



WERKBOEK

Workshop voor digitale kunst
vormen en kleur

Naam:

◀/CODED.

ontdek wat jouw code kan maken

INTRODUCTIE

Vandaag ga je ontdekken hoe je met code je eigen kunst kunt maken. Misschien denk je bij programmeren aan moeilijke computers of ingewikkelde opdrachten, maar eigenlijk is code een manier om de computer te vertellen wat hij moet doen. En dat kan heel creatief zijn!

Tijdens deze workshop ga jij de baas zijn over wat er op het scherm verschijnt. Met code kun je vormen tekenen, kleuren kiezen en dingen laten bewegen. Je kunt bijvoorbeeld een patroon maken van gekleurde cirkels, een bewegende tekening ontwerpen of een kunstwerk maken dat steeds verandert. Met een paar kleine aanpassingen kun je telkens iets nieuws creëren.

Je hoeft hiervoor geen ervaring met programmeren te hebben. We beginnen rustig en leren stap voor stap hoe je eenvoudige opdrachten aan de computer geeft. Daarna ga je zelf experimenteren. Wat gebeurt er als je een andere kleur kiest? Of als je een vorm groter maakt? En hoe ziet jouw kunstwerk eruit als alles gaat bewegen?

Het leuke aan digitale kunst is dat er niet één goed antwoord is. Iedereen maakt iets anders. Je kunt je fantasie gebruiken, nieuwe ideeën uitproberen en ontdekken wat jij mooi of interessant vindt. Soms ontstaan de leukste ontwerpen zelfs per ongeluk!

Aan het einde van de workshop heb je niet alleen geleerd hoe je met code kunt werken, maar heb je ook een eigen digitaal kunstwerk gemaakt. Zo ontdek je dat programmeren niet alleen slim en handig is, maar ook een manier om creatief te zijn en je eigen ideeën tot leven te brengen. Misschien maak jij vandaag wel een kunstwerk dat niemand anders had kunnen bedenken!

PLANNING

0 - 5

Welkom &
introductie

5 - 25

Code uitleg

25 - 45

Inspiratie

45 - 60

Demo

60 - 75

Pauze /
Inloggen

75 - 95

Wat wordt
mijn werk?

95 - 140

Aan de
slag

140 - 160

Afwerken &
Presenteren

WAT IS CODE?

Code is eigenlijk een uitgebreid stappenplan voor de computer. Een computer kan niet zelf nadenken, zoals mensen dat doen. Dus hij heeft heel duidelijke instructies nodig. Die instructies schrijf je in een programmeertaal, zoals Python, JavaScript of Scratch. Dat noemen we “code”.

Met code vertel je stap voor stap wat er moet gebeuren. Bijvoorbeeld:

- maak de achtergrond blauw
- teken een cirkel in het midden van het scherm
- verander de kleur van de cirkel naar rood
- laat de cirkel langzaam naar rechts bewegen

De computer leest deze opdrachten één voor één en voert ze precies uit. Als de instructies goed zijn geschreven, doet de computer precies wat jij wilt. Als er een fout in de code staat, begrijpt de computer de opdracht niet goed of gebeurt er iets anders dan bedoeld.

Met programmeren kun je allerlei dingen maken, zoals games, websites, apps en animaties. Je kunt ook zorgen dat dingen automatisch gebeuren.

Bijvoorbeeld dat een knop reageert wanneer je erop klikt of dat een personage springt als je op de spatiebalk drukt.

Code werkt dus een beetje als een recept in een kookboek:

- het recept vertelt welke stappen je moet volgen
- code vertelt de computer welke stappen hij moet uitvoeren

Hoe beter en duidelijker de stappen zijn, hoe beter het eindresultaat werkt.

mini opdracht

Schrijf zelf 3 opdrachten die je aan een computer zou geven om een simpel kunstwerk te maken.

BASIS VAN P5.JS

p5.js is een programmeertaal waarmee je afbeeldingen en animaties maakt in een webpagina. Je schrijft code en de computer tekent wat jij hebt bedacht.

setup()

De functie setup() wordt één keer uitgevoerd wanneer het programma start.

Wat doet dit?

- function setup() maakt een functie met de naam setup
- { } bevatten de opdrachten die uitgevoerd worden
- createCanvas (600, 400) maak een tekengebied van 600 x 400 pixels

Denk aan setup () als het klaarzetten van een klaslokaal voordat de les begint

draw()

De functie draw() wordt steeds opnieuw uitgevoerd ongeveer 60 keer per seconde

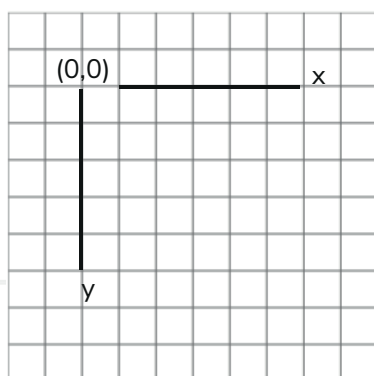
Wat gebeurt hier?

De computer

- voert draw() uit
- begint weer opnieuw
- voert draw opnieuw
- blijft dit doen

Begrijpen waar dingen staan

Op het canvas heeft elke plek een adres:



- In P5.js zit het punt (0,0) linksboven in het scherm
- De X-as bepaalt hoe ver iets naar rechts staat:
 - * Kleine X = meer links
 - * Grote X = meer rechts
- De y-as bepaalt hoe ver iets naar beneden staat:
 - * Kleine Y = bovenaan
 - * Grote Y = verder naar beneden

BASIS VAN VORMEN

In p5.js begin je vaak met simpele vormen.

Met die vormen kun je later grotere tekeningen, animaties en games maken.

De belangrijkste basisvormen zijn:

- Lijn
- Vierkant
- Cirkel
- Driehoek

Elke vorm heeft een eigen functie.

line ()

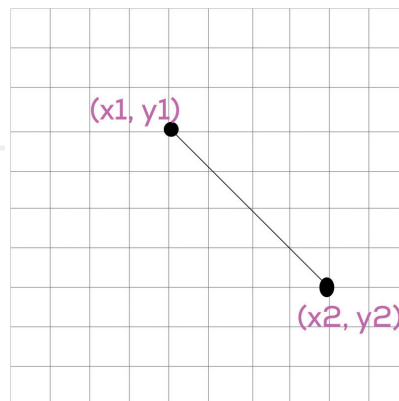
Deze functie tekent een lijn.

Dit codeer je zo:

`line(x - coördinaat 1, y-coördinaat 1, x - coördinaat 2, y-coördinaat 2)`

Voorbeeld:

`line(100, 300, 300, 100)`



rect ()

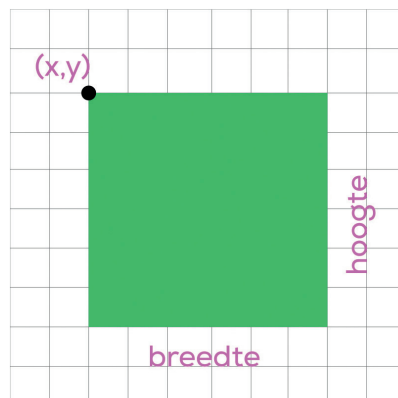
Deze functie tekent een rechthoek of vierkant.

Dit codeer je zo:

`rect (x - coördinaat, y-coördinaat, breedte in pixels, hoogte in pixels)`

Voorbeeld:

`rect(0, 0, 100, 100)`



ellipse ()

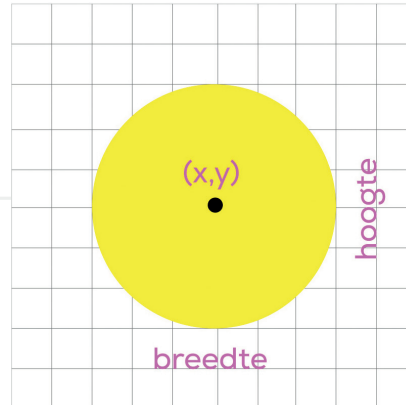
Deze functie tekent een ovaal of cirkel.

Dit codeer je zo:

ellipse (x - coördinaat, y-coördinaat, breedte in pixels, hoogte in pixels)

Voorbeeld:

ellipse(200, 200, 100, 100)



triangle ()

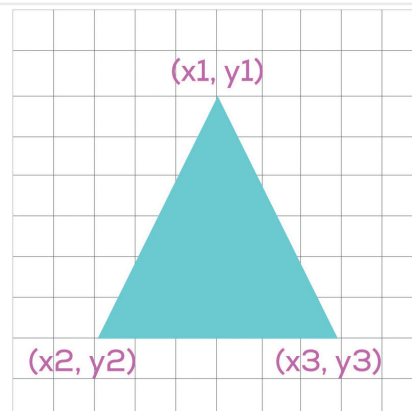
Deze functie tekent een driehoek.

Dit codeer je zo:

Triangle (x - coördinaat 1, y-coördinaat 1, x - coördinaat 2, y - coördinaat 2, x - coördinaat 3, y - coördinaat 3)

Voorbeeld:

triangle(100, 300, 200, 100, 300, 300)



KLEUREN

RGB in p5.js is een manier om kleuren te maken met drie getallen: rood (R), groen (G) en blauw (B).

Elke kleur op het scherm bestaat uit licht. Een computer mengt rood, groen en blauw licht om alle andere kleuren te maken. Elke van de drie waarden kan meestal een getal zijn van 0 tot 255:

0 = geen licht van die kleur

255 = zoveel mogelijk licht van die kleur

Door deze drie waarden te combineren, ontstaat een kleur. Bijvoorbeeld: meer rood geeft een rode tint, en een mix van alle drie geeft andere kleuren zoals geel, blauw of wit.

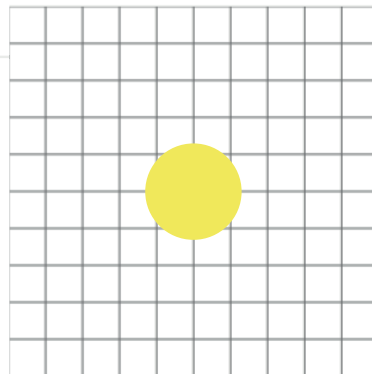
fill ()

Deze functie zorgt ervoor dat de vorm een kleur krijgt.

Voorbeeld:

fill(255, 230, 0)

ellipse(200, 200, 100, 100)



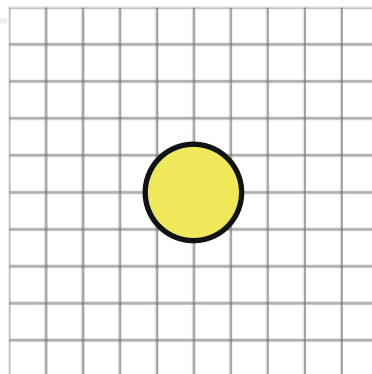
stroke ()

Deze functie tekent een lijn om een vormen heen.

Voorbeeld:

stroke(0)

ellipse(200, 200, 100, 100)



strokeWeight()

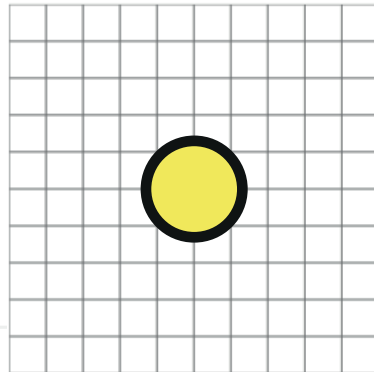
Deze functie bepaalt hoe dik de omlijning wordt.

