**Aim:**

**To study concepts of video processing and perform operations using Opencv for:**

1. **Capturing and Displaying a video**
2. **Capturing and saving a video**
3. **To display various properties of a Video**

## 1) Capturing and Displaying a Video

**Code:**

```
import cv2
vid_capture = cv2.VideoCapture(r"D:\\SEM6\\ImageProcessing\\Lab8\\video.mp4")
if (vid_capture.isOpened() == False):
    print("Error opening the video file")
else:
    fps = vid_capture.get(5)
    print('Frames per second : ', fps, 'FPS')
    frame_count = vid_capture.get(7)
    print('Frame count : ', frame_count)
while (vid_capture.isOpened()):
    ret, frame = vid_capture.read()
    if ret == True:
        cv2.imshow('Frame', frame)
        key = cv2.waitKey(20)
        if key == ord('q'):
            break
    else:
        break
vid_capture.release()
cv2.destroyAllWindows()
```

**Output**:


## 2) Capturing and saving a video

**Code:**

```
import cv2
video = cv2.VideoCapture(r"D:\\SEM6\\ImageProcessing\\Lab8\\video.mp4")


if (video.isOpened() == False):
    print("Error reading video file")

frame_height = int(video.get(4))
frame_width = int(video.get(3))
size = (frame_width, frame_height)


result = cv2.VideoWriter('filename.avi',
                         cv2.VideoWriter_fourcc(*'MJPG'),
                         10, size)

while (True):
```

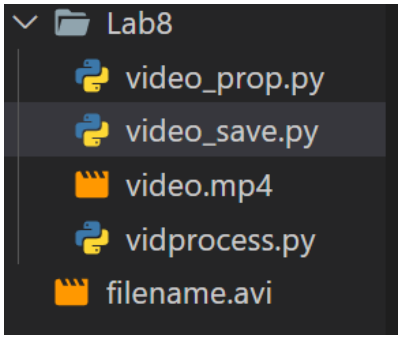
```
    ret, frame = video.read()

    if ret == True:
        result.write(frame)
        cv2.imshow('Frame', frame)
        if cv2.waitKey(1) & 0xFF == ord('s'):
            break
    else:
        break

video.release()
result.release()

cv2.destroyAllWindows()
```

**Output:**




**3) To display various properties of a Video**

**Code:**

```
import cv2

# For Video File
capture=cv2.VideoCapture(0)

print("CV_CAP_PROP_FRAME_WIDTH:
'{}'".format(capture.get(cv2.CAP_PROP_FRAME_WIDTH)))
```

```
print("CV_CAP_PROP_FRAME_HEIGHT :
'{}'".format(capture.get(cv2.CAP_PROP_FRAME_HEIGHT)))
print("CAP_PROP_FPS : '{}'".format(capture.get(cv2.CAP_PROP_FPS)))
print("CAP_PROP_POS_MSEC :
'{}'".format(capture.get(cv2.CAP_PROP_POS_MSEC)))
print("CAP_PROP_FRAME_COUNT  :
'{}'".format(capture.get(cv2.CAP_PROP_FRAME_COUNT)))
print("CAP_PROP_BRIGHTNESS :
'{}'".format(capture.get(cv2.CAP_PROP_BRIGHTNESS)))
print("CAP_PROP_CONTRAST :
'{}'".format(capture.get(cv2.CAP_PROP_CONTRAST)))
print("CAP_PROP_SATURATION :
'{}'".format(capture.get(cv2.CAP_PROP_SATURATION)))
print("CAP_PROP_HUE : '{}'".format(capture.get(cv2.CAP_PROP_HUE)))
print("CAP_PROP_GAIN  : '{}'".format(capture.get(cv2.CAP_PROP_GAIN)))
print("CAP_PROP_CONVERT_RGB :
'{}'".format(capture.get(cv2.CAP_PROP_CONVERT_RGB)))


capture.release()
cv2.destroyAllWindows()
```

**Output:**