Министерство науки и высшего образования Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования «Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых» (ВЛГУ)

Кафедра ФиПМ

ЛАБОРАТОРНАЯ РАБОТА № 3

по дисциплине: «Технология разработки веб-приложений»

на тему: «Использование объектной модели документа (DOM)»

Выполнил:

студент группы ИТ-122

Овчинникова В.А.

Проверил:

Болачков А. В.

Владимир 2024

**Цель работы**. Практическое освоение возможностей языка JavaScript и модели Document Object Model (DOM) для динамической работы с содержимым веб-страницы.

**Задание**. Модифицировать результаты лабораторной работы №4 (или №5) по дисциплине «Языки разметки и основы веб-дизайна», реализовав на языке JavaScript динамическую работу с содержимым страницы и водимыми пользователем данными.

1. Реализовать при выборе пункта меню выделение соответствующего раздела документа изменением его фона. Если у какого-либо раздела фон уже был изменён, то при выборе другого раздела цвет фона необходимо вернуть в начальное состояние.

2. При нажатии на названии столбца таблицы реализовать выделение тенью содержимого остальных ячеек данного столбца. При повторном нажатии на названии или при выборе названия другого столбца выделение должно пропадать.

3. Нажатие на элементы страницы, описанные дескриптором aside, должно приводить к отображению соответствующего содержимого во всплывающем окне браузера.

4. Для опросной формы реализовать обработку нажатий кнопок сброса и отправки данных.

• Нажатие на кнопку сброса данных должно сопровождаться появлением диалогового окна запроса на подтверждение сброса. Если пользователь подтверждает свои действия, то должен произойти сброс содержимого полей в начальное состояние, а также односекундное изменение цвета фона каждого из полей на красный цвет.

• Нажатие на кнопку отправки данных () должно приводить к односекундному изменению фона каждого из полей формы на синий цвет и последующему появлению информационного диалогового окна с сообщением о том, что данные отправлены.

**Код HTML-файла**

<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>Title</title>  
 <link rel="stylesheet" href="stile.css">  
 <link rel="stylesheet" href="stile1.css">  
</head>  
<body>  
<dialog>  
 <p id = "itemDialog"></p>  
 <button id="close" onclick="*dialog*.close();" style="">Зкрыть окно</button>  
</dialog>  
<header>  
 <h1>MySyte.com - Сайт о Lorem Ipsum</h1>  
 <nav>  
 <ul>  
 <li><a href="#" class="menu-item" data-section="section1">Главная</a></li>  
 <li><a href="#" class="menu-item" data-section="section2">Таблица</a></li>  
 <li><a href="#" class="menu-item" data-section="section3">Контакты</a></li>  
 </ul>  
 </nav>  
</header>  
<div class="wrapper">  
 <aside id="sidebar1" >  
 <h2 class="item">The standard Lorem Ipsum passage</h2>  
 <p class="item">"Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt  
 ut labore et dolore magna aliqua..."</p>  
 </aside>  
 <aside id="sidebar2" >  
 <h2 class="item">1914 translation by H. Rackham</h2>  
 <p class="item">It is a fact that a reader will be distracted by the readable  
 content of a page when looking at its layout.</p>  
 <h3 >Options</h3>  
 <ul class="item">  
 <li>Item1</li>  
 <li>Item2</li>  
 <li>Item3</li>  
 </ul>  
 </aside>  
 <article id="section1" class="section">  
 <h2>What is Lorem Ipsum?</h2>  
 <p>Lorem Ipsum is simply dummy text of the printing and typesetting industry...</p>  
 <p>Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of  
 classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock,  
 a Latin professor at Hampden-Sydney College in Virginia...</p>  
 </article>  
 <table id="section2" class="section">  
 <caption>Таблица</caption>  
 <thead>  
 <tr>  
 <th onclick="toggleHighlight(0)">Столбец 1</th>  
 <th onclick="toggleHighlight(1)">Столбец 2</th>  
 <th onclick="toggleHighlight(2)">Столбец 3</th>  
 </tr>  
 </thead>  
 <tbody>  
 <tr>  
 <td>Ячейка 1-1</td>  
 <td>Ячейка 1-2</td>  
 <td>Ячейка 1-3</td>  
 </tr>  
 <tr>  
 <td>Ячейка 2-1</td>  
 <td>Ячейка 2-2</td>  
 <td>Ячейка 2-3</td>  
 </tr>  
 <tr>  
 <td>Ячейка 3-1</td>  
 <td>Ячейка 3-2</td>  
 <td>Ячейка 3-3</td>  
 </tr>  
 </tbody>  
 </table>  
  
 <form id="surveyForm" >  
 <input type="text" placeholder="Ваше имя" required>  
 <input type="email" placeholder="Ваш email" required>  
 <input type="reset" value="Сбросить">  
 <input type="submit" value="Отправить">  
 </form>  
</div>  
<footer id="section3" class="section">  
 <p >Contacts: admin@mysyte.com</p>  
 <p>Copyright © MySyte.com, 2016</p>  
</footer>  
<script src="index.js"></script>  
</body>  
</html>

**Код JS-файла**

*document*.addEventListener('DOMContentLoaded', () => {  
 const menuItems = *document*.querySelectorAll('.menu-item');  
 const sections = *document*.querySelectorAll('.section');  
 menuItems.forEach(item => {  
 item.addEventListener('click', (event) => {  
 sections.forEach(section => {  
 section.classList.remove('active');  
 });  
 const sectionId = item.getAttribute('data-section');  
 const selectedSection = *document*.getElementById(sectionId);  
 selectedSection.classList.add('active');  
 setTimeout(() => {  
 selectedSection.classList.remove('active');  
 }, 4000);  
  
 });  
  
 });  
});  
let *currentHighlightedColumn* = null;  
function toggleHighlight(columnIndex) {  
 const rows = *document*.querySelectorAll('table tbody tr');  
 if (*currentHighlightedColumn* === columnIndex) {  
 // Убрать выделение  
 for (const row of rows) {  
 row.cells[columnIndex].classList.remove('highlight');  
 }  
 *currentHighlightedColumn* = null;  
 } else {  
 // Убрать выделение с предыдущего столбца, если есть  
 if (*currentHighlightedColumn* !== null) {  
 for (const row of rows) {  
 row.cells[*currentHighlightedColumn*].classList.remove('highlight');  
 }  
 }  
 // Выделить новый столбец  
 for (const row of rows) {  
 row.cells[columnIndex].classList.add('highlight');  
 }  
 *currentHighlightedColumn* = columnIndex;  
 }  
}  
*document*.addEventListener('DOMContentLoaded', () => {  
 const items = *document*.querySelectorAll('.item');  
 items.forEach(item => {  
 item.addEventListener('click', () => {  
 const content = item.textContent;  
 const i = *document*.getElementById('itemDialog');  
 i.innerHTML = content;  
 *dialog*.show();  
 });  
 });  
});  
const *form* = *document*.getElementById('surveyForm');  
const *dialog* = *document*.querySelector('dialog')  
const *inputs* = *form*.querySelectorAll('input[type="text"], input[type="email"]');  
*form*.addEventListener('reset', (e) => {  
 if (confirm('Вы уверены, что хотите сбросить форму?')) {  
 *inputs*.forEach(input => {  
 input.style.backgroundColor = 'red';  
 });  
 setTimeout(() => {  
 *inputs*.forEach(input => input.style.backgroundColor = '');  
 }, 1000);  
 } else {  
 e.preventDefault();  
 *inputs*.forEach(input => {  
 input.style.backgroundColor = 'green';  
 });  
 }  
 setTimeout(() => {  
 *inputs*.forEach(input => input.style.backgroundColor = '');  
 }, 1000);  
});  
*form*.addEventListener('submit', (e) => {  
 e.preventDefault();  
 *inputs*.forEach(input => {  
 input.style.backgroundColor = 'blue';  
 });  
 setTimeout(() => {  
 *inputs*.forEach(input => input.style.backgroundColor = '');  
 const i = *document*.getElementById('itemDialog');  
 i.innerHTML = 'Данные отправлены!'  
 *dialog*.show();  
 }, 1000);  
});

