

Flexbox

Michal Kučera



Použití materiálů

Toto dílo je licencováno pod

Creative Commons Attribution-NonCommercial-ShareAlike 4.0
International License



Co je Flexbox

Flexbox



- nástroj pro tvorbu layoutu - rozložení prvků na stránce
- v CSS pravěku se k tomu účelu používaly tabulky nebo floaty
- ani jedno z toho ale nikdy nebylo zamýšleno pro tvorbu layoutu stránky
- Flexbox = webdesignerský dar z nebes! Konečně máme vhodný nástroj navržený pro layout.

Flexbox



- dává věci vedle sebe (resp. podél jedné osy - buď vedle sebe do řádku nebo pod sebe do sloupce)
- přiděluje prvkům uvnitř flexboxu prostor podle nastavených pravidel
- flex = pokud to dovolíme, tak umí prvky smršťovat nebo natahovat tak, aby se vešly vedle sebe

Flexbox



HTML

```
<div class="kontejner">
```

```
  <div>Potomek 1</div>
```

```
  <div>Potomek 2</div>
```

```
</div>
```

Flexbox se aplikuje na rodičovský prvek (**flex container**).

Všechny prvky uvnitř se stávají položkami flexboxu (**flex items**).

Flexbox v CSS

CSS

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

Jednodušší už to být nemůže!

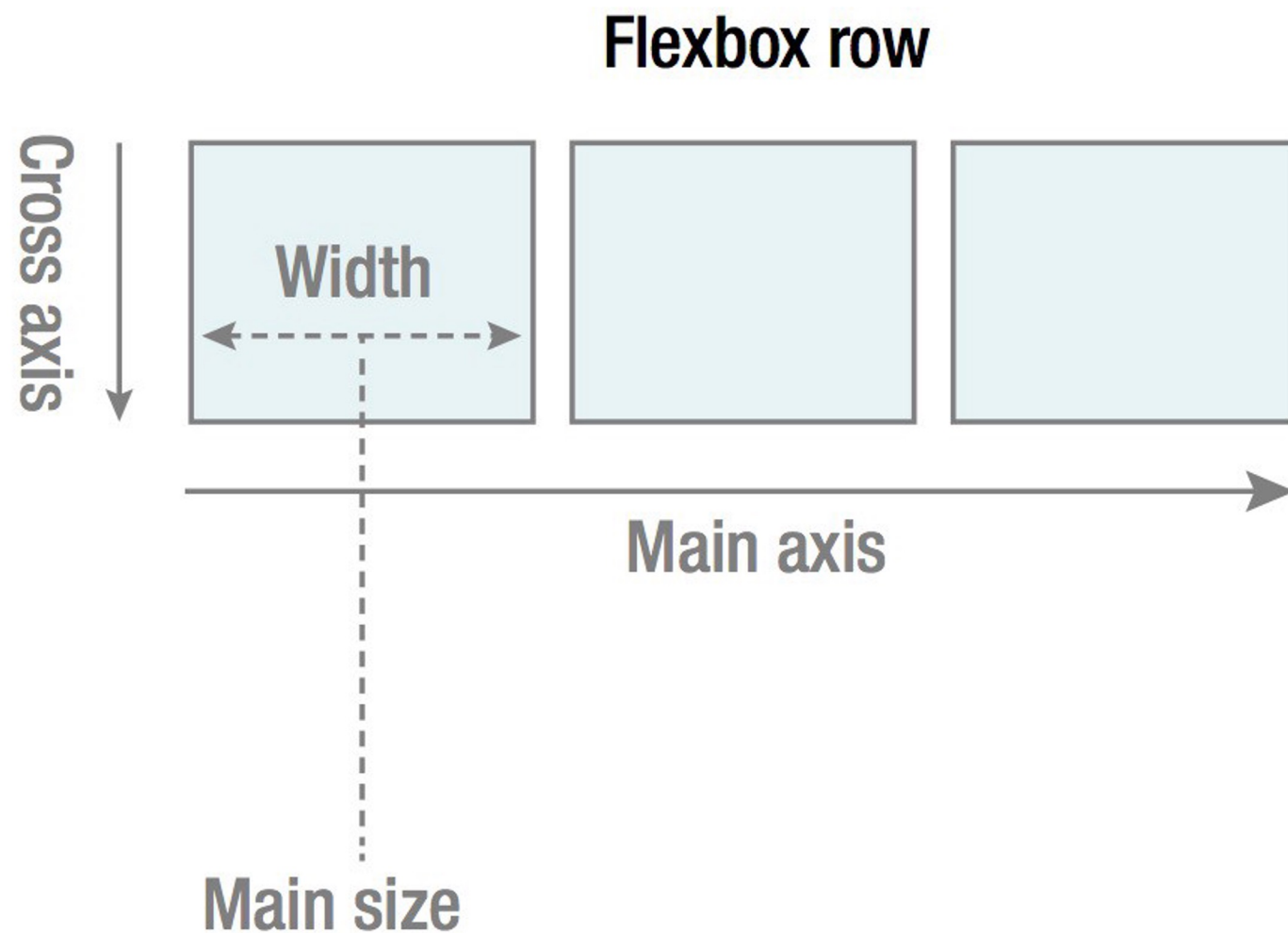
Flexbox - rozmisťuje prvky v jednom smereu 



