



Bootstrap



Wstęp do Bootstrapa

Czyli co to jest i z czym się to je

Spis treści i nasz plan

- Czym jest Bootstrap
- Plusy i minusy
- Filozofia Bootstrapa
- Co z BEM-em?
- Dokumentacja
- Instalacja - możliwości
- Layout
- Podstawowe elementy
- Narzędzia
- Najważniejsze komponenty
- Customizacja w wersji Sass
- Dodatek: dodatkowe komponenty
- Dodatek: gotowe szablony

Czym jest Bootstrap

Bootstrap to najpopularniejszy **framework CSS-owo - JS-owy** z komponentami. Nastawiony na **responsywność, accessibility i mobile-first**.

Pierwsza wersja powstała w 2011.

Będziemy mówili o najnowszym **Bootstrap 4**, który zasadniczo różni się od poprzednich wersji.



Do czego i kto go używa?

Pozwala bardzo szybko prototypować, a także tworzyć całe serwisy.

Chętnie wykorzystywany od amatorów do dużych firm.

Od razu dostaje się schludny wygląd.

Zobaczymy dziś, że są do wyboru także dużo profesjonalniejsze wyglądy, a nawet można szybko robić swoje.

Plusy i minusy

Plusy

- Możliwość bardzo szybkiego prototypowania
- Bardzo bogata biblioteka komponentów
- Można stworzyć schludną stronę, bez umiejętności designerskich i wyczucia
- Pozwala na rozszerzanie o dodatkowe możliwości i zmianę wyglądu
- Własna, bardzo spójna filozofia

Minusy

- Dosyć ciężkie rozwiązanie, to spora ilość CSS i JS
- Wymaga do pełnego działania **jQuery**
- W przeszłości mylnie często wykorzystywany do samego systemu grida
- Niezgodny z BEM, OOCSS itd.



Jak wygląda praca z Bootstrapem?

Filozofia Bootstrap

1. Oparty na **własnych klasach CSS** - wszystko jest zrobione w ten sposób
2. Oparty na kilku rozmiarach (breakpointach)
 - **0 - 575px** - **xs**, extra small, np. małe smartfony
 - **576 - 767px** - **sm**, small, np. większe smartfony
 - **768 - 991px** - **md**, medium, np. tablety
 - **992 - 1199px** - **lg**, large, np. komputery
 - **1200+px** - **xl**, extra large, np. komputery z wysoką rozdzielczością

Co z BEM-em?


Nic :) Bootstrap istniał przed popularyzacją rozwiązań mających poukładać CSS modułowo, takich jak BEM, czy OOCSS.

Korzystając z Bootstrapa korzystamy głównie z jego klas.

Oczywiście można dalej tworzyć własne elementy, sekcje itd i stosować tam BEM. Jednak może to wyglądać nieco chaotycznie.



Dokumentacja



Już od wersji 2 była dobra. Teraz jest jeszcze lepsza! Warto korzystać na co dzień pracując z Bootstrapem.

getbootstrap.com/docs

Instalacja - możliwości

1. Pobranie.
2. Gotowe komponenty React, Angular i inne.
3. CDN.
4. Gotowy szablon.
5. npm.

Poznamy trzy ostatnie - najważniejsze - opcje.

Poznajmy drugą na żywo!



Stosowanie Bootstrapa w praktyce - najpierw kilka pomocnych narzędzi



Layout



Kontener

Czyli .container



display: ...

- d-flex
 - d-inline
 - d-block
 - d-inline-block
-



System grida

Zobaczmy na żywo - najpierw stały grid.



Grid o określonych proporcjach



Grid responsywny

Oraz filozofia xs, sm, ..., xl



Ręczne przerwanie wiersza

- Nowy .row
- Lub .w-100



Przesunięcie

Czyli offset.



Grid w gridzie?



Stylowanie podstawowych elementów



Typografia

Jest gotowa za Ciebie!

Sprawdźmy.

Stylowanie obrazów

1. Responsywny obraz - `img-fluid`
2. Miniaturka - `.img-thumbnail`
3. Zaokrąglone - `.rounded`



Tabele

Czyli `.table` i dzieje się magia.



Ciemna tabela

 .table-dark

+

.thead-light

.thead-dark



Tabela w paski

`.table-striped`



Hover na wierszach

Pamiętaj o `thead`, `tbody`(, `tfoot`)!

`.table-hover`



Tabela na mobile? Kompaktowa

.table-sm

Jeszcze lepsza tabela - responsywna

Polecam dodawać do każdej większej.

`.table-responsive`

lub

`.table-responsive-{rozmiar np. md}`



Przedstawienie komponentów

Najfajniejsze w Bootstrapie są gotowe komponenty :)

Alert

Komponent do komunikatów.

A simple primary alert—check it out!

A simple secondary alert—check it out!

A simple success alert—check it out!

A simple danger alert—check it out!



Linki w alertach

Używaj klasy `.alert-link`

Możliwość zamknięcia alertu

Szablon:

```
<div class="alert alert-warning alert-dismissible fade show" role="alert">
```

Tekst...

```
<button type="button" class="close" data-dismiss="alert" aria-label="Zamknij">
```

```
<span aria-hidden="true">&times;</span>
```

```
</button>
```

```
</div>
```

Badge

Mały znaczek np. z ikoną, czy liczbą.

Example heading **New**

Stały lub z linkiem.

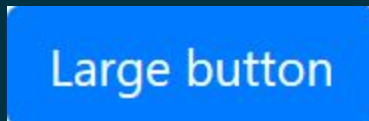
Kolory badge

badge-primary
badge-secondary
badge-success
badge-danger
badge-warning
badge-info
badge-light
badge-dark



Button

Przyciski. Posiadają bardzo wiele możliwości, stylów i są bardzo dopracowane.



Kolory przycisków

btn-primary

btn-secondary

btn-success

btn-danger

btn-warning

btn-info

btn-light

btn-dark

btn-link



Przyciski z samą ramką

Warto ich używać np. do przycisków o mniejszym znaczeniu.

Tworzymy poprzez użycie **btn-outline-{kolor}** zamiast **btn-{kolor}**.

Rozmiary przycisków

- Domyślny
- Duży - .btn-lg
- Mały - .btn-sm

Zablokowany przycisk

Wystarczy użyć przycisku (`button`, `input`) i nadać mu atrybut `disabled`.

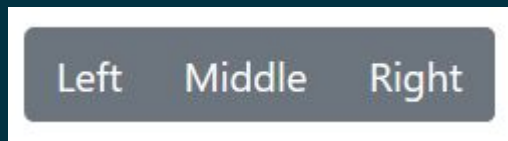
Istnieje również możliwość zablokowania przycisku na innym elemencie (np. linku *a*), jednak nie jest to zalecane.

Grupa przycisków

Pozwala łączyć wiele przycisków w jedną grupę - po prostu zlepia przyciski.

Przydatne np. do paginacji.

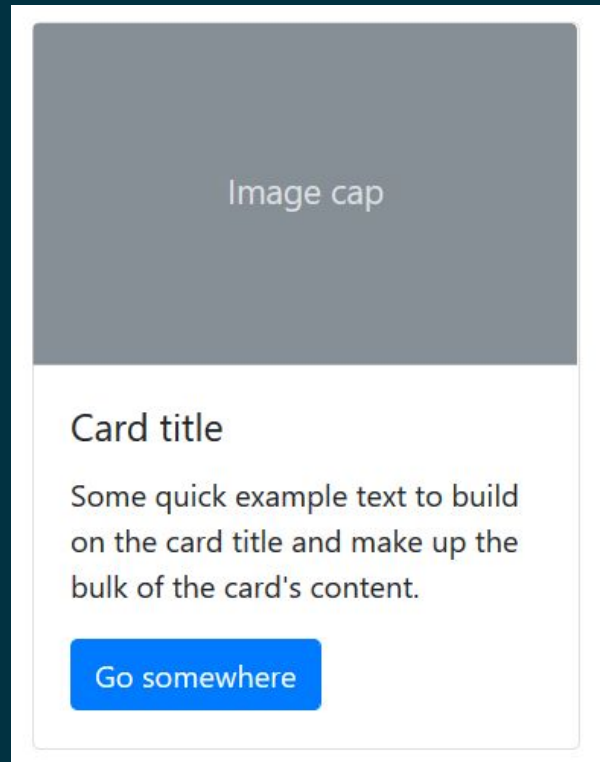
Wystarczy przyciski umieścić we wspólnym np. divie z klasą **.btn-group**.



Card

Komponent karty - składający się z kilku elementów kefelek.

Posiada wiele możliwości, omówimy najważniejsze z nich.



Jak użyć karty? Szablon

```
<div class="card" style="width: 18rem;">
```

```

```

```
<div class="card-body">
```

```
<h5 class="card-title">Tytuł</h5>
```

```
<p class="card-text">Tekst...</p>
```

```
<a href="#" class="btn btn-primary">Link</a>
```

```
</div>
```

```
</div>
```

Jak użyć karty? Szablon poziomy

```
<div class="card" style="width:450px">  
  <div class="row no-gutters">  
    <div class="col-md-4">  
        
    </div>  
    <div class="col-md-8">  
      <div class="card-body">  
        <h5 class="card-title">Tytuł</h5>  
        <p class="card-text">Tekst...</p>  
      </div>  
    </div>  
  </div>  
</div>
```

Układanie kart

Kart można używać w gridzie.

Można również użyć kontenera z klasą **.card-deck**, co ułoży je równo obok siebie.

Istnieje jeszcze jeden ciekawy sposób, tzw. Mansory cards - używamy do tego kontenera z klasą **.card-columns**.

Grupy list w środku kart

Przydatne dla zawartości ułożonych pod sobą mniejszych kafelek.

W kontenerze **.list-group.list-group-flush** umieszczamy elementy blokowe z klasą **.list-group-item**.

Nagłówek i stopka

- Nagłówek - **.card-header** - jako pierwszy element w karcie.
- Stopa - **.card-footer** - jako ostatni element w karcie.

Kolory kart

Dodajemy do `.card` kolor tekstu (np. `.text-white`) oraz jeden z kolorów tła:

`bg-primary`

`bg-secondary`

`bg-success`

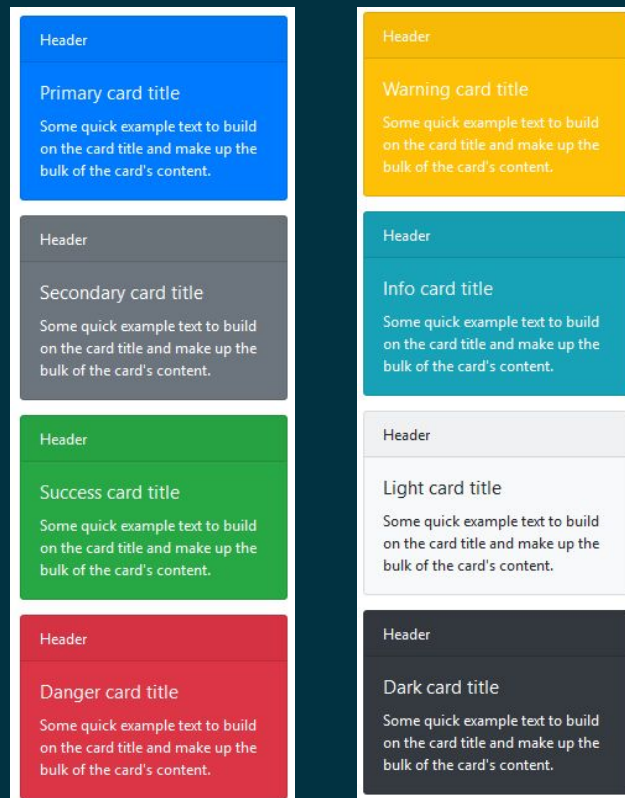
`bg-danger`

`bg-warning`

`bg-info`

`bg-light`

`bg-dark`



Carousel

Karuzela, slideshow, slider - różnie nazywany komponent do pokazywania kilku treści naprzemiennie.

Tak - dostajesz to **od razu używając Bootstrapa** :)

Ponieważ wymaga to większej ilości kodu - przeanalizujemy na żywo kod z przykładów z dokumentacji.



