

Tvorba moderního rozložení 1

Úvod

Následujících několik hodin se začneme bavit o moderních webových stránkách, společně s tím začneme již plně využívat všech nabytých zkušeností. Pokud v něčem stále máte nedostatky, doporučuji si připravit otázky. Hlavním tématem těchto bloků bude především tvorba moderních webových rozložení. Doteď znáte zarovnání doprostřed a různé další zastaralé možnosti.

Historie rozložení

Když se začaly tvořit první webové stránky na internetu, nebylo zcela jasné, co bude pro tvoření potřeba. Někteří z Vás si jistě mohou vzpomenout na časy, kdy celé stránky byly zarovnané k levému okraji monitoru či tabulkové rozložení. Jak jsem již zmiňoval, dříve se celé webové stránky stavěly z vnořených tabulek. V dnešní době - především díky mnohým přenosným zařízením a lenosti uživatelů - je tento způsob velmi špatný.

Víte proč jsou tabulky špatné?

Tabulky jsou špatné z velmi jednoduchého důvodu - jsou jednoduše hloupé. Tabulky se mění podle toho, jaký obsah obsahují a již víme, že každé zařízení (počítač, notebook, mobil, tablet, ...) má jinou šířku zařízení. Tabulky se pro tyto zařízení chovají nedefinovaně a je mnohdy těžké aby dělali to, co chceme.

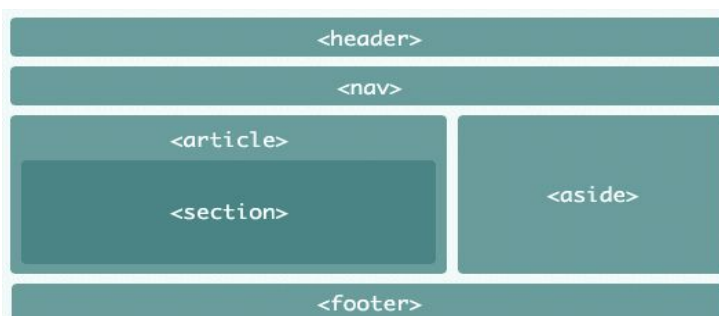
Tabulky jsou také špatné v tom, že byly určeny na "seznamové" věci či výpisy, to však stránka není, a víme, že webové stránky se dělí do sloupců a do různých sekcí a ty jsou někdy odlišené.

Z tohoto důvodu nebudeme pro tvorbu webových stránek využívat tabulky. Tabulky můžeme využívat např. při výpisu skóre, seznamu zákazníků, ale i je lze obejít (a usnadnit si práci). Dalším problémem tabulek je to, že se špatně nastavují pomocí responzivity, což se budeme učit v následujícím bloku.

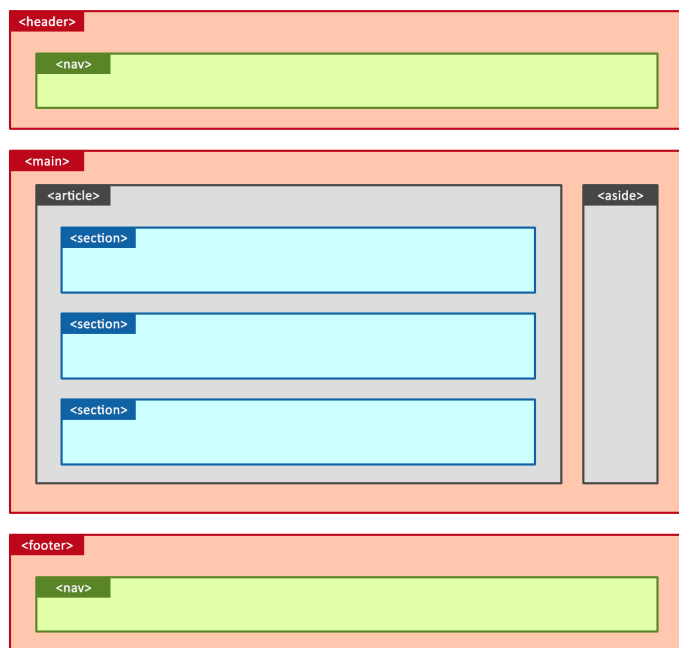
Moderní struktura webových stránek

Moderní webové stránky se většinou dělí do pěti hlavních tagů: header, nav, main, footer, sections, article.

