

// introduktion til

PROGRAMMERING/KODNING & IT-TANKEGANG (4/5)



Claus Brabrand

(brabrand@itu.dk)

(<http://www.itu.dk/people/brabrand/>)

Full Professor, Ph.D.
Head of Center for Computing Education Research
Department of Computer Science
 IT University of Copenhagen, Denmark

<http://studwww.itu.dk/~brabrand/ITU-PC/>

<https://tinyurl.com/itupc>

KURSUSPLAN

Fredag (28/02): 09:30 – 15:30
HTML & CSS

Fredag (07/03): 09:30 – 15:30
HTML Forms & Kodning...

Fredag (14/03): 09:30 – 15:30
Mere kodning...

Fredag (21/03): 09:30 – 15:30
Endnu mere kodning...

Fredag (28/03): 09:30 – 15:30
Databaser & Kommunikation & Web Services

KURSUSPLAN

Resumé

Konstant, Variabel, Operator
Tildeling, Inkrementering &
Bruger-Input, if

Operatorer

Man kan bruge mange forskellige operatorer:
...og de kan *nestes* lige så meget man vil:

```
document.write( 1 + (2 * 3) );
```

7

```
var x = 2;  
document.write( (((x + 1) * (x + 2)) / (x * x)) );
```

3

Beskrivelse af *lovlig syntaks (grammatik)*:

Exp	\rightarrow	Konstant	7
	\rightarrow	Variable	x
	\rightarrow	Exp '+' Exp	x + 1
	\rightarrow	Exp '-' Exp	x - 2
	\rightarrow	Exp '*' Exp	x * x
	\rightarrow	Exp '/' Exp	x / y
	\rightarrow	' (' Exp ')' '	(x + 1)

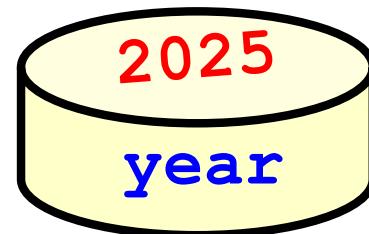
Inkrementering

```
var year = 2025;  
document.write("I år er det ");  
document.write(year);  
document.write("<br/>");  
  
year = year + 1; // inkrementering!  
document.write("Næste år er ");  
document.write(year);
```

I år er det 2025
Næste år er 2026

Variabler kan have **forskellige værdier** på
forskellige tidspunkter:

year = 2025;



year = year + 1;

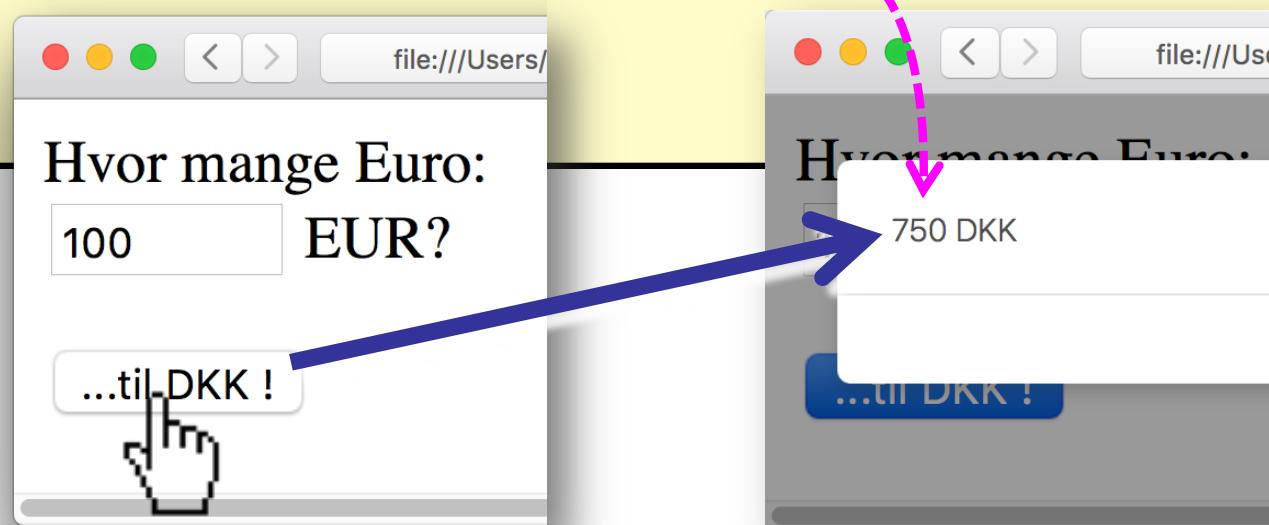


<https://www.w3schools.com/code/tryit.asp?filename=G3RZDDWD45WK>

Bruger-Input fra HTML

Valutaomregning:

```
<html>
  https://www.w3schools.com/code/tryit.asp?filename=FV29ZHGXQTF
  <body>
    Hvor mange Euro:<br/>
    <input id="euro"/> EUR?
    <p/>
    <button onclick="var e = document.getElementById('euro').value;
      var d = 7.5 * e;
      alert(d + ' DKK');
    ">
      ...til DKK !
    </button>
  </body>
</html>
```

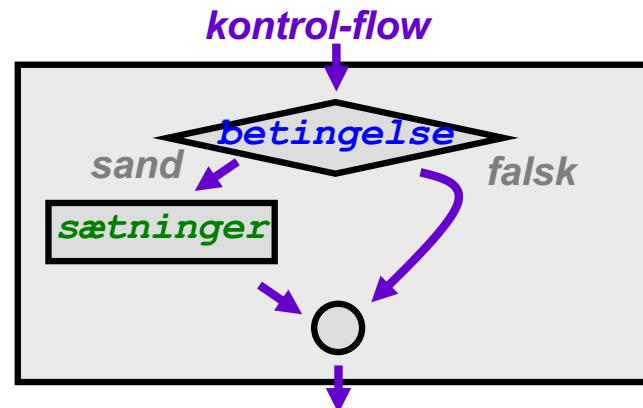


'if'-Sætning

Syntaks:

```
if ( betingelse ) {  
    sætninger  
}
```

Semantik:



Eksempel:

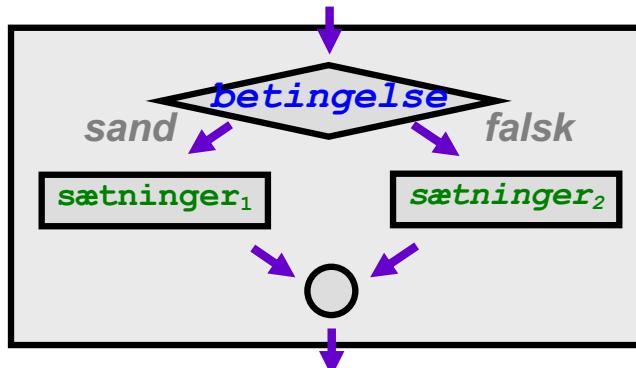
```
document.write("<h3>Er det torsdag?</h3>");  
if ( ugedag == 4 ) {  
    document.write("Ja, det er torsdag!");  
}  
document.write("<br/>The End.");
```

'if-else'-Sætning

Syntaks:

```
if ( betingelse ) {  
    sætninger1  
} else {  
    sætninger2  
}
```

Semantik:

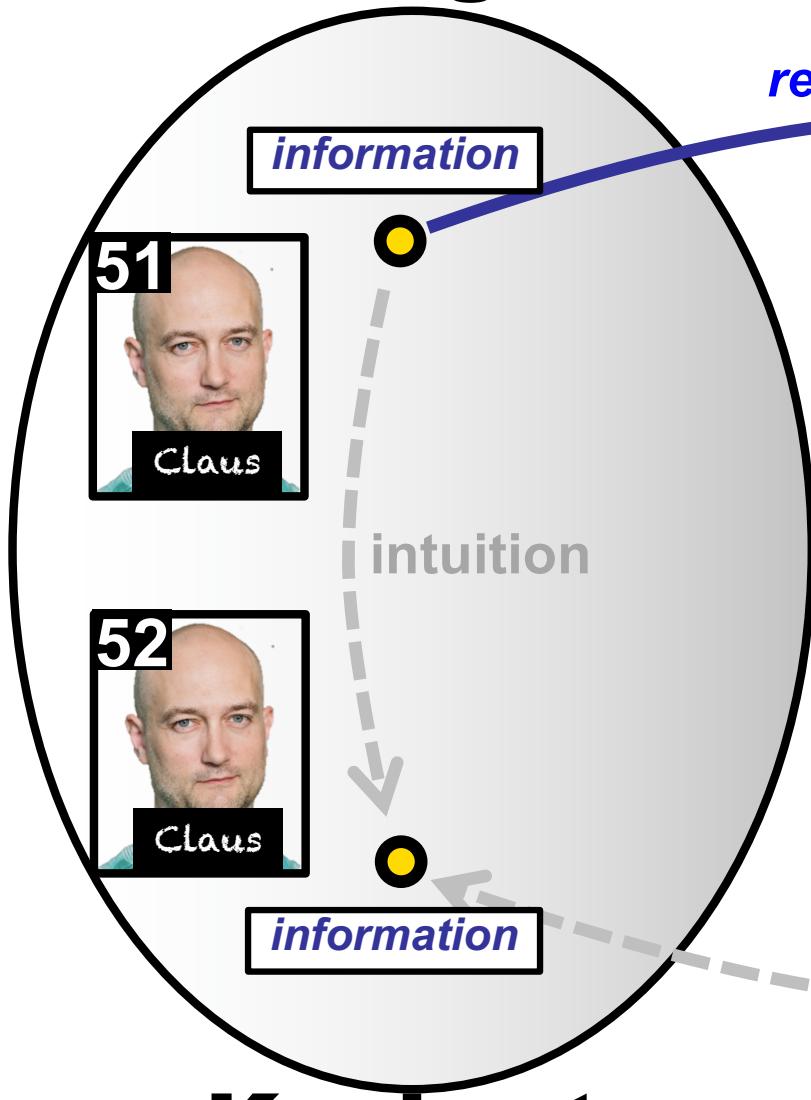


Eksempel:

