LEKCE: Margin a padding

Tak trochu opakování základů, ale také hodně nové látky pro hlubší pochopení toho, jak funguje margin a padding.  
  
Dozvíš se něco o **collapsing margins** i o tzv. **box-sizingu**, který zásadním způsobem ovlivňuje, jak se počítá šířka daného prvku.  
  
Jako **dobrovolný miniúkol** je v podkladech samostatné cvičení na nastavování marginu a paddingu.

Výpisky z videa

* Margin a padding jsou rozestupy (primárně)
* Odstavci nastavím rámeček a primárně by byl těsně kolem textu, pokud chci, aby nebyl ten text nacpaný přímo na rámečku, můžu nastavit padding (polštářek, rozestup mezi obsajem prvku a rámečkem prvku – směrem dovnitř prvku
* Margin je naopak rozestup od rámečku směrem ven
  + Rozestup mezi odstavcem a okolními prvky
* Margin je osobní zóna prvku = NIKDO KE MNĚ NESMÍ JÍT BLÍŽ NEŽ 20PX
* Padding je rozestup směrem doprostřed
  + Padding – čtyři hodnoty – nahoře, vpravo, dole, vlevo (po směru hodinových ručiek)
  + Když napíšu dvě hodnoty, tak se ta první aplikuje nahoře a dole a ta druhá se aplikuje vpravo a vlevo
  + Ty dvě hodnoty se používají často
* Margin taky dělá prázdné místo, ale směrem ven od rámečku, rozestup od okolních prvků
* Pletou se proto, že oba nastavují stejnou věc = prázdné místo, v určitých případech je možné nastavit jenom jednu tuto vlastnost a vyjde to „nastejno“
* Ctrl+/ = zapne se nám komentář v CSS
* Všechny značky mají nastavené nějaké vlastnosti jakožto výchozí hodnoty – vidíme právě ve vývojářských nástrojích

Collapsing margins

* Odstavce mají výchozí margin
* Margin směrem dolů a margin směrem nahoru jsou „přes sebe“ – není tam jeden a pod tím druhý, ale jsou přes sebe (vidíme ve vývojářských nástrojích)
  + Tomu se říká collapsing margin
  + V určitých případech se marginy nesečtou, ale jsou „přes seb“
* Dochází k tomu u blokových prvků (POUZE U BLOKOVÝCH PRVKŮ)
* Ve chvíli, kdy máme dva „sourozence“ = prvky na stejné úrovni, pod sebou, např. p aul, které jsou vnořeny v jedné úrovni (v body) X li není sourzenec p, protože li je vnořeno do u.
* Prolínají se tak, že se použije větší z marginů
* Pokud mají prvky marginy směrem nahoru a dolů a zároveň jsou uvnitř boxu a mezi boxem a prvkem není žádný rámeček nebo žádný padding, tak se ten margin vystrčí z boxu ven
* Pokud by box měl rámeček nebo padding, tak by margin netrčel z boxu ven, ale bude uvnitř boxu
  + Proto musíme udělat nějaký rámeček, aby nám margin netrčel ven (vidíme ve vývojářských nástrojích nebo padding
  + Budou se nám dělat mezery mezi boxy (sekcemi)
* Pokud se nám budou dělat mezery mezi sekcemi (divy s nastavenou barvou pozadí)
  + Je třeba udělat padding, aby se nám odsunul nadpis v rámci druhého boxu
  + Margin zničit a nastavit padding boxu dole
* 2 způsoby – přidat border nebo přidat padding
* Když jsou dva prvky pod sebou a jeden má nějaký margin, druhý má větší margin, tak ty marginy se prolnou a použije se větší z těch čísel
* Pokud bychom měli box bez borderu a paddingu, budou marginy trčet ven
* Centrování margin auto
  + Pokud mám box, který nezabírá celou stránku, například pouze 60 % stránky, tak pokud použijeme hodnotu text-align, užije se pouze na text v rámci bloku, ale nikoli na blok celý
  + Margin-left a margin-right mají zvláštní hodnotu auto, která nám udělá to, že i když box zabírá 60 % stránky, tak použitím text-align vytvoříme box, který zabírá 60 % stránky, ale zároveň je „veprostřed“ a testy v něm jsou taky „veprostřed“
  + Nefunguje to nahoru a dolů (vertikálně), ale funguje to horizontálně – vpravo, vlevo
  + Centrujeme tak blokové elementy horizontálně
  + Zápis **margin: 0 auto**;
* Box-sizing
  + V příkladu jsou divisions s třídami „box contentbox“ a „box borderbox“
  + Když zadám pouze třídu .box, tak styluji oba boxy najednou, když konkrétní „podtřídu“, tak konkrétní box
  + Zadal šířku 600px, ale reálně má boxík šířku 750px, souvisí to s tím, jak stránka počítá šířku
  + 600px zabírá obsah uvnitř, ale box má nastavený ještě padding 50px na všechny strany plus rámeček 25px
  + Ve výchozím nastavení nastavujeme šířku obsahu prvku a pokud máme ještě padding a border, šířka se přičte
  + Vlastnost box sizing = změníme chování toho, čemu nastavujeme šířku
  + Výchozí hodnota box-sizing: content-box
    - Když to nastavíme, tak se to chová přesně tak, jak známe – přičítá se border a padding
  + Hodnota box-sizing: border-box
    - Udělá nám to, že automaticky změní celý rámeček, včetně paddingu a borderu, nastaví na šířku, kterou chceme
  + \* = univerzální selektor = aplikuje se na všechny elementy
  + Mohu takto nastavit všem prvkům modernější, intuitivnější chování pomocí přiřazení vlastnosti box-sizing: border-box
    - **DOPORUČUJE NÁM, ABYCHOM TO MĚLI NASTAVENO VŽDY NA ZAČÁTKU, VŠEM PRVKŮM TO TAKTO NASTAVIT**