

Podstawy informatyki

HTML5 / CSS3

Laboratorium 1

Wprowadzenie do HTML5 i struktury dokumentu

Czas trwania: 2 × 45 min**Tryb:** studia stacjonarne**Jednostka:** Katedra Elektrotechniki i Technologii Nadprzewodnikowych**Prowadzący:** dr inż. Michał Łanczont

Cel ćwiczenia

Celem ćwiczenia jest opanowanie poprawnego szkieletu dokumentu HTML, konfiguracji metadanych, hierarchii nagłówków, akapitów i list oraz bezpiecznego osadzania dodatkowego dokumentu przy użyciu `iframe`. Student ma również rozróżniać rozwiązania aktualne od elementów historycznych i przestarzałych.

Wstęp

HTML (HyperText Markup Language) jest językiem znaczników służącym do opisu **struktury i znaczenia treści** dokumentu WWW. Nie jest językiem programowania: sam HTML nie opisuje algorytmów, lecz wskazuje przeglądarce, które fragmenty dokumentu są nagłówkami, akapitami, listami, odsyłaczami, formularzami, tabelami lub osadzonymi zasobami.

Szkielet dokumentu HTML5

Minimalny współczesny dokument HTML powinien posiadać deklarację typu dokumentu, element główny `html`, część metadanych `head` oraz widoczną zawartość w `body`.

```
<!DOCTYPE html>
<html lang="pl">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Podstawy informatyki - Laboratorium 1</title>
</head>
<body>
  <h1>Wprowadzenie do HTML</h1>
  <p>Pierwszy akapit dokumentu.</p>
</body>
</html>
```

Element	Znaczenie
<code><!DOCTYPE html></code>	Deklaruje współczesny tryb HTML i zapobiega uruchomieniu trybu zgodności <i>quirks mode</i> .
<code><html></code>	Element główny dokumentu. Atrybut <code>lang="pl"</code> informuje m.in. czytniki ekranu o języku treści.
<code><head></code>	Metadane, tytuł, odwołania do arkuszy CSS i innych zasobów.
<code><body></code>	Treść renderowana w oknie przeglądarki.
<code><title></code>	Tytuł dokumentu wyświetlany m.in. na karcie przeglądarki i używany jako ważna informacja opisująca stronę.

Metadane w sekcji head

Kodowanie UTF-8 należy deklarować możliwie wcześnie w **head**:

```
<meta charset="UTF-8">
```

Znacznik viewport jest szczególnie istotny na urządzeniach mobilnych:

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

Nie tworzy on sam z siebie responsywnego układu, ale powoduje, że przeglądarka mobilna interpretuje szerokość strony zgodnie z szerokością obszaru wyświetlania.

Opis strony może być zapisany w metadanych:

```
<meta name="description"
      content="Ćwiczenie laboratoryjne z podstaw HTML i organizacji treści.">
```

Wyszukiwarki mogą wykorzystać taki opis jako fragment wyniku wyszukiwania, ale nie ma gwarancji, że zostanie pokazany dosłownie. Historyczny metatag **keywords** można znać, jednak nie jest on wymagany w ćwiczeniu i nie należy traktować go jako współczesnej techniki SEO.

Nagłówki i akapity

Nagłówki **h1–h6** tworzą hierarchię treści. Numer poziomu powinien wynikać z logiki dokumentu, a nie z oczekiwanego rozmiaru tekstu. W prostym dokumencie warto stosować jeden wyraźny nagłówek poziomu 1 opisujący całą stronę, a dalsze poziomy rozwijać bez nieuzasadnionego pomijania poziomów.

```
<h1>Dokumentacja projektu</h1>
<p>Krótki opis dokumentu.</p>

<h2>Założenia</h2>
<p>Opis założeń.</p>

<h3>Wymagania sprzętowe</h3>
<p>Opis wymagań.</p>
```

Listy uporządkowane i nieuporządkowane

Lista nieuporządkowana **ul** prezentuje elementy, dla których kolejność nie jest najważniejsza. Lista uporządkowana **ol** opisuje sekwencję, procedurę lub ranking. Elementy obu list zapisuje się w **li**.

```
<ul>
  <li>HTML</li>
  <li>CSS</li>
  <li>Walidacja</li>
</ul>

<ol>
  <li>Utwórz dokument.</li>
  <li>Dodaj treść.</li>
  <li>Sprawdź poprawność.</li>
</ol>
```

Dla **ol** poprawne są m.in. atrybuty **start**, **reversed** i **type**. Atrybut **start** zawsze przyjmuje liczbę całkowitą, nawet jeżeli lista jest wyświetlana literami lub liczbami rzymskimi.

```
<ol start="4" type="I">
  <li>Etap czwarty</li>
  <li>Etap piaty</li>
</ol>

<ol reversed>
  <li>Krok koncowy</li>
  <li>Krok posredni</li>
  <li>Krok poczatkowy</li>
</ol>
```

Listy wielopoziomowe tworzy się przez zagnieżdżenie kompletnej listy w elemencie `li` listy nadrzędnej:

```
<ol>
  <li>Rozdział 1</li>
  <li>Rozdział 2
    <ol>
      <li>Sekcja 2.1</li>
      <li>Sekcja 2.2
        <ul>
          <li>Podpunkt A</li>
          <li>Podpunkt B</li>
        </ul>
      </li>
    </ol>
  </li>
</ol>
```

Ramki historyczne i współczesny `iframe`

Znaczniki `frameset` i `frame` występowały w dawnych wersjach HTML i pozwalały podzielić całe okno przeglądarki na niezależne dokumenty. Są dziś rozwiązaniem przestarzałym i nie należy wykorzystywać ich w nowych projektach. Zostają w instrukcji wyłącznie jako kontekst historyczny.

Współczesny element `iframe` osadza pojedynczy dokument lub zasób wewnątrz normalnego `body`. Typowe atrybuty to `src`, `srcdoc`, `title`, `width`, `height`, `loading`, `sandbox` i `allow`. Wymiary w atrybutach HTML podaje się jako liczby odpowiadające pikselom CSS, np. `width="600"`, a nie `width="600px"`.

```
<iframe
  src="ramka.html"
  title="Dodatkowe informacje do ćwiczenia"
  width="600"
  height="240"
  loading="lazy">
</iframe>
```

Można również stworzyć prostą ramkę bez dodatkowego pliku:

```
<iframe
  title="Przykład dokumentu osadzonego"
  srcdoc="<h2>Dokument osadzony</h2><p>Przykładowa tresc.</p>">
</iframe>
```

Ważne ograniczenie iframe

Nie każda obca witryna pozwala na osadzenie jej w ramce. Serwer może zablokować takie użycie nagłówkiem `X-Frame-Options` lub polityką CSP `frame-ancestors`. Dlatego do ćwiczeń należy preferować plik lokalny lub `srcdoc`, zamiast zakładać, że dowolna strona zewnętrzna zadziała.

Przykład kompletny – dokument z listą i ramką

Poniższy przykład łączy najważniejsze elementy instrukcji. Zwróć uwagę na kolejność metadanych, hierarchię nagłówków, poprawne zagnieżdżenie listy oraz tytuł ramki.

```
<!DOCTYPE html>
<html lang="pl">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta name="description" content="Krotka dokumentacja projektu pomiarowego.">
  <title>Dokumentacja projektu</title>
</head>
<body>
  <h1>Projekt pomiarowy</h1>
  <p>Strona opisuje kolejne etapy przygotowania stanowiska.</p>

  <h2>Etapy</h2>
  <ol start="1">
    <li>Przygotowanie sprzętu</li>
    <li>Konfiguracja
      <ul>
        <li>ustawienie interfejsu</li>
        <li>sprawdzenie połączeń</li>
      </ul>
    </li>
    <li>Pomiar</li>
  </ol>

  <h2>Informacje dodatkowe</h2>
  <iframe src="ramka.html" title="Informacje dodatkowe" width="500" height="180"></
  iframe>
</body>
</html>
```

Co należy zauważyć

- lista zagnieżdżona znajduje się *wewnątrz* elementu `li`, a nie obok niego,
- `iframe` ma opisowy `title`,
- metadane nie zastępują treści widocznej w dokumencie,
- poziomy nagłówków opisują strukturę, a nie rozmiar czcionki.

Typowe błędy

- zapis `<meta charset="UTF8">` zamiast standardowego `UTF-8`,
- umieszczanie widocznej treści w `head`,
- zamykanie listy przed jej zagnieżdżonym podpoziomem,
- próba osadzania witryny, która blokuje `iframe`, i uznawanie tego za błąd HTML,

- stosowanie historycznego `frameset` w nowym dokumencie.

Przebieg ćwiczenia

Zadanie 1. Dokumentacja osobista i metadane

Do wykonania

Utwórz plik `dokumentacja.html`. Zaimplementuj kompletny szkielet HTML, ustaw `lang="pl"`, kodowanie UTF-8, viewport i tytuł `Moje informacje osobiste`. W body umieść `h1` z imieniem i nazwiskiem albo tytułem przykładowego dokumentu.

Zadanie 2. Struktura artykułu

Do wykonania

Dodaj metadane `description`. Pod nagłówkiem głównym umieść akapit wstępu, a następnie co najmniej dwie logiczne części opisane nagłówkami `h2`. W jednej z nich utwórz podrozdział `h3`. Pod każdym nagłówkiem dodaj minimum trzy zdania treści i zastosuj polskie znaki.

Zadanie 3. Listy ustrukturyzowane

Do wykonania

Dodaj listę `ul` z trzema elementami oraz listę `ol` z co najmniej czterema krokami procedury. W drugim elemencie `ol` zagnieźdź listę `ul` z dwoma elementami. Następnie sprawdź osobno działanie `start="10"`, `reversed` i `type="I"`.

Zadanie 4. Porównanie HTML i CSS w numerowaniu list

Do wykonania

Zapisz krótką notatkę w komentarzu HTML: które cechy list opisują strukturę i kolejność, a które dotyczą tylko wyglądu. Nie stosuj przestarzałego atrybutu `type` dla `ul`; jego punktory będą później kontrolowane przez CSS.

Zadanie 5. Osadzanie lokalnego dokumentu

Do wykonania

Utwórz plik `ramka.html` z nagłówkiem i dwoma akapitami. Osadź go w `dokumentacja.html` przez `iframe`. Ustaw `title`, `width` i `height`. Następnie przygotuj drugi `iframe` korzystający z `srcdoc`. Porównaj oba sposoby.

Zadanie 6. Analiza błędnej ścieżki i blokady osadzania

Do wykonania

Celowo podaj nieistniejącą ścieżkę w `src` i zaobserwuj zachowanie ramki. Następnie spróbuj osadzić wybraną publiczną stronę WWW. Jeżeli przeglądarka odmówi wyświetlenia, sprawdź w DevTools komunikat konsoli i wyjaśnij, że brak obrazu nie musi oznaczać błędu składni HTML.

Zadanie 7. Walidacja dokumentu

Do wykonania

Sprawdź pliki w walidatorze HTML. Usuń błędy strukturalne i błędne zagnieżdżenia. W kodzie zachowaj czytelne wcięcia. Nie twórz dokumentu z `frameset` i nie wymagaj `meta keywords`; oba zagadnienia zostały omówione tylko jako informacja historyczna.

Zadanie końcowe

Integracja

Przygotuj stronę informacyjną o wybranym zagadnieniu technicznym. Dokument ma zawierać kompletny `head`, logiczną hierarchię `h1–h3`, akapity, listę wielopoziomową, listę uporządkowaną z co najmniej jednym atrybutem sterującym numerowaniem oraz działający lokalny `iframe`. Kod ma przejść walidację.

Kryteria zaliczenia

- poprawny szkielet HTML i UTF-8,
- logiczna hierarchia nagłówków i list,
- brak `frameset/frame` oraz zbędnego `meta keywords`,
- działający lokalny `iframe` z atrybutem `title`,
- poprawna walidacja dokumentu.

Sprawozdanie

Do odbioru należy przygotować pliki źródłowe utworzone podczas ćwiczenia. Jeżeli prowadzący wymaga sprawozdania, powinno ono zawierać: krótki opis rozwiązania, najważniejsze fragmenty kodu, zrzut ekranu wyniku oraz wnioski dotyczące napotkanych problemów. Pliki HTML i CSS należy zachować w strukturze katalogów zgodnej z instrukcją.

Materiały pomocnicze

- HTML Living Standard: <https://html.spec.whatwg.org/>
- MDN Web Docs – HTML: <https://developer.mozilla.org/docs/Web/HTML>
- MDN Web Docs – CSS: <https://developer.mozilla.org/docs/Web/CSS>
- Walidator HTML: <https://validator.w3.org/>