Předtím, než Vám povím na co tyto tagy slouží, bylo by dobré si uvědomit proč na stránkách vůbec nějakou takovou strukturu máme mít. Tato struktura zajišťuje že bude logicky stránka postavena do bloků, které se sebou



souvisí - pomáhá to logicky postavit stránku. Tyto bloky lze stylovat tak jak je potřeba a není nutné se limitovat již na známe rozložení. Tato struktura identifikuje k čemu blok slouží - v případě bez této struktury to stroj nebude moci pochopit.



Webová stránka - tedy tag body, se v moderních webových stránkách rozděluje do tří hlavních tagů - a to header, který značí úvodní blok, který obsahuje například jméno, logo, navigaci, main - který obsahuje obsah stránky, který se na stránkách mění. Poslední je footer, který obsahuje například copyright text, rychlou navigaci, kontakt, nebo autora webové stránky.

O tagu main je potřeba znát více věcí - jak jsem zmiňoval, je to reálný obsah stránky, který se vůči ostatním (HTML) souborům mění. V tomto tagu můžeme nalézt tag aside, který značí postranní - méně důležitý text. Můžeme si tam

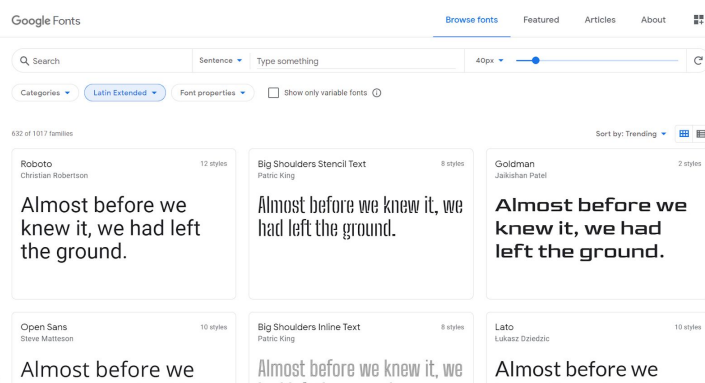
představit například autora stránky, tlačítka ke sdílení obsahu, metadata příspěvku a mnoho dalšího. V tagu main se má také nacházet (několik) article, který značí nějaký typ obsahu, v něm se na víc má používat tag section, který značí sekci příspěvku. Často však narazíte, že se article využívá opravdu pouze k nám známým článkům a že section se využívá bez něj. Je to samozřejmě možné. Na stránce se může nacházet několik navigací - menu.

Webové fonty písma

Již jsme si zmiňovali, jak nastavit font písma - pomocí vlastnosti font-family, nyní si ukážeme jak na stránku vložit webový font, který získáme na stránce [Google fonts](#). Na této stránce si můžete jednoduše najít tisíce fontů, které se Vám mohou hodit kdekoli na stránce. K vyhledání pouze fontů, které podporují české znaky, je potřeba vyhledávat jazyk "Latin extended", které tyto znaky obsahuje.

Při zvolení fontu a jejich tlustoty je možno font stáhnout, nebo rovnou vložit na webovou stránku. Při získání odkazu jej stačí dát do hlavičky stránky v tagu head a při použití font-family již bude možno zadat dané jméno.

Je důležité si uvědomit, že fonty nejsou malé soubory a při tvorbě webové stránky jste velmi limitováni počtem použití. Optimální je mít cca. dva fonty, jeden na obyčejný text, jeden na



nadpisy a s nějakým počtem tlouštěk a variací. Při importování fontů si rozmyslete zda není lepší použít již použité fonty. Při speciálním fontu, například na logo, je lepší logo vytvořit v rozumné velikosti v programu jako obrázek s daným fontem a poté používat jen fotografii. Fonty jsou podobné fotografiím a jejich použití si je důležité rozmyslet. Na druhou stranu je nutné počítat s tím, že uživatel webových stránek bude mít daný font nainstalovaný na počítači, je naivní a proto je lepší vždy použít webový font.

Review

Roboto

Regular 400

Black 900 italic

[Add more styles](#)

[Remove all](#)

Use on the web

To embed a font, copy the code into the <head> of your html

☒ <link> ☐ @import

```
<link href="https://fonts.googleapis.com/css2?family=Roboto:ital,wght@0,400;1,900&display=swap" rel="stylesheet">
```

CSS rules to specify families

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

Flexbox

Flexbox je první reálný nástroj pro rozvržení webových stránek, který se budeme učit. Naše znalosti spojíme s zarovnáním pomocí margin a hodnoty auto.

