Федеральное государственное автономное

Образовательное учреждение

Высшего образования

«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт Космических и информационных технологий

институт

Кафедра «Информатика»

кафедра

**ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ №9**

Манипулирование DOM-деревом

тема

Преподаватель \_\_\_\_\_\_\_\_\_ А.М. Скрипачев

подпись, дата инициалы, фамилия

Студент КИ16-16Б, №031620303 \_\_\_\_\_\_\_\_\_ В.В. Кокташев

Номер группы, зачетной книжки подпись, дата инициалы, фамилия

Красноярск 2018

# Цель работы

# Изучение основ манипулирования с DOM-деревом.

# Задание

Реализуйте динамическое изменение дерева узлов вашей странички. Изменение должно инициироваться событиями. Для регистрации событий используйте метод addEventListener, для выборки элементов привязки используйте различные способы.

# Краткий теоретический материал

## Представление документов в виде дерева

HTML-документы имеют иерархическую структуру вложенных тегов, которая в DOM представлена в виде дерева объектов. Узлы дерева представляют различные типы содержимого документа. В первую очередь, древовидное представление HTML-документа содержит узлы, представляющие элементы или теги, такие как <body> и <p>, и узлы, представляющие строки текста. HTML-документ также может содержать узлы, представляющие HTML-комментарии.

Узел, расположенный непосредственно над данным узлом, называется родительским по отношению к данному узлу. Узлы, расположенные на один уровень ниже другого узла, являются дочерними по отношению к данному узлу. Узлы, находящиеся на том же уровне и имеющие того же родителя, называются братьями (siblings). Узлы, расположенные на любое число уровней ниже другого узла, являются его потомками. Родительские, прародительские и любые другие узлы, расположенные выше данного узла, являются его предками.

Интерфейс Node определяет свойства и методы для перемещения по дереву и манипуляций им. Свойство childNodes объекта Node возвращает список дочерних узлов, свойства firstChild, lastChild, nextSibling, previousSibling и parentNode предоставляют средство обхода узлов дерева.

Такие методы, как appendChild(), removeChild(), replaceChild() и insertBefore(), позволяют добавлять узлы в дерево документа и удалять их.

## Выбор элементов документа

Работа большинства клиентских программ на языке JavaScript так или иначе связана с манипулированием элементами документа. В ходе выполнения эти программы могут использовать глобальную переменную document, ссылающуюся на объект Document. Однако, чтобы выполнить какие-либо манипуляции с элементами документа, программа должна каким-то образом получить, или выбрать, объекты Element, ссылающиеся на эти элементы документа. Модель DOM определяет несколько способов выборки элементов. Выбрать элемент или элементы документа можно:

1. по значению атрибута id;
2. по значению атрибута name;
3. по имени тега;
4. по имени класса или классов CSS;
5. по совпадению с определенным селектором CSS.

## Выбор элементов по значению атрибута id

Все HTML-элементы имеют атрибуты id. Значение этого атрибута должно быть уникальным в пределах документа - никакие два элемента в одном и том же документе не должны иметь одинаковые значения атрибута id. Выбрать элемент по уникальному значению атрибута id можно с помощью метода getElementById() объекта Document.

## Выбор элементов по значению атрибута name

HTML-атрибут name первоначально предназначался для присваивания имен элементам форм, и значение этого атрибута использовалось, когда выполнялась отправка данных формы на сервер. Подобно атрибуту id, атрибут name присваивает имя элементу. Однако, в отличие от id, значение атрибута name не обязано быть уникальным: одно и то же имя могут иметь сразу несколько элементов, что вполне обычно при использовании в формах радиокнопок и флажков.

Выбрать HTML-элементы, опираясь на значения их атрибутов name, можно с помощью метода getElementsByName() объекта Document.

## Выбор элементов по типу

Метод getElementsByTagName() объекта Document позволяет выбрать все HTML или XMLэлементы указанного типа (или по имени тега). Подобно методу getElementsByName(), getElementsByTagName() возвращает объект NodeList. Элементы документа включаются в массив NodeList в том же порядке, в каком они следуют в документе.

## Выбор элементов с использованием селекторов CSS

Каскадные таблицы стилей CSS имеют очень мощные синтаксические конструкции, известные как селекторы, позволяющие описывать элементы или множества элементов документа. Наряду со стандартизацией селекторов CSS3, другой стандарт консорциума W3C, известный как Selectors API, определяет методы JavaScript для получения элементов, соответствующих указанному селектору.

Ключевым в этом API является метод querySelectorAll() объекта Document. Он принимает единственный строковый аргумент с селектором CSS и возвращает объект NodeList, представляющий все элементы документа, соответствующие селектору.

В дополнение к методу querySelectorAll() объект документа также определяет метод querySelector(), подобный методу querySelectorAll(), - с тем отличием, что он возвращает только первый (в порядке следования в документе) соответствующий элемент или null, в случае отсутствия соответствующих элементов.

## Добавление и удаление узлов

Методы для создания узлов:

document.createElement(tag) – создает элемент

document.createTextNode(value) – создает текстовый узел

elem.cloneNode(deep) – клонирует элемент, если deep == true, то со всеми потомками, если false – без потомков.

Вставка и удаление узлов:

parent.appendChild(elem)

parent.insertBefore(elem, nextSibling)

parent.removeChild(elem)

parent.replaceChild(newElem, elem)

Все эти методы возвращают elem.

# Ход работы

1. Реализовано динамическое изменение узлов странички посредством добавления элементов в список. Для выборки элементов использовались различные способы (выбор элементов по классу CSS, выбор элементов по значению атрибута id и выбор элементов по имени) Результат добавления элементов изображен на рисунках 1 и 2.



Рисунок 1 – Функция вставки строки в таблицу

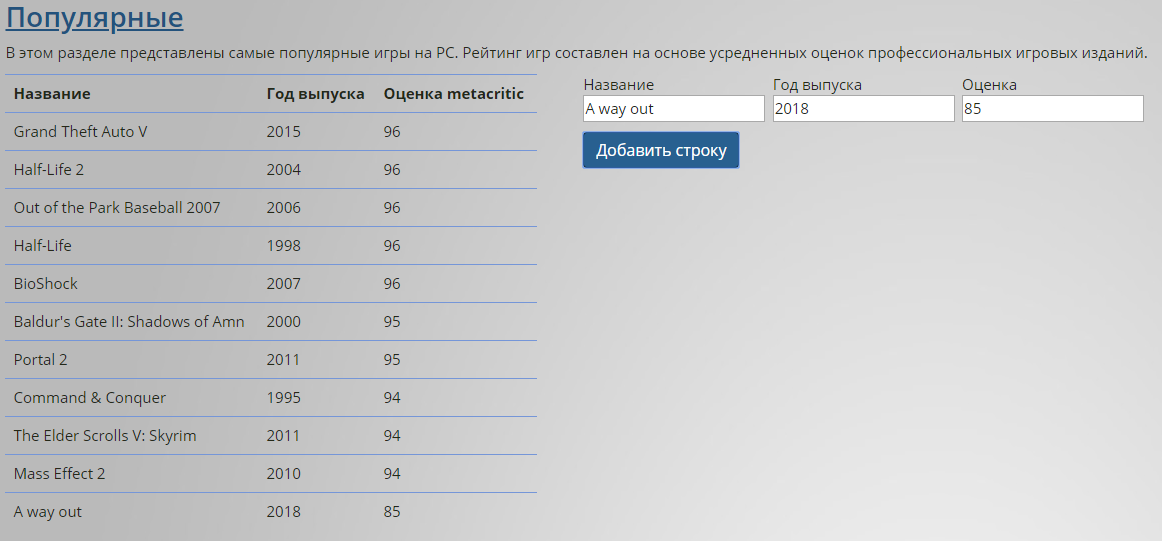


Рисунок 2 – Вставка строки в таблицу

# Вывод

В ходе лабораторной работы были изучены основы манипулирования с DOM-деревом, также полученные знания закреплены на практике путем добавления и удаления списков и увеличения изображения при нажатии на него.

# ПРИЛОЖЕНИЕ

Листинг 1 – файл scripts.js

var mas = ["img1.jpg", "img2.jpg", "img3.jpg", "img4.jpg"];

var i = 0;

function prevImage()

{

i--;

if (i < 0)

i = mas.length - 1;p

document.imagge.src = mas[i];

}

function nextImage()

{

i++;

if (i == mas.length)

i = 0;

document.imagge.src = mas[i];

}

document.getElementsByName("addRowBut")[0].addEventListener("click", addRow, false);

function addRow()

{

var newRow = document.createElement('tr');

var table = document.getElementsByClassName("tbodyy")[0];

var newColumn = document.createElement('td'); newColumn.innerHTML = document.getElementById('gameName').value;

var newColumn1 = document.createElement('td'); newColumn1.innerHTML = document.getElementById('gameYear').value;

var newColumn2 = document.createElement('td'); newColumn2.innerHTML = document.getElementById('gameScore').value;

newRow.appendChild(newColumn)

newRow.appendChild(newColumn1);

newRow.appendChild(newColumn2);

table.appendChild(newRow);

}

Листинг 2 – Lab\_1.html

<!DOCTYPE html>

<html lang="ru">

<head>

<meta charset="utf-8" />

<meta content="игра, ключи, game, steam, стратегия, шутер, гонки, rpg, uplay,

origin, gog" name="keywords">

<meta content="Интернет-магазин игр .Продажа лицензионных ключей steam, uplay,

origin, gog, battle.net по разумным ценам." name="description">

<title>Магазин игр GameShoр</title>

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

<link href="css/bootstrap.css" rel="stylesheet">

<script src="jquery-3.3.1.js"></script>

<script src="js/bootstrap.js"></script>

<script src="scripts.js" defer></script>

<script>

function up() {

window.scrollTo(0,0);

}

</script>

<style type="text/css">

body

{

color: #242818;

font-family: 'Open Sans';

background-image: url(background.jpg);

background-attachment: fixed;

}

.item

{

background: #333;

text-align: center;

height: 390px;

}

.carousel

{

margin-top: -23px;

}

</style>

</head>

<body

bgcolor="#edefea" onload="task1()">

<button class = "btn btn-primary" type="button" onclick="up()" style="position: fixed; bottom: 5%; right: 5%; z-index: 1" ><span class="glyphicon glyphicon-arrow-up"></span></button>

<nav class="navbar navbar-inverse">

<div class="container-fluid">

<div class="navbar-header">

<a class="navbar-brand" href="#" style="font-family: cursive"><span class="glyphicon glyphicon-home"></span> GameShop</a>

</div>

<ul class="nav navbar-nav">

<li><a href="#genres">Игры по жанрам</a></li>

<li><a href="#popular">Популярные</a></li>

<li><a href="#task">Сайты со схожей тематикой</a></li>

<li><a href="page2.html">Обратная связь</a></li>

</ul>

</div>

</nav>

<div class="row">

<div id="carousel1" class="carousel slide">

<ol class="carousel-indicators">

<li data-target="#carousel1" data-slide-to="0"></li>

<li data-target="#carousel1" data-slide-to="1"></li>

<li data-target="#carousel1" data-slide-to="2" class="active"></li>

<li data-target="#carousel1" data-slide-to="3"></li>

</ol>

<div class="carousel-inner" role="listbox">

<div class="item">

<img src="img1.jpg" class="img-responsive center-block">

<div class="carousel-caption">

<h2>The Witcher 3: Wild Hunt</h2>

<p>Ролевая игра нового поколения</p>

</div>

</div>

<div class="item">

<img src="img2.jpg" class="img-responsive center-block">

<div class="carousel-caption">

<h2>Sea of Thieves</h2>

<p>Настоящие морские приключения</p>

</div>

</div>

<div class="item active">

<img src="img3.jpg" class="img-responsive center-block" >

<div class="carousel-caption">

<h2>Assassin's Creed Origins</h2>

<p>История о том, как все начиналось</p>

</div>

</div>

<div class="item">

<img src="img4.jpg" class="img-responsive center-block">

<div class="carousel-caption">

<h2>FINAL FANTASY XV</h2>

<p>Долгожданная новая глава легендарной серии</p>

</div>

</div>

</div>

<a class="left carousel-control" href="#carousel1" role="button" data-slide="prev">

<span class="glyphicon glyphicon-chevron-left"></span>

</a>

<a class="right carousel-control" href="#carousel1" role="button" data-slide="next">

<span class="glyphicon glyphicon-chevron-right"></span>

</a>

</div>

</div>

<div class ="row" style="margin-left: 1%; font-size: 15px">

<h3 class="col-md-12"><a name="genres"> Игры по жанрам </a></h3>

<p class="col-md-12"> Здесь вы можете выбрать интересующий вас жанр игр.</p>

<p class="col-md-12">На нашем сайте предствалены следующие категории:</p>

<ul type="disc">

<li> Action </li>

<li> RPG </li>

<li> Гонки </li>

<li> Стратегии </li>

<li> Спортивные </li>

<li> Шутеры </li>

<li> Симуляторы </li>

</ul>

</div>

<hr width="100%" size="1" color="#000000" />

<div class ="row" style="margin-left: 1%; font-size: 15px">

<h3 class="col-md-12"> <a name="popular"> Популярные </a></h3>

<p class="col-md-12"> В этом разделе представлены самые популярные игры на PC. Рейтинг игр составлен на основе усредненных оценок профессиональных игровых изданий.</p>

<div class="col-md-5">

<table class="table table-hover">

<tbody class="tbodyy">

<tr>

<th>Название</th>

<th>Год выпуска</th>

<th>Оценка metacritic</th>

</tr>

<tr><td>Grand Theft Auto V</td><td>2015</td><td>96</td></tr>

<tr><td>Half-Life 2</td><td>2004</td><td>96</td></tr>

<tr><td>Out of the Park Baseball 2007</td><td>2006</td><td>96</td></tr>

<tr><td>Half-Life</td><td>1998</td><td>96</td></tr>

<tr><td>BioShock</td><td>2007</td><td>96</td></tr>

<tr><td>Baldur's Gate II: Shadows of Amn</td><td>2000</td><td>95</td></tr>

<tr><td>Portal 2</td><td>2011</td><td>95</td></tr>

<tr><td>Command & Conquer</td><td>1995</td><td>94</td></tr>

<tr><td>The Elder Scrolls V: Skyrim</td><td>2011</td><td>94</td></tr>

<tr><td>Mass Effect 2</td><td>2010</td><td>94</td></tr>

</tbody>

</table>

</div>

<div class="col-md-7">

<div class="col-md-3">

Название <br>

<input type="text" id="gameName">

<button class="btn btn-primary" name = "addRowBut" style="margin-top: 10px">Добавить строку</button>

</div>

<div class="col-md-3">

Год выпуска <br>

<input type="text" id="gameYear">

</div>

<div class="col-md-3">

Оценка <br>

<input type="text" id="gameScore">

</div>

</div>

</div>

<hr width="100%" size="1" color="#000000" />

<div class ="row" style="margin-left: 1%; font-size: 15px">

<img src="img1.jpg" name="imagge" class="col-md-8 col-md-push-2">

</div>

<div class ="row" style="margin-top: 2%">

<button class="btn btn-primary col-md-1 col-md-push-4" onclick="prevImage()">Назад</button>

<button class="btn btn-primary col-md-1 col-md-push-6" onclick="nextImage()">Вперед</button>

</div>

<hr width="100%" size="1" color="#000000" />

<div class ="row" style="margin-left: 1%; margin-top: -20px; font-size: 12px">

<h4><a name="task">Сайты со схожей тематикой</a></h4>

<a href="https://gamazavr.ru/">Гамазавр</a>

<a href="http://store.steampowered.com/?l=russian">Steam</a>

<a href="https://store.playstation.com/ru-ru/home/games">Playstation Store</a>

</div>

</div>

</body>

</html>