
КАФЕДРА

ОТЧЕТ
ЗАЩИЩЕН С ОЦЕНКОЙ
РУКОВОДИТЕЛЬ

должность, уч. степень, звание

подпись, дата

инициалы, фамилия

Отчет о лабораторной работе №7
Асинхронное клиент-серверное взаимодействие

с использованием Ajax и JSON

По дисциплине: Web-технологии

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. №

подпись, дата

инициалы, фамилия

Санкт-Петербург 2024

Содержание отчета:

| | |
|---|----|
| Цель работы: | 3 |
| Вариант задания..... | 3 |
| Базовое задание: | 3 |
| Расширенное задание: | 4 |
| Адаптированная формулировка задания | 4 |
| Структура таблиц БД | 5 |
| Скриншоты страниц сайта и таблиц базы данных..... | 6 |
| Код javascript | 14 |
| php код | 15 |
| HTML код | 16 |

Цель работы:

изучение и применение на практике технологии AJAX

Вариант задания

| № варианта | Тип запроса |
|---------------|----------------|
| 6 | Post |

| № варианта | Тема |
|---------------|-------------------|
| 18 | Нейронные сети |

Базовое задание:

Разработать AJAX-приложение для доступа к базе данных, подготовленной в предыдущей лабораторной работе. Вариант задания и тип запроса указаны в таблице 1. Формулировку варианта адаптировать под свою базу данных. В сценарии javascript запрещается использование библиотек. Новая страница должна быть встроена в сайт, подготовленный в предыдущих разделах.

Вариант:

6. Оповещение о появлении новых сообщений (на форумах). При получении нового сообщения на сайте «всплывает» информационное окно с первыми 30-ю символами.

Расширенное задание:

2 пункта:

- 1.. использовать json
- 2.. проверить работу в разных браузерах. В сценарии на javascript добавить вывод на страницу названия объекта, выполняющего асинхронный обмен, и название браузера. В отчете привести доказательства этой работы.
- 3.. использовать библиотеку jQuery и добавить еще один вариант применения AJAX

Адаптированная формулировка задания

Оповещение о добавлении новых алгоритмов или проектов. Сайт будет включать функцию асинхронного оповещения пользователя о добавлении новых алгоритмов разработки нейронных сетей или новых проектов, связанных с ними. Когда кто-то добавляет новый алгоритм или проект на сайт, другие пользователи, находящиеся на сайте в этот момент, получают оповещение. Это оповещение будет в виде всплывающего информационного окна, которое появляется на экране без необходимости обновления страницы. Окно будет содержать краткое описание добавленной записи – например, название нового алгоритма или проекта и первые 30 символов его описания. Для реализации этой функции используется технология Ajax с запросами типа POST, чтобы асинхронно получать данные из базы данных и отображать их на сайте.

Структура таблиц БД

