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

In [2]:
```
#0. CV_CAP_PROP_POS_MSEC Current position of the video file in milliseconds.
#1. CV_CAP_PROP_POS_FRAMES 0-based index of the frame to be decoded/captured next.
#2. CV_CAP_PROP_POS_AVI_RATIO Relative position of the video file
#3. CV_CAP_PROP_FRAME_WIDTH Width of the frames in the video stream.
#4. CV_CAP_PROP_FRAME_HEIGHT Height of the frames in the video stream.
#5. CV_CAP_PROP_FPS Frame rate.
#6. CV_CAP_PROP_FOURCC 4-character code of codec.
#7. CV_CAP_PROP_FRAME_COUNT Number of frames in the video file.
#8. CV_CAP_PROP_FORMAT Format of the Mat objects returned by retrieve() .
#9. CV_CAP_PROP_MODE Backend-specific value indicating the current capture mode.
#10. CV_CAP_PROP_BRIGHTNESS Brightness of the image (only for cameras).
#11. CV_CAP_PROP_CONTRAST Contrast of the image (only for cameras).
#12. CV_CAP_PROP_SATURATION Saturation of the image (only for cameras).
#13. CV_CAP_PROP_HUE Hue of the image (only for cameras).
#14. CV_CAP_PROP_GAIN Gain of the image (only for cameras).
#15. CV_CAP_PROP_EXPOSURE Exposure (only for cameras).
#16. CV_CAP_PROP_CONVERT_RGB Boolean flags indicating whether images should be converted to RGB.
#17. CV_CAP_PROP_WHITE_BALANCE Currently unsupported
#18. CV_CAP_PROP_RECTIFICATION Rectification flag for stereo cameras (note: only supported by DC1394 v 2.x backer
```

In [3]:
```
framewidth = 1000
frameheight = 1000
cap = cv2.VideoCapture(0) # by default 0 isliye liya h live cam k liye
cap.set(3,framewidth)
cap.set(4,frameheight)
cap.set(10,500)
cap.set(15,300)

while True:
    success,img = cap.read()
    cv2.imshow('Webcam',img)
    if cv2.waitKey(5) & 0xFF==ord('q'): ## wiatkey(5) frame k beech ka delay hai
        break

cap.release()
cv2.destroyAllWindows()
```

In [4]:
```
## VEDIO CAPTURING
import cv2
cap = cv2.VideoCapture('C:/Users/user/Desktop/surfing.avi')
frame_time = 1

while True:
    success,img = cap.read()
    if not success:
        print("Vedio is Ended or corrupt")
        break
    cv2.imshow('Vedio',img)
    if cv2.waitKey(frame_time) & 0xFF==ord('q'):## wiatkey(5) frame k beech ka delay hai
        break
cap.release()
cv2.destroyAllWindows()
```

In [5]:
```
## Vedio RGB to Grayscale
import cv2
cap = cv2.VideoCapture('C:/Users/user/Desktop/surfing.avi')
frame_time = 20

while True:
    success,img = cap.read()
    if not success:
        print("Vedio is Ended or corrupt")
        break
    grey = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)
    cv2.imshow('Vedio',grey)
    if cv2.waitKey(frame_time) & 0xFF==ord('q'):## wiatkey(5) frame k beech ka delay hai
        break
cap.release()
cv2.destroyAllWindows()
```

In [6]:
```
# Vedio BGR to black and white
import cv2
cap = cv2.VideoCapture('C:/Users/user/Desktop/surfing.avi')
frame_time = 20
```

```
while True:
    success,img = cap.read()
    if not success:
        print("Vedio is Ended or corrupt")
        break
    grey = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)
    ret,bin_image = cv2.threshold(grey,100,255,cv2.THRESH_BINARY)
    cv2.imshow('Black White',bin_image)
    #cv2.imshow('Vedio',grey)
    if cv2.waitKey(frame_time) & 0xFF==ord('q'):## wiatkey(5) frame k beech ka delay hai
        break
cap.release()
cv2.destroyAllWindows()
```

In [7]:

```
# RGB BGR Black and white run in one format
import cv2
cap = cv2.VideoCapture('C:/Users/user/Desktop/surfing.avi')
frame_time = 20

cap.set(3,100)
cap.set(4,100)

while True:
    success,img = cap.read()
    if not success:
        print("Vedio is Ended or corrupt")
        break
    cv2.imshow('Color',img)
    grey = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)
    cv2.imshow('Gray',grey)
    ret,bin_image = cv2.threshold(grey,100,255,cv2.THRESH_BINARY)
    cv2.imshow('Black White',bin_image)
    #cv2.imshow('Vedio',grey)
    if cv2.waitKey(frame_time) & 0xFF==ord('q'):## wiatkey(5) frame k beech ka delay hai
        break
cap.release()
cv2.destroyAllWindows()
```

In [11]:

```
## Saving a Vedio
framewidth = 640
frameheight = 480
cap = cv2.VideoCapture(0)
cap.set(3,framewidth)
cap.set(4,frameheight)
cap.set(10,800)
cap.set(15,800)

fourcc = cv2.VideoWriter_fourcc(*'XVID')
out = cv2.VideoWriter('class_20_feb.avi',fourcc,60,(framewidth,frameheight))

while True:
    s,img = cap.read()
    out.write(img)
    cv2.imshow('Live Cam',img)
    if cv2.waitKey(frame_time) & 0xFF==ord('q'):
        break
cap.release()
cv2.destroyAllWindows()
print("The Vedio is Saved Successfully")
```

```
The Vedio is Saved Successfully
```

In [ ]: