

školení frontend

# Organizace CSS

CSS

# Metodiky zápisu CSS

- **OOCSS**

- <http://EditorConfig.org>
- jak psát komponenty v CSS

- **BEM**

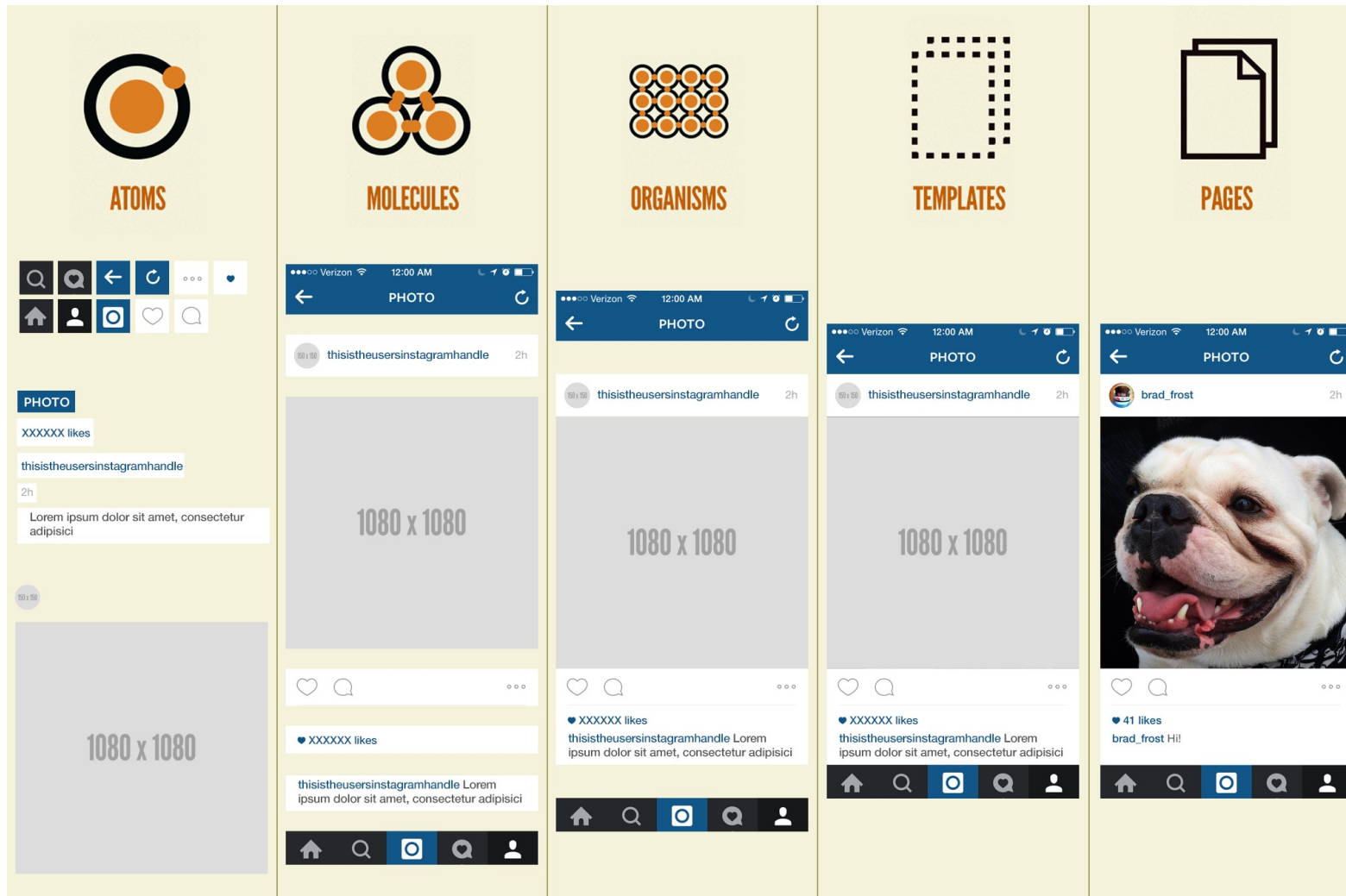
- <http://getbem.com/>
- jak pojmenovávat komponenty v CSS

- **SMACSS, ITCSS**

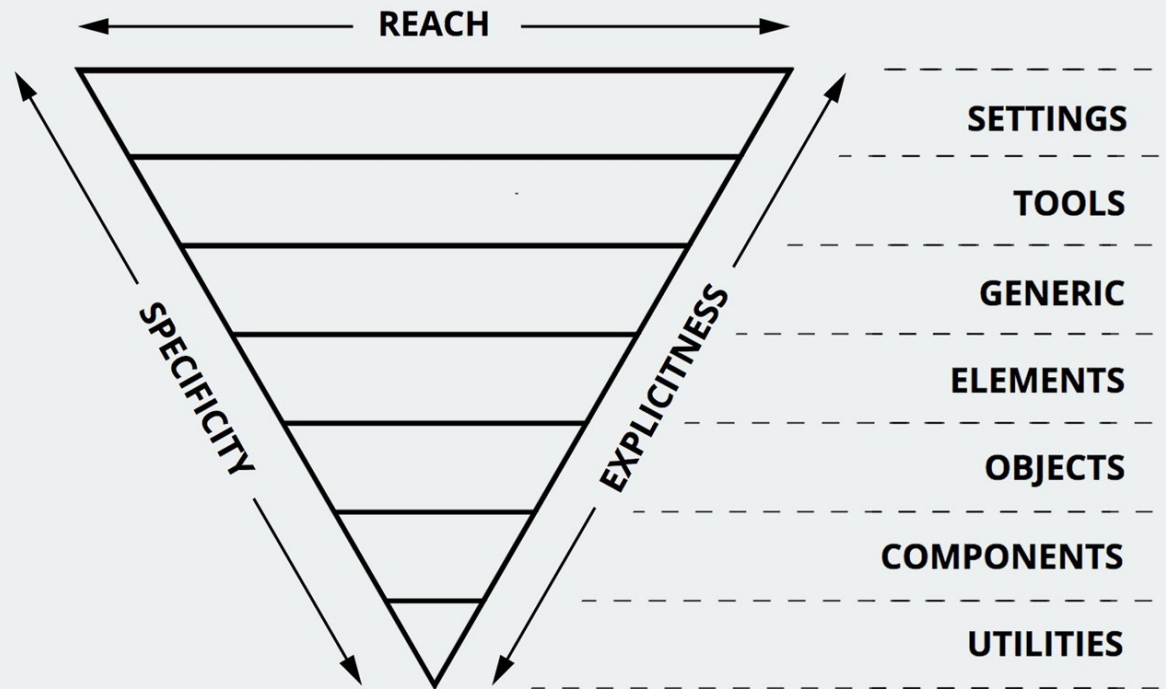
- <https://smacss.com/>
- metodiky organizace celého CSS v rámci projektu

# CSS Metodiky

## Atomic webdesign



## Define an Architecture (ITCSS)



# CSS Metodiky

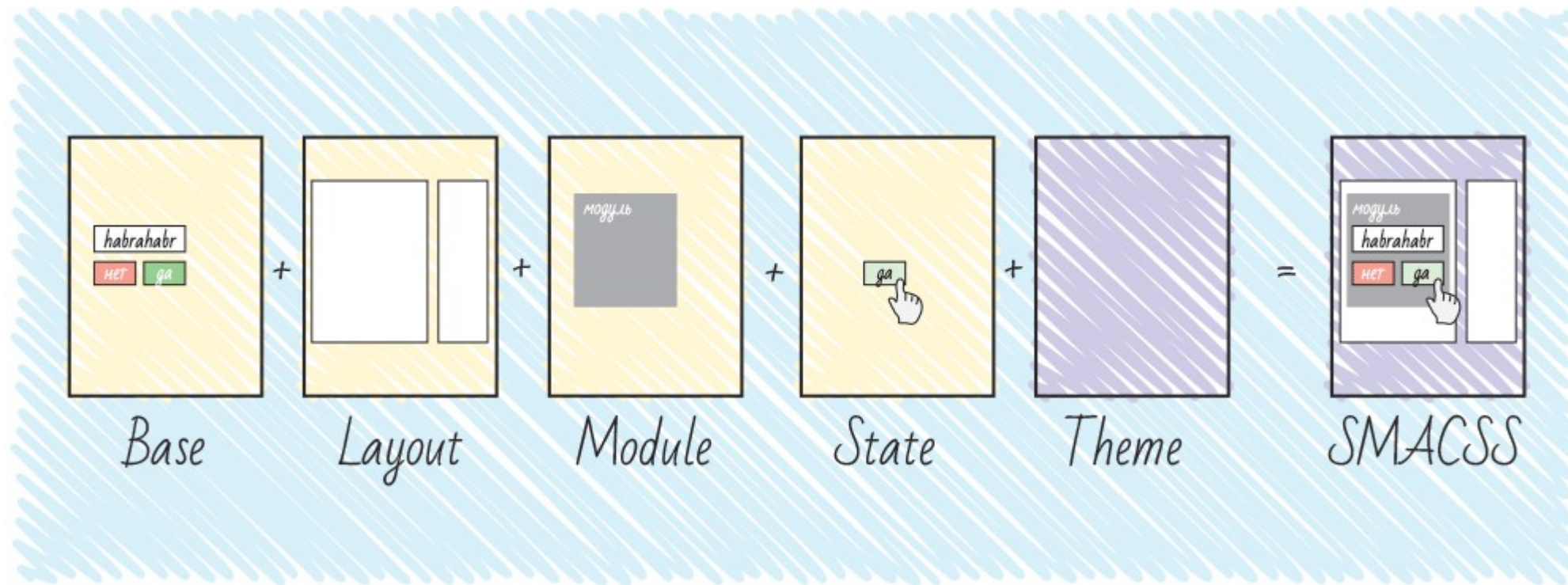
## ITCSS

ITCSS	co tam patří	příklady css tříd
Settings	<i>proměnné</i>	\$bg-color, \$font-base, ...
Tools	<i>funkce, mixins</i>	@function calc-spacing ...
Generic	<i>reset (Reboot, Normalize)</i>	body, input, ...
Elements	<i>základní nastavení layoutu</i>	body, input, h1, p, ...
Objects	<i>základní komponenty</i>	.media, .button,
Components	<i>ostatní komponenty</i>	.article-item, .list, .footer-nav, ...
Utilities	<i>pomocné třídy</i>	.img-fluid, .text-center, ...

# CSS Metodiky

## SMACSS

### Scalable and Modular Architecture for CSS



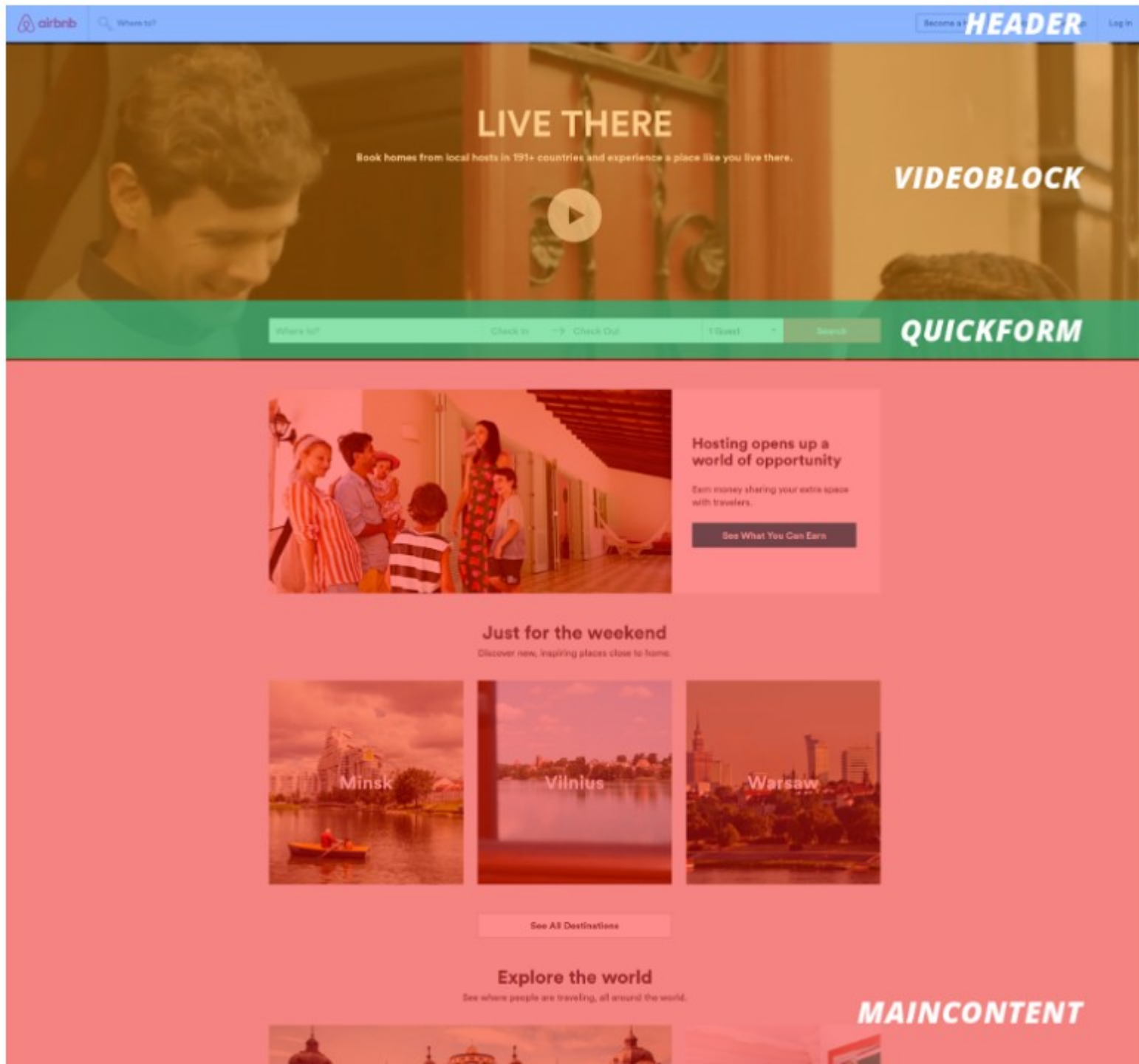
# CSS Metodiky

## SMACSS

SMACSS	co tam patří	příklady css tříd
Base	<i>normalizace, základní styly</i>	body, input, h1, p, ...
Layout	<i>grid, responsivní fw, wrappery</i>	.row, .container, .header, ...
Modules	<i>komponenty</i>	.article-list, .header-nav, ...
State	<i>stavy</i>	.active, .focus, ...

