

PROGRAMMERINGS INTRO MED P5

Citat Khan A.: "Programming is the process of creating a set of instructions that tell a computer how to perform a task. Programming can be done using a variety of computer "languages," such as SQL, Java, Python, and C++."

Materiale:

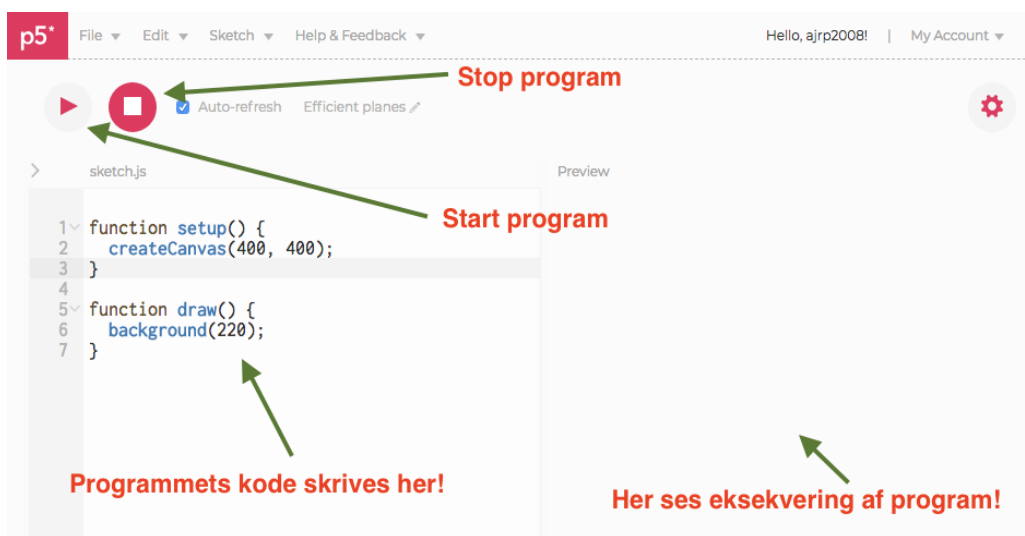
programmerings editor: <https://editor.p5js.org/>

p5.js reference/manual: <https://p5js.org/reference/>

vejr-webservice: <https://weatherstack.com/>

STEP NR.1: ÅBEN [HTTPS://EDITOR.P5JS.ORG/](https://editor.p5js.org/)

Til venstre ses programmets kode / til højre ses det der sker når du kører koden dvs., starter programmet.



Figur 1: Prøv forskellige knapper og se hvordan de virker

1. Start programmet
2. Stop programmet
3. Tegn en anden baggrundsfarve
4. Lav anden størrelse baggrund

STEP Nr.2: PROCEDURAL PROGRAMMING

Programmeringssprog virker ligesom en opskrift... computeren læser koden en linje ad gangen. Prøv følgende opgaver:

1. Hvad sker der når du skriver følgende kode!
2. Hvad sker der når du bytter på linjerne 7 til 11 rundt?
3. Hvordan virker **rect**? Se manualen <https://p5js.org/reference/>
4. Hvordan virker **fill**? Se manualen <https://p5js.org/reference/>
5. Prøv at lave en lille tegning af et hus vha. **rect** og **fill**

```
1 ▼ function setup() {  
2   createCanvas(400, 400);  
3 }  
4  
5 ▼ function draw() {  
6   background(220);  
7   rect(50,50,100,200);  
8   fill(255,0,0);  
9   rect(100,100,100,100);  
10  fill(0,255,0);  
11  rect(300,300,50,50);  
12 }
```

STEP NR.3: VARIABLER

Du kan også lave variabler i stedet for at skrive tal direkte ind i **rect**.

