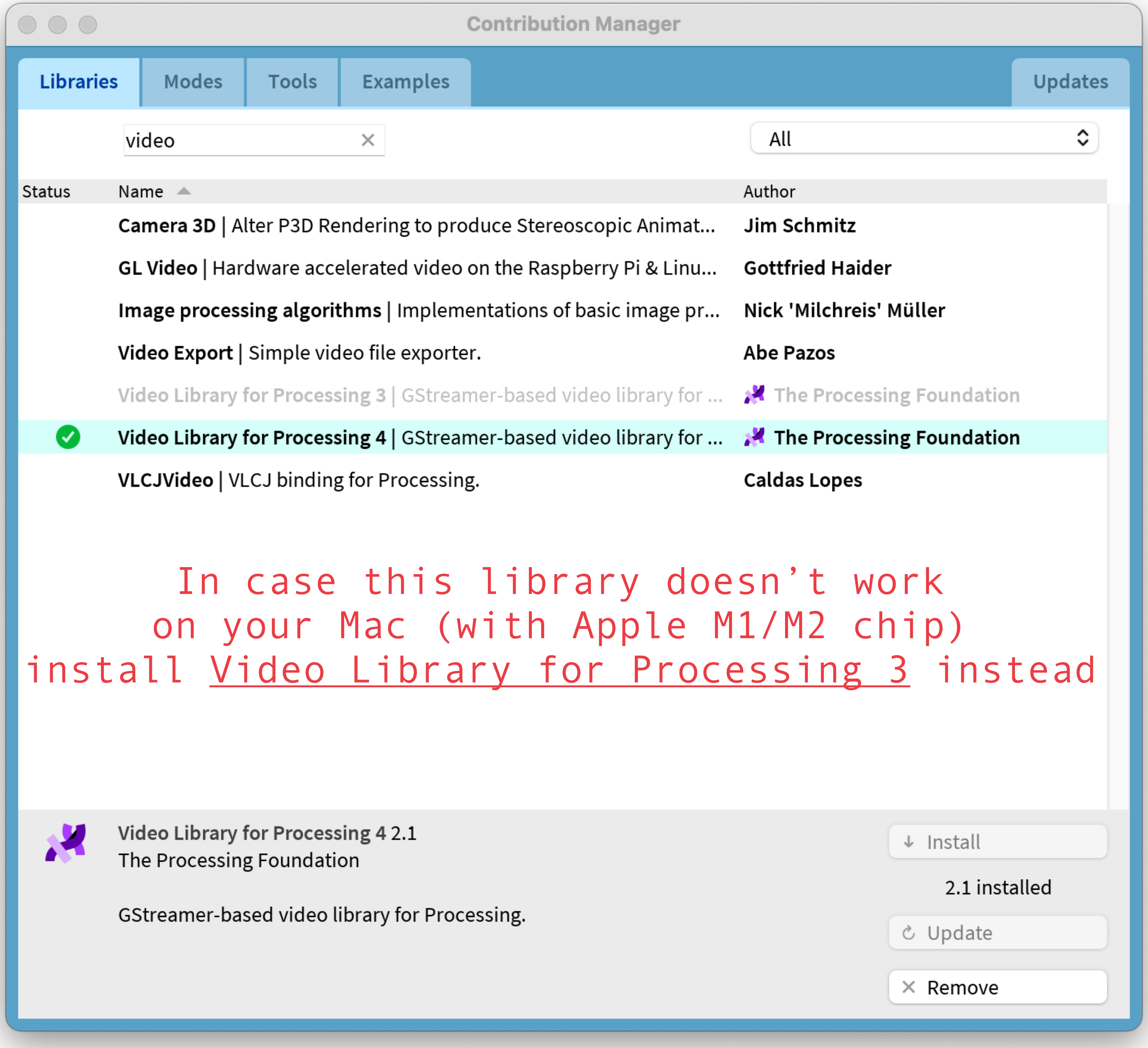
WEEK 5
VIDEO PROCESSING

Installing Processing Video Library



video ✕

All ⌃

Status	Name	Author
	Camera 3D | Alter P3D Rendering to produce Stereoscopic Animat...	**Jim Schmitz**
	GL Video | Hardware accelerated video on the Raspberry Pi & Linu...	**Gottfried Haider**
	Image processing algorithms | Implementations of basic image pr...	**Nick 'Milchreis' Müller**
	Video Export | Simple video file exporter.	**Abe Pazos**
	Video Library for Processing 3 | GStreamer-based video library for ...	**The Processing Foundation**
✓	**Video Library for Processing 4** | GStreamer-based video library for ...	**The Processing Foundation**
	VLCJVideo | VLCJ binding for Processing.	**Caldas Lopes**

In case this library doesn't work
on your Mac (with Apple M1/M2 chip)
install Video Library for Processing 3 instead

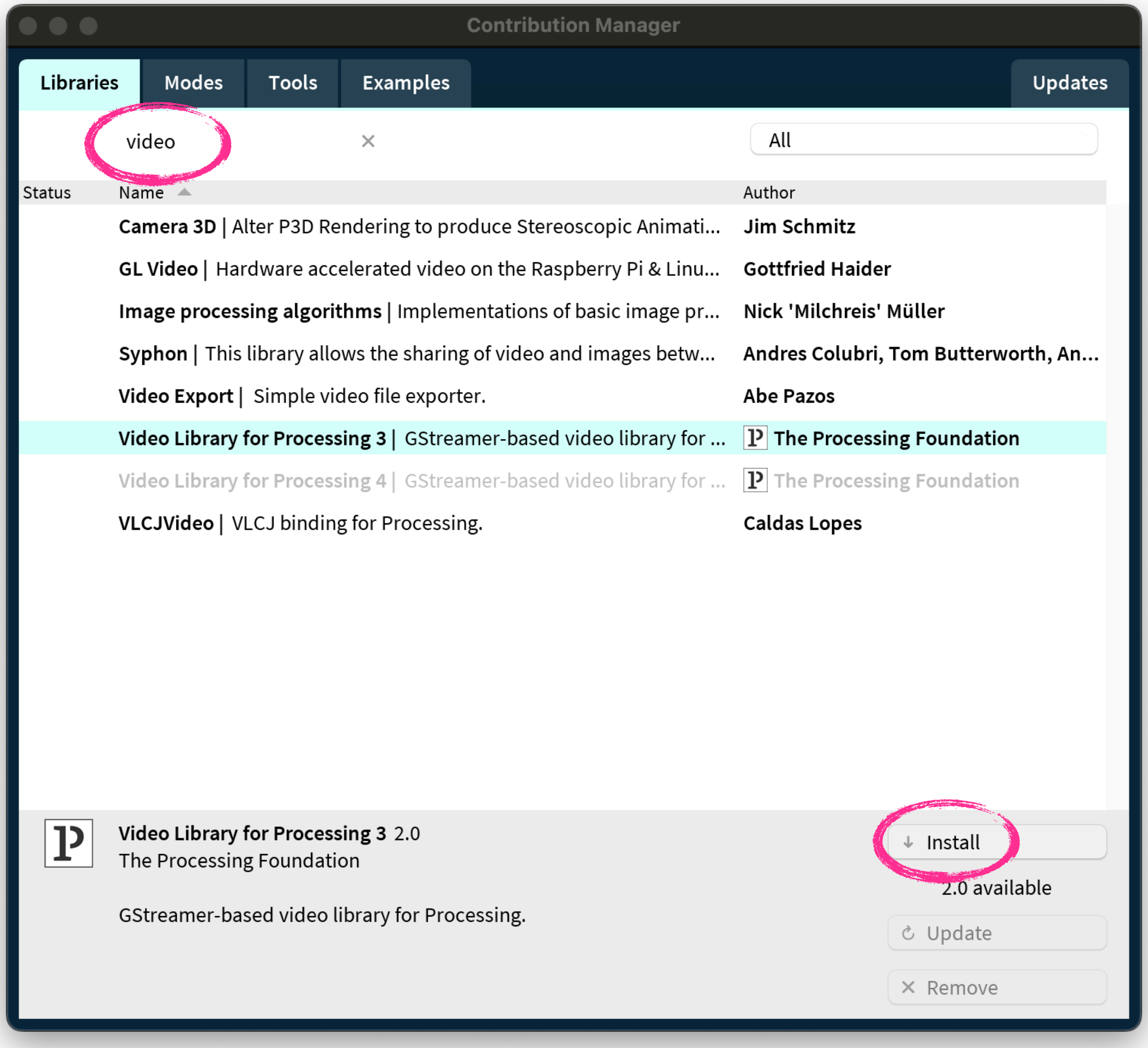
Video Library for Processing 4 2.1
The Processing Foundation

GStreamer-based video library for Processing.

