PGS
SOFTWARE

PGS SOFTWARE

Akademia Developera – edycja FrontDev

(Nieco bardziej) zaawansowany CSS

KONRAD PRZYDZIAŁ **FRONT-END DEVELOPER**

Co już poznaliśmy

- składnia css
- osadzanie styli w HTML
- selektory
- pseudo-klasy
- pseudo-elementy
- priorytety stylów
- CSS box model
- deklarowanie kolorów
- właściwość display
- właściwość position
- pozycjonowanie (float)

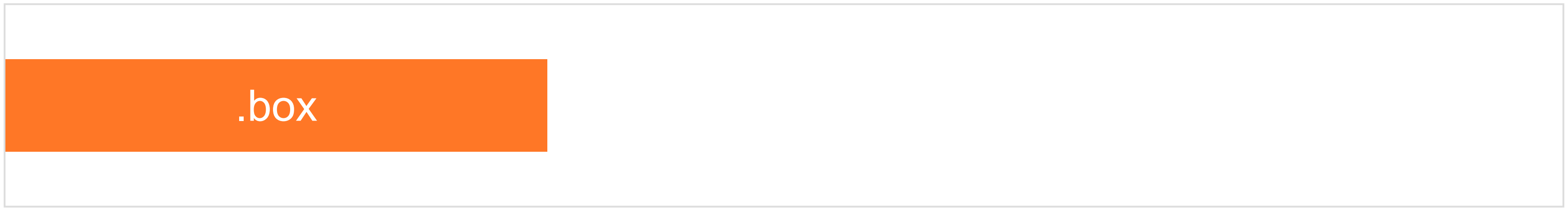
Co poznamy dziś?

- jednostki miary
- media queries, calc()
- flexbox
- prefixy
- animacje
- transformacje
- konwencje CSS

Jednostki miary

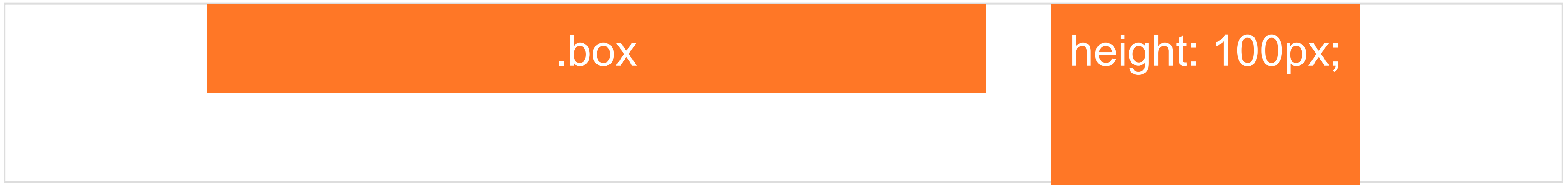
px

```
.box {
        width: 300px;
}
```



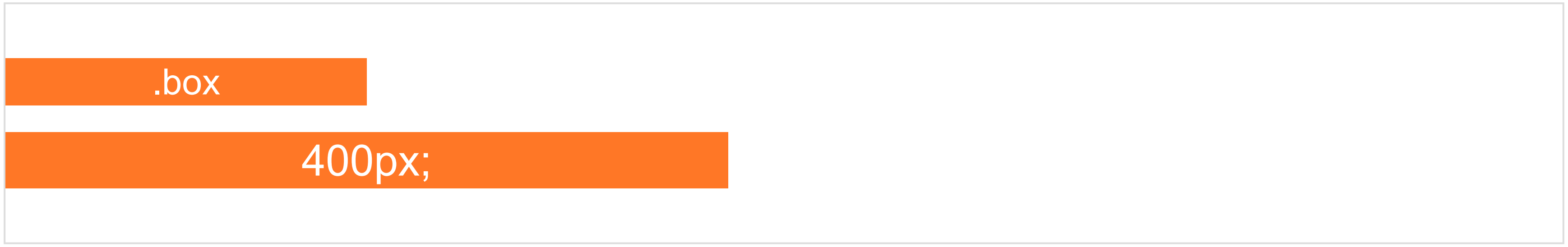
%

```css
.container {
    height: 100px;
}
.box {
    width: 50%;
    height: 50%;
    display: inline-block;
}
```



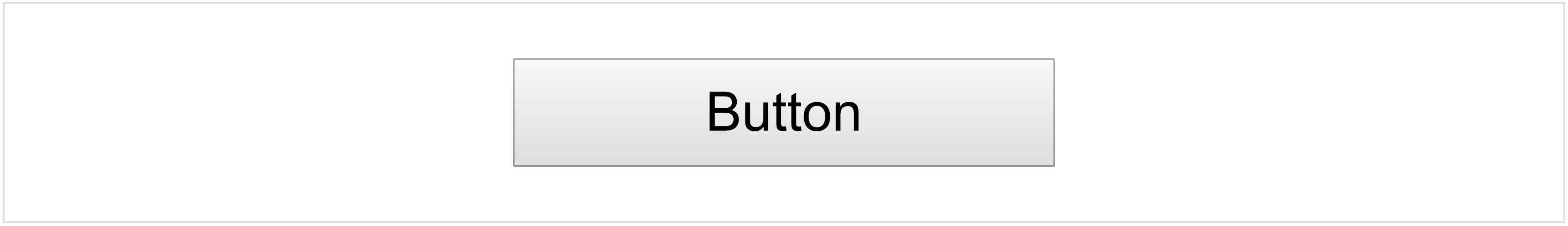
em

```
.box {
    width: 10em;
    font-size: 20px;
}
```



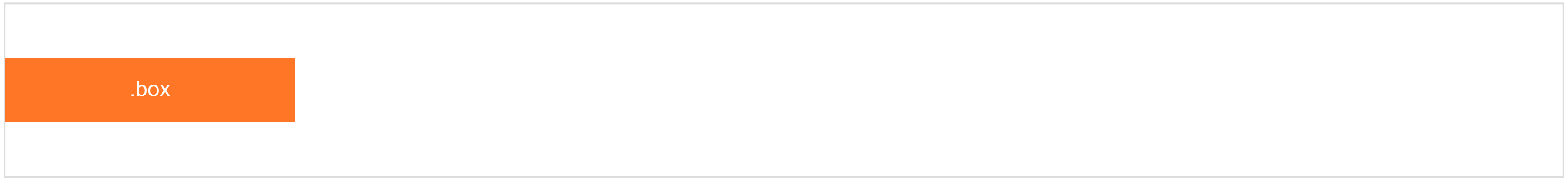
em - realny przykład

```
button {
    height: 60px;
    width: 300px;
    font-size: 30px;
}
```



rem

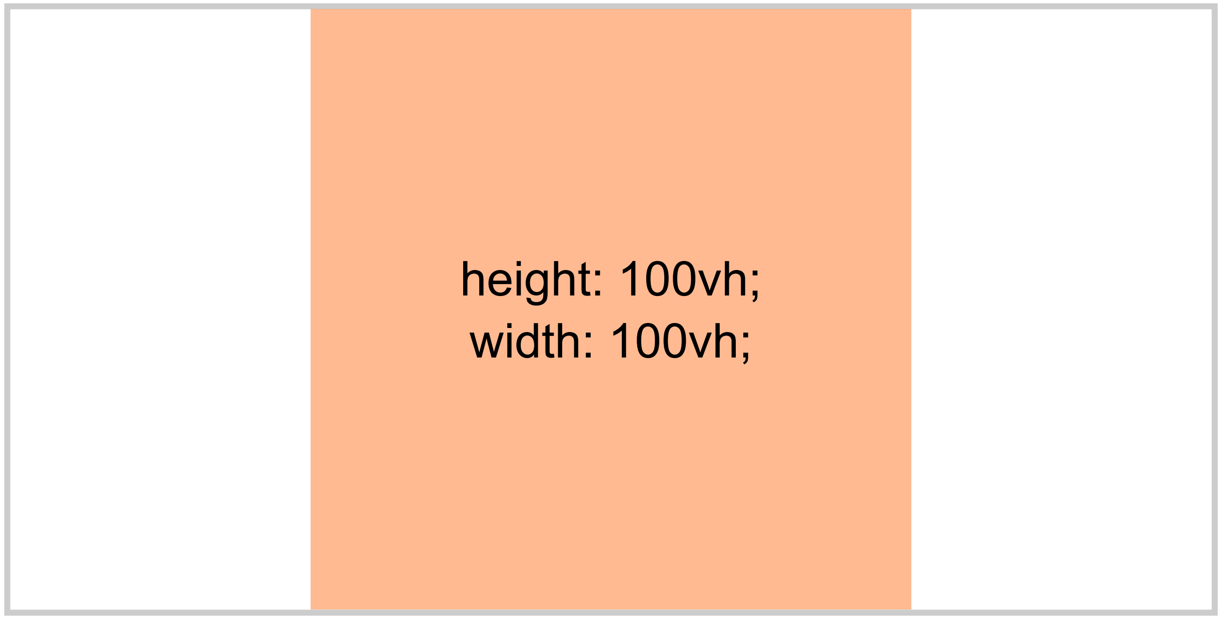
```
.box {
    width: 10rem;
    font-size: 12px;
}
```

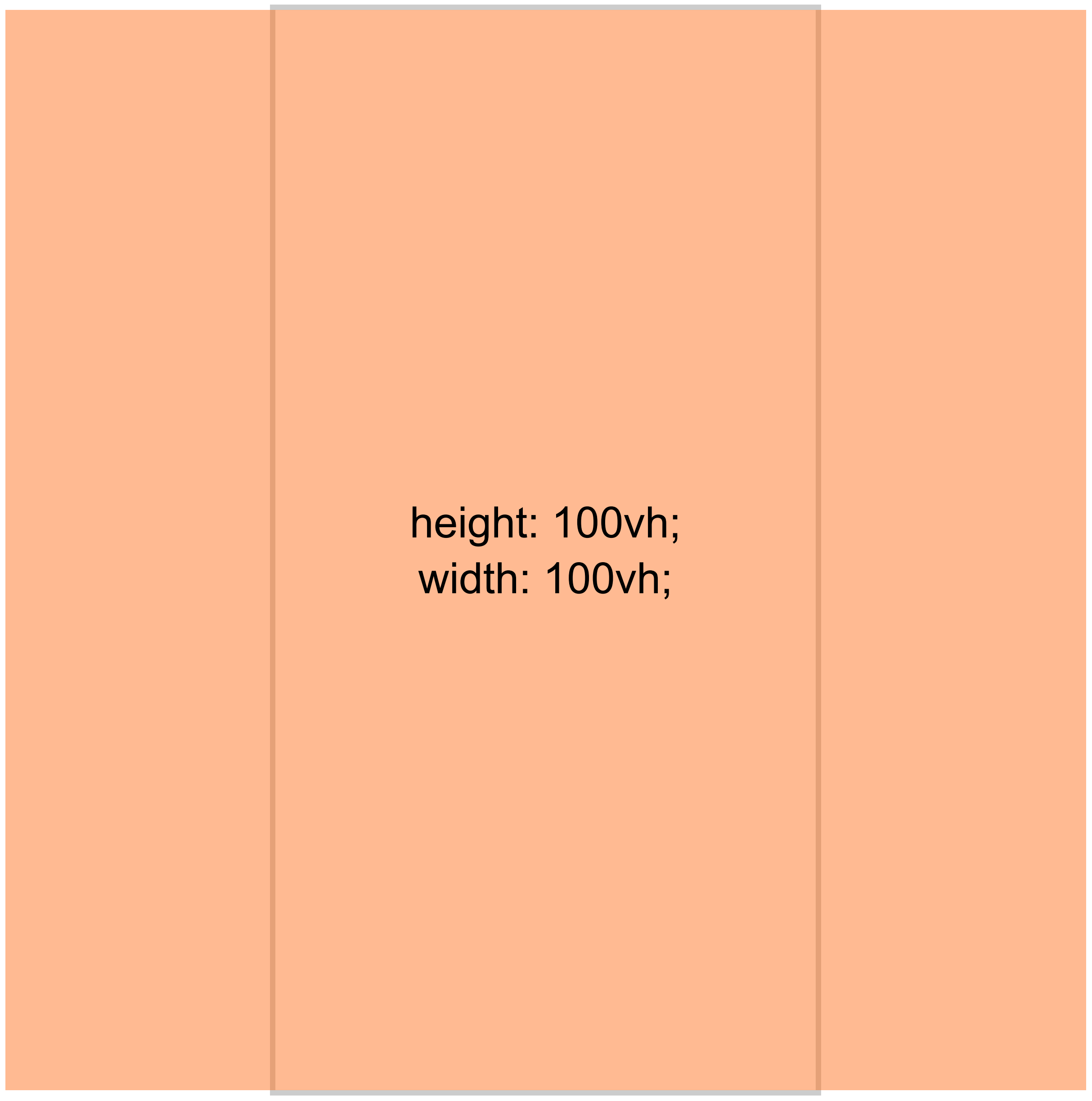


vh, vw

```
div {
    width: 50vw;
    height: 50vh;
}
```

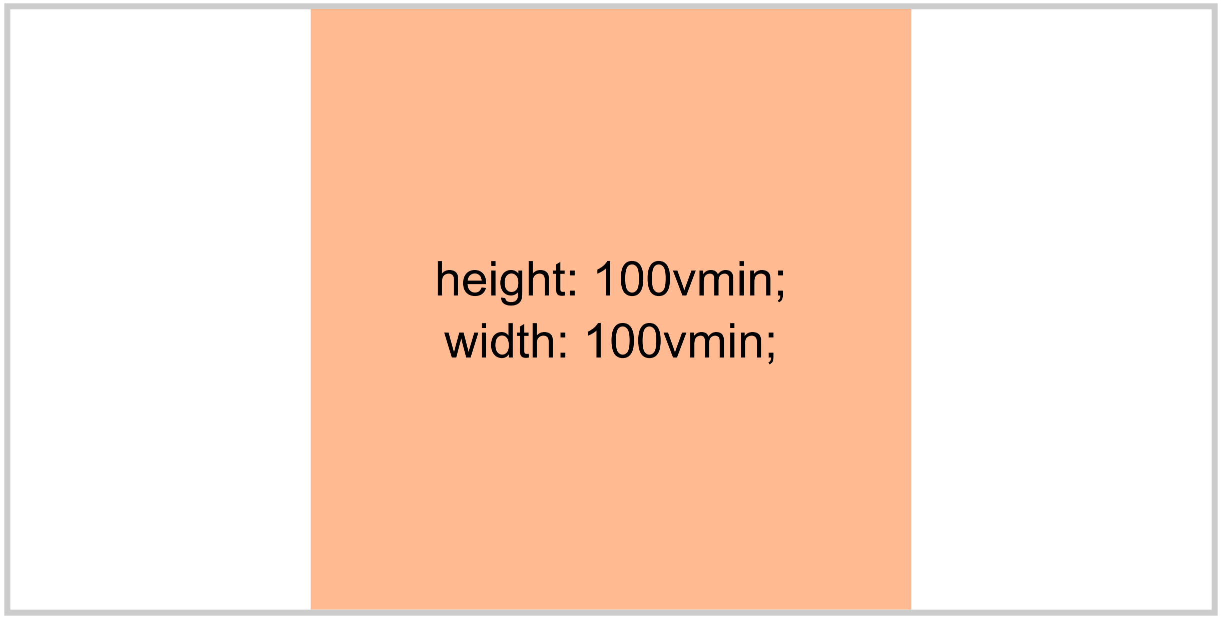
Zagadka

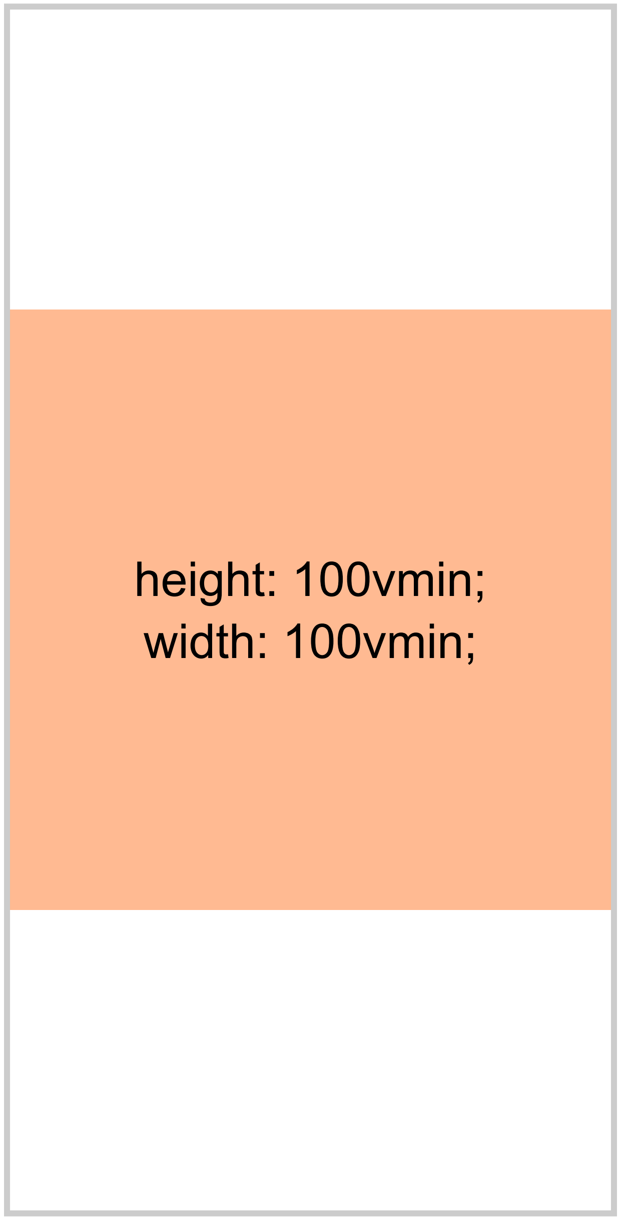

height: 100vh;
width: 100vh;


height: 100vh;
width: 100vh;

vmin, vmax

```css
div {
    width: 50vmin;
    height: 50vmax;
}
```


height: 100vmin;
width: 100vmin;


height: 100vmin;
width: 100vmin;

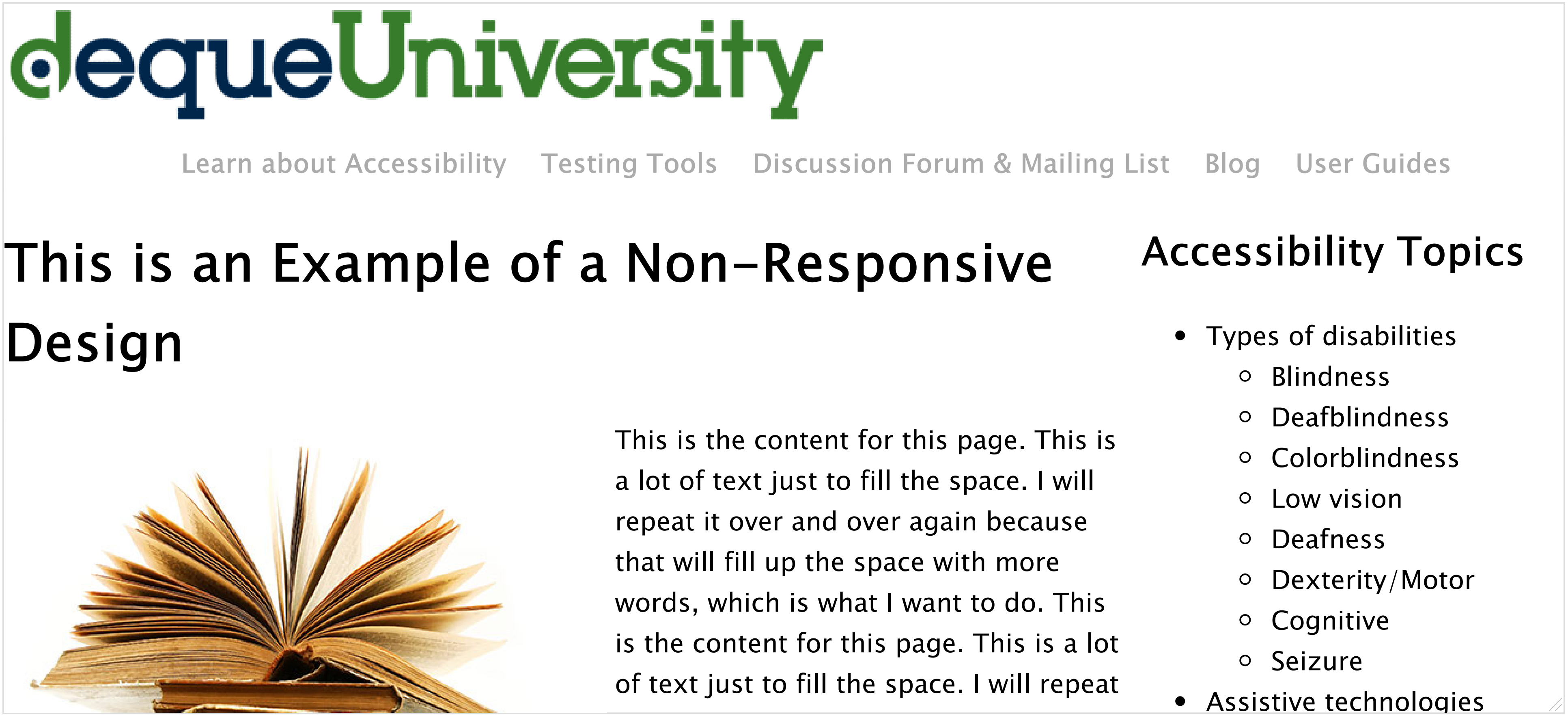
pozostałe jednostki

- cm
- in
- mm
- pc
- pt
- ex

Przydatne linki:

https://css-tricks.com/the-lengths-of-css

Responsive Web Design



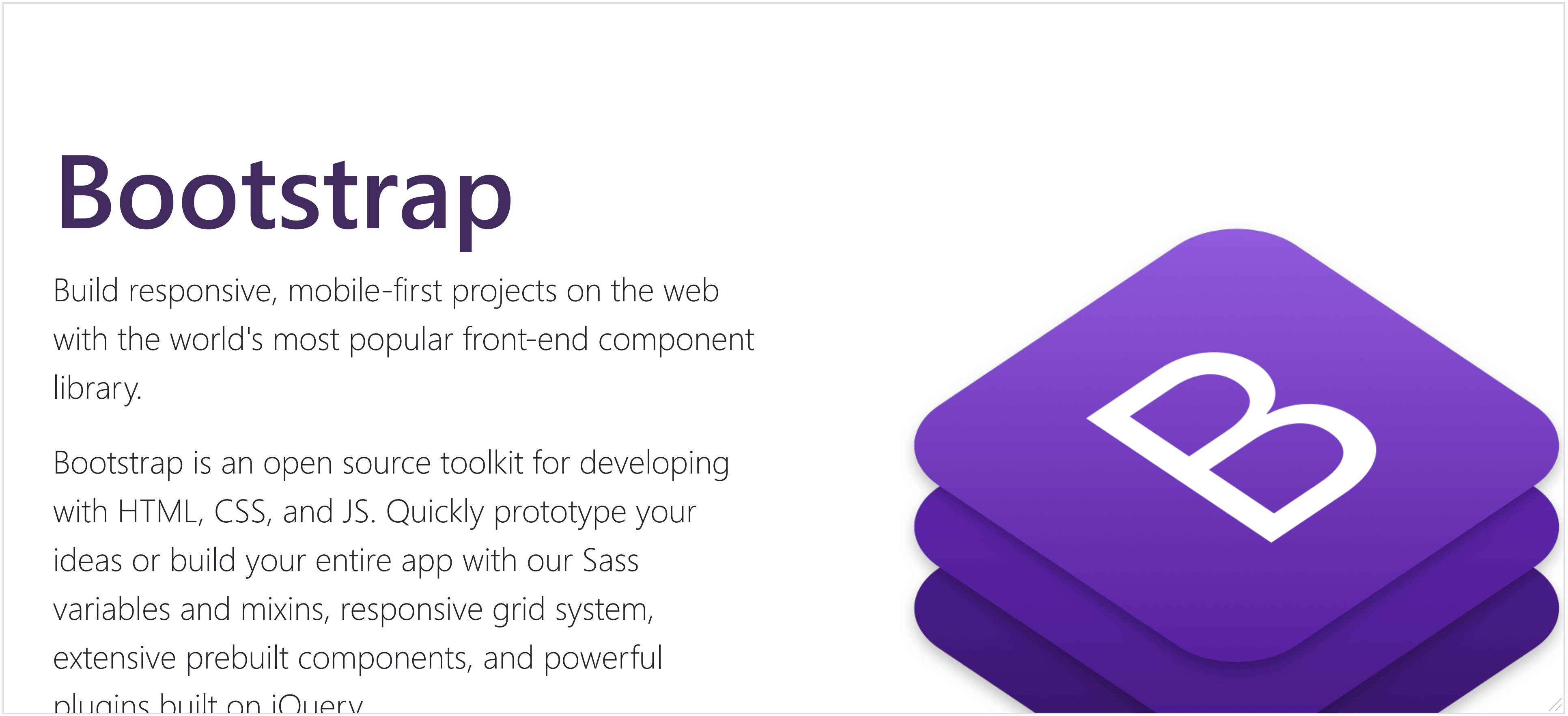
Learn about Accessibility   Testing Tools   Discussion Forum & Mailing List   Blog   User Guides

This is an Example of a Non-Responsive Design

This is the content for this page. This is a lot of text just to fill the space. I will repeat it over and over again because that will fill up the space with more words, which is what I want to do. This is the content for this page. This is a lot of text just to fill the space. I will repeat

Accessibility Topics

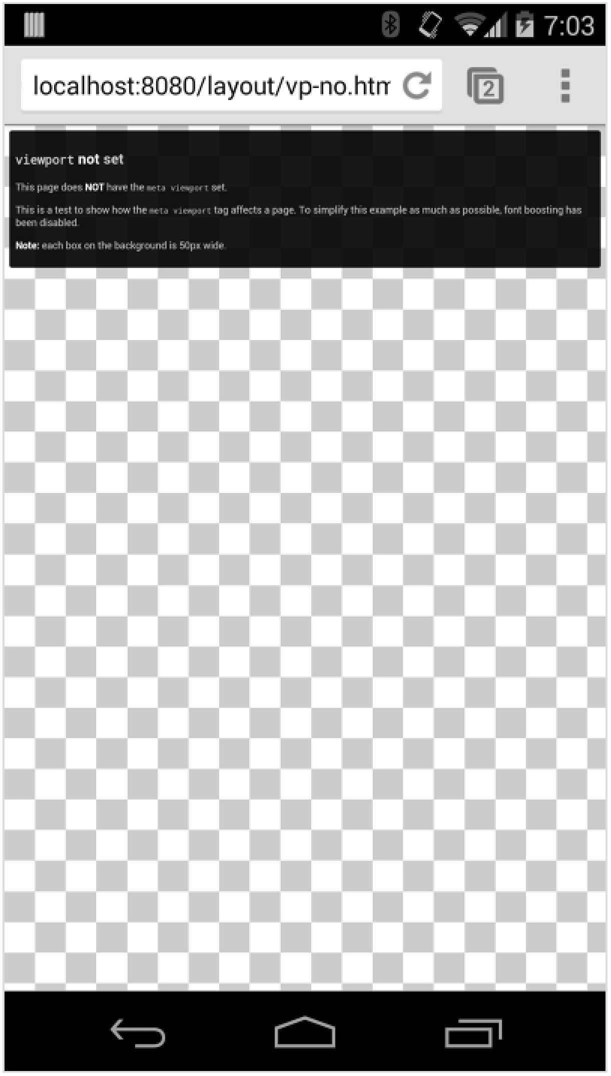
- Types of disabilities
  - Blindness
  - Deafblindness
  - Colorblindness
  - Low vision
  - Deafness
  - Dexterity/Motor
  - Cognitive
  - Seizure
- Assistive technologies



Bootstrap

Build responsive, mobile-first projects on the web with the world's most popular front-end component library.

Bootstrap is an open source toolkit for developing with HTML, CSS, and JS. Quickly prototype your ideas or build your entire app with our Sass variables and mixins, responsive grid system, extensive prebuilt components, and powerful plugins built on jQuery



`viewport` **not set**

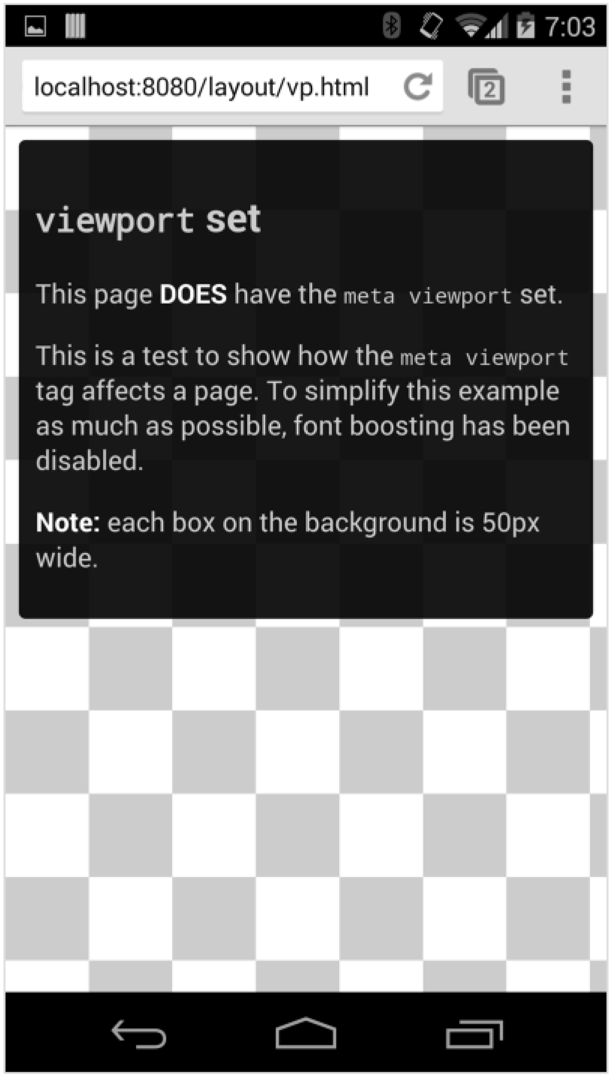
This page does **NOT** have the `meta viewport` set.

This is a test to show how the `meta viewport` tag affects a page. To simplify this example as much as possible, font boosting has been disabled.

Note: each box on the background is 50px wide.

META TAG VIEWPORT

```
<meta name="viewport" content="width=device
```



viewport set

This page **DOES** have the `meta viewport` set.

This is a test to show how the `meta viewport` tag affects a page. To simplify this example as much as possible, font boosting has been disabled.

Note: each box on the background is 50px wide.

Media queries

Jak można użyć media queries

1. W pliku css lub tagu <style>

```
@media (max-width: 768px) {
    /* your css goes here */
}
```

Jak można użyć media queries

2. Przy linkowaniu pliku css

```
<link media="(max-width: 768px)" href="tabl
```

Jak można użyć media queries

3. Przy imporcie w css

```
@import url("tablet.css") (max-width: 768px
```

Inne użyteczne media queries

```css
@media screen and (max-width: 768px) {}
```

```css
@media print {}
```

```css
@media (orientation: landscape) {}
```

```css
@media (orientation: portrait) {}
```

```css
@media (max-width: 800px) and (min-width: 400px
                (min-width: 1920px) {}
```

Pytanie do publiczności:

```
@media screen and (max-width: 900px)
       and (min-width: 600px), (min-width: 1100
    /* styles */
}
```

Przydatne linki

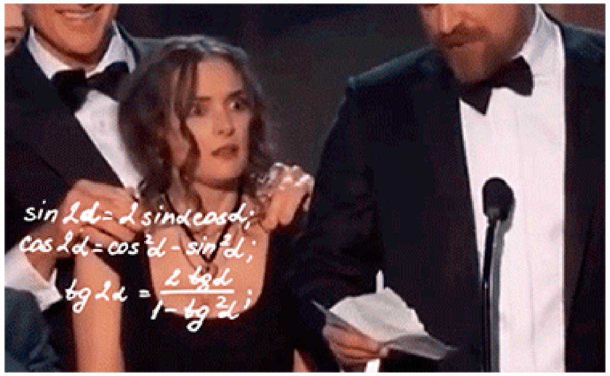
https://developer.mozilla.org/pl/docs/Mozilla/Mobile/Viewport_meta_tag

https://developer.mozilla.org/pl/docs/Web/CSS/Media_Queries/Using_media_queries

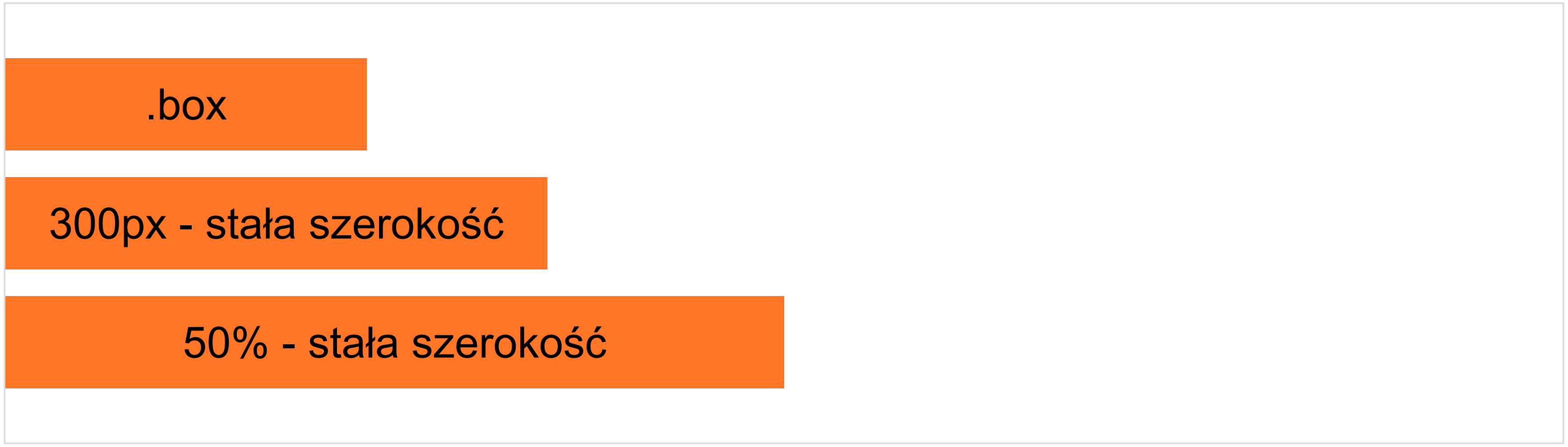
https://css-tricks.com/snippets/css/media-queries-for-standard-devices

Jak dodać dwie wartości w CSS?

calc()



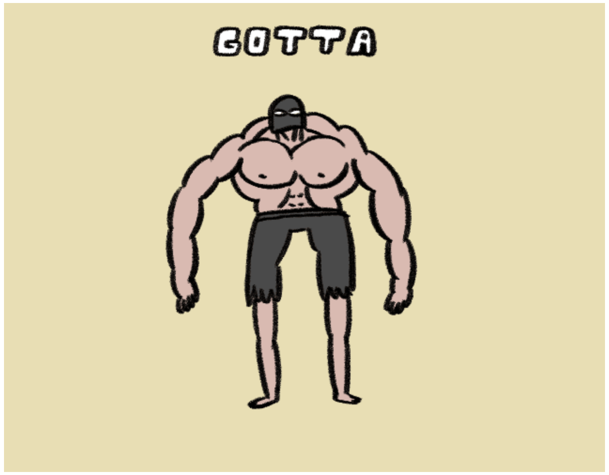
```css
.box {
    width: calc(100px + 100px)
}
```



Przydatne linki:

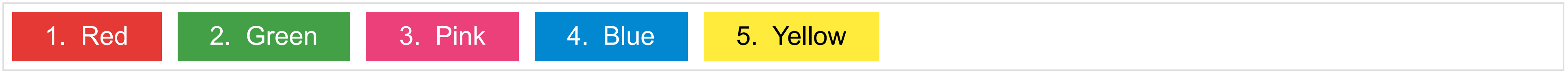
https://developer.mozilla.org/en-US/docs/Web/CSS/calc

Flexbox



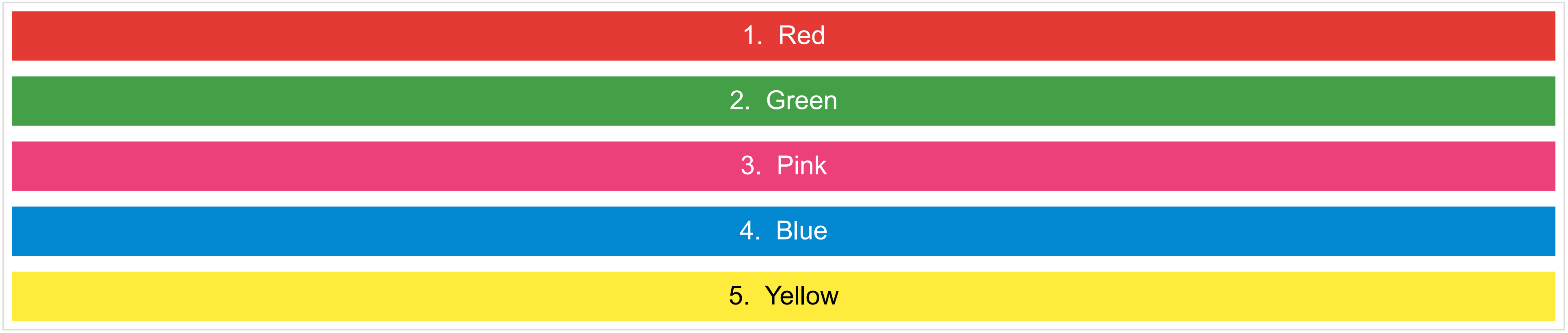
```html
<div class="container">
    <div class="item"></div>
    <div class="item"></div>
    <div class="item"></div>
    <div class="item"></div>
    <div class="item"></div>
</div>
```

```css
.container {
    display: flex;
}
```



Flex-direction

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



1. Red
2. Green
3. Pink
4. Blue
5. Yellow

Order

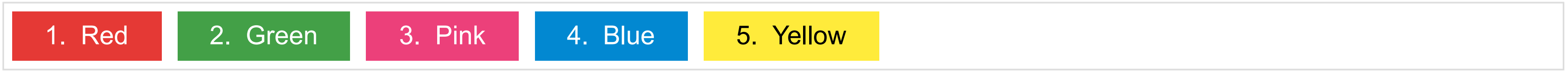
```
.item:nth-child(5) {
    order: -1;
}
.item:nth-child(2) {
    order: 1;
}
```

5. Yellow   1. Red   3. Pink   4. Blue   2. Green

Justify-content

FLEX-START

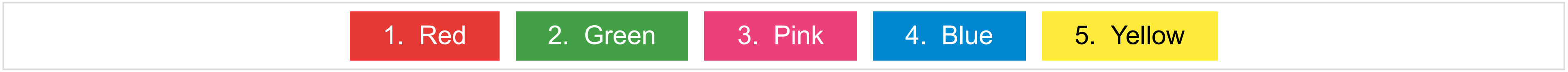
```css
.container {
    display: flex;
    justify-content: flex-start;
}
```



Justify-content

CENTER

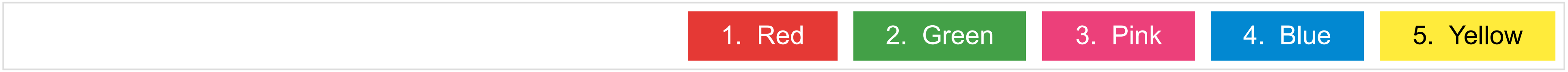
```css
.container {
    display: flex;
    justify-content: center;
}
```



Justify-content

FLEX-END

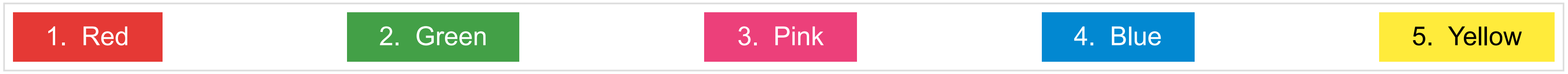
```css
.container {
    display: flex;
    justify-content: flex-end;
}
```



Justify-content

SPACE-BETWEEN

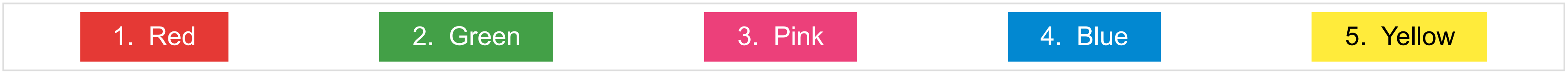
```css
.container {
    display: flex;
    justify-content: space-between;
}
```



Justify-content

SPACE-AROUND

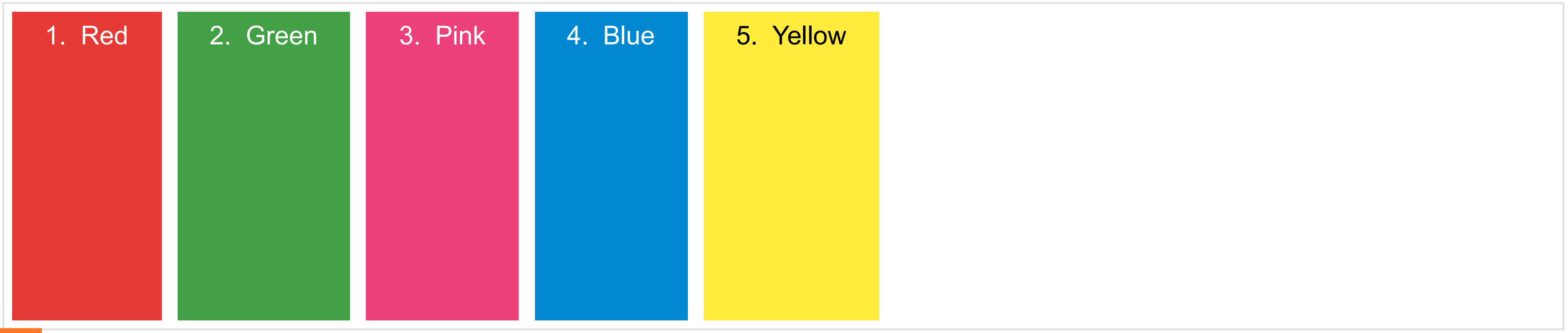
```
.container {
    display: flex;
    justify-content: space-around;
}
```



Align-items

ALIGN-STRETCH

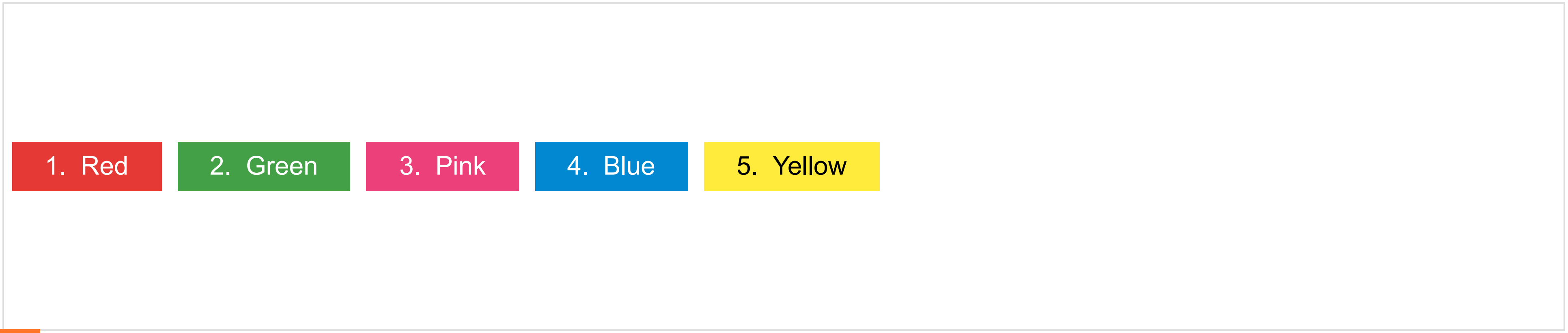
```
.container {
    display: flex;
    height: 200px;
    align-items: align-stretch;
}
```



