

## Loops

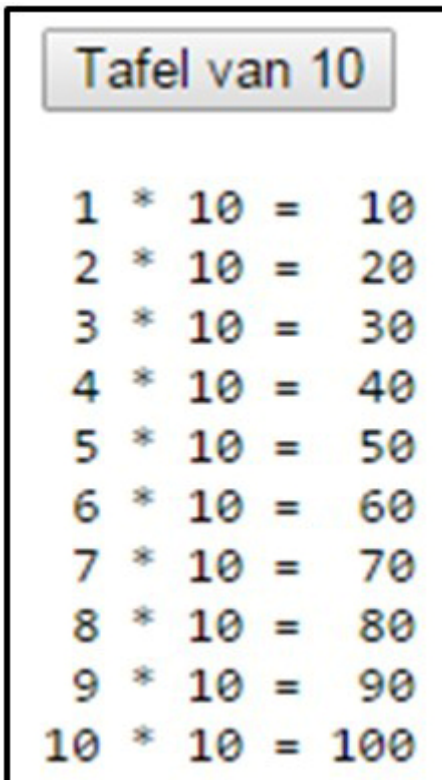
### Het for-statement

De meest toegepaste vorm van een herhalingsopdracht is een for-statement. Hier is een voorbeeld:

```
<button id = "knop">Tafel van 10</button>
<p id="demo"></p>

<script>
document.getElementById("knop").addEventListener("click", myFunction);
function myFunction() {
    var s = "";
    for (var i = 1; i <= 10; i++) {
        s += i + " * 10 = " + (i*10) + "<br>";
    }
    document.getElementById("demo").innerHTML = s;
}
</script>
```

De uitvoer ziet er zo uit:



Tafel van 10			
1	*	10	= 10
2	*	10	= 20
3	*	10	= 30
4	*	10	= 40
5	*	10	= 50
6	*	10	= 60
7	*	10	= 70
8	*	10	= 80
9	*	10	= 90
10	*	10	= 100

De kern van het programma wordt gevormd door het for-statement:

```
for (var i = 1; i <= 10; i++) {  
    s += i + "* 10 = " + (i*10) + "<br>";  
}
```

Het for-statement begint met het woord `for` en daarna staan tussen haakjes drie dingen:

- De declaratie en initialisatie van een variabele (in dit geval `i`): `var i = 1`.
- Een conditie die `true` of `false` kan opleveren, waarin `i` een rol speelt: `i <= 10`.
- Een wijziging van de variabele: `i++`.

Tussen de accolades staat de body van het for-statement. Deze body wordt, in dit geval, tien keer uitgevoerd.

### Wat gebeurt er precies in een for-statement?

In de volgende beschrijving staat wat er precies gebeurt bij de uitvoering van het `for`-statement uit de vorige paragraaf.

	I	C	V	B
<code>for</code>	<code>i = 1;</code>	<code>i &lt;= 10;</code>	<code>i++</code>	Body van <code>for</code> -statement
	initialisatie van controle- variabele	conditie met eindwaarde van controlevariabele	verhoging van controle- variabele	<code>s += i + "* 10 = " + (i*10) + &lt;br&gt;;</code>

1. Het eerste dat er gebeurt bij elk `for`-statement is dat de controlevariabele (`i` in dit geval) een beginwaarde krijgt (I = initialisatie van de controlevariabele).
2. Het tweede dat er gebeurt bij elk `for`-statement is de controle (test of conditie) C: levert deze waar of niet waar, `true` of `false`? Omdat `i` gelijk is aan `1`, is `i <= 10` waar. Als uit de test niet waar zou komen, dan zou het `for`-statement onmiddellijk stoppen. De waarde `10` noemen we de eindwaarde van het `for`-statement.
3. Omdat de conditie in dit geval `true` is, wordt de body B uitgevoerd. In dit voorbeeld het schrijven van de eerste regel van de tafel van `10`.
4. Vervolgens wordt de controlevariabele verhoogd in V, in dit geval door `i++`, dus `i` krijgt de waarde `2`.
5. Na deze verhoging wordt de test C opnieuw uitgevoerd. Geldt nog steeds `i <= 10`? Ja, dat is het geval. Omdat de test `true` is, wordt de body B opnieuw uitgevoerd, `i` wordt verhoogd in V, de test C wordt gedaan.