2.1 installed

Installing
Video Library for Processing 3

- Step 1: Quit Processing 4 completely
- Step 2: Launch Processing 3.5.4, install “Video Library for Processing 3” in the Contribution Manager
- Step 3: Quit Processing 3.5.4 completely, launch Processing 4



Status	Name	Author
	Camera 3D | Alter P3D Rendering to produce Stereoscopic Animati...	**Jim Schmitz**
	GL Video | Hardware accelerated video on the Raspberry Pi & Linu...	**Gottfried Haider**
	Image processing algorithms | Implementations of basic image pr...	**Nick 'Milchreis' Müller**
	Syphon | This library allows the sharing of video and images betw...	**Andres Colubri, Tom Butterworth, An...**
	Video Export | Simple video file exporter.	**Abe Pazos**
	Video Library for Processing 3 | GStreamer-based video library for ...	**The Processing Foundation**
	Video Library for Processing 4 | GStreamer-based video library for ...	The Processing Foundation
	VLCJVideo | VLCJ binding for Processing.	**Caldas Lopes**

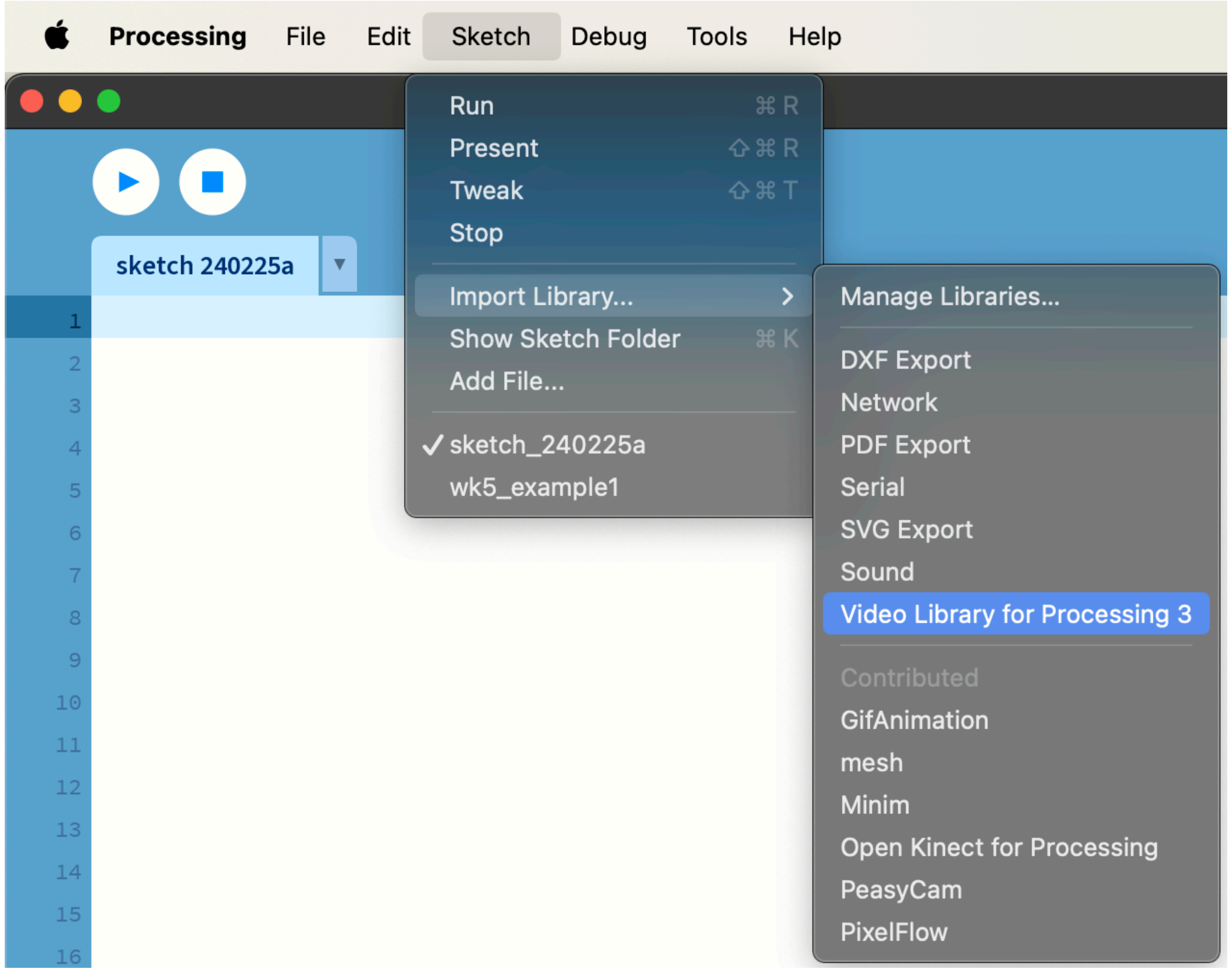
Video Library for Processing 3 2.0
The Processing Foundation

GStreamer-based video library for Processing.

2.0 available

Processing Video library

```
import processing.video.*;
```



Two classes:

Movie (file) and Capture (camera)

MOVIE class

Loading and playing movies located in the sketch's data folder or an accessible place on the network, for example:

```
import processing.video.*;
Movie myMovie;

void setup() {
  myMovie = new Movie(this, “filename.mp4");
  myMovie.play(); //or myMovie.loop();
}

void draw() {
  image(myMovie, 0, 0);
}

void movieEvent(Movie m) {
  m.read();
}
```

Import the video library

Declare a Movie instance

Load movie (constructor)

Play the movie

Display the current frame from the movie clip.

Called every time a new frame is available to read

EXAMPLE 1

```
// Street of Crocodiles (1986)
// by Brothers Quay

import processing.video.*;

Movie movie;

void setup() {
  size(480, 360);
  background(0);
  movie = new Movie(this, "Street of Crocodiles excerpt 1.mp4");
  //movie.play(); // plays one time
  movie.loop(); // plays continuously
}

// Called every time a new frame is available to read
void movieEvent(Movie m) {
  m.read(); // reads the current frame
}

void draw() {
  // alternative to movieEvent()
  //if (movie.available() == true) {
  //  movie.read();
  //}
  image(movie, 0, 0);
}
```

Video loading might take a few seconds. Be patient!

