In [6]:
```
import os
import cv2
import numpy as np
import matplotlib.pyplot as plt
from IPython.display import Image
%matplotlib inline
```

In [3]:
```
pip install opencv-python
```

```
Collecting opencv-python
  Downloading opencv_python-4.12.0.88-cp37-abi3-win_amd64.whl.metadata (19 kB)
Collecting numpy<2.3.0,>=2 (from opencv-python)
  Downloading numpy-2.2.6-cp313-cp313-win_amd64.whl.metadata (60 kB)
Downloading opencv_python-4.12.0.88-cp37-abi3-win_amd64.whl (39.0 MB)
   ---------------------------------------- 0.0/39.0 MB ? eta -:--:--
   ----- ---------------------------------- 5.2/39.0 MB 34.1 MB/s eta 0:00:01
   ----------- ---------------------------- 10.7/39.0 MB 26.1 MB/s eta 0:00:02
   ---------------- ----------------------- 16.3/39.0 MB 26.6 MB/s eta 0:00:01
   -------------------------- ------------- 26.2/39.0 MB 32.1 MB/s eta 0:00:01
   --------------------------------- ------ 32.2/39.0 MB 31.4 MB/s eta 0:00:01
   ---------------------------------------  38.8/39.0 MB 31.5 MB/s eta 0:00:01
   ---------------------------------------- 39.0/39.0 MB 27.2 MB/s  0:00:01
Downloading numpy-2.2.6-cp313-cp313-win_amd64.whl (12.6 MB)
   ---------------------------------------- 0.0/12.6 MB ? eta -:--:--
   ---------------------------------------- 12.6/12.6 MB 82.1 MB/s  0:00:00
Installing collected packages: numpy, opencv-python

   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
```

```
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------------------------------------- 0/2 [numpy]
   ---------- ---------- 1/2 [opencv-python]
   ---------- ---------- 1/2 [opencv-python]
   ---------- ---------- 1/2 [opencv-python]
   ---------- ---------- 1/2 [opencv-python]
   ---------- ---------- 1/2 [opencv-python]
   ---------- ---------- 1/2 [opencv-python]
   ---------- ---------- 1/2 [opencv-python]
   ---------- ---------- 1/2 [opencv-python]
   ---------- ---------- 1/2 [opencv-python]
   ---------- ---------- 1/2 [opencv-python]
   ---------- ---------- 1/2 [opencv-python]
   ---------- ---------- 1/2 [opencv-python]
   ---------- ---------- 1/2 [opencv-python]
   ---------------------------------------- 2/2 [opencv-python]

Successfully installed numpy-2.2.6 opencv-python-4.12.0.88
Note: you may need to restart the kernel to use updated packages.
```

In [5]:
```
pip install matplotlib
```

```
Collecting matplotlib
  Downloading matplotlib-3.10.8-cp313-cp313-win_amd64.whl.metadata (52 kB)
Collecting contourpy>=1.0.1 (from matplotlib)
  Downloading contourpy-1.3.3-cp313-cp313-win_amd64.whl.metadata (5.5 kB)
Collecting cycler>=0.10 (from matplotlib)
  Downloading cycler-0.12.1-py3-none-any.whl.metadata (3.8 kB)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\madhu\appdata\local
\programs\python\python313\lib\site-packages (from matplotlib) (4.61.0)
Collecting kiwisolver>=1.3.1 (from matplotlib)
  Downloading kiwisolver-1.4.9-cp313-cp313-win_amd64.whl.metadata (6.4 kB)
Requirement already satisfied: numpy>=1.23 in c:\users\madhu\appdata\local\progra
ms\python\python313\lib\site-packages (from matplotlib) (2.2.6)
Requirement already satisfied: packaging>=20.0 in c:\users\madhu\appdata\local\pr
ograms\python\python313\lib\site-packages (from matplotlib) (25.0)
Requirement already satisfied: pillow>=8 in c:\users\madhu\appdata\local\programs
\python\python313\lib\site-packages (from matplotlib) (12.0.0)
Collecting pyparsing>=3 (from matplotlib)
  Downloading pyparsing-3.3.1-py3-none-any.whl.metadata (5.6 kB)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\madhu\appdata\loc
al\programs\python\python313\lib\site-packages (from matplotlib) (2.9.0.post0)
Requirement already satisfied: six>=1.5 in c:\users\madhu\appdata\local\programs
\python\python313\lib\site-packages (from python-dateutil>=2.7->matplotlib) (1.1
7.0)
Downloading matplotlib-3.10.8-cp313-cp313-win_amd64.whl (8.1 MB)
   ---------------------------------------- 0.0/8.1 MB ? eta -:--:--
   ---------------------------------------- 8.1/8.1 MB 49.3 MB/s  0:00:00
Downloading contourpy-1.3.3-cp313-cp313-win_amd64.whl (226 kB)
Downloading cycler-0.12.1-py3-none-any.whl (8.3 kB)
Downloading kiwisolver-1.4.9-cp313-cp313-win_amd64.whl (73 kB)
Downloading pyparsing-3.3.1-py3-none-any.whl (121 kB)
Installing collected packages: pyparsing, kiwisolver, cycler, contourpy, matplotl
ib

   ---------------------------------------- 0/5 [pyparsing]
   -------- ------------------------------- 1/5 [kiwisolver]
   ------------------------ --------------- 3/5 [contourpy]
   -------------------------------- ------- 4/5 [matplotlib]
   -------------------------------- ------- 4/5 [matplotlib]
   -------------------------------- ------- 4/5 [matplotlib]
   -------------------------------- ------- 4/5 [matplotlib]
   -------------------------------- ------- 4/5 [matplotlib]
   -------------------------------- ------- 4/5 [matplotlib]
   -------------------------------- ------- 4/5 [matplotlib]
   -------------------------------- ------- 4/5 [matplotlib]
   -------------------------------- ------- 4/5 [matplotlib]
   -------------------------------- ------- 4/5 [matplotlib]
   -------------------------------- ------- 4/5 [matplotlib]
   -------------------------------- ------- 4/5 [matplotlib]
   -------------------------------- ------- 4/5 [matplotlib]
   -------------------------------- ------- 4/5 [matplotlib]
   -------------------------------- ------- 4/5 [matplotlib]
   -------------------------------- ------- 4/5 [matplotlib]
   -------------------------------- ------- 4/5 [matplotlib]
   -------------------------------- ------- 4/5 [matplotlib]
   -------------------------------- ------- 4/5 [matplotlib]
   -------------------------------- ------- 4/5 [matplotlib]
   -------------------------------- ------- 4/5 [matplotlib]
   -------------------------------- ------- 4/5 [matplotlib]
   -------------------------------- ------- 4/5 [matplotlib]
   -------------------------------- ------- 4/5 [matplotlib]
```

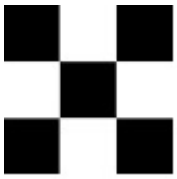
```
   ---------------------------------- ------- 4/5 [matplotlib]
   ---------------------------------- ------- 4/5 [matplotlib]
   ---------------------------------- ------- 4/5 [matplotlib]
   ---------------------------------- ------- 4/5 [matplotlib]
   ---------------------------------- ------- 4/5 [matplotlib]
   ---------------------------------- ------- 4/5 [matplotlib]
   ---------------------------------- ------- 4/5 [matplotlib]
   ---------------------------------- ------- 4/5 [matplotlib]
   ---------------------------------- ------- 4/5 [matplotlib]
   --------------------------------------- 5/5 [matplotlib]

Successfully installed contourpy-1.3.3 cycler-0.12.1 kiwisolver-1.4.9 matplotlib-
3.10.8 pyparsing-3.3.1
Note: you may need to restart the kernel to use updated packages.
```

In [14]:
```
# Displaying 18x18 pixel image.
Image(filename="C:/Users/MADHU/Downloads/chessboard18X18.png")
```

Out[14]:

In [10]:
```
Image(filename="C:/Users/MADHU/Downloads/chessboard84X84.jpg")
```

Out[10]:



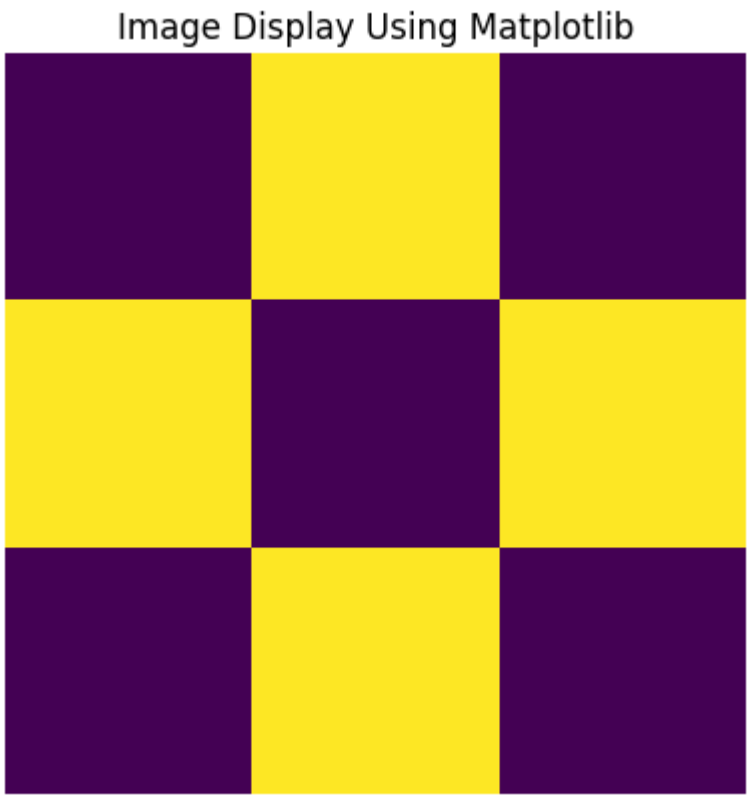
In [15]:
```
# Reading image as gray scale.
cb_img = cv2.imread("C:/Users/MADHU/Downloads/chessboard18X18.png", cv2.IMREAD_G

# Print the image data (pixel values), element of a 2D numpy array.
# Each pixel value is 8-bits [0,255]
print(cb_img)
```

```
[[  0   0   0   0   0   0 255 255 255 255 255 255   0   0   0   0   0   0]
 [  0   0   0   0   0   0 255 255 255 255 255 255   0   0   0   0   0   0]
 [  0   0   0   0   0   0 255 255 255 255 255 255   0   0   0   0   0   0]
 [  0   0   0   0   0   0 255 255 255 255 255 255   0   0   0   0   0   0]
 [  0   0   0   0   0   0 255 255 255 255 255 255   0   0   0   0   0   0]
 [  0   0   0   0   0   0 255 255 255 255 255 255   0   0   0   0   0   0]
 [255 255 255 255 255 255   0   0   0   0   0   0 255 255 255 255 255 255]
 [255 255 255 255 255 255   0   0   0   0   0   0 255 255 255 255 255 255]
 [255 255 255 255 255 255   0   0   0   0   0   0 255 255 255 255 255 255]
 [255 255 255 255 255 255   0   0   0   0   0   0 255 255 255 255 255 255]
 [255 255 255 255 255 255   0   0   0   0   0   0 255 255 255 255 255 255]
 [255 255 255 255 255 255   0   0   0   0   0   0 255 255 255 255 255 255]
 [  0   0   0   0   0   0 255 255 255 255 255 255   0   0   0   0   0   0]
 [  0   0   0   0   0   0 255 255 255 255 255 255   0   0   0   0   0   0]
 [  0   0   0   0   0   0 255 255 255 255 255 255   0   0   0   0   0   0]
 [  0   0   0   0   0   0 255 255 255 255 255 255   0   0   0   0   0   0]
 [  0   0   0   0   0   0 255 255 255 255 255 255   0   0   0   0   0   0]
 [  0   0   0   0   0   0 255 255 255 255 255 255   0   0   0   0   0   0]]
```

