In [1]:
```
import cv2
import numpy as np
import matplotlib.pyplot as plt
```

In [6]:
```
#read an image
path="car1.jpg"
image = cv2.imread(path)
path
```

Out[6]:
```
'car1.jpg'
```

In [7]:
```
#display an image
cv2.imshow("Original image",image)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

In [8]:
```
image.shape
```

Out[8]:
```
(337, 600, 3)
```

In [9]:
```
#Gray Scale image

img_gray=cv2.imread(path,cv2.IMREAD_GRAYSCALE)
cv2.imshow("GrayScale car image",img_gray)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

In [10]:
```
img_gray.shape
```

Out[10]:
```
(337, 600)
```

In [11]:
```
#image Slicing
image=cv2.imread(path)
crop=image[0:100,0:100]
cv2.imshow("Crop Image",crop)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

In [12]:
```
# image resize

scale= 200
image=cv2.imread(path)
width= int(image.shape[1]* scale/100)
height= int(image.shape[0]*scale/100)

dim= (width,height)

#resize
img_resized= cv2.resize(image,dim,interpolation= cv2.INTER_AREA)

cv2.imshow("resized image",img_resized)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

In [13]:
```
img_resized.shape
```

Out[13]:
```
(674, 1200, 3)
```

In [14]:
```
#rotate an image

image=cv2.imread(path)
h,w,c=image.shape

scale=1
center=(w/2,h/2)
angle=90
m=cv2.getRotationMatrix2D(center,angle,scale)
img_rotate=cv2.warpAffine(image,m,(h,w))
cv2.imshow("rotated Image",img_rotate)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

In [15]:
```
#finding edges from an image
image=cv2.imread(path)
edges=cv2.Canny(image,150,250)
cv2.imshow("Canny image",edges)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

In [16]:
```
# blur filter on image
image = cv2.imread(path)
blur = cv2.blur(image,(4,4))
cv2.imshow('Blurred image',np.hstack((image,blur)))
cv2.waitKey(1)
cv2.destroyAllWindows()
```

In [17]:
```
#Gaussian / Median filter to blur
image=cv2.imread(path)
dst=median=cv2.medianBlur(image,5)
cv2.imshow("Median Blurred image",np.hstack((image,dst)))
cv2.waitKey(0)
cv2.destroyAllWindows()
```

In [18]:
```
#Bilateral filter
image=cv2.imread(path)
kernel=np.ones((5,5),np.float32)/25

bilateral_blur=cv2.bilateralFilter(image,9,25,25)
cv2.imshow("Median Blurred image",np.hstack((image,bilateral_blur)))
cv2.waitKey(0)
cv2.destroyAllWindows()
```

In [19]:
```
#Box Filter
image=cv2.imread(path)
box_img=cv2.boxFilter(image,0,(3,3),(-1,-1))
cv2.imshow("Box filter image",np.hstack((image,box_img)))
cv2.waitKey(0)
cv2.destroyAllWindows()
```

In [20]:
```
#image binarization

image=cv2.imread(path)
img_gray=cv2.imread(path,cv2.IMREAD_GRAYSCALE)
value,thresh=cv2.threshold(img_gray,100,250,cv2.THRESH_BINARY)
cv2.imshow("Binary image",thresh)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

In [21]:
```
#Draw a line

image=cv2.imread(path)
cv2.line(image,(0,0),(170,270),(255,255,0),5)
cv2.imshow("Line on image",image)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

In [22]:
```
# draw a rectangle
image= cv2.imread(path)
cv2.rectangle(image,(15,20),(570,280),(170,25,215),3)
cv2.imshow('Rectangle on image',image)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

In [ ]:
```

```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: