**Temat**

Strona o miejscowości, jednostce terytorialnej, krainie historycznej lub geograficznej, z informacjami demograficznymi, historycznymi, itp;

1. Stosowane technologie:
   1. HTML5, CSS3 i JavaScript
   2. Dane w formacie JSON transformowane do kodu HTML poprzez autorskie skrypty JS
   3. Zastosowanie gotowej biblioteki JS Lightbox2
   4. Użycie pluginu XML Tools do formatowania tekstu w Notepad++
2. Konwencja Kodu
   1. Nazwy znaczników i atrybutów HTML – wszystkie z małych liter
   2. Kolejność atrybutów HTML
      1. href
      2. class
      3. id
      4. src
      5. alt
      6. style
   3. Kolejność własności CSS
      1. display
      2. wymiary
      3. position
      4. margin
      5. padding
      6. border
      7. tło
      8. kolor tekstu
      9. dekoracja tekstu
      10. inne atrybuty tekstu
      11. inne atrybuty alfabetycznie
   4. nie łamiemy długich linii, stosujemy zawijanie wierszy w trakcie edycji
   5. sposób wcięć – pojedynczy tabulator
   6. kiedy stosujemy wcięcia
      1. wszędzie zastosowany został pojedynczy tabulator
   7. nie pomijamy domknięć znaczników ani definicji html, head i body
   8. wszystkie atrybuty muszą mieć wartości objęte cudzysłowami podwójnymi:

<div class="Menu">

* 1. łączenie otwarć i zamknięć powiązanych znaczników:
     1. elementy listy otwieramy i zamykamy w osobnych od obejmującej listy liniach

<ol>

<li>

<a href="index.html">Geologia</a>

</li>

<li>

<a href="Lokacja.html">Lokalizacja</a>

</li>

<li>

<a href="galeria.html">Galeria</a>

</li>

<li>

<a href="#" class="NotActive">Kontakt</a>

</li>

</ol>

* 1. w kodzie CSS nie wolno zapisywać krótkich styli we wspólnej linii z selektorem
  2. konwencja nazw własnych
     1. ID elementów:
        1. opisowe
        2. zaczynające się wielką literą
        3. w razie powielenia się nazw, dołączenie numeru sekcji po myślniku
     2. Nazwy klas:
        1. opisowe w języku angielskim
        2. wyrazy łączone myślnikami
        3. pełna nazwa (brak skrótów)
     3. Nazwy JavaScript:
        1. nazwy funkcji i zmiennych zapisujemy za pomocą camelCase
  3. formatowanie definicji funkcji, obiektów JS:
     1. ciało funkcji zawsze zapisujemy w nowej linii

*function myFunction() {  
 pojedyncza instrukcja;  
}*