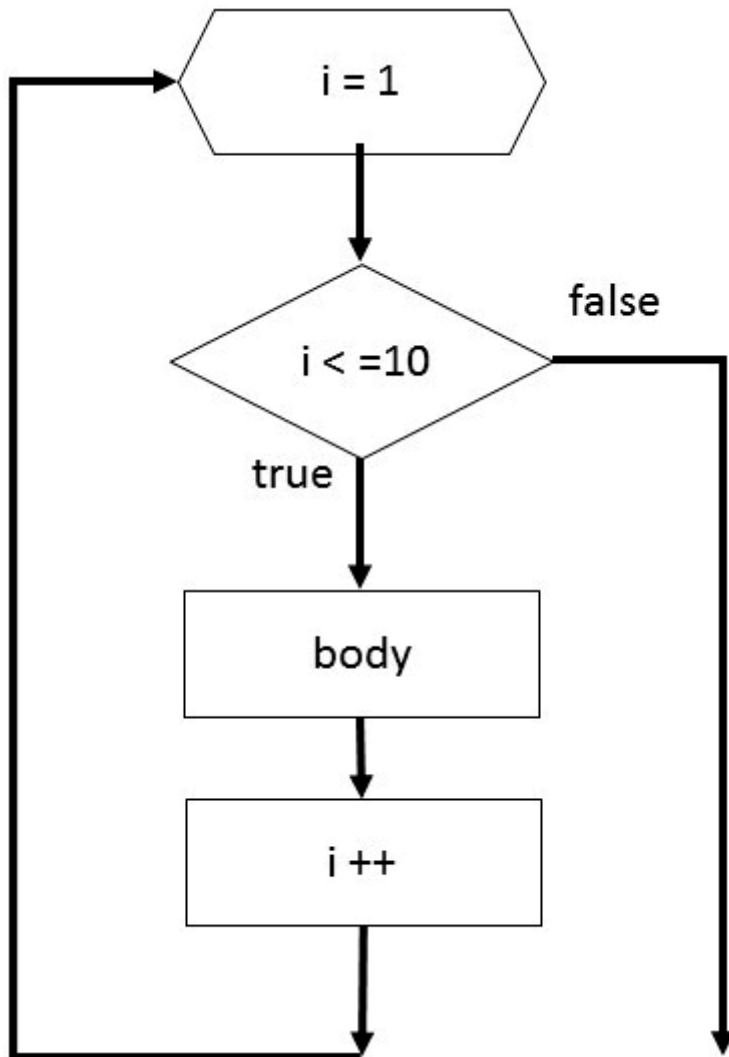
Dit proces herhaalt zich net zolang tot de conditie `false` is. Dit zal het geval zijn op het moment dat `i` verhoogd is tot `11`. Dan is voor het eerst `i <= 10` niet waar. Het `for`-statement stopt zijn werk. Dat betekent dat bij elk van de waarden `1` tot en met `10` de body is uitgevoerd en dus komen er tien regels in de uitvoer. Door in het controlegedeelte het getal `10` in bijvoorbeeld `20` te veranderen, kunt u met even weinig moeite `20` regels laten schrijven.

Het hele proces kunt u met de letters I (Initialisatie), C<sub>t</sub> (Conditie is `true`), B (Body) en V (Verandering van de Variabele) en C<sub>false</sub> (Conditie is `false`) zo samenvatten:

I C<sub>t</sub>BV C<sub>t</sub>BV C<sub>t</sub>BV ... C<sub>t</sub>BV C<sub>t</sub>BV C<sub>false</sub>

Anders gezegd: een `for`-statement begint altijd met de initialisatie die één keer wordt uitgevoerd, doorloopt een aantal keren het rijtje CBV zolang de conditie `true` is en eindigt als de conditie `false` is.

In de volgende figuur is het principe van een `for`-statement in beeld gebracht.



Bij de uitvoering van een `for`-statement wordt elke keer de test gedaan vóórdat (eventueel) de `body` wordt uitgevoerd. Als de conditie `true` is, wordt de `body` uitgevoerd; is dit niet het geval, dan stopt het `for`-statement.

Het kortst mogelijke `for`-statement bestaat dus uit: `! Cfalse`. Met andere woorden: als de conditie meteen `false` is, wordt de `body` helemaal niet uitgevoerd.