1. Reading a Video

```
import cv2

# Video file path
video_path = r'C:\Users\HP\Downloads\archive (1)\drivingDataset\normalDay\nD_20.mp4'

# Create VideoCapture object
cap = cv2.VideoCapture(video_path)

# Check if video opened successfully
if not cap.isOpened():
    print("Error opening video file")
    exit()

# Read and display frames
while cap.isOpened():
    ret, frame = cap.read()
    if ret:
        cv2.imshow('Video Frame', frame)

        # Press 'q' to quit
        if cv2.waitKey(25) & 0xFF == ord('q'):
            break
    else:
        break

# Release resources
cap.release()
cv2.destroyAllWindows()
```

2). Write Video

```
import cv2
import os

# Video file path
video_path = r'C:\Users\HP\Downloads\archive (1)\drivingDataset\normalDay\nD_20.mp4'

# Create VideoCapture object
cap = cv2.VideoCapture(video_path)

# Check if video opened successfully
if not cap.isOpened():
    print("Error opening video file")
```

```
    exit()

# Get video properties
frame_width = int(cap.get(cv2.CAP_PROP_FRAME_WIDTH))
frame_height = int(cap.get(cv2.CAP_PROP_FRAME_HEIGHT))
fps = int(cap.get(cv2.CAP_PROP_FPS))

print("Frame width:", frame_width)
print("Frame height:", frame_height)
print("FPS:", fps)

# Define codec and output path
fourcc = cv2.VideoWriter_fourcc(*'XVID')
output_path = os.path.abspath("output.mp4")
print("\nSaving output video at →", output_path)

# Create VideoWriter object
out = cv2.VideoWriter(output_path, fourcc, fps, (frame_width, frame_height))

while True:
    ret, frame = cap.read()
    if not ret:
        print("Can't receive frame (stream end?). Exiting...")
        break

    # Display frame
    cv2.imshow('Video Frame', frame)

    # Flip the frame vertically
    flipped_frame = cv2.flip(frame, 0)

    # Write flipped frame
    out.write(flipped_frame)

    # Exit on ESC key
    if cv2.waitKey(25) == 27:
        break

# Release resources
cap.release()
out.release()
cv2.destroyAllWindows()

# Auto-open the output video (Windows)
```

```
print("\nOpening output video...")
os.startfile(output_path)
```