Conditional Statements

if, if-else & else-if

Computer Science

KØBENHAVNS ERHVERVSAKADEMI

Agenda

- if og if-else
- Operatorer og sammenligning
- else-if
- Vis “Game Over” eller “Level Complete”
- Genstart animationer
- Opsamling og vejledning

Øvelser

- Tæl et tal op og ned (1)
- Tæl et tal op og ned (2)
- Tæl et tal op og ned (3)
- Child, adult or senior? (1)
- Child, adult or senior? (2)
- Vis “Game Over” eller “Level Complete”
- Clicker-spil - Game Over og Level Complete

Programming knowledge

JavaScript

JS
TIPS
Aa

Often, when we code, we want to decide what to do based on a condition, a number, a string, a variable...

Når vi ikke har flere liv?

Når vi fx har fået 10 point?

Når vi får mere end 3 liv?

Vis “Game Over” eller “Level Complete”

Conditional Statements

“

... are used to perform different actions based on different conditions.

”

Conditional Statement

```
if (the condition is met) {
    Then do this...
}
```

Conditional Statement

```
if (lives < 0) {
    gameOver();
}
```

Conditional Statement

```
if (lives <= 0) {
    gameOver();
}
```

Lad os lege med det!

```
let num = 10;

if (num > 5) {
    console.log("Tallet er større end 5");
}
```

“

I dette eksempel definerer vi en variabel `num` og checker så om det er større end 5. Hvis betingelsen er sand, vil koden inden i krølleparenteserne blive udført, i dette tilfælde at logge beskeden "Tallet er større end 5". Hvis betingelsen er falsk, vil koden inde i krølleparenteserne blive sprunget over.

ChatGBT

”

Lad os lege med det!

```
let num = 2;

if (num > 5) {
    console.log("Tallet er større end 5");
} else {
    console.log("Tallet er mindre end eller lig med 5");
}
```

“

Du kan også inkludere en "else" del i din if-sætning for at definere, hvad der skal ske, hvis betingelsen ikke er opfyldt.

I dette eksempel vil koden inde i "if" blokken ikke blive udført, fordi betingelsen ikke er opfyldt. Koden i "else" blokken vil blive udført i stedet, og beskeden "Tallet er mindre end eller lig med 5" vil blive logget.

ChatGBT

”

Tæl et tal op og ned (1)

Back

https://github.com/cederdorff/variables-and-outputs/tree/solution

- Vis teksten “The number is above 10”, når tallet er større end 10.

The use of

`==` `===` `<` `>` `>=` `<=` `!=`

Operatorer

`==`	Er lig med (værdien)	`x == 5`
`!=`	Er ikke lig med (er forskellig fra)	`2 != 5`
`>`	Er større end	`5 > 2`
`<`	Er mindre end	`2 < 5`
`>=`	Er større end eller lig med	`5 >= 2`
`<=`	Er mindre end eller lig med	`2 <= 5`

JavaScript Comparison and Lo

w3schools.com/js/js_comparisons.asp

HTML CSS JAVASCRIPT SQL PYTHON JAVA

Comparison Operators

Comparison operators are used in logical statements to determine equality or difference between variables or values.

Given that `x = 5`, the table below explains the comparison operators:

Operator	Description	Comparing	Returns	Try it
==	equal to	x == 8	false	Try it »
		x == 5	true	Try it »
		x == "5"	true	Try it »
===	equal value and equal type	x === 5	true	Try it »
		x === "5"	false	Try it »
!=	not equal	x != 8	true	Try it »
!==	not equal value or not equal type	x !== 5	false	Try it »
		x !== "5"	true	Try it »
		x !== 8	true	Try it »
>	greater than	x > 8	false	Try it »
<	less than	x < 8	true	Try it »
>=	greater than or equal to	x >= 8	false	Try it »
<=	less than or equal to	x <= 8	true	Try it »

