

Podstawy informatyki

HTML5 / CSS3

Laboratorium 10

Semantyczna struktura HTML5 i kontenery

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

Celem jest świadome stosowanie `header`, `nav`, `main`, `section`, `article`, `aside`, `footer` oraz rozróżnienie ich od neutralnych `div` i `span`. Student ma budować strukturę wynikającą ze znaczenia treści, a nie z oczekiwanego wyglądu.

Wstęp

Semantyczny HTML opisuje rolę fragmentu treści, a nie tylko jego wygląd. Przeglądarki, wyszukiwarki i technologie asystujące mogą wykorzystać tę informację do lepszego zrozumienia dokumentu.

Główne elementy strukturalne

Element	Typowe zastosowanie
<code>header</code>	treść wprowadzająca dla strony lub sekcji; logo, tytuł, informacje wstępne
<code>nav</code>	ważny blok nawigacji
<code>main</code>	dominująca, unikalna treść dokumentu; dokument nie powinien mieć więcej niż jednego nieukrytego elementu <code>main</code>
<code>section</code>	tematyczna część dokumentu, zwykle z własnym nagłówkiem
<code>article</code>	samodzielna treść mająca sens również poza bieżącym kontekstem
<code>aside</code>	treść poboczna, luźniej związana z głównym nurtem
<code>footer</code>	informacje końcowe dla strony lub sekcji

```
<body>
  <a class="skip-link" href="#main-content">Przejdź do treści</a>

  <header>
    <h1>Nazwa serwisu</h1>
  </header>

  <nav>
    <ul>
      <li><a href="index.html">Start</a></li>
      <li><a href="uslugi.html">Usługi</a></li>
    </ul>
  </nav>

  <main id="main-content">
    <article>
      <h2>Aktualność</h2>
      <p>Samodzielna treść artykułu.</p>
    </article>
```

```
<section>
  <h2>Opinie</h2>
  <p>Tematycznie powiazana sekcja.</p>
</section>
</main>

<footer>
  <p>&copy; 2026 Autor</p>
</footer>
</body>
```

Natywna semantyka a ARIA

Element `nav` ma już natywną rolę nawigacji, dlatego `role="navigation"` jest zwykle zbędne. Zasadą jest preferowanie semantycznego elementu HTML zamiast dodawania ARIA do neutralnego elementu, jeżeli istnieje właściwy element natywny.

section a article

`section` grupuje temat w ramach dokumentu. `article` oznacza samodzielną jednostkę, np. wpis, aktualność, komentarz lub kartę produktu mogącą istnieć niezależnie. Nie każdy `div` powinien zostać mechanicznie zamieniony na `section`; semantyka powinna wynikać z roli treści.

div i span

Gdy brak elementu o odpowiednim znaczeniu, używa się neutralnych kontenerów:

```
<div class="karta-produktu">
  <h2>Produkt A</h2>
  <p>Opis produktu...</p>
</div>

<p>
  Kluczowe slowo to
  <span class="wyroznienie">semantyka</span>.
</p>
```

`div` jest neutralnym kontenerem blokowym, a `span` neutralnym kontenerem liniowym.

Skip link

Długi nagłówek lub menu można ominąć z klawiatury przez link prowadzący do `main`:

```
<a class="skip-link" href="#main-content">Przejdź do treści głównej</a>
```

Przykładowy CSS może ukrywać link wizualnie poza fokusem, ale po Tab musi on stać się widoczny.

Bazowy reset

```
*, *::before, *::after {
  box-sizing: border-box;
}

body {
  margin: 0;
}
```

```
img {  
  max-width: 100%;  
  height: auto;  
}
```

Pełne resetowanie wszystkich marginesów nagłówków i list jest decyzją projektową, a nie bezwzględnym wymogiem. W prostym ćwiczeniu wystarczy świadomie usunąć te wartości, które utrudniają planowany layout.

Przykład analizy semantycznej

Rozważ trzy fragmenty:

```
<div class="aktualnosc">...</div>  
<section class="aktualnosc">...</section>  
<article class="aktualnosc">...</article>
```

Nie istnieje jedna poprawna odpowiedź wynikająca z klasy CSS. Jeśli fragment jest samodzielną aktualnością, może pasować `article`. Jeśli jest tematycznym rozdziałem bieżącej strony, pasuje `section`. Jeżeli kontener służy tylko układowi i nie ma własnej roli, właściwy może być `div`.

Typowe błędy semantyki

- zamiana każdego `div` na `section` bez względu na znaczenie,
- używanie kilku widocznych `main`,
- nadawanie `role="navigation"` elementowi `nav` bez potrzeby,
- sekcje pozbawione sensownego nagłówka tam, gdzie powinny mieć własny temat,
- hierarchia nagłówków wybierana według wielkości tekstu zamiast struktury,
- brak mechanizmu ominięcia długiego menu klawiaturą.

Przebieg ćwiczenia

Zadanie 1. Szkielet semantyczny

Do wykonania

Utwórz `struktura-semantyczna.html` zawierający `header`, `nav`, jeden widoczny `main` i `footer`. W `nav` umieść listę odsyłaczy. Nie dodawaj redundantnego `role="navigation"`.

Zadanie 2. Section i article

Do wykonania

W `main` utwórz jedną sekcję tematyczną `section` z nagłówkiem i jeden samodzielny `article`. W komentarzu HTML uzasadnij wybór każdego z tych elementów. Dodaj `aside` tylko wtedy, gdy masz rzeczywiście treść poboczną.

Zadanie 3. div i span

Do wykonania

W artykule użyj `div` do neutralnego zgrupowania kilku elementów, gdy nie ma lepszego znacznika semantycznego. W akapicie zastosuj `span` do wyróżnienia pojedynczego słowa. Ostyluj oba kontenery i sprawdź ich domyślne zachowanie `block/inline`.

Zadanie 4. Skip link i nawigacja klawiaturą

Do wykonania

Dodaj link `Przejdź do treści głównej` prowadzący do ID elementu `main`. Ostyluj go tak, by był widoczny po otrzymaniu fokusu. Przetestuj stronę bez myszy.

Zadanie 5. Prosta galeria jako ćwiczenie z div

Do wykonania

W jednej z sekcji utwórz neutralny kontener `div.galeria-zdjec` i trzy `div.miniatura`. Ułóż je Flexboxem, dodaj `gap` i nie nadawaj `div` sztucznej semantyki. Jeśli miniatury reprezentują samodzielne artykuły, zastanów się, czy właściwsze byłyby inne elementy.

Zadanie 6. Walidacja struktury

Do wykonania

Sprawdź hierarchię nagłówków, liczbę nieukrytych `main`, poprawność odsyłaczy oraz walidację HTML. Upewnij się, że żadnego elementu semantycznego nie użyto wyłącznie jako “haka” do CSS.

Zadanie końcowe

Integracja

Przebuduj jedną wcześniejszą stronę na semantyczny dokument z `header/nav/main/section/article/footer`, poprawnymi nagłówkami i skip linkiem. Użyj `div/span` tylko w miejscach neutralnych semantycznie.

Kryteria zaliczenia

- jeden widoczny `main`,
- sensowne rozróżnienie `section` i `article`,
- brak redundantnych ról ARIA dla natywnych elementów,
- rozsądne użycie `div/span`,
- działający skip link.

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>
- Validator HTML: <https://validator.w3.org/>