V první řadě si je potřeba uvědomit že flexbox je náročný nástroj, který bude stát trochu Vašeho času, ale v případě pochopení Vám usnadní práci natolik, že se již nebudete žádného rozložení bát. Pokud něco nepochopíte, připravte si na konzultaci otázky a já se pokusím Vám problematiku znovu vysvětlit.

První krůčky

K tomu, aby se blok (většinou tag div) choval podle zadaného chování, je potřeba mu nastavit speciální zobrazování a to pomocí tagu display a hodnoty flex. Každé přímé dítě tohoto rodiče se bude automaticky zobrazovat jako blok.

```
.block {  
  display: flex;  
}
```

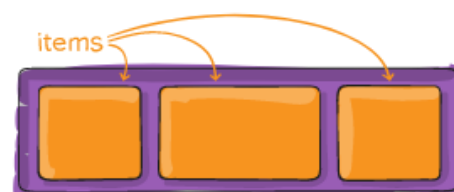
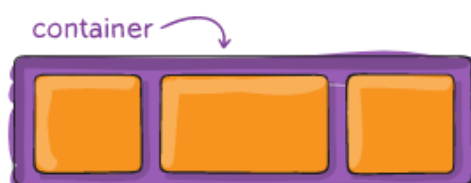
Základní chování tohoto rozložení je to, že nemůže přetéct na další řádek - existuje pouze jedna osa a to osa X. Chceme-li něco takového povolit, musíme nastavit vlastnost flex-wrap na hodnotu wrap. Lze i nastavit, jakou osou se budou prvky řadit,

```
.block {  
  display: flex;  
  flex-direction: column;  
}
```

jak jsem řekl, základní hodnota je osa X - řádek - row, druhou možností je osa Y - sloupec - column. Tu nastavíme pomocí vlastnosti flex-direction a hodnoty column.

```
.block {  
  display: flex;  
  flex-wrap: wrap;  
}
```

Ve flexboxu se často mluví o tzv. kontejneru (container) a předmětech (dětí, childs, items). Je důležité si uvědomit, o co vlastně jde. Z obrázku by vše mělo být jasnéjší.

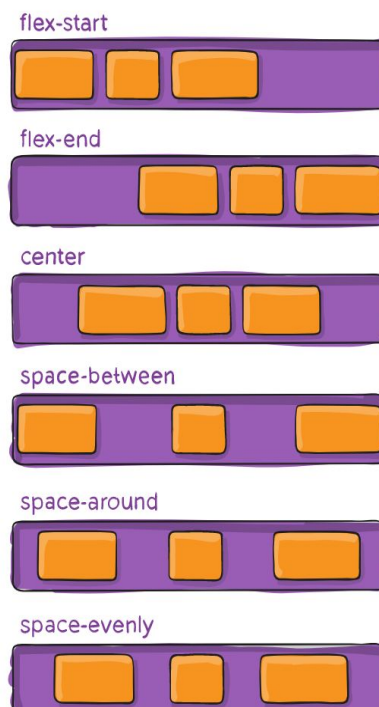
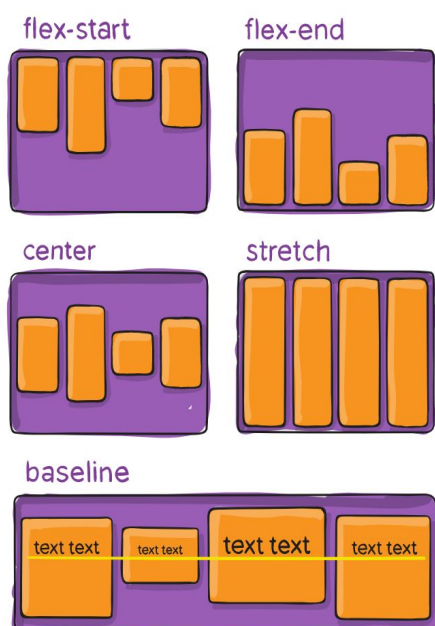


Kontejner

Základní další vlastnost kontejneru spojená s flexboxem je vlastnost, která zarovnává předměty na hlavní ose (podle flex-direction). Jmenuje se justify-content a její hodnoty lze vidět na levo v obrázku. Toto zarovnání se v případě povoleného přetékání - více řádku, přizpůsobí na každý řádek.

Další důležitou vlastností je zarovnání vůči druhé osy (znovu podle flex-direction), obvykle na sloupci. Vlastnost

se jmenuje align-items. Toto zarovnání má odlišné hodnoty než justify-content a znovu je můžete vidět na přiloženém obrázku vlevo.



