**Міністерство Освіти І НАУКИ України**

**Національний університет "Львівська політехніка"**

Інститут **ІКНІ**

Кафедра **ПЗ**

### ЗВІТ

До лабораторної роботи № 1

**З дисципліни:** *“Програмування в Інтернет”*

**На тему:** *“Структура DOM**та методи доступу до вузлів дерева.”*

**Лектор:**

проф. каф. ПЗ

Мельник Р. А.

**Виконала:**

ст. гр. ПЗ-26

Гуль О. М.

**Прийняла:**

асист. каф. ПЗ

Майхер В. Ю.

« \_\_\_\_ » \_\_\_\_\_\_\_\_ 2021 р.

∑= \_\_\_\_\_ .

Львів – 2021

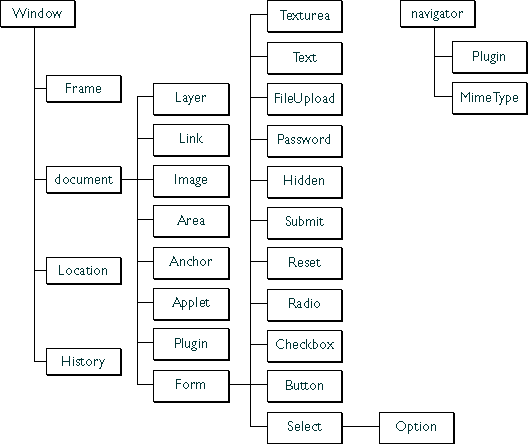
**Тема роботи:** Структура DOM та методи доступу до вузлів дерева.

**Мета роботи:** Ознайомитись з ієрархічною структурою об’єктів DOM.

**ТЕОРЕТИЧНІ ВІДОМОСТІ**

web-сторінка може мати вигляд дерева, вузли якого є об’єктами, до властивостей яких доступаються операторами мови програмування Javascript.

Модель DOM (Document Object Model, об’єктна модель документа) містить низку стандартних глобальних об’єктів. Зокрема, це **window, navigator, document, screen, history, location**. Ієрархію основних об’єктів у моделі DOM подано   
на рис. 1.2.



*Рис. 1.2. Дерево моделі DOM об’єкта window браузера*

**Типи вузлів та методи доступу до них**

За місцем розташування на дереві виділено такі типи вузлів: **parent** – батьківський, **child** – породжений, **sibling** – сусідній, **firstchild** – перший породжений, **lastchild** – останній породжений.

Назви вузлів є складовими назвами властивостей вузлів та параметрів методів, які ними оперують (ліквідують, модифікують, додають тощо). Вузли DOM-дерева мають такі властивості:

* **firstChild, lastChild** – початковий та кінцевий породжені вузли;
* **childNodes** – масив породжених вузлів;
* **nextSibling, previousSibling** – сусідні вузли (по одному), що мають одного батька;
* **parentNode** – елемент, що породив цей вузол.

Мова програмування дає змогу змінювати склад вузлів, вигляд та зміст web-сторінки. На моделі DOM програміст має можливість виконувати операції двох типів: на рівні вузлів для зміни структури дерева додаванням або відніманням вузла та заміни вузла; на рівні вузла для зміни змісту елемента. Зокрема, для першого типу операцій застосовуються методи:

* [**appendChild**](http://www.w3schools.com/dom/met_node_appendchild.asp)**(node)**: розміщає вузол у кінці списку породжених вузлів до заданого;
* [**insertBefore**](http://www.w3schools.com/dom/met_node_insertbefore.asp)**(newChild, oldChild)**: розміщає новий вузол перед вузлом oldChild у списку породжених вузлів до заданого;
* [**removeChild**](http://www.w3schools.com/dom/met_node_removechild.asp)**(node)**: знищує наведений вузол зі списку породжених вузлів до заданого;
* [**replaceChild**](http://www.w3schools.com/dom/met_node_replacechild.asp)**(newChild, oldChild)**: заміняє пород­жений на новий вузол.

**Опрацювання подій**

Події та обробники подій є дуже важливою частиною у програмування на JavaScript. ***Події*** (*Events*), головним чином, ініціюються тими або іншими діями користувача. Події – це дії, які відбуваються, внаслідок того, що робить користувач. Наприклад, якщо користувач клацає по деякій кнопці, відбувається подія Click. Якщо миша перетинає яке-небудь посилання - відбувається подія MouseOver. Існує певний набір подій, які розпізнає той чи інший броузер.

