



Front End 4 Beginners

Omówienie pierwszej pracy domowej

Złe wcięcia w kodzie

- Niepoprawne wcięcia sprawiają, że kod staje się nieczytelny oraz trudniej znaleźć w nim błędy

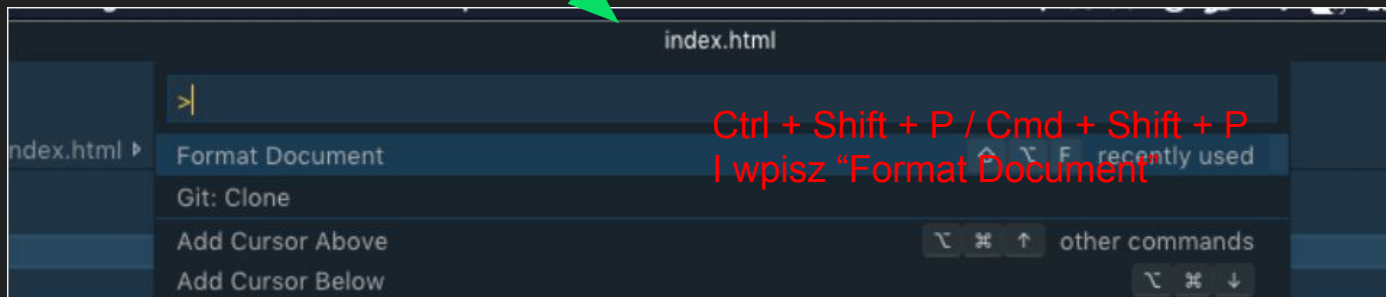
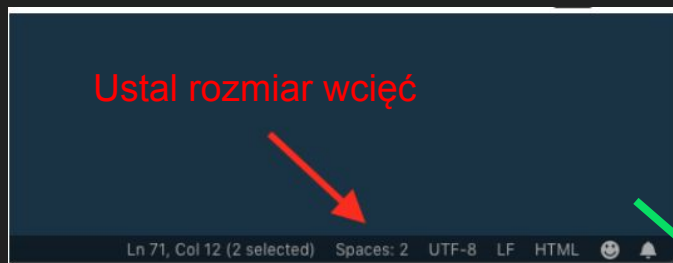
```
1      <form action="" method= "GET">
2      <label for="name">EMAIL</label>
3      <input id="mail"/>
4      <br/>
5      <label for="name">Topic</label>
6      <input id="topic"/>
7      <br/>
8      <br/>
9      <button type="submit">CONTACT US!</button>
10     </form>
```



```
1      <form action="" method= " GET">
2      <label for="name">EMAIL</label>
3      <input id="mail" />
4      <br />
5      <label for="name">Topic</label>
6      <input id="topic" />
7      <br />
8      <br />
9      <button type="submit">CONTACT US!</button>
10     </form>
```

Złe wcięcia w kodzie

- Jak łatwo poprawić wcięcia w VS Code?



Złe wcięcia w kodzie

- Automatyczne formatowanie pozwoli również wykryć błędy w składni

```
<span>
   </span> <span>
   </span> <span>
   </span> <span>
   </span> <p>KAMIL ZGONDEK</p>
  <p>Keys</p>

  <div>
     </div> <div>
     </div> <div>
```

Nadmiarowe użycie znaczników semantycznych

- Znaczniki semantyczne powinny być używane jedynie tam, gdzie są faktycznie potrzebne, np. Znacznik `<nav>` powinien obejmować jedynie panele nawigacyjne

```
<nav class="sidebar">
  <nav id="sidebar-logo" class="sidebar_section">
    
    <h1 class="page-heading">Strona oficjalna zespołu 6BM</h1> <!-- to be hidden in CSS -->
  </nav>
  <nav id="sidebar-menu" class="sidebar_section">
    <ul class="menu">
      <li class="menu__item"><a href="#section-home">Home</a></li>
      <li class="menu__item"><a href="#section-about">About</a></li>
      <li class="menu__item"><a href="#section-concerts">Concerts</a></li>
      <li class="menu__item"><a href="#section-video">Video</a></li>
      <li class="menu__item"><a href="#section-photos">Photos</a></li>
      <li class="menu__item"><a href="#section-contact">Contact</a></li>
    </ul>
  </nav>
  <nav id="sidebar-social_media" class="sidebar_section">

    <div id="social_media-fb" class="sidebar_social_media_item">
      <a href="#section-social-media">
    <div id="social_media-insta" class="sidebar_social_media_item"></div>
      <a href="#section-social-media">
    <div id="social_media-last" class="sidebar_social_media_item"></div>
      <a href="#section-social-media">
    <div id="social_media-yout" class="sidebar_social_media_item"></div>
      <a href="#section-social-media">
  </nav>
</nav>
```

