

Rekursie

Steven Bronsveld

February 23, 2019

1 Gegevens

Uiterlijke inleverdatum: **Datum 1**

1.1 Links

- [Github.com/StevenBrons](https://github.com/StevenBrons)
- <https://natureofcode.com/book>
- <http://hello.processing.org/editor/>

2 Leerdoelen

Deze week leer je omgaan met de volgende concepten:

- Variabelen: `int`
- Standaard functions:
 - `size(int width, int height)`
 - `circle(int x, int y, int r)`
 - `random(float bound)`
 - `stroke(float r, float g, float b)`
 - `fill(float r, float g, float b)`
 - `background(float r, float g, float b)`
 - `strokeWeight(int w)`
- Globale variabelen: `width`, `height`, `mouseX`, `mouseY`
- Het aanroepen van methods
- Het maken van eigen methods
- Het gebruiken van for-loops
- Standaard operatoren: `a + b`, `a - b`, `a * b`, `a / b`, `a % b`, `a ^ b`

3 [optioneel] Voorbeelden

Bekijk de volgende voorbeelden. Ga per regel na of je snapt wat er gebeurt. Pas eventueel wat dingen aan om te kijken wat het effect is. Als je codeert mag je het internet en de voorbeelden altijd gebruiken, maak hier dus gebruik van!

1. `houses`
2. `chess`

4 Do you want to build a snowman?

Je gaat de eerste code schrijven! Zorg ervoor je een nieuwe *sketch* aanmaakt.

4.1 Een cirkel

Plak de volgende code in je lege sketch. Run de sketch om te kijken wat dit doet!

```
1 void setup() {  
2     size(500,500);  
3     circle(100,100,70);  
4 }  
5 void draw() {  
6 }
```

Pas de code aan zodat de cirkel in het midden van het scherm staat.

4.2 Het midden

Als je de grote van het scherm aanpast (pas de arguments van **size** aan), zul je zien dat de cirkel niet meer in het midden staat. Gebruik de variabelen **width** en **height** om het midden van het scherm te berekenen. Zorg ervoor dat de cirkel in het midden van het scherm blijft staan, onafhankelijk van de grote van het scherm.

4.3 Een sneeuwpop

Maak een sneeuwpop door twee cirkels half-boven de eerste cirkel te tekenen. Zorg ervoor dat deze cirkels ook op de juiste plek blijven staan als je de schermgrootte verandert!

4.4 Een method

Het tekenen van de sneeuwpop is een simpel **algoritme**. Als je een programma wil maken is het enorm belangrijk je het op te delen. Dit opdelen gebeurt onder meer door het maken van **methods**. Plaats de volgende method **helemaal onderaan** je de sketch.

```
1 void drawSnowman(int x, int y) {  
2 }
```

Plaats de volgende regel code onderaan in **setup**.

```
1     drawSnowman(int(random(width)),int(random(height)));
```

Verplaats het tekenen van je sneeuwpop naar de **drawSnowman** method. Het is belangrijk dat je de sneeuwman op de coördinaten **x,y** plaatst.

4.5 Meer sneeuwpoppen

Maak een **for** loop in de **setup** die 10 sneeuwpoppen op willekeurige plekken op het scherm plaatst.

4.6 Gekleurde sneeuwpoppen

In de wereld van code kan alles! Waarom zouden we ons limiteren tot witte sneeuwpoppen? Zorg ervoor dat alle sneeuwpoppen verschillende kleuren hebben!

4.7 Nog meer sneeuwpoppen

Voeg de volgende method toe:

```
1 void mouseClicked() {  
2 }
```

Deze method word uitgevoerd op het moment dat je met de muis op de sketch klikt. Zorg ervoor dat er een sneeuwpop op de plek van de muis verschijnt als je klikt. (tip: zoek op het internet naar een manier om de muis coördinaten te krijgen)

4.8 [Bonus] Hogere sneeuwmannen

Pas de method `drawSnowman` aan zodat er nog een derde variable is:

```
1 void drawSnowman(int x, int y, int n) {  
2 }
```

Deze derde variable geeft de hoogte van de sneeuwpop aan. Zorg ervoor dat ook de hoogte (het aantal cirkels) variabel is! Let op! De cirkels moeten zowel kleiner worden, als boven op elkaar gestapeld worden! Zorg dat je ook 100-ballen-hoge sneeuwpoppen kunt maken! Pas vervolgens

```
1 drawSnowman(int(random(width)), int(random(height)));
```

aan zodat de sneeuwpoppen tussen de 2 en 6 hoog zijn.

4.9 [Bonus] Sneeuwmannen dans

Maak de method `makeCircle`, met als parameters een `int n`

```
1 void makeCircle(int n) {  
2 }
```

Deze method moet `n` sneeuwmannen in een cirkel om het midden van het scherm tekenen. De sneeuwmannen moeten allemaal op gelijke afstand van elkaar staan.

5 Inleveren

Als je klaar bent met de hele opdracht kun je deze naar je *repository* pushen.