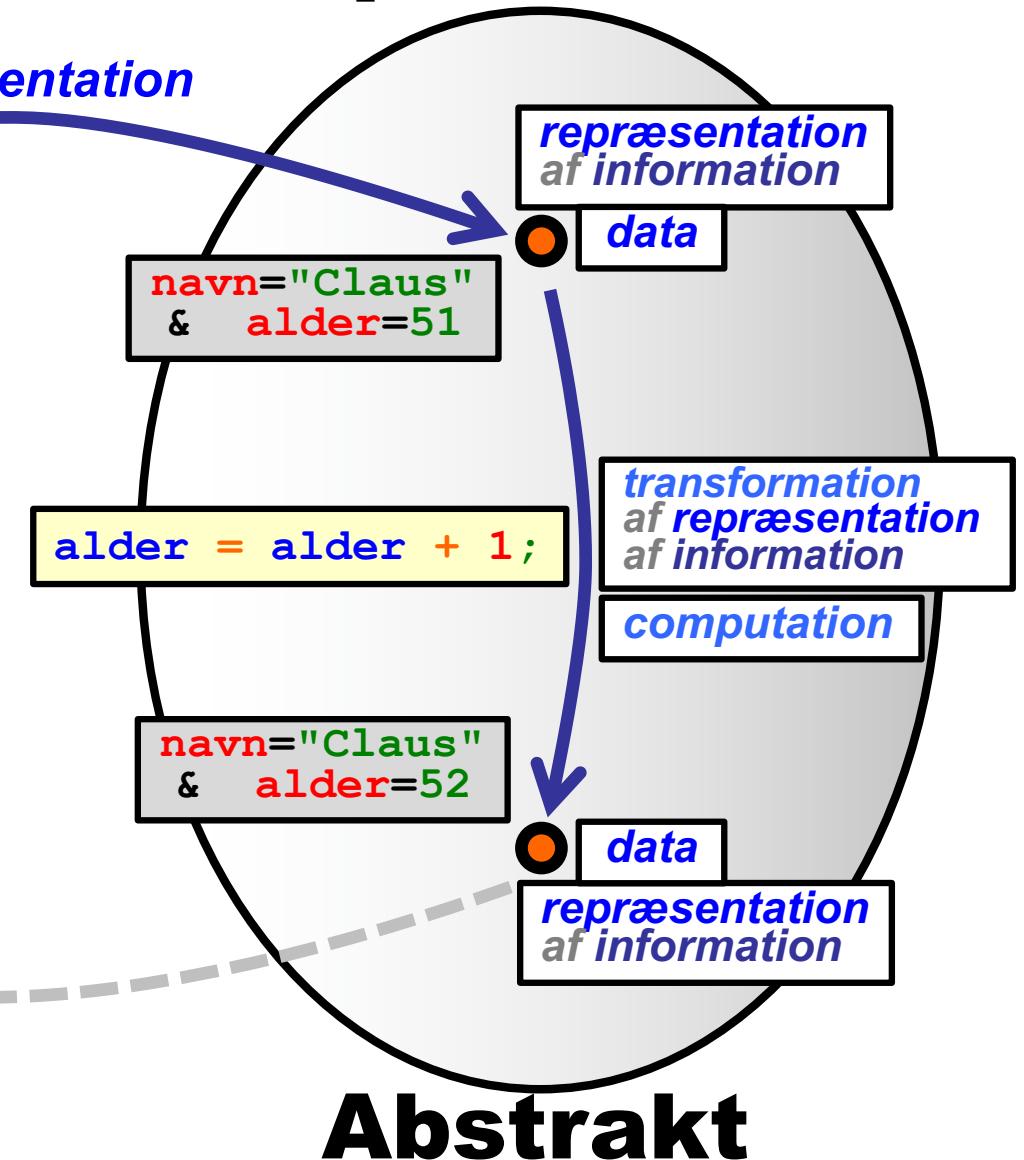
```
document.write("<h3>Er det fredag?</h3>");  
if ( ugedag == 5 ) {  
    document.write("Ja, det er fredag!");  
} else {  
    document.write("Nej, det er ik' fredag.");  
}  
document.write("<br/>The End.");
```

Informationsrepræsentationstransformation

Virkelighed



Computer-model



Konkret

Abstrakt

Transformation of Representation of Information

Nestede 'if'-sætninger

if-Sætninger: Simplifikation

```
if ( alder == 1 ) {  
    document.write("en");  
}  
else {  
    if ( alder == 2 ) {  
        document.write("to");  
    } else {  
        document.write("mange");  
    }  
}
```

III

```
if ( alder == 1 )  
    document.write("en");  
else {  
    if ( alder == 2 )  
        document.write("to");  
    else  
        document.write("mange");  
}
```

Grammatik (lovlig syntaks):

STM → if (EXP) STM else STM
→ { STMS }

Man behøver ikke at benytte '{' og '}',
hvis der kun er **én sætning** (én STM).

Hvis der er **fleste sætninger**, så skal
de "pakkes ind" i en '{' og '}'.
≡

```
if ( alder == 1 )  
    document.write("en");  
else  
    if ( alder == 2 )  
        document.write("to");  
    else  
        document.w' ("mange");
```

<https://www.w3schools.com/code/tryit.asp?filename=GORLVDVXSZ0I>

if-Sætninger: Indentering

Grammatik (lovlig syntaks):

```
if ( alder == 1 )
    d'w' ("en");
else
    if ( alder == 2 )
        d'w' ("to");
else
    én sætning
    if ( alder == 3 )
        d'w' ("tre");
else
    if ( alder == 4 )
        d'w' ("fire");
else
    d'w' ("mange");
```

STM → if (EXP) STM else STM
→ { STMS }

```
if ( alder == 1 )
    d'w' ("en");
else if ( alder == 2 )
    d'w' ("to");  én sætning
else if ( alder == 3 )
    d'w' ("tre");
else if ( alder == 4 )
    d'w' ("fire");
else
    d'w' ("mange");
```

d'w' skal være "document.write"

<https://www.w3schools.com/code/tryit.asp?filename=FS77F8FYJGOB>
<https://www.w3schools.com/code/tryit.asp?filename=FXJB4PX5IV6Z>

BMI

med (if) forgreninger



BMI Beregner 1.0



```
<html>
  <head>
    <meta charset="UTF-8"/>
  </head>
  <body>
    <script>
      /* === Body Mass Index (BMI) beregner === */
      var h = 184; // højde
      var v = 84; // vægt
      document.write("Højde: " + h + " cm.");
      document.write("<br/>");
      document.write("Vægt: " + v + " kg.");
      document.write("<p/>");

      /* Programmér dette (evt afrundet til 2 decimaler):
       Hint-1: 'bmi' er vægten divideret med højden i meter i anden'
       Hint-2: 'x i anden' er jo bare lig med: 'x gange x'. */
    </script>
  </body>
</html>
```

Højde: 184 cm.

Vægt: 84 kg.

Din BMI er **24.8109640831758**

OPGAVE

<https://www.w3schools.com/code/tryit.asp?filename=FS4TZDDLE4YN>



BMI Beregner 1.0 Løsning

```
<html>
  <head>
    <meta charset="UTF-8"/>
  </head>
  <body>
    <script>
      /* === Body Mass Index (BMI) beregner === */
      var h = 184; // højde
      var v = 84; // vægt
      document.write("Højde: " + h + " cm.");
      document.write("<br/>");
      document.write("Vægt: " + v + " kg.");
      document.write("<p/>");

      var bmi = v / ((h / 100) * (h / 100)) ;
      document.write("Din BMI er <b>" + bmi + "</b>");

    </script>
  </body>
</html>
```

Højde: 184 cm.

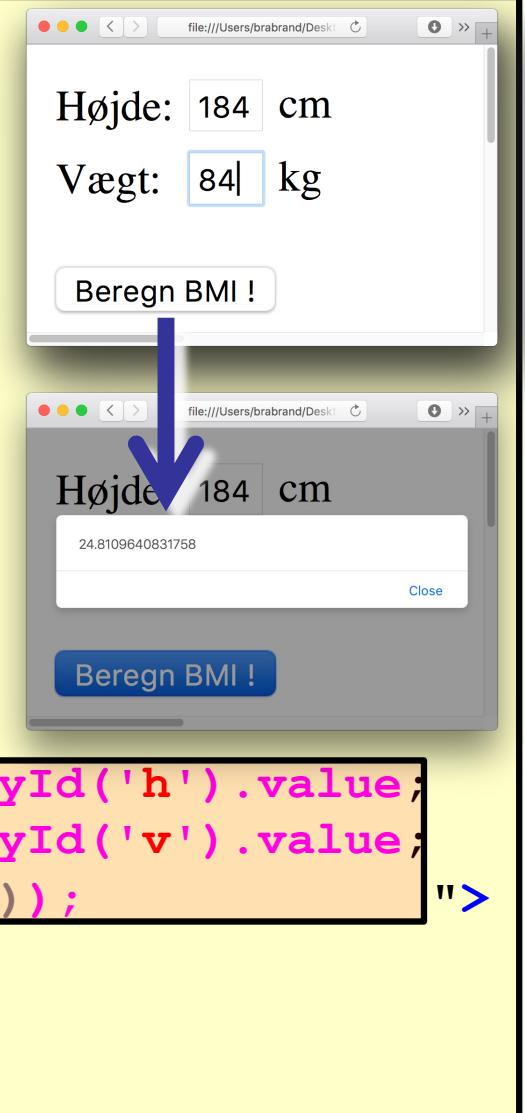
Vægt: 84 kg.

Din BMI er **24.8109640831758**

<https://www.w3schools.com/code/tryit.asp?filename=FS4TZDDLF4YN>

Alternativ med <input> felter

```
<html><head><meta charset="UTF-8"/></head>
<body>
  <table>
    <tr>
      <td>Højde:</td>
      <td><input id="h" size="3" /> cm</td>
    </tr>
    <tr>
      <td>Vægt:</td>
      <td><input id="v" size="3" /> kg</td>
    </tr>
  </table>
  <p>
    <button onclick="var h = document.getElementById('h').value;
                    var v = document.getElementById('v').value;
                    alert(v / ((h/100) * (h/100)))">
      Beregn BMI !
    </button>
  </p>
</body>
</html>
```



<https://www.w3schools.com/code/tryit.asp?filename=FV5M0LUN4NYR>

BMI Beregner 2.0 (★)

Udvid BMI-programmet fra en af forrige slide(s), så der testes om vægten er...:

for lidt

// bmi mindre end 20

for normalt

// bmi mellem 20 og 25

for meget

// bmi mere end 25

Eksempel output:

Højde: 184 cm.

Vægt: 84 kg.

Din BMI er **24.8109640831758** som er **normalt**

Højde: 184 cm.

Vægt: 85 kg.

Din BMI er **25.10633270321361** som er **for meget**



OPGAVE



BMI Beregner 2.0 Løsning

```
<html><head><meta charset="UTF-8" /></head><body><script>
    /* === Body Mass Index (BMI) beregner === */
    var h = 184; // højde
    var v = 84; // vægt
    document.write("Højde: " + h + " cm.");
    document.write("<br/>");
    document.write("Vægt: " + v + " kg.");
    document.write("<p/>");

    var bmi = v / ((h / 100) * (h / 100)) ;

    document.write("Din BMI er <b>" + bmi + "</b> som er ");

    if ( bmi < 20 ) {
        document.write("<font color='orange'>for lidt</font>");
    } else if ( bmi > 25 ) {
        document.write("<font color='red'>for meget</font>");
    } else {
        document.write("<font color='green'>for normalt</font>");
    }
</script></body></html>
```

<https://www.w3schools.com/code/tryit.asp?filename=G3XE4EXPPVVT>



if-Sætninger...



Bryg selv videre på følgende program...:

```
var alder = prompt("Hvad er din alder?");  
  
if ( alder <= 2 ) {  
    document.write("Du er en baby!");  
} else if ( alder <= 12 ) {  
    document.write("Du er et barn!");  
}  
// ...lav videre på programmet...
```

<https://www.w3schools.com/code/tryit.asp?filename=FS2SMM4T6SSI>

0-2	=>	baby
3-12	=>	barn
13-19	=>	teenager
20-35	=>	ung
36-66	=>	voksen
67+	=>	pensionist



OPGAVE



Sandhedsværdier **(Boolean Values)**

sandt / falsk
(true / false)

Sandhedsværdier

Der findes *blot to sandhedsværdier*:

```
true      // konstanten 'sandt'  
false     // konstanten 'falsk'
```

Eksempler:

```
if ( true ) {  
    document.write("Sandt");  
} else {  
    document.write("Falsk");  
}
```

Sandt

```
if ( false ) {  
    document.write("Sandt");  
} else {  
    document.write("Falsk");  
}
```

Falsk

Sandhedsudtryk

Sammenlignings-operatorer:

<code>==</code> (er lig med)	(1 == 1)
<code>!=</code> (er forskellig fra)	(2 != 3)
<code><</code> (er mindre end)	(4 < 5)
<code>></code> (er større end)	(7 > 6)
<code><=</code> (er mindre end eller lig med)	(8 <= 9)
<code>>=</code> (er større end eller lig med)	(0 >= 0)

Eksempel:

```
var x = 1 ;
var y = 2 ;
document.write("Resultatet er: ");
document.write( ( x > y ) );
```

Resultatet er: **false**

Logiske operatorer:

! (ikke)

! (alder >= 18)

&& (og) (year >= 1990) && (year <= 1999)

|| (eller)

(alder < 18) || (alder >= 67)

Nogle gange skriver man (i matematik)....:

20 ≤ bmi ≤ 25

...hvilket skrives som følgende i JavaScript:

(20 <= bmi) && (bmi <= 25)

'Og' og/eller 'Eller'

'Og' (konjunktion):

```
if ( alder >= 13 ) {  
    if ( alder <= 19 ) {  
        document.write("Teenager");  
    }  
}
```

≡

```
if ( alder >= 13 && alder <= 19 ) {  
    document.write("Teenager");  
}
```

<https://www.w3schools.com/code/tryit.asp?filename=G3XE6NLBY4Q2>

'Eller' (disjunktion):

```
if ( alder <= 10 ) {  
    document.write("halv pris!");  
}  
  
if ( alder >= 67 ) {  
    document.write("halv pris!");  
}
```

≡

```
if ( alder <= 10 || alder >= 67 ) {  
    document.write("halv pris!");  
}
```

<https://www.w3schools.com/code/tryit.asp?filename=G3XE8A9NDBOV>

Negation

Negation:

```
if ( is_female ) {  
    document.write("hun");  
} else {  
    document.write("han");  
}
```

≡

```
if ( ! is_female ) {  
    document.write("han");  
} else {  
    document.write("hun");  
}
```

negation!

ombyt

<https://www.w3schools.com/code/tryit.asp?filename=G0JMKBF4G748>

while

Iteration

Iteration: 'while'

Man kan bruge en **while-sætning** til at eksekvere et **stykke kode** igen og igen og...

indtil en **betingelse** bliver **falsk**:

```
while ( betingelse ) {  
    sætninger  
}
```

```
document.write("JavaScript er ikke det samme som Java!");  
document.write("JavaScript er ikke det samme som Java!");
```

JavaScript er ikke det samme som Java!
JavaScript er ikke det samme som Java!

III

```
var i = 0;                                // initialisering  
while ( i < 5 ) {                          // betingelse  
    document.write("JavaScript er ikke det samme som Java!"); // krop  
    i = i + 1;                                // inkrementering  
}
```

<https://www.w3schools.com/code/tryit.asp?filename=G3XEEQYIP31H>

Iteration: 'while'

Man kan bruge en **while-sætning** til at eksekvere et **stykke kode** igen og igen og...

indtil en **betingelse** bliver **falsk**:

```
while ( betingelse ) {  
    sætning  
}
```

```
document.write("<li>0</li>");  
document.write("<li>1</li>");  
document.write("<li>2</li>");  
document.write("<li>3</li>");  
document.write("<li>4</li>");  
document.write("<li>5</li>");  
document.write("<li>6</li>");  
document.write("<li>7</li>");  
document.write("<li>8</li>");  
document.write("<li>9</li>");
```

=

```
var i = 0;  
while ( i < 10 ) {  
    document.write("<li>" + i + "</li>");  
    i = i + 1;  
}
```

- 0
- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9

<https://www.w3schools.com/code/tryit.asp?filename=G3XE1YH8Y8>

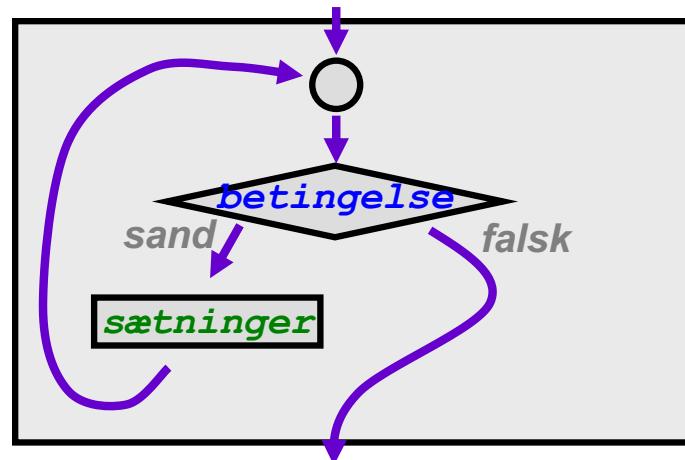
Q: Hvad ville der ske, hvis vi lavede om til 0-til-99?

Iteration: 'while'

Syntaks:

```
while ( betingelse ) {  
    sætninger  
}
```

Semantik:



Eksempel:

```
var i = 0;  
while ( i < 10 ) {  
    document.write("<li>" + i + "</li>");  
    i = i + 1;  
}
```

- 0
- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9



while-Sætninger...



Modificér følgende program, således at det beregner og udskriver **syv-tabellen**:

```
<ul>
  <script>
    var i = 0;
    while ( i < 10 ) {
      document.write("<li>" + i + "</li>");
      i = i + 1;
    }
  </script>
</ul>
```

- 7
- 14
- 21
- 28
- 35
- 42
- 49
- 56
- 63
- 70

<https://www.w3schools.com/code/tryit.asp?filename=G3XE1YH8Y8>



OPGAVE



Tre Forskellige Løsninger

```
<html><body><ul><script>
var i = 7;
while ( i <= 70 ) {
    document.write("<li>" + i + "</li>");
    i = i + 7;
}

var j = 1;
while ( j <= 10 ) {
    document.write("<li>" + ( j * 7 ) + "</li>");
    j = j + 1;
}

var k = 0;
while ( k < 10 ) {
    document.write("<li>" + ( (k+1) * 7 ) + "</li>");
    k = k + 1;
}
</script></ul></body></html>
```

<https://www.w3schools.com/code/tryit.asp?filename=G3RZG5KPGJBP>



'while' Quizzzzzz...



```
var i = 10;           // Program 0:  
while ( i > 0 ) {  
    document.write(i);  
    i = i - 1;  
}
```

→ Resultat?

```
var i = 1;           // Program 1:  
while ( i <= 10 ) {  
    document.write(i);  
    i = i + 2;  
}
```

→ Resultat?

```
var i = 1;           // Program 2:  
while ( i < 100 ) {  
    document.write(i);  
    i = i * 2;  
}
```

→ Resultat?

```
var i = 1;           // Program 3:  
while ( i < 42 ) {  
    document.write(i);  
    i = i * i;  
}
```

→ Resultat?



OPGAVE





'while' Quizzzzzz...



```
// Program 4:  
var i = [REDACTED];  
while ([REDACTED]) {  
    document.write(i);  
    [REDACTED];  
}
```

1
4
7
10
13
16



```
// Program 5:  
var i = [REDACTED];  
while ([REDACTED]) {  
    document.write(i);  
    [REDACTED];  
}
```

1
10
100
1000
10000



```
// Program 6:  
var i = [REDACTED];  
while ([REDACTED]) {  
    document.write(i);  
    [REDACTED];  
}
```

64
32
16
8
4
2

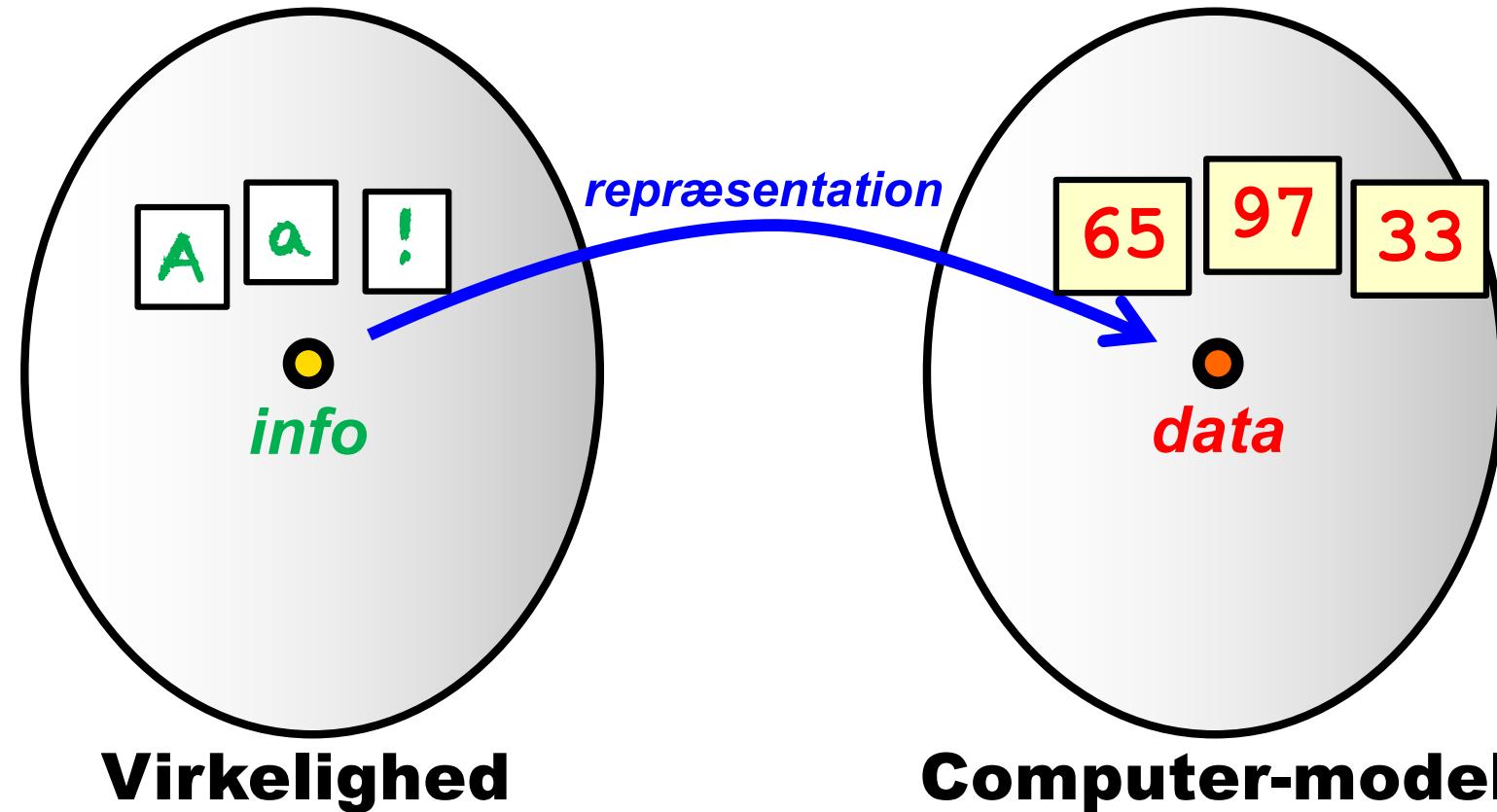


OPGAVE



Informationsrepræsentation

Informationsrepræsentation af tegn (characters):



Alle **tegn** (characters) har fået et (ASCII) **nummer**, som er det der rent faktisk gemmes på computeren

ASCII Tabellen

ASCII printable characters (character code 32-126):

<u>TAL</u>	<u>SYM</u>
#32	--> ' '
#33	--> ' !'
#34	--> ' "'
#35	--> ' #'
#36	--> ' \$'
#37	--> ' %'
#38	--> ' &'
#39	--> ' ' '
#40	--> ' ('
#41	--> ') '
#42	--> ' *'
#43	--> ' +'
#44	--> ' , '
#45	--> ' -'
#46	--> ' . '
#47	--> ' / '

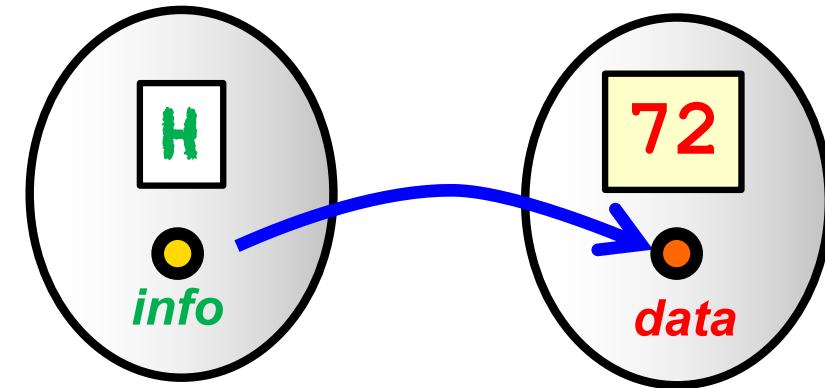
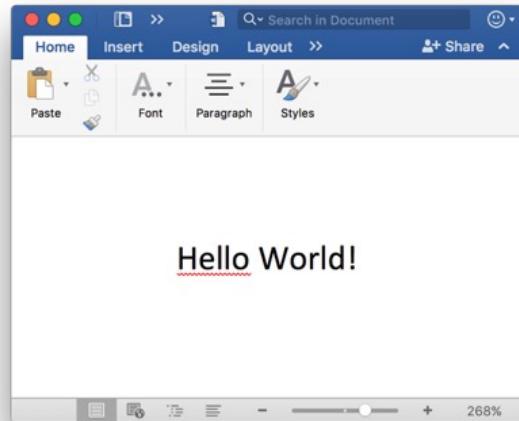
#48	--> ' 0 '
#49	--> ' 1 '
#50	--> ' 2 '
#51	--> ' 3 '
#52	--> ' 4 '
#53	--> ' 5 '
#54	--> ' 6 '
#55	--> ' 7 '
#56	--> ' 8 '
#57	--> ' 9 '
#58	--> ' : '
#59	--> ' ; '
#60	--> ' < '
#61	--> ' = '
#62	--> ' > '
#63	--> ' ? '
#64	--> ' @ '

#65	--> ' A '
#66	--> ' B '
#67	--> ' C '
#68	--> ' D '
#69	--> ' E '
#70	--> ' F '
#71	--> ' G '
#72	--> ' H '
#73	--> ' I '
#74	--> ' J '
#75	--> ' K '
#76	--> ' L '
#77	--> ' M '
#78	--> ' N '
#79	--> ' O '
#80	--> ' P '
#81	--> ' Q '

#82	--> ' R '
#83	--> ' S '
#84	--> ' T '
#85	--> ' U '
#86	--> ' V '
#87	--> ' W '
#88	--> ' X '
#89	--> ' Y '
#90	--> ' Z '
	:
#97	--> ' a '
#98	--> ' b '
#99	--> ' c '
#100	--> ' d '
	:
#125	--> ' } '
#126	--> ' ~ '

Informationsrepræsentation

Husk "Hello World!"-dokumentet fra Word:



Informationen "Hello World!":

```
| H | e | l | l | o |   | W | o | r | l | d | ! |
```

Repræsenteres som **ASCII værdier**:

```
072 101 108 108 111 032 087 111 114 108 100 033
```

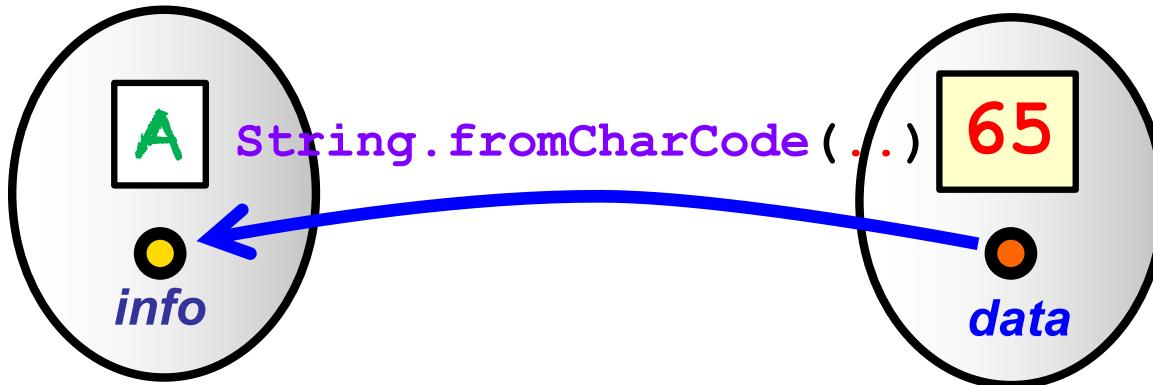
Som gemmes på computeren i **binær format**:

```
01001000 01100101 01101100 01101100 01101111 ....
```

Representation of Information

Opslag i ASCII-tabellen

Man kan "slå op i" ASCII-tabellen ved:



```
var tal = 65;
var symbol = String.fromCharCode(tal);
document.write("#<font color='red'>" + tal + "</font>") ;
document.write(" = ");
document.write("'"<font color='green'>" + symbol + "</font>'") ;
```

Mere info (w3schools):

#65 = 'A'

https://www.w3schools.com/charsets/ref_html_ascii.asp



ASCII Tabe [1/2]



Udskriv tallene fra #32 (første synlige ASCII-tegn) til #126 (sidste universelle ASCII-tegn)

forventet output



Hint:

Brug en **while**-løkke til at tælle op fra **tal = 32**, så længe **tal ≤ 126**

#32	=	' '
#33	=	' '
#34	=	' '
#35	=	' '
#36	=	' '
#37	=	' '
#38	=	' '
#39	=	' '
#40	=	' '
#41	=	' '
#42	=	' '
#43	=	' '
#44	=	' '
#45	=	' '
#46	=	' '
#47	=	' '
:		
#126	=	' '



OPGAVE



ASCITabe [2/2] (★)

Udskriv nu tabellens tegn fra #32
(som er et mellemrum) ' ' til #126
(som er en tilde) ' ~ '

forventet output →

Hint:

Brug en **while**-løkke til at tælle op
fra **tal** = 32, så længe **tal** ≤ 126

Hint: opslag i tabellen foretages således:

```
var symbol = String.fromCharCode(tal);
```

#32	=	'	'
#33	=	'!	'
#34	=	'"	'
#35	=	'#	'
#36	=	'\$	'
#37	=	'%	'
#38	=	'&	'
#39	=	''	'
#40	=	'(')'
#41	=	')'	
#42	=	'*'	
#43	=	'+'	
#44	=	','	
#45	=	'-'	
#46	=	'.'	
#47	=	'/'	
:			
#126	=	'~'	



OPGAVE



for

Iteration

Iteration: 'for'

En **for-løkke** gør nogenlunde det samme som en **while-løkke**, men koden kan ofte skrives mere kompakt:

Java er ikke det samme som JavaScript!
Java er ikke det samme som JavaScript!

initialisering

```
var i = 0;  
while ( i < 10 ) {  
    document.write("Java er ikke det samme som JavaScript!<br/>");  
    i++;  
}
```

betingelse

inkrementering

VS

initialisering betingelse inkrementering

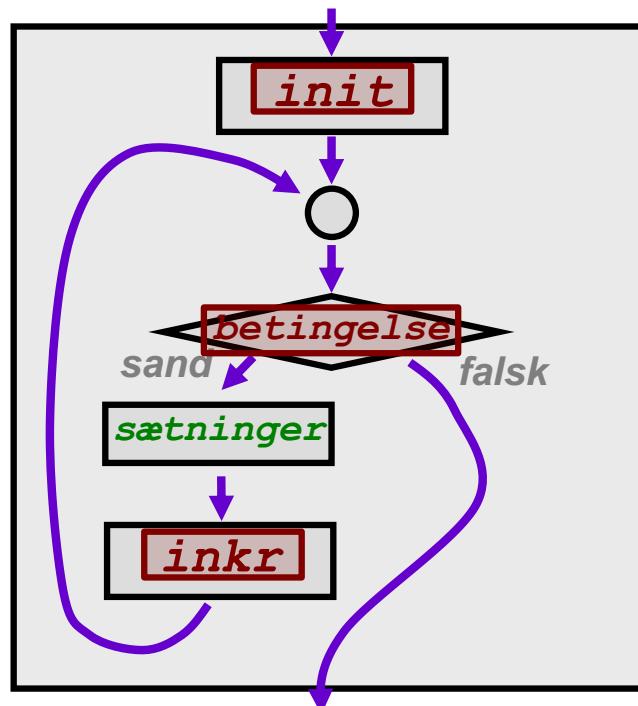
```
for ( var i = 0 ; i < 10 ; i++ ) {  
    document.write("Java er ikke det samme som JavaScript!<br/>");  
}
```

Iteration: 'for'

Syntaks:

```
for ( init ; betingelse ; inkr ) {  
    sætninger  
}
```

Semantik:



Eksempel:

```
for ( var i = 0 ; i < 10 ; i = i + 1 ) {  
    document.write("<li>" + i + "</li>");  
}
```

- 0
- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9

EXERCISE: for loop quizzzz

```
for ( var i = 10 ; i >= 0 ; i = i - 3 ) {  
    document.write(i);  
}
```

→ Result?

```
for ( var i = 1 ; i < 10 ; i = i + 4 ) {  
    document.write(i);  
}
```

→ Result?

```
for ( var i = 0 ; i < 10 ; i = i * i ) {  
    document.write(i);  
}
```

→ Result?

EXERCISE: for loop quizzzz

```
for ( [ ] ) {  
    document.write(i);  
}
```

0
3
6
9
12
15



```
for ( [ ] ) {  
    document.write(i);  
}
```

1
10
100
1000
10000



```
for ( [ ] ) {  
    document.write(i);  
}
```

64
32
16
8
4
2



'for' vs 'while'

for loops er mere benyttede end **while** loops:

Højere abstraktionsniveau:

Init, **Cond**, **Incr** hører begrebsmæssigt sammen

Mere kompakt notation

Lettere at læse (når man vænner sig til det)

I while loops kan man fx glemme at tælle op

While bruges typisk når man skal blive ved indtil et eller andet er opfyldt (fx man har fundet det man leder efter)

For bruges typisk hvis man skal gentage et eller andet et bestemt antal gange.

Strenge

Operationer på tekst-strenge

Operationer på Strenge

Operationer på tekst-strenge:

<code>s.length</code>	// længen af <code>s</code>
<code>s.charAt(0)</code>	// første tegn i <code>s</code>
<code>s.charAt(s.length-1)</code>	// sidste tegn i <code>s</code>

```
<html>
  <body>
    <code>
      <script>
        var s = "once upon a time, there was a language called javascript.";
        document.write("Længde af s er: " + s.length);
        document.write("<br/>");
        document.write("Første tegn er: '" + s.charAt(0) + "'");
        document.write("<br/>");
        document.write("Sidste tegn er: '" + s.charAt(s.length-1) + "'");
      </script>
    </code>
  </body>
</html>
```

<https://www.w3schools.com/code/tryit.asp?filename=G3XF9XZBHECD>

Længde af s er: 57
Første tegn er: 'o'
Sidste tegn er: .'

Gennemløb af Strenge

Operationer på tekst-strenge:

<code>s.length</code>	// længen af s
<code>s.charAt(0)</code>	// første tegn i s
<code>s.charAt(s.length - 1)</code>	// sidste tegn i s

```
<html>
  <body>
    <code>
      <script> 012
        ↓↓
        var s = "Strenge gennemløbes og udskrives ét tegn ad gangen.";
        for ( var index = 0 ; index < s.length ; index++ ) {
          var tegn = s.charAt(index); // find index'te tegn i strengen
          document.write(tegn);     // udskriv tegn'et
        }
      </script>
    </code>
  </body>
</html>
```

<https://www.w3schools.com/code/tryit.asp?filename=G3XFA4ZU68F5>

Strenge gennemløbes og udskrives ét tegn ad gangen.

Antal a'er i en Streng (★)

Hvad gør følgende program?

```
<html>
  <body>
    <code>      012
      <script>    ↓↓
      var s = "once upon a time, there was a language called javascript.";
      for ( var index = 0 ; index < s.length ; index++ ) {
        var tegn = s.charAt(index);
        document.write(tegn); // <-- erstat denne linje med anden kode!
      }
    </script>
  </code>
</body>
</html>
```

<https://www.w3schools.com/code/tryit.asp?filename=G3XFAO1HMVDI>

Modificér programmet så det i stedet finder ud af **hvor mange a'er** der er i følgende streng:

"once upon a time, there was a language called javascript."



OPGAVE



#vokaler i en Streng (★)

```
<html>
  <body>
    <code>      012
      <script> ↓↓↓
      var s = "once upon a time, there was a language called javascript.";
      for ( var index = 0 ; index < s.length ; index++ ) {
        var c = s.charAt(index);
        document.write(c);
      }
    </script>
  </code>
</body>
</html>
```

<https://www.w3schools.com/code/tryit.asp?filename=G3XFC4ATBV7Z>

s.length-1
↓

Generalisér nu dit program således at det beregner antallet af **vokaler** i strengen:

"once upon a time, there was a language called javascript."



OPGAVE



Syntaks

JavaScript

Grammatik for JavaScript

EXP expression (udtryk)	→ KONSTANT → VAR → EXP + EXP	7 x x + 1
STM statement (sætning)	→ VAR = EXP ; → var VAR = EXP ; → if (EXP) STM → if (EXP) STM else STM → while (EXP) STM → for (EXP ; EXP ; EXP) STM → { STMS } → NAVN (ARGS) ;	x=x+1 ;

Grammatik for JavaScript

EXP	→	KON	7
expression	→	VAR	x
(udtryk)	→	EXP + EXP	x + 1
	→	EXP * EXP	2 * 2
	→	EXP - EXP	y - 1
	→	EXP / EXP	1 / 3
	→	EXP == EXP	x == 0
	→	EXP != EXP	x != 1
	→	EXP > EXP	x > 0
	→	EXP < EXP	y < 10
	→	EXP >= EXP	x >= y
	→	EXP <= EXP	0 <= x
	→	EXP && EXP	a && b
	→	EXP EXP	a b
	→	EXP % EXP	x % 2
	→	! EXP	! true
	→	(EXP)	(x>0)

Grammatik for JavaScript

Argumenter (**ARGS**) er bare udtryk (**EXP**) med komma i mellem:

ARGS arguments (argumenter)	→	/* no arguments */
	→	EXPS
EXPS expressions (udtryk)	→	EXP
	→	EXP , EXPS
		// one expression
		// more expressions
STMS statements (sætninger)	→	/* empty statement */
	→	STM STMS

IT-Tankegang

(Computational Thinking)

Programmering derhjemme



Programming at Home

Udtrykskraft (Ekspressibilitet)

Følgende (del af JavaScript) er **Turing-Komplet**:

Alt der kan programmes kan skrives i dette sprog^{*}:

EXP	→	KONSTANT
	→	VAR
	→	EXP + 1
	→	EXP == 0
 STM	→	VAR = EXP ;
	→	if (EXP) STM
	→	while (EXP) STM

Man ville kunne lave en **oversætter** (der kan oversætte)
fra alle computer-sprog (inkl hele JavaScript eller Java)
til denne simple del-mængde !

Expressibility

^{*}) det vil naturligvis være bøvet, men det vil være muligt

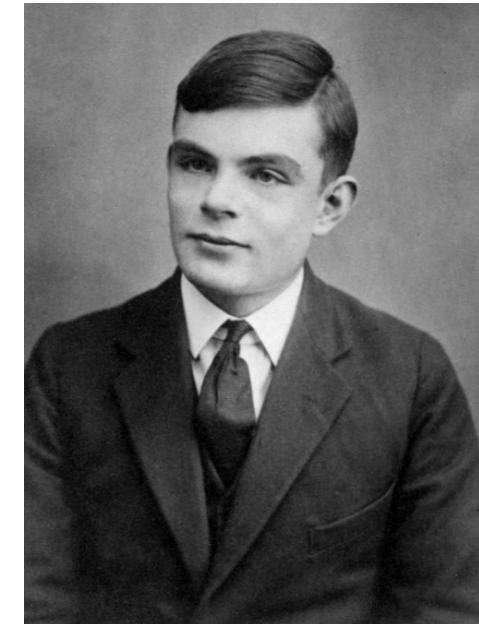
Alan Turing

Alan Turing (1912 – 1954)

Britisk datalog && computer-pionér

Opfandt "**Turing-maskinen**" som formaliserer ***alt der kan beregnes!***

En **Turing-maskine** kan eksekvere alt der overhovedet kan programmeres:



<https://www.youtube.com/watch?v=FTSAiF9AHN4>

*Turingmaskine
bygget i LEGO*

Nåh ja, ...og så byggede han også liiige en maskine under WWII, der dekrypterede nazisternes **Enigma** (som – ifl kilder – forkortede krigen med et par år)

Se evt filmen "The Imitation Game (2014)":

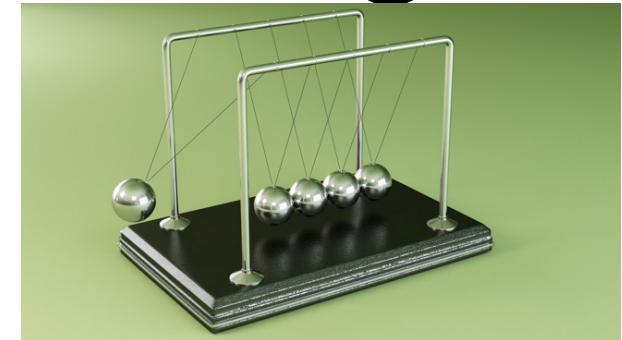
<https://www.imdb.com/title/tt2084970/>

Alan Turing

Fundamentale Begrænsninger

"Evighedsmaskinen":

Som aldrig ville kunne laves



Den strider mod fundamentale naturlove

(Termodynamikkens anden lov)

Tilsvarende findes der nogle **fundamentale begrænsninger** på **hvad en computer kan** (og nogensinde ville kunne) **løse / beregne!**

- 1) *Umulige problemer*
- 2) *Upraktiske problemer*
- 3) *Upræcise problemer*

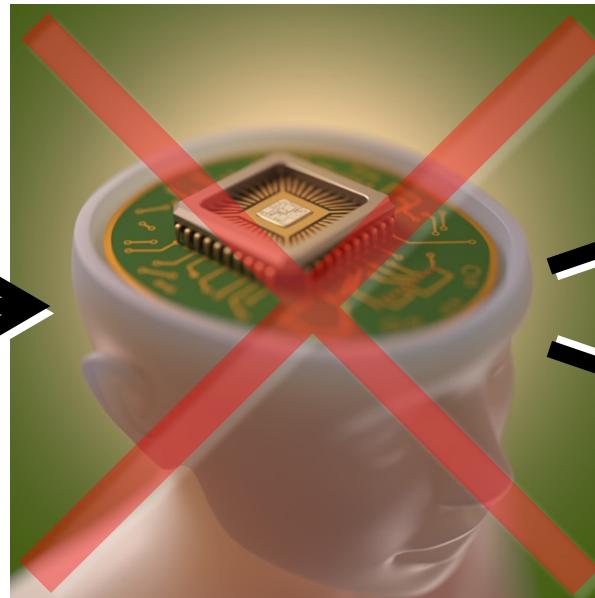
Fundamental Limitations

1) Umulige Problemer

PROBLEM: *The-Halting-Problem*

Afgøre om et givet input program altid stopper (eller ej)?

```
var x = prompt();
while ( x != 1 ) {
    d'w'(x + "<br/>");
    if ( (x%2) == 0 ) {
        // 'x' er lige:
        x = x / 2;
    } else {
        // 'x' er ulige:
        x = 3*x + 1;
    }
}
document.write("stop");
```



stopper
altid

stopper
ikke altid

Det er *umuligt* at nogensinde at skrive et program der løser opgaven (selv med kvante-computere, alien-teknologi, ...) !

Faktisk vides det ikke om programmet til venstre *altid* stopper !

https://en.wikipedia.org/wiki/Collatz_conjecture

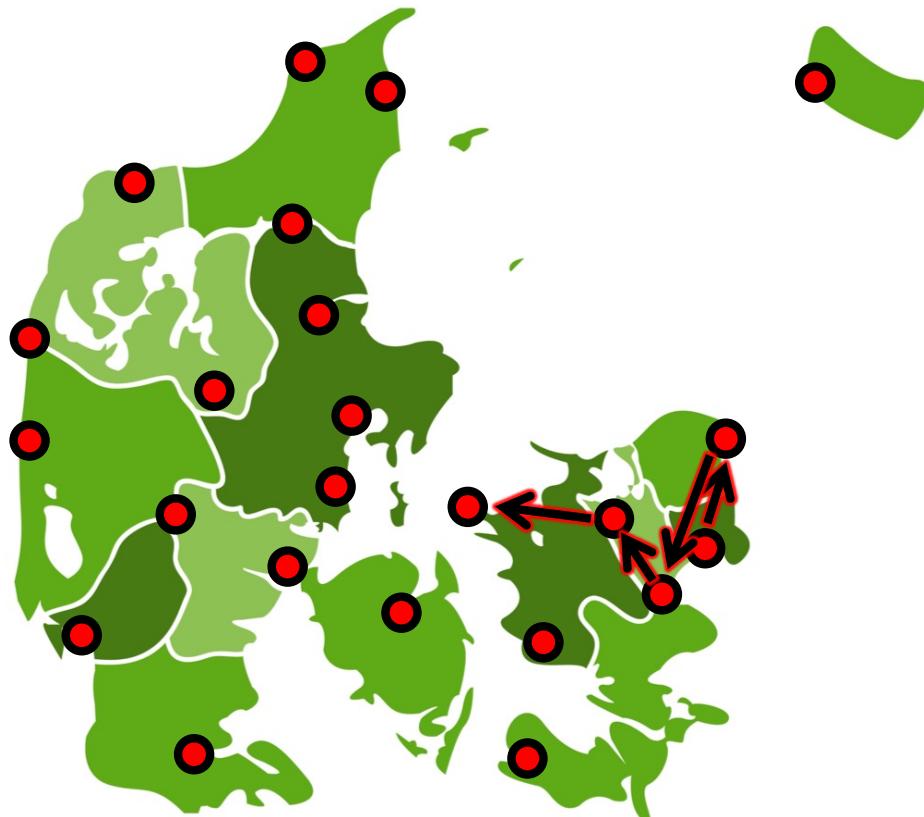
<https://www.w3schools.com/code/tryit.asp?filename=FS7H8HXV9F5W>

1) Undecidable Problems

2) Upraktiske Problemer

PROBLEM: *The-Travelling-Salesman*

Hvilken rute (rækkefølge af byer) er kortest?



#atomer i hele universet < 2^{320}

"Kombinatorisk Ekspllosion":

- Givet N input (fx N byer)
- Checke alle 2^N muligheder

$$2^1 = 2$$

$$2^2 = 4$$

$$2^3 = 8$$

$$2^4 = 16$$

$$2^5 = 32$$

$$2^6 = 64$$

$$2^7 = 128$$

$$2^8 = 256$$

$$2^9 = 512$$

$$2^{10} = 1.024 \quad \text{tusinde}$$

$$2^{20} = 1.048.576 \quad \text{million}$$

$$2^{30} = 1.073.741.824 \quad \text{milliard}$$

$$2^{40} = 1.099.511.627.776 \quad \text{trillion}$$

2) Intractable Problems

Kombinatorisk Ekspllosion

Eksponentiel vækst **vokser meget hurtigt!**

Størrelse af input (N) :	Beregning (0,001 s/mulighed)
$2^1 = 2$	0,002 sek
$2^2 = 4$	0,004 sek
$2^3 = 8$	0,008 sek
$2^4 = 16$	0,016 sek
$2^5 = 32$	0,032 sek
$2^6 = 64$	0,064 sek
$2^7 = 128$	0,128 sek
$2^8 = 256$	0,256 sek
$2^9 = 512$	0,512 sek
$2^{10} = 1.024$	1,024 sek = 1 sek
$2^{20} = 1.048.576$	1.024,576 sek = 17 min
$2^{30} = 1.073.741.824$	1.073.741,824 sek = 12 dage
$2^{40} = 1.099.511.627.776$	1.099.511.627,776 sek = 36 år
$2^{50} = 1.000.000.000.000.000$	1.000.000.000.000,000 sek = 35.700 år

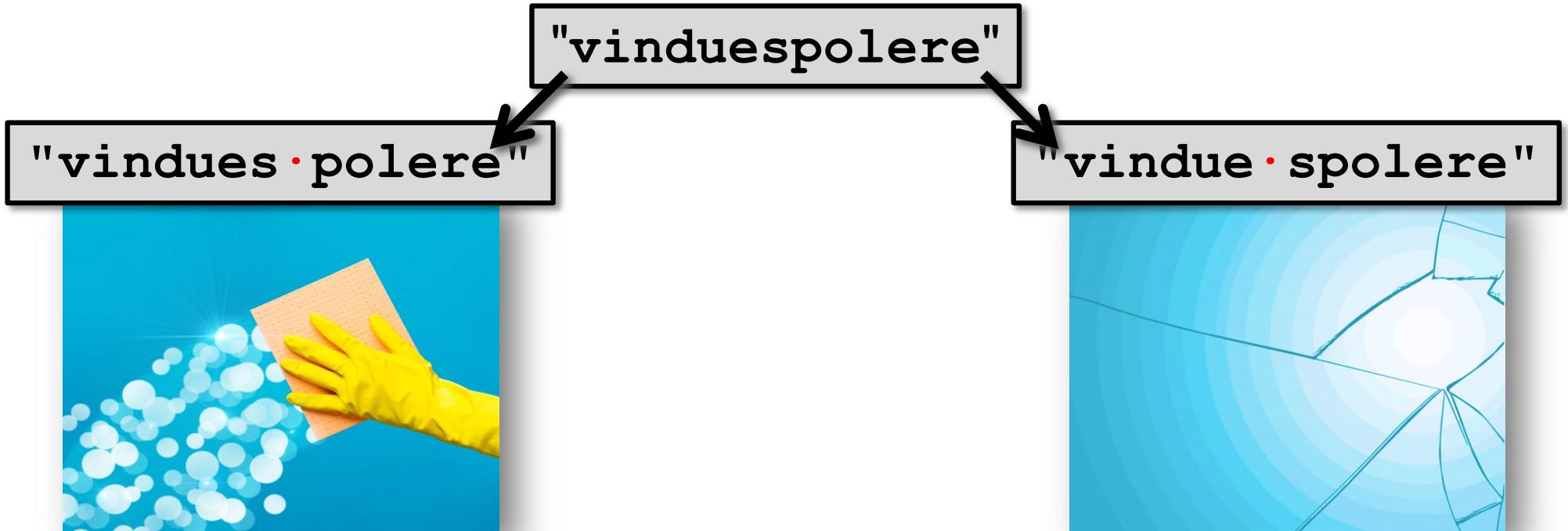
Kryptologi virker netop pga dette; man kan vælge X-bit nøgler (blot X bliver stor nok), således at det ville tage længere tid end universets aktuelle alder (13,8 B år) at bryde!

Combinatorial Explosion

3) Upræcise Problemer

PROBLEM: *Ord-deling*

Hvor skal ordet deles?



Kræver **kontekst** for at afgøre:

I princippet kræves en **komplet model af hele verden**

Hvad er **muligt/umuligt, sandsynligt/usandsynligt**, ...

3) Imprecise Problems

Tak

Spørgsmål? Kommentarer? Klager?

**TAK
FOR
IDAG**

BONUS SLIDES

Teori & Praksis

Teori er når man ved alt,
men intet virker.

Praksis er når *man intet ved*,
men alt virker.

Programmering = teori & praksis:
intet virker & vi ved ikke hvorfor!

Theory & Practice

Funktioner

Funktion

En **funktion** er et stykke kode, der kan køres igen og igen. Man giver funktionen et navn...:

```
function hej() {  
    document.write("Hej med dig!"); // krop!  
}
```

DEFINITION

...og så bruges navnet til at **kalde** funktionen:

```
document.write("<b>");  
hej(); // kald!  
document.write("</b><br/><i>");  
hej(); // kald!  
document.write("</i>");
```

**BRUG
(x2)**

Hej med dig!
Hej med dig!

<https://www.w3schools.com/code/tryit.asp?filename=FSC3QQ1T9JE0>

Funktion

Syntaks:

```
function NAVN() {  
    sætninger...  
}
```

DEF

```
NAVN();
```

BRUG

Eksempel:

```
function hej() {  
    document.write("Hej!");  
}
```

```
document.write("<b>");  
hej(); // kald!  
document.write("</b>");
```

Semantik:

```
function hej() {  
    document.write("Hej!");  
}
```

kald

retur

```
document.write("<b>");
```

```
hej(); // kald!
```

```
document.write("</b>");
```

Argumenter

Funktioner kan tage *argumenter*, så koden bliver *parameteriseret*:

```
function skriv(tekst){  
    document.write('<b><i>');  
    document.write(tekst);  
    document.write('</i></b>');  
}
```

Argumenter er li'som
funktionens variabler

...og kan efterfølgende
bruges i funktionens krop

Når vi kalder funktionen, skal vi give en værdi:

```
skriv("Velkommen!");  
skriv("Hej med dig!");  
skriv("Vi ses!");
```

Velkommen!
Hej med dig!
Vi ses!

<https://www.w3schools.com/code/tryit.asp?filename=FS7O7V8QV7ES>

Argumenter

Man kan have så mange argumenter man vil:

```
function skriv2(tekst, farve) {          // her: to argumenter!
    document.write('<font color=""');
    document.write(farve);
    document.write('">');
    document.write(tekst);
    document.write('</font>');
    document.write('<br/>');
}
```

Vi skal nu give funktionen **to argumenter**:

```
skriv2("Velkommen!", "orange");
skriv2("Hej med dig!", "purple");
skriv2("Vi ses!", "pink");
```

Velkommen!
Hej med dig!
Vi ses!

<https://www.w3schools.com/code/tryit.asp?filename=FSC4J28B8XOL>

Eksempel

Parameteriseret funktion:

```
function repeat(text, n) {  
    for ( var i = 0 ; i < n ; i = i + 1 ) {  
        document.write(text + "<br/>");  
    }  
}
```

Kald med variabel input:

```
repeat("JavaScript er ikke det samme som Java", 5);
```

```
repeat("IT-Universitetet i København", 4);
```

Javascript er ikke det samme som Java
Javascript er ikke det samme som Java

IT-Universitetet i København
IT-Universitetet i København
IT-Universitetet i København
IT-Universitetet i København
IT-Universitetet i København

<https://www.w3schools.com/code/tryit.asp?filename=FSC4JW8FGY9G>

Retur-værdi

En funktion kan også **returnere en værdi**:

```
function square(x) {  
    return x*x; // return bruges til at returnere en værdi med  
}
```

NB: "return" er en såkaldt kontrol-struktur

Returværdien kan så **modtages** af kaldet:

```
var tal = 7;  
var resultat = square(tal);  
document.write(tal + "2 er " + resultat);
```

7^2 er 49

<https://www.w3schools.com/code/tryit.asp?filename=FSC56GM5X475>



Retur-værdi



1) Lav en funktion **mystyle**, som tager et argument **tekst** og **udsriver** tekst'en med diverse HTML-formattering:

Fx bold, italic, font, font-farver, ...

```
mystyle("Hej med dig!");
```

Hej med dig!

2) Lav nu funktionen om så den i stedet **returnerer** teksten med HTML-formattering:

```
document.write( mystyle("Hej med dig!") );
```

Hej med dig!

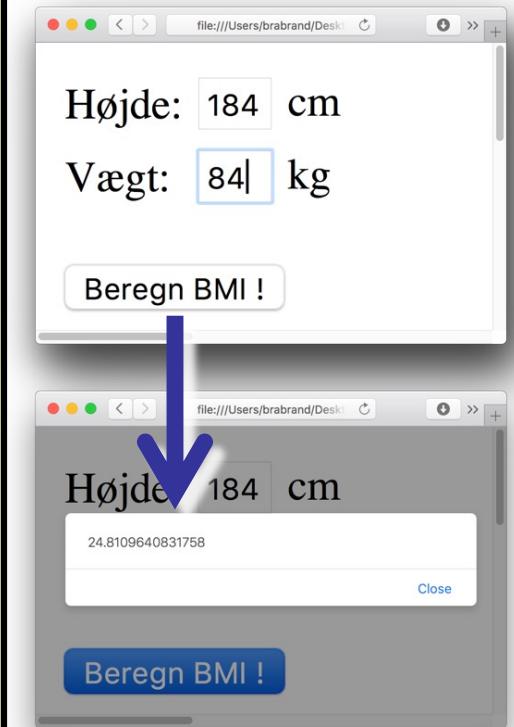


OPGAVE



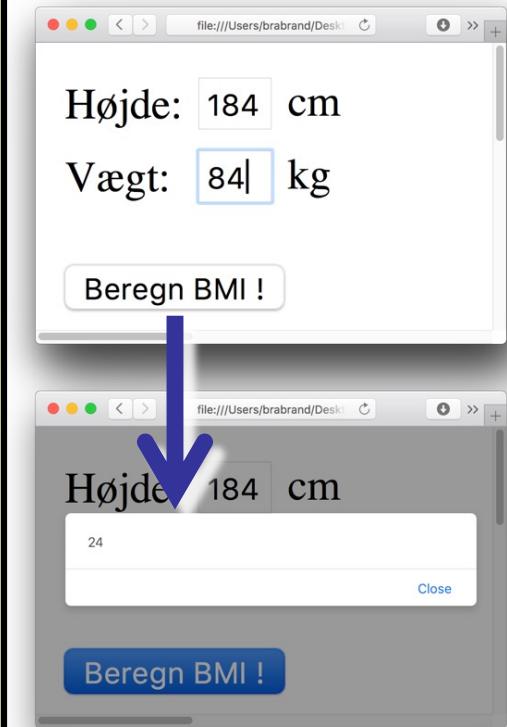
BMI Funktion med Returværdi

```
<html><head><meta charset="UTF-8"/></head>
<script>
    function bmi() {
        var h = document.getElementById('h').value;
        var v = document.getElementById('v').value;
        return v / ((h/100) * (h/100));
    }
</script>
<body>
    <table><tr>
        <td>Højde:</td><td><input id="h" /> cm</td>
    </tr><tr>
        <td>Vægt:</td><td><input id="v" /> kg</td>
    </tr>
    </table>
    <p>
        <button onclick="alert( bmi() );">
            Beregn BMI !
        </button>
    </p>
</body>
</html>
```