Align-items

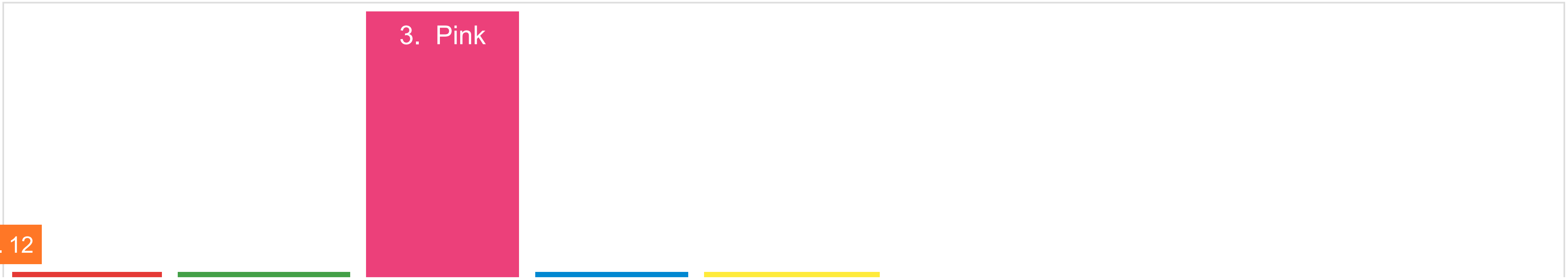
CENTER

```
.container {
    display: flex;
    height: 200px;
    align-items: center;
}
```



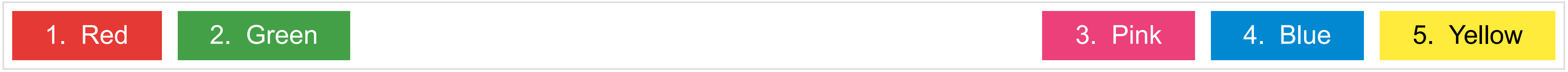
Align-self

```
.container {
    display: flex;
    height: 200px;
    align-items: flex-end;
}
.item:nth-child(3) {
    align-self: stretch;
}
```



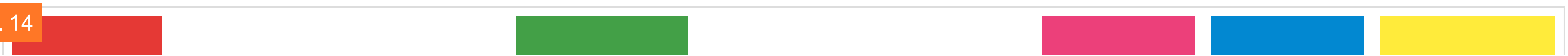
Margin

```css
.container {
    display: flex;
}
.item:nth-child(3) {
    margin-left: auto;
}
```



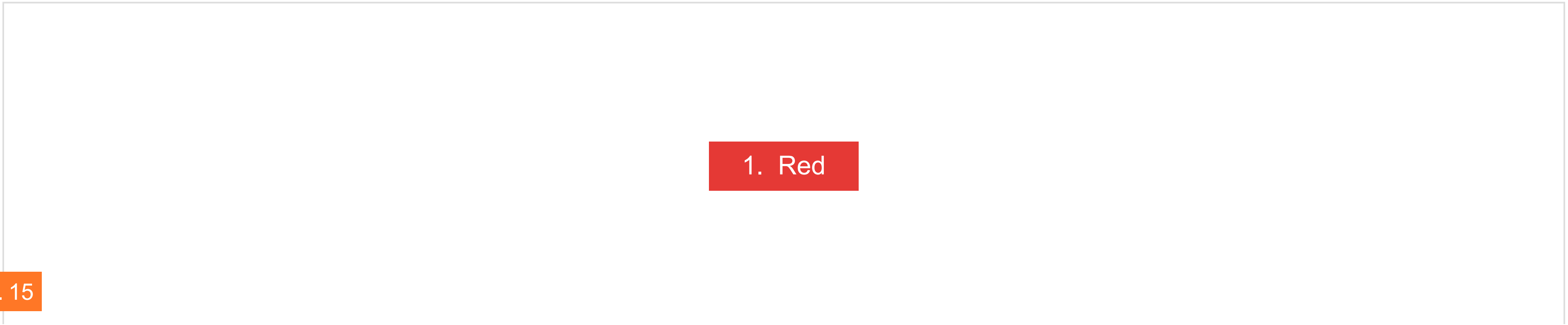
Margin

```
.container {
    display: flex;
    height: 200px;
}
.item:nth-child(1) {
    margin-right: auto;
}
.item:nth-child(3) {
    margin-left: auto;
}
```



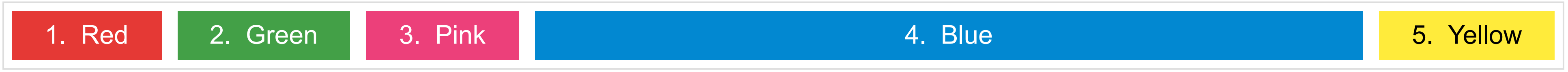
Margin

```
.container {
    display: flex;
    height: 200px;
}
.item {
    margin: auto;
}
```



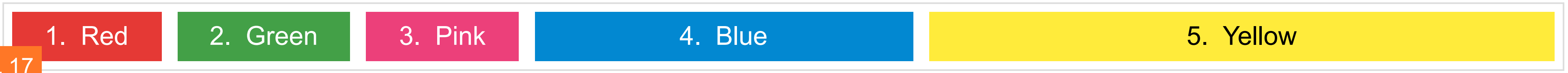
Flex-grow

```
.container {
    display: flex;
}
.item:nth-child(4) {
    flex-grow: 1;
}
```



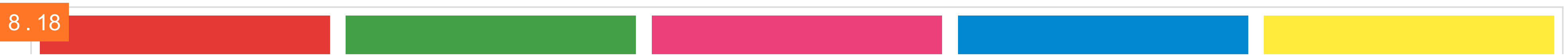
Flex-grow

```
.container {
    display: flex;
}
.item:nth-child(4) {
    flex-grow: 1;
}
.item:nth-child(5) {
    flex-grow: 2;
}
```



Flex-shrink

```css
.container {
    display: flex;
}
.item {
    flex-grow: 1;
    flex-basis: 20%;
}
.item:nth-child(1) {
    flex-shrink: 1;
}
```

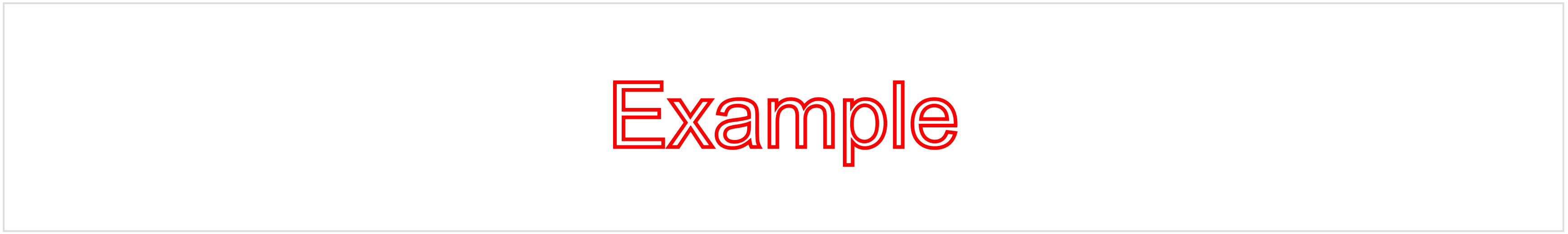


Przydatne linki:

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

Prefixy

```css
.box {
    -webkit-text-stroke: 2px red;
}
```



Example

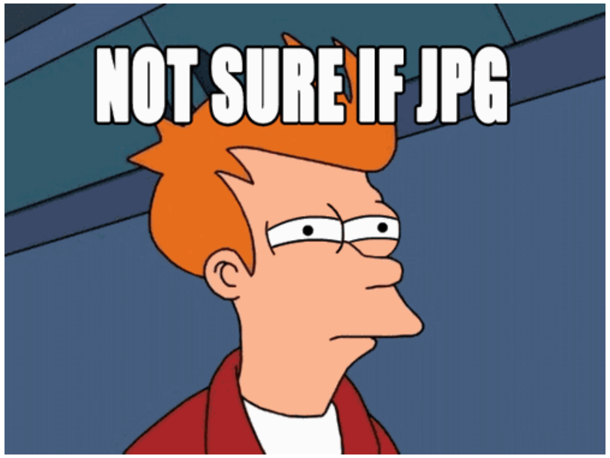
```css
.box {
    -webkit-text-stroke: 2px red;
       -moz-text-stroke: 2px red;
        -ms-text-stroke: 2px red;
         -o-text-stroke: 2px red;
}
```

Example

Przydatne linki:

https://developer.mozilla.org/en-US/docs/Glossary/Vendor_Prefix

Animacje



Transition

```css
.box {
    background: white;
    transition: 1s ease background;
}
.box:hover {
    background: #ff7726;
}
```

Example

@keyframes & animation

```
@keyframes first-animation {
    from {background-color: #FFFFFF;}
    to {background-color: #ff7726;}
}
.box:hover {
    background-color: #FFFFFF;
    animation: 2s first-animation 2;
}
```

Example

@keyframes & animation

```
@keyframes second-animation {
    0% {background-color: #FFFFFF;}
    50% {background-color: #ff7726;}
    100% {background-color: #FFFFFF;}
}
.box:hover {
    background-color: #FFFFFF;
    animation: 2s second-animation 2;
}
```

Example

Ograniczenia

- display
- visibility
- position
- etc...