Flexbox - rozmisťuje prvky v jednom směru

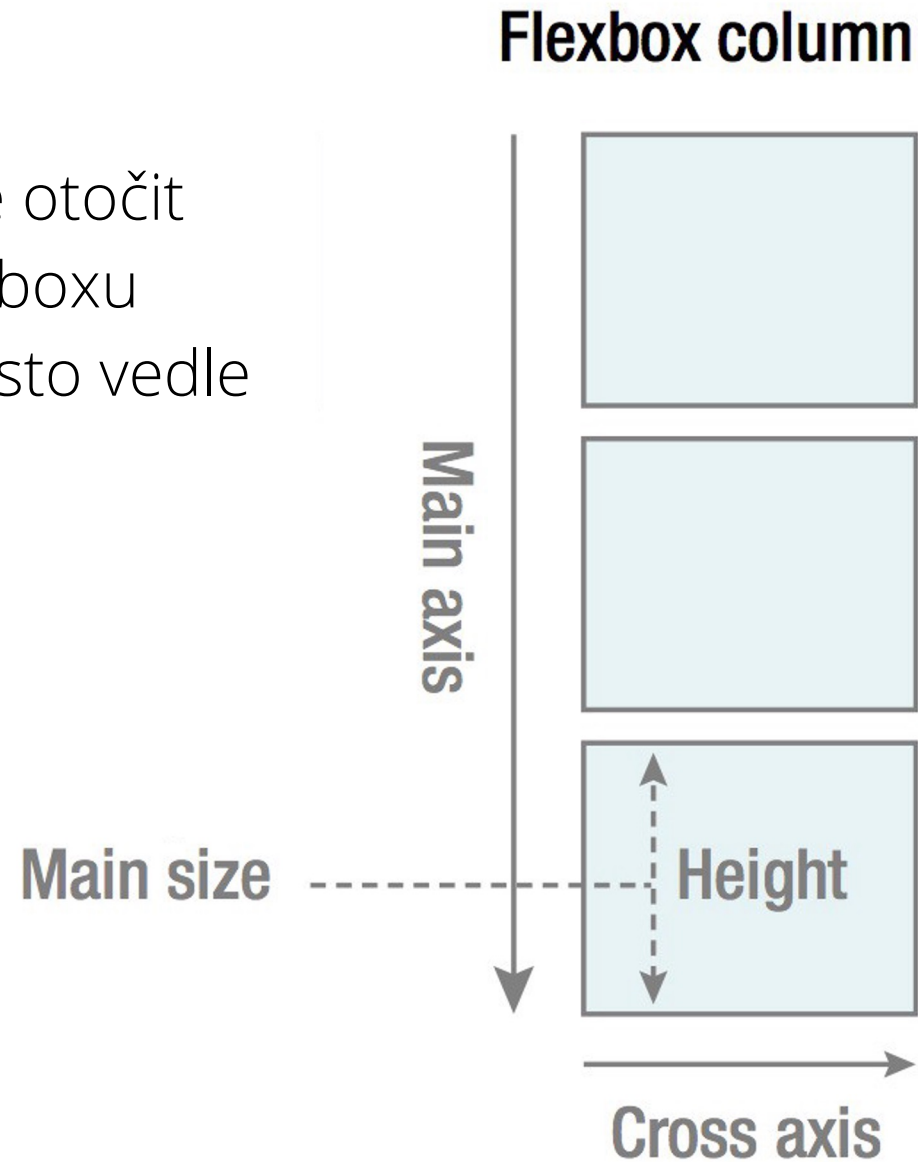
- flexbox má dvě osy
- hlavní osa - **main axis**
 - ve výchozím nastavení vede zleva doprava
 - podél hlavní osy se prvky ve flexboxu rozmisťují
- vedlejší osa - **cross axis**
 - je kolmá na hlavní osu
 - ve výchozím nastavení vede shora dolů
 - ve směru vedlejší osy jsou prvky zarovnávat