BMI Funktion med Returværdi

```
<html><head><meta charset="UTF-8"/></head>
<script>
    function bmi() {
        var h = document.getElementById('h').value;
        var v = document.getElementById('v').value;
        return v / ((h/100) * (h/100));
    }
</script>
<body>
    <table><tr>
        <td>Højde:</td><td><input id="h" /> cm</td>
    </tr><tr>
        <td>Vægt:</td><td><input id="v" /> kg</td>
    </tr>
    </table>
    <p>
        <button onclick="alert( Math.round( bmi() ) ) ;">
            Beregn BMI !
        </button>
    </p>
</body>
</html>
```



<https://www.w3schools.com/code/tryit.asp?filename=FS7HVC2HNAAL>

Vælutaomregning

1) Lav en funktion **eur2dkk**, der omregner *fra euro til kroner*:

(Gang med 7.5)

Test funktionen ved at bruge den og udskriv resultatet

2) Lav en funktion **dkk2eur**, der omregner *fra kroner til euro*:

(Divider med 7.5)

Test funktionen ved at bruge den og udskriv resultatet

3) Prøv at omregne en værdi ved først at omregne med **eur2dkk** dernæst **dkk2eur**

OPGAVE

Temperaturomregning

- 1)** Lav en funktion **c2f**, der omregner
fra Celcius til Fahrenheit:

(Celcius gange 9, divideret med 5, plus 32)

Test funktionen ved at bruge den og udskriv resultatet

- 2)** Lav en funktion **f2c**, der omregner
fra Fahrenheit til Celcius:

(Fahrenheit minus 32, divideret med 9, ganget med 5)

Test funktionen ved at bruge den og udskriv resultatet

- 3)** Prøv at omregne en værdi ved først at
omregne med **c2f** og dernæst med **f2c**



OPGAVE





Spejlvending (★★)



```
<html>
  <body>
    <code>
      <script> 012
        ↓↓↓
        var s = "Strenge gennemløbes og udskrives ét tegn ad gangen.";
        for ( var index = 0 ; index < s.length ; index++ ) {
          var c = s.charAt(index); // find index'te tegn i strengen
          document.write(c); // udskriv tegn'et
        }
      </script>
    </code>
  </body>
</html>
```

<https://www.w3schools.com/code/tryit.asp?filename=FXJHETT1RUZH>

Strenge gennemløbes og udskrives ét tegn ad gangen.

Lav en funktion **reverse(s)**, som tager en tekststreng som argument og spejlvender den:

```
var r = reverse("hello");
document.write(r);
```

olleh



OPGAVE



if-Sætninger: Simplifikation

```
if ( alder == 1 ) {  
    d'w' ("en");  
} else {  
    if ( alder == 2 ) {  
        d'w' ("to");  
    } else {  
        if ( alder == 3 ) {  
            d'w' ("tre");  
        } else {  
            if ( alder == 4 ) {  
                d'w' ("fire");  
            } else {  
                d'w' ("mange");  
            }  
        }  
    }  
}  
}
```

d'w' skal være "document.write"

Nemmere at overskue (strukturelt):

```
if ( alder == 1 )  
    d'w' ("en");  
else if ( alder == 2 )  
    d'w' ("to");  
else if ( alder == 3 )  
    d'w' ("tre");  
else if ( alder == 4 )  
    d'w' ("fire");  
else  
    d'w' ("mange");
```

<https://www.w3schools.com/code/tryit.asp?filename=FS77F8FYJGOB>
<https://www.w3schools.com/code/tryit.asp?filename=FXJB4PX5IV6Z>

if-Sætninger: Simplifikation

```
if ( alder == 1 )
    d'w' ("en");
else {
    if ( alder == 2 )
        d'w' ("to");
    else
        if ( alder == 3 )
            d'w' ("tre");
    else
        if ( alder == 4 )
            d'w' ("fire");
    else
        d'w' ("mange");
}
```

Nemmere at overskue (strukturelt):

```
if ( alder == 1 )
    d'w' ("en");
else if ( alder == 2 )
    d'w' ("to");
else if ( alder == 3 )
    d'w' ("tre");
else if ( alder == 4 )
    d'w' ("fire");
else
    d'w' ("mange");
```

d'w' skal være "document.write"

<https://www.w3schools.com/code/tryit.asp?filename=FS77F8FYJGOB>
<https://www.w3schools.com/code/tryit.asp?filename=FXJB4PX5IV6Z>

BONUS: switch-Sætning

```
if ( alder == 1 ) {  
    d'w'("en");  
} else if ( alder == 2 ) {  
    d'w'("to");  
} else if ( alder == 3 ) {  
    d'w'("tre");  
} else if ( alder == 4 ) {  
    d'w'("fire");  
} else {  
    d'w'("mange");  
}
```

d'w' skal være "document.write"

Kan være (endnu) nemmere at overskue:

```
switch ( alder ) {  
    case 1:  
        d'w'("en");  
        break;  
    case 2:  
        d'w'("to");  
        break;  
    case 3:  
        d'w'("tre");  
        break;  
    case 4:  
        d'w'("fire");  
        break;  
    default:  
        d'w'("mange");  
        break;  
}
```

<https://www.w3schools.com/code/tryit.asp?filename=G3XE3HC800AI>

Ændringer foretages ét sted!

```
function besked() {  
    document.write('<font color="red">Kodning</font>!'); // krop!  
}
```

DEF

```
document.write("<b>");  
besked(); // kald  
document.write("</b><br/><i>");  
besked(); // kald  
document.write("</i>");
```

USE

ændring!

Kodning!
Kodning!

```
function besked() {  
    document.write('<font color="blue">Programmering</font>!'); // krop!  
}
```

DEF

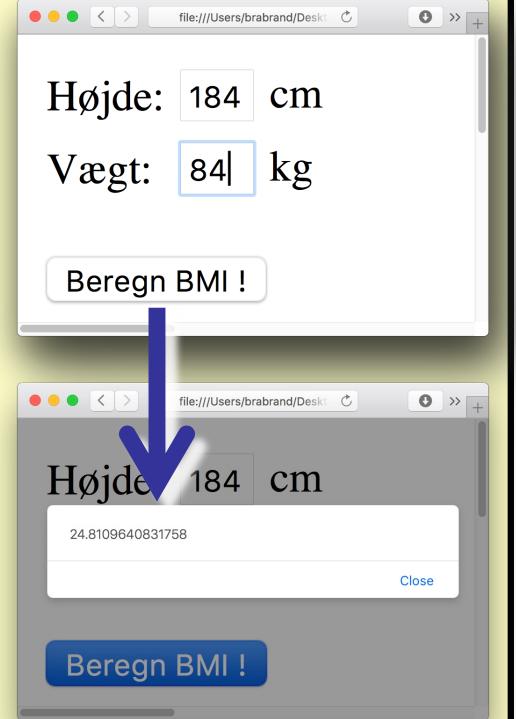
```
document.write("<b>");  
besked(); // kald  
document.write("</b><br/><i>");  
besked(); // kald  
document.write("</i>");
```

USE

Programmering!
Programmering!

BMI Beregner

```
<html><head><meta charset="UTF-8"/></head>
<body>
  <table>
    <tr>
      <td>Højde:</td>
      <td><input id="h" size="3" /> cm</td>
    </tr>
    <tr>
      <td>Vægt:</td>
      <td><input id="v" size="3" /> kg</td>
    </tr>
  </table>
  <p>
    <button onclick="var h = document.getElementById('h').value;
                    var v = document.getElementById('v').value;
                    alert( v / ((h/100) * (h/100)) )">
      Beregn BMI !
    </button>
  </p>
</body>
</html>
```

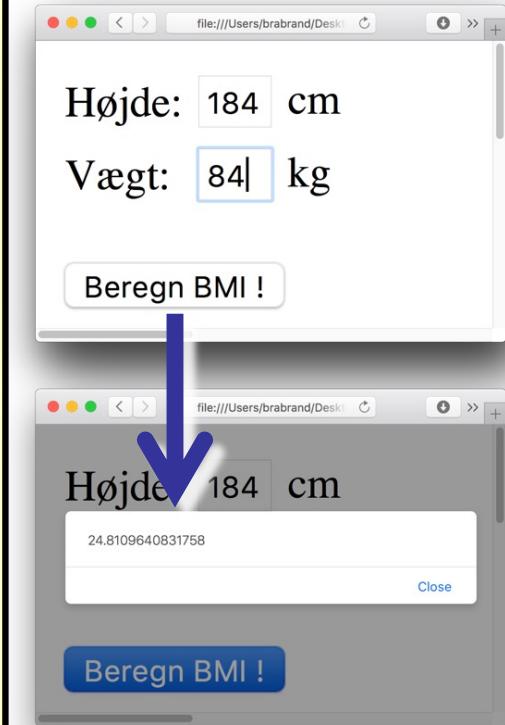


The screenshot shows a window titled "Beregn BMI !". It contains two input fields: "Højde" with value "184" and "cm", and "Vægt" with value "84" and "kg". Below the inputs is a button labeled "Beregn BMI !". A blue arrow points from the "Højde" input field down to the "Højde" input field in the code editor.

<https://www.w3schools.com/code/tryit.asp?filename=FV5M0LUN4NYR>

BMI Beregner med Funktion

```
<html><head><meta charset="UTF-8"/></head>
<script>
    function bmi() {
        var h = document.getElementById('h').value;
        var v = document.getElementById('v').value;
        alert( v / ((h/100) * (h/100)) );
    }
</script>
<body>
    <table><tr>
        <td>Højde:</td><td><input id="h" /> cm</td>
    </tr><tr>
        <td>Vægt:</td><td><input id="v" /> kg</td>
    </tr>
    </table>
    <p>
        <button onclick="bmi(); ">
            Beregn BMI !
        </button>
    </p>
</body>
</html>
```



<https://www.w3schools.com/code/tryit.asp?filename=FS7HJQGT4NWW>