Niepoprawne użycie BEM

- Nie należy zagnieżdżać elementów BEMowych, w poniższym przypadku pozycje menu powinny mieć klasę tego samego rzędu co obejmujący je <div>, gdyż wszystkie są elementami bloku “navbar”

Wtedy byśmy mieli np. “navbar__sections” i “navbar__sections-row”

```
<nav class='navbar'>
  <div class='navbar__link-logo-container'>
    <a href='#home'><img src='./images/logo.png' alt='Logo zespołu'></a>
  </div>
  <div class='navbar__sections'>
    <div class='navbar__sections__row'><a href='#home' class='navbar__sections__row__link'>HOME</a></div>
    <div class='navbar__sections__row'><a href='#about' class='navbar__sections__row__link'>ABOUT</a></div>
    <div class='navbar__sections__row'><a href='#concerts' class='navbar__sections__row__link'>CONCERTS</a></div>
    <div class='navbar__sections__row'><a href='#video' class='navbar__sections__row__link'>VIDEO</a></div>
    <div class='navbar__sections__row'><a href='#photos' class='navbar__sections__row__link'>PHOTOS</a></div>
    <div class='navbar__sections__row'><a href='#contact' class='navbar__sections__row__link'>CONTACT</a></div>
  </div>
```

Niepoprawne użycie BEM

- Jeśli chcemy nazwać głęboko zagnieżdżone elementy, musimy się zastanowić, czy nie powinniśmy z nich uczynić osobnych komponentów.

Nazwa “page-content” sugeruje, że elementy tego komponentu dotyczą ogólnego kontenera na treść strony i element “page-content__band-member” wydaje się być już za bardzo specyficzny. Można z niego uczynić nowy komponent “band-member” i przypisać mu elementy “band-member__picture”, “band-member__details”, “band-member__name” i “band-member__instrument”

```
<main class="page-content">
  <section id="section-home" class="page-content__section">
    <div id="section-home-band-member01" class="page-content__band-member">
      <div id="band-member01-picture" class="band-member__picture">
        
      </div>
      <div id="band-member01-details" class="band-member__details">
        <p class="band-member-name">Kamil Zgondek</p>
        <p class="band-member-instrument">Keys</p>
      </div>
    </div>
  </div>
```

Użycie pojedynczych cudzysłówów zamiast podwójnych

- Użycie pojedynczych cudzysłówów w opisie atrybutów może spowoduje problemy, gdy zechcemy w nich umieścić zwroty w języku angielskim, np.:

```
<img src='./error.png' alt='That's an error!' />
```

W takiej sytuacji atrybut “alt” zakończy się po słowie “That”, a dalsza część stwierdzenia będzie traktowana jako błąd składni.

```
<nav class='navbar'>
  <div class='navbar__link-logo-container'>
    <a href='#home'><img src='./images/logo.png' alt='Logo zespołu' /></a>
  </div>
  <div class='navbar__sections'>
    <div class='navbar__sections__row'><a href='#home' class='navbar__sections__row__link'>HOME</a></div>
    <div class='navbar__sections__row'><a href='#about' class='navbar__sections__row__link'>ABOUT</a></div>
    <div class='navbar__sections__row'><a href='#concerts' class='navbar__sections__row__link'>CONCERTS</a></div>
    <div class='navbar__sections__row'><a href='#video' class='navbar__sections__row__link'>VIDEO</a></div>
    <div class='navbar__sections__row'><a href='#photos' class='navbar__sections__row__link'>PHOTOS</a></div>
    <div class='navbar__sections__row'><a href='#contact' class='navbar__sections__row__link'>CONTACT</a></div>
  </div>
```

Więcej: https://developer.mozilla.org/en-US/docs/Learn/HTML/Introduction_to_HTML/Getting_started#Single_or_double_quotes

Nadmierne użycie nagłówków

- Nagłówki z założenia mają opisywać większe sekcje strony, dlatego nie należy przesadzać z ich użyciem
- W poniższym przykładzie wystarczyło by użycie znacznika `<p>`

```
ion id= concerts >  
  <h1>Concerts</h1>  
  <div>  
      
    <h2>20.12.2018</h2>  
    <h3>Lublin, klub XXX</h3>  
    <button>BUY TICKET</button>  
  </div>
```

Nadmierne użycie atrybutów “id”