**Код CSS-файла**

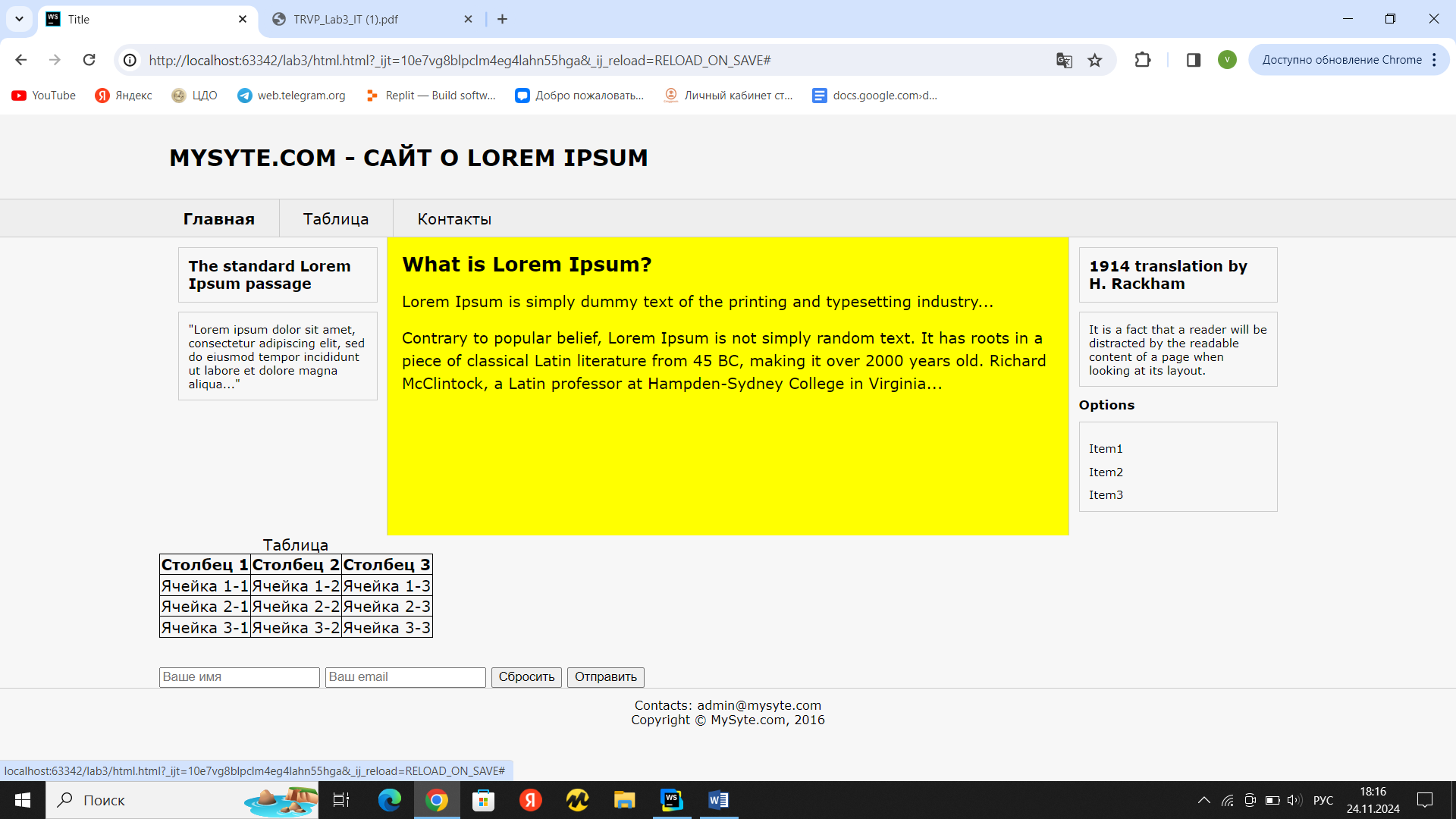
.active {

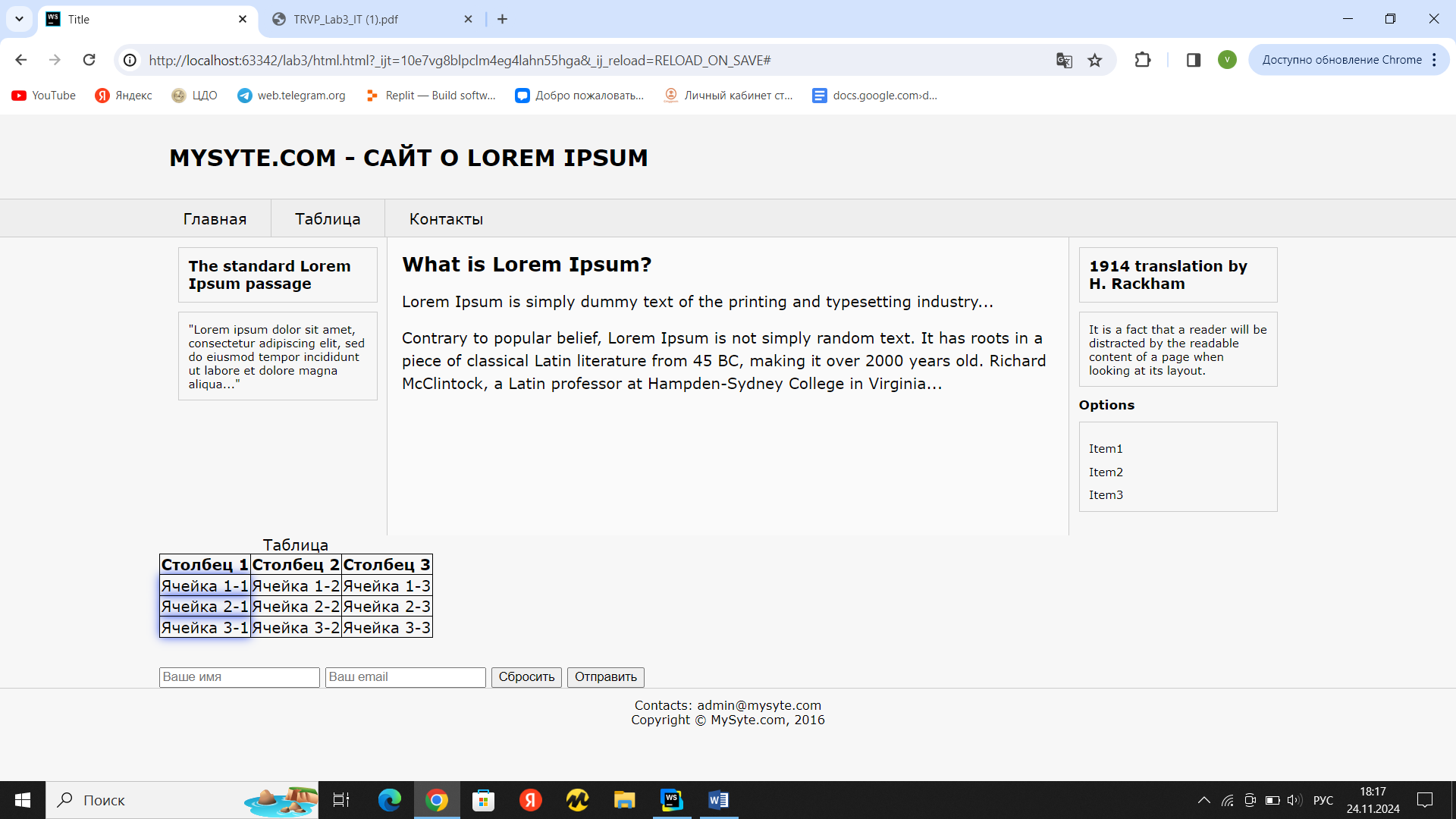
display: block;  
 background-color: yellow; /\* Цвет фона выделенного раздела \*/

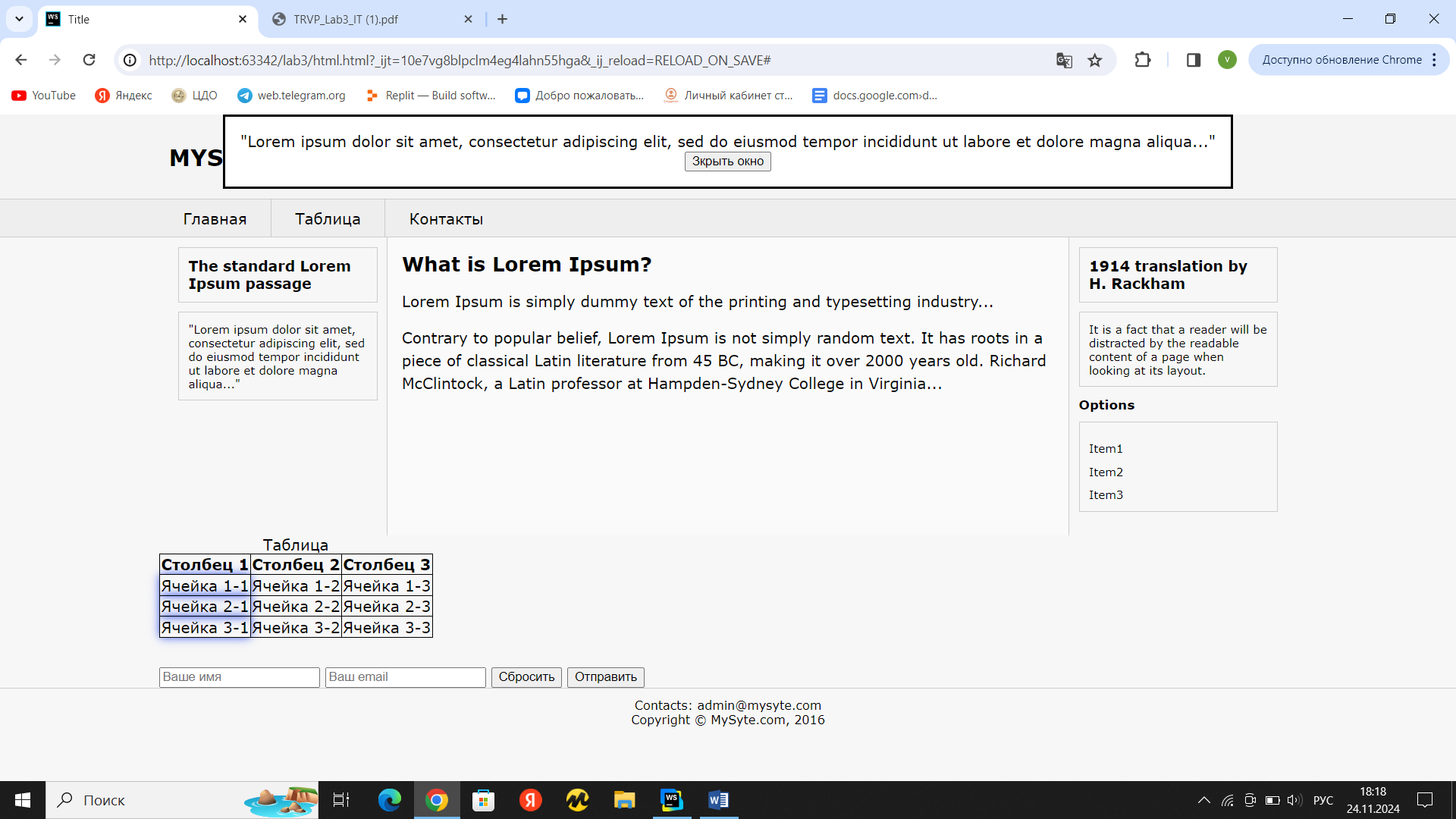
}  
.highlight {  
 box-shadow: 0 0 10px rgb(36, 71, 241); /\* Тень для выделения \*/}  
.item {  
 cursor: pointer;  
 margin: 10px 0;  
 padding: 10px;  
 border: 1px solid #ccc;  
}  
dialog{  
 text-align: center  
}

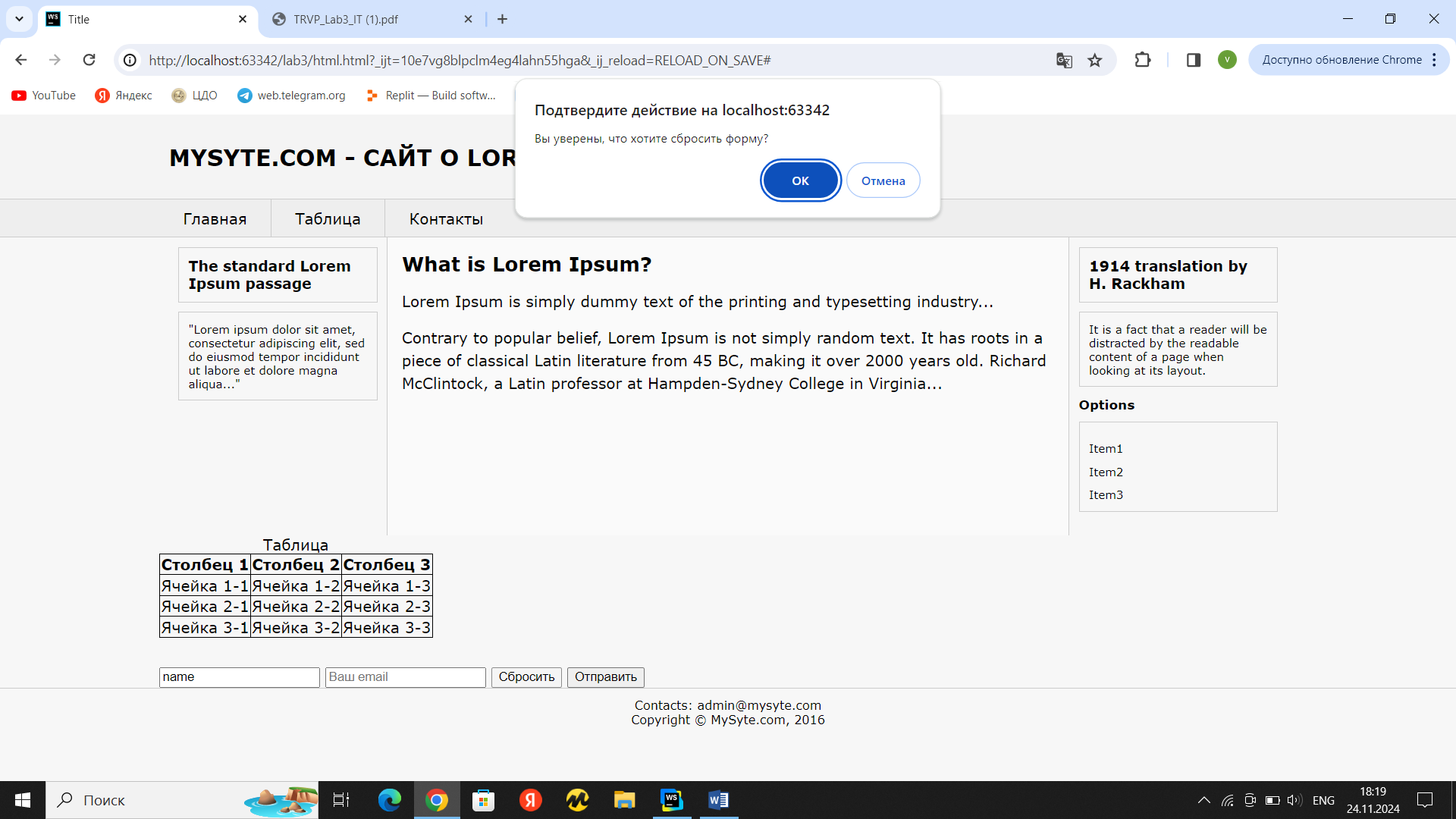
stile1.css

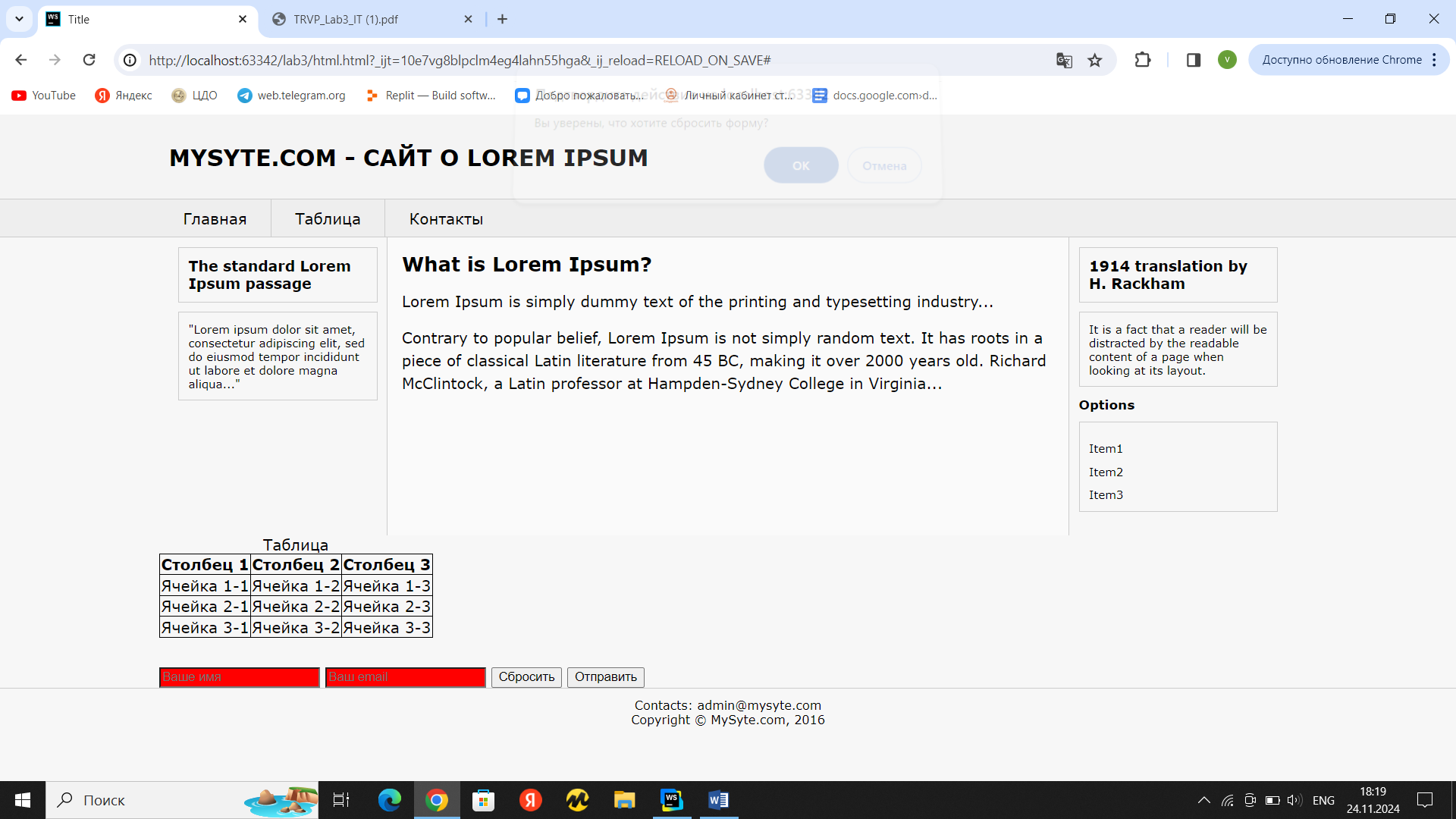
\* {  
 box-sizing: border-box;  
}  
html, body, div, span, h1, h2, h3, h4, h5, h6, p, a, ul, li{  
  
 margin: 0;  
 padding: 0;  
 border: 0;  
 font-size: 100%;  
 vertical-align: baseline;  
}  
body {  
 font-family: Verdana, Arial, sans-serif;  
 background-color: #f7f7f7;  
}  
header{  
 background-color: #f4f4f4;  
}  
header h1 {  
 font-size: 24px;  
 text-transform: uppercase;  
 font-weight: bold;  
 padding: 30px 30px 30px 10px;  
 clear: both;  
}  
nav {  
 background-color: #eee;  
 border-top: 1px solid #ccc;  
 border-bottom: 1px solid #ccc;  
}  
nav li {  
 float: left;  
 list-style: none;  
}  
nav a {  
 display: block;  
 color: black;  
 padding: 10px 25px;  
 text-decoration: none;  
 border-right: 1px solid #ccc;  
}  
nav li:last-child a {  
 border-right: none;  
}  
nav a:hover {  
 font-weight: bold;  
}  
nav:after {  
 content: " ";  
 display: table;  
 clear: both;  
}  
.wrapper{  
 background-color: #f7f7f7;  
}  
aside h2 {  
 font-size: 0.95em;  
 margin-top: 15px;  
}  
aside h3 {  
 font-size: 0.85em;  
 margin-top: 10px;  
}  
aside p, .aside li {  
 font-size: .75em;  
 margin-top: 10px;  
}  
aside li{  
 list-style-type: none;  
}  
#sidebar1 {  
 float: left;  
 width: 20%;  
 padding: 0 10px 0 20px;  
}  
#sidebar2 {  
 float: right;  
 width: 20%;  
 padding: 0 20px 0 10px;  
}  
article{  
 background-color: #fafafa;  
 border-left: 1px solid #ccc;  
 border-right: 1px solid #ccc;  
 margin-left: 20%;  
 margin-right: 20%;  
 padding: 15px;  
 width: 60%;  
}  
article:after{  
 clear:both;  
 display:table;  
 content:'';  
}  
article h2{  
 font-size: 1.3em;  
 margin-bottom:15px;  
}  
article p{  
 line-height: 150%;  
 margin-bottom: 15px;  
}  
footer{  
 border-top: 1px solid #ccc;  
 font-size: .8em;  
 text-align: center;  
 padding: 10px 10px 30px 10px;  
}  
nav ul, #header h1, .wrapper, #footer p {  
 max-width: 1200px;  
 margin: 0 auto;  
}  
.wrapper, #nav, #header, #footer{  
 min-width: 768px;  
}  
header{  
 background-color: #f4f4f4;  
}  
header h1 {  
 font-size: 24px;  
 text-transform: uppercase;  
 font-weight: bold;  
 padding: 30px 30px 30px 10px;  
 clear: both;  
}  
nav {  
 background-color: #eee;  
 border-top: 1px solid #ccc;  
 border-bottom: 1px solid #ccc;  
}  
nav li {  
 float: left;  
 list-style: none;  
}  
nav a {  
 display: block;  
 color: black;  
 padding: 10px 25px;  
 text-decoration: none;  
 border-right: 1px solid #ccc;  
}  
nav li:last-child a {  
 border-right: none;  
}  
nav a:hover {  
 font-weight: bold;  
}  
nav:after {  
 content: " ";  
 display: table;  
 clear: both;  
}  
.wrapper{  
 background-color: #f7f7f7;  
}  
aside h2 {  
 font-size: 0.95em;  
 margin-top: 15px;  
}  
aside h3 {  
 font-size: 0.85em;  
 margin-top: 10px;  
}  
aside p, .aside li {  
 font-size: .75em;  
 margin-top: 10px;  
}  
aside li{  
 list-style-type: none;  
}  
#sidebar1 {  
 float: left;  
 width: 20%;  
 padding: 0 10px 0 20px;  
}  
#sidebar2 {  
 float: right;  
 width: 20%;  
 padding: 0 20px 0 10px;  
}  
article{  
 background-color: #fafafa;  
 border-left: 1px solid #ccc;  
 border-right: 1px solid #ccc;  
 margin-left: 20%;  
 margin-right: 20%;  
 padding: 15px;  
 width: 60%;  
}  
article:after{  
 clear:both;  
 display:table;  
 content:'';  
}  
article h2{  
 font-size: 1.3em;  
 margin-bottom:15px;  
}  
article p{  
 line-height: 150%;  
 margin-bottom: 15px;  
}  
footer{  
 border-top: 1px solid #ccc;  
 font-size: .8em;  
 text-align: center;  
 padding: 10px 10px 30px 10px;  
}  
nav ul, header h1, .wrapper, #footer p {  
 max-width: 1200px;  
 margin: 0 auto;  
}  
.wrapper, nav, header, footer{  
 min-width: 768px;  
}  
table { border: 1px solid grey; border-collapse: collapse }  
th,td { border: 1px solid grey; }

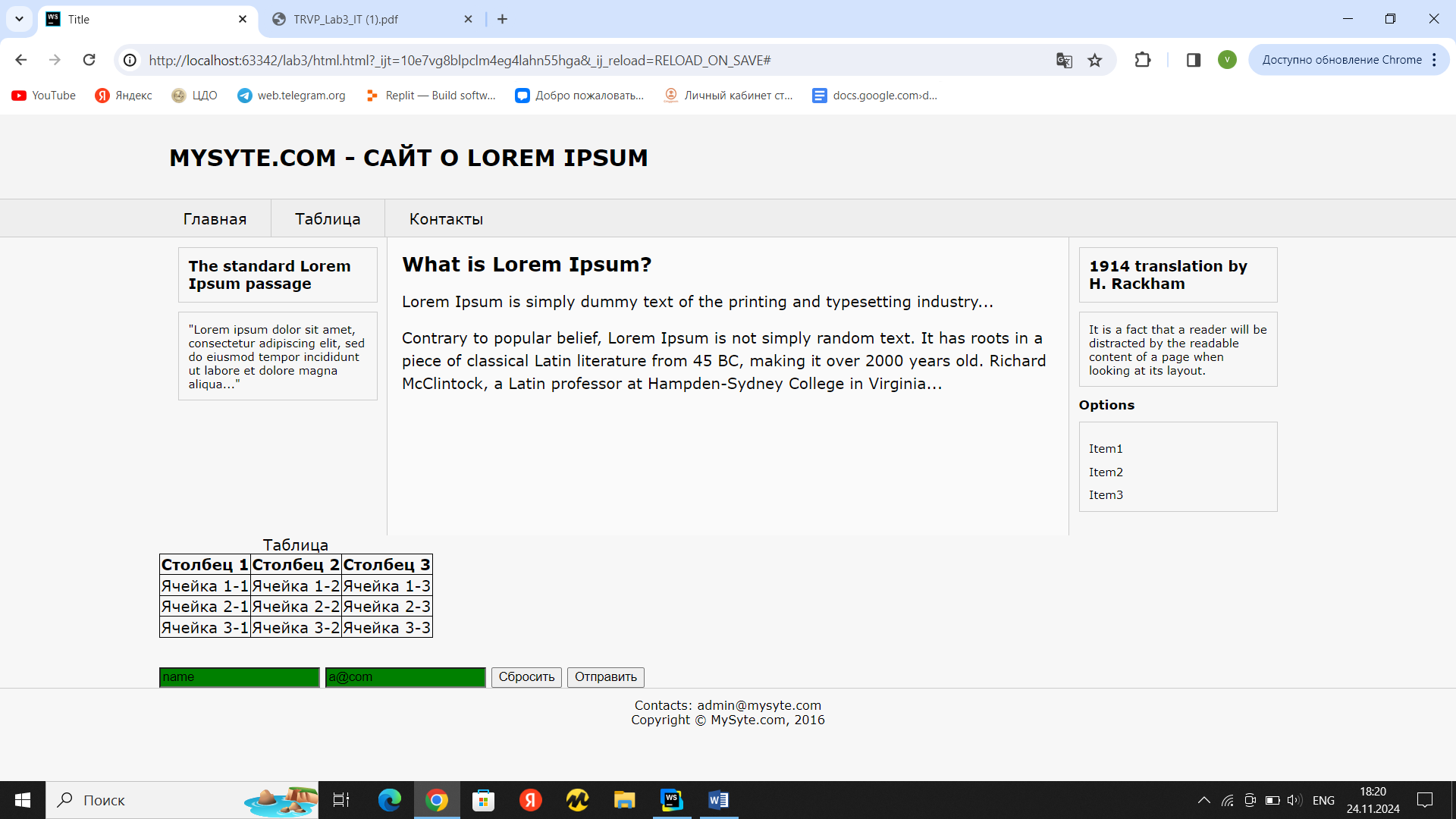


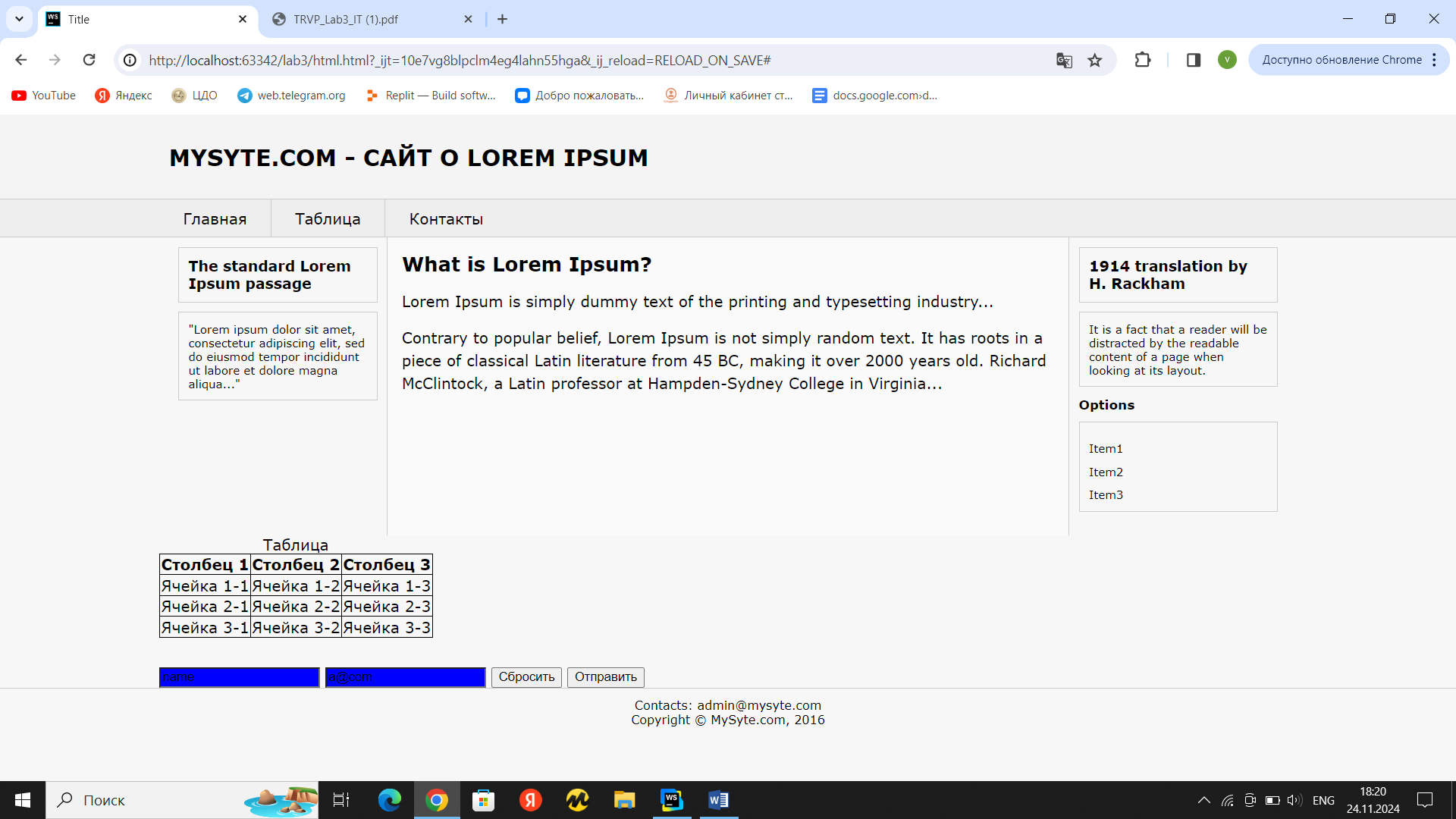


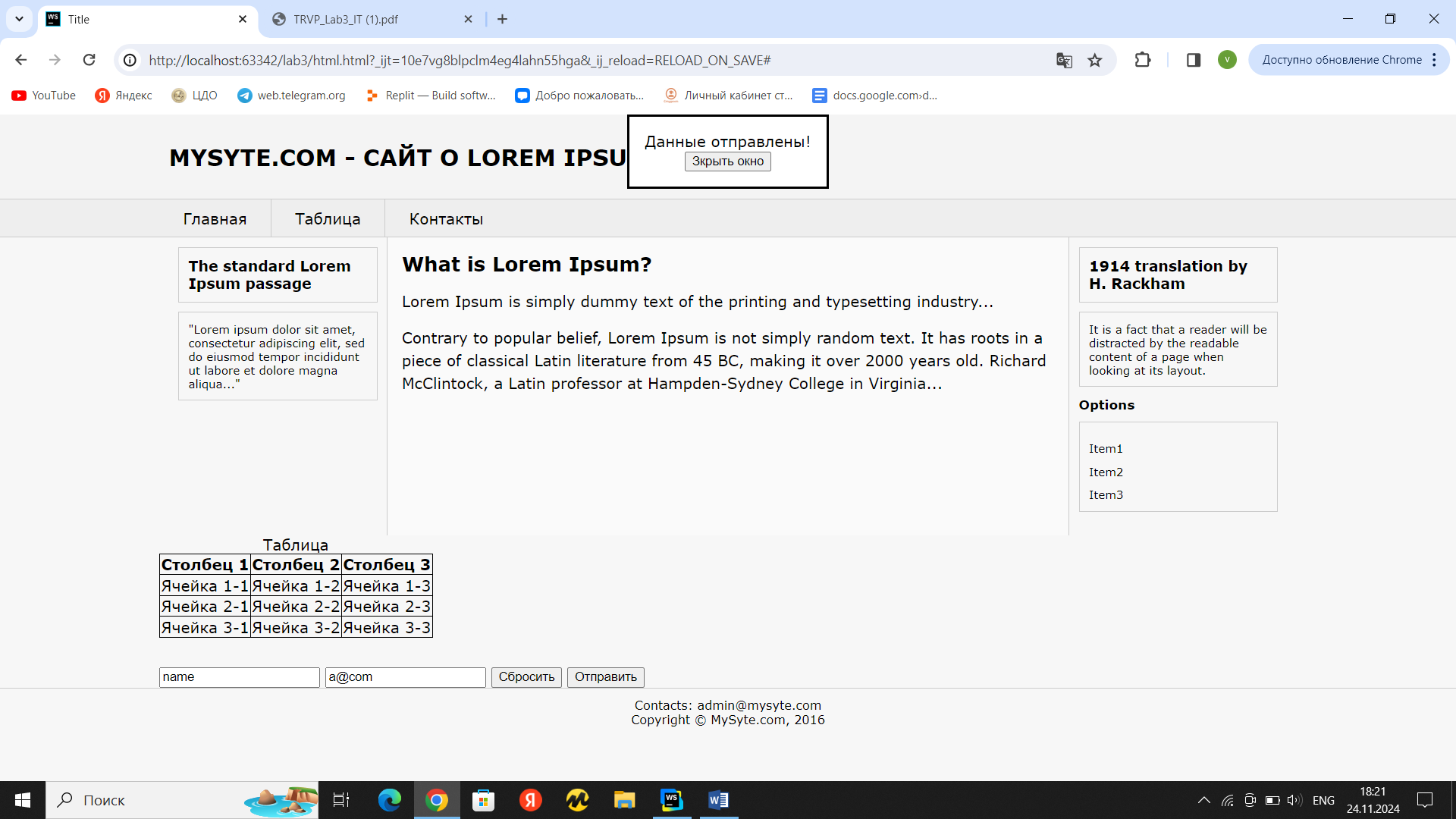












**Вывод: в ходе лабораторной работы рассмотрели** возможности языка JavaScript и модели Document Object Model (DOM) для динамической работы с содержимым веб-страницы. DOM — это программный интерфейс для HTML и XML документов, который представляет структуру документа в виде дерева узлов. Уровни DOM:

- Уровень документа: доступ к самому документу.

- Уровень элементов: работа с элементами на странице.

-Уровень узлов: взаимодействие с узлами дерева (текст, атрибуты и т.д.).

Проблемы, которые могут возникать при использовании DOM:

- Низкая производительность при больших DOM-деревьях.

- Сложности с кроссбраузерной совместимостью.

Решения этих проблем:

- Минимизация изменений в DOM.

- Использование виртуального DOM