

# Javascript intro

## OPERATOREN

# Inhoud

- **Javascript operatoren:**
  - **Rekenkundige operatoren**
  - **Logische operatoren**
  - **Vorrangsregels bij operatoren**
- **condities**

# Javascript Syntax - Herhaling

Declaratie van variabelen

```
let x, y;           // How to declare variables
```

Toekenning van waarden

```
x = 5; y = 6;       // How to assign values
```

Berekeningen maken

```
z = x + y;          // How to compute values
```

Vaste waarden worden 'literals' genoemd

Vb Numerieke literals

10.50

1001

Vb Tekst-literals

"John Doe"

'John Doe'

Vraag: wat schrijft de volgende statement naar de web pagina?

```
document.write("John Doe");
```

# Javascript syntax - Herhaling

Javascript variabelen

```
let x;  
x = 6;
```

Javascript commentaren

```
let x = 5;    // I will be executed  
// let x = 6;  I will NOT be executed
```

Javascript statements

```
let x, y, z;    // Statement 1  
x = 5;          // Statement 2  
y = 6;          // Statement 3  
z = x + y;      // Statement 4
```

# Javascript syntax - Herhaling

Puntkomma's (semicolons) versus komma's

```
let a, b, c;      // Declareer 3 variables
a = 5;           // Ken de waarde 5 toe aan variabele a
b = 6;           // Ken de waarde 6 toe aan variabele b
c = a + b;
// Ken de waarde van de som van a en b toe aan variabele c
```

# Variabele declaraties – keyword let

Declaratie van variabelen:

```
let person = "John Doe";  
let carName = "Volvo";  
let price = 200;
```

Ofwel ook mogelijk:

```
let person = "John Doe", carName = "Volvo", price = 200;
```

Nog voorbeelden van variabele declaraties:

```
let price1 = 5;  
let price2 = 6;  
let total = price1 + price2;
```

# Variabelen versus Constanten

Variabele kan in de loop van het programma veranderen

Constante blijft een vaste waarde behouden

een constante bevat een constante waarde, deze mag tijdens het programma niet veranderen.

Bv Declaratie van constante PI

`const` PI = 3.14;

Opdracht:

Bereken de oppervlakte van een cirkel met straal = 2.5 en de omtrek van een cirkel met straal = 3.5):

# Voorbeeld output via document.write()

Is meestal in combinatie met HTML (Web document)

```
<!DOCTYPE html>
<html>
<body>
<h1>My First Web Page</h1>
<p>My first paragraph.</p>
<script>
document.write("My Age is :" + (2 + 2*22));
</script>
</body>
</html>
```



# Opdracht

Je koopt een TV aan ...euro nettoprijs(excl BTW)

De BTW is steeds 21%

Bereken de brutoprijs (incl BTW) van de TV.

Wat definieer je als variabele en wat als constante?

# Operatoren bij numerieke datatypes

Operator	Beschrijving
+	Optellen
-	Aftrekken
*	Vermenigvuldigen
/	Delen
%	Modulus (Rest bij deling)
++	Incrementeren (1 bij optellen)
--	Decrementeren (1 van aftrekken)

# Numerieke datatypes - Operatoren

- + - \* / operator bij integer type (<https://www.learn-js.org/>)

```
let a = 1;
let b = 2;
let c = a + b; // c is now equal to 3
console.log(c);

console.log(3 - 5); // outputs -2
console.log(3 * 5); // outputs 15
console.log(3 / 5); // outputs 0.6
```

## Oefening

**Vraag:** wat geeft de volgende code als resultaat?

```
let getal1 = 1;
let getal2 = 5;
console.log("de som is " + getal1 + getal2);
console.log("de som is " + (getal1 + getal2));
```

# Voorbeelden numerieke operatoren

Addition (optelling)

```
let x = 5;  
let y = 2;  
let z = x + y;  
console.log(z);
```

Subtraction (aftrekking)

```
document.write(5-2);
```

Multiplication(vermenigvuldiging)

```
document.write(5*2);
```

Division(deling)

```
let x = 5;  
let y = 2;  
let z = x / y;  
document.write(z);
```

# Voorbeelden numerieke operatoren

Modulus (Rest bij deling)

```
let x = 5;  
let y = 2;  
let z = x % y;  
document.write(z);
```

Increment(eren) (Unaire operator)

```
let x = 5;  
x++;  
document.write(x);
```

Decrement(eren)(Unaire operator)

```
let x = 5;  
x--;  
document.write(x);
```

# (Assignment) toewijzings-Operatoren

wordt meestal gebruikt bij numerieke variabelen

Operator	Voorbeeld	Zelfde als
=	$x = y$	$x = y$
+=	$x += y$	$x = x + y$
-=	$x -= y$	$x = x - y$
*=	$x *= y$	$x = x * y$
/=	$x /= y$	$x = x / y$
%=	$x \% = y$	$x = x \% y$
**=	$x ** = y$	$x = x ** y$

# Oefeningen

Wat geven de volgende javascript oefeningen als resultaat ?