Osy flexboxu



Osy flexboxu

- směr hlavní osy jde otočit
- pak se položky flexboxu dávají pod sebe místo vedle sebe



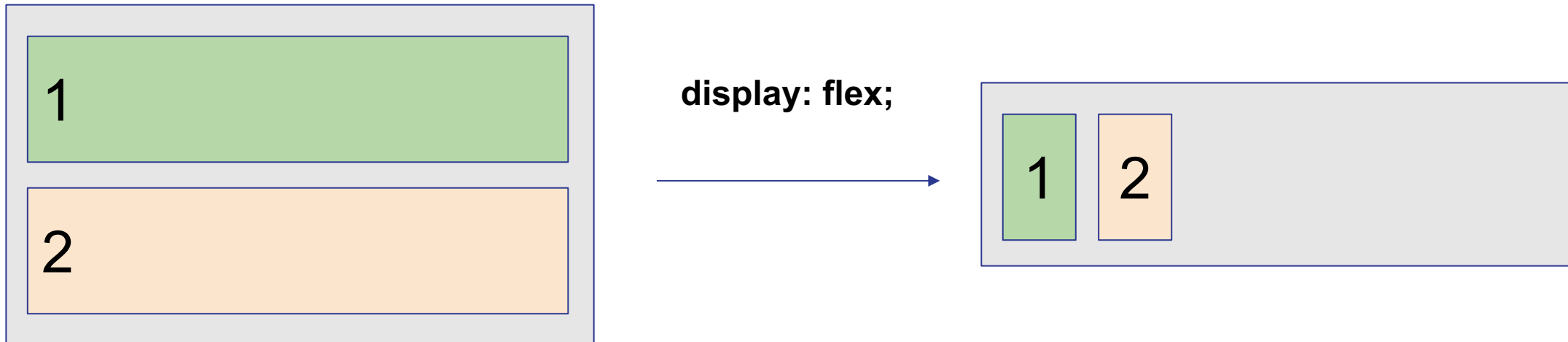
Flexbox kontejner a položky



- vlastnosti flexboxu se dělí na ty, které nastavujeme
 - kontejneru
 - položkám flexboxu
- flexbox zapínáme vždy na kontejneru pomocí `display: flex;`

Velikost položek ve flexboxu

- když zapneme na kontejneru flexbox, všechny položky uvnitř se dají vedle sebe a smrsknou se na velikost podle jejich obsahu



Velikost položek ve Flexboxu

Velikost položek ve flexboxu

Velikost položek ve flexboxu se řídí pomocí tří vlastností:

- flex-grow
- flex-shrink
- flex-basis

Zvětšování/natahování položky - flex-grow



- Nastavuje schopnost položky zvětšovat se (natáhnout se) v případě potřeby.
- Podle nastavení flex-grow flexbox rozděluje zbylý volný prostor ve flexboxu mezi jeho položky.
- Nastavuje se jako číslo bez jednotky a znamená to poměr, jakým se bude položky zvětšovat vůči ostatním položkám.

Zvětšování/natahování položky - flex-grow



CSS

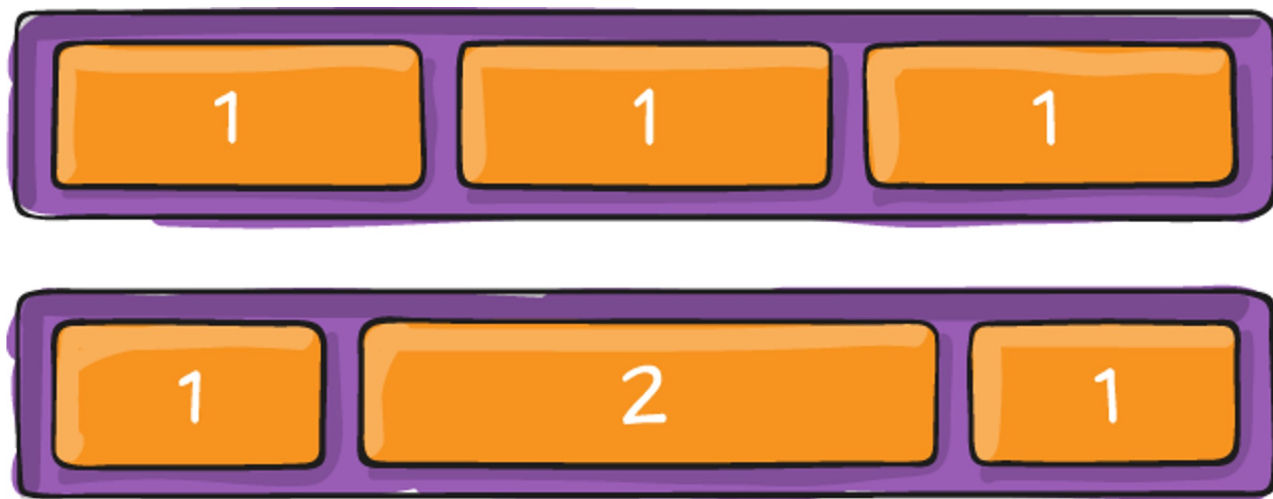
```
.polozka {  
    flex-grow: 1;  
}
```

Výchozí hodnota: 0 (= položka se nemůže natahovat).

