

Assignment 3: CSS (Part 2)

Part 1: Styling using pseudo-selectors (20%)

Download the provided files and open [position.html](#).

In the provided file `position.html`, make the text "CSS is awesome" into a link that points to <https://developer.mozilla.org/en-US/docs/Web/CSS>.

We want to decide for ourselves how the link must look on hover, and not use the default styles. We do not want to look like everybody else, do we? Write your code under `/* Part 1 */` in the `head` section. Use the appropriate [pseudo-selectors](#) to style the link. The link shall have a style for when it is unvisited, visited, when the mouse hovers over it, and when it has been selected.

If you want, you are allowed to style the visited and unvisited links in a similar manner.

Part 2: Styling the footer (20%)

In this part we will also give full score if the footer sticks to the bottom of the browser window. If you want an extra challenge you can complete the task as described below.

Use the provided file `list.html`. Even though our webpage does not have enough content to push the footer to the bottom, we would like it to be at the bottom. "Why?", you might ask. Because it looks silly when it is in the middle of the page.

We have set up the HTML, you have to add the CSS. Add the correct attributes and values so that the footer always is at the bottom of the page. If there is enough content to fill the page, the footer must be pushed below the content, not being visible before the user scrolls down. You are not supposed to change the HTML for this part, only the CSS!

Hint: The wrapper must at least have the same height as the browser window, and the footer must be at the bottom of the wrapper. Test your code by adding some paragraphs with text.

Part 3: List Styling (20%)

Inside `list.html`, below your header, create an unordered list with the same content as you see below.

- HTML
- CSS
 - LESS
 - Sass
 - SCSS
- JavaScript
 - React
 - Node
 - JQuery

- AngularJS
- Backbone
- Ember
- Babel

Use CSS, included as an external stylesheet, to style the list in the following way:

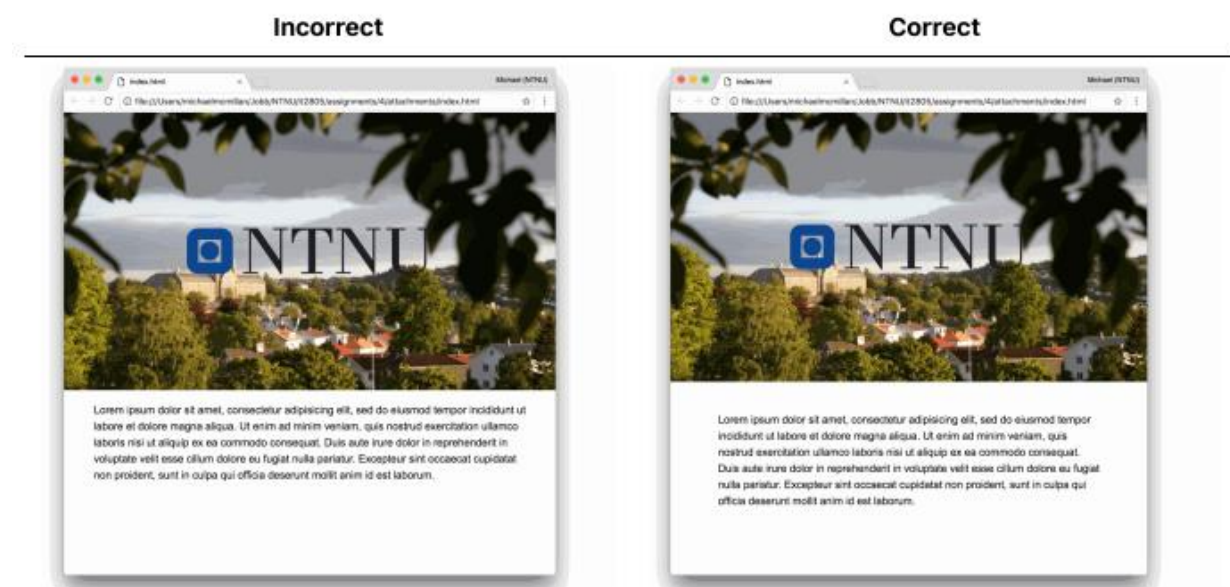
- Change the bullets into squares.
- Use **pseudo-selectors** to style the last element of a sublist differently than the other list elements. E.g., 'SCSS' and 'Babel' could have a different font color.

Part 4: Responsive design (25%)

Attached you will find an HTML file ([index.html](#)) that you are going to embed a video in. The video is supplied in three different formats: OGG theora ([hovedbygget.ogv](#)), WebM ([hovedbygget.webm](#)) and MPEG-4 ([hovedbygget.mp4](#)).

Although the size of the video element changes proportionally to the window width of the browser, the article body does not. Change the CSS so that the article always has a left and right margin of 10% of the width of the browser. The article should also be positioned in the center - but the text should stay aligned to the left.

Please note that this must be accomplished using CSS only.



Finally you are going to replace the video with a still image ([hovedbygget.png](#)) using media queries. If the visitor is using a mobile device (screen width lower than 940px) the video should be replaced with the image.

Questions (5%)

1. What is the difference between `position: absolute` and `position: fixed`?
2. With CSS3, how do we access the third element within another element?
3. What will the CSS snippet at the end of this section do if the visitor is browsing on a device with `320 x 568` resolution?

```
@media only screen
and (min-width : 768px)
and (max-width : 1024px) {
  #logo {
    display: none;
  }
}
```

Deliver the answers as a .txt file together with your other deliverables.

Deliverables

Submission should be uploaded as a zip file into Blackboard before the deadline. Submissions are ONLY accepted via Blackboard. We DON'T accept late assignments. Emails or any other messages with late assignments are automatically discarded without further communication.