# Hodina 22.3.

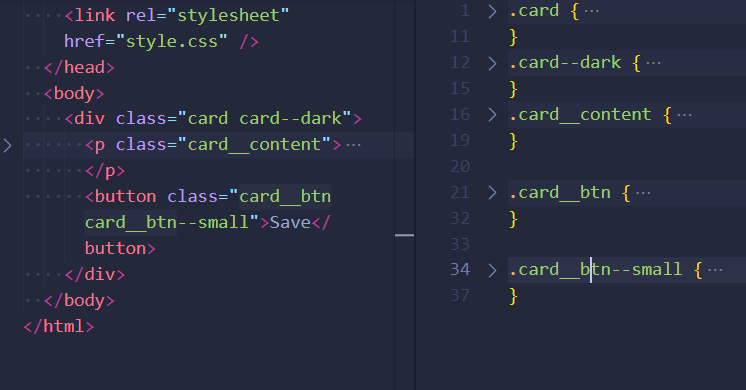
## Osnova

* Opakování – Bem, Sass, Bootstrap – ukázky
* Selektory – kombinátory, specificita. výklad, kvíz a cvičení

BEM

způsob, konvence, logika, jak se pojmenovávají třídy v CSS, aby se nepletly. Zároveň nám název třídy indikuje, co přidáním třídy nastylujeme.

Nic k tomu není potřeba. Stylování si musíme napsat sami



Popsat a ukázat stylování

Sass

CSS preprocesor (jeden z mnoha, máme i další jako Less, )

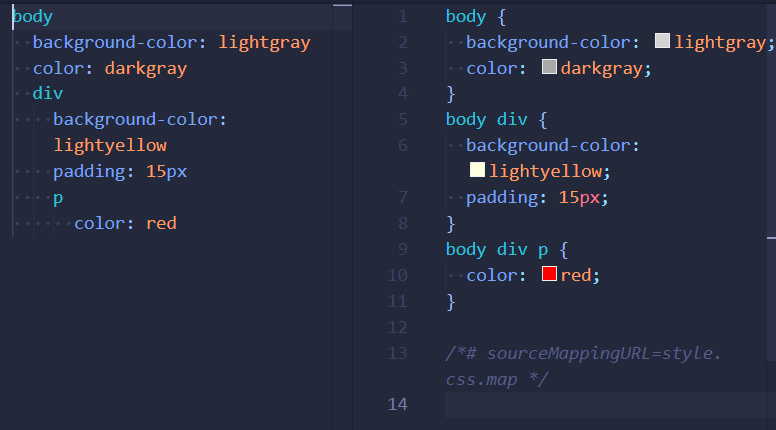
* umožňuje vývojářům používat věci, které jazyk CSS sám neumí.
* Preprocesor umožňuje zapsat CSS kód novým způsobem, čímž mu nabízí větší možnosti
  + využívání proměnných, funkcí, zanořování atd.
* Hotový kód pak preprocesor zpracuje, tj. udělá z něj standardní kód jazyka CSS, se kterým umí pracovat váš webový prohlížeč (Internet Explorer, Chrome, Firefox).
* Samotnému preprocesorovému zápisu by totiž prohlížeč jinak nerozuměl.

Sass je potřeba nainstalovat, aby dokázal naše stylování „přechroustat“ do css.

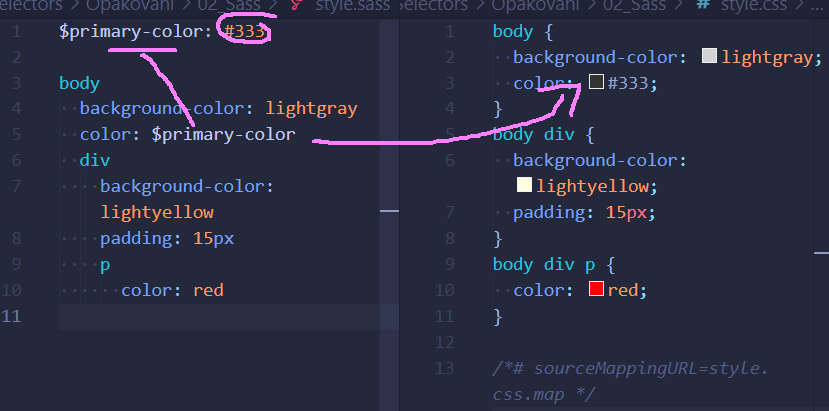
Terminál: **Sass --version** – mám ho

Potom ve složce mám obojí style.sass, style.css –

**sass --watch style.sass:style.css spustím tento příkaz a to, co budeme psát v sassu, se nám přeprocesuje, přepíše, přetransformuje do css.**toto css nalinkuji na html, ne sass!

V css.map – vidíme nějaké další informace. ****

Je možné používání proměnných – uložíme si nějakou hodnotu a pak ji můžeme používat v našem celém souboru.

****

Nevýhodou je, že se s tím občas dá až moc kouzlit a není nějaký univerzální návod. Kód může být tak hůř čitelný a spravovatelný.

V CSS je potřeba rozumět, jak se jednotlivé prvky stylují.   
zatím známe stylování elementů, tříd, možná id.   
Pro dnešní hodinu je potřeba si připomenout pseudotřídy   
:hover, :focus   
atributy   
<https://www.w3schools.com/css/css_attribute_selectors.asp>  
dokážeme vybrat prvky podle toho, jaké mají atributy (končí, začínají, obsahují, neobsahují nějakou hodnotu)

Můžeme přidat prvkům id, to je unikátní.   
  
  
Selektory   
Jak to udělat, pokud máme spoustu paragrafů nebo nadpisů?  
není potřeba na každý jednotlivý prvek přidávat třídu, ale můžeme chytrým použitím selektorů vybrat ten správný prvek, nebo prvky.

1. **.review, .title** Nejčastější, co jste už asi viděli, je **vícenásobný selektor**. Takto vybereme všechny prvky p a všechny div. Na specificitu to nemá moc vliv, ale je dobré tento selektor znát.
2. Často mají prvky více tříd. Pokud bychom vybírali takový prvek, píšeme selektory hned za sebou bez mezery pokud. Používáme tag, tak musí být první.   
   **div.comment**  **.comment.recent**
3. **div p** pokud bychom rádi vybrali všechny elementy p, které se nacházejí uvnitř divu.
4. další možnost je použití zápisu div > p   
   toto vybere p, kteří jsou přímí potomci divu
5. Přímý sourozenec h2 + p
6. N-tý potomek div:nth-child(3)
7. Každý n-tý potomek div:nth-child(3n)
8. div:nth-child(3n+2)

Pokud všechno stylujeme pomocí tříd   
1. co je níž, to platí – css se čte odshora dolů – mají stejnou specificitu (poslední vyhrává)   
  
Co se ale stane, pokud jednou budeme stylovat nějaký prvek pomocí tagu a potom pomocí třídy?   
V tom případě stylování pomocí třídy „přebije“ stylování elementu. Třída má totiž větší specificitu.   
  
Specificita se v CSS totiž počítá takto:   
každý prvek má specificitu 1   
třída, pseudotřída nebo atribut má specificitu 10   
ID má specificitu 100

Takže pokud stejnému prvku přidáme ID, které nastylujeme, přebijeme i třídu, protože máme najednou id o 10x vyšší specificitě než je třída a 100x vyšší specificitě než pouhý tag.

A jelikož už víme, že můžeme vybrat jeden prvek pomocí více selektorů, můžeme je prostě začít sčítat!