Voorbeeld:

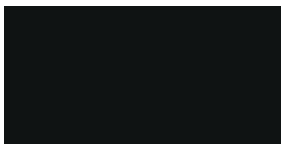
```
stroke(0)
```

```
strokeWeight(4)
```

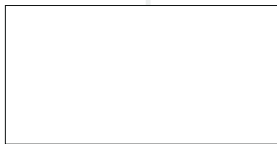
```
ellipse(200, 200, 100, 100)
```



Hier zijn wat basis kleuren:



(0)



(255)



(200)



(255, 230, 0)



(255, 0, 0)



(0, 255, 0)



(0, 0, 255)



(155, 0, 255)



(255, 0, 255)



(250, 150, 0)

LOG IN

Stap 1: open je webbrowser en ga naar <https://editor.p5js.org/login>

Stap 2: vul de gegevens in:

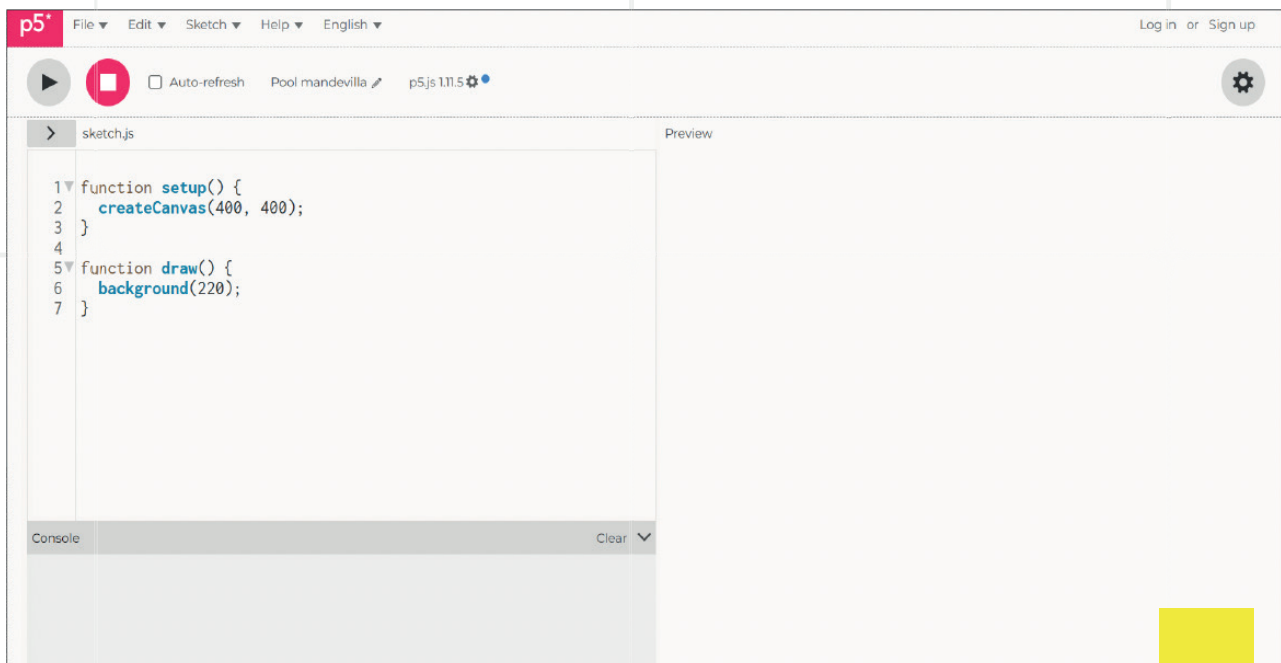
Gebruikersnaam: jouw school e-mail
Wachtwoord: Coded.2026