Collapse

Collapse, wykorzystywany także jako harmonijka, akordeon czy spoiler. Pozwala ukrywać część treści i wyświetlać ją dopiero kiedy użytkownik się na to zdecyduje.

[Collapsible Group Item #1](#)

Anim pariatur cliche reprehenderit, enim eiusmod high life accusamus terry richardson ad squid. 3 wolf moon officia aute, non cupidatat skateboard dolor brunch. Food truck quinoa nesciunt laborum eiusmod. Brunch 3 wolf moon tempor, sunt aliqua put a bird on it squid single-origin coffee nulla assumenda shoreditch et. Nihil anim keffiyeh helvetica, craft beer labore wes anderson cred nesciunt sapiente ea proident. Ad vegan excepteur butcher vice lomo. Leggings occaecat craft beer farm-to-table, raw denim aesthetic synth nesciunt you probably haven't heard of them accusamus labore sustainable VHS.

[Collapsible Group Item #2](#)

[Collapsible Group Item #3](#)

Zasada działania collapse 1/2

1. Aby użyć collapse potrzebujesz linku lub przycisku oraz elementu, który będzie wyświetlany i ukrywany.
2. Na elemencie chowanym ustaw klasę **.collapse** oraz nadaj mu unikalne ID.
3. Na linku/przycisku sterującym dodaj atrybuty:
 - data-toggle="collapse"
 - data-target="#" + ID elementu chowanego.

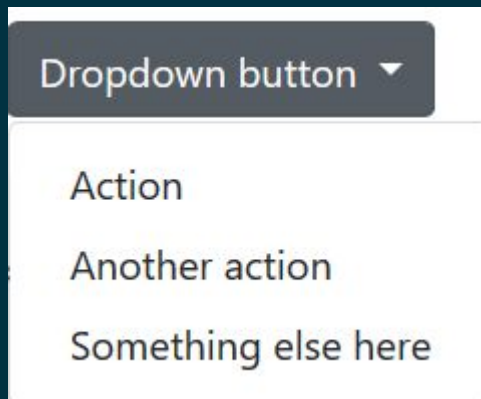


Akordeon

Zerknijmy do gotowego kodu, żeby sprawdzić w jaki sposób wykorzystać collapse do stworzenia akordeonu.

Dropdown

Dropdown pozwala wysunąć menu z linkami lub przyciskami po kliknięciu na przycisk.



Podstawowy dropdown

```
<div class="dropdown">
```

```
  <button class="btn btn-secondary dropdown-toggle" type="button" id="dropdownMenuButton"  
  data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
```

Przycisk

```
</button>
```

```
<div class="dropdown-menu" aria-labelledby="dropdownMenuButton">
```

```
  <a class="dropdown-item" href="#">Akcja 1</a>
```

```
  <a class="dropdown-item" href="#">Akcja 2</a>
```

```
  <a class="dropdown-item" href="#">Akcja 3</a>
```

```
</div>
```

```
</div>
```

Stylowanie przycisku dropdown

Przycisk dropdown poddaje się takim samym zasadom jak każdy inny przycisk.

Można więc zmieniać mu rozmiar np. poprzez nadanie klasy **.btn-lg** lub kolor za pomocą np. **.btn-success**.



Rozdzielenie grup akcji

Używamy do tego elementu z klasą **.dropdown-divider**.

