

# HTML and CSS

Margit Tennosaar

# Last week

**CSS borders** // 1px solid green

**Box model** // top, right, bottom, left; padding inside, margin outside

**Universal reset** // \* { margin: 0; padding: 0; box-sizing: border-box; }

**Float** // use clear to end float in next elements

# CSS colors

**Hexadecimal colors**

`#ff0000;`

**RGB colors**

`rgb(255, 0, 0);`

**RGBA colors**

`rgba(255, 0, 0, 0.3);`

**HSL colors**

`hsl(120, 100%, 50%);`

**HSLA colors**

`hsla(120, 100%, 50%, 0.3);`

**Cross-browser color names** `green`

# CSS backgrounds

**background-color**

**background-image**

**background-repeat**

**background-attachment**

**background-position**

**background-size**

**background:** background-image

background-repeat background-

attachment background-position

background-size

# Fonts

# Types

- **Serif fonts** have a small stroke at the edges of each letter. They create a sense of formality and elegance.
- **Sans-serif** fonts have clean lines (no small strokes attached). They create a modern and minimalistic look.
- **Monospace fonts** - here all the letters have the same fixed width. They create a mechanical look.
- **Cursive** fonts imitate human handwriting.
- **Display** fonts are decorative/playful fonts.

# Font in CSS

font-family

font-size

font-weight // normal, bold, 100-900

line-height // 1; 1.2; 2;

# Fallback font

A **fallback font** is a reserve **typeface** containing symbols for as many Unicode characters as possible. When a display system encounters a character that is not part of the repertoire of any of the other available **fonts**, a symbol from a **fallback font** is used instead.

```
font-family: 'Lato', sans-serif;
```



Watch the video on how to add Google Fonts:

<https://youtu.be/E7QPO8P8nls?si=JKQIF-O0fE0gg1EN>

# Pseudo classes

# Pseudo class

A CSS pseudo-element is used to style specified parts of an element but only when it is in a **certain state**.

:active	:lang(language)	:only-child
:checked	:last-child	:only-of-type
:disabled	:last-of-type	:optional
:empty	:link	:out-of-range
:enabled	:not(selector)	:read-only
:first-child	:nth-child()	:root
:first-of-type	:nth-last-child()	:target
:focus	:nth-last-of-type()	:valid
:hover	:nth-of-type()	:visited
:in-range		
:invalid		

# Pseudo class – most used example

**a:hover** MUST come after **a:link** and **a:visited** in the CSS definition in order to be effective

**a:active** MUST come after **a:hover** in the CSS definition in order to be effective!

Pseudo-class names are not case-sensitive.

```
/* unvisited link */  
a:link {  
    color: #FF0000;  
}
```

```
/* visited link */  
a:visited {  
    color: #00FF00;  
}
```

```
/* mouse over link */  
a:hover {  
    color: #FF00FF;  
}
```

```
/* selected link */  
a:active {  
    color: #0000FF;  
}
```

# Pseudo classes and childs

```
<ul>  
  <li><a href="#">Home</a></li>  
  <li><a href="#">CV</a></li>  
  <li><a href="#">Contact</a></li>  
</ul>
```

```
li:first-child{  
  background-color: blue;  
}
```

# Some tricks

`p:nth-child(2)`

`p:nth-child(even)`

`p:nth-child(2n)`

`p:nth-child(2n+4)`

# Pseudo elements

A CSS pseudo-element are like virtual elements that we can create in CSS. They don't exist in DOM.

::after

::before

::first-letter

::first-line

::selection

::backdrop

Watch the video about pseudo-classes:

<https://youtu.be/e1KpKBHJOrA?si=kZdwI3H8IJSmXktX>



