

Listy: <https://www.youtube.com/watch?v=cLssy731qDQ> - film

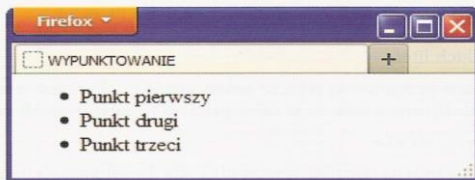
## Material:

**Lista**, nazywana również często wykazem, daje możliwość tworzenia uszeregowanych grup informacji w postaci punktów uporządkowanych (lista numerowana) lub nieuporządkowanych (lista punktowa, nienumerowana). Jest to jeden z częściej stosowanych elementów konstrukcyjnych, używany na przykład do budowy menu strony internetowej.

Listę punktową (nienumerowaną) tworzymy, wykorzystując dwa rodzaje znaczników. Pierwszy znacznik `<ul> ... </ul>` tworzy zewnętrzną ramę listy. Za każdy element listy odpowiada znacznik `<li> ... </li>`. Lista może składać się z dowolnej liczby elementów.

## Ćwiczenie 1.

```
<ul>
  <li> Punkt pierwszy </li>
  <li> Punkt drugi </li>
  <li> Punkt trzeci </li>
</ul>
```

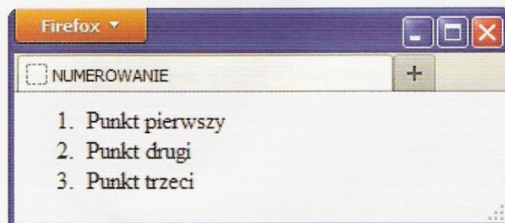


Domyślnym graficznym punktatorem jest wypełnione koło. Istnieje możliwość zmiany tej grafiki przez wprowadzenie dla znacznika `ul` atrybutu `type`, który przyjmuje jeden z trzech wariantów:

- **disc** (domyślny) – wypełnione koło,
- **circle** – okrąg,
- **square** – wypełniony kwadrat.

## Ćwiczenie 2.

```
<ol>
  <li> Punkt pierwszy </li>
  <li> Punkt drugi </li>
  <li> Punkt trzeci </li>
</ol>
```



Domyślnym numerowaniem dla listy uporządkowanej są cyfry arabskie. Istnieje możliwość zmiany wyświetlanych cyfr przez wprowadzenie dla znacznika `ol` atrybutu **type**, który przyjmuje jeden z pięciu wariantów:

- **1** (domyślny) – numeracja według liczb arabskich,
- **I** – według liczb rzymskich,
- **i** – według małych liczb rzymskich zapisanych małymi literami,
- **a** – według małych liter,
- **A** – według wielkich liter.

Znacznik `ol` może przyjmować jeszcze jeden atrybut, jakim jest **start**. Atrybut ten umożliwia rozpoczęcie numerowania w zależności od podanej wartości **n**:

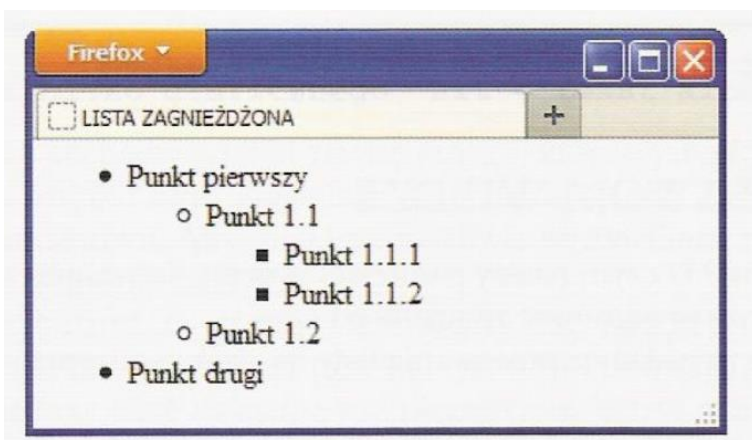
```
<ol start="n"> ... </ol>
```

Zmianę numeracji można również wprowadzić dla dowolnego elementu listy. Umożliwia to atrybut **value** przypisany do znacznika `li`. Wartość **n** ustawiona dla atrybutu **value** spowoduje zmianę kolejności numerowania również dla kolejnych elementów listy:

```
<li value="n"> ... </li>
```

Istnieje możliwość zagnieżdżenia wykazów, zarówno uporządkowanych, jak i nieuporządkowanych. Przykład (list. 8.3, rys. 8.3) przedstawia listę zagnieżdżoną składającą się z trzech poziomów. Należy zwrócić uwagę na miejsca zamykania odpowiednich znaczników `ol`, `ul` i `li` (znaczniki te wytłuszczono).

```
<ul>
  <li> Punkt pierwszy
    <ul>
      <li> Punkt 1.1
        <ul>
          <li> Punkt 1.1.1 </li>
          <li> Punkt 1.1.2 </li>
        </ul>
      </li>
      <li> Punkt 1.2 </li>
    </ul>
  </li>
  <li> Punkt drugi </li>
</ul>
```



## Zadanie sprawdzające.

Wykonaj listę wg wzoru:

- I. Wprowadzenie
  - 1. Składnia języka HTML
    - a. Znaczniki i elementy
    - b. Struktura dokumentu HTML
    - c. Pierwsza strona WWW
  - 7. Kaskadowe arkusze stylów
    - a. Struktura i wygląd dokumentu
    - b. Dołączanie stylów do dokumentów
      - Style zewnętrzne
      - Style wewnętrzne
    - c. Domyślny język stylów
  - 8. Składnia kaskadowych arkuszy stylów
- II. Dodatki
  - A. Instalacja oprogramowania
- III. Skorowidz

### Tabele

Film: <https://www.youtube.com/watch?v=kr22teOrCK8>

#### Material:

Tabele stanowią obecnie jeden ze sposobów prezentowania danych. Początkowo były wykorzystywane do tworzenia układów stron, które teraz opierają się na elementach `div`.

Tabela jest strukturą składającą się z wierszy podzielonych na komórki. Zewnętrzną ramę tabeli stanowi znacznik `<table> ...</table>`. Wewnątrz umieszczamy wiersze `<tr> ... </tr>`, a wewnątrz nich – poszczególne komórki `<td> ... </td>`.

Podstawową strukturę tabeli 2x2 prezentuje przykład (list. 10.1, rys. 10.1). Dodatkowo dla znacznika `table` wprowadzono argument `border`, którego wartość odpowiada grubości obramowania. Jeżeli nie zostanie on wprowadzony, obramowanie tabeli nie zostanie wyświetlone, widoczna będzie jedynie jej zawartość.

```
<table border="1">
  <tr>
    <td>komórka 1</td>
    <td>komórka 2</td>
  </tr>
  <tr>
    <td>komórka 3</td>
    <td>komórka 4</td>
  </tr>
</table>
```

```
<table border="1" cellpadding="10" cellspacing="15">
```

Poza zwykłymi komórkami do tabeli możemy również wprowadzić **komórki nagłówkowe** `<th> ... </th>`, zarówno poziome, jak i pionowe. Wstawia się je podobnie jak zwykłe komórki. Jedyna różnica polega na ich wyglądzie. Tekst wyświetlany w komórkach nagłówkowych jest pogrubiony i wyśrodkowany.