Ми можемо примусити нашу JavaScript-програму реагувати на деякі з них. І це може бути виконано за допомогою спеціальних програм ***обробки подій***. Так, в результаті клацання по кнопці може створюватися випадаюче вікно. Це означає, що створення вікна повинно бути реакцією на подію Click. Програма - обробник подій, яку ми повинні використати в даному випадку, називається onClick. І вона повідомляє комп'ютер, що потрібно робити, якщо відбудеться дана подія.

Обробник подій записується в документ як атрибут тега HTML, до якого ви приписуєте код JavaScript. Наприклад, ви створили ***функцію*** JavaScript, і назвали її compute.

Ви можете поміщати будь-які твердження JavaScriptусередині кавичок onClick. Ці твердження будуть виконані, коли користувач натискатиме на кнопку. Якщо Ви хочете включити більш ніж одне твердження, то окремі твердження записуються через крапку з комою (;).

Ви можете використовувати в скрипті безліч різних типів функцій обробки подій. Щоб дізнатись про усі існуючі обробники подій, звертайтеся до відповідного довідника. Наведемо лише деякі з них:

* **onLoad** - виконання скрипта або функції при завантаженні;
* **onChange** - породжується при зміні значення елемента форми;
* **onClick** - породжується при виборі об'єкту (button, checkbox і т.п.);
* **onSelect** - породжується при виборі текстового об'єкту (text, textarea);
* **onSubmit** - при натисненні на кнопку Submit;
* **onUnload** - при переході до іншої сторінки.

**ІНДИВІДУАЛЬНЕ ЗАВДАННЯ**

1. Пройти наступні частини [курсу по html](https://www.codecademy.com/learn/learn-html) на codecademy: [Elements and Structure](https://www.codecademy.com/courses/learn-html/lessons/intro-to-html/exercises/intro) та [Tables](https://www.codecademy.com/courses/learn-html/lessons/html-tables/exercises/why-tables)

2. Розробити web-сторінку яка відображатиме поле для гри у шахи з положенням фігур головоломки узятої із [списку головоломок](https://www.chess.com/puzzles/problems) (головоломку обрати відносно номеру розташування у списку відповідно до вашого порядкового номера). Сторінка має:

* містити елементи HTML для занесення даних гравця,
* використовувати HTML таблицю,
* містити посилання на правила шахів та головоломку.
* містити зображення фігур.

