



Workshop

p5.js

Codle

Inhoud

De basis	1
Function setup()	1
Funtion draw()	2
Background (achtergrondkleur)	3
Raster	4
Vormen en vormen kleur geven	5
Text	8

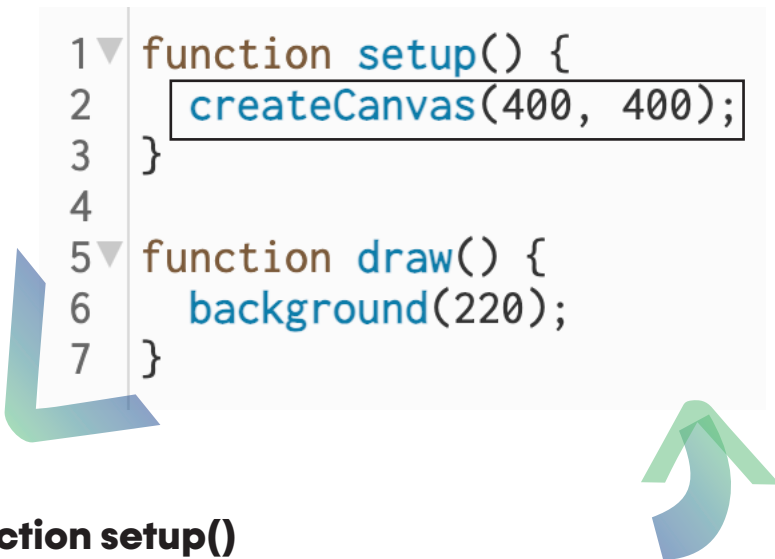
{ Open internet, ga naar: }
<https://editor.p5js.org>

De basis

Hier vind je uitleg over de basisbegrippen van coderen en p5js.

Daarna gaan jullie zelf aan de slag met het maken van opdrachten.

Wanneer je naar internet bent gegaan en je de p5js editor hebt geopend zie je als het goed is het volgende staan:



```
1 ▼ function setup() {  
2   createCanvas(400, 400);  
3 }  
4  
5 ▼ function draw() {  
6   background(220);  
7 }
```

Function setup()

De function setup zorgt ervoor dat de code die je hierin zet een keer word uitgevoerd. Als je kijkt naar de function setup() zie je dat er al code staat tussen de {}.

Je gebruikt de function `setup()` altijd om de grootte van je canvas/werkveld te bepalen. Hiervoor gebruik je het stukje code `createCanvas`. De `createCanvas` heeft altijd 2 getallen tussen de `()` staan. Getal 1 zorgt voor **de breedte** van je canvas. Getal 2 zorgt voor **de hoogte**.

Als je eens wilt zien hoe je canvas uit ziet kun je bovenin links in de editor op de play button klikken.

(Omcirkel het juiste woord)

Deze play button zorgt ervoor dat de computer je code uitvoert en laat zien. Als het goed is zie je nu een grijs vierkant. Stel nou dat je de getallen tussen `()` hoger maakt, dan word het grijze canvas *groter/ kleiner*.

Probeer maar eens!

Je ziet dat `createCanvas` uit 2 woorden bestaat = `create` en `Canvas`. Als je een code schrijft met meerdere woorden, schrijf je deze aan elkaar. Het eerste woord begint altijd met een kleine letter, de woorden die erna komen krijgen een hoofdletter.

Function `draw()`

Nu je in de `setup()` het canvas hebt gemaakt ben je klaar met de `setup()`.

Wanneer je verder kijkt zie je de function `draw()`. Hierin komt de rest van de code die je gaat schrijven. Alles wat hierin komt word 60x per seconde uitgevoerd.

Background (achtergrondkleur)

Als je kijkt na de function draw() zie je dat er al code staat tussen de {}.

Als het goed is staat hier:
background(220);

```
1 ▼ function setup() {  
2   createCanvas(400, 400);  
3 }  
4  
5 ▼ function draw() {  
6   background(220);  
7 }
```

Wanneer je eerder op de play-button hebt geklikt, heb je gezien dat je een grijs canvas hebt. Als je de kleur van je canvas wilt veranderen gebruik je background. Het getal dat je tussen de () zet is de kleur. Dus 220 = grijs. Als je een kleur wilt maken heb je 3 cijfers nodig. Deze 3 cijfers staan voor **RGB: rood, groen, blauw**

Rood=(255,0,0)



Groen= (0,255,0)



Blauw= (0,0,255)



Alle 3 de RGB kleuren hebben een maximum van 255. Dus je kan geen getal boven de 255 invullen. Zo zie je dat je voor rood dus maximaal 255 aan het eerste getal moet geven en aan de andere 2 getallen 0. Stel je zou de andere 2 een hoger getal geven, krijg je een gemengde kleur.

Zo kun je paars bijvoorbeeld maken met de kleurcode:

(140,0,255)

140 van rood - 0 van groen - 255 (maximaal) van blauw

Zo kun je ook turquoise maken door groen en blauw te mengen. Dan word de kleurcode van turquoise ? (vul in)

turquoise = (, ,)

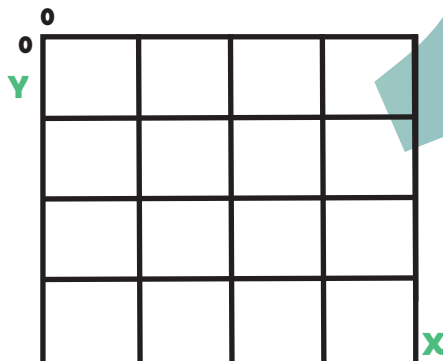


Raster

Om vormen te kunnen plaatsen moet je eerst begrijpen hoe een raster werkt. Als je codeert werk je met een denkbeeldig raster. Dit raster is niet te zien, maar is handig om te maken. Teken eerst een raster op papier, dan weet je precies waar je coördinaten zijn. Stel dat je een canvas van 400 bij 400. *Dus createCanvas(400,400).*

De computer begint altijd **linksboven** bij de 0. Zo teken je vanaf dat punt 400 pixels naar **onder**, dit is de **y - as**. Op de **x - as** begin je aan de linkerkant bij 0, vanaf hier tekent de computer 400 pixels naar **rechts**. Zo zie je dat je altijd 2 coördinaten nodig hebt om iets te plaatsen, een x en y.

Probeer nu eens een punt te plaatsen op (150,220).



Vormen en vormen kleur geven

Vierkant

Een vierkant = in code een rectangle. In code gebruik je de afkorting **rect**.

Wanneer je dus begint met een vierkant te plaatsen begin je met `rect()` te typen.

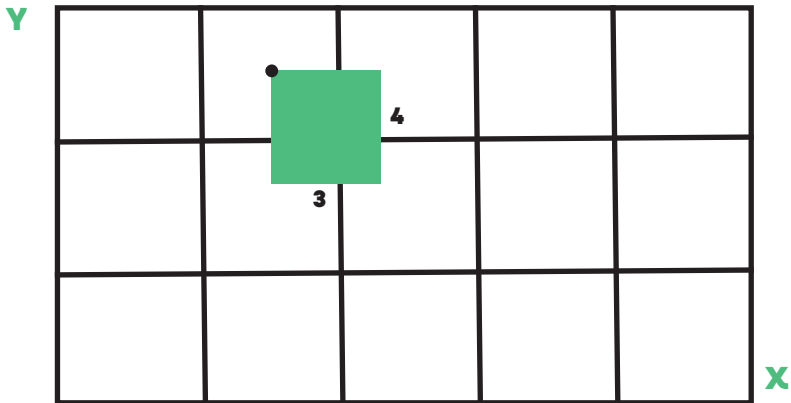
Tussen de () komen vier getallen te staan:

Getal 1: beginpunt op de x - as

Getal 2: beginpunt op de y - as

Getal 3: breedte van je vierkant

Getal 4: hoogte van je vierkant.



Cirkel

Een cirkel = in code een ellipse. Wanneer je dus begint met een cirkel te plaatsen begin je met `ellipse()` te typen.