## Oefening 1

```
let textje = "Good ";  
textje += "Morning";  
document.write(textje);
```

## Oefening 2

```
let x = 10;  
x += 5;  
document.write(x);
```

## Oefening 3

```
let x = 10;  
x -= 5;  
document.write(x);
```

# Oefeningen - vervolg

## Oefening 4

```
let x = 10;  
x *= 5;  
document.write(x);
```

## Oefening 5

```
let x = 10;  
x /= 5;  
document.write(x);
```

## Oefening 6

```
let x = 10;  
x %= 5;
```

## Oefening 7

```
let x = 10;  
x **= 2;  
document.write(x);
```



# Voorrangsregels van Operatoren

Vermenigvuldiging (\*) en deling(/) hebben voorrang op optelling (+) en aftrekking(-)

```
let x = 100 + 50 * 3;
```

```
document.write(x);
```

```
x = (100 + 50) * 3;
```

```
document.write(x);
```

```
let y = 100 + 50 - 3;
```

```
document.write(y);
```

# Vergelijkingsoperatoren: geven true of false als resultaat (boolean)

Operator	Beschrijving	Voorbeeld
==	Is gelijk aan (waarde)	document.write(3 == "3") geeft true
===	Is gelijk aan (waarde en type)	document.write(3 === "3") geeft false
!=	Is verschillend van	day != "Monday"
>	Is groter dan	salary > 9000
>=	Is groter dan of gelijk aan	salary >=9000
<	Is kleiner dan	age < 18
<=	Is kleiner dan of gelijk aan	age <= 18

# VergelijkingsOperatoren voorbeelden

voorbeeld: `let x = 5;`

Operator	Beschrijving	Vergelijking	Resultaat
==	Is gelijk aan (waarde)	<code>x == 8</code>	false
		<code>x == 5</code>	true
		<code>x == "5"</code>	true
===	Is gelijk aan (waarde en type)	<code>x === 5</code>	true
		<code>x === "5"</code>	false
!=	Is niet gelijk aan	<code>x != 8</code>	true
!==	Is niet gelijk aan waarde of niet hezelfde type	<code>x !== 5</code>	false
		<code>x !== "5"</code>	true
		<code>x !== 8</code>	true
>	Is groter dan	<code>x &gt; 8</code>	false
<	Is kleiner dan	<code>x &lt; 8</code>	true
>=	Is groter dan of gelijk aan	<code>x &gt;= 8</code>	false
<=	Is kleiner dan of gelijk aan	<code>x &lt;= 8</code>	true

# Vorrangsregels (boven= hoogste prioriteit)

Operator	Beschrijving	Voorbeeld
( )	Expression grouping	(3 + 4)
++	Postfix Increment	i++
--	Postfix Decrement	i--
!	Logical not	!(x==y)
**	Exponentiation	10 ** 2
*	Multiplication	10 * 5
/	Division	10/5
%	Division Remainder	10 % 5
+	Addition	10 + 5
-	Subtraction	10-5
<	Less than	x < y
<=	Less than or equal	x <= y
>	Greater than	x > y
>=	Greater than or equal	x >= y
==	Equal	x == y
!=	Unequal	x != y
&&	Logical AND	x && y
	Logical OR	x    y

# Voorbeelden vergelijingsoperatoren

```
let getal1 = 10;
```

```
let getal2 = 5;
```

```
document.write(getal1 == getal2);
```

```
document.write(getal1 != getal2);
```

```
document.write(getal1 > getal2);
```

```
document.write(getal1 >= getal2);
```

```
document.write(getal1 < getal2);
```

```
document.write(getal1 <= getal2);
```

```
document.write(getal1 === getal2);
```

```
document.write(getal1 !== getal2);
```

# Booleans en Booleaanse operatoren

Booleaanse waarden (booleans) zijn ofwel **true** ofwel **false**  
voorbeelden van variabele declaratie voor booleans

```
let isOud = false;
```

```
let isSchuldig = true;
```

**Operatoren op booleans**

& & (AND)

| | (OR)

! (NOT)

== (gelijkheid)

!= (niet gelijk aan)

# Conditionele operatoren

- Worden gebruikt om een conditie te testen
- Het resultaat van een conditie is true of false

Operator	Description	Example
==	equal to	if (day == "Monday")
>	greater than	if (salary > 9000)
<	less than	if (age < 18)

Voorbeeld van gebruik:

```
let age = 16;
if (age < 18)
{
    text = "Too young!";
    alert(text); //shows the text in a popup-alert
                window
}
```

# Javascript Conditie bij programmatie

## Syntax

```
if (conditie) {  
    codeblok wordt uitgevoerd als conditie true is  
}
```

## Voorbeeld van gebruik:

```
let hour = 16;  
if (hour < 18) {  
    greeting = "Good day";  
}
```



# Oefeningen Javascript online

[https://www.w3schools.com/js/exercise\\_js.asp?filename=exercise\\_js\\_variables1](https://www.w3schools.com/js/exercise_js.asp?filename=exercise_js_variables1)

# Referenties

- [https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Text\\_formatting](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Text_formatting)
- <https://www.learn-js.org/>
- [https://www.w3schools.com/js/tryit.asp?filename=tryjs\\_const](https://www.w3schools.com/js/tryit.asp?filename=tryjs_const)