**ХІД ВИКОНАННЯ**

1. Пройшла частину Elements and Structure курсу по html на codacademy

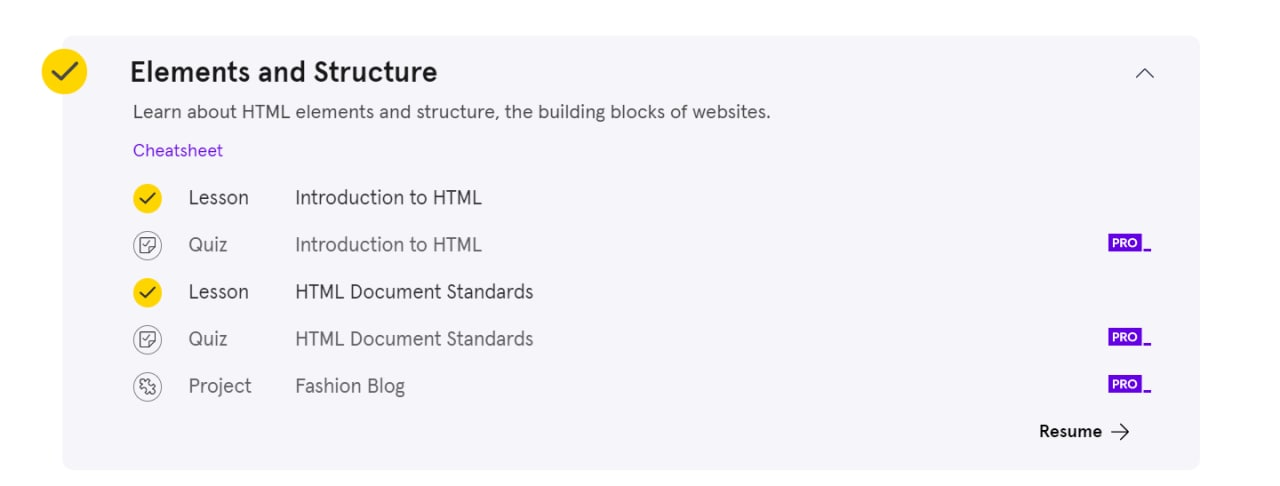


Рис. 1. Пройдена частина Elements and Structure

1. Пройшла частину Tables курсу по html на codacademy

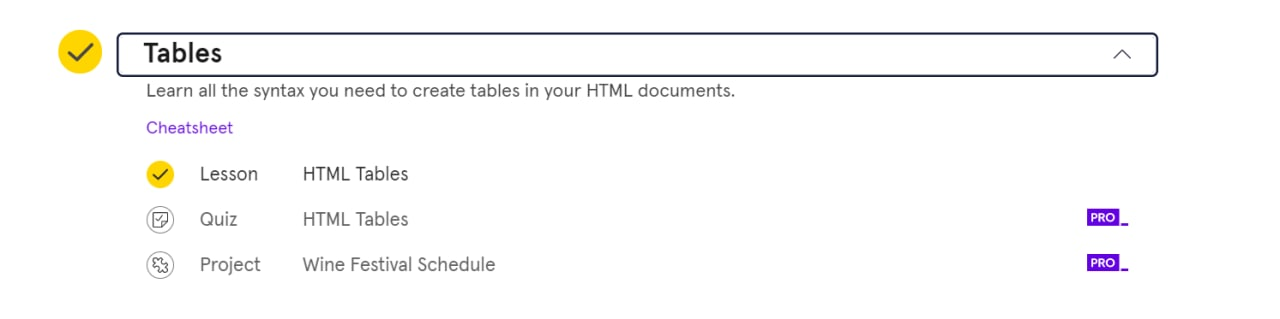


Рис. 2. Пройдена частина Tables

1. Розробила web-сторінку яка відображатиме поле для гри у шахи з положенням фігур головоломки

Варіант головоломки для відтворення:

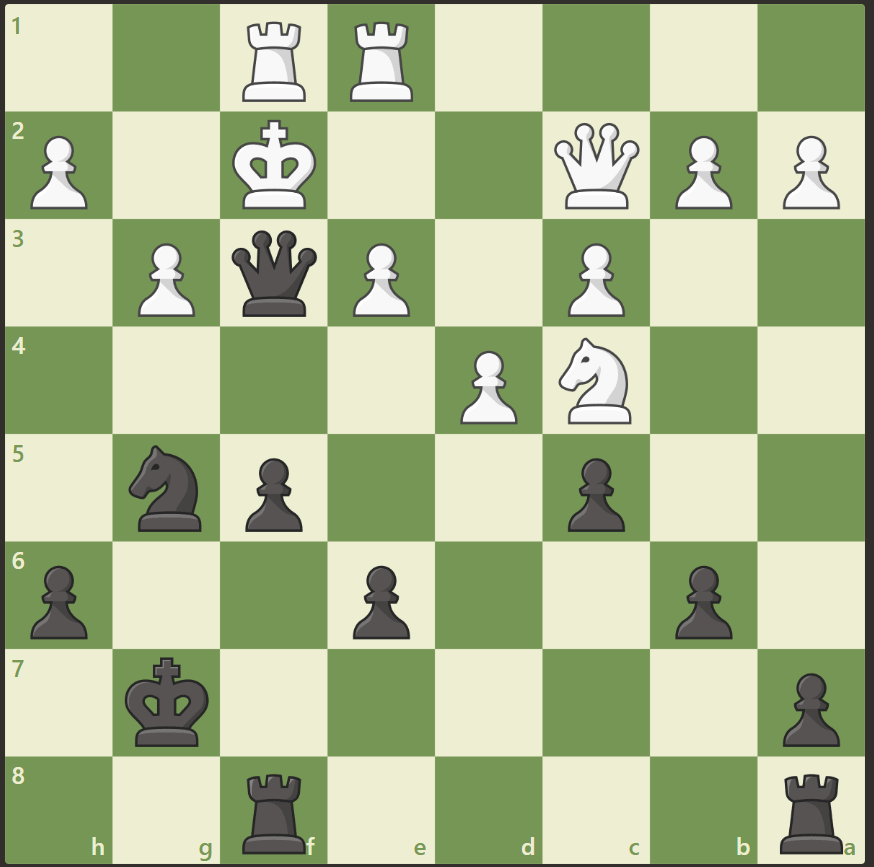


Рис. 3. Головоломка

**Код завдання:**

**index.html**

<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>Chess</title>  
 <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css" integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T" crossorigin="anonymous">  
 <link rel="stylesheet" href="style.css">  
  
</head>  
<body>  
 <div class="center">  
 <p><a class="text-dark" href="https://www.chesscoachonline.com/chess-articles/chess-rules">Chess rules</a></p>  
 <p><a class="text-dark" href="https://www.chess.com/puzzles/problem/134311">The problem</a></p>  
 <form>  
 <label for="name">Name</label>  
 <input id="name" placeholder="Enter name" name="name" type="text">  
 <input type="submit" value="OK">  
 </form>  
 </div>  
  
 <table>  
 <tbody>  
 <tr>  
 <td class="cell white">  
 <span class="box top">8</span>  
 </td>  
 <td class="cell black"></td>  
 <td class="cell white">  
 <img src="images/wr.png">  
 </td>  
 <td class="cell black">  
 <img src="images/wr.png">  
 </td>  
 <td class="cell white"></td>  
 <td class="cell black"></td>  
 <td class="cell white"></td>  
 <td class="cell black"></td>  
 </tr>  
 <tr>  
 <td class="cell black">  
 <span class="box top">7</span>  
 <div class="box">  
 <img src="images/wp.png">  
 </div>  
 </td>  
 <td class="cell white"></td>  
 <td class="cell black">  
 <img src="images/wk.png">  
 </td>  
 <td class="cell white"></td>  
 <td class="cell black"></td>  
 <td class="cell white">  
 <img src="images/wq.png">  
 </td>  
 <td class="cell black">  
 <img src="images/wp.png">  
 </td>  
 <td class="cell white">  
 <img src="images/wp.png">  
 </td>  
 </tr>  
 <tr>  
 <td class="cell white">  
 <span class="box top">6</span>  
 </td>  
 <td class="cell black">  
 <img src="images/wp.png">  
 </td>  
 <td class="cell white">  
 <img src="images/bq.png">  
 </td>  
 <td class="cell black">  
 <img src="images/wp.png">  
 </td>  
 <td class="cell white"></td>  
 <td class="cell black">  
 <img src="images/wp.png">  
 </td>  
 <td class="cell white"></td>  
 <td class="cell black"></td>  
 </tr>  
 <tr>  
 <td class="cell black">  
 <span class="box top">5</span>  
 </td>  
 <td class="cell white"></td>  
 <td class="cell black"></td>  
 <td class="cell white"></td>  
 <td class="cell black">  
 <img src="images/wp.png">  
 </td>  
 <td class="cell white">  
 <img src="images/wn.png">  
 </td>  
 <td class="cell black"></td>  
 <td class="cell white"></td>  
 </tr>  
 <tr>  
 <td class="cell white">  
 <span class="box top">4</span>  
 </td>  
 <td class="cell black">  
 <img src="images/bn.png">  
 </td>  
 <td class="cell white">  
 <img src="images/bp.png">  
 </td>  
 <td class="cell black"></td>  
 <td class="cell white"></td>  
 <td class="cell black">  
 <img src="images/bp.png">  
 </td>  
 <td class="cell white"></td>  
 <td class="cell black"></td>  
 </tr>  
 <tr>  
 <td class="cell black">  
 <span class="box top">3</span>  
 <div>  
 <img src="images/bp.png">  
 </div>  
 </td>  
 <td class="cell white"></td>  
 <td class="cell black"></td>  
 <td class="cell white">  
 <img src="images/bp.png">  
 </td>  
 <td class="cell black"></td>  
 <td class="cell white"></td>  
 <td class="cell black">  
 <img src="images/bp.png">  
 </td>  
 <td class="cell white"></td>  
 </tr>  
 <tr>  
 <td class="cell white">  
 <span class="box top">2</span>  
 </td>  
 <td class="cell black">  
 <img src="images/bk.png">  
 </td>  
 <td class="cell white"></td>  
 <td class="cell black"></td>  
 <td class="cell white"></td>  
 <td class="cell black"></td>  
 <td class="cell white"></td>  
 <td class="cell black">  
 <img src="images/bp.png">  
 </td>  
 </tr>  
 <tr>  
 <td class="cell black">  
 <span class="bottom-box top">h</span>  
 <span class="box top">1</span>  
 </td>  
 <td class="cell white">  
 <span class="bottom-box top">g</span>  
 </td>  
 <td class="cell black">  
 <span class="bottom-box top">f</span>  
 <div class="box">  
 <img src="images/br.png">  
 </div>  
 </td>  
 <td class="cell white">  
 <span class="bottom-box top">e</span>  
 </td>  
 <td class="cell black">  
 <span class="bottom-box top">d</span>  
 </td>  
 <td class="cell white">  
 <span class="bottom-box top">c</span>  
 </td>  
 <td class="cell black">  
 <span class="bottom-box top">b</span>  
 </td>  
 <td class="cell white">  
 <span class="bottom-box top">a</span>  
 <div class="box">  
 <img src="images/br.png">  
 </div>  
 </td>  
 </tr>  
 </tbody>  
 </table>  
</body>  
</html>

