

A

T H R O B B E R

AGENDA

I. Functions

II. MINI_EX3

III. Winnie's kode

IV. Feedback i grupper

V. Coding time

CONDITIONAL STATEMENTS

KAN BRUGES HVIS...

- Min baggrund skal blive hvid, når jeg...
- Hvis x er sandt = udfør koden
- Hvis x ikke er sandt = udfør ikke koden

```
if (boolean expression er sand) {  
    så udfør dette;  
}
```

```
if (mouseX > 200) {  
    background(0);  
}
```

DER KAN BYGGES OVENPÅ

```
if (mouseX > 200) {  
    background(0);  
}  
else {  
    background(255);  
}
```

ELLER

```
if (mouseX > 1000) {  
    rect(200, 200, 80, 80);  
} else if (mouseX > 700) {  
    ellipse(200, 200, 80,  
80);  
} else if (mouseX > 300) {  
    line(200, 200, 200,  
600);  
  
}  
  
}
```


RELATIONAL OPERATORS

| | |
|--------|-------------------------|
| $>$ | større end |
| $<$ | mindre end |
| \geq | større end eller lig |
| \leq | mindre end eller lig |
| $==$ | lig med |

||

eller

33

og

LOOPS

- Loops kan udføre en blok kode om og om igen
- Brugbart, hvis du vil udføre en blok kode igen og igen, men med ny værdi hver gang

LOOPS

| | Udføres... | |
|--------------------------------|--------------------------|----------------------|
| <code>if (ikke et loop)</code> | hvis betingelsen er sand | udføres én gang |
| <code>while</code> | mens betingelsen er sand | udføres igen og igen |
| <code>for</code> | mens betingelsen er sand | udføres igen og igen |

LOOPS

Men er for loops ikke bare det samme som while loops så?

det her er det
samme som

det her

```
function draw() {  
    background(0);  
    strokeWeight(4);  
    stroke(255);  
  
    var x = 0;  
  
    while(x < width) {  
        ellipse(x, 200, 25, 25);  
        x += 50;  
    }  
  
    for (var x = 0; x < width; x += 50) {  
        ellipse(x, 100, 25, 25);  
    }  
}
```

LAD OS ZOOME IND

```
for (var x = 0; x < width; x += 50)
```

1. Initialisation

2. Boolean test

3. Incrementation
operation

FUNCTIONS

- Blok af kode der skal udføre en bestemt opgave
- Defineret ved 'function' keyworded + et brugerdefineret navn

FUNCTIONS

De er begge funktioner, men hvad er forskellen?

```
rect(200, 100, 50, 50);
```

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

FUNCTIONS

`rect(200, 100, 50, 50);` `<-` calling

`function setup() {` `<-` defining

`}`

ALTSÅ...

`rect()` er allerede **DEFINERET** i `p5.js` library,
men skal **KALDES** af brugeren

`function setup()` skal **DEFINERES** af brugeren,
men `p5` ved, hvornår den skal **KALDES**

men man kan også lave sine egne funktioner...


SYNTAX

```
function navn (evt. parametrer) {  
  
}
```

```
function bil (x, y, hjul) {  
  
}
```

SYNTAX

hvor får de[parametrene] deres værdi?



```
function bil (x, y, hjul) {  
  
}
```

Fra hvor du kalder funktionen:

```
    bil (50, 40, 100); //argumenter
```

```
function bil (x, y, d) { //parametere  
  
}
```

```
function draw () {  
    background(0);  
    bil(200, 100, 40); //2.initialise  
}
```

```
function bil (x, y, diameter) { //1.declare  
    rect (x, y, diameter+20);  
    ellipse(x-10, y-10, diameter) //3.use  
    ellipse(x+10, y+10, diameter)  
}
```


MINI_EX3

Study the syntaxes that have been discussed in class, and then **redesign** an **animated throbbber**.

[https://github.com/nynnelucca/Mini-exercises/tree/gh-pages/mini_ex3]

https://nynnelucca.github.io/Mini-exercises/mini_ex3/empty-example/

[https://github.com/kris03/AP-17/tree/master/mini_ex3]

https://cdn.rawgit.com/kris03/AP-17/167e444d/mini_ex3/index.html

[https://github.com/a9neh/Anines-mini_ex/tree/gh-pages/miniex3]

https://rawgit.com/a9neh/Anines-mini_ex/gh-pages/miniex3/index.html

WINNIE'S KODE

```
function setup() {  
  createCanvas(windowWidth, windowHeight);  
  background(10);  
  frameRate (10); //rotationshastighed  
}
```

```
function draw() {  
  noStroke();  
  fill(10,80); //alpha værdi -> gråskala værdi + opacitet  
  rect(0, 0, width, height); //denne bliver loaded 'ovenpå' hele tiden  
  drawThrobber(9);  
  /*  
    En funktion kaldes, og den gives et argument; dette tal  
    kan ændres, og hvis det ændres til det større kommer cirklerne  
    til at ligge tættere på hinanden.  
  */  
}
```

```
function drawThrobber(num) { //den nye funktion med ét parameter, som altså er 9

    push();

    translate(width/2, height/2);

    // 360/num >> degree of each ellipse' move ;frameCount%num >> get the remainder that indicates the movement
of the ellipse

    var cir = 360/num*(frameCount%num);
    /*
    1. Lokal variabel, der gives en værdi med det samme.
    2. frameCount = antal frames der har været siden programmet blev loaded
    3. % = en modulo operator. Med den spørger vi: kan frameCount divideres med num.
    */

    rotate(radians(cir)); //radians konverterer cir til dens korresponderende værdi i radians

    noStroke();

    fill(255,255,0);

    ellipse(35,0,22,22); //the moving dot(s), the x[35] is the distance from the center

    pop();

    //push() og pop() bruges, så det hele ikke roterer

    stroke(255,0,0);

    line(60,0,60,height); //a static line

}
```

WINNIE'S KODE

- Jeg uploader en ny sketch.js med mine kommentarer

FEEDBACK FACE-TO-FACE

1. Alexander, Teis, Nina, Line, Mathias
2. Frederik, Margrete, Sissel, Mikael
3. Johannes, Helena, Camilla, Nanna
4. Maiken, Anne, Jonas, Mathias
5. Nicoline, Mathilde, Ester, Cathrine Dybro

- Hvad har I lavet?
- Hvad havde I svært/nemt ved?
- Måtte i gå på kompromis med jeres ønsker?
- Hvad er jeres fremadrettet ambition?

[P A U S E]

C O D I N G .
T I M E .