Log In

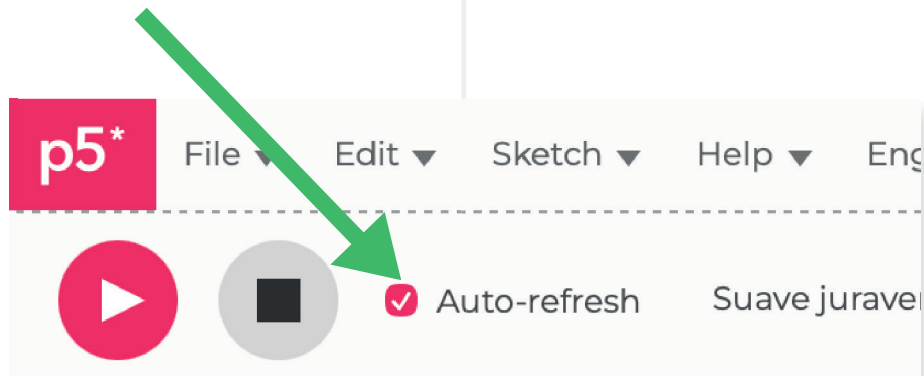
Email or Username

Password

Dan kom op deze pagina terecht:



Zet deze aan. Zo zorg dat je altijd kan zien wat je maakt



'setup()' gebruik je om alles klaar te zetten wanneer het programma start. Deze functie loopt maar één keer.

'draw()' gebruik je voor alles wat steeds moet veranderen of opnieuw getekend moet worden. Deze functie blijft zichzelf herhalen zolang het programma loopt.

'createCanvas()' bepaalt de grootte van je canvas in pixels. Deze staat standaard ingesteld op (400, 400).

background()' bepaalt de achtergrondkleur van je canvas.

Bij pagina basis p5.js vind je een uitgebreide uitleg hierover.

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

IDEE VOOR EIGEN KUNSTWERK

Voordat je wat gaan maken, ga je eerst kort nadenken.

Mijn kunstwerk wordt:

- ☐ Een patroonkunstwerk
- ☐ Een bewegend kunstwerk
- ☐ Een interactief kunstwerk
- ☐ Iets anders: _____

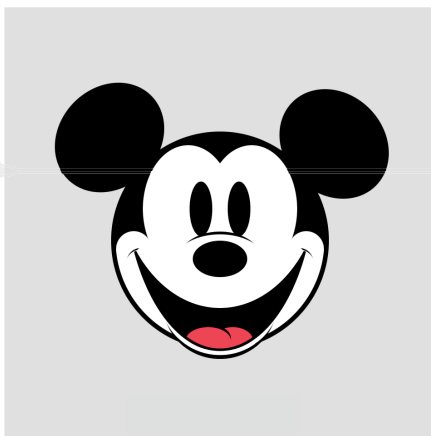
Mijn thema of gevoel:

Bijvoorbeeld: druk, rustig, vrolijk, chaos, ruimte, stad, natuur, game, muziek.

Kleuren die ik wil gebruiken:

Vormen die ik wil gebruiken:

- ☐ Cirkels
- ☐ Vierkanten
- ☐ Driehoeken
- ☐ Lijnen
- ☐ Stippen
- ☐ Eigen vormen

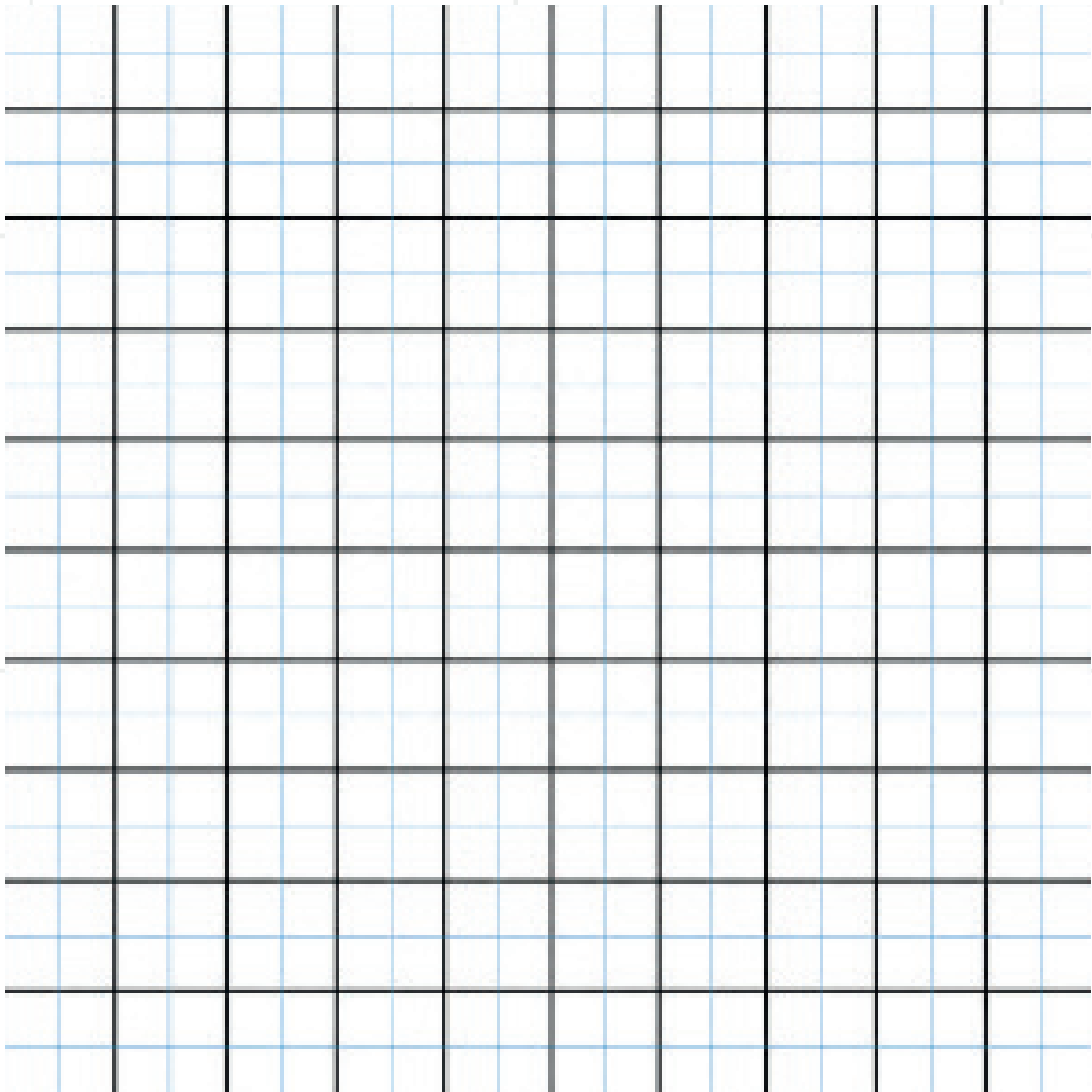


SCHETS JE WERK

maak eerst een snelle schets. Het hoeft niet mooi te zijn. het is een plan voor je digitale kunstwerk.

Het help om deze vragen te stellen aan zelf alvast:

- Welke vormen ga je gebruiken?
- hoe groot mogen de vormen zijn?
- Welke kleuren ga je gebruiken?
- Wil je dat iets gaat bewegen?





p5.js cheatsheet



programma structuur

```
//runt een keer wanneer het programma  
wordt opgestart  
function setup(){  
  createCanvas(800,600); } //breedte,  
hoogte in pixels
```

systeemvariabelen

width / height
//breedte / hoogte van het canvas

mouseX / mouseY
//huidige horizontale muispositie / huidige
verticale muispositie

kleur

```
fill(120); //grijs: 0-255  
fill(100,125,255); //r, g, b: 0-255  
fill('#ccc'); //3-cijferige hex-code  
fill('#222222'); //6-cijferige hex-code  
color(0, 0, 255); //p5 kleurobject  
fill(255, 0, 0); // rood  
fill(0, 255, 0); // groen  
fill(0, 0, 255); // blauw  
fill(255, 255, 0); // geel
```

