

Objecten

Wat zijn objecten (in het echte leven)

- In het echte leven kunnen we voorwerpen, zoals bijvoorbeeld een auto, zien als een object.
- Een auto heeft een set eigenschappen en methodes:
 - Alle auto's hebben dezelfde eigenschappen, maar de waarden van de eigenschappen verschillen.
 - Alle auto's hebben dezelfde methodes, maar deze worden uitgevoerd op een verschillend moment.

Wat zijn objecten (in het echte leven)

auto.merk = Mustang	auto.start()
auto.model = Mach 1	auto.stop()
auto.bouwjaar = 1973	auto.rem()
auto.gewicht = 1547	auto.veranderSnelheid()
auto.kleur = grijs	auto.draaiRechts()
...	...

Wat zijn objecten (in Javascript)

- In Javascript bestaat een object ook uit een set van eigenschappen (properties) en methoden (methods).
- Objecten zijn erg handig om bij elkaar horende informatie te groeperen en te organiseren. Ze zijn zelfs eenvoudiger in het gebruik dan arrays omdat ze werken met 'genaamde waarden' in plaats van met een numerieke index.

```
// Notatie op één regel
```

```
let object = { naam1: waarde, naam2: waarde, naam3: waarde };
```

```
// Notatie op meerdere regels.
```

```
let object = {  
    naam1: waarde,  
    naam2: waarde,  
    naam3: waarde  
};
```

```
// Een voorbeeldobject 'persoon' met drie eigenschappen.
```

```
let persoon = {  
    voornaam: 'Kimberly',  
    achternaam: 'Willems',  
    beroep: 'lector'  
};
```

Object properties

De eigenschappen (properties) in een object hebben elk:

- een naam (string)
- een waarde (eender welke geldige Javascript-waarde)
 - Variabelen
 - Diverse gegevenstypen
 - Functies
 - Arrays
 - objecten

```
// Een voorbeeldobject 'game' met verschillende waarden.  
let game = {  
    titel: 'Assassin's Creed Valhalla',      // Stringwaarde  
    editie: 'Drakkar Edition',                // Stringwaarde  
    ean: 3307216169116,                       // Numerieke waarde  
    pegi: { leeftijd: 18, inhoud: 'Geweld' } // Objectwaarde  
};
```

Object properties uitlezen en gebruiken

Twee verschillende manieren uitlezen:

- *Puntnotatie*: we schrijven eerst de objectnaam en daarna verbinden we de eigenschapnaam aan de objectnaam met een punt.
- *Arraynotatie*: we schrijven eerst de objectnaam en daarna schrijven we de eigenschapnaam tussen blokhaken én aanhalingstekens.


```
// Een voorbeeldobject 'persoon' met drie eigenschappen.
```

```
let persoon = {  
    voornaam: 'Kimberly',  
    achternaam: 'Willems',  
    beroep: 'lector'  
};
```

```
// Puntnotatie
```

```
document.getElementById('uitkomst').innerHTML = 'Mijn naam is ' +  
persoon.voornaam + ' ' + persoon.achternaam + ' en ik ben ' + persoon.beroep + '  
van beroep.';
```

```
// Arraynotatie
```

```
document.getElementById('uitkomst').innerHTML = 'Mijn naam is ' +  
persoon["voornaam"] + ' ' + persoon["achternaam"] + ' en ik ben ' +  
persoon["beroep"] + ' van beroep.';
```

Object methods

- De methoden in een object zijn de acties die kunnen worden uitgevoerd op het object.
- Methoden worden opgeslagen in eigenschappen als functiedefinities.

```
// Een voorbeeldobject 'game' met eigenschappen en methoden aanmaken.
let game = {
  titel: 'Assassin's Creed Valhalla',
  editie: 'Drakkar Edition',
  prijs: getGamePrice('3307216169116'), // Waarde uit functieresultaat
  volledige_titel: function() {          // Functiewaarde
    return this.titel + ' - ' + this.editie;
  }
};
```

We gebruiken in de functiedefinitie het woord 'this' om te refereren naar de 'eigenaar van de functie, namelijk het object.

Object methods

- We kunnen de objectmethode benaderen via een puntnotatie.

```
// Resultaat in html-element bevat de eigenlijke titel en editie.  
document.getElementById('uitkomst').innerHTML = game.volledige_titel();
```

```
// Resultaat in html-element bevat de functiedefinitie zelf  
document.getElementById('uitkomst').innerHTML = game.volledige_titel;
```