Jakákoliv hodnota větší než nula znamená, že položka se může natahovat.

Číslo znamená, jak moc se položka zvětšuje v poměru k ostatním.

Zvětšování/natahování položky - flex-grow



- Má-li jedna položka flex-grow: 1 a druhá flex-grow:2, znamená to, že druhá položka se bude zvětšovat dvakrát rychleji než ta první.
- Neznačí to, že bude dvakrát větší. Ale pokud se položky zvětšují, zvětšuje se dvakrát tolik.

Zmenšování položky - flex-shrink



- Funguje podobně jako flex-grow, ale obráceně:
- Pokud se položky do flexboxu nevejdou, tak ty, co mají nastavený flex-shrink, flexbox zmenší.
- Flexbox od položek odebere patřičný poměr z přebytečného prostoru, který ve flexboxu zbývá.

Zmenšování položky - flex-shrink

```
.polozka {  
    flex-shrink: 1;  
}
```

Výchozí hodnota: 1 (= položka se může zmenšovat).

Číslo znamená, jak moc se položka zvětšuje v poměru k ostatním.

Nastavením na 0 můžeme zakázat zmenšování položky.

Výchozí velikost - flex-basis

- Nastavuje výchozí velikost položky pro účely výpočtu, předtím než je mezi položky rozděleno volné/přebytečné místo.
- Může být nastaveno jako libovolná délková jednotka: 100px, 10rem, 30%, apod.
- Hodnota **auto** znamená, že se bere v potaz nastavená šířka a výška prvku (width, height)

