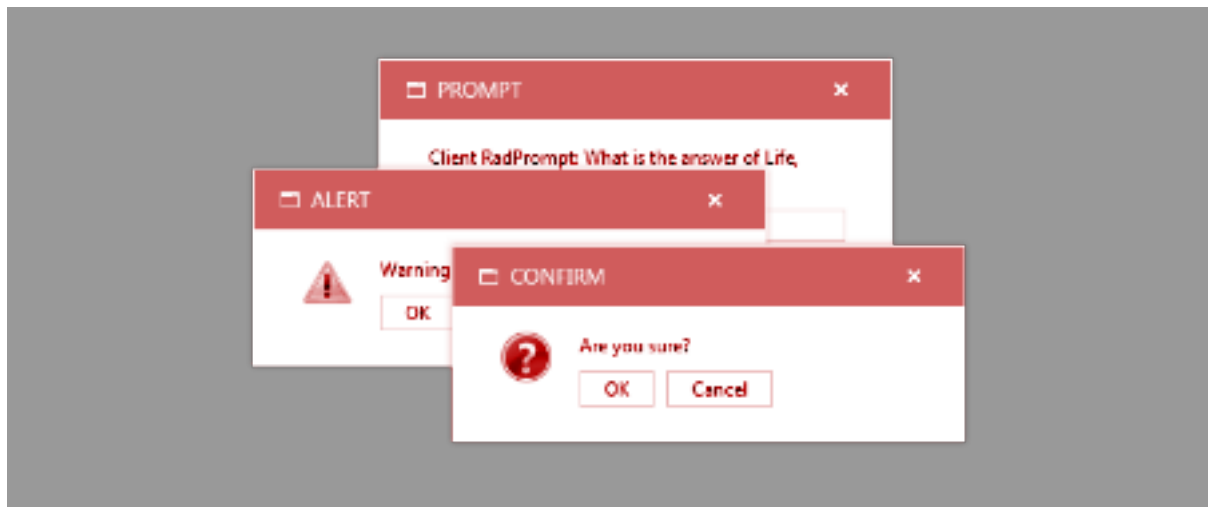


Dialog Box Game - Opgave



Målet for denne opgave er at udvikle et simpelt spil kodet i JavaScript. Spillet skal bestå af en række dialogbokse, hvor spilleren skal svare på spørgsmål eller indtaste information. For hvert korrekt svar eller valg tildeles spilleren 1 point. Når spilleren har svaret på samtlige spørgsmål afsluttes spillet med at vise en infoboks med en oversigt over alle spørgsmål og svar samt den samlede score, som spilleren har opnået.

Du kan se en demo [her](#).

Spillets gang (game play)

1. Spillet skal starte med at bede spilleren om at indtaste sit navn (prompt box)
2. Spilleren præsenteres for en velkomsthilsen (confirm box)
3. Spilleren svarer på første spørgsmål (prompt box). Korrekt svar giver 1 point
4. Spilleren svarer på andet spørgsmål (prompt box). Korrekt svar giver 1 point
5. Spilleren svarer på tredje spørgsmål (prompt box). Korrekt svar giver 1 point
6. Udvid antallet af spørgsmål til minimum 5
7. Når der ikke er flere spørgsmål skal spillet afsluttes ved at spilleren præsenteres for en infoboks med en oversigt over alle sine svar samt en samlet score

Forberedelse

Før du begynder at kaste dig over selve programmeringsdelen skal du planlægge, hvad dit spil skal handle om og hvilke spørgsmål og svarmuligheder, der skal indgå i spillet.

Opret HTML-dokument

1. Opret et nyt tomt HTML-dokument med [standard-kode](#) og navngiv det **dialog-box-game.html**
2. **<title> tagget** skal have titlen "Dialog Box Game"
3. Tilføj et **<script> tag** i bunden af HTML-dokumentet lige over **</body> tagget**.
I hjælpe-dokumentet under afsnittet "JavaScript kode" finder du et eksempel på, hvordan koden skal se ud. Al fremtidig JavaScript-kode skal placeres mellem disse to **<script>** tags
4. Mens du er ved HTML-delen kan du ligeså godt tilføje endnu et element, som skal bruges. Det drejer sig om et **<div> tag** som skal have en **id** med titlen **result**. Det fulde element skal se sådan ud: **<div id="result"></div>**.
Elementet skal bruges som container for de elementer, der skal vise det endelige resultat og samlet score
5. **<div> tagget** skal indsættes ovenover **<script> tagget**

Tilføj JavaScript

Ifølge **game play'et** skal spilleren starte med at indtaste sit navn. Den udfordring løser du med en dialogboks af typen **prompt**. Du skal desuden gemme brugerens indtastning i en **variabel** ved navn **userName**, fordi navnet skal benyttes senere.

Placér følgende kode mellem de to **<script> tags** i dit **HTML-dokument**.

Javascript:

```
var userName = prompt('Indtast dit navn:');
```

Når spilleren har indtastet sit navn, er det meningen, at han skal præsenteres for en velkomsthilsen, hvor hans indtastede navn indgår. Men allerede her støder du ind i det første dilemma.