Kierunek otwarcia menu

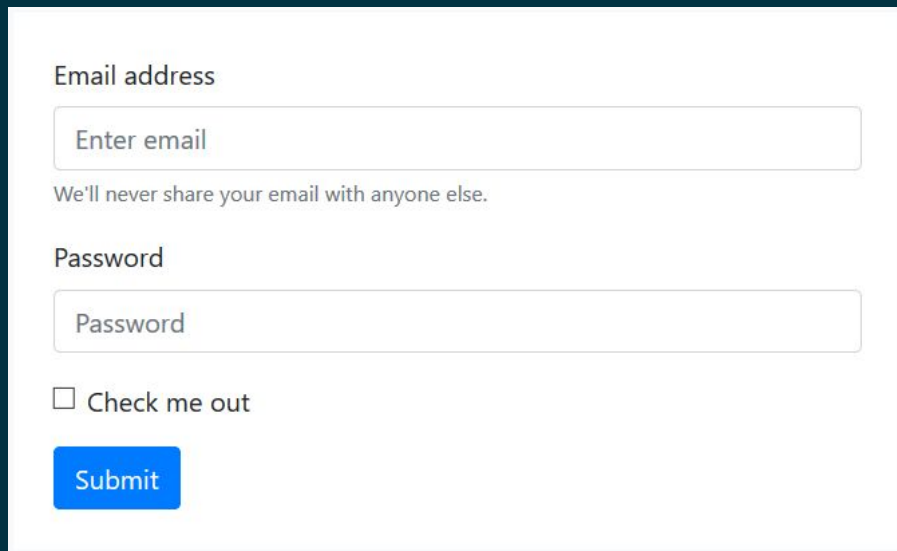
Można go wyznaczać za pomocą klas:

- **.dropup**
- **.dropleft**
- **.dropright**

Form i Forum group

Formularze to ogromna część Bootstrapa. Można tworzyć bardzo eleganckie i złożone pola w bardzo szybki sposób.

Ponieważ formularze mają bardzo wiele możliwości - poznamy teraz większość z nich - na żywo.



Email address

We'll never share your email with anyone else.

Password

☐ Check me out

Jumbotron

Jumbotron, zwany także komponentem hero, świetnie nadaje się na przedstawienie produktu, czy wstępu do aplikacji.

Hello, world!

This is a simple hero unit, a simple jumbotron-style component for calling extra attention to featured content or information.

It uses utility classes for typography and spacing to space content out within the larger container.

[Learn more](#)

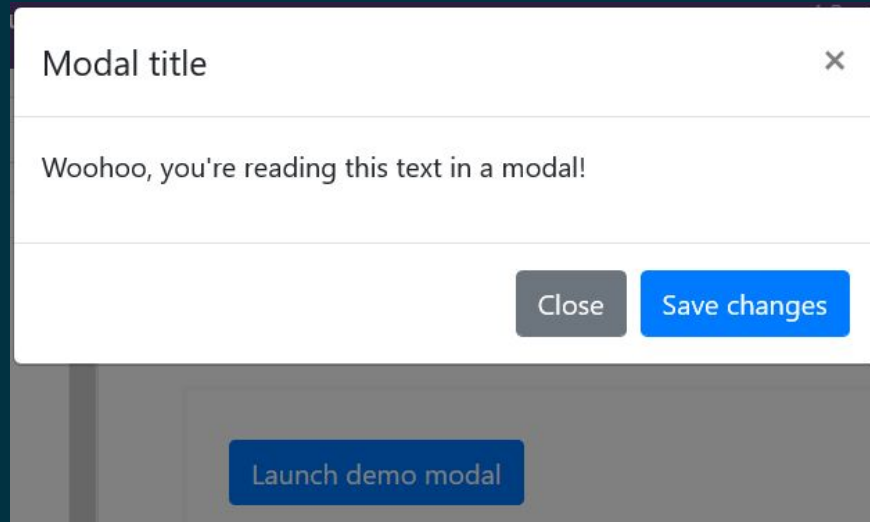
Szablon jumbotrona

```
<div class="jumbotron">  
  <h1>Tytuł</h1>  
  <p class="lead">Lorem ipsum treść...</p>  
  <hr>  
  <p>Kolejna treść...</p>  
  <a class="btn btn-primary btn-lg" href="#" role="button">Przycisk</a>  
</div>
```

Modal

Okienka modalne, czyli takie wyskakujące nad treścią.

One również są gotowe kiedy masz Bootstrapa! Zerknijmy na żywo jak to działa.



Nav

Uniwersalny komponent nawigacji, który może być umieszczony w dowolnym miejscu strony.

Active

Link

Link

Disabled

Jak użyć nawigacji?

1. Stwórz kontener `ul` o klasie `.nav`.
2. Każdy `li` powinien mieć klasę `.nav-item`.
3. Każdy odnośnik w `li` powinien mieć klasę `.nav-link`.
4. Jeżeli chcesz zaznaczyć aktualny odnośnik - użyj na nim klasy `.active`.