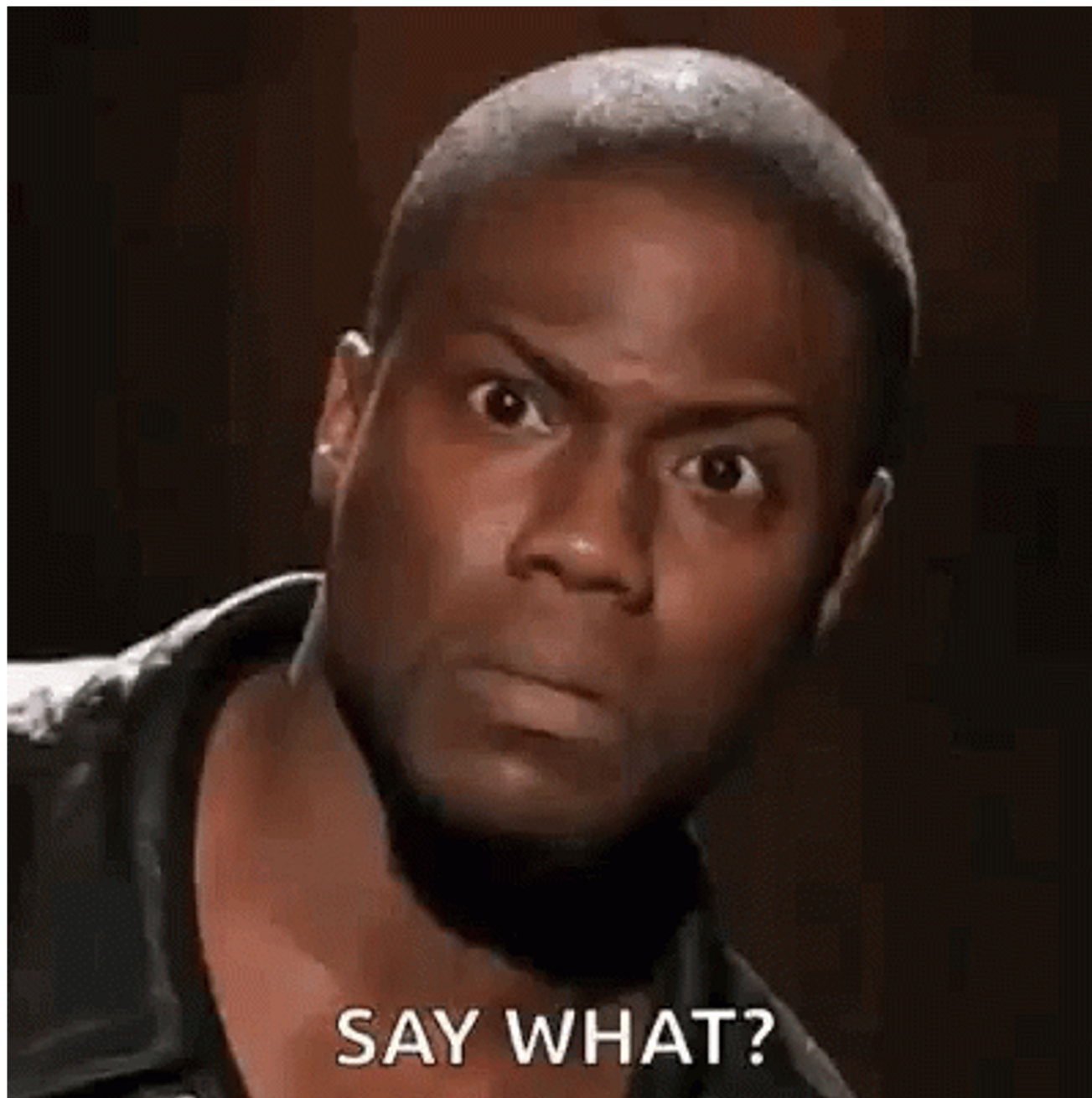
Výchozí velikost - flex-basis

CSS

```
.polozka {  
    flex-basis: 25%;  
}
```

Výchozí hodnota: auto (= položka je široká, jako obsah uvnitř).

Nastavením na konkrétní hodnotu (např. v px nebo v %) říkáme, s jakou šířkou položky má flexbox počítat, nezávisle na obsahu uvnitř.



Zkratka pro grow, shrink a basis

CSS

```
.polozka {  
  flex: 0 1 auto;  
}
```

flex-grow

flex-basis

flex-shrink

Pořadí položek ve Flexboxu

Pořadí položek ve flexboxu



- Standardně jsou položky uvnitř flexboxu v pořadí, v jakém jsou uvedeny v HTML.
- Pořadí položek ve flexboxu ale můžeme měnit nezávisle na pořadí v HTML pomocí vlastnosti **order**.

Pořadí položek ve flexboxu

CSS

```
.flexitem {  
  order: 2;  
}
```

Pořadí položek ve flexboxu



- Pokud výslovně neuvedeme, tak výchozí hodnota vlastnosti order je 0 (nula).
- Pozor na to, když měníme pořadí, je potřeba nastavit order všem položkám.
- Když jen jedné nastavíme order: 2; a ostatním nic, tak to znamená, že položka bude na konci, protože všechny se poskládají podle hodnoty order v pořadí např. 0, 0, 0, 2

Směr Flexboxu

Směr flexboxu

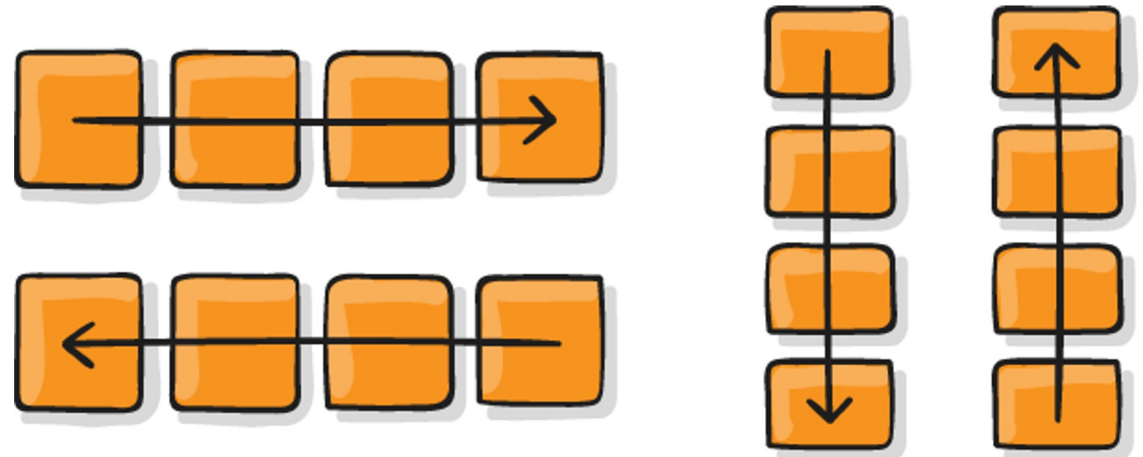
CSS

```
.kontejner {  
    flex-direction: column;  
}
```

Určí směr hlavní osy flexboxu, podél které se dávají položky vedle sebe.