Hvad nu hvis spilleren ikke indtaster et navn, men bare trykker 'ok' eller 'cancel'?

Pas på 'bugs'

Du har sikkert hørt om "bugs" i spil og programmer. "Bugs" er mindre fejl i koden, der fx opstår hvis programmøren "overser" nogle handlinger, som en spiller kan finde på at udføre, men som koden ikke er skabt til at håndtere.

Du er derfor nødt til at tage højde for muligheden. Ellers skabes der en "bug".

Det gør du ved at oprette en betingelse i form af et **if-statement**. Betingelsen sikrer, at spillet kun kan fortsætte, hvis spilleren har indtastet sit navn (eller anden tekst).

Tilføj betingelse

Nedenstående eksempel viser den nødvendige kode. Som betingelse for dit **if-statement** skal du benytte variabelen **userName**.

Hvis den indeholder en værdi, hvilket den kun gør, hvis spilleren har indtastet et navn, returneres værdien 'true', hvilket gør betingelsen sand. I så fald afvikles koden mellem den krøllede parenteser.

Hvis feltet er tomt, returneres værdien 'false' og koden mellem de krøllede parenteser springes over.

Indsæt nedenstående **if-statement** i din kode og placér al efterfølgende kode indeni, som vist i eksemplet herunder.

Javascript:

```
if(userName){  
    // Kode afvikles kun, hvis betingelse er sand  
}
```

Opret velkomsthilsen

Nu kan du endelig oprette kodelinjen, der tilføjer en velkomsthilsen til spilleren. Husk, at den skal placeres mellem de krøllede parenteser i det **if-statement**, du lige har lavet.

Ved at placere koden til velkomsthilsenen indeni betingelsen, så sikrer du at den kun vises, hvis spilleren har indtastet sit navn.



Javascript:

```
var welcome = confirm('Hej ' + userName + '. Er du klar til at  
spille?');
```

BEMÆRK: Velkomsthilsenen oprettes med en **confirm box**, der indeholder en statisk velkomsttekst, der er flettet sammen med spillerens navn via **concatenate-metoden** (plustegn) for at gøre hilsenen personlig.

Det karakteristiske ved en **confirm box** er, at den er udstyret med en 'ok' og 'cancel' knap. Hvis spilleren klikker på 'ok' returneres værdien 'true'. Hvis spilleren klikker på knappen 'cancel', returneres værdien 'false'.

Opret ny betingelse

Din næste opgave er derfor at oprette endnu en betingelse, som denne gang skal teste, om spilleren har klikket på 'ok' knappen. Ellers skal spillet afbrydes.

Bemærk de dobbelte lighedstegn. De betyder at du sammenligner de to værdier - altså om **variablen welcome** indeholder værdien 'true', hvilket den kun vil gøre, hvis spilleren har klikket på knappen 'ok'.

Javascript:

```
if(welcome == true){  
    // Kode afvikles kun, hvis betingelse er sand  
}
```

Point-system

Du er nu næsten kommet til det punkt, hvor du skal til at oprette de spørgsmål og svar, som skal udgøre selve spillet.

Men inden du begynder på det, skal du lave et pointsystem. Det gør du ved at oprette en **variabel** ved navn **score** og tildele den **værdien 0**, som vist i eksemplet herunder. Hver gang spilleren svarer rigtigt på et spørgsmål vil der blive lagt 1 til værdien af variabelen **score**. Den vil derfor altid indeholde spillerens samlede antal point.

Husk at koden skal placeres mellem de krøllede parenteser i det **if-statement**, du lige har lavet.



Javascript:

```
var score = 0;
```

Opret spørgsmål og svar

Du skal nu til at oprette de spørgsmål og svar, som udgør selve spillet. Kodemæssigt udgør de også den mest komplicerede del af spillet.

I stedet for at præsentere dig for en kodeblok opdelt i mindre afsnit, får du her vist hele koden til ét spørgsmål og svar. Under eksemplet vil kodelinjerne blive forklaret én for én.

Javascript:

```
var question1 = prompt('Spørgsmål 1:\nHvad er 2 + 2?');
if(question1 == 4){
    alert('Korrekt');
    var answer1 = 'Spørgsmål 1: Korrekt';
    score++;
} else {
    alert("Forkert");
    var answer1 = 'Spørgsmål 1: Forkert';
}
```

1. I første kodelinje oprettes en variabel ved navn **question1**, som gemmer det svar, som spilleren indtaster i **prompt-boksen**
2. I anden kodelinje oprettes et **if-statement**, der tester, om spillerens svar matcher tallet 4, som i dette tilfælde er det korrekte svar
3. Hvis betingelsen er sand, så afvikles koden mellem de krøllede parenteser
 - a. Først vises en **alert-boks** med teksten **"Korrekt"**
 - b. Derefter oprettes en ny **variabel** ved navn **answer1** og tildeles værdien **'Spørgsmål 1: Korrekt'**. Denne skal bruges til visning sammen med den samlede score
 - c. **score++** lægger 1 til den aktuelle værdi i variablen score. Ved spillets start blev den sat til 0 svarende til 0 point. Ved at svare rigtigt på første spørgsmål får spilleren derfor 1 point.
4. Hvis betingelsen IKKE er sand, så springes kodeblokken over. I stedet afvikles den kodeblok der er placeret indenfor de krøllede parenteser efter **else-statementet**
 - a. Først vises en alert-boks med teksten **"Forkert"**