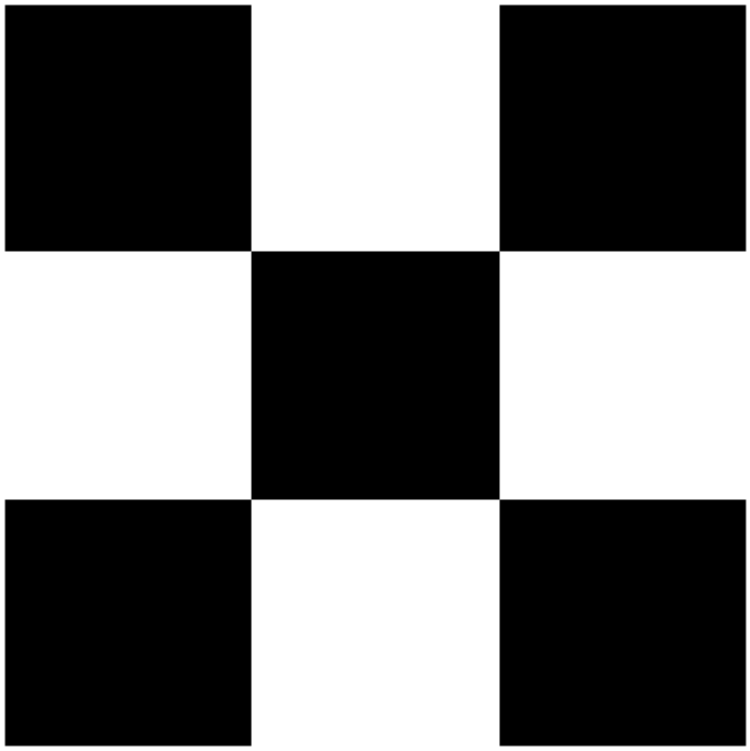
In [16]:
```
# Displaying the image attributes
print("Shape of the image: ", cb_img.shape)
print("Data type of the image: ", cb_img.dtype)
print("Size of the image: ", cb_img.size)
```

```
Shape of the image:  (18, 18)
Data type of the image:  uint8
Size of the image:  324
```

In [17]:
```
# Displaying image using Matplotlib.
plt.imshow(cb_img)
plt.title("Image Display Using Matplotlib")
plt.axis("off") # To turn off axes
plt.show()
```

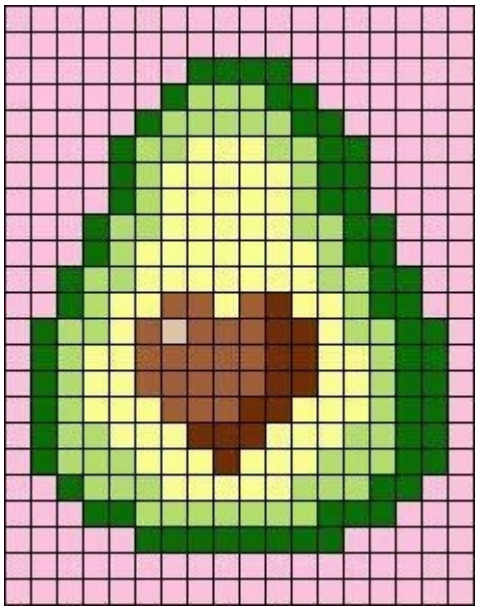


In [18]:
```
# Set color map to gray scale for proper rendering.
plt.imshow(cb_img, cmap = "gray")
plt.axis("off")
plt.show()
```



In [19]:
```
Image("C:/Users/MADHU/Downloads/avacado.jpg")
```

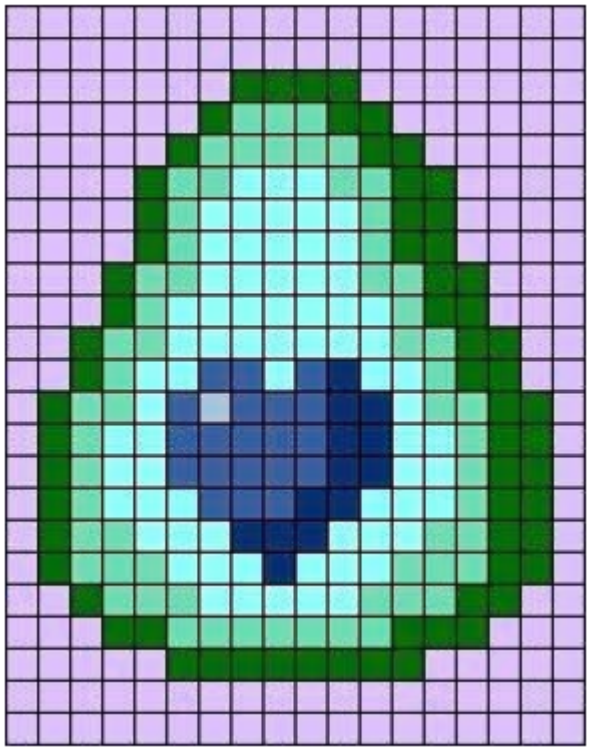
Out[19]:



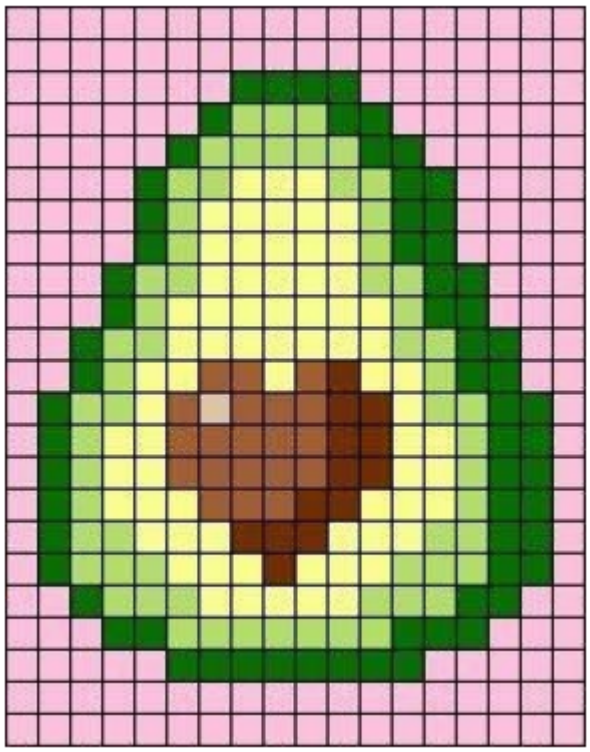
In [21]:
```
# Reading the image
avacado_img = cv2.imread("C:/Users/MADHU/Downloads/avacado.jpg", cv2.IMREAD_COLO

# Printing the shape of the image
print("Image shape (height, width, channels) is:", avacado_img.shape)

# Printing the size of the image
print("Image size is: ", avacado_img.size)

# Printing the data-type of the image
print("Data type of image is:", avacado_img.dtype)
```

```
Image shape (height, width, channels) is: (300, 235, 3)
Image size is:  211500
Data type of image is: uint8
```

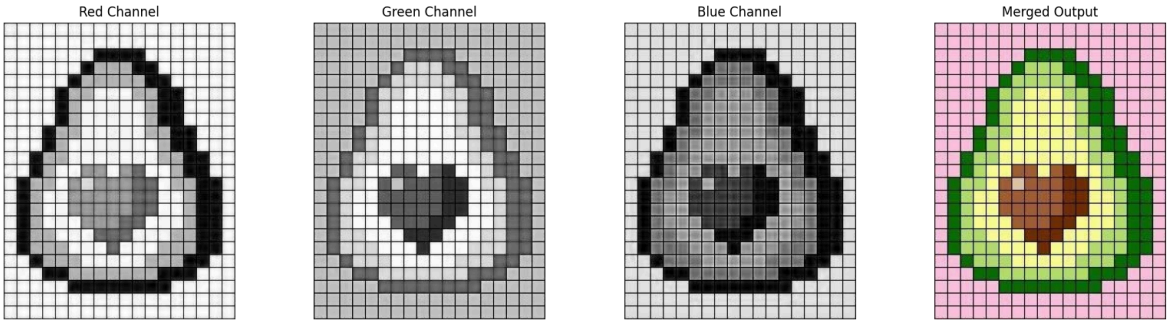
In [22]:
```
# Displaying the image using matplotlib
plt.imshow(avacado_img)
plt.axis("off")
plt.show()
```



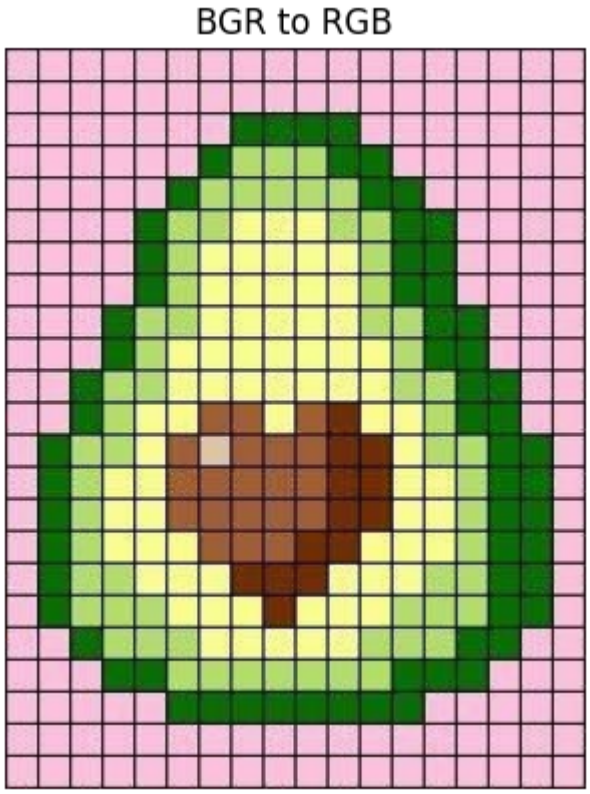
In [23]:
```
# Reversing the channels of the color image
avacado_img_channels_reversed = avacado_img[:, :, ::-1]
plt.imshow(avacado_img_channels_reversed)
plt.axis("off")
plt.show()
```



In [24]:
```
# Split the image into the B,G,R components
b, g, r = cv2.split(avacado_img)

# Show the channels
plt.figure(figsize = [20, 5])

# Red Channel
plt.subplot(1, 4, 1)
plt.imshow(r, cmap = "gray")
plt.title("Red Channel")
plt.axis("off")

# Green Channel
plt.subplot(1, 4, 2)
plt.imshow(g, cmap = "gray")
plt.title("Green Channel")
plt.axis("off")

# Blue Channel
plt.subplot(1, 4, 3)
plt.imshow(b, cmap = "gray")
plt.title("Blue Channel")
plt.axis("off")

# Merge the individual channels into a BGR image
merged_img = cv2.merge((b, g, r))

# Show the merged output
plt.subplot(1, 4, 4)
plt.imshow(merged_img[:, :, ::-1])
plt.title("Merged Output")
plt.axis("off")
plt.show()
```

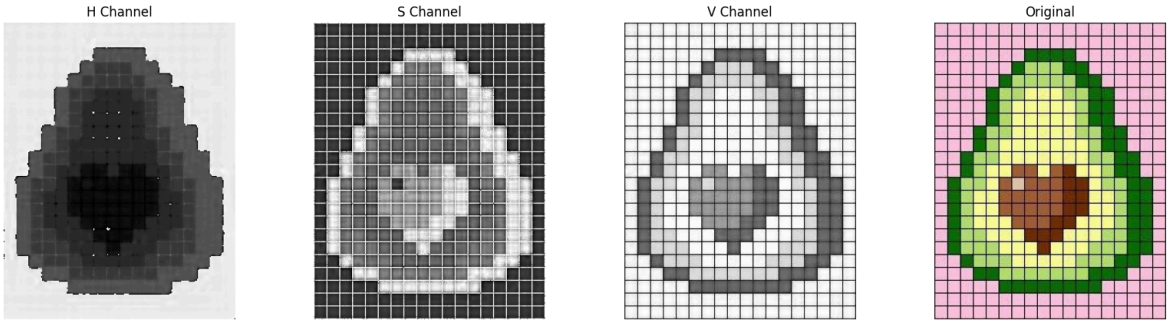


In [26]:
```
# OpenCV stores color channels in a differnet order than most other applications
avacado_img_bgr = cv2.imread("C:/Users/MADHU/Downloads/avacado.jpg", cv2.IMREAD_

# Converting BGR to RGB
avacado_img_rgb = cv2.cvtColor(avacado_img_bgr, cv2.COLOR_BGR2RGB)
plt.imshow(avacado_img_rgb)
plt.title("BGR to RGB")
plt.axis("off")
plt.show()
```