Każda tabela powinna mieć tytuł. Wprowadza się go za pomocą znacznika `<caption> ... </caption>` umieszczonego zaraz za znacznikiem `table`. Posiada on również atrybut `align` odpowiadający na wyrównanie tytułu (`left`, `right`, `center`, `top`, `bottom`):

```
<table>
<caption align="wyrównanie"> Tytuł tabeli </caption>
<!-- pozostała część tabeli-->
</table>
```

Atrybuty `width` (szerokość) oraz `height` (wysokość) można zastosować do ustawienia wymiarów tabeli. Można je wprowadzić wewnątrz znacznika `table` dla zmiany rozmiaru całej tabeli lub wewnątrz znacznika `td` dla określenia wymiarów konkretnej komórki. Oba atrybuty przyjmują konkretne wartości podane w pikselach lub wartości procentowe (w stosunku do wielkości strony):

```
<table width="200" height="100">
```

lub

```
<td width="40%" height="20%">
```



Tworzona tabela może mieć różną liczbę komórek w wierszach i kolumnach. Dzieje się tak, ponieważ język HTML pozwala na scalanie komórek w tabeli. Komórki mogą zostać scalone w wierszu przez wprowadzenie wewnątrz znacznika **td** atrybutu **colspan**, przyjmującego wartość odpowiadającą liczbie scalonych komórek (list. 10.3, rys. 10.3). Należy pamiętać, że przy scalaniu komórek musimy dostosować ich liczbę w odpowiednich wierszach (dwie scalone w pierwszym wierszu, dwie pojedyncze w drugim wierszu).

Listing 10.3

```
<table border="1">
<tr>
<td colspan="2">komórka 1 i 2</td>
</tr>
<tr>
<td>komórka 3</td>
<td>komórka 4</td>
</tr>
</table>
```

Komórki mogą również być scalone w kolumnie przez wprowadzenie wewnątrz znacznika **td** atrybutu **rowspan**, przyjmującego wartość odpowiadającą liczbie scalonych komórek (list. 10.4, rys. 10.4). Należy pamiętać, że przy scalaniu komórek musimy dostosować ich liczbę (dwie scalone w pierwszym wierszu i pojedyncza komórka, jedna komórka w drugim wierszu).

Listing 10.4

```
<table border="1">
<tr>
<td rowspan="2">komórki 1 i 3</td>
<td>komórka 2</td>
</tr>
<tr>
<td>komórka 4</td>
</tr>
</table>
```

Pozostało jeszcze dodać tabeli nieco koloru. HTML daje nam kilka możliwości. Pierwszą z nich jest wprowadzenie jednolitego koloru tła. Umożliwia to atrybut **bgcolor**, poznany już wcześniej przy ustawianiu tła całej strony. Aby ustawić odpowiedni kolor, wystarczy podać jego nazwę. Atrybut **background**, odwołujący się do odpowiedniego pliku graficznego (ścieżka dostępu wraz z nazwą i rozszerzeniem), pozwala wprowadzić tło graficzne. Ostatnim elementem, którego kolor można zmienić, jest obramowanie. Odpowiednim do tego atrybutem jest **bordercolor**, który jako wartość przyjmuje również nazwę koloru. Wszystkie te atrybuty możemy wprowadzić dla całej tabeli, dla pojedynczego wiersza lub dla jednej komórki (list. 10.5, rys. 10.5).

Listing 10.5

```
<table border="5" bgcolor="yellow" bordercolor="blue">
<tr bgcolor="green">
<td>komórka 1</td>
<td>komórka 2</td>
</tr>
<tr>
<td bgcolor="teal">komórka 3</td>
<td background="tlo.png">komórka 4</td>
</tr>
</table>
```

L.p.	Nazwisko	Imię	Klasa
1.	Nowak	Adam	2A
2.	Kowalski	Piotr	
3.	Piotrowski	Michał	2B
4.	Jerdamska	Oliwia 2C	

## Tabela dla zaawansowanych:

1. Wykonaj poniższe tabelki na stronie WWW. Każdą z nich sformatuj według własnego pomysłu, wykorzystując poznane atrybuty (zmiana koloru, tła, rozmiaru).

Tabela 1


Tabela2


Tabela 3
