255])
mask = cv2.inRange(hsv, lower, upper)
result = cv2.bitwise_and(image, image, mask=mask)

figure(figsize=(15, 15), dpi=100)

plt.imshow(cv2.cvtColor(result, cv2.COLOR_BGR2BGRA))
plt.title('hMin = 140 , sMin = 118, vMin = 0, hMax = 167 , sMax = 255, vMax = 255')
plt.axis('off')
plt.show()
```

hMin = 140 , sMin = 118, vMin = 0, hMax = 167 , sMax = 255, vMax = 255



## 2. Blending Images (result = img1 + img2)

```
In [24]: import cv2
import numpy as np
from matplotlib import pyplot as plt
from matplotlib.pyplot import figure

img1 = cv2.imread('./images/practice_img/ggl_logo.png', cv2.IMREAD_UNCHANGED) #
img2 = cv2.imread('./images/practice_img/mountain.jpg')
img3 = cv2.imread('./results/assgn03.jpg')

figure(figsize=(15, 10), dpi=100)

plt.subplot(131),plt.imshow(cv2.cvtColor(img1, cv2.COLOR_BGR2RGB)),plt.title('img1')
plt.subplot(132),plt.imshow(cv2.cvtColor(img2, cv2.COLOR_BGR2RGB)),plt.title('img2')
plt.subplot(133),plt.imshow(cv2.cvtColor(img3, cv2.COLOR_BGR2RGB)),plt.title('Result(img1+img2)')
plt.show()
```



**Write and execute code to synthesize the above img1 and img2 to generate Result(img1+img2) img and plot the result.**

```
In [98]: import cv2
import numpy as np
from matplotlib import pyplot as plt
from matplotlib.pyplot import figure

img1 = cv2.imread('./images/practice_img/ggl_logo.png', cv2.IMREAD_UNCHANGED) #
img2 = cv2.imread('./images/practice_img/mountain.jpg')

mask = np.full_like(img1, 255)

height, width = img2.shape[:2]
center = (width//2, height//2) # img center

normal = cv2.seamlessClone(img1, img2, mask, center, cv2.NORMAL_CLONE) # seamLe
mixed = cv2.seamlessClone(img1, img2, mask, center, cv2.MIXED_CLONE )

krrnl_blr = np.ones((3,3), np.float32) / 9
mixed_blr2 = cv2.filter2D(mixed, -1, krrnl_blr )

img3 = cv2.addWeighted(img2, 0.8, mixed_blr2, 0.2, 0)
plt.show()

figure(figsize=(15, 10), dpi=100)

plt.subplot(131),plt.imshow(cv2.cvtColor(img1, cv2.COLOR_BGR2RGB))
plt.title('img1'),plt.axis('off')
plt.subplot(132),plt.imshow(cv2.cvtColor(img2, cv2.COLOR_BGR2RGB))
plt.title('img2'),plt.axis('off')
plt.subplot(133),plt.imshow(cv2.cvtColor(img3, cv2.COLOR_BGR2RGB))
plt.title('Result(img1+img2)'),plt.axis('off')
plt.show()
```



In [ ]: