

## 1 Leerdoelen

---

- Het maken en gebruiken van classes
- Het gebruiken van objecten (instances of classes)
- Het maken van methods
- Het werken met een ArrayList

## 2 Uitleg

---

- 8: Object-Oriented Programming  
[https://www.youtube.com/watch?v=YcbcfkLzgvs&list=PLRqwX-V7Uu6bb7z2IJaTlwzIg\\_5yvL4i](https://www.youtube.com/watch?v=YcbcfkLzgvs&list=PLRqwX-V7Uu6bb7z2IJaTlwzIg_5yvL4i)
- 9: Arrays en ArrayList  
<https://www.youtube.com/watch?v=NptnmWvkbTw&list=PLRqwX-V7Uu6b09RKxH0bluh-aPgrrvb4a&index=1>
- Particle Systems - The Nature of Code  
<https://natureofcode.com/book/chapter-4-particle-systems/>  
Alleen 4.1, 4.2, 4.3, 4.4, 4.5
- Chapter 2. Particle Systems  
<https://www.youtube.com/watch?v=vdgiqMkFygc&list=PLRqwX-V7Uu6Z9hI4mSgx2F1E5w8zvjmEy&index=1>  
Alleen 4.1, 4.2, 4.3, 4.4, 4.5

## 3 Voorbeelden

---

- clouds
- ball

## 4 Opdrachten

---

Deze week gaan we voor het eerst *beweging* maken! Voortaan is het belangrijk dat je alleen maar tekent in de `draw` functie. Het is extra belangrijk dat je de uitlegvideo's over Object Geïntendeerd Programmeren!

### 4.1 Een class

Het is super handig om bepaalde variabelen en functies op die variabelen samen in één object te bundelen. Hiervoor gebruiken we een **class**. In een class kun je meerdere variabelen en functies stoppen. Zorg ervoor dat je goed snapt hoe de volgende sketch werkt:

```
Ball ball1;
Ball ball2;

void setup() {
  size(500,500);
  ball1 = new Ball(new PVector(width / 2, height / 2));
  ball2 = new Ball(new PVector(width / 2, 0));
}

void draw() {
  background(255);
  ball1.move();
  ball1.draw();
  ball2.move();
  ball2.draw();
}

class Ball {
  // Variabelen van deze class
  PVector pos = new PVector(width / 2,height / 2);
  PVector gravity = new PVector(0,5);

  // De constructor
  Ball(PVector beginPos) {
    this.pos = beginPos;
  }

  void move() {
    pos.add(gravity);
  }

  void draw() {
    circle(pos.x,pos.y,50);
  }
}
```

Pas de code aan zodat je naast de begin positie van de bal, ook de kleur kan aangeven.

## 4.2 Stuiterballen

Pas de sketch aan door het volgende toe te voegen (en de method `move` te vervangen).

```
PVector bounceForce = new PVector(0,0);

void move () {
    bounceIfBottom();
    pos.add(gravity);
    pos.add(bounceForce);
    bounceForce.mult(0.9);
}

void bounceIfBottom() {
    //TODO
}
```

Deze method moet de bal laten stuiteren als deze de onderkant van het scherm raakt.

## 4.3 Een ArrayList

In het de sketch van de vorige opdracht worden er twee variabelen gebruikt (`ball1` en `ball2`) om de ballen op te slaan. Maar wat nu als we 4 ballen op willen slaan, of 100000? We willen niet een hele lijst met variabelen maken natuurlijk. Daarom gebruiken we een **ArrayList**. Pas je gemaakte sketch aan zodat je met behulp van een **ArrayList** zoveel ballen kan maken als je wil! Pas vervolgens de code aan zodat je 100 ballen op willekeurige plekken maakt!

### Tip

Je krijgt een willekeurig getal door `int(random(maximum.hoeveelheid))`