

# Labb 5 – Animation

Multimedia 7.5 hp VT-14

## Introduktion

I denna laboration kommer vi att arbeta med animation för webben. Detta genom tre eskalerande steg. Först kommer vi att fokusera på att transformera objekt. Sedan på automatiserade transitions mellan olika transformationer. Slutligen på keyframe-baserad animation som bygger på tidigare två.

## Inlämning

Din inlämning ska bestå av en .zip-fil eller .rar-fil (inga andra komprimeringsformat är tillåtna!) innehållandes följande (med följande struktur):

- labb5\_fornamn\_efternamn.zip
  - uppgift1 (mapp)
    - \* index.html
    - \* transformations.css
  - uppgift2 (mapp)
    - \* index.html
    - \* transitions.css
  - uppgift3 (mapp)
    - \* index.html
    - \* animations.css

## Krav

- Alla bilder du använder i denna laboration ska “djuplänkas”. Inga bildfiler ska således finnas i dina inlämningar.
- All kod ska vara korrekt indenterad. Läs mer på [HTMLHunden](#) om du är osäker på hur man indenterar.

# Uppgifter

Nedan följer uppgifterna som resulterar i inlämningarna ovan.

## Uppgift 1 CSS3 Transformations

Den här övningen syftar till att träna på våra kunskaper i **CSS3 Transformations**. Övningen går helt enkelt ut på att hitta ett antal bilder, visa dem på en **html**-sida och sedan applicera **css**-transformationer på dem. Detta kommer alltså innebära att bilderna kommer att transformeras så fort sidan laddas.

1. Skapa ett **.html**-, ett **.css**-dokument och “koppla in” **css**:en.
2. Leta rätt på 4 st bilder och lägg in dem i din **html**-fil (eller använd en bildtjänst såsom t.ex. **placekitten** eller **lorempixel**.)
3. Applicera nu **css**-transformationer på alla 4 bilder (t.ex. **skew**, **rotate**, **scale** etc.)

### Krav

- Använd minst 2 transformationer per bild
- Transformera alla bilderna olika
- Ditt projekt ska vara resonabelt välstilat. Föreställ dig t.ex. att det vi skapar är ett bildgalleri.

## Uppgift 2 CSS3 Transitions

I denna övning ska vi fortsätta arbeta med det projekt vi skapade i förra uppgiften. Det enda vi nu vill uppnå nu, är att de transformationer vi applicerat, endast ska appliceras antingen vid **hover** eller **click**. Om du vill arbeta med **click** så behöver du använda **:target** eftersom vi i denna övning inte får använda oss av **JavaScript**.

1. Gör en kopia på ditt projekt ifrån uppgift 1
2. Förändra ditt projekt fritt så att de transformationer du skapade i uppgift 1 endast appliceras när användaren för musen över en bild, eller klickar på den. Det är självklart tillåtet att modifiera sina transformationer för att de ska vara mer anpassade för detta scenario.

Exempel på vad som åsyftas kan du hitta **här**.

## Uppgift 3 CSS3 Animations

I denna övning ska vi göra någonting helt nytt. Vi ska skapa en scrollande bakgrundsbild. På denna bakgrund ska vi sedan scrolla ytterligare (minst) ett objekt för att uppnå en **parallaxeffekt**. Du väljer själv om din bakgrundsbild ska täcka hela skärmen, eller endast en liten del. Det är dock viktigt att du ser till sidan ser lika bra ut oavsett om man förstorar eller förminskar webbläsarfönstret.

1. Skapa ett `.html`-, ett `.css`-dokument och “koppla in” `css`:en.
2. Leta rätt på en bild på internet som kan fungera “panorama-bakgrund” \*\*
3. Använd CSS-animations för låta bakgrunden scrolla i något led (x/y) för evigt.
4. Leta rätt på en bild som du vill applicera ovanpå bakgrunden.
5. Använd CSS-animations för att låta denna andra bild animeras i samma riktning som bakgrunden, fast snabbare, för att uppnå en parallaxeffekt.

**\*\*** Egentligen skulle ju denna bild förstås behöva kunna “sömlöst” återupprepas i x-led för att “sömmen” som orsakas när man upprepar bilden inte ska vara uppenbar. Men detta är ok att ignorera vid denna övning. Fundera dock gärna på hur du skulle kunna uppnå en mindre uppenbar “söm”.

**Tips** Sätt din bakgrundsbild som `background` snarare än `src` i en `<img>`-tagg. Då kan du modifiera bildens `background-position` istället för att behöva flytta hela elementet. Om du har bilden som bakgrund kan du använda egenskapen `background-repeat` för att upprepa den i t.ex. x-led.