

Övning 9: Web - Front-end och JavaScript

Övning och tillämpning av klient funktionalitet

Bish-Bosh

Bish-Bosh listar alla tal mellan 1 och 100, men vissa undantag: Om talet är jämnt delbart med 3 listas '*Bish*' istället för talet. Om talet är jämnt delbart med 4 listas '*Bosh*' istället för talet. Om talet är jämnt delbart med både med 3 och 4 skrivs "*Bish-Bosh*" istället för talet.

Exempel:

1, 2, *Bish*, *Bosh*, 5, *Bish*, 7, *Bosh*, *Bish*, 10, 11, *Bish-Bosh*, 13, 14, *Bish* [...]

En första implementation

Denna implementation ställer inga krav på övrig front-end utan enbart att javascriptet fungerar och applikationen gör som den ska - alltså exekverar enligt ovan exempel. För att testa om något är jämnt delbart så kollar man om resten är noll vid division (tips: modulus-operatorn %).

I denna implementation går det bra att skriva javascriptet inom <script>-taggar direkt i HTML-filen och göra sina utskrifter till konsolen.

En andra implementation: Bish-Bosh 2.0

Nu skall ni skriva om *Bish-Bosh* med ett visuellt gränssnitt och presentation på en HTML-sida. Ni skall även låta en användare sätta indata, alltså:

Vilken siffra som är Bish (tidigare 3), vilken siffra som är Bosh (tidigare 4) och hur långt loopen skall gå (tidigare 100) inga negativa värden tillåts.

- Det skall användas variabler för:
 - Loop tal (alltså vart loopen ska sluta)
 - Första divisionstal (det som ska ersättas med Bish)
 - Andra divisionstal (det som ska ersättas med Bosh)
- Ett webbaserat UI som innehåller:
 - Inputfält för ovanstående variabler
 - En header (Bish-Bosh 2.0)
 - En footer med ert namn och copyright
 - En text som förklarar 'Bish-Bosh' och i vilket syfte ni skapat sidan.
 - En presentation av talen
- Separata filer:
 - index.html
 - style.css
 - bishbosh.js

För att uppdatera det faktiska innehållet på sidan så kan ni hitta bra material här:

http://www.w3schools.com/js/js_intro.asp

Som vi lärt oss i andra kursavsnitt så vill jag att ni håller era script i en separat fil för att sedan referera den från HTML-dokument.

Inga script referenser eller liknande i HTML-dokumentet såsom onclick!

ToDo list

Övningen består i att skapa en applikation som skapar en inköpslista.

Användaren ska ha möjlighet att skriva in olika saker som ska inhandlas som sedan läggs till i listan.

1. Skapa en html sida där användaren har möjlighet att skriva in en vara som ska läggas till listan i ett input fält.
2. Skapa en knapp som lägger till varan i listan.
3. När varan har lagts till ska man ha möjlighet att markera varan som inköpt genom att klicka på varan.

Texten blir då överstruken samt elementet ändrar bakgrundsfärg.

4. Det ska gå att ångra att varan blev tillagd.

Texten återgår då från överstruken till normal samt elementet återfår sin ursprungliga färg.

5. Det ska även finnas en möjlighet att ta bort varan helt från listan.

Använd er gärna av Bootstrap och av era CSS kunskaper för att styla applikationen, experimentera!

HTML CSS samt JavaScript filerna ska ligga i olika dokument!

Extra funktionalitet (helt valfritt)

Spara undan inköpslistan persistent

När man skapar upp en ny lista kan man välja utseende eller liknande

Markera/Avmarkera alla varor

Hantera flera inköpslistor

Flytta saker mellan listor

Går givetvis att bygga ut applikationen för ytterligare funktionalitet.

Extra

Extra (Valfritt)

Flera övningar:

<https://www.w3resource.com/javascript-exercises/>

<https://javascript30.com/>

<https://open.kattis.com/>

<https://practity.com/582-2/>

<https://edabit.com/challenges/javascript#>!

<https://exercism.io/tracks/javascript/exercises>

Välj efter egen nivå! Men se till att utmana er själva!

För de som redan jobbat med Javascript börja gärna titta på något av ramverken i frontend tex React, Angular eller Vue. Men rekommenderar enbart det för de som redan känner att de behärskar grunderna i Javascript.

Lycka till!