Multimedia-Programmierung
Übung 5

Ludwig-Maximilians-Universität München

Sommersemester 2011

Today

- SVG
  - Text
  - Drawing
  - Animations
  - Interactions

- Processing
  - Images
  - Drawing
  - Interaction

SVG – What’s that?

- Scalable Vector Graphics
- Developed by World Wide Web Consortium (W3)
- XML based, so only text editor is needed
- Supported by Firefox & Chrome, IE needs plugin
- Files: *.svg

Get it running

- what you have to do: nothing ☺
- create *.svg-file in any text editor
- drag&drop the file into your browser (Firefox)
- W3C Scalable Vector Graphics (SVG) 1.1 Specification

  http://www.w3.org/TR/2003/REC-SVG11-20030114/

Document Structure

- XML-based

```
<?xml version=„1.0“ encoding=„UTF-8“?>
<!DOCTYPE svg PUBLIC „-//W3C//DTD SVG 1.1//EN“
            „http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd“>

<svg xmlns="http://www.w3.org/2000/svg" version="1.1">

<title>Example</title>

      Contents of the File

</svg>
```

View Text

- <text>…</text>
- specify x- and y-coordinates
- http://www.w3.org/TR/2003/REC-SVG11-20030114/text.html#TextElement

```
<svg xmlns="http://www.w3.org/2000/svg" version="1.0">

…

<text x="100" y="100">
      Klick mich und ich drehe mich!
</text>

</svg>
```

Geometry

- <line /> <rect /> <polygon /> …
- http://www.w3.org/TR/2003/REC-SVG11-20030114/shapes.html

```
<svg xmlns="http://www.w3.org/2000/svg" version="1.0">
…

      <rect x="100" y="100" width="300" height="200">
      </rect>

</svg>
```

Colors, Strokes, Fills

- <line /> <rect /> <polygon /> …

```
<svg xmlns="http://www.w3.org/2000/svg" version="1.0">
…

    <rect x="100" y="100" width="300" height="200"
        fill="grey" stroke="green" stroke-width="15">
    </rect>

</svg>
```

Animations

- <animate />
- Choose an attribute to be changed
- Choose begin and duration
- fill: describes the end of the animation

- http://www.w3.org/TR/2003/REC-SVG11-20030114/animate.html

```
<svg xmlns="http://www.w3.org/2000/svg" version="1.0">
…
     <rect x="100" y="100" width="300" height="200“ fill="grey">
          <animate
               attributeType="XML" attributeName="x"
               begin="2s" dur="3s"
               from="100" to="300"
               fill="freeze" />    </rect>
</svg>
```

Animations & Paths

- <animateMotion />
- Object can be moved along paths (M: start, L: line, Z: close)
- Change speed with dur
- Change number of iterations with repeatCount

- http://www.w3.org/TR/2003/REC-SVG11-20030114/paths.html

```
<svg xmlns="http://www.w3.org/2000/svg" version="1.0">
…
    <rect x="100" y="100" width="300" height="200" fill="grey">
        <animateMotion
            dur="1s"
            path="M 100,100 L 100,300 L 100,100 Z"
            fill="freeze"
            repeatCount="indefinite" />
    </rect>
</svg>
```

Processing – What’s that??

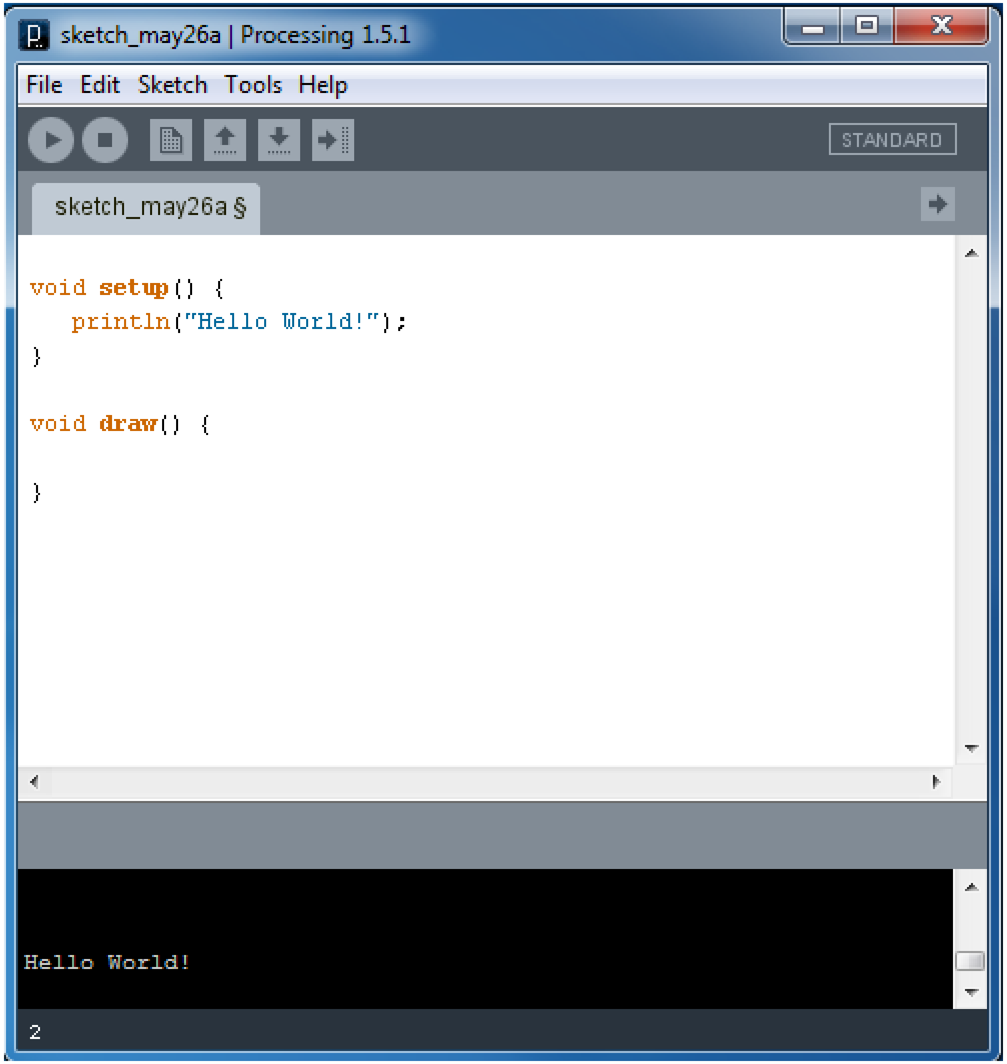
“Processing is an open source programming language and environment for people who want to create images, animations, and interactions.” (processing.org)