Tyto dvě vlastnosti se samozřejmě dají kombinovat a jejich kombinace Vám s vytvořením rozložení velmi pomůže. Nesmíte zapomínat již na znalosti, které jsme od začátku roku získali - nejvíce odsazení jak uvnitř tak i venku nebo například vlastnosti width.

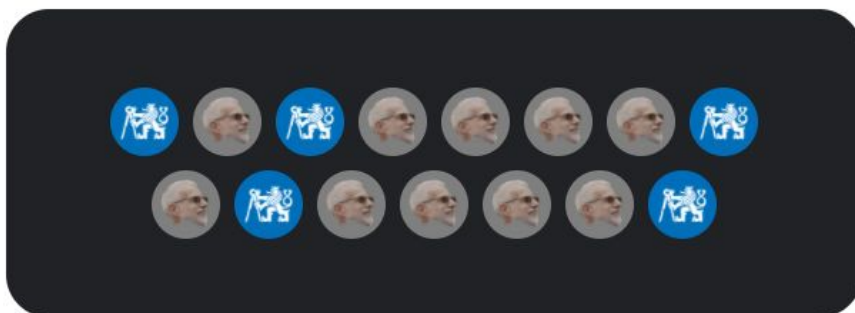
Pokračování

S rozpořádáním (layoutem) různých prvků budeme pracovat především na nadcházející konzultaci a i následující hodiny. Vynechal jsem spoustu vlastností, které jsou s flexboxem spojené ale používají se méně často. Jako povinný, ale nekontrolovaný úkol máte přečíst [tuto stránku](#), kde naleznete spoustu zajímavých informací o flexboxu. Obrázky, které jsem použil, jsou také z této stránky.

Procvičení

K procvičení této znalosti máte si prosím zkuste vytvořit webovou stránku k následujícímu obrázku. Pro procvičení se podívejte na zdrojový kód [ZDE](#).

Všichni známe Discord - proto jsem jako cvičení vybral volbu serveru. Stačí napodobit rozložení. Pořádní fajnšmekři mohou při najetí na server změnit tvar kruhu na čtverec s zaoblením.



(Discord servery nepředstavují žádné reálné servery)

Další procvičení

Další procvičení buď dělejte sami (zkuste replikovat nějaké webové stránky) a nebo počkejte na naší konzultaci, kde toho budeme spoustu řešit a budu Vám sdílet své kódy. Tato látka je poměrně složitá, ale určitě to společně zvládneme.

Další nové CSS vlastnosti

Mimo vlastnosti, které jsou spojené s flexboxem, Vám dnes ukážu vlastnost `overflow`. Overflow česky znamená přetečení a mnohým z Vás je již jasné co dělá - její hlavní účel je určit chování, když daný element přeteče na jakékoliv straně. Základní hodnota je `visible`, která vykreslí vše mimo. Hodnota `hidden` naopak přetékající věci skryje na velikost rodiče (bylo např. potřeba na procvičení na začátku této stránky). Hodnota `auto` se chová automaticky podle toho, zda opravdu něco přetéká, popř. zapne skrolovací lištu. Tu lze zapnout také pomocí hodnoty `scroll`.

```
.servers .server {  
  overflow: hidden;  
}
```

Promluva do duše 😂

Ne, opravdu, teď trochu vážně. V minulém týdnu se v domácím úkolu objevilo v obou třídách několik duplikátů úkolu s pouze změněnými barvami a s přeházeným kódem. V tomto úkolu (i předešlých, ale tam jsme to ještě toleroval) to samozřejmě bylo zakazané. Jedinou výjimku měli lidi, kteří jasně specifikovali s kým spolupracovali popř. kdo mu s čím pomohl.

Moje pozice učitele tohoto, velmi oblíbeného, předmětu je *zvláštní*. Spousta z Vás již s webovkami umí pracovat a tedy můj pomoc nepotřebují, ale je tu také spousta lidí, kteří do začátku roku nevěděli co webová stránka je. Chtěl bych, abyste si uvědomili že jsem tu pro Vás a snažím se Vás naučit co nejvíce, abyste si popřípadně mohli v budoucnu vybrat právě webové stránky jako svojí práci, pokud to nechcete, je to samozřejmě normální, ale v IT jsou webové stránky základem spousty odvětví a znalost alespoň základu HTML jsou nezbytná k jakékoliv práci. Většina dokumentací, frameworků má právě vytvořené webové stránky a v

nich se budete muset orientovat. Na konci roku začneme pracovat s programovacím jazykem JavaScript, který je aktuálně nejpoužívanější v IT vůbec.