wiskunde

```
+ = plus  
- = min  
/ = delen  
* = keer  
// = comments
```



2-D shapes

```
line(x1, y1, x2, y2);
```

```
ellipse(x, y, breedte, hoogte);
```

```
rect(x, y, breedte, hoogte);
```

```
triangle(x1, y1, x2, y2, x3, y3); //tekent een  
driehoek vanaf 3 punten. er is geen  
breedte hoogte nodig
```

```
text("mijn tekst", x, y, breedte, hoogte);
```

attributen

```
background(kleur);  
//stel de achtergrondkleur in
```

```
fill(kleur);  
//stel de vulkleur in
```

```
noFill();  
//geen vulkleur
```

```
stroke(kleur);  
//stel de lijnkleur in
```

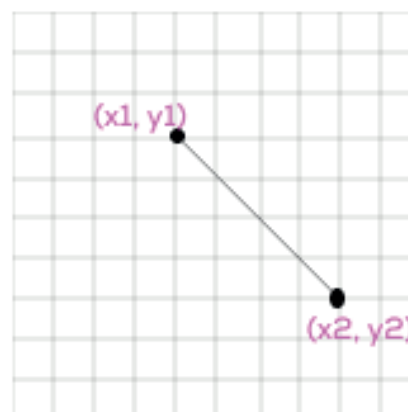
```
strokeWeight(dikte);  
//stel de lijndikte in
```

```
noStroke();  
//geen lijn
```

```
textSize(pixels);  
// bepaalt de grootte van de tekst in pixels
```



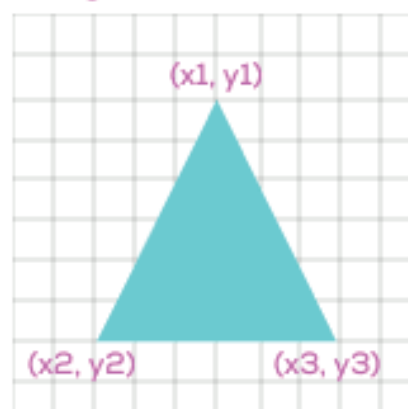
line()



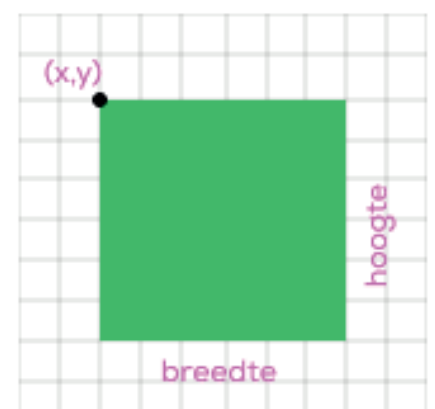
ellipse()



triangle()



rect()



MIJN EINDWERK

Titel van mijn kunstwerk:

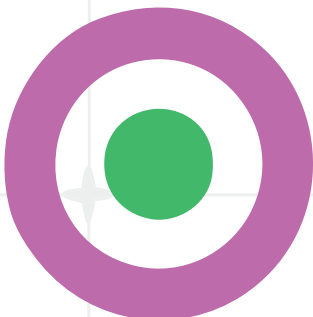
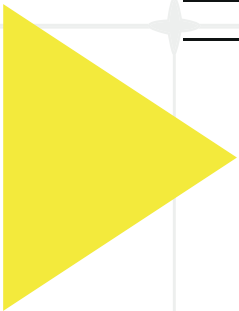
Wat heb ik gemaakt?

Welke vormen heb je gebruikt?

Wat vind ik goed gelukt?

Wat vond ik lastig?

Wat zou ik nog willen toevoegen als ik meer tijd had?



AFSLUITING

Wat heb je vandaag geleerd over code?

Wat vond je verrassend?

Waar zie jij code terug in het echte leven?

Zou je vaker iets willen maken met code?