Przydatne linki:

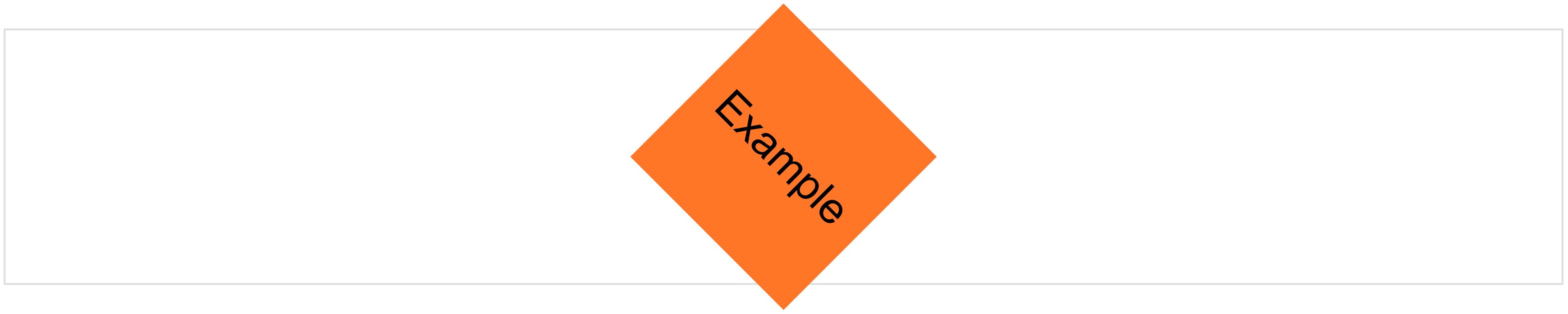
https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Transitions/Using_CSS_transitions

https://css-tricks.com/almanac/properties/a/animation

Transformacje

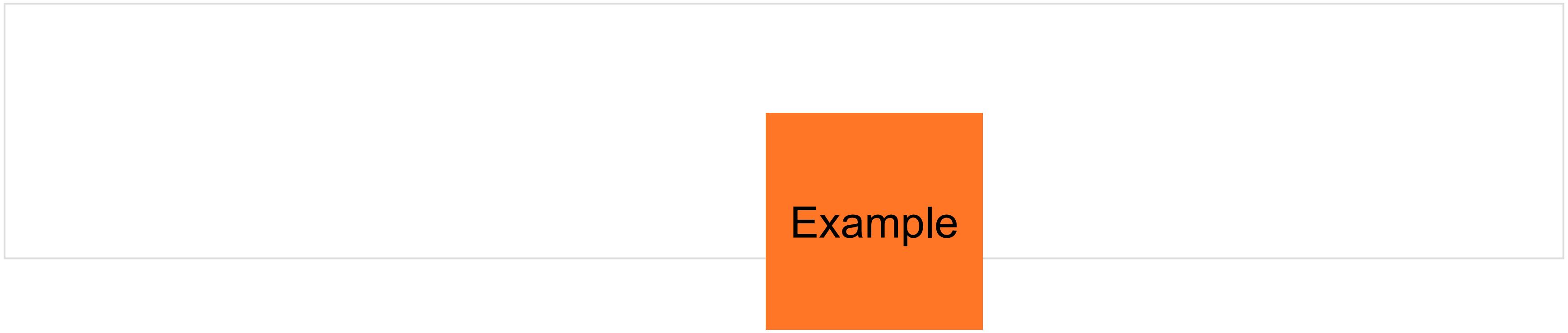
rotate(angle)

```
.box {
    transform: rotate(45deg);
}
```



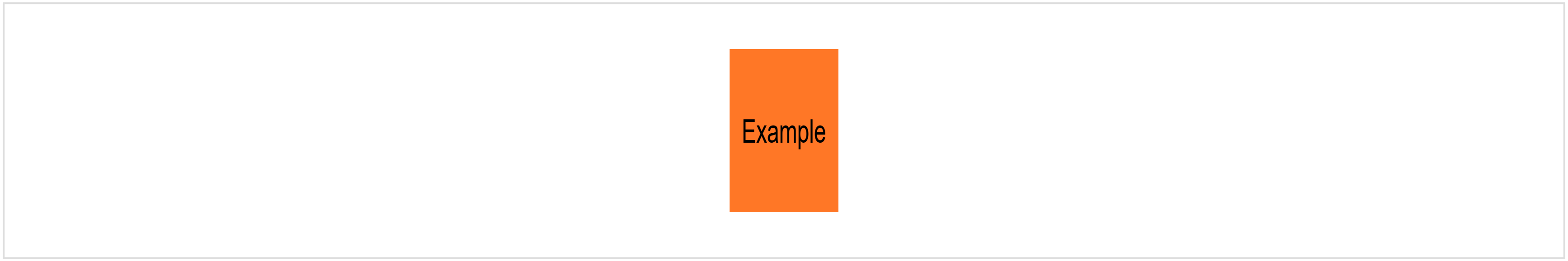
translate(x, y)

```css
.box {
    transform: translate(50px, 50px);
}
```



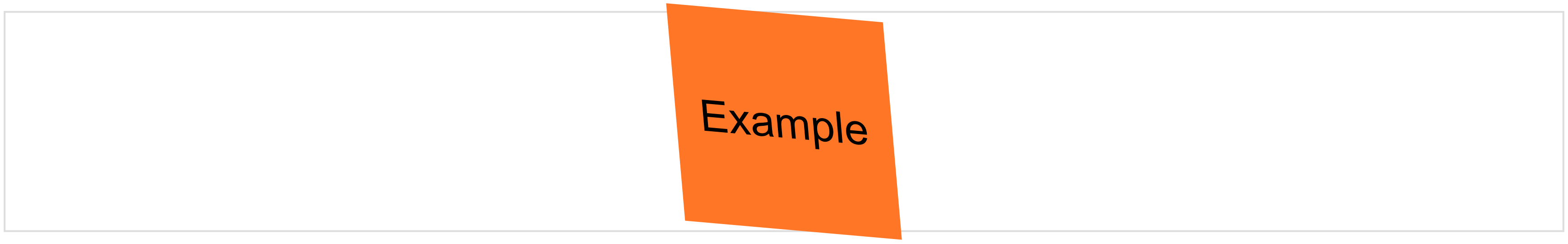
scale(x, y)

```
.box {
    transform: scale(0.5, 0.75);
}
```



skew(x, y)

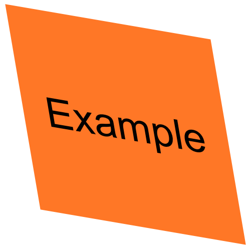
```
.box {
    transform: skew(5deg, 5deg);
}
```



3D

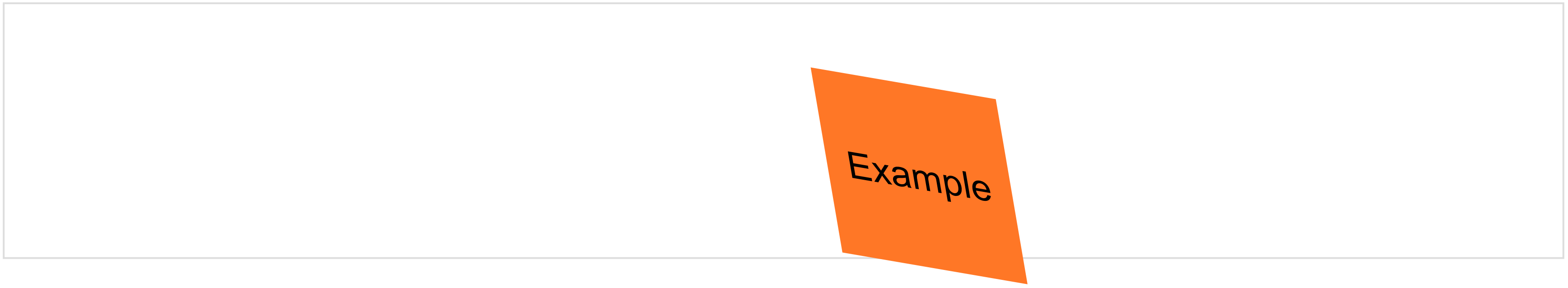
rotate3d(x, y, z)

```
.box {
    transform: rotate3d(1,1,0,45deg);
}
```