Tyto znalosti zajisté ještě v životě použijete a je hloupé, když máte možnost studovat na této velmi dobré, prestižní, střední škole, zahazovat svojí budoucnost.

Vaše podvádění mě mrzí, ale věřím že kořeny tohoto problému jsou v tom, že jste něco neuměli. Prosím, přestaňte se bát ptát a zkuste si něco dělat sami. Moje práce je Vás naučit základy a pomáhat Vám. Když se budete ptát, vy se naučíte a já si budu připadat tak, že jsem Vám užitečný a náš vztah bude jenom lepší. Nezlobím se že jste to udělali, jen Vás prosím, už to nedělejte. Radši odevzdejte Vaší, nefunkční práci, kterou Vám zadarmo opravím a vy pochopíte v čem jste měl problém. V nabalování Vaší neznalosti nikomu nepomáháte a postupně se vše bude nalepovat do té doby, než nebudete chápat nic a případně budete muset jít na komisionální přezkoušení. Tomu se říká snowball efekt a platí v každém odvětví.

Proč nepodvádět? To je dobrá otázka na dlouhé zimní večery, moje odpověď je však opravdu jednoduchá: přetvářka. Svým podváděním předvádíte učiteli to, že všemu perfektně rozumíte a on nevědomky bude přidávat na rychlosti či bude zvyšovat náročnost předmětu. I když může být problém na straně učitele, je nutné mu to říct. Zjednodušte škodíte jak sobě, tak protějšku a to nikdy není dobře. Spousta z Vás chce jít na vysoké školy a věřte, že lehké pochopení nebo podvádění znamená jednosměrnou letenku z Vaší vysněné vysoké školy. Takto to nefunguje ani v životě, někdy prostě nebudete mít “od koho opsat” a budete pouze na svých nohou. Znalosti, které tento předmět požaduje, jsou minimální oproti jakémukoliv jinému na této škole.

Naučit se odborný předmět někdy nemusí být těžké, ale jak se říká, “bolest přináší ovoce”, k naučení jakékoliv věci je nutné se na věc soustředit, snažit se a pracovat. Bez tohoto nepůjde. Doporučuji si najít projekt (tj. webovou stránku), který chcete mít, a tu postupně dělat a zkoušet si věci z hodin.

To by asi bylo vše, děkuju všem, co si tohle přečetli, moc si toho Vážím.

TLDR: Ptejte se, nebojte se, jsem tu pro Vás, webovky jsou sranda a já jsem hodný, nekoušu, zeptejte se mých kocourů!

Konzultace

Na následující hodinu si do VSCode **prosím nainstalujte plugin Live share**, který Vám umožní, abyste mi rovnou pomáhali vytvářet skvělé webové stránky. Za to můžete dostat body za aktivitu, které tak potřebujete.

Nevíme, kdy se do školy vrátíme a proto Vás nezdržuji testy, Vaše pololetní známka bude tvořena z aktivit z domácích úkolů, dostaneme-li se do školy do půlky ledna, budeme psát větší test, který ověří že znáte to, co máte. Ale toho je zbytečné se takto brzy obávat. Vše se včas dozvíte.

Z Vašich reakcí na různých platformách jsem pochopil že Vás Kahoot bavil, Kahoot od nyní budu aktivně přidávat do svých hodin. Kahooty překvapivě berou mnoho času na

vytvoření (a jak jste sami viděli, někdy formulovat otázku je nadlidský úkol), proto budou méně časté.

Úkoly

V této části látky se samotným použitím daných rozvržení budeme soustředit na tvorbu moderních webových stránek. Při online konzultacích budeme řešit různé příklady rozložení a postupně je budeme přidávat do úkolů. Na tento úkol je výjimečně více času, protože na online konzultaci budeme ukazovat spoustu příkladů, které se Vám budou k vytvoření úkolu hodit.

Vytvořte webovou stránku dle přiloženého obrázku

(max. 6 bodů, bonus: 10 bodů, min. 0) SPOLUPRÁCE/PODVÁDĚNÍ ZAKÁZÁNO



(Fotka je k dispozici ve větší kvalitě [ZDE](#))

Zkuste se co nejvíce přiblížit obrázku, obrázky ani barvy s textem nemusí být stejné. Jedná se o celostránkovou stránku, na pozicování použijte flexbox.

Pro obě třídy: **ÚKOL SE MUSÍ ODEVZDAT DO STŘEDEČNÍ (11. 11) 20 HODINY. Na pozdní odevzdání není brán ohled.**