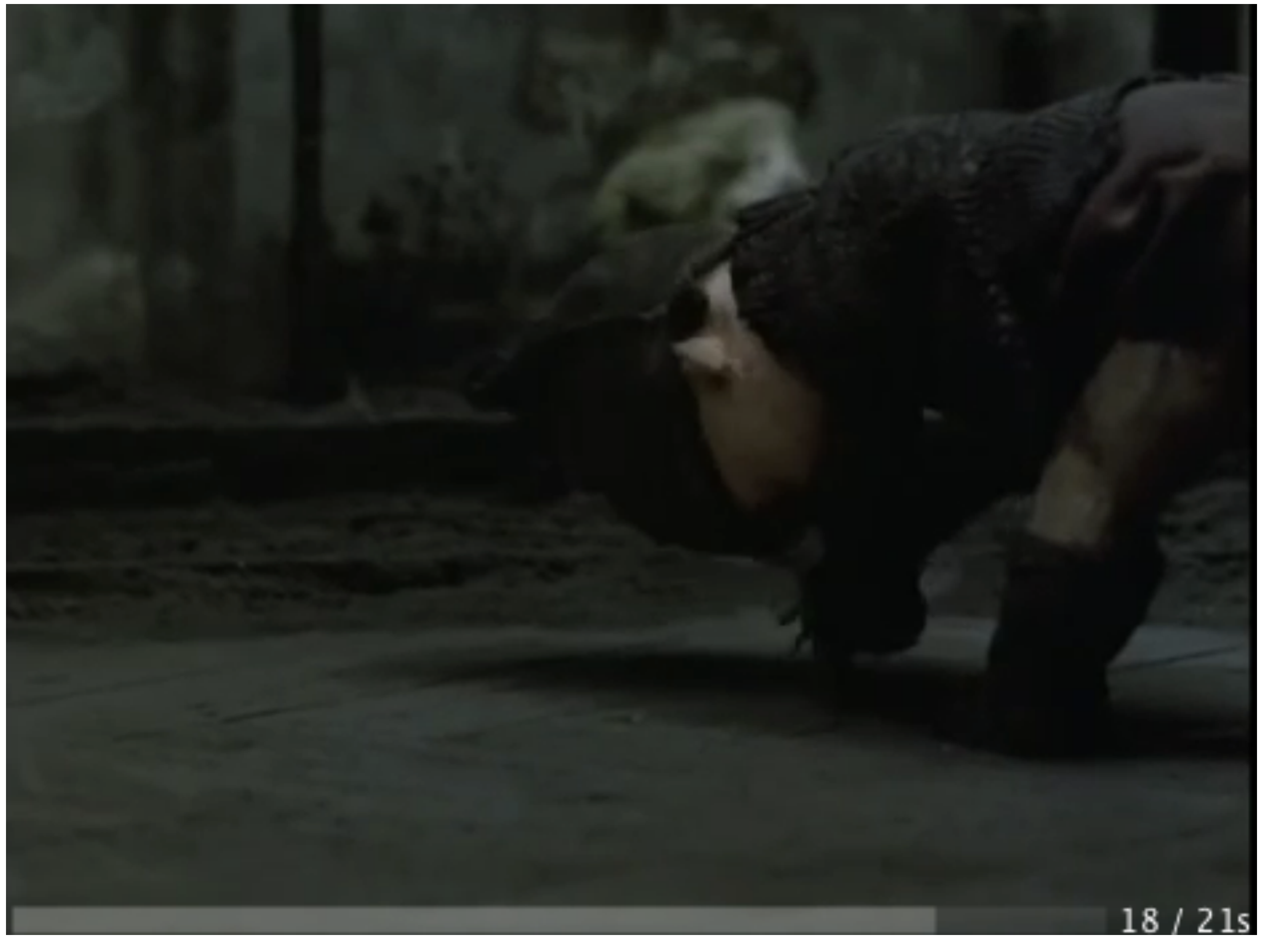
Method 1 →

Method 2 →

Methods of the Movie class

Methods	`frameRate()`	Sets how often frames are read from the movie.
	`speed()`	Sets the relative playback speed of the movie.
	`duration()`	Returns the length of the movie in seconds.
	`time()`	Returns the location of the playback head in seconds.
	`jump()`	Jumps to a specific location within a movie.
	`available()`	Returns ”true” when a new movie frame is available to read.
	`play()`	Plays a movie one time and stops at the last frame.
	`loop()`	Plays a movie continuously, restarting it when it’s over.
	`noLoop()`	If a movie is looping, this will cause it to play until the end and then stop on the last frame.
	`pause()`	Pauses a movie during playback.
	`stop()`	Stops a movie from continuing.
	`read()`	Reads the current frame of the movie.

EXAMPLE 2 ADDING TIME BAR

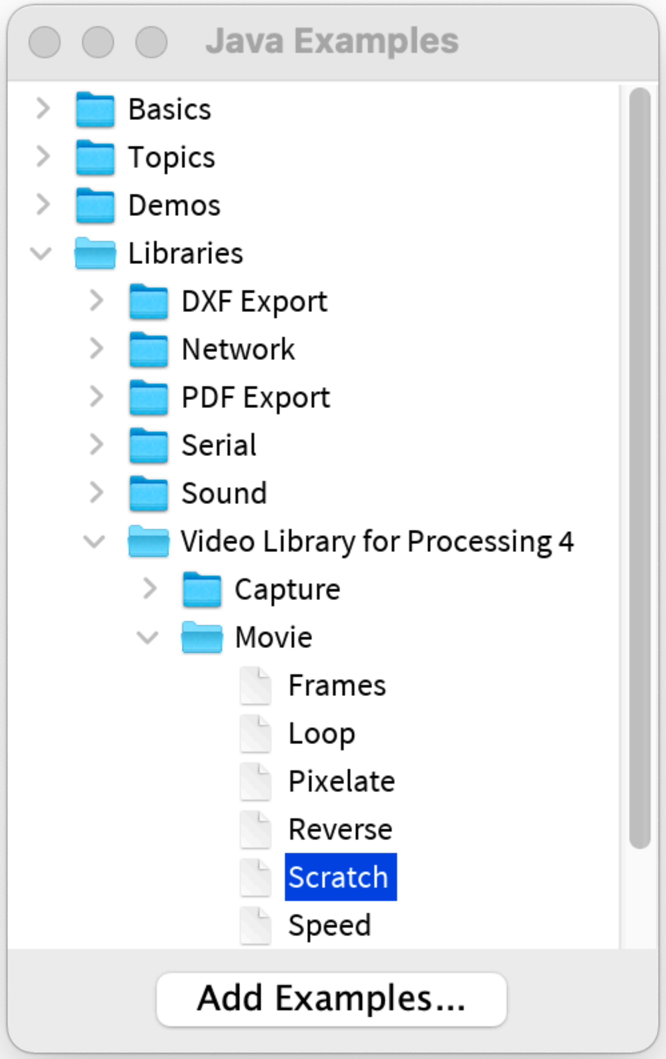


```
line(10, height-10, map(movie.time(), 0, movie.duration(), 10, width-60), height-10);
text(round(movie.time())+” / "+round(movie.duration())+"s", width-55, height-5);
```

EXERCISE 1

- Based on Example 1, create a sketch that jumps to a random frame within a movie whenever the mouse button is clicked

