

NATALIA SOKOŁOWSKA

WYZWANIA BEDEPROGRAMISTKA

**Poznam
Bootstrapa 4
ze
Strefą Kursów**

BEDEPROGRAMISTKA.PL

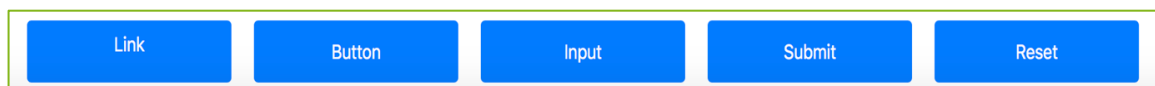
Komponenty są to elementy stron internetowych. Przykładem komponentów może być nawigacja, przyciski, komentarze, nagłówki. Są to bardzo popularne elementy, które składają się z kodu *HTML*, *CSS* i często *JS*. Samodzielnie zaprojektowanie ich może wymagać wiele pracy i wysiłku. Bootstrap oferuje nam już gotowe komponenty, dzięki czemu ich czas tworzenia skraca się do niezbędnego minimum.

Przyciski – button

Tworzenie przycisków jest dziecinnie proste. Wystarczy użyć klasy *.btn*. Zawsze rozpoczynamy od tej klasy, jeśli chcemy stworzyć jakiś *<button>*.

- *.btn* – bazowa klasa do tworzenia przycisków (możemy ją dodawać do elementu *<button>*, *<input>*, *<submit>*, *<reset>* lub np. do tagu *<a href>*).

Jeśli dodajmy klasę *.btn* do linku *<a>* należy dodać dodatkowy zapis *role="button"* – aby czytniki ekranowe wiedziały o co chodzi.



```
<a class="btn btn-primary" href="#" role="button">Link</a>
<button class="btn btn-primary" type="submit">Button</button>
<input class="btn btn-primary" type="button" value="Input">
<input class="btn btn-primary" type="submit" value="Submit">
<input class="btn btn-primary" type="reset" value="Reset">
```

- *.btn-** - pod gwiazdkę należy podstawić: *primary*, *secondary*, *success*, *danger*, *warning*, *info*, *light*, *dark*, *link* (patrz rozdział o kolorach w Bootstrapie). Klasa *.btn-link* zachowuje się jak button, ale wygląda jak zwykły link.
- *.btn-sm* – tworzy mały button.
- *.btn-lg* – tworzy duży button.

Button success small

Button success normall

Button success large

```
<button class="btn btn-success btn-sm" type="submit">Button success small</button>  
<button class="btn btn-success" type="submit">Button success normall</button>  
<button class="btn btn-success btn-lg" type="submit">Button success large</button>
```

- *.btn-outline-** - przycisk bez wypełnienia. Jedynie kolor i napis ma w wybranym kolorze. Pod gwiazdkę wstawiamy: *primary, secondary, success, danger, warning, info, light, dark, link*. Kolor pojawia się po najechaniu myszką na przycisk.

Primary

Secondary

Success

Danger

Warning

Info

Light

Dark

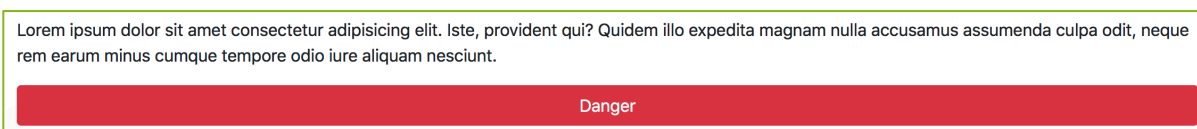
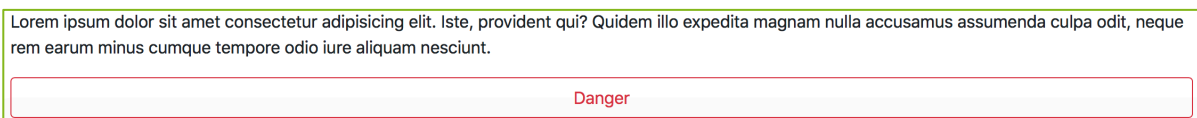
```
<button type="button" class="btn btn-outline-primary">Primary</button>  
<button type="button" class="btn btn-outline-secondary">Secondary</button>  
<button type="button" class="btn btn-outline-success">Success</button>  
<button type="button" class="btn btn-outline-danger">Danger</button>  
<button type="button" class="btn btn-outline-warning">Warning</button>  
<button type="button" class="btn btn-outline-info">Info</button>  
<button type="button" class="btn btn-outline-light">Light</button>  
<button type="button" class="btn btn-outline-dark">Dark</button>
```

- *.btn-block* - to przycisk, który ma 100% szerokości.
- *.btn-disabled* - wygląd ma przycisku, ale jest nieklikalny, nieaktywny.
- *.active* - to przycisk, który wygląda jakby był wciśnięty. W przypadku użycia tej klasy, należy dodać atrybut dla czytników ekranowych *aria-pressed="true"*.

Zadanie 12

Stwórz `<div>` z klasą `.container`. Wewnątrz niej umieść w tagu `<p>` fragment `lorem ipsum`. Następnie dodaj `<button>`, który:

- Nie będzie miał wypełnienia, będzie miał czerwoną obramówkę i po najechaniu jego kolor będzie czerwony.
- Będzie rozciągnięty na całą szerokość strony



Kod do zadania 12

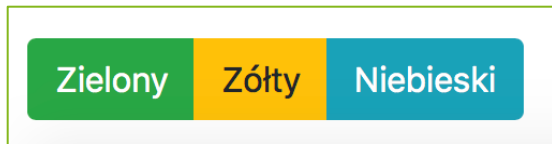
```
<div class="container">
  <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Iste, provident qui? Quidem illo expedita magnam nulla accusamus assumenda culpa odit, neque rem earum minus cumque tempore odio iure aliquam nesciunt.</p>
  <button type="button" class="btn btn-outline-danger btn-block">Danger</button>
</div>
```

Grupowanie przycisków

Istnieje możliwość grupowania przycisków w jedną całość, ale tak, aby każdy z nich był klikalny. Służy do tego klasa `.btn-group`.

Aby stworzyć grupę przycisków należy stworzyć `<div>` o klasie `.btn-group`, atrybucie `role="group"` oraz o `aria-label` z opisem dla czytników czyli np. `aria-label="button group"`.

Wewnątrz `<div>` należy umieścić kilka przycisków.

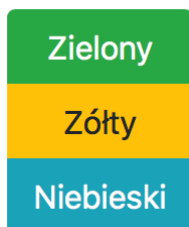


```
<div class="btn-group" role="group" aria-label="button group">
  <button type="button" class="btn btn-success">Zielony</button>
  <button type="button" class="btn btn-warning">Zółty</button>
  <button type="button" class="btn btn-info">Niebieski</button>
</div>
```

Przy grupowaniu przycisków możemy skorzystać z klas określających ich wielkość - *.btn-group-lg* – dla przycisków dużych oraz *.btn-group-sm* – dla przycisków małych.

Przyciski grupowe możemy też ułożyć pionowo. Jest to bardzo przydatna funkcja przy małych rozdzielczościach.

Aby ułożyć przyciski pionowo wystarczy zamiast klasy *.btn-group* użyć klasy *.btn-group-vertical*.



```
<div class="btn-group-vertical" role="group" aria-label="button group">
  <button type="button" class="btn btn-success">Zielony</button>
  <button type="button" class="btn btn-warning">Zółty</button>
  <button type="button" class="btn btn-info">Niebieski</button>
</div>
```

Pasek narzędzi z przycisków

Grupowanie przycisków możemy wykorzystać do zbudowania paska narzędzi na stronie.

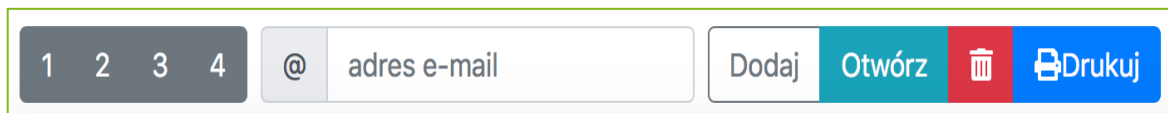
Aby stworzyć pasek narzędzi należy stworzyć `<div>`, który będzie kontenerem dla grupy przycisków. Następnie należy do niego dodać klasę `.btn-toolbar`, `role="toolbar"` oraz informacje dla czytników np. `aria-label="Toolbar with button groups"`.

Następnie tworzymy grupę przycisków, poprzez dodanie klasy `.btn-group`, `role="group"` oraz informacji dla czytników `aria-label`.

Do przycisków możemy oprócz tekstu dodawać np. ikony z *Font Awesome*.

Aby to zrobić wystarczy podpiąć link do serwera *CDN Font Awesome* w sekcji `<head>` przed linkiem do naszego pliku ze stylami *CSS*, i w odpowiednie miejsce wrzucić tag `<i>` z *Font Awesome*.

Aby oddzielić grupy przycisków od siebie, należy do klasy `.btn-group` dodać klasę `.mr-2`.



```
<div class="container">
  <div class="btn-toolbar mb-3" role="toolbar" aria-label="Toolbar with button groups">
    <div class="btn-group mr-2" role="group" aria-label="First group">
      <button type="button" class="btn btn-secondary">1</button>
      <button type="button" class="btn btn-secondary">2</button>
      <button type="button" class="btn btn-secondary">3</button>
      <button type="button" class="btn btn-secondary">4</button>
    </div>
    <div class="input-group mr-2">
      <div class="input-group-prepend">
        <div class="input-group-text" id="btnGroupAddon">@</div>
      </div>
      <input type="text" class="form-control" placeholder="adres e-mail" aria-label="Input group example" aria-describedby="btnGroupAddon">
    </div>
    <button type="button" class="btn btn-secondary">Dodaj</button>
    <button type="button" class="btn btn-primary">Otwórz</button>
    <button type="button" class="btn btn-danger"><img alt="trash icon" data-bbox="725 575 755 595"/></button>
    <button type="button" class="btn btn-primary"><img alt="print icon" data-bbox="765 575 795 595"/> Drukuj</button>
  </div>
</div>
```

```

</div>

<div class="btn-group mr-2" role="group" aria-label="Second group">
  <button type="button" class="btn btn-outline-secondary">Dodaj</button>
  <button type="button" class="btn btn-info">Otwórz</button>
  <button type="button" class="btn btn-danger"><i class="fas fa-trash-alt"></i> </button>
  <button type="button" class="btn btn-primary"><i class="fas fa-print"></i>Drukuj</button>
</div>
</div>
</div>

```

Buttons z rozwijaną listą

Możemy stworzyć też bardzo fajną rzecz. Mianowicie `<button>` jako rozwijana lista linków. Po kliknięciu na `<button>` pojawi się rozwijana lista.

Aby stworzyć taki `<button>` do sekcji `<body>` między linkiem do *jQuery* a linkiem do skryptu *bootstrap.min.js* należy wkleić link do jeszcze jednego skryptu:

```

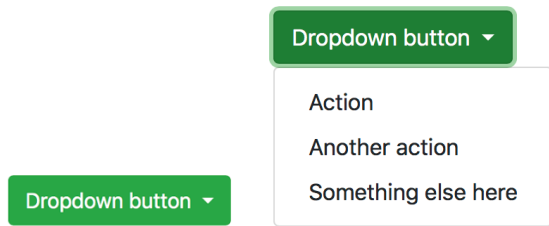
<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.9/umd/popper.min.js" integrity="sha384-
ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPskvXusvfa0b4Q"
crossorigin="anonymous"></script>

```

Ten skrypt możemy mieć podpięty przez cały czas. Jest on potrzebny do działania kilku komponentów w Bootstrapie.

Następnie należy stworzyć kontener, wewnątrz którego umieszczamy `<div>` o klasie `.dropdown`. Wewnątrz tego `<div>` tworzymy `<button>` o klasach `.btn` oraz `.dropdown-toggle`. Możemy też dodać klasę odpowiadającą za wygląd naszego buttonu np. `.btn-success`. Dodajemy `type="button"`, `id="dropdownMenuButton"`, `data-toggle="dropdown"`, `aria-haspopup="true"` oraz `aria-expanded="false"`.

Następnie tworzymy `<div>` z klasą `.dropdown-menu` oraz `aria-labelledby="dropdownMenuButton"`. Na koniec tworzymy linki, które mają pokazywać się po rozwinięciu listy, za pomocą tagu `<a href>` z klasą `.dropdown-item`.



```
<div class="container">
  <div class="dropdown">
    <button class="btn btn-success dropdown-toggle" type="button" id="dropdownMenuButton" data-
toggle="dropdown" aria-haspopup="true" aria-expanded="false">
      Dropdown button
    </button>
    <div class="dropdown-menu" aria-labelledby="dropdownMenuButton">
      <a class="dropdown-item" href="#">Action</a>
      <a class="dropdown-item" href="#">Another action</a>
      <a class="dropdown-item" href="#">Something else here</a>
    </div>
  </div>
</div>
```