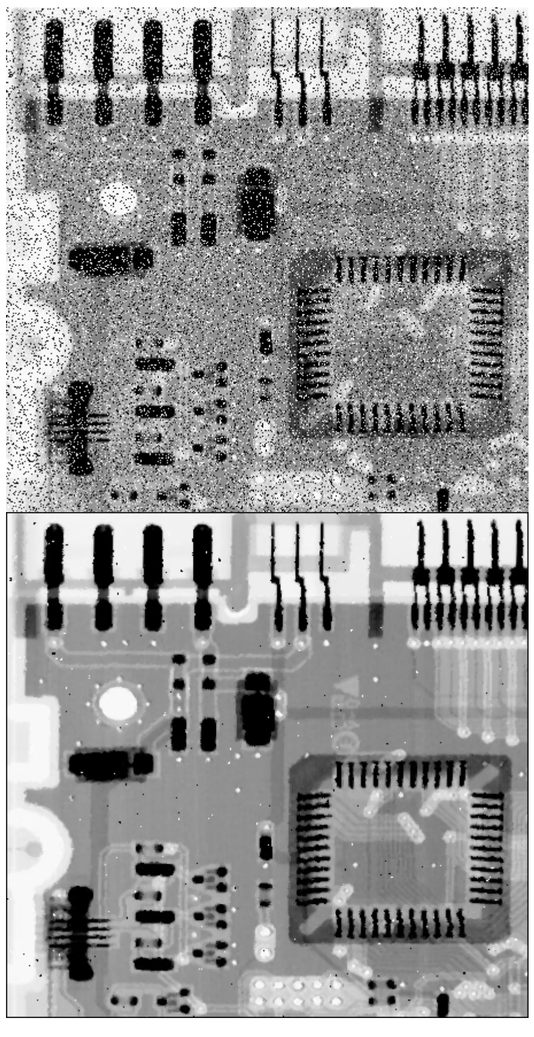
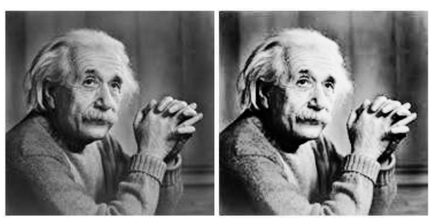
```
#·Median·Spatial·Domain·Filtering


import cv2
import numpy as np
from google.colab.patches import cv2_imshow

# Read the image
img = cv2.imread('/content/noisysalterpepper.png', 0)
cv2_imshow(img)

m, n = img.shape
newimage = np.zeros([m, n])

for i in range(1, m-1):
  for j in range(1, n-1):
    temp = [img[i-1, j-1],
      img[i-1, j],
      img[i-1, j + 1],
      img[i, j-1],
      img[i, j],
      img[i, j + 1],
      img[i + 1, j-1],
      img[i + 1, j],
      img[i + 1, j + 1]]

    temp = sorted(temp)
    newimage[i, j]= temp[4]

newimage = newimage.astype(np.uint8)
cv2_imshow(newimage)
```



```
# Histogram Equalization of Image
import cv2
import numpy as np
from google.colab.patches import cv2_imshow

# read a image using imread
img = cv2.imread('/content/einstein.jpg', 0)

# creating a Histograms Equalization
# of a image using cv2.equalizeHist()
equ = cv2.equalizeHist(img)

# stacking images side-by-side
res = np.hstack((img, equ))

# show image input vs output
cv2_imshow(res)

cv2.waitKey(0)
cv2.destroyAllWindows()
```



```
#·Shaprening·Image·Using·Kernel
import cv2
import numpy as np
from google.colab.patches import cv2_imshow

image = cv2.imread('/content/koala1-300x225.jpeg', flags=cv2.IMREAD_COLOR)
cv2_imshow(image)
kernel = np.array([[-1, -1, -1],
                   [-1, 9,-1],
                   [-1, -1, -1]])
image_sharp = cv2.filter2D(src=image, ddepth=-1, kernel=kernel)
cv2_imshow(image_sharp)
cv2.waitKey()
cv2.destroyAllWindows()
```

```
# Shaprening Image Using Kernel

import cv2
import numpy as np
from google.colab.patches import cv2_imshow

image = cv2.imread('/content/koala1-300x225.jpeg', flags=cv2.IMREAD_COLOR)
```

```
    cv2_imshow(image)
    kernel = np.array([[1, 1, 1],
                       [1, -7,1],
                       [1, 1, 1]])
    image_sharp = cv2.filter2D(src=image, ddepth=-1, kernel=kernel)
    cv2_imshow(image_sharp)
    cv2.waitKey()
    cv2.destroyAllWindows()
```

✓ 0s completed at 11:33 PM