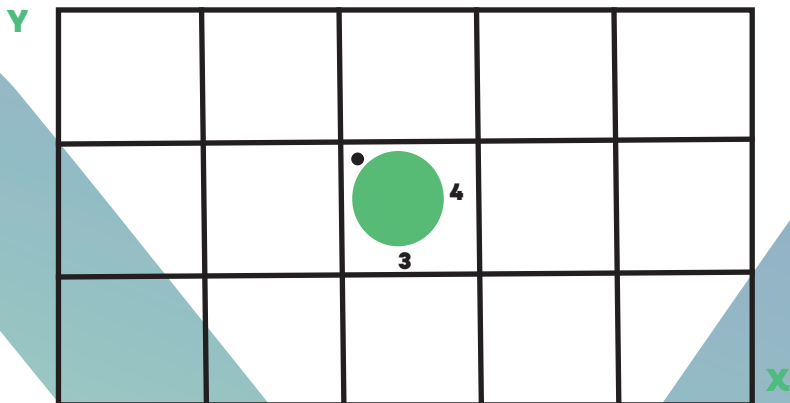
Tussen de `()` komen vier getallen te staan:

Getal 1: beginpunt op de x - as

Getal 2: beginpunt op de y - as

Getal 3: breedte van je cirkel

Getal 4: hoogte van je cirkel.



Om een cirkel te krijg die helemaal rond is moeten de getallen 3 en 4 dus hetzelfde zijn. Zo kun je met een ellipse ook een ovaal maken, dan moet getal 3 groter zijn als getal 4.

Triangle (Fill en noFill)

Nu je weet hoe je een vorm moet plaatsen, wil je hem misschien ook een leuke kleur geven. Een vorm krijgt standaard vanaf het begin een witte kleur, je kunt dit veranderen door de functie `fill()`.

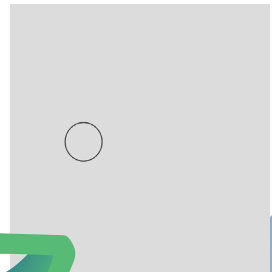
Je zet dit altijd boven de vorm die je deze kleur wilt hebben. Tussen de `()` komt de kleur. Dit kun je doen met de kleurcodes die je eerder hebt geleerd.

Een andere optie is om de naam van kleur tussen de `()` te zetten.

*Als je bijvoorbeeld rood wilt:
`fill("red")`*

Je ziet dat "red" tussen " " is geschreven, dus de kleuren in **woord-vorm** schrijf je tussen " ". Wanneer je wilt dat de vorm geen kleur heeft, kun je `noFill()` gebruiken. Hier komt niets tussen de `()`.

```
5 function draw() {  
6   background(220);  
7   noFill();  
8   ellipse(100,180,50,50);  
9 }
```



Stroke

Vormen krijgen ook standaard een rand eromheen. Wil je deze rand veranderen van kleur gebruik je **stroke()**. Tussen de () komt dan de kleur die je de rand wilt geven. Je kunt de dikte van de rand aanpassen met **strokeWeight()**.

Tussen de () komt een getal te staan, meestal tussen de 1 en 10. Als je wilt geen rand om je vorm wilt, gebruik je **noStroke()**. Dit zet je boven de vorm die je zonder rand wilt.

Text

Je kunt ook tekst toevoegen. Hiervoor gebruik je de functie **text()**. Tussen de () komen 3 dingen te staan. Eerst komt de tekst die je wilt hebben erin te staan, deze moet altijd tussen ''.


Dus bijvoorbeeld: 'codeCanvas'

Plaats 1: tekst altijd tussen ''

Plaats 2: positie op x - as

Plaats 3: positie op y - as

```
5 function draw() {  
6   background(220);  
7   text('code is belangrijk',50,170);  
8 }
```



code is belangrijk

Als je de tekst groter wilt hebben, gebruik je **textSize()**. Dit betekent letterlijk tekstgrootte. Tussen de () kun je een groter of kleiner getal zetten, ligt eraan of je de tekst kleiner of groter wilt.

Deze **textSize()** komt altijd boven de **text()**. Stel dat je de tekst een andere kleur wilt geven doe je dit gewoon weer met de **fill()**.