Lav følgende opgaver:

1. Prøv følgende, og forklar hvad der sker på de tre nedenstående billeder?

```
4
5 function draw() {
6   background(220);
7   var a = 100;
8   rect(a,50,100,200);
9   rect(a,100,100,100);
10
11 }
```

```
5 function draw() {
6   background(220);
7   var a = 200;
8   rect(a,50,100,200);
9   rect(a,100,100,100);
10
11 }
```

```
4
5 function draw() {
6   background(220);
7   var a = 100;
8   rect(a*2,50,100,200);
9   rect(a,100,100,100);
10
11 }
```

STEP Nr.4: BRUGERDEFINEREDE FUNKTIONER

Du kender allerede funktionen `rect` men du kan også lave dine egne funktioner:

1. Skriv følgende kode. Prøv at se om du kan regne ud hvordan koden virker?

```
1 function setup() {  
2   createCanvas(400, 400);  
3 }  
4  
5 function draw() {  
6   background(220);  
7   var a = 100;  
8  
9   tegnFirkant(a);  
0  
1 }  
2  
3 function tegnFirkant(a_input){  
4   rect(a_input*2, 50, 100, 200);  
5   rect(a_input, 100, 100, 100);  
6 }
```

2. Lav en funktioner der tegner et "hus". Prøv at tegne mange huse ved at anvende funktionen flere gange.

STEP NR.5: SETUP OG DRAW

Du har sikkert undret dig over hvad “**function setup**” og “**function draw**” er?

1. Se om du selv kan regne ud hvad meningen er med **setup** og **draw** vha. følgende to eksperimenter.

Prøv først følgende

```
1  var x= 0;
2
3  function setup() {
4    createCanvas(400, 400);
5    x=x+1;
6    rect(x,100,100,100);
7  }
8
9  function draw() {
10
11 }
```

Prøv herefter:

```
1  var x= 0;
2
3  function setup() {
4    createCanvas(400, 400);
5
6  }
7
8  function draw() {
9    x=x+1;
10   rect(x,100,100,100);
11 }
```

2. Kan du tegne en lille mand der bevæger sig hen over skærmen?

STEP Nr.6: LAV DIT EGET VEJR-SITE - MED DATA FRA VIRKELIGHEDEN

Nederst ses et udkast til et vejrsite. Ret koden og lav dit helt eget personlige vejr-site:

(her et link til den originale kode:

<https://editor.p5js.org/ajrp/sketches/3uSCvDto>)

Koden kan virke meget uoverskuelig. Formålet er helt overordnet at hente virkelig vejrdato fra et sted på jorden lige nu. Her er i grove træk hvad der sker:

- Vejr-API: Vi ønsker at få oplysninger om vejret (som temperatur) fra internettet.
- CORS-proxy: Nogle gange vil websteder ikke dele dataene med os. En CORS-proxy hjælper os med at bede om dataene på den rigtige måde, så vi kan få dem alligevel.
- Koden: Vi siger til computeren, at den skal hente vejrdato ved at besøge en bestemt webadresse. Vi bruger en proxy, som beder om dataene for os og får dem til os.
- Data: Når vi har dataene, viser vi dem på skærmen. For eksempel viser vi byens navn og temperaturen.
- Tegninger: Vi tegner også en simpel sol, jord og skyer på skærmen for at gøre det mere interessant.

Opgaver: (lav så meget du kan nå)

1. **Registrer dig som bruger på "api.weatherstack.com" og få din egen API-nøgle. Indsæt din nøgle i koden.
2. Hvis du oplever problemer med at hente data, kan du klikke på den URL, der vises i "Console" vinduet nedenfor, og følge vejledningen i det åbnede vindue.
3. Nu er du klar til at begynde at designe dit vejr-site. Start med at oprette baggrunden og gør skyen mørkere, når det er koldt.
4. Tegn regn, for eksempel blå streger på himlen eller lignende, når fugtigheden er over 80%.
5. Jo højere temperaturen er, jo større skal solen være.
6. Overvej at finde en måde at "simulere" vinden på eller prøve andre ideer, du måtte have.

Så er du godt på vej til at skabe dit eget unikke vejr-site!