***Hoofdstuk 3***

- let functienaam = (parameters) => { … je code … }. Deze functie kan je **callen**, om vervolgens parameters mee te geven.

Local scope 🡪 binnen de curly brackets van een functie bijv. **Bij var is dit niet het geval, bij const en let wel.**

Global scope 🡪 slaat op de scope van het hele document.

**lexical scoping:** elke local scope kan zijn parent scope zien, plus de global scope.

Een **function declaration** is als je waardes meegeeft aan de parameters.

- Javascript is zo slim dat als je teveel of te weinig parameters meegeeft, de rest wordt genegeerd. Het is daarnaast ook mogelijk om een parameter een default value te geven voor als er niks wordt meegegeven.

- **Closure:** een functie heeft toegang tot de omvattende functie, ook als deze niet op dat moment aan het runnen is.

- Er is altijd een dilemma tussen leesbaarheid en snelheid, maar de leesbaarheid staat op 1.

- Een functie die zichzelf één of meerdere keren aanroept noemen we **recursive**.

- Het is belangrijk om algemene functies te schrijven voor alles wat je meerdere keren gaat doen, om zo DRY te waarborgen.