JavaScript Operators

w3schools.com/js/js_operators.asp

HTML CSS JAVASCRIPT

JavaScript Comparison Operators

Operator	Description
==	equal to
===	equal value and equal type
!=	not equal
!==	not equal value or not equal type
>	greater than
<	less than
>=	greater than or equal to
<=	less than or equal to
?	ternary operator

== VS ===

=== IS MORE STRICT

=== compares both types and values
== compares values

"32" == 32 is true
"32" === 32 is false
32 === 32 is true

Conditional Statement

```
if (the condition is met) {
    Then do this...
} else {
    Do this...
}
```

Tæl et tal op og ned (2)

Back

- Vis teksten “The number is above 10, når tallet er større end 10.
- Vis teksten “The number is below 10”, når tallet er under 10.

Conditional Statement

```
if (lives <= 0) {
    gameOver();
} else {
    levelComplete();
}
```

Conditional Statement

```
if (lives <= 0) {
    gameOver();
} else {
    levelComplete();
}

function gameOver() {
    console.log("Game Over");
}

function levelComplete() {
    console.log("Level Complete");
}
```

- Split your code into functions

“Separation of Concerns”

```
if (lives <= 0) {
    gameOver();
} else {
    levelComplete();
}

function gameOver() {
    console.log("Game Over");
}

function levelComplete() {
    console.log("Level Complete");
}
```

- Call one function inside of your if statement
- And one function inside of your else statement.
- ALWAYS!

Conditional Statements

https://www.w3schools.com/js/js_if_else.asp

`if` to specify a block of code to be executed, if a specified condition is true

```
if (condition) {
  //  block of code to be executed if the condition is true
}
```

`else` to specify a block of code to be executed, if the same condition is false

```
if (condition) {
  //  block of code to be executed if the condition is true
} else {
  //  block of code to be executed if the condition is false
}
```

`else if` to specify a new condition to test, if the first condition is false

```
if (condition1) {
  //  block of code to be executed if condition1 is true
} else if (condition2) {
  //  block of code to be executed if the condition1 is false
and condition2 is true
} else {
  //  block of code to be executed if the condition1 is false
and condition2 is false
}
```

https://www.w3schools.com/js/js_if_else.asp

else if

```
if (the condition is met) {
    Then do this...
} else if (another condition) {
    Do this...
} else {
    Do this...
}
```

else if

```
if (lives <= 0) {
    gameOver();
} else if (points >= 10) {
    levelComplete();
} else {
    gameOver();
}
```

Tæl et tal op og ned (3)

Back

- Vis teksten “The number is above 10, når tallet er større end 10.
- Vis teksten “The number is below 10”, når tallet er under 10.
- Vis teksten “The number is 10”, når tallet er lig med 10.

Back

Child, adult or senior? (1)

1. Lav et program, der kan afgøre om en person er voksen eller barn.
2. Anvend en variable til at holde alder.
3. Brug dernæst if-else til at afgøre om personen er voksen eller barn:
    1. Du skal kalde en funktion der hedder `isAdult`, hvis betingelsen mødes, ellers skal du kalde en funktion, der hedder `isChild`. Funktionerne skal anvende `console.log()` til at teste. Fx `console.log("Du er voksen").`
4. Tilføj logik, der gør at funktionerne kan vise en tekst i DOM'en, så du ikke blot får `console.log`'s. Du skal bruge `.textContent`.

```
if (the condition is met) {
    Then do this...
} else {
    Do this...
}
```

Back

Child, adult or senior? (2)

5. Udvid dit program, så du kan afgøre om en person er barn, voksen eller pensionist.
6. Anvend fortsat en variable til at holde alder.
7. Brug dernæst if-elseif-else til at afgøre om personen er barn, voksen eller pensionist. Pensionistalderen kan sættes til 72 år.
8. Du skal fortsat anvende funktioner der `console.log`'er.
9. Tilføj logik, der gør at funktionerne kan vise en tekst i DOM'en, så du ikke blot får `console.log`'s. Du skal bruge `.textContent`.

```
if (the condition is met) {
    Then do this...
} else if (another condition) {
    Do this...
} else {
    Do this...
}
```