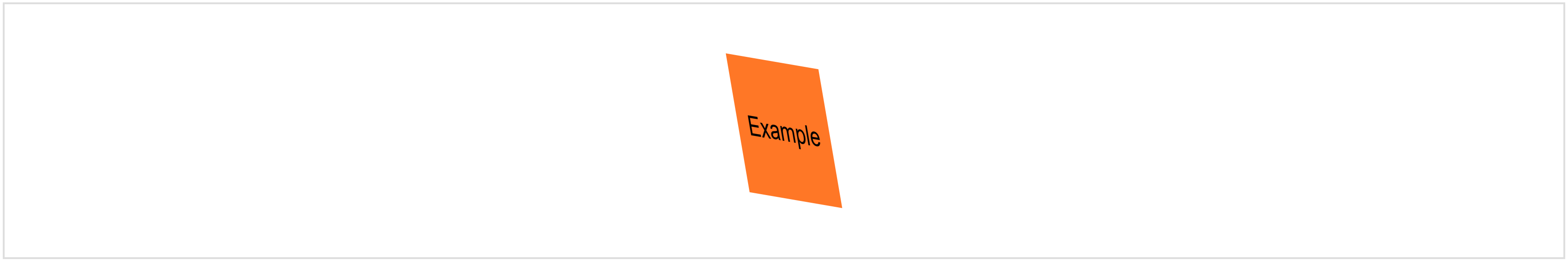
translate3d(x, y, z)

```
.box {
    transform: rotate3d(1,1,0, 45deg)
               translate3d(50px, 50px, 50px);
}
```



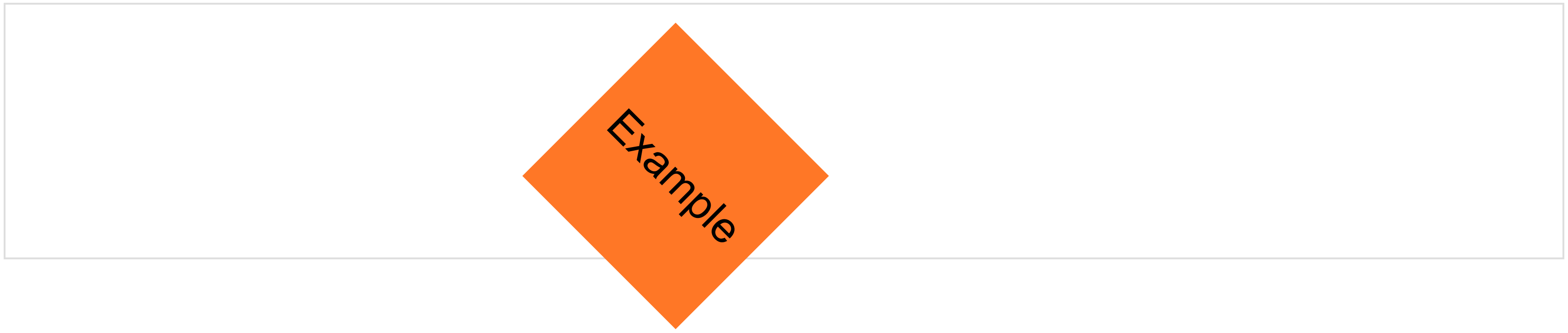
scale3d(x, y, z)

```css
.box {
    transform: rotate3d(1, 1, 0, 45deg)
               scale3d(0.5, 0.75, 1);
}
```



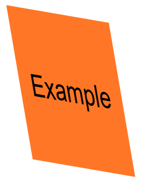
transform-origin

```
.box {
    transform: rotate(45deg);
    transform-origin: 0 0;
}
```



transform-style

```
.box {
    transform: rotate3d(1, 1, 0, 45deg)
               scale3d(0.5, 0.75, 1);
    transform-style: preserve-3d;
}
```



Przydatne linki:

https://css-tricks.com/almanac/properties/t/transform

https://css-tricks.com/almanac/properties/t/transform-origin

https://css-tricks.com/almanac/properties/t/transform-style

Problem:

- Pracujesz nad projektem z innymi programistami,
- Masz zmienic coś w CSS napisanym przez kolegę, ktory od dawna z Wami nie pracuje,
- Nazywał on elementy (selektory) zupełnie inaczej niż Ty,
- Marnujesz czas na próbie zrozumienia "co autor miał na myśli" i który selektor za co odpowiada.

Jak można tego uniknąć?

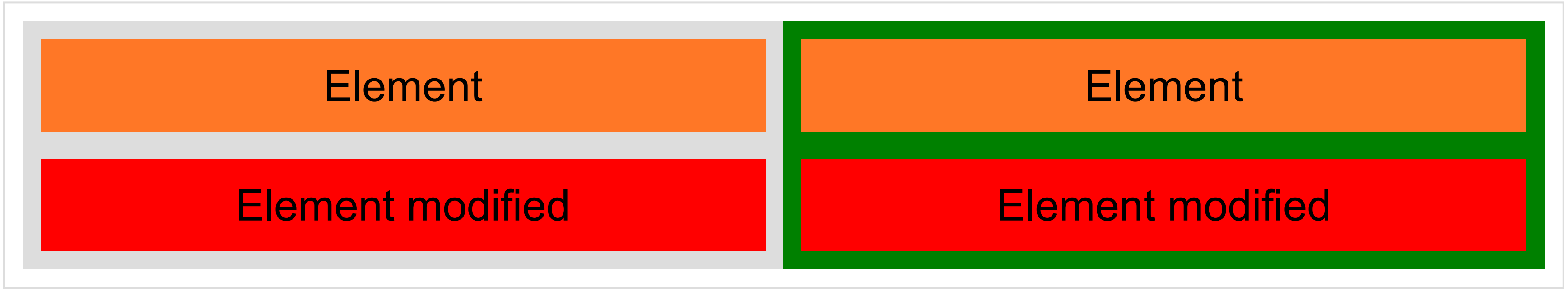
Konwencje CSS

BEM

http://getbem.com

Block-Element-Modifier

```
.block {
    background: #DDDDDD;
}
.block--modifier {
    background: green;
}
.block__element {
    background: #ff7726
}
.block__element--modifier {
    background: red;
}
```



Block-Element-Modifier

```html
<div class="block">
    <div class="block__element">
        Element
    </div>
    <div class="block__element block__element--modifier">
        Element modified
    </div>
</div>
<div class="block block--modifier">
    <div class="block__element">
        Element
    </div>
    <div class="block__element block__element--modifier">
        Element modified
    </div>
</div>
```

SUIT CSS

http://suitcss.github.io

SUIT CSS: Utilities

```
u-[sm-|md-|lg-]utilityName
```

```
<div class="u-cf">
    <a class="u-floatLeft" href="{{url}}">
        <img class="u-block" src="{{src}}" alt=
    </a>
    <p class="u-sizeFill u-textBreak">
        …
    </p>
</div>
```

SUIT CSS: Components

```
[namespace-]ComponentName[-descendentName][--modifierNa
```

SUIT CSS: namespace

```
.pgs-Button { /* … */ }
.pgs-Tabs { /* … */ }
```

SUIT CSS: ComponentName

```
.PgsComponent { /* … */ }
```

```
<article class="PgsComponent">
    …
</article>
```

SUIT CSS: ComponentName--modifierName

```
/* Core button */
.Button { /* … */ }
/* Default button style */
.Button--default { /* … */ }
```

```
<button class="Button Button--default" type="bu
```

SUIT CSS: ComponentName-descendentName

```html
<article class="Pgs">
    <header class="Pgs-header">
        <img class="Pgs-avatar" src="{{src}}" alt="{{alt}}"
        …
    </header>
    <div class="Pgs-bodyText">
        …
    </div>
</article>
```

SUIT CSS: ComponentName.is-stateOfComponent

```
.Pgs { /* … */ }
.Pgs.is-presenting { /* … */ }
```

```
<article class="Pgs is-presenting">
    …
</article>
```

Po co używac konwencji?

- ustalony sposób "nazywania" elementów w CSS
- reużywalne elementy
- mniej kosztowne utrzymanie kodu

Pytania?

Akademia Developera

Wprowadzenie do PHP

27.03.2018, 18:00 • Uniwersytet Rzeszowski, Budynek A0, sala 127

softwaretalks.pl/wydarzenia/akademia-developera-wprowadzenie-php

Akademia Developera

Dziękuję za uwagę !

Odwiedź:

www.facebook.com/AkademiaDeveloperaRzeszow

KONRAD PRZYDZIAŁ • FRONT-END DEVELOPER