# CSS Metodiky

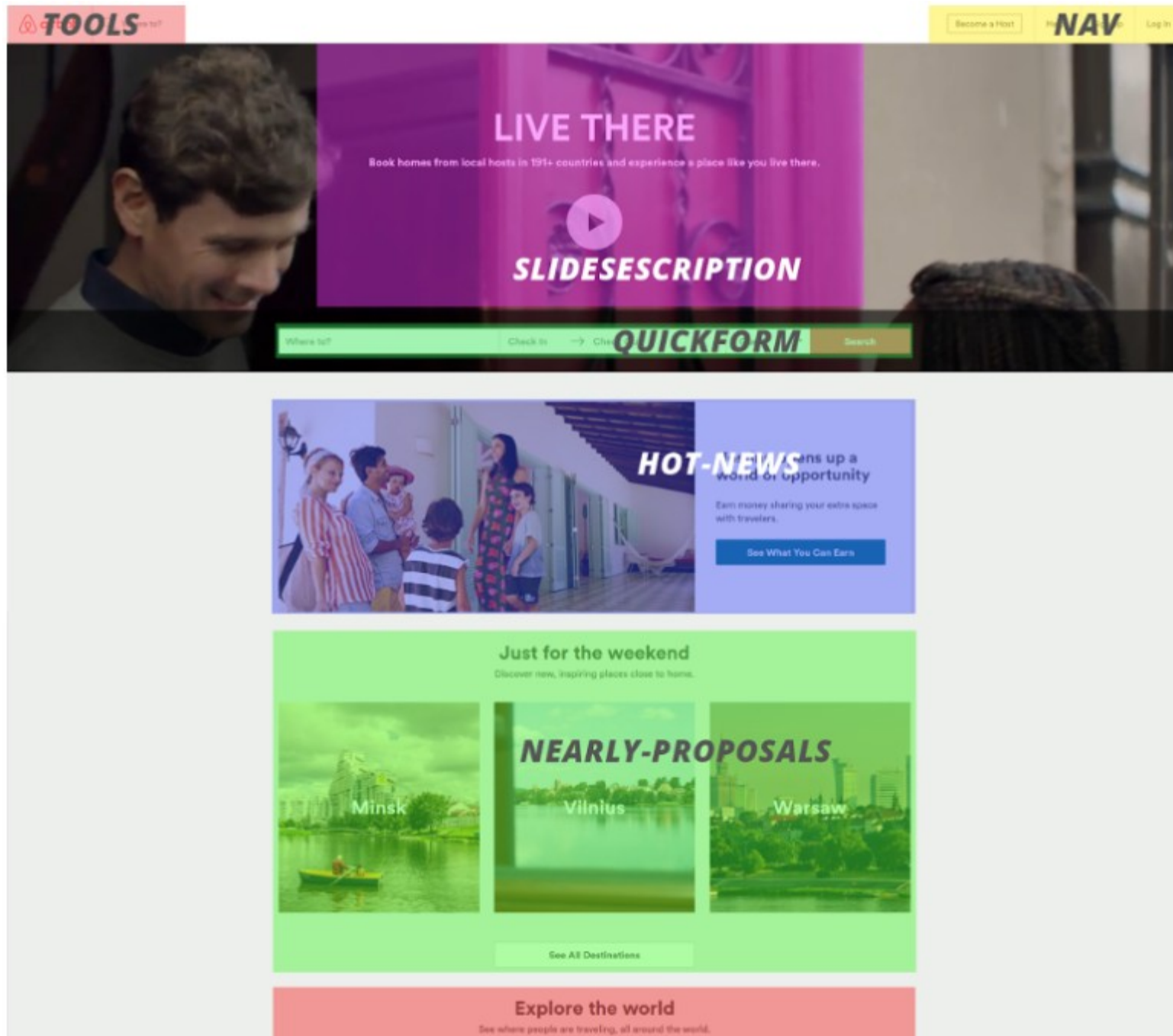
## SMACSS





# CSS Metodiky