Back

Vis “Game Over” eller “Level Complete”

1. Download: Coinclicker-mini-2023 (lives-and-points)
2. I `index.html`, tilføj indhold til `#game_over` og `#level_complete` - test ved at fjerne klassen `hidden`.
3. I `script.js`, lav to funktioner, `gameOver()` og `levelComplete()`, der `console.log`’er hhv “Game Over” og “Level Complete”.
4. I `decrementedLives()`
   Hvis lives er mindre end eller lig med 1, kald `gameOver()`.
5. I `incrementPoints()`
   Hvis `points` er større end eller lig med fx 10, kald `levelComplete()`
6. Husk fortsat at separere logik i funktioner (hvis du har fulgt ovenstående, har du gjort det).
7. Funktionerne `gameOver()` og `levelComplete()` skal nu fjerne CSS-klassen `hidden` på hhv `#game_over` og `#level_complete`, så vi får vist “Game Over”- eller “Level Complete”-skærmen.
8. Sørg for at du ikke får en fejl, når du klikker på et hjerte og allerede har 3 liv. Hint: det er også noget med `if`!
9. Ekstra: Når spillet er slut, skal du kalde en funktion, der fjerner alle animationer og events (dvs det modsatte af hvad `start()` gør). Kald funktionen `stop()`.

Back

Clicker-spil - Game Over og Level Complete

- Brug de beslutninger, du har fastlagt i dit aktivitetsdiagram.
- Game Over eller Level Complete skærmen skal vises – og de skal som minimum fortælle brugeren om deres spil var en succes eller ej.
- Spillet skal stoppe, når det er slut – animationerne skal ophøre, og det må ikke længere være muligt at klikke på noget. Lav fx en funktion, der fjerner alle animationer og alle events (det modsatte af hvad `start()` gør).
- Spillet skal kun kunne spilles én gang – der behøver ikke være nogen knap til at prøve igen (kommer senere).
- Der er ingen timer i spillet endnu (kommer senere).
- Tip: Lav så mange funktioner (små) som muligt, der tager sig af en afgrænset logik!

Genstart Animationer

og force reflow

Force reflow

For at genstarte en keyframe-animation, skal man tage animationsklassen af, og sætte den på igen:

```
document.querySelector("selector").classList.remove("class");
document.querySelector("selector").classList.add("class");
```

JavaScript er dog så hurtig til at udskifte animationen at browseren ikke opfatter det. Man blive derfor nødt til at tvinge den til at “se” at noget er ændret.

Det gør man ved at bede den om at finde ud af hvor et element befinder sig. Det kaldes at gennemtvinge (force) et reflow:

```
document.querySelector("selector").classList.remove("class");
document.querySelector("selector").offsetLeft;
document.querySelector("selector").classList.add("class");
```

Force reflow

```
function bombGone() {
    // fjern event der bringer os herind
    document.querySelector("#bomb_container").removeEventListener("animationend", bombGone);

    // fjern forsvind-animation
    document.querySelector("#bomb_sprite").classList.remove("zoom_in");

    // fjern pause
    document.querySelector("#bomb_container").classList.remove("paused");

    // genstart falling animation
    document.querySelector("#bomb_container").classList.remove("falling");
    document.querySelector("#bomb_container").offsetWidth;
    document.querySelector("#bomb_container").classList.add("falling");

    // gør det muligt at klikke på bomb igen
    document.querySelector("#bomb_container").addEventListener("click", clickBomb);
}
```

Genstart animationer

Inspiration til brug af if, if-else og else-if

Inspiration, inspiration, inspiration

... eller stjæl med arm(e) og ben 🤷🏼‍♂️

Code
Every
Day
... and have fun!