Taby - zakładki

Można zmienić styl na zakładki, takie jak w przeglądarce.

W tym celu nadajemy kontenerowi **ul** dodatkową klasę: **.nav-tabs**.

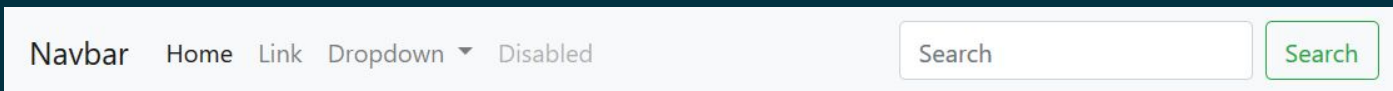


Nav jest wielki :)

Zachęcam do sprawdzenia dokumentacji - nav ma bardzo wiele opcji i może łączyć się z innymi komponentami.

Navbar

Kolejny komponent nawigacyjny - tym razem najczęściej używany jako główne menu na stronie.



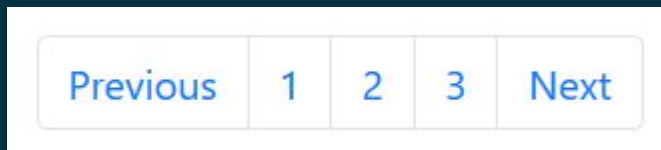
Ma wiele możliwości użycia, dlatego zobaczmy w praktyce użycie navbar z:

- Logo
- Menu
- Wyszukiwarką
- Automatycznym przyklejaniem się do góry przy scrollowaniu

Pagination

Paginacja - podział wielu elementów na podstrony.

Można wykonać ją samemu bądź skorzystać z gotowca w postaci komponentu **.pagination**.



Użycie pagination

Użycie tego komponentu jest proste:

1. Stwórz wrapper - element **nav**, a w nim kontener - element **ul** z klasą **.pagination**.
2. Każdy **li** w środku powinien mieć klasę **.page-item**.
3. Każdy odnośnik w **li** powinien mieć klasę **.page-link**.

Progress

Elegancki komponent, służący pokazywaniu aktualnego postępu.



Użycie progress

Użycie progress jest bardzo proste i sprowadza się do dwóch elementów:

1. Kontenera z klasą **.progress**.
2. Elementu blokowego w środku z klasą **.progress-bar**.
3. Na wewnętrznych elemencie ustawia się styl CSS width, żeby wyznaczyć długość.

Zaleca się dodanie odpowiednich informacji aria dla WCAG, oto pełen przykład:

```
<div class="progress">  
  <div class="progress-bar" role="progressbar" style="width: 25%;"  
    aria-valuenow="25" aria-valuemin="0" aria-valuemax="100">25%</div>  
</div>
```

Style paska postępu

Można zmienić m.in.:

1. Kolor: używając na wewnętrznym pasku klasy **.bg-{kolor}**, np. **.bg-success**.
2. Paski: używając na wewnętrznym pasku klasy **.progress-bar-striped**.
3. Animację pasków: do p. 2 dodając klasę **.progress-bar-animated**.



Jak zmieniać postęp?

Zmieniając w kodzie styl width paska wewnętrznego.

Zobaczmy to na żywo.

Spinner

Spinner, czy loader - animowany element pokazujący, że coś się aktualnie wczytuje.



Polecany szablon

```
<div class="spinner-border" role="status">
```

```
  <span class="sr-only">Loading...</span>
```

```
</div>
```

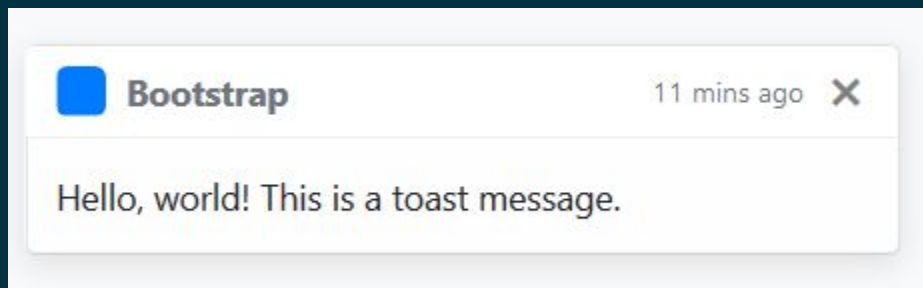
Zmiana stylu

Można zmienić m.in.:

1. Całkiem styl na inny spinner: **.spinner-grow**.
2. Kolor za pomocą **.text-{kolor}**, np. **.text-warning**.
3. Wielkość na małą dodając odpowiednio **.spinner-border-sm** lub **.spinner-grow-sm**.

