



Technologie Frontendowe – HTML

By Szumski Szymon

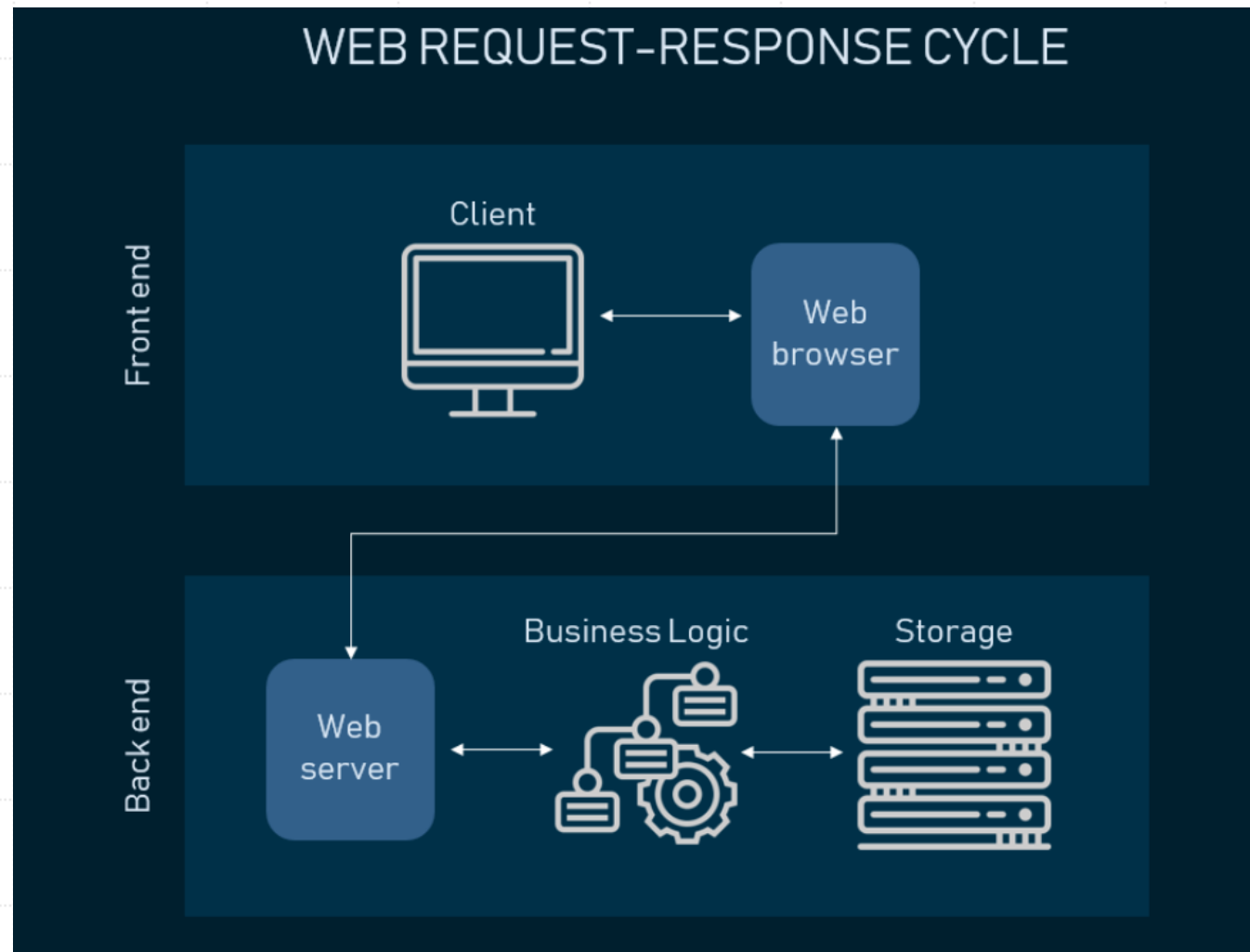


Plan:

1. Czym jest frontend
2. Czym jest HTML
3. Narzędzia deweloperskie
4. Struktura
5. Semantyka
6. Znaczniki
7. Komentarze
8. Atrybuty
9. Elementy blokowe i liniowe
10. Sekcja HEAD
11. Sekcja BODY
12. Tabele
13. Input
14. Formularz

1. Czym jest frontend

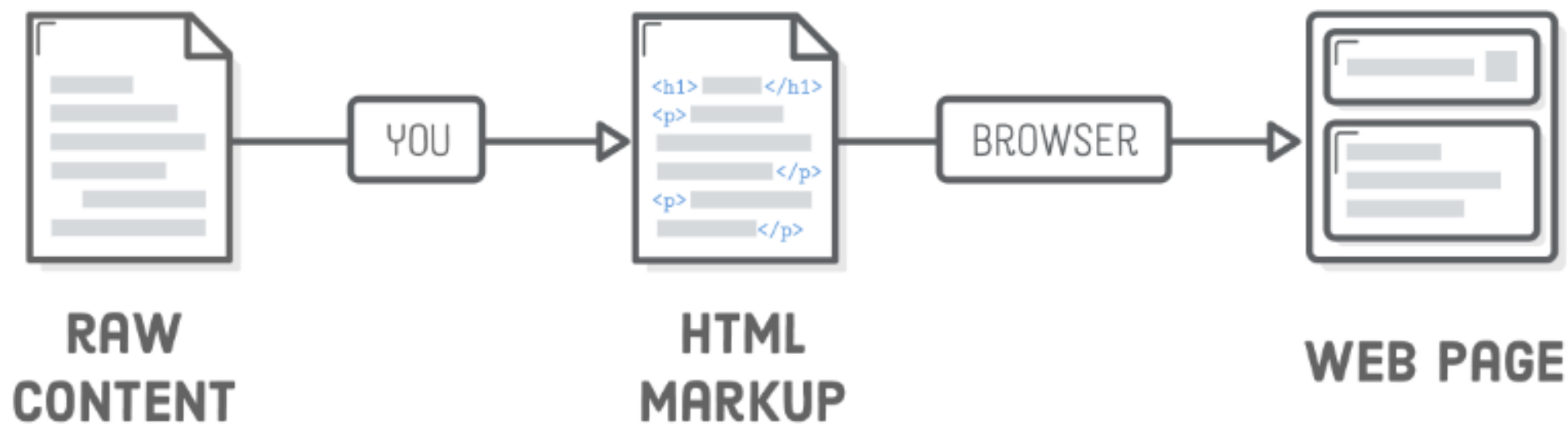
Frontend to część aplikacji lub systemu, która odpowiedzialna jest za interakcję z użytkownikiem. Frontend'em (frontem, front-endem) nazwiemy zatem to, co widzi i czego bezpośrednio używa i doświadcza użytkownik – kolory, style, media, grafiki, menu, przyciski... Można w przybliżeniu powiedzieć, że jest to ta część systemu, która odpowiada za graficzny interfejs użytkownika. Na przestrzeni tego kursu możemy przyjąć, że frontendem będzie ta część, która będzie działać w przeglądarce użytkownika.



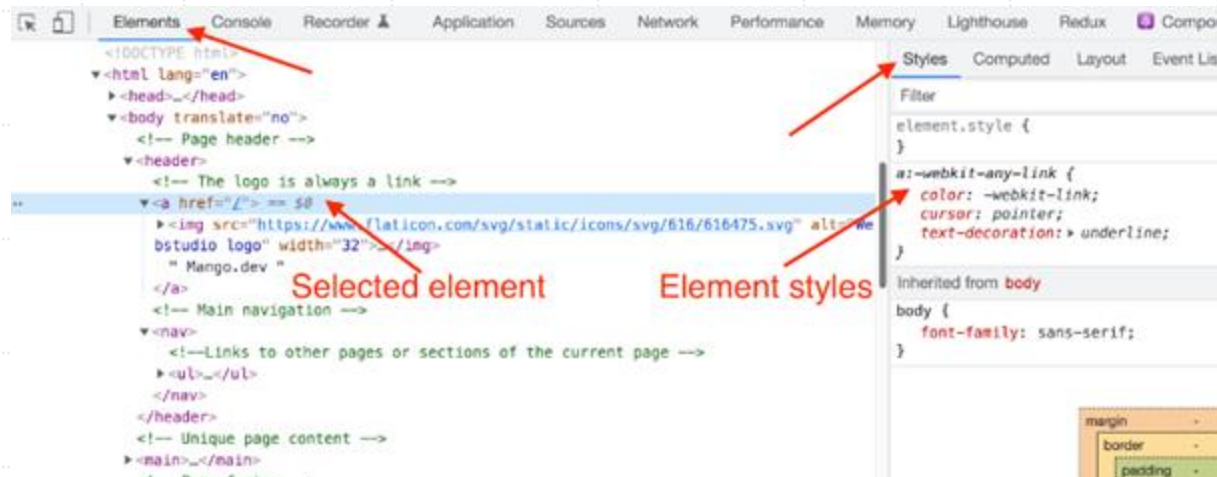
2. Czym jest HTML

HyperText Markup Language (hipertekstowy język znaczników) - język znaczników dokumentów internetowych. Zestaw reguł do strukturyzowania (znaczników) informacji tekstowej, dodawania zdjęć, tworzenia tabel, formularzy, list itp.

Dokument HTML - plik tekstowy z rozszerzeniem .html. Dokument oznaczony za pomocą HTML, interpretowany przez przeglądarkę, dzięki czemu użytkownicy widzą nie kod źródłowy z elementami znaczników, ale ostateczny wynik przetwarzania - stronę internetową.



3. Narzędzia deweloperskie



Dla efektywnego programowania są potrzebne narzędzia. Dlatego nauczymy się pracować z narzędziami deweloperskimi które oferują przeglądarki są one wbudowane we wszystkich nowoczesnych przeglądarkach i nie wymagają dodatkowych instalacji.

Narzędzia deweloperskie - wbudowane rozszerzenie przeglądarki, które umożliwia uzyskanie informacji o kodzie źródłowym strony, stylach, żądaniach sieciowych i wiele więcej.

Narzędzia deweloperskie można wywołać za pomocą następujących metod.

- Naciskając klawisz F12
- Kombinacja klawiszy Ctrl+Shift+I dla Windows
- Kombinacja klawiszy Cmd+Opt+I dla MacOS
- Na stronie internetowej kliknąć prawym przyciskiem myszy i wybrać z menu opcję „Wyświetl kod”

Domyślnie w narzędziach deweloperskich otwiera się karta Elements. Tam jest przedstawiona wewnętrzna struktura HTML dokumentu. Przy tym gdy najedziesz kursorem myszy na jakiś element, zostanie on podświetlony w oknie przeglądarki (viewport).

4. Struktura HTML

Dokument HTML składa się z „drzewa” znaczników. Poniżej przedstawiony jest wymagany minimalny zestaw znaczników, który służy jako podstawa dla dowolnego dokumentu HTML, podobnie do fundamentu dla domu.

<!DOCTYPE > - to nie jest znacznik, ale obowiązkowa deklaracja typu dokumentu. Jest potrzebna, aby poinformować przeglądarkę, w której wersji HTML jest napisany dokument.

Na podstawie DOCTYPE przeglądarka określa wersję HTML i poprawnie wyświetla stronę. Deklaracja typu dokumentu powinna być pierwszą rzeczą, którą przeglądarka widzi podczas przetwarzania dokumentu HTML.

```
<!DOCTYPE html>
<html lang="pl">
  <head>
    <!-- Informacje serwisowe -->
  </head>
  <body>
    <!-- Zawartość -->
  </body>
</html>
```

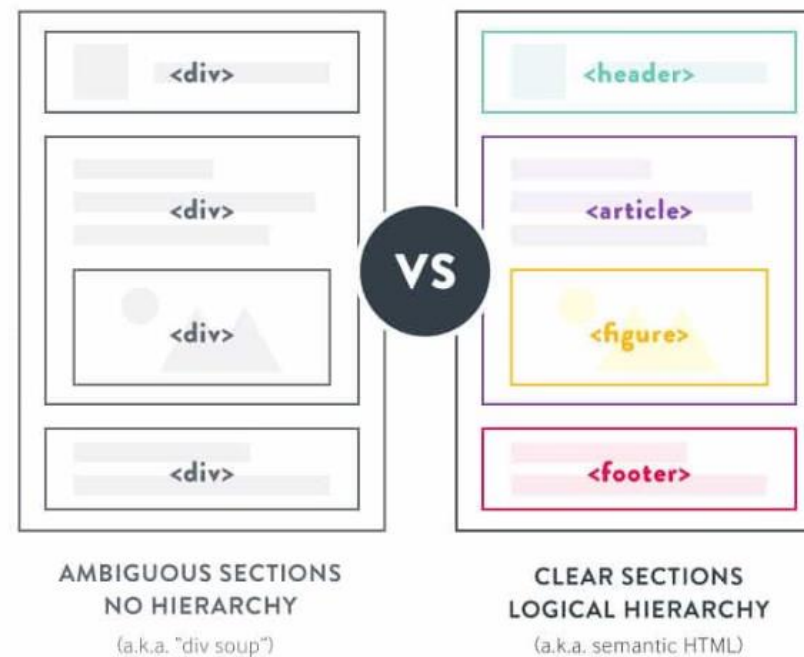
5. Semantyka

Znaczniki przyszłej strony internetowej powinny mieć sens, jak strona gazety lub magazynu. Strona zawiera sekcje, nagłówki, listy, obrazki, akapity tekstu itd. Aby opisać całe bogactwo gatunkowe treści, są odpowiednie znaczniki. Oznacza to, że semantyka jest pierwotnie częścią kodu HTML, ale ona jest bezużyteczna, jeśli jest nieprawidłowo zrealizowana.

Znaczniki semantyczne - podejście do tworzenia znaczników HTML na podstawie używania znaczników zgodnie z ich przeznaczeniem w oparciu o specyfikację oraz odpowiednie typy oznaczanych treści.

W większości przypadków znaczniki semantyczne to po prostu użycie zdrowego rozsądku. Zaznaczając akapit tekstu, używamy znacznika `<p>`, żeby oznaczyć link jest znacznik `<a>`, obraz z treścią jest reprezentowany przez tag `` itp.

Czyli znaczniki semantyczne oznaczają, że znaczniki **nie są wybierane** na podstawie tego, w jaki sposób one są wyświetlane w przeglądarce, one są dobierane na podstawie rodzaju i struktury zaznaczanych treści.



6. Znaczniki

Znacznik (tag) — element hipertekstowego języka znaczników. Jest to najmniejszy element składowy każdej strony internetowej. Każdy tag oznacza jakąś jednostkę: nagłówek, listę, akapit tekstu, obraz.

Aby zaznaczyć znaczniki w tekście dokumentu, stosowane są nawiasy ostre, w których jest określona nazwa znacznika i jego atrybuty.

`<nazwa_znacznika>...</nazwa_znacznika>`

Znacznik otwierający wskazuje, gdzie zaczyna się element, a znacznik zamykający gdzie kończy się. Znacznik zamykający jest tworzony przez dodanie ukośnika (/) przed nazwą znacznika. Zawartość znacznika – treść – znajduje się między otwierającym a zamykającym znacznikiem.

```
<section>Sekcja</section>  
<p>Akapit</p>  
<a>Link</a>  
<button>Przycisk</button>
```


6. Znaczniki (tekstowe)

1. **Akapit** – Znacznik `<p>` - uniwersalny tag do grupowania małych elementów frazowych, oddzielania ich od siebie i dalszej stylizacji. Domyślnie akapit jest elementem blokowym, to znaczy, że zaczyna się w nowej linii, domyślne zachowanie można zmienić za pomocą CSS.
2. **Nagłówek** – Grupa znaczników `<h1>...<h6>` - definiuje nagłówki tekstowe sekcji semantycznych różnych poziomów, które wskazują znaczenie sekcji treści znajdującej się po nich. Jest to narzędzie do strukturyzacji treści tekstowych. Znacznik `<h1>` jest najczęściej używany tylko raz, jako nagłówek strony głównej. Reszta nagłówków może być używana wiele razy ale z uwzględnieniem hierarchii.
3. **Listy** - pozwalają uporządkować kolekcje i zaprezentować je w sposób wizualny i wygodny dla użytkownika. Lista to kontener, którego elementami podrzędnymi mogą być tylko elementy listy, znaczniki ``.
 - Znacznik `` - tworzy listę numerowaną (uporządkowaną), to znaczy każdy element listy jest numerowany. Przeglądarka automatycznie numeruje elementy w kolejności, a jeśli usunąć jeden lub więcej elementów takiej listy, pozostałe numery zostaną automatycznie przeliczone. Numerację elementów można kontrolować za pomocą specjalnych atrybutów listy.
 - Znacznik `` - tworzy listę wypunktowaną (nieuporządkowaną), której każdy element zaczyna się od małego znaku (punktora). Dzięki CSS punktory można usunąć lub zastąpić.

```
<h1>Jak zaparzyć herbatę</h1>
<p>
  Szczegółowe instrukcje dla opornych, po prostu powtórz, a odniesiesz sukces!
</p>

<ol>
  <li>Zagotować wodę</li>
  <li>Wsypać herbatę do kubka</li>
  <li>Wlać wrzątek do kubka</li>
  <li>Czekać 10 minut, po czym można pić</li>
</ol>
```

```
<h1>Najgorętsze kurorty</h1>
<p>Ekspertki zalecają odwiedzenie w tym roku następujących lokalizacji.</p>

<ul>
  <li>Tunezja</li>
  <li>Turcja</li>
  <li>Grecja</li>
  <li>Egipt</li>
</ul>
```



Zadanie 1

- Utwórz główny nagłówek strony (h1)
- Utwórz 3 nagłówki dla sekcji (h2)
- Każda z sekcji powinna zawierać odpowiednio: akapit, liste (ol), liste (ul)

Pamiętamy o przeznaczeniu tagów :)

6. Znaczniki (tekstowe)

Link – Znacznik `<a>` - służy do tworzenia linków, tekstu, po kliknięciu którego przechodzimy do innej strony, ściągamy plik itp. Tekst linku jest wyświetlany w przeglądarce z podkreśleniem, kolor czcionki jest niebieski, po najechaniu na link kursor myszy zmienia widok. Adres linku jest określony w wymaganym atrybucie `href="adres"`. Adres to URL, który może wskazywać stronę, plik, dowolny zasób.

```
<a href="https://google.com">Link do strony głównej Google</a>
```

Atrybut target

Domyślnie link otwiera się w bieżącej zakładce przeglądarki. Atrybut `target="wartość"` określa, w której zakładce ma zostać otwarty dokument, do którego prowadzi link. Jeśli wartość `_blank`, to strona otworzy się na nowej karcie przeglądarki.

Specjalne wartości href

Linki umożliwiają nie tylko przejście do innych stron i pobieranie plików, ale także wykonywanie połączeń telefonicznych, wysyłanie wiadomości czy wykonywanie połączeń przez Skype.

```
<!-- Link do numeru telefonu -->  
<a href="tel:+14251234563">+1 (425) 123-45-63</a>
```

```
<!-- Link do adresu e-mail -->  
<a href="mailto:example@mail.ru">example@mail.ru</a>
```



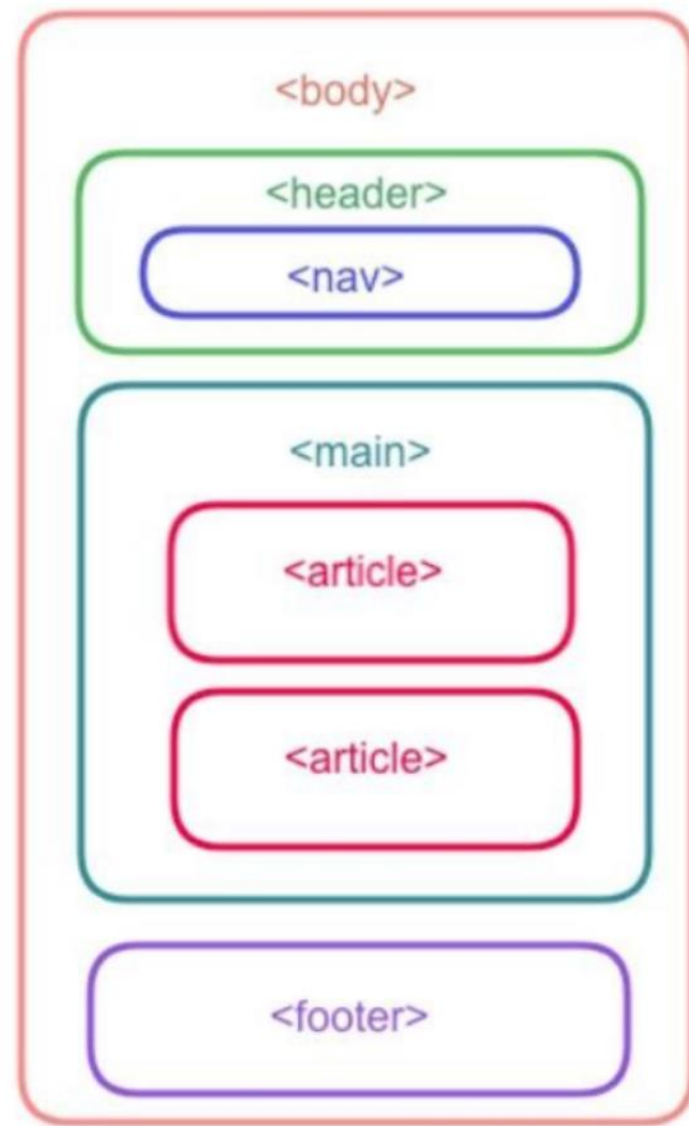
Zadanie 2

Utwórz kolejną sekcję z nagłówkiem odpowiadającej liście linków

- Wybieramy 3 różne typy linków (różne wartości href) – np. Telefon, link, mail

6. Znaczniki (semantyczne)

Wraz z pojawieniem się *HTML5*, do języka zostało dodanych sporo nowych znaczników, w tym tagi **semantyczne**. Ich rolą jest wskazanie typu zawartości, jaką reprezentują. Dzięki temu uzyskujemy czytelniejszy, "samokomentujący się" kod.





Zadanie 3

Utwórz strukturę strony bazując na znacznikach semantycznych

- Zawiera 3 główne sekcje: header, main, footer
- Sekcja main może zawierać dowolną liczbę sekcji, artykułów, bloków itp

6. Znaczniki (obrazki)

Zastosowanie grafiki sprawia, że strony internetowe są bardziej atrakcyjne wizualnie. Obrazki pomagają lepiej przekazać istotę i treść dokumentu. Znacznik `` jest przeznaczony do oznaczania obrazków w różnych formatach graficznych.

- `src="ścieżka"` - obowiązkowy atrybut, który wskazuje adres obrazka. Ścieżka dostępu do obrazka może być bezwzględna lub względna.
- `alt="opis"` - obowiązkowy atrybut, alternatywny opis.
- `width="wartość"` i `height="wartość"` - ustawiają wymiary obrazu w pikselach. Bez zmiany rozmiaru obraz jest wyświetlany na stronie w oryginalnym rozmiarze. Jeśli ustawić tylko jedną wartość, przeglądarka automatycznie obliczy drugą, aby zachować proporcje.

```

```



Zadanie 4

Utwórz liste 3 obrazków

- Linki do grafik pobieramy z google

7. Komentarze

Komentarze służą do tego, żeby pozostawiać objaśnienia, uwagi w kodzie źródłowym, tymczasowo skomentować fragment kodu itp.

```
<!-- To jest komentarz, jego treść nie będzie wyświetlana na stronie -->
```

```
<p>To jest akapit tekstu, on będzie wyświetlany na stronie.</p>
```

```
<!--
```

```
  Komentarz może być wielowierszowy.
```

```
  Jest to wygodne w przypadku bardziej obszernych opisów.
```

```
-->
```



Zadanie 5

Utwórz komentarze dla każdej sekcji na stronie (np. Header)

- Komentarze będą służyć do wydzielenia sekcji i odpowiedzi do czego służy dana sekcja na stronie

8. Atrybuty

Atrybuty to dodatkowe ustawienia znaczników, za pomocą których można zmieniać właściwości i zachowanie elementu. Każdy znacznik ma obowiązkowe i nieobowiązkowe atrybuty, są opcjonalne i może ich być kilka w jednym tagu.

Atrybuty są zapisywane wewnątrz znacznika otwierającego, a ich wartości znajdują się w podwójnych cudzysłowach. Kilka atrybutów oddzielamy spacją.

- `href` - adres strony, na którą przejdzie użytkownik po kliknięciu w tekst linku.
- `target` - wskazuje, w której zakładce strona zostanie otwarta po kliknięciu w w link.
- `title` - dodaje podpowiedź do tekstu linku.

```
<a href="https://google.com" class="link">...</a>
```

```

```

```
<input type="text" name="user_name" />
```

```
<button type="submit">...</button>
```

```
<p class="text">...</p>
```



Zadanie 6

- Do linków z zadania 2 dorzucamy target z wartością "_blank" dla dowolnego linku tak aby strona otwierała się w nowej karcie
- Do każdego znacznika dorzucamy atrybut class – wartość tego atrybutu powinna mówić nam do jakiego elementu należy klasa

9. Elementy blokowe i liniowe

Znaczniki w HTML można podzielić na dwa typy, które wyróżnia ilość miejsca zajmowanego domyślnie na wyświetlanej stronie internetowej.

Elementy liniowe to takie, które zachowują się jak *linie* tekstu. Zajmują tyle miejsca, ile zawarta w nich treść. Ponadto:

- Nie można określić ich wymiarów za pomocą właściwości `width` oraz `height`.
- Nie można określić marginesu górnego oraz dolnego.
- Każdy element liniowy jest ustawiany obok drugiego.

Przykładowe znaczniki liniowe to `<a>` i ``.

Elementy blokowe to takie, które wyświetlają się jak **bloki** na stronie. Domyślnie zajmują **100%** szerokości swojego **rodzica**. Ponadto:

- Można określić ich wymiary za pomocą właściwości `width` oraz `height`.
- Można określić wszystkie (cztery) marginesy.
- Każdy element blokowy umieszczany jest jeden **pod** drugim.

Przykładowe znaczniki blokowe to, np. `<p>`, `<div>` oraz poznane wcześniej znaczniki semantyczne.



Zadanie 7

Utwórz 2 dodatkowe sekcje z nagłówkami (elementy blokowe, elementy liniowe)

- Dodaj tam po kilka przykładowych elementów oraz zobacz jaka będzie różnica na stronie przy wyświetlaniu elementów

10. Sekcja HEAD

```
<head>
  <title>My website</title>
  <meta charset="utf-8">
  <meta content="width=device-width, initial-scale=1.0" name="viewport" />
  <meta name="Description" content="Description of the website" />
  <meta name="Keywords" content="website, technology, course, html" />
  <link rel="stylesheet" href="img/style.css" />
  <script src="js/somescript.js"></script>
</head>
```

- znaczniki w sekcji head nie są interpretowane jako elementy do wyświetlenia w oknie przeglądarki
- zawartość tej sekcji nie jest widoczna dla użytkownika
- dane tu umieszczone potrzebne są samej przeglądarce, by wiedziała, jak wyświetlić podaną treść
- dane z sekcji head odczytywane są też przez inne programy (np. roboty wyszukiwarek internetowych), by właściwie sklasyfikować stronę www

Na powyższym przykładzie widać, jakie elementy mogą znajdować się w sekcji *head*:

- tytuł strony - `title`,
- metadane - `meta` - informacje dla przeglądarki czy wyszukiwarek (sposób wyświetlania, opis strony, słowa kluczowe itp),
- linki do plików ze stylami CSS,
- linki do skryptów JavaScript.

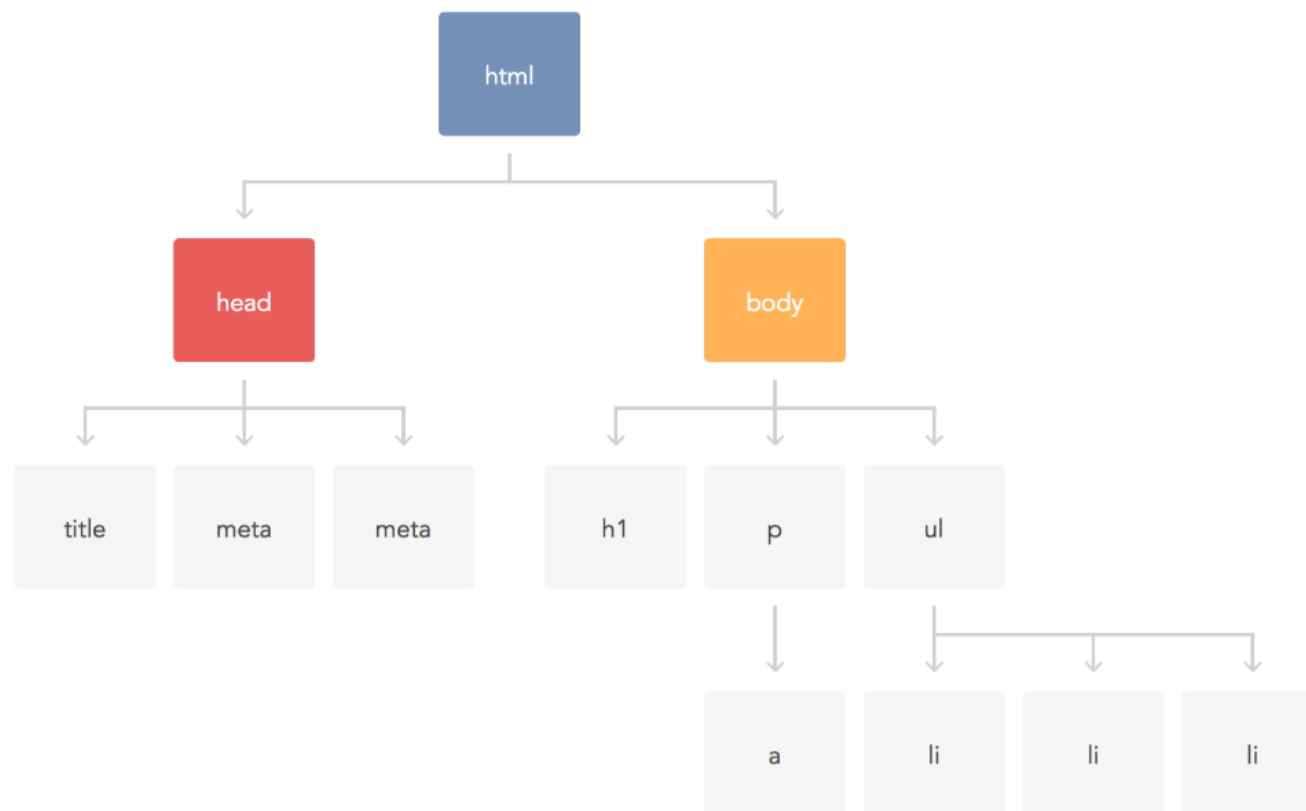


Zadanie 8

W sekcji HEAD zmien tytuł strony oraz nadaj opis i dodaj kilka słów kluczowych za pomocą `<meta name="keywords" />`

11. Sekcja BODY

Zawiera treść przyszłej strony internetowej. Treść, która ma być wyświetlana na stronie, powinna znajdować się wewnątrz tego znacznika.



12. Tabele

Tabele pozwalają wyświetlać złożone relacje, umieszczając dowolną zawartość w komórkach. Należy ich używać tylko do zaznaczania danych tabelarycznych: harmonogram transportu, kalendarz, wyniki meczów, transakcje finansowe, menu w restauracji, cennik itp. To znaczy informacje, które logicznie można przedstawić w formie tabelarycznej, na przykład za pomocą Google Sheets.

Tabela to zbiór wierszy z komórkami. Przyjrzyjmy się podstawowym znacznikom.

- `<table>` - określa całą tabelę, kontener dla zawartości.
- `<tr>` - wiersz tabeli (table row). Nie może być pusty, musi zawierać co najmniej jedną komórkę.
- `<td>` - komórka tabeli (table data). Może być używana tylko wewnątrz wiersza. Każdy wiersz tabeli musi mieć taką samą liczbę komórek.

Komórki nagłówkowe

Aby utworzyć wiersz z nagłówkami kolumn, używamy standardowego znacznika `<tr>` dla wiersza i oznaczamy komórki specjalnym znacznikiem `<th>` (table header) - semantycznym znacznikiem komórki nagłówkowej. Domyślnie tekst wewnątrz `<th>` jest zaznaczony pogrubioną czcionką i jest wyśrodkowany.

```
<table>
  <tr>
    <th>Numer</th>
    <th>Trasa</th>
    <th>Czas podróży</th>
  </tr>
  <tr>
    <td>433</td>
    <td>Kijów - Kowel</td>
    <td>5 godzin</td>
  </tr>
  <tr>
    <td>701</td>
    <td>Charków - Dniepr</td>
    <td>7 godzin</td>
  </tr>
  <tr>
    <td>258</td>
    <td>Lwów - Odessa</td>
    <td>4 godziny</td>
  </tr>
</table>
```

12. Tabele

Sekcje tabeli

Aby ulepszyć semantykę znaczników tabeli, istnieje kilka znaczników strukturalnych.

- `<thead>` - nagłówek, który opakowuje grupę wierszy tabeli w obszarze nagłówka.
- `<tbody>` - ciało, opakowuje główną grupę wierszy tabeli.
- `<tfoot>` - stopka, oznacza grupę wierszy tabeli w obszarze stopki.

```
<table>
  <thead>
    <tr>
      <th>Numer</th>
      <th>Trasa</th>
      <th>Czas podróży</th>
    </tr>
  </thead>

  <tbody>
    <tr>
      <td>433</td>
      <td>Kijów - Kowel</td>
      <td>5 godzin</td>
    </tr>
    <tr>
      <td>701</td>
      <td>Charków - Dniepr</td>
      <td>7 godzin</td>
    </tr>
    <tr>
      <td>258</td>
      <td>Lwów - Odessa</td>
      <td>4 godziny</td>
    </tr>
  </tbody>
</table>
```



Zadanie 9

Utwórz tabele rozkładu jazdy autobusów na 5 dni tygodnia

13. Input

Podstawowym elementem formularza jest input – czyli element umożliwiający wysłanie jednej informacji. W formularzu może być wiele inputów. Rodzaj inputa definiujemy atrybutem *type*. Poniżej lista możliwych typów:

- text – umożliwia wpisanie tekstu
- password – umożliwia wpisanie hasła (tekst z maskowaniem)
- email – umożliwia wpisanie adresu e-mail
- date – umożliwia wybór daty z kalendarza
- checkbox – umożliwia zaznaczenie (lub odznaczenie) opcji do wyboru
- radio – umożliwia wybór jednej z wielu opcji
- submit – wysyła formularz
- hidden – ukryty input (nie widać go, ale przechowuje wartość)
- color – umożliwia wybór koloru

```
<form>
  <div>
    <input type="text" name="your_name" />
  </div>
  <div>
    <input type="date" name="start_date" />
  </div>
  <div>
    <input type="email" name="your_email" />
  </div>
  <div>
    <input type="hidden" name="secret_data" value="1" />
  </div>
  <div>
    <input type="radio" name="abc" value="a" />
    <input type="radio" name="abc" value="b" />
    <input type="radio" name="abc" value="c" />
  </div>
</form>
```



Zadanie 10

Utwórz sekcje z 4 dowolnymi inputami

14. Formularz

```
<form action="str.html" method="post">
  <div>
    <input type="text" name="your_name" />
  </div>
  <div>
    <input type="submit" name="send" value="send" />
  </div>
</form>
```

Strona www to nie tylko statyczna treść. Może też pozyskać i przekazać informacje od użytkownika. Jednym ze sposobów na otrzymanie danych od użytkownika jest formularz, który użytkownik wypełnia i wysyła.

Zawartość formularza trafia na serwer, gdzie jest przetwarzana. Może też być "przechwycona" w przeglądarce przez JavaScript (jeśli do strony jest dołączony odpowiedni kod).

- Element `<form>` oznacza formularz. Między jego otwarciem i zamknięciem powinny znaleźć się wszystkie elementy wysyłane w formularzu.
- Atrybut *action* oznacza adres, na który wysłane zostaną dane z formularza (jeśli nie ustawimy mu wartości, wyśle się na ten sam adres, pod którym znajduje się formularz).
- Atrybut *method* oznacza metodę przesyłu danych (najpopularniejsze to *get* i *post*).
- Jeśli formularz będziemy obsługiwać przy pomocy JS, nie musimy używać atrybutów *method* i *action*.
- Element `<input>` - podstawowy element formularza. Zawiera pojedynczą informację (np. wpisany tekst, wybraną opcję itp).
- Atrybut *type* - rodzaj elementu typu *input*. Pole tekstowe, checkbox do zaznaczenia, a może wybór daty? Szczegóły w następnym podrozdziale.
- Atrybut *name* - nazwa, pod którą jest dostępna informacja z danego inputa.
- Atrybut *value* - wartość inputa (może ją wpisać użytkownik, może ona też być już wypełniona w kodzie HTML).
- Element `<input type="submit">` - przycisk do wysłania formularza.



Zadanie 11

Utwórz formularz pobierający dane klienta (imie, nazwisko, email, wiek)

- Pamiętaj, że każdy formularz musi mieć przycisk z typem submit