Možné hodnoty:

row (výchozí), row-reverse, column, column-reverse



Zarovnávání položek ve Flexboxu

Zarovnání položek ve směru hlavní osy

```
.kontejner {  
    justify-content: center;  
}
```

Zarovnání položek flexboxu ve směru hlavní osy (tj. při flex-direction: row; zarovnává v horizontálním směru).

Možné hodnoty:

flex-start (výchozí), flex-end, center, space-between, space-around, space-evenly

flex-start



flex-end



center



space-between



space-around



space-evenly



Zarovnání položek ve směru vedlejší osy



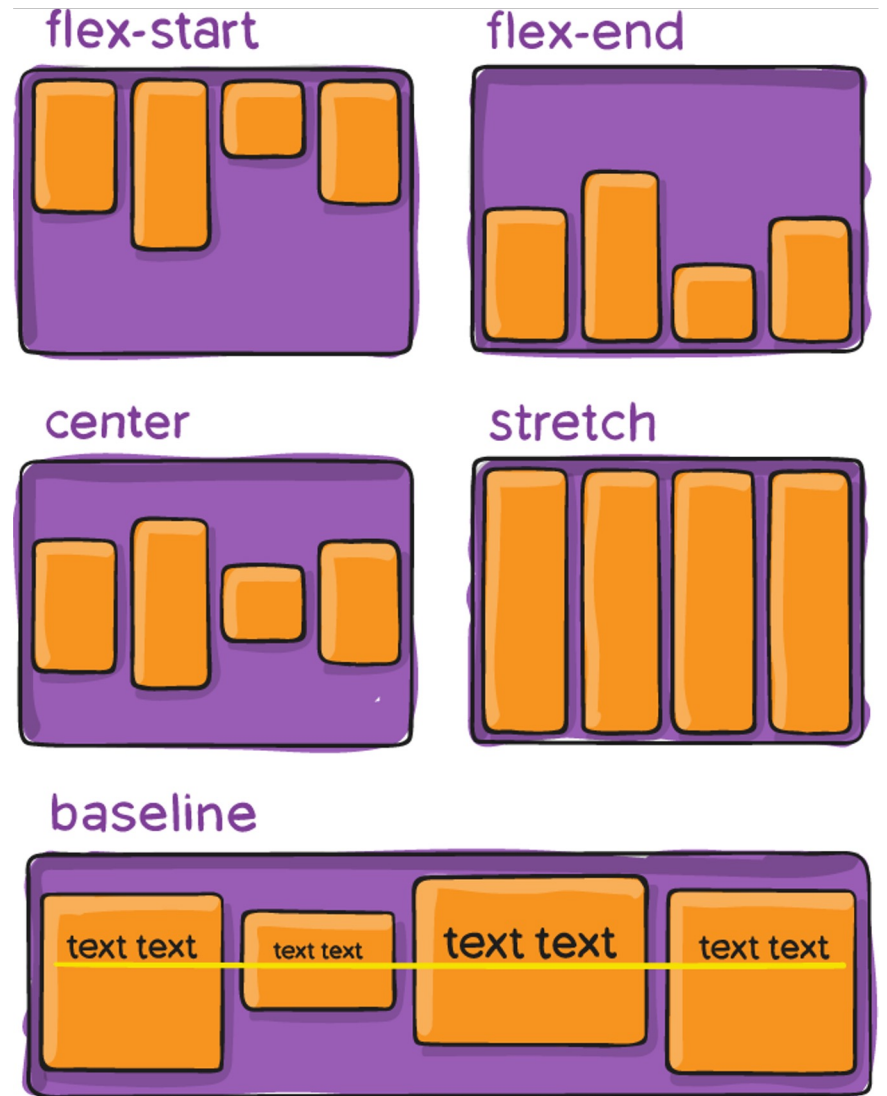
CSS

```
.kontejner {  
  align-items: center;  
}
```

Zarovnání položek flexboxu ve směru vedlejší osy (tj. při flex-direction: row; zarovnává ve vertikálním směru).