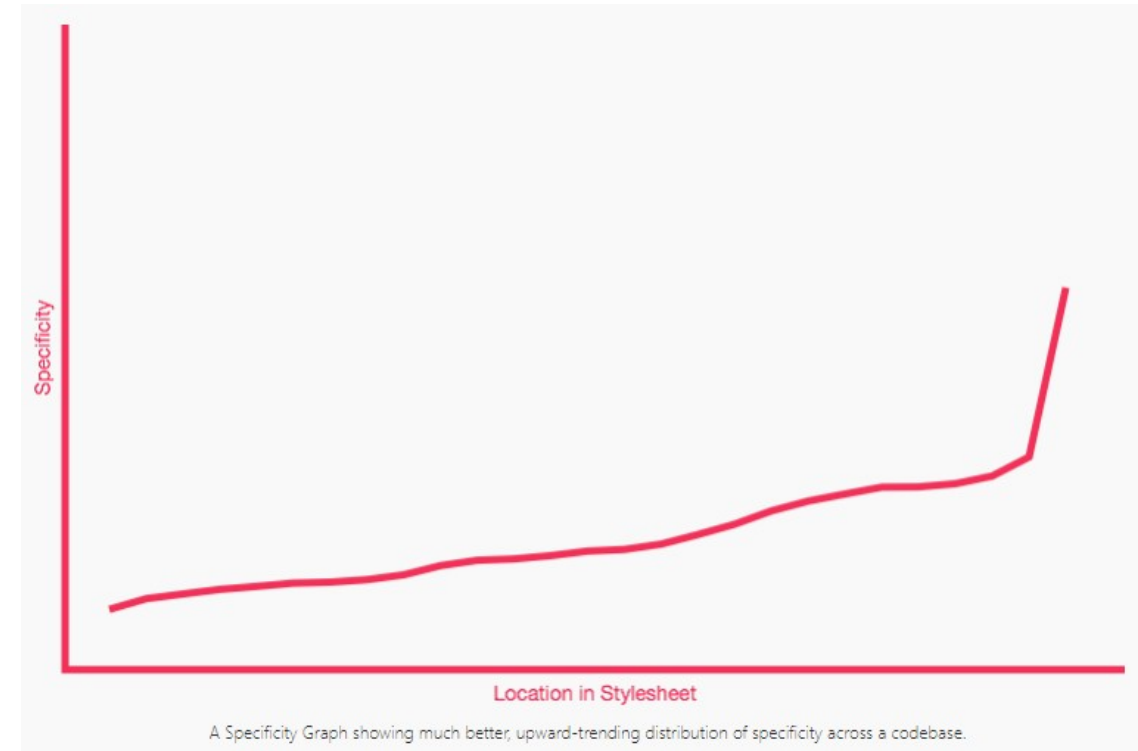
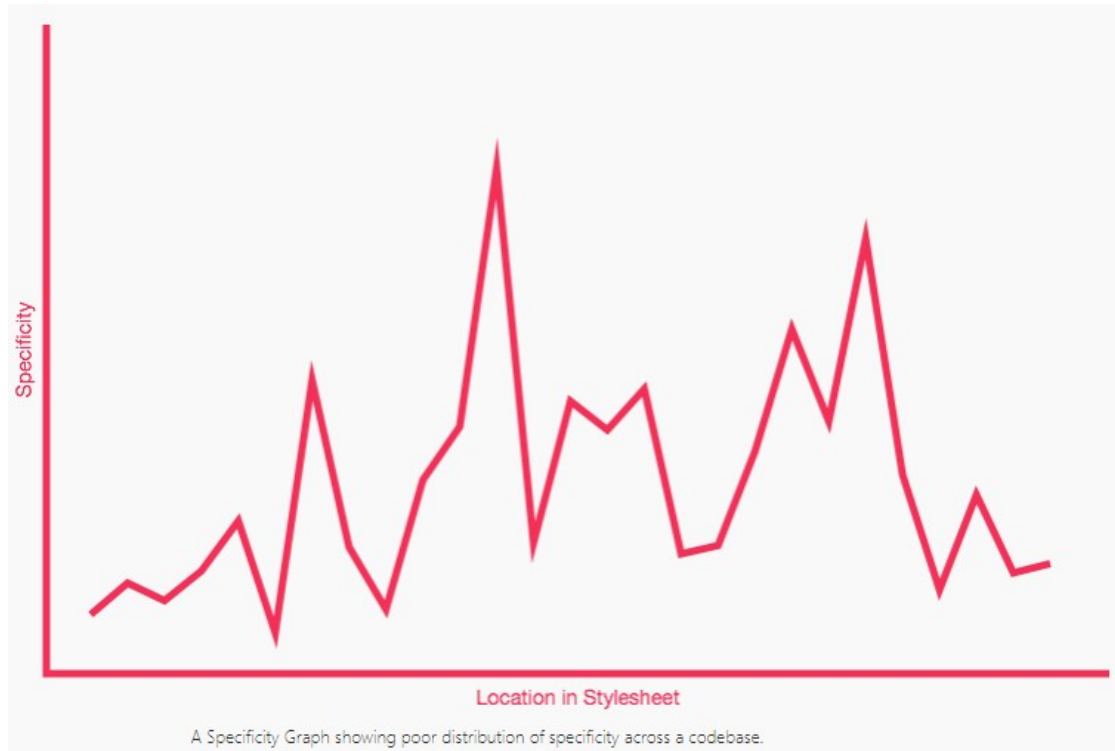
## SMACSS



