

# Processing.org workshops

## Workshop 3

Open Space Aarhus

Bryggervej 30, 8240 Århus N

29. november 2011



# Dagens program

- Introduktion
- Resume af sidste gang
- Kode
  - Løkker
  - Arrays
  - Funktioner
- Workshop
  - Mange bolde på en gang!
- *Afslutning*

## Hvad har de flittige lavet

- `http://www.openprocessing.org/classrooms/?classroomID=1075`

## Slides og processing filer

<http://poodle/processing>

Slides kan sikkert bruges til at kigge i eller kopiere fra.

- Variable: `float boldX = 200;`
- Operationer: `boldX = boldX + deltaX;`
- Løkker:
  - `while (betingelse) { ... }`
  - `for (start; betingelse; opdatering;) { ... }`
- Forgreninger:
  - `if (x > width) { dx = -dx; }`
  - `if (betingelse) { ... }`
  - `if (betingelse) { ... } else { ... }`
- Simpel fysik.
  - `acceleration = summen af kræfter delt med partiklens masse`
  - `ny hastighed = gammel hastighed + acceleration`
  - `ny position = gammel position + hastighed`

# Løkker

Lav en løkke som tegner 10 bolde ved siden af hinanden. Brug x som tæller

med while-løkken

```
int x = 0;
while (x < 10) {
    ellipse(20 + x * 40, 200, 30, 30);
    x += 1;
}
```

med for-løkken

```
for (int x = 0; x < 10; x += 1) {
    ellipse(20 + x * 40, 200, 30, 30);
}
```

# Opgave: Et gitter af figurer

Lav to løkker indeni hinanden, så rækken bliver gentaget under hinanden. Brug x og y som tæller.

## Løkker i løkker

```
for (int y = 20; y < height; y += 40) {  
  for (int x = 20; x < width; x += 40) {  
    ellipse(x, y, 30, 30);  
  }  
}
```

Prøv at bruge x og y eller random() til at styre farve eller størrelse. Brug evt forgreninger til at tegne forskellige figurer.

# Arrays

en opslagstabel

- deklaration: `float[] boldX = new float[10];`
- tildeling: `boldX[0] = 100; boldX[1] = 120;`
- læsning: `ellipse(boldX[0], boldX[0], 30, 30);`



# Arrays til mange bolde

- `float[] boldX = new float[10];`
- `float[] boldY = new float[10];`
- `float[] deltaX = new float[10];`
- `float[] deltaY = new float[10];`

# Arrays og Løkker

A match made in heaven

# Funktioner

Små genbrugelige stumper kode. Også nyttig til at gøre koden mere overskuelig. Du har allerede brugt en masse funktioner fra processing. Du har også skrevet dine egne fx. `setup()` og `draw()`. Nu vil vi lave vore egne

# Funktioner

funktioner til dagens opgave...?

100