Možné hodnoty:

`flex-start`, `flex-end`, `center`, `stretch` (výchozí), `baseline`

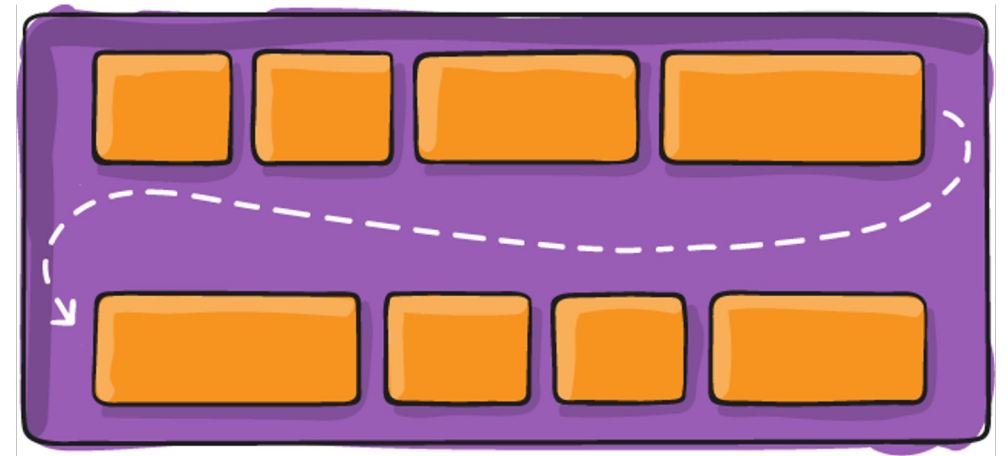


Zalamování řádku ve Flexboxu (wrap)

Zalamování řádku ve flexboxu

CSS

```
.kontejner {  
  flex-wrap: wrap;  
}
```



Když se položky do flexboxu nevejdou, tak nebudou z flexboxu vyčnívat ven, ale zalomí se na další řádek.

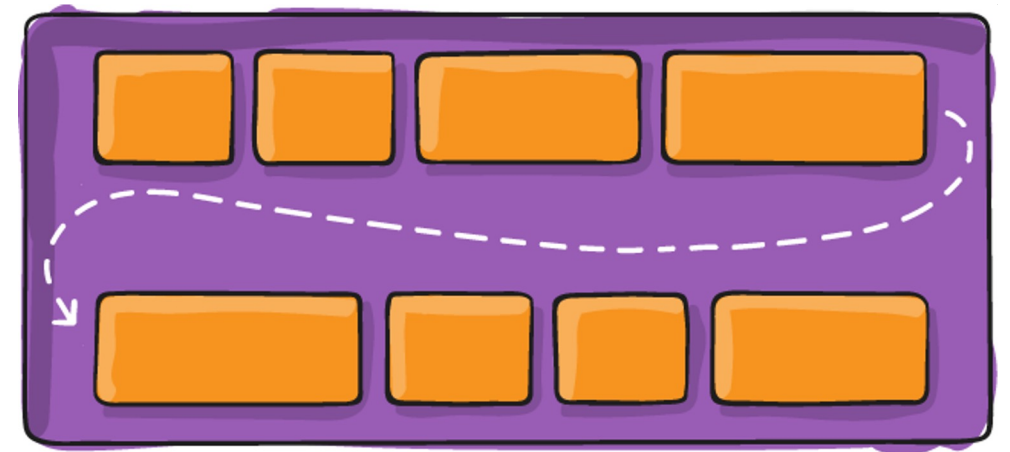
Možné hodnoty:

`nowrap` (výchozí), `wrap`, `wrap-reverse`

Zalamování řádku ve flexboxu

Kde se řádek zalomí se vyhodnotí podle výchozích šířek položek (jejich **flex-basis**).

Na každém řádku samostatně pak probíhá výpočet, zda se mají položky roztáhnout, smrsknout, jak se mají zarovnat, apod.



Občas se hodí

(ale používá se relativně zřídka)

Zarovnání jedné položky

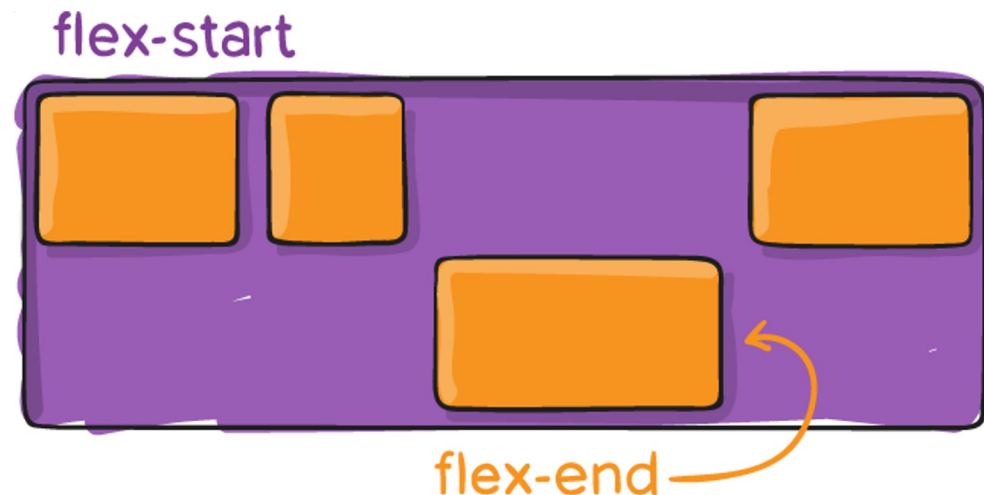
CSS

```
.flexitem {  
  align-self: flex-end;  
}
```