# organizace CSS

## Rozložení specificity

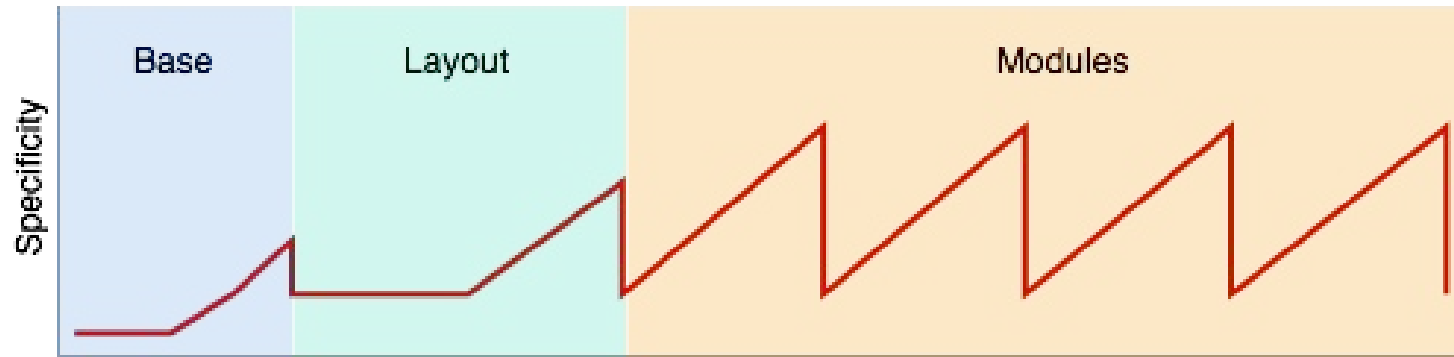
<https://cssstats.com/>



<https://csswizardry.com/2014/10/the-specificity-graph/>

# organizace CSS

## Rozložení specificity



H1, p, a, ... | .header, .grid, ... | .media-article, .pagination, ...

# CSS Metodiky

## Object oriented CSS

### **Cíle OOCSS**

1. Zrychlení psaní kódu i načtení CSS souboru
2. Škálovatelnost
3. Efektivita
4. Snazší údržba
5. Lepší čitelnost
6. Příbuznost s ostatními koncepty zápisu CSS

### **Nevýhody**

7. Roste počet tříd u elementu
8. Může být zbytečné u menších projektů
9. Chvilku trvá, než se dostane do krve


### 1. Oddělení CSS od struktury

- nikdy nepoužíváme HTML tagy jako selektory (kromě základní typografie)

### 2. Nezávislost na kontextu

`.footer .nav { ... }`  `.nav-footer { ... }`

### 3. Snaha o minimální specifičnost

- nepoužíváme id selektory, !important nebo kombinace selektorů
- `.nav.nav-footer { ... }`  `.nav-footer { ... }`

# CSS Metodiky

## OOCSS

```
<a href="#" class="button-blue-small">Click me!</a>
<style>
  .button-blue-small {
    display: inline-block;
    text-align: center;
    margin: 10px 5px;
    text-decoration: none;
    color: #FFF;
    border: 1px solid #0082BE;
    background: #00A4EF;
    font-size: 13px;
    padding: 5px 20px;
  }
</style>
```

# CSS Metodiky

## **OOCSS**

1. **Struktura**
2. **Velikost**
3. **Styly**

# OOCSS – základní struktura

```
.button {  
  display: inline-block;  
  text-align: center;  
  margin: 10px 5px;  
  text-decoration: none;  
}
```



CSS Metodiky

## OOCSS – velikost

```
.button-small {  
    font-size: 13px;  
    padding: 6px 20px;  
}
```

# CSS Metodiky

## OOCSS – styly

```
.button-blue {  
    color: #FFF;  
    border: 1px solid #0082BE;  
    background: #00A4EF;  
}
```

# CSS Metodiky

## OOCSS

```
<a href="#" class="button button-small button-blue">Click me!</a>
```

```
<style>  
  .button { ... }  
  .button-small { ... }  
  .button-blue { ... }  
</style>
```

## OOCSS – problémy

```
<div class="article article-list">  
  <div class="article-header">  
    ...  
  </div>  
</div>
```

CSS Metodiky

# **komponenta versus kontext**

<https://codepen.io/machal/pen/JmdRaa>

# Block – Element – Modifier

### Výhody

1. "Nadstavba" OOCSS (platí všechny podmínky OOCSS)
2. Jedná se především o metodiku jak pojmenovávat komponenty
3. Jednoduše lze poznat význam třídy
4. Snadno pochopitelný

### Nevýhody

5. Hledání vhodného jména
6. Nezvyklý a často zdlouhavý zápis tříd

# Block – Element – Modifier

<http://getbem.com/>

## **Blok** `.nav-main {}`

- Nezávislá část stránky, znovupoužitelná
- Bloky lze do sebe zanořit

## **Element** `.nav-main__list {}`

- Prvek uvnitř bloku, nefunguje bez bloku

## **Modifikátor** `.nav-main--dark {}`

- Varianta bloku nebo elementu
- Mění vzhled, chování nebo stav

## BEM – modifikátory

```
<!-- Takhle ne -->
```

```
<figure class="photo">
```

```
  
```

```
  <figcaption>Look at me!</figcaption>
```

```
</figure>
```

```
<style>
```

```
  .photo { }
```

```
  .photo img { }
```

```
  .photo figcaption { }
```

```
</style>
```



## CSS Metodiky

# BEM – modifikátory

```
<!-- Takhle ano -->
```

```
<figure class="photo">
```

```
  
```

```
  <figcaption class="photo__caption">Look at me!</figcaption>
```

```
</figure>
```

```
<style>
```

```
  .photo { }
```

```
  .photo__img { }
```

```
  .photo__caption { }
```

```
</style>
```

## BEM - modifikátory

```
<a href="#" class="button button--small button--blue">  
  Click me!  
</a>
```

```
<style>  
  .button { ... }  
  .button--small { ... }  
  .button--blue { ... }  
</style>
```

# CSS Metodiky

## BEM - modifikatory

```
<!-- Takhle ne -->
```

```
<figure class="photo">  
    
  <figcaption class="photo__caption">  
    <blockquote  
      class="photo__caption__quote">  
        Look at me!  
      </blockquote>  
    </figcaption>  
  </figure>
```

```
<style>  
.photo { }  
.photo__img { }  
.photo__caption { }  
.photo__caption__quote { }  
</style>  
</body>
```