Toast

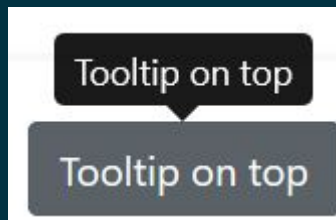
Notyfikacja, powiadomienie. Elegancki komponent, który pozwoli powiadomić użytkownika o dowolnej sprawie.



Zobaczmy ten komponent na żywo.

Tooltip

Komponent dzięki któremu można wyświetlić szybką informację kontekstową - np. po najechaniu na jakiś element (domyślnie).



Użycie tooltip

Trzeba na dowolnym elemencie, po najechaniu na który ma się pojawić tooltip dodać atrybuty:

- `data-toggle="tooltip"`
- `data-placement= {top, bottom, right lub left}`
- `title= {treść tooltipa}`
- `data-html="true"` - opcjonalnie - jeżeli chcesz umieścić w treści HTML.

Tooltip nie działa? Włącz go :)

```
$(function () {  
    $('[data-toggle="tooltip"]').tooltip();  
})
```




Narzędzia



Cienie

- `.shadow-none`
 - `.shadow-sm`
 - `.shadow`
 - `.shadow-lg`
-

+

+



Ułożenie elementów liniowych, głównie tekstu

- `.text-justify`
 - `.text-left`
 - `.text-center`
 - `.text-right`
 - Istnieją wersje responsywne np. `.text-md-right`
-

Wyrównanie

- .float-left
- .float-right
- .float-none
- .float-sm-left
- .float-sm-right
- .float-sm-none
- .float-md-left
- .float-md-right
- .float-md-none
- .float-lg-left
- .float-lg-right
- .float-lg-none
- .float-xl-left
- .float-xl-right
- .float-xl-none



Widoczność

- .visible
- .invisible



Customizacja zmiennych w wersji Sass

Co potrzebujesz?

Aby móc zmieniać wewnętrzne ustawienia Bootstrapa, jak np. kolorystykę, zaokrąglenia i prawie wszystko inne potrzebne będą:

1. Bootstrap zainstalowany za pomocą npm.
2. Sass.
3. Nadpisanie odpowiednich zmiennych w odpowiedniej kolejności.

Zobaczmy to na żywo.

Skąd wiedzieć jakie zmienne można nadpisać?

Z dokumentacji :)

Dobre miejsce gdzie warto zacząć to:

<https://getbootstrap.com/docs/4.3/getting-started/theming/>

Spróbujmy zmienić główną kolorystykę całego serwisu jedną zmienną.



Dodatek: gotowe komponenty i szablony

Bootstrap to biblioteka świetnych komponentów, ale jest ich więcej!

Ponieważ sposób, w jaki zbudowany jest Bootstrap 4 pozwala na jego rozszerzanie i wykorzystywanie jego możliwości istnieją liczne dodatkowe komponenty jak np. kalendarze.

Można szukać ich w wielu miejscach, ja polecam jedną z największych, sprawdzonych list:

<https://github.com/coliff/awesome-bootstrap-1>

MDBootstrap

Użycie najnowszego designu by Google - Material Design - wprost w komponentach Bootstrap.

Świetny, odświeżony wygląd.

Polski startup, który podbija programistyczny świat :)

Oficjalne, płatne szablony

Można również przeglądać na stronie

<https://themes.getbootstrap.com/>

oficjalne szablony, które można zakupić. Ich cena waha się od \$20 - \$100, ale są przygotowane bardzo profesjonalnie.



Dzięki za udział w części teoretycznej!
Czas na projekt :)