- teach fundamentals of visual programming
- students, artists, designers, researchers, and hobbyists
- learning, prototyping and production
- amazing examples: http://processing.org/exhibition/
- siles: *.pde

Get it running

- Download: http://processing.org/download/
- Unpack
- Start ☺
- No “installation” needed

User Interface



```
void setup() {
  println("Hello World!");
}

void draw() {

}
```

```
Hello World!
```

Code is based on Java!

Setup() & Draw()

- important concept!
- setup() - function: statements executed once when program starts
- draw() – function: statements executed over and over until program stopped (after last line, first line is executed again)

→ Loop

```
void setup() {
    size(200, 200);
    background(0);
    noStroke();
    fill(102);
}

int a = 20;

void draw() {
    rect(20, 20, a++, a++);
}
```

You can Stop the Loop

- noLoop() in setup()

- Code in draw() will only execute once

```
void setup() {
    size(200, 200);
    background(0);
    noStroke();
    fill(102);
    noLoop();
}

int a = 20;

void draw() {
    rect(20, 20, a++, a++);
}
```

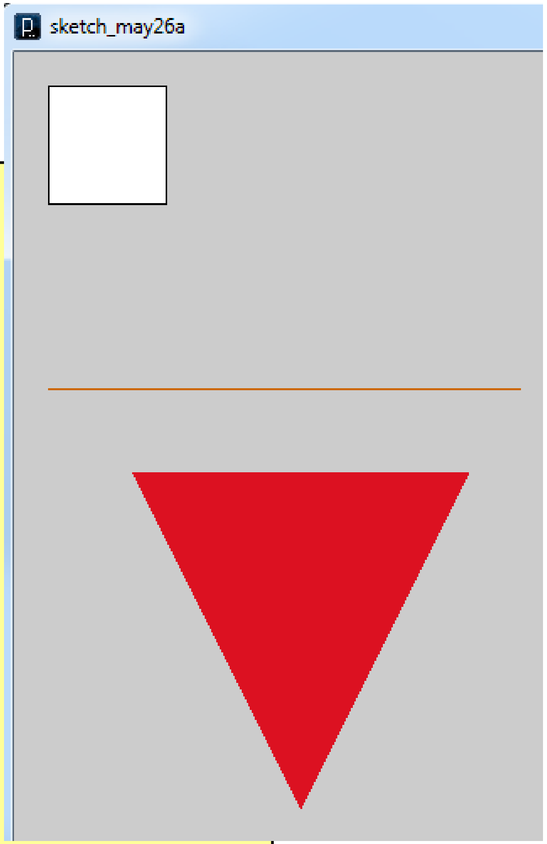
Drawing is really simple

http://processing.org/reference/

- rect()
- line()
- triangle()
- …

- fill(r, g, b)
- noStroke()
- stroke(r, g, b)
- …

```
size(500, 500);

rect(20, 20, 70, 70);

stroke(204, 102, 0);

line(20, 200, 300, 200);

noStroke();

fill(220, 17, 33);

triangle(70, 250, 270, 250, 170, 450);
```



Display Images

- Save your current sketch
- Image (jpg) needs to be in “data” folder inside the sketch folder
- Use “Sketch” → “Add File”

```
size(700, 700);

PImage img;

img = loadImage(“xyz.jpg");

image(img, 0, 0);
```

Mouse Interactions

- mouseClicked()
- mouseMoved()
- mouseDragged()
- …

- mouseX
- mouseY
- …

```
void setup() {
	size(400, 400);
	fill(230, 15, 44);
}


void draw() {
}


void mouseClicked() {
	rect(mouseX, mouseY, 50, 50);
}
```

Useful Links

- W3C Scalable Vector Graphics (SVG) 1.1 Specification
  http://www.w3.org/TR/2003/REC-SVG11-20030114/

- Processing Reference
  http://processing.org/reference/

- Processing Tutorials
  http://processing.org/learning/