# CSS Metodiky

## BEM - modifikátory

```
<!-- Takhle ano -->
```

```
<figure class="photo">  
    
  <figcaption class="photo__caption">  
    <blockquote  
      class="photo__quote">  
        Look at me!  
      </blockquote>  
    </figcaption>  
  </figure>
```

```
<style>  
.photo { }  
.photo__img { }  
.photo__caption { }  
.photo__quote { }  
</style>  
</body>
```

## CSS Metodiky

# BEM - modifikátory

```
<!-- Takhle ne -->
```

```
<figure class="photo">
```

```
  
```

```
  <figcaption class="photo__caption photo__caption--highlighted">
```

```
    Look at me!
```

```
  </figcaption>
```

```
</figure>
```

```
<style>
```

```
  .photo__img--highlighted { }
```

```
  .photo__caption--highlighted { }
```

```
</style>
```

## CSS Metodiky

# BEM - modifikátory

```
<!-- Takhle ano -->
```

```
<figure class="photo photo--highlighted">
```

```
  
```

```
  <figcaption class="photo__caption">
```

```
    Look at me!
```

```
  </figcaption>
```

```
</figure>
```

```
<style>
```

```
  .photo--highlighted .photo__img { }
```

```
  .photo--highlighted .photo__caption { }
```

```
</style>
```

# CSS Metodiky

## **BEM - modifikátory**

```
<!-- Takhle ne -->
```

```
<div class="somethesis somethesis--fastread">  
  <div class="somethesis__someelement"></div>  
</div>
```

```
<style>  
  .somethesis { }  
  .somethesis--fastread { }  
  .somethesis__someelement { }  
</style>
```

# CSS Metodiky

## BEM - modifikátory

```
<!-- Takhle ano -->
```

```
<div class="some-thesis some-thesis--fast-read">  
  <div class="some-thesis__some-element"></div>  
</div>
```

```
<style>  
  .some-thesis { }  
  .some-thesis--fast-read { }  
  .some-thesis__some-element { }  
</style>
```



organizace CSS

## Refaktoring – BEM

- Upravte v naší refaktorované komponentě zápis, aby odpovídal metodice BEM.
- Můžete využít codepen: <https://codepen.io/svobodalukas/pen/XxZmwY>

# organizace CSS

## Základní pravidla

### Co nejnižší specifita

```
.heading {  
  font-size: 24px;  
}
```

```
#header .nav h3 {  
  font-size: 16px;  
}
```



```
.heading {  
  font-size: 24px;  
}
```

```
.heading--sm {  
  font-size: 16px;  
}
```

# organizace CSS

## Základní pravidla

### CSS nezávislé na HTML

```
h1.heading {  
  font-size: 24px;  
}
```

```
nav > ul > li {  
  font-size: 16px;  
}
```



```
.heading {  
  font-size: 24px;  
}
```

```
.list-item {  
  font-size: 16px;  
}
```

# organizace CSS

## Základní pravidla

### Snadno pochopitelné CSS

```
.heading {  
  @include my-font-b(24px, 30px, #d5d5d5);  
}
```



```
.heading {  
  font-size: 24px;  
  margin-bottom: 30px;  
  border-bottom: 1px solid #d5d5d5;  
}
```

# organizace CSS

## Základní pravidla

### Využívání globálních a lokálních proměnných

```
.heading {  
  font-size: 24px;  
  color: #d5d5d5;
```

```
  @media (min-width: 768px) {  
    ...  
  }  
}
```



```
.heading {  
  font-size: $text-xl;  
  color: $color-secondary;
```

```
  @media (min-width: $screen-md) {  
    ...  
  }  
}
```

# organizace CSS

## Základní pravidla

**Snažíme se nezanořovat (pokud to není nezbytně nutné)**

```
.heading .nav .nav-footer ul > li a {  
  ...  
}
```



```
.heading-link {  
  ...  
}
```

# organizace CSS

## Základní pravidla

**Zbytečně nevyužíváme zanořování pomocí &**

```
.header {  
  background: #f5f5f5;  
  &__nav {  
    font-size: 16px;  
    &--large {  
      font-size: 18px;  
    }  
  }  
}
```



```
.header {  
  background: #f5f5f5;  
}  
  
.header__nav {  
  font-size: 16px;  
}  
  
.header__nav--large {  
  font-size: 18px;  
}
```

# organizace CSS

## Základní pravidla

### Matematické výrazy píšeme pochopitelně

```
.content {  
  border-top: 76px;  
}
```



```
$header-height: 60px;  
$spacing-base: 16px;
```

```
.content {  
  border-top: calc($header-height + $spacing-base);  
}
```



# organizace CSS

## Základní pravidla

**Komentujeme kód, ideálně v angličtině**

```
.content {  
  position: relative;  
}
```



```
.content {  
  // used by .ad-banner (has absolute position)  
  position: relative;  
}
```