I když jsou položky nějak zarovnané pomocí align-items ve směru vedlejší osy, můžeme konkrétní položku zarovnat jinak.

Možné hodnoty:

`auto` (výchozí), `flex-start`, `flex-end`, `center`, `baseline`, `stretch`



Zarovnání položek ve směru vedlejší osy

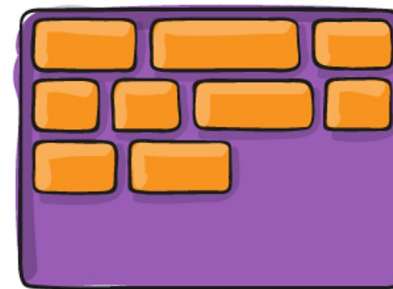
```
.kontejner {  
    align-content: center;  
}
```

Když má flexbox nastavenou větší výšku, než je výška jeho obsahu, můžeme uvnitř flexboxu celý obsah vertikálně zarovnat. (ne vertikálně, ale ve směru vedlejší osy)

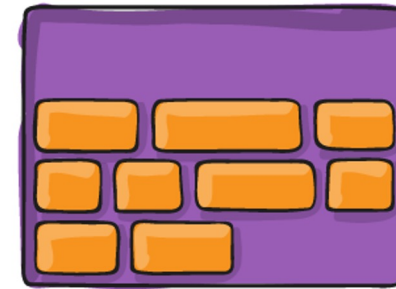
Možné hodnoty:

`flex-start` (výchozí), `flex-end`, `center`, `stretch`, `space-between`, `space-around`

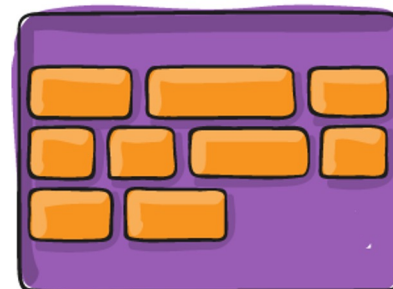
flex-start



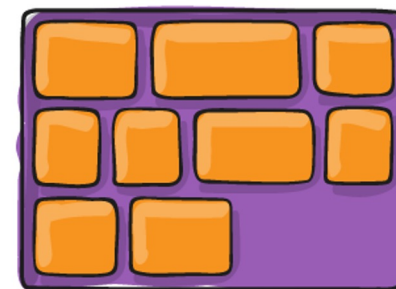
flex-end



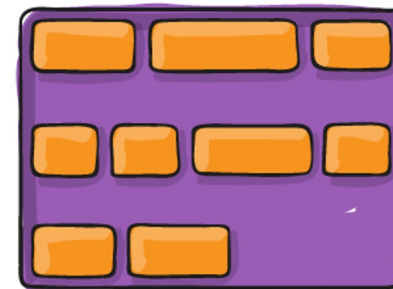
center



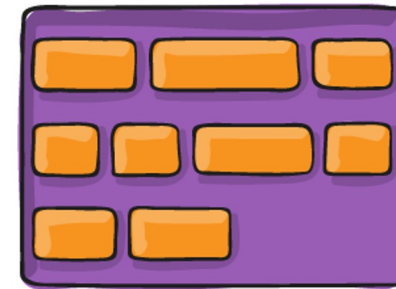
stretch



space-between



space-around



Zdroje dalších informací

Video kurz



Skvělý video kurz od Wes Bos:

<https://flexbox.io/>

Je nutné zadat email, na který přijdou odkazy na videa.
Neboj, nebude ti chodit žádný spam.

Referenční příručka



Kompletní referenční příručka/návod k flexboxu:

<https://css-tricks.com/snippets/css/a-guide-to-flexbox/>

Zde najdeš popis všech vlastností Flexboxu včetně ilustrací,
co přesně dělají.

Hra

Interaktivní hra na procvičení Flexboxu:

<https://flexboxfroggy.com/>

Zadávej vlastnosti flexboxu tak, abys dostala žabku na leknín.