- Identyfikatorów używamy do
 - Ustalania kotwic dla nawigacji, np.:
`Link do sekcji`
`<section id="sekcja-strony">Sekcja docelowa</section>`
 - Parowania inputów z labelkami
`<label for="form-field">Opis pola formularza</label>`
`<input type="text" id="form-field" />`
 - Zaawansowanych operacji z użyciem JavaScriptu
 - Jeżeli programując funkcjonalność w JavaScriptcie musimy się odnieść do elementu strony, możemy użyć identyfikator (ale nie zawsze!)

```
<section id="main-page">
  
  <nav id="page-navigation">
    <ul id="page-sections">
      <li id="section-main">HOME</li>
      <li id="section-about">ABOUT</li>
      <li id="section-concerts">CONCERTS</li>
      <li id="section-photos">PHOTOS</li>
      <li id="section-contact">CONTACT</li>
```

```
<article id="About">
  <h2>About</h2>
  
  <section id="description">
    <h3>Straight From Lublin</h3>
    <p>6BM is a young band which has its origins
```

Użycie (lub) kiedy kontekst tego nie wymaga

- Znaczników pogrubienia nie stosujemy do wizualnego pogrubienia tekstu (do tego jest CSS!)
- Znaczniki i służą do podkreślenia ważności danej sekcji tekstu

Więcej: <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/strong>

```
<div>
  
  <p><b>My Caption here</b></p>
  keys
</div>
```

Nieprawidłowe użycie znacznika <a>

- Znacznik <a> służy do tworzenia linków
- Żeby działał poprawnie, potrzebuje atrybutu "href", który zawiera miejsce docelowe, w które poprowadzi nas kliknięcie
- Wewnątrz znacznika <a> musimy umieścić treść (tekst, obrazek, itp.), w którą będzie można kliknąć, np.:
Kliknij mnie!

```
<a name="ABOUT">
  <div id='ABOUT'>
    <h2>ABOUT</h2>
  </div>
```

Linki nie są elementami blokowymi, nie umieszczamy w nich rozbudowanych struktur

```
title1"><a NAME="about">About</h1>
title2">STRAIGHT FROM LUBLIN</div>
a. hndv1">
```

Umieszczając treść w linkach musimy je domykać za pomocą

```
<li class="menu_content">
  <a href="section_about">Home</a>
</li>
<li class="menu_content">
```

Upewniamy się, że adres linka jest poprawny (żeby odnieść się do elementu z atrybutem ID, w adresie stosujemy przedrostek #, np. "#section_about")

Nadmierne informacje w linku

- Gdy chcemy linkować do identyfikatora na stronie, nie ma potrzeby dopisywania nazwy pliku - nie jest to błąd, ale generuje dodatkową pracę przy ewentualnej zmianie nazwy pliku

```
<ul class="sidemenu">  
  <li><a href="index.html#Home">Home</a></li>  
  <li><a href="index.html#About">About</a></li>  
  <li><a href="index.html#Concerts">Concerts</a></li>
```

Tabele użyte niezgodnie z przeznaczeniem

- Tabele należy stosować wyłącznie w celu prezentowania danych tabelarycznych - stosowanie ich w celu budowania struktury wizualnej strony jest pozostałością po latach 90'tych, kiedy nie posiadano innych narzędzi do tego celu. Drastycznie źle wpływa na dostępność strony!

```
<table class="Galery">
  <tr>
    <td>
      
    </td>
    <td>
      
    </td>
    <td>
      
    </td>
  </tr>
```

Nieprawidłowa implementacja list

- Każda lista musi się składać ze znacznika kontenera (dla list nienumerowanych, dla list numerowanych) i elementów zawierających elementy listy

```
<div class ="social">  
  <li></li>  
  <li></li>  
  <li></li>  
  <li></li>  
</div>
```


Podłączanie <label> do <input>

- Labelki można podłączyć do inputów na dwa sposoby:
 - Za pomocą atrybutu "for" (umieszczamy w nim identyfikator inputu docelowego)
`<label for="input-field">Labelka</label>`
`<input type="text" id="input-field" />`
 - Umieszczając input wewnątrz labelki
`<label>`
Labelka
`<input type="text" />`
`</label>`
- Nie stosujemy obydwu metod na raz!

```
<form action="" method="GET">  
  <label for="email">Email</label>  
  <input type="text" name="email"/>  
  <br/>
```

```
<label for="mail">Mail:  
  <input id="name" type="text" name="usermail" size="15" maxlength="100"/>  
</label>
```

Umieszczanie prac na GitHub

- Stronę umieszczamy na GitHubie w postaci niespakowanej
- Uploadujemy zarówno kod jak i assety (obrazki, fonty, itp.)

A screenshot of a file upload interface, likely from a web browser. It shows a single file named 'Praca_Domowa01.rar' with a document icon to its left. The entire interface is enclosed in a red rectangular border.

 Praca_Domowa01.rar