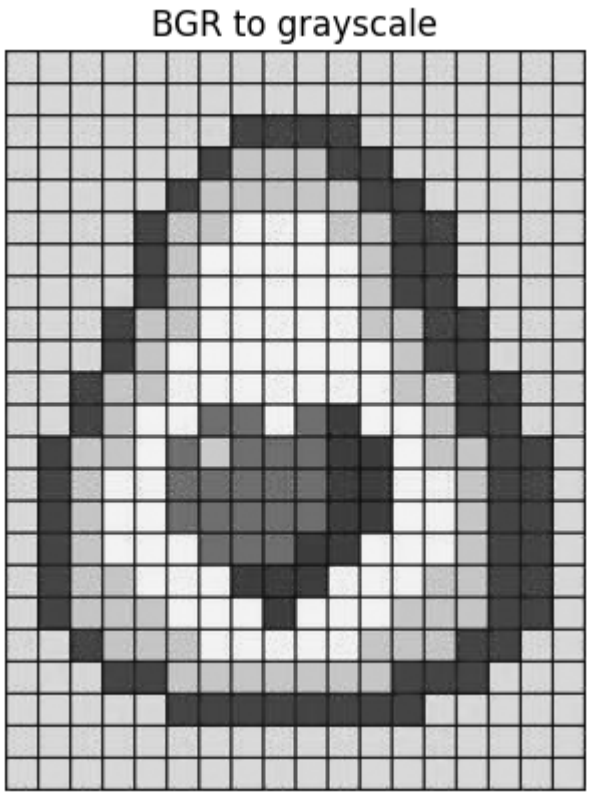


In [27]:

```
# Converting to HSV
avacado_img_hsv = cv2.cvtColor(avacado_img_bgr, cv2.COLOR_BGR2HSV)

# Split the image into H, S, V components
h, s, v = cv2.split(avacado_img_hsv)

# Show the channels
plt.figure(figsize = [20, 5])

# Hue Channel
plt.subplot(1, 4, 1)
plt.imshow(h, cmap = "gray")
plt.title("H Channel")
plt.axis("off")

# Saturation Channel
plt.subplot(1, 4, 2)
plt.imshow(s, cmap = "gray")
plt.title("S Channel")
plt.axis("off")

# Value Channel
plt.subplot(1, 4, 3)
plt.imshow(v, cmap = "gray")
plt.title("V Channel")
plt.axis("off")

# Show the original image
plt.subplot(1, 4, 4)
plt.imshow(avacado_img_rgb)
plt.title("Original")
plt.axis("off")
plt.show()
```

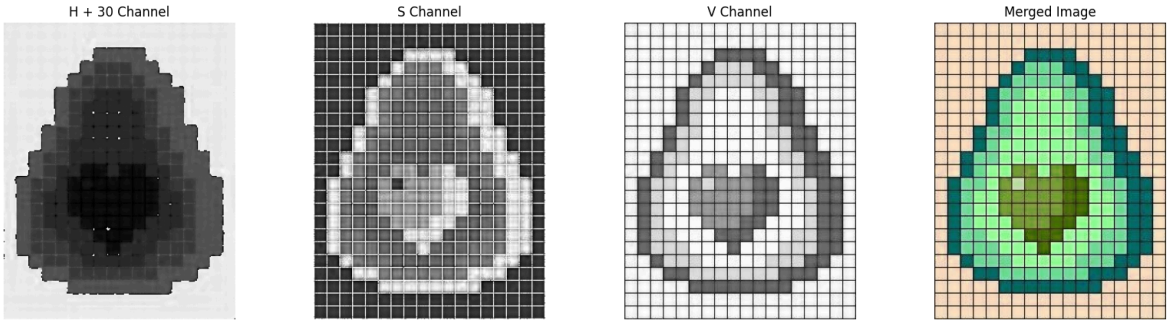


In [28]:
```
# Converting BGR to grayscale
avacado_img_gray = cv2.cvtColor(avacado_img_bgr, cv2.COLOR_BGR2GRAY)
plt.imshow(avacado_img_gray, cmap = "gray")
plt.title("BGR to grayscale")
plt.axis("off")
plt.show()
```



In [29]:
```
h_new = h + 30
avacado_img_merged = cv2.merge((h_new, s, v))
avacado_img_rgb = cv2.cvtColor(avacado_img_merged, cv2.COLOR_HSV2RGB)

# Split the image into the B,G,R components
h,s,v = cv2.split(avacado_img_merged)

# Show the channels
plt.figure(figsize = [20, 5])

# Hue Channel
plt.subplot(1, 4, 1)
plt.imshow(h, cmap = "gray")
plt.title("H + 30 Channel")
plt.axis("off")

# Saturation Channel
```

```
plt.subplot(1, 4, 2)
plt.imshow(s, cmap = "gray")
plt.title("S Channel")
plt.axis("off")

# Value Channel
plt.subplot(1, 4, 3)
plt.imshow(v, cmap = "gray")
plt.title("V Channel")
plt.axis("off")

# Show the original image
plt.subplot(1, 4, 4)
plt.imshow(avacado_img_rgb)
plt.title("Merged Image")
plt.axis("off")
plt.show()
```



In [30]:
```
# Saving the image
cv2.imwrite("avacado_saved.jpg", avacado_img_rgb)
```

Out[30]: True

In [ ]:
```

```