

# Labb 3 – Interaktivitet I

Multimedia 7.5 hp VT-14

## Introduktion

I denna laboration kommer vi att arbeta med javascript för att skapa interaktivitet. Tanken med laborationen är att vi kommer att befästa och lära oss javascript, html och css parallellt. Vi förväntar oss att ni har goda kunskaper i programmering och att inläsning på iteration, selektion, variabler och metoder görs innan.

## Inlämning

Denna laboration består av tre faser där du ska lämna in varje fas i en egen mapp. Din inlämning ska alltså bestå av en .zip-fil eller .rar-fil (inga andra komprimeringsformat är tillåtna!) innehållandes följande (med följande struktur):

- labb2\_fornamn\_efternamn.zip
  - uppgift1 (mapp)
    - \* hello.js
    - \* index.html
  - uppgift2 (mapp)
    - \* randomizer.js
    - \* index.html
  - uppgift3 (mapp)
    - \* menu.js
    - \* menu.css
    - \* index.html

## Kodstandard

- Ingen Javascript eller CSS ska placeras “inline” i något htmdokument. All javascript-kod skall alltså skrivas i .js-filer och css i .css-filer.
- All kod ska vara korrekt indenterad! (Läs mer om indentering på [htmlhunden](#) om du är osäker)

# Uppgifter

Nedan följer uppgifterna som resulterar i inlämningarna ovan.

## Uppgift 1 Hej! *Namn*

I denna uppgift skall ni skapa ett formulär i HTML som kan med hjälp av javascript läsa in från textboxen och sedan säga Hej + *textboxens värde* Alltså kommer ert formulär se ut som nedanstående:

Skriv in ditt namn:

OK

1. Börja med att skapa en .html-fil där du skapar upp ett skal för din html.
2. Skapa sedan en .js-fil där du kommer att skapa metoder som du behöver.
3. Skapa sedan en .css-fil där all formatering skall finnas!
4. Försök nu skapa de element du behöver i .html-filen, alltså ett `<input>`-fält och en knapp.
5. Börja sedan med det enklare problemet: Att få ett statiskt meddelande att visas när användaren trycker på knappen. Förslagsvis kan du använda dig av `alert()` för att visa meddelandet.
6. Först när du lyckats med det, kan du gå vidare till att försöka byta ut det statiska meddelandet mot strängen "Hello `<name>`" där `<name>` ersätts med det användaren skrivit in i textboxen.

### Tips

- Om du vill undvika att behöva bry dig om vilka filer som ska vara vart och hur de hänger ihop kan du arbeta direkt i webbläsaren genom [jsfiddle](#) istället. Du kommer dock att behöva lära dig att koppla ihop dokumenten själv i slutändan eftersom uppgifterna ska lämnas in.

## Uppgift 2 Slumpmässig färg

I denna övning ska vi träna på att förändra CSS-egenskaper genom JavaScript. Målet är en helt tom sida med en knapp/länk. När man trycker på länken så ska hela sidans bakgrundsfärg ändras. En ny bakgrundsfärg ska slumpas fram vid varje knapptryckning.

Tänk på att den här uppgiften kan delas upp i (t.ex.) följande delproblem:

- “Lyssna” på knapptryckningen
- Slumpa fram tal
- Slumpa fram en färg
- Ändra bakgrundsfärgen

Tips: Det kan underlätta att sätta färgerna genom `rgb(x,y,z)` istället för att använda hexadecimal notation (e.g. `#333333`).

### Uppgift 3 Interaktiv meny

Denna övning ger dig lite större frihet än de tidigare. Målet med denna övning är att producera en `.html`-sida innehållandes tre menyknappar. Sidan ska även innehålla tre informationsrutor (ex. `<div>:ar`). När sidan laddas ska alla informationsrutorna utom den första vara dold. När användaren sedan klickar på respektive meny-knapp så ska respektive informationsruta visas, och övriga döljas.

Endast seriösa försök beaktas som inlämningar. Skapa en vision i ditt huvud kring hur sidan borde se ut för att på lättast möjliga sätt kommunicera sitt syfte (navigationen) till användaren och exekvera din vision med hjälp av `css`.

Tips: Försök att, precis som i förra uppgiften, bryta upp problemet i delproblem, och använd dig av funktioner för att strukturera din Javascript-kod!