# organizace CSS

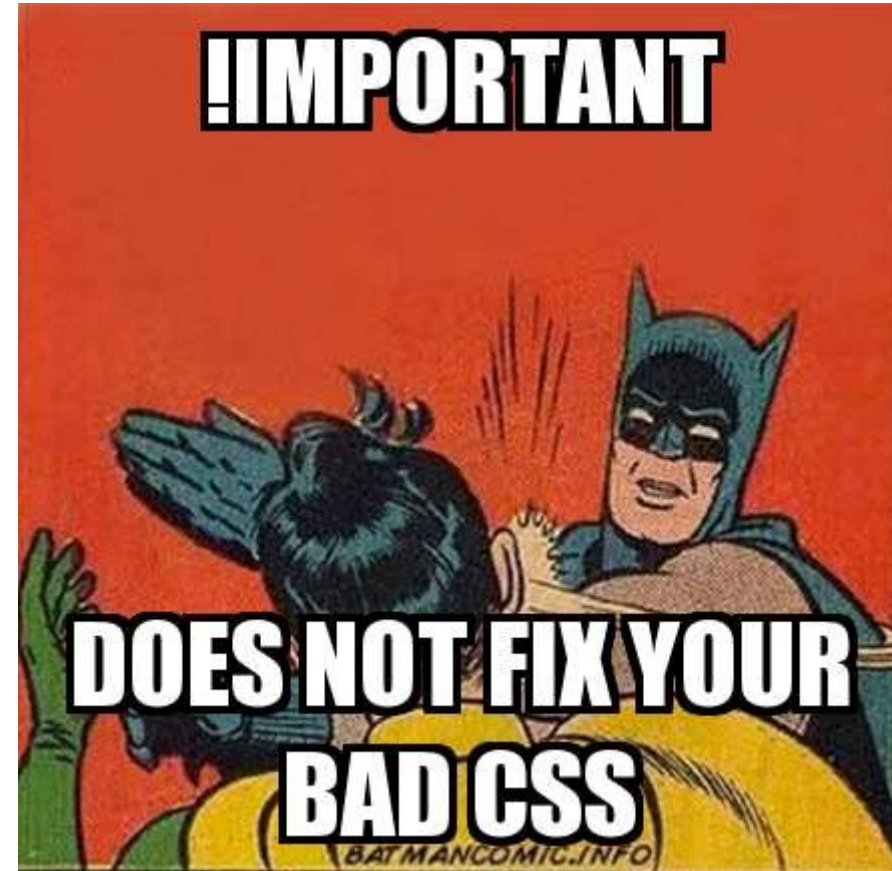
## Základní pravidla

Vyhnete se !important

```
.text-red {  
  color: red !important;  
}
```

```
.text-red.text-more-red {  
  color: #e30604 !important;  
}
```

```
#main.text-blue {  
  color: blue !important;  
}
```



# organizace CSS

## Základní pravidla

### Nepoužíváme inline styly

```
.text-red {  
  color: red;  
}
```

```
<li class="text-red" style="color: blue;" >  
  ...  
</li>
```

# organizace CSS

## Základní pravidla

**nemícháme různé jednotky**

```
h1 {  
  font-size: 24px;  
}
```

```
.heading {  
  font-size: 3em;  
}
```

```
.heading-2 {  
  font-size: 2rem;  
}
```



```
h1 {  
  font-size: 1.5rem;  
}
```

```
.heading {  
  font-size: 3rem;  
}
```

```
.heading-2 {  
  font-size: 2rem;  
}
```

# organizace CSS

## Základní pravidla

**fallbacky necháme strojům**

```
.block {  
  -ms-transform: rotate(90deg);  
  -webkit-transform: rotate(90deg);  
  transform: rotate(90deg);  
}
```



```
.block {  
  transform: rotate(90deg);  
}  
  
/* autoprefixer for build */
```

# organizace CSS

## Kdy to smrdí

### podezřelé CSS konstrukce

```
h2 {  
  font-size: 2em;  
  margin-bottom: 0.5em;  
  padding-bottom: 0.5em;  
  border-bottom: 1px solid #ccc;  
}  
.no-border {  
  padding-bottom: 0;  
  border-bottom: none;  
}
```



```
h2 {  
  font-size: 2em;  
  margin-bottom: 0.5em;  
}  
.headline {  
  padding-bottom: 0.5em;  
  border-bottom: 1px solid #ccc;  
}
```

# organizace CSS

## Kdy to smrdí

### no Single Source of Truth

```
.btn {}
```

```
.header .btn,
```

```
.header .btn:hover {}
```

```
.sidebar .btn {}
```

```
.modal .btn {}
```

```
.page aside .btn {}
```

```
nav .btn {}
```



```
.btn {}
```

```
.btn--large {}
```

```
.btn--primary {}
```

```
.btn--nav {}
```

# organizace CSS

## Kdy to smrdí

### Brute Force

```
.foo {  
  margin-top: -12px;  
  position: relative;  
  z-index: 1000;  
  height: 48px;  
  float: left;  
}
```



organizace CSS  
**Základní pravidla**

**Zásady psaní respektujícího CSS**

<https://www.vzhurudolu.cz/prirucka/rcss-zasady>