- b. Derefter oprettes en ny variabel ved navn **answer1** og tildeles værdien **'Spørgsmål 1: Korrekt'**. Denne skal bruges til visning med den samlede score

BEMÆRK: Hele kodeblokken skal gentages for hvert spørgsmål og svar, dit spil skal indeholde. Dog med den forskel, at de nye spørgsmål og svar, du opretter, samt navngivning af variabler skal være forskellig fra dem, du allerede har oprettet. Dette er afgørende. Ellers vil spillet ikke virke.

Dit spil skal indeholde mindst 5 spørgsmål og svar.

Infoboks med resultat

Spillet afsluttes automatisk, når der ikke er flere spørgsmål. Når det sker, skal spilleren præsenteres for en oversigt med korrekte/forkerte svar samt den samlede score.

Det endelige resultat skal ikke vises i en dialog box, men udskrives som HTML-kode og vises på websiden.

Det kan du gøre ved at benytte koden i eksemplet herunder.

Javascript:

```
var result = document.querySelector('#result');
result.innerHTML = '<h2>Samlet score for ' + userName + ': ' + score +
'</h2>';
result.innerHTML += answer1 + '<br>';
result.innerHTML += answer2 + '<br>';
result.innerHTML += answer3;
```

1. I første kodelinje udvælges det HTML-element på websiden, som har en **id** ved navn **result**, hvilket er **<div> elementet**
2. I anden kodelinje benyttes metoden **innerHTML** på variablen **result** til at tilføje en **<h2> overskrift** til **<div> elementet**. Overskriften indeholder en statisk tekst, som er flettet sammen men variablerne **userName** og **score** via **concatenate-metoden**
3. I tredje kodelinje benyttes metoden **innerHTML** til at tilføje variablen **answer1**, som indeholder svaret på spørgsmål 1
4. I fjerde kodelinje benyttes metoden **innerHTML** til at tilføje variablen **answer2**, som indeholder svaret på spørgsmål 2



5. I tredje kodelinje benyttes metoden **innerHTML** til at tilføje variablen **answer3**, som indeholder svaret på spørgsmål 3

Tilføj startknap til spillet

I stedet for at spillet bare starter, når websiden indlæses, så kan du relativt nemt lave en tilføjelse til din kode, som gør det muligt at starte spillet ved at klikke på en knap.

Det vil gøre spiloplevelsen mere autentisk.

Det første krav er, at du placerer al din kode i en **JavaScript funktion**. En funktion afvikles kun, når man beder den om det, så det vil forhindre spillet i bare at starte, når websiden indlæses.

Funktionen skal have en navn: **startGame** er et passende navn til funktionen, som vist i eksemplet herunder.

Javascript:

```
function startGame(){  
    // Hele dit JavaScript placeres her  
}
```

Så mangler du bare en "udløser", som skal være en knap, der starter spillet, når spilleren klikker på den.

Knappen skal laves i HTML med et `<button>` tag, som vist i eksemplet herunder. For at få knappen til at virke, skal der tilføjes et **JavaScript onclick-event**, der kan udføre en handling, når der klikkes.

OnClick-eventet skal indeholde navnet på din funktion, hvilket får den til at starte, når spilleren klikker på knappen.

HTML/JavaScript:

```
<button onclick="startGame()">Start spil</button>
```

CSS styling

Dit spil er nu næsten færdigt. Du har oprettet en struktur i HTML. Derefter har du tilføjet interaktion med JavaScript.



Det sidste, som mangler er stylingen - d.v.s. hvordan spillet skal se ud uds. Det styres med kodesproget CSS.

Du kan tilføje en standard-styling ved at benytte koden i eksemplet herunder, men du er også velkommen til at eksperimentere med din egen.

Det kan ikke lade sig gøre at style dialogboksene. Deres styling styres af browseren. Derfor er deres udseende også forskelligt alt efter, hvilken browser, du benytter.

Stylingen rammer kun websidens indhold - d.v.s. knappen og infoboksen med spillets resultat.

CSS:

```
body{
  display: flex;
  flex-direction: column;
  justify-content: center;
  align-items: center;
  min-height: 100vh;
  background: orange;
}
button{
  background: red;
  color: white;
  padding: 2rem 3rem;
  font-size: 2rem;
  border: none;
}
#result{
  font-size: 1.4rem;
  line-height: 1.2;
  padding: 1rem;
}
```

Når du har tilføjet CSS styling, er dit spil færdigt og klar til at gå i produktion.

Godt arbejde :-)