**style.css**

.cell {  
 width: 4em;  
 height: 4em;  
 position: relative;  
}  
  
.black {  
 background-color: #ff645a;  
 color: #d2d2d2;  
}  
  
.white {  
 background-color: #d2d2d2;  
 color: #ff645a;  
}  
  
.box{  
 width: 100%;  
 height: 100%;  
 position: absolute;  
 top: 0;  
 left: 0;  
 margin-left: 0.15em;  
}  
  
.bottom-box{  
 position: absolute;  
 right: 0;  
 bottom: 0;  
 margin-right: 0.15em;  
}  
  
.top{  
 z-index: 9;  
}  
  
table {  
 margin: auto;  
 border-spacing: 0;  
}  
  
img {  
 max-width: 100%;  
 max-height: 100%;  
 margin: auto;  
}  
  
.center {  
 text-align: center;  
}  
  
p {  
 font-size: 1.5em;  
}  
  
span {  
 font-weight: bold;  
}

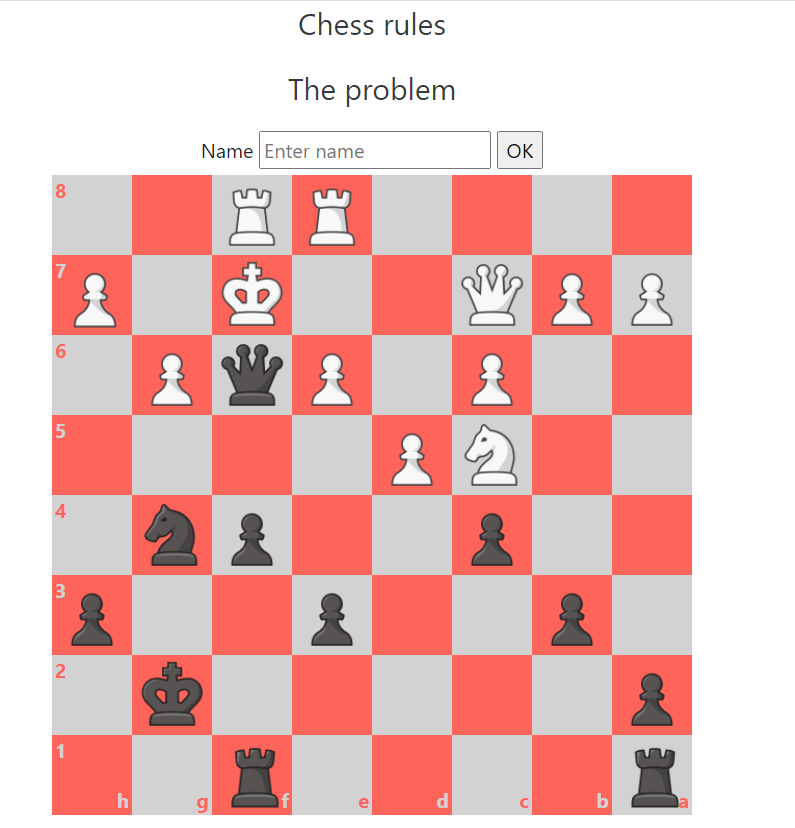
****

Рис. 4. Розроблена web-сторінка

**ВИСНОВКИ**

На лабораторній роботі я ознайомилась з ієрархічною структурою об’єктів DOM. Пройшла елементи курсу по HTML на codeacademy. Створила веб сторінку, що відображає поле для гри в шахи з положенням фігур головоломки. Навчилась працювати з таблицями в HTML, додавати зображення, гіперпосилання та форму для вводу даних користувача.