

Programmering med P5.js

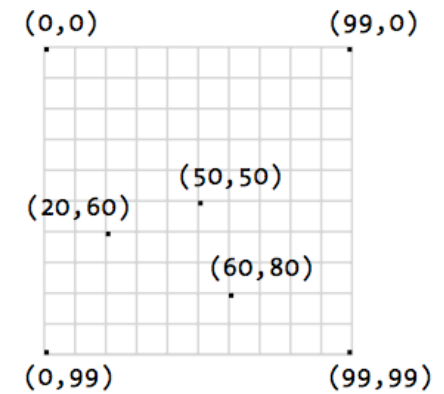
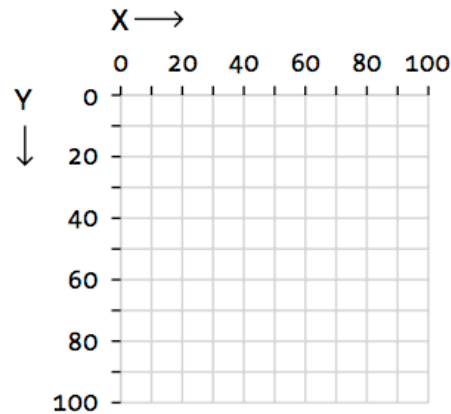
```
function setup() {  
}
```

```
function draw() {  
}
```

Canvas

```
function setup() {  
  createCanvas(____,____);  
}
```

```
function draw() {  
}
```



Console

```
function setup() {  
  console.log(_____);  
  createCanvas(____,____);  
}
```

```
function draw() {  
}
```

Preload-function

```
Var awesomePic;  
function preload() {  
    awesomePic = loadImage("images/awesome.png");  
  
}
```

→ vil blive loaded før resten af koden

Godt til f.eks. Billeder, lyd, tekst, gif osv.

Shapes!

```
line(x1,y1,x2,y2);  
ellipse(x,y,w,h);  
rect(x,y,w,h);  
m.fl.
```

Kig i reference 😊 : <https://p5js.org/reference/>

Kig f.eks. På: triangle(), ellipseMode(), strokeWeight()...

Colours!

Greyscale: et tal fra 0-255

RGB= rød, grøn, blå

P5.js:

```
background(__,[__],[__]);
```

```
fill(__,[__],[__]);
```

```
noFill();
```

```
stroke(__,[__],[__]);
```

```
noStroke();
```

Kig i reference 😊 : <https://p5js.org/reference/>

Kig f.eks. På: `colorMode()`

Alpha

`fill(__,__);` → gråskala værdi + opacitet/gennemsigtighed

Eller

`fill(__, __, __, __);` → RGB farver + opacitet/gennemsigtighed

Alpha 0 → 100% transparent

Alpha 255 → 0% transparent, dvs. udfyldt

Gælder også for `stroke();`

Varying variables

```
var x = 0;
```

```
function setup(){...}
```

```
function draw() {  
  x = x+1  
  Eller  
  x+=1  
}
```

- Definer variabler, så du kun behøver at ændre et sted i programmet for at ændre værdier
- Gør det øverst i koden
- Variabler kan ændres senere, og i løbet af programmet

Make your own functions

```
function setup() {...}
```

```
function draw(){
```

```
    myAwesomeFunction(5,5);
```

```
}
```

```
function myAwesomeFunction(height,width) {
```

```
    ...
```

```
}
```

- Opdel dit program i funktioner!
- Giver overskuelighed både for dig, og den der skal læse din kode
- Hjælper til at undgå at skrive det samme flere gange

Comment your code!

```
function draw() {  
    //The ellipse that follows the cursor  
    ellipse(mouseX,mouseY,20,20);  
    ...  
}
```

- Overskuelighed
- Læselighed
- Både for andre og for dig selv, hvis du vender tilbage til et gammelt program du har lavet

Winnie's sample code

[https://rawgit.com/AUAP/AP2018/master/class02/sket
ch02/index.html](https://rawgit.com/AUAP/AP2018/master/class02/sket
ch02/index.html)

```
6 var img;
7 var throbber;
8 var siz = 25;
```

```
10 function preload() {
11   img = loadImage("images/Theartgalleryofthefuture.gif");
12   throbber=createImg("images/throbber.gif");
13 }
```

Preload function

```
15 function setup() {
16   createCanvas(489, 500);
17   noCursor();
18   throbber.size(siz,siz);
19   console.log("hello " + "world");
20 }
```

Ingen synlig mus

Giv throbberen en størrelse



```
22 function draw() {
23   throbber.position(mouseX,mouseY); //locate the gif
24   image(img, 0,0);
25   noStroke();
26   fill(188, 185,169);
27   beginShape(); //left top
28   vertex(1,80);
29   vertex(53,75);
30   vertex(54,121);
31   vertex(1,124);
32   endShape(CLOSE);
```

Positioner throbberen

Positioner billedet

Udfyld billederne

```
79 function mousePressed() {
80   console.log(mouseX, mouseY);
81 }
```

mousePressed funktion

- Er allerede defineret i P5.js biblioteket
- Alt heri sker når der trykkes på musen

MiniEx 2

Winnie skriver:

- Make sure you have read Goriunova's text: Goriunova, Oga. **Fun and Software**: Exploring Pleasure, Paradox and Pain in Computing. Bloomsbury Academic, 2014, pp.1-19.(see weekly ref list)
- Design a program that expresses and addresses Goriunova's notion of **fun by using basic shapes** primarily. “(it is ok to be simple, and it is ok to draw something static. Perhaps you can try to implement from scratch by thinking about what you want and what you want to show/tell)”
- Upload your program/result to your own Github account under a folder called **mini_ex2**. (Make sure your program can be run on a web browser)
- Create a **readme** file (README.md) and upload to the same mini_ex2 directory

Dvs.

- Læs jeres lektier...
- Overvej hvad Goriunova mener med fun/sjov inden for software og kode
 - Prøv at udtrykke dette gennem din egen kode.
 - Prøv først at forestille dig hvad du vil lave, og så udforsk hvordan du kan få det til at ske med din kode, i stedet for lave dit program ud fra det syntax du allerede kender
- Upload til Github
- Lav ReadMe

ReadMe

Winnie skriver:

- A screenshot of your program
- A URL link to your program and run on a browser, see: <https://rawgit.com/>
- Describe your program and what you have used and learnt
- **What is fun?** To what extent your program addresses Goriunova's notion of fun? Is it fun in relation to geekiness, or humor, or paradox, or other modes of thoughts? (This question requires your articulation of your work beyond just describing how it works technically)

miniEx — 3,2,1,CODE!

Husk peer-feedback til 2 andre