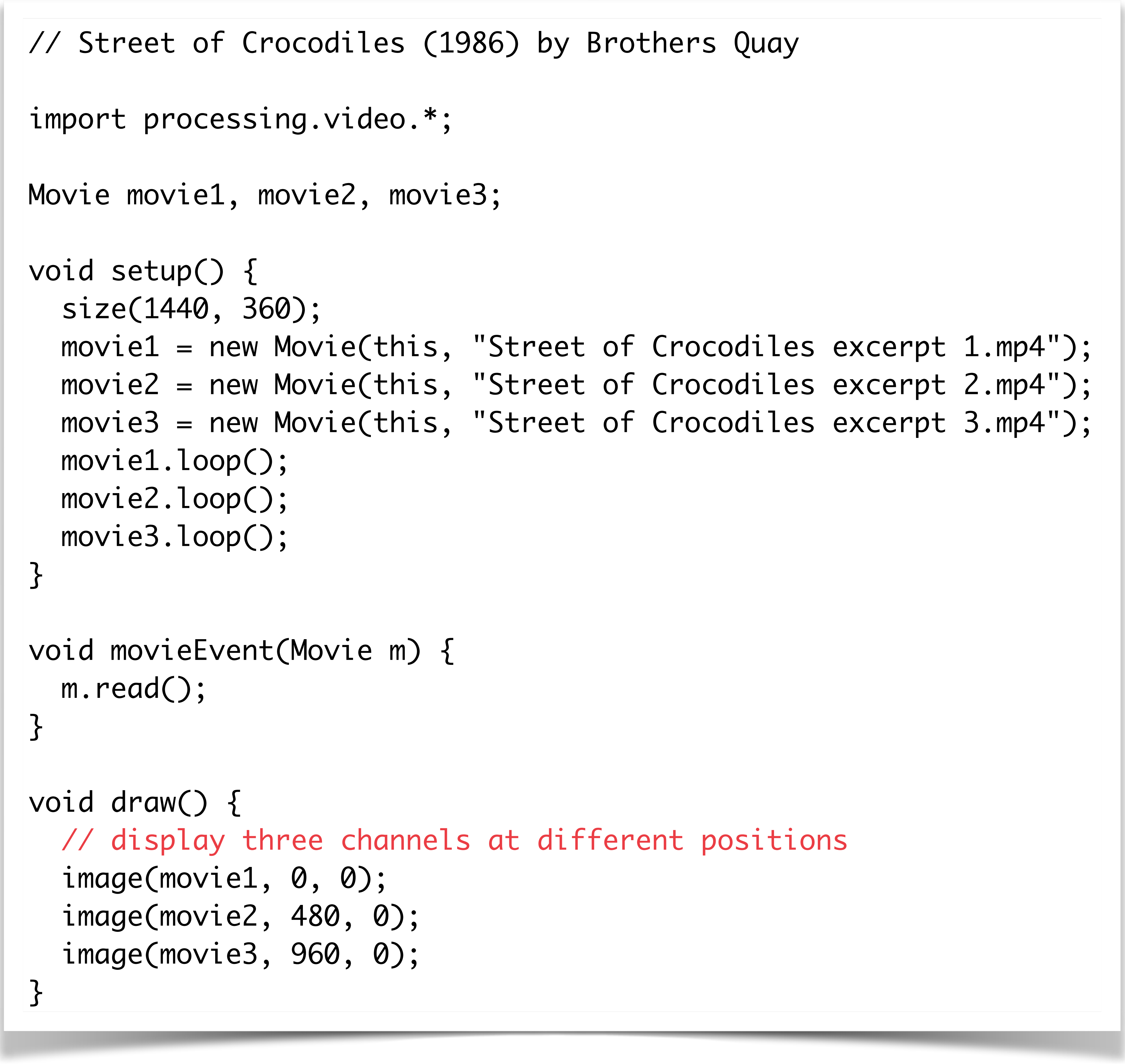
Example/Libraries/Video/Movie/Scratch



```
if (mov.available()) {
  mov.read();
  float f = map(mouseX, 0, width, 0, 1);
  float t = mov.duration() * f;
  mov.play();
  mov.jump(t);
  mov.pause();
}
```

EXAMPLE 3

Only one movieEvent is needed →



```
// Street of Crocodiles (1986) by Brothers Quay

import processing.video.*;

Movie movie1, movie2, movie3;

void setup() {
  size(1440, 360);
  movie1 = new Movie(this, "Street of Crocodiles excerpt 1.mp4");
  movie2 = new Movie(this, "Street of Crocodiles excerpt 2.mp4");
  movie3 = new Movie(this, "Street of Crocodiles excerpt 3.mp4");
  movie1.loop();
  movie2.loop();
  movie3.loop();
}

void movieEvent(Movie m) {
  m.read();
}

void draw() {
  // display three channels at different positions
  image(movie1, 0, 0);
  image(movie2, 480, 0);
  image(movie3, 960, 0);
}
```

Movie.speed() Method

The Movie.speed() method allows to change the playback speed, for example:

```
      //plays at half the speed (slow motion)
      myMovie.speed(0.5);

      //plays the movie twice as fast (fast forward)
      myMovie.speed(2);
```

Backwards playback is no longer supported

~~Use negative values for backwards playback, for example:~~

~~//reverse in normal speed~~

~~myMovie.speed(-1);~~

~~NOT all video formats support backwards playback. The theora codec (.mkv) does support backward playback, but not so the H264 codec (.mp4).~~

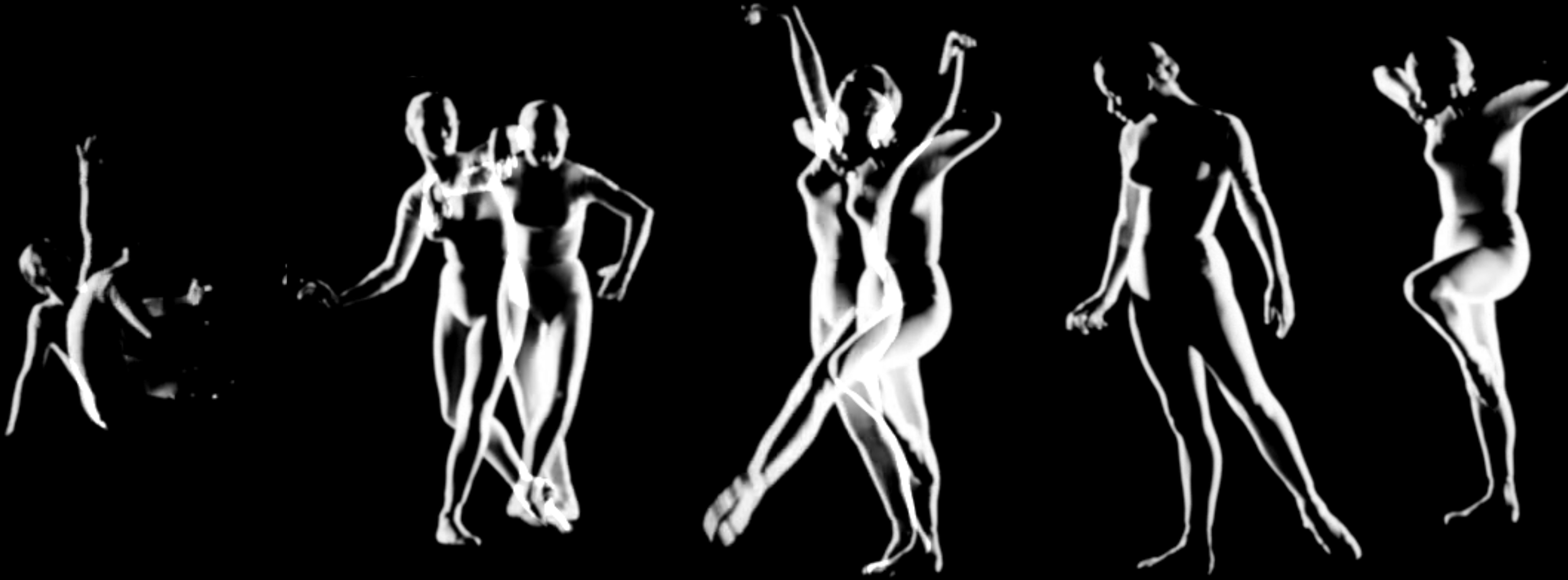
blend()

- Blends a region of pixels from one image into another (or in itself again) with full alpha channel support
- blend(sx, sy, sw, sh, dx, dy, dw, dh, mode)

  blends a region of pixels from the display window to another area of the display window
- blend(src, sx, sy, sw, sh, dx, dy, dw, dh, mode)

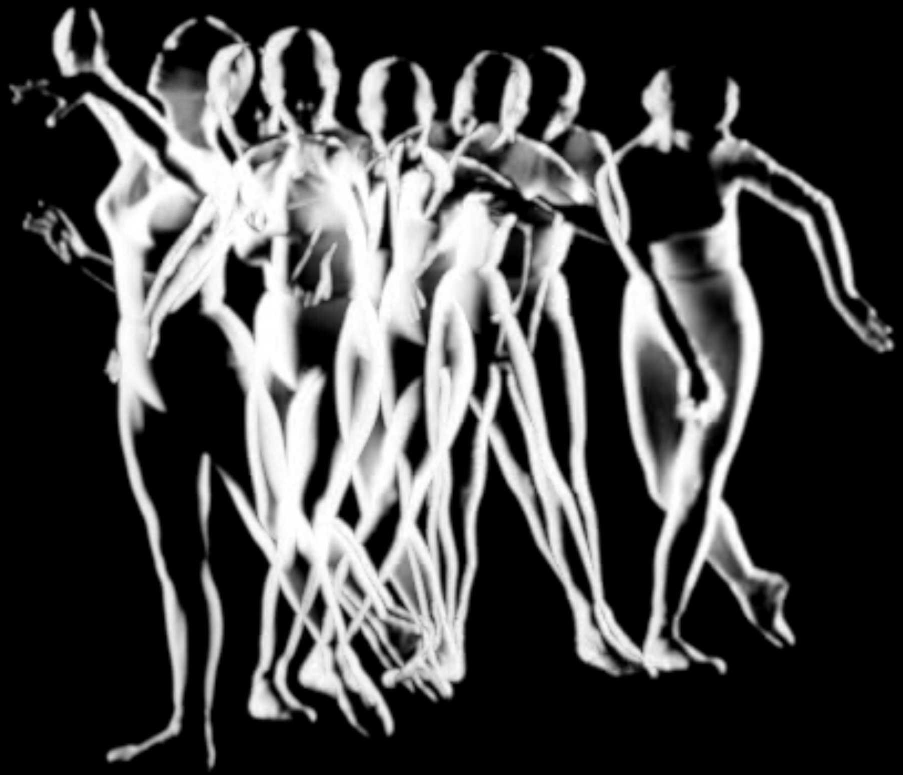
  blends a region of pixels from an image into the display window
- if the source and destination regions aren't the same size, it will automatically resize the source pixels to fit the specified target region
- Mode: BLEND, ADD, SUBTRACT, DARKEST, LIGHTEST, DIFFERENCE, EXCLUSION, MULTIPLY, SCREEN, OVERLAY, HARD_LIGHT, SOFT_LIGHT, DODGE & BURN
- imageMode() is ignored
- blend() is also a method for PImage

EXAMPLE 4 Additive Blending

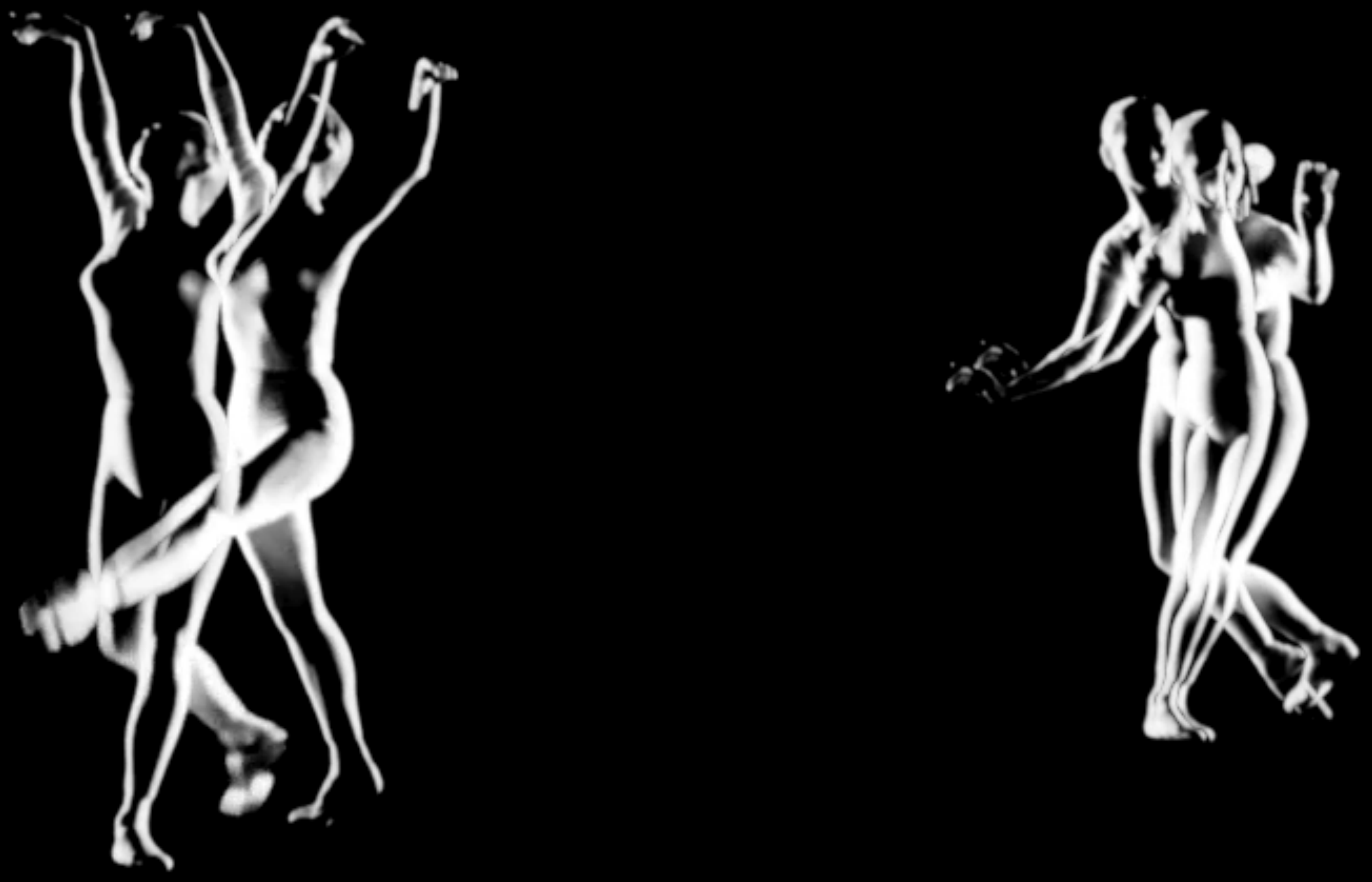


EXERCISE 2

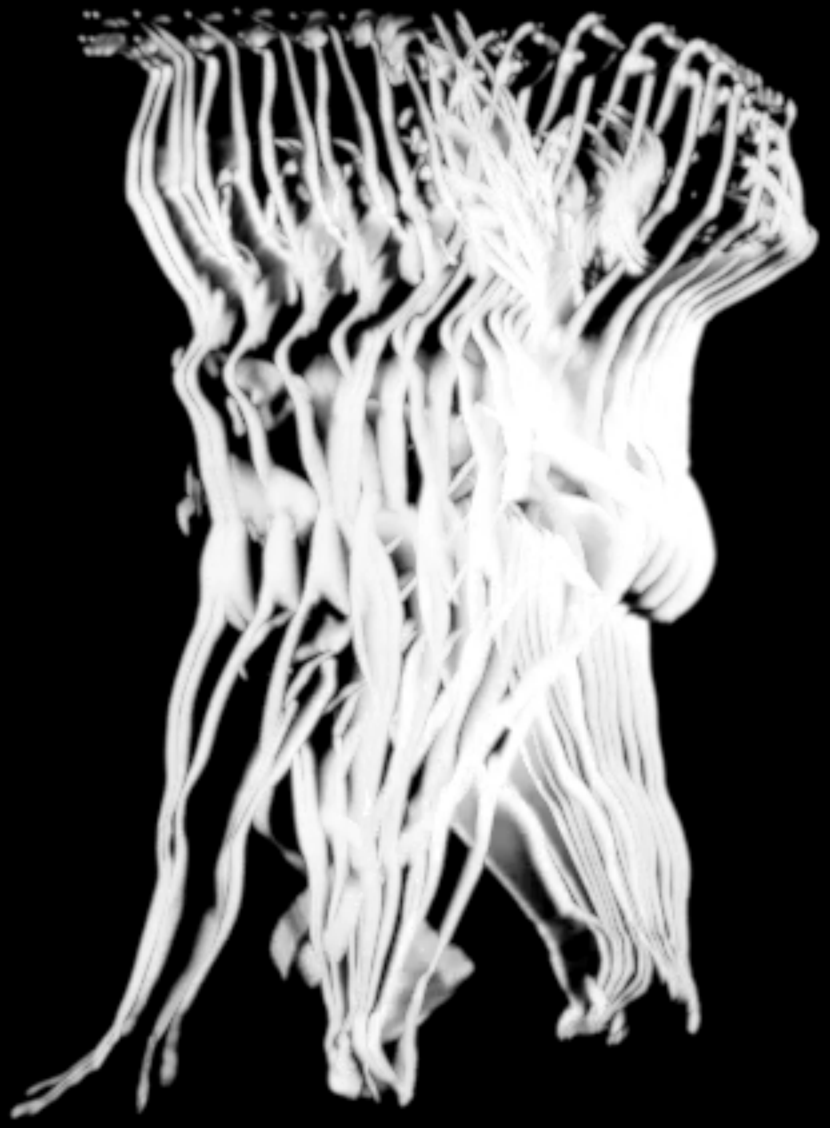
- Based on Example 4, add one key frame using the mouse left-click while the video is playing
- display and overlap the whole stack of key frames
- right-click to clear all key frames
- Hints: you may either use createImage() to create a empty image and blend the key frames to it (use blend as method), or create an array of PImage to store all key frames



EXAMPLE 5 Pas de deux

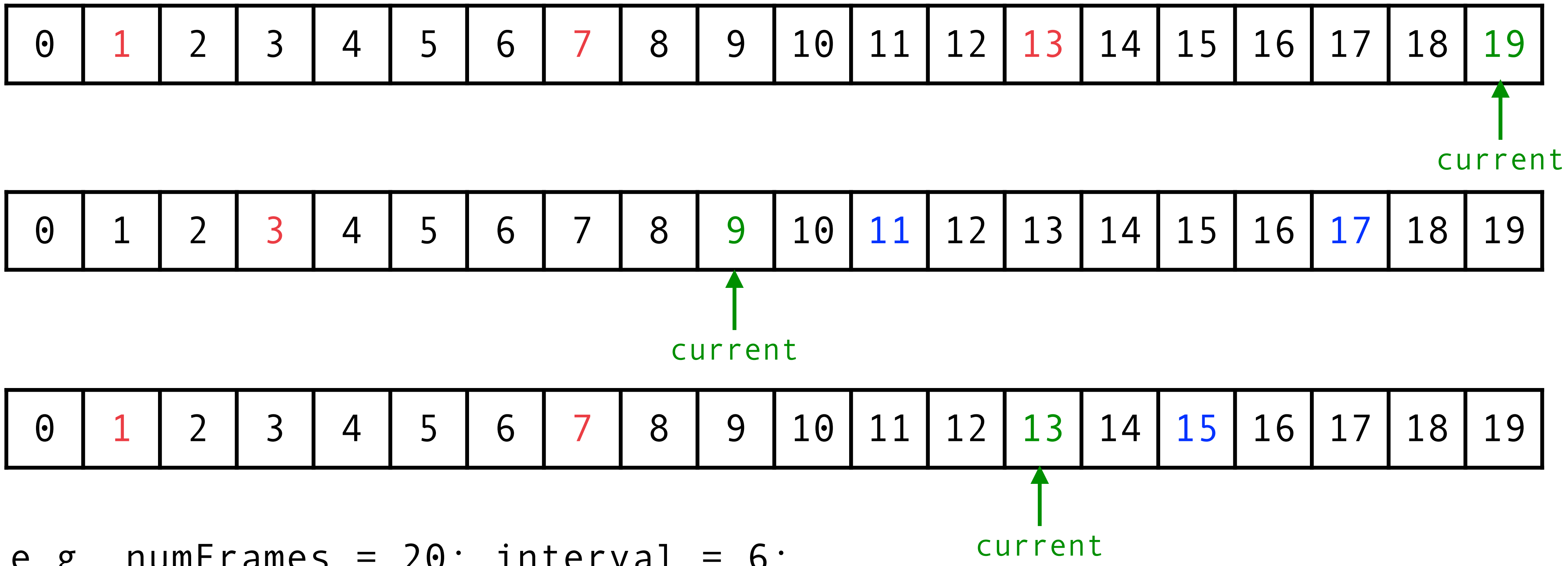


EXAMPLE 6



EXAMPLE 6

```
void draw() {
  //image(movie, 0, 0);
  image(bufferFrames[current], 0, 0);
  // buffer frames before current
  for (int i=current-interval; i>=0; i-=interval)
    blend(bufferFrames[i], 0, 0, width, height, 0, 0, width, height, LIGHTEST);
  // buffer frames after current
  for (int i=numFrames-(interval-current%interval); i>current; i-=interval)
    blend(bufferFrames[i], 0, 0, width, height, 0, 0, width, height, LIGHTEST);
  //println(frameRate);
}
```



0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

e.g. numFrames = 20; interval = 6;

CAPTURE class

Constructors

```
Capture(parent)
Capture(parent, device)
Capture(parent, width, height)
Capture(parent, width, height, fps)
Capture(parent, width, height, device)
Capture(parent, width, height, device, fps)
```

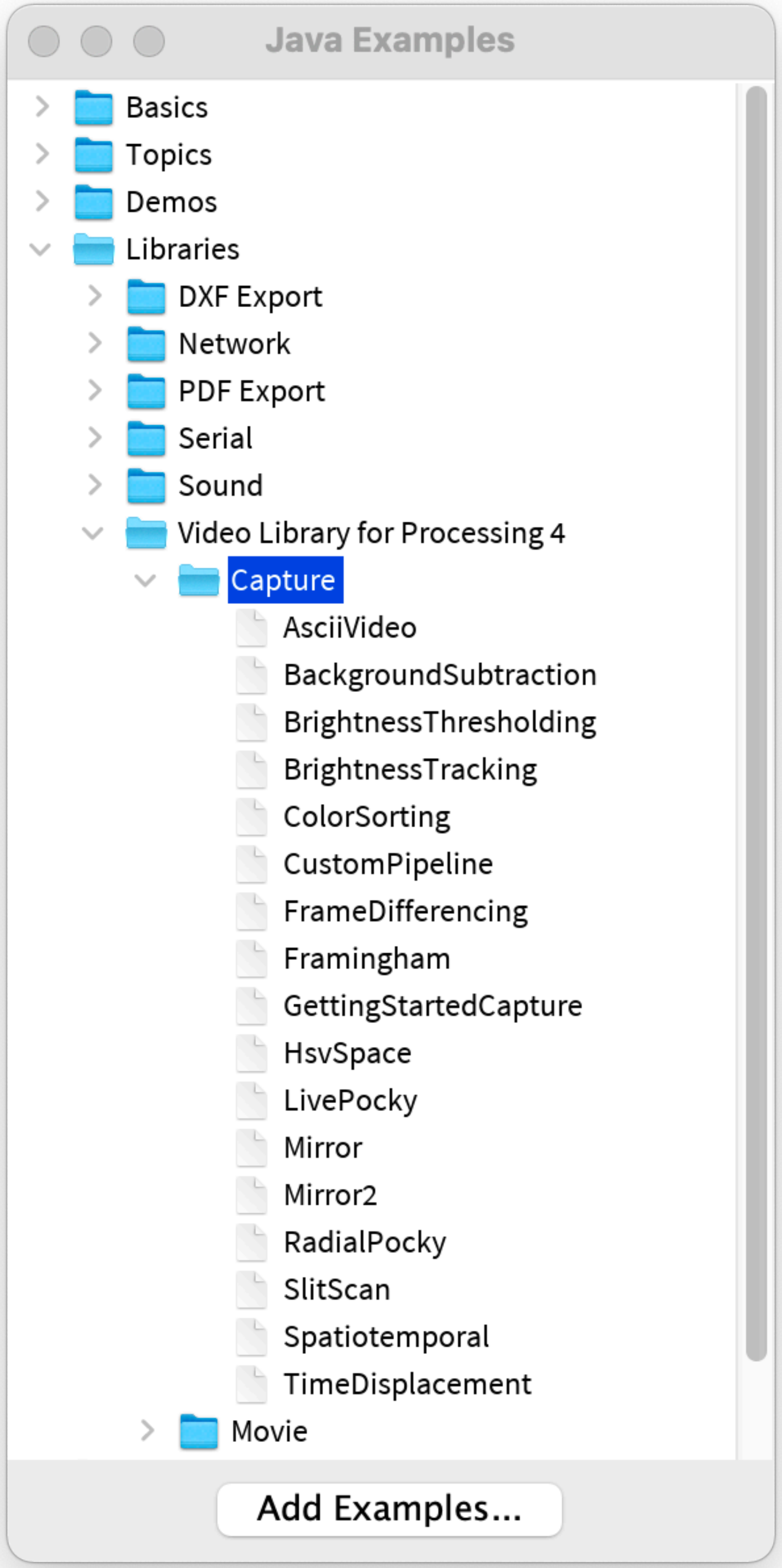
Parameters

`parent`	PApplet, typically ”this”
`device`	device name
`width`	width in pixels
`height`	height in pixels
`fps`	frames per second

Methods

`frameRate()`	Sets how often frames are read from the capture device.
`available()`	Returns ”true” when a new frame from the device is available to read.
`start()`	Starts capturing frames from an attached device.
`stop()`	Stops capturing frames from an attached device.
`read()`	Reads the current frame of the device.

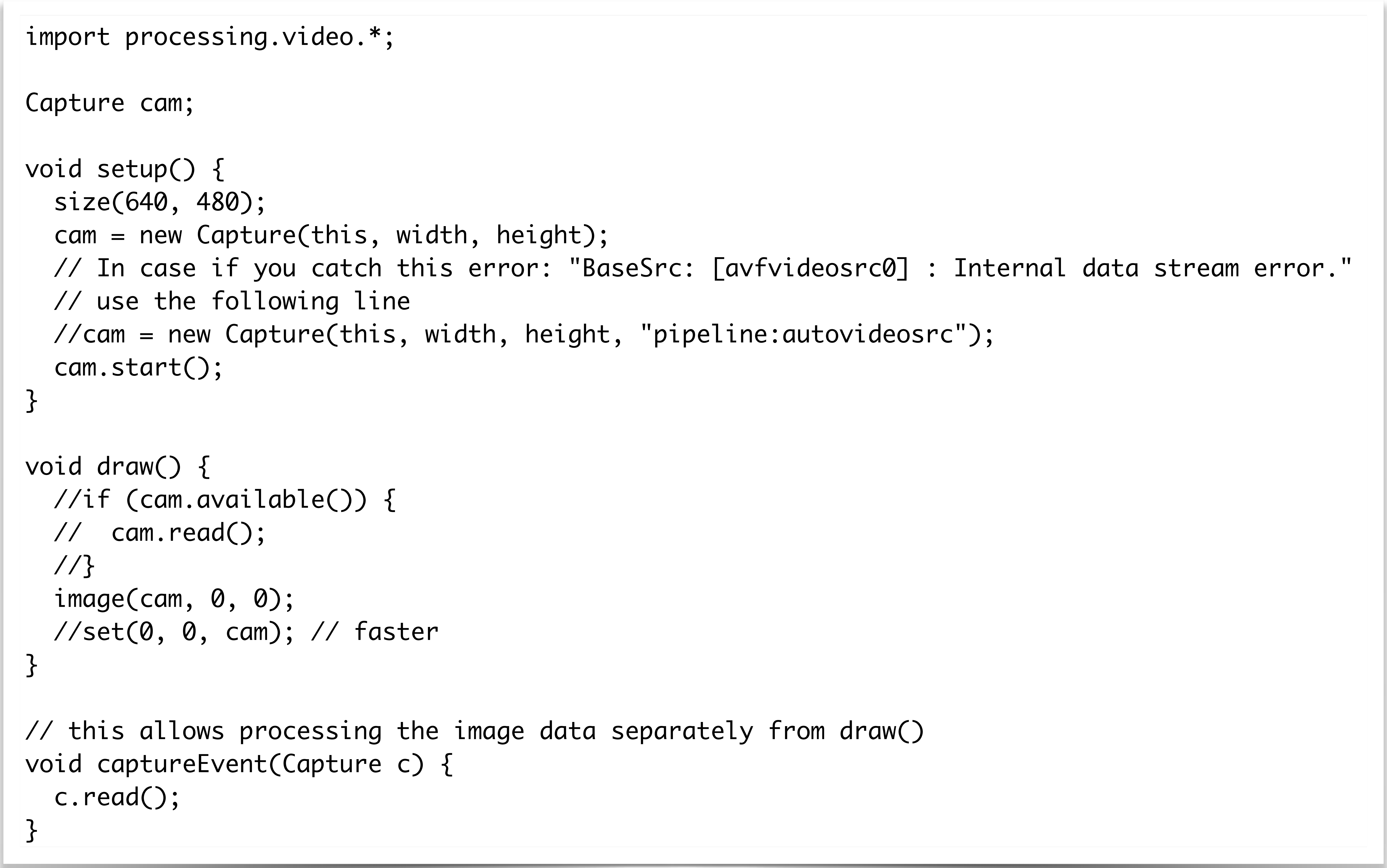
Official examples for CAPTURE class



Java Examples

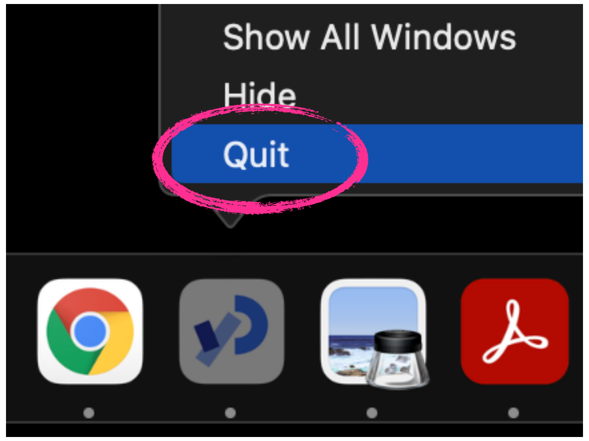
- Basics
- Topics
- Demos
- Libraries
  - DXF Export
  - Network
  - PDF Export
  - Serial
  - Sound
  - Video Library for Processing 4
    - Capture
      - AsciiVideo
      - BackgroundSubtraction
      - BrightnessThresholding
      - BrightnessTracking
      - ColorSorting
      - CustomPipeline
      - FrameDifferencing
      - Framingham
      - GettingStartedCapture
      - HsvSpace
      - LivePocky
      - Mirror
      - Mirror2
      - RadialPocky
      - SlitScan
      - Spatiotemporal
      - TimeDisplacement
    - Movie

EXAMPLE 7 SIMPLE CAPTURE

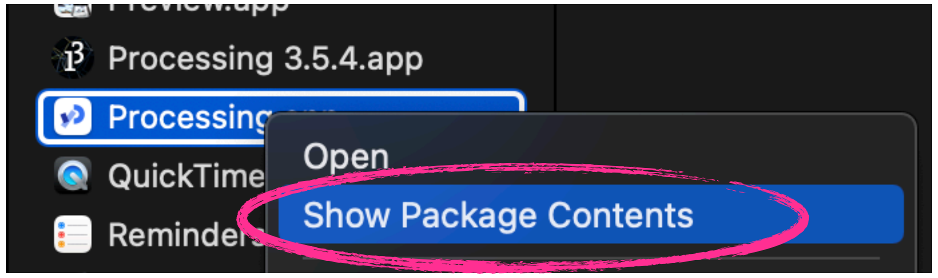


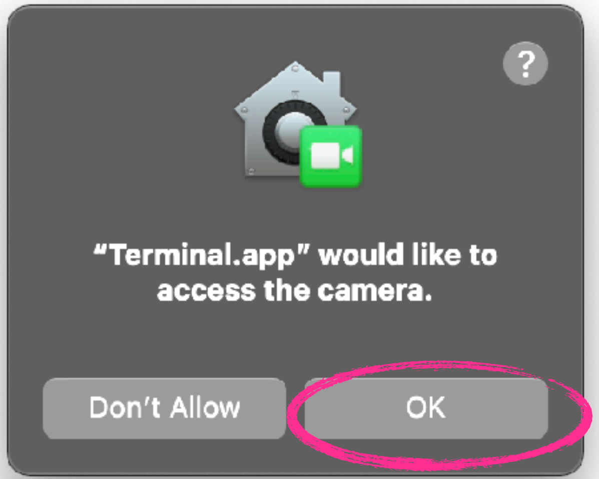
```
import processing.video.*;

Capture cam;

void setup() {
  size(640, 480);
  cam = new Capture(this, width, height);
  // In case if you catch this error: "BaseSrc: [avfvideosrc0] : Internal data stream error."
  // use the following line
  //cam = new Capture(this, width, height, "pipeline:autovideosrc");
  cam.start();
}

void draw() {
  //if (cam.available()) {
  //  cam.read();
  //}
  image(cam, 0, 0);
  //set(0, 0, cam); // faster
}

// this allows processing the image data separately from draw()
void captureEvent(Capture c) {
  c.read();
}
```

PROBLEM ON CAMERA ACCESS ON MACOS



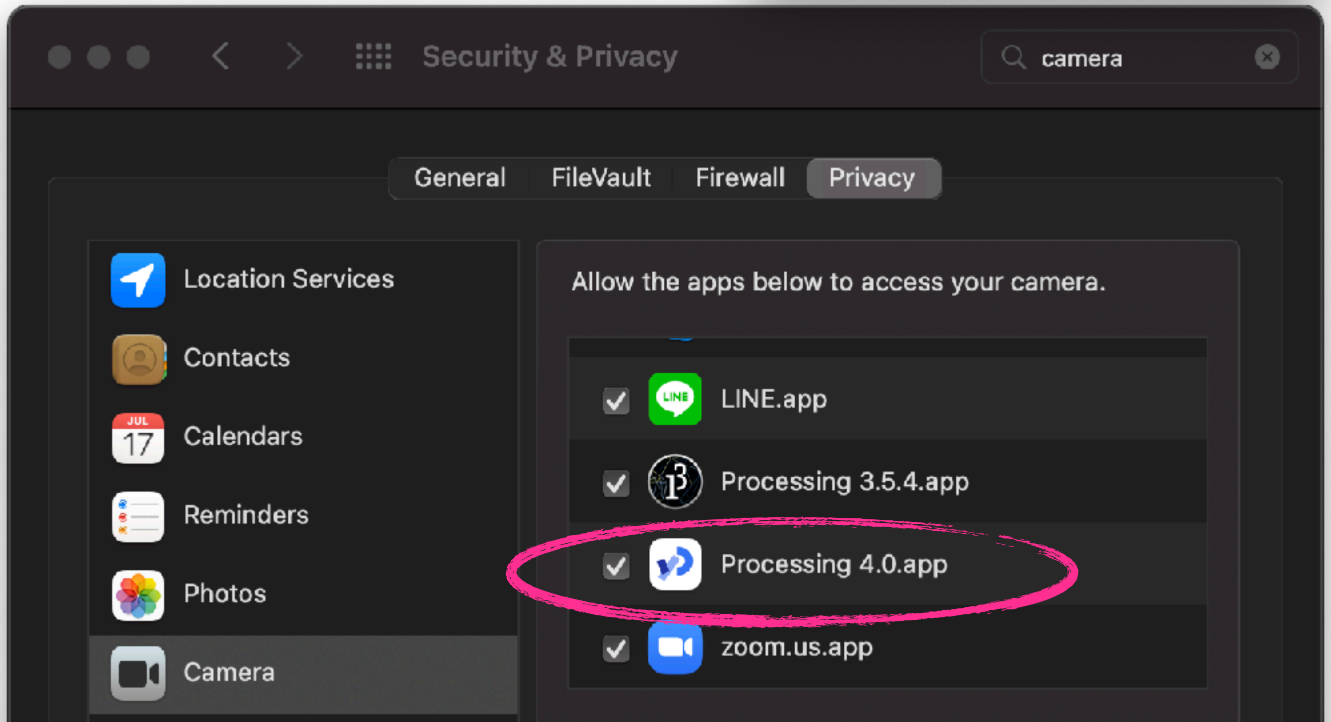
- Completely quit Processing
- Go to Processing.app in Finder, choose “Show Package Contents. Then, go to Contents/MacOS/, and double click Processing. This will open the executable in Terminals.





“Terminal.app” would like to access the camera.

- Run Example 7. The computer will then ask for camera access. Click “OK”.
- Next time you don’t have to start Processing from Terminal.



Allow the apps below to access your camera.

☑ LINE.app
☑ Processing 3.5.4.app
☑ Processing 4.0.app
☑ zoom.us.app

EXERCISE 3

- Based on wk3_example_07 Preset Filter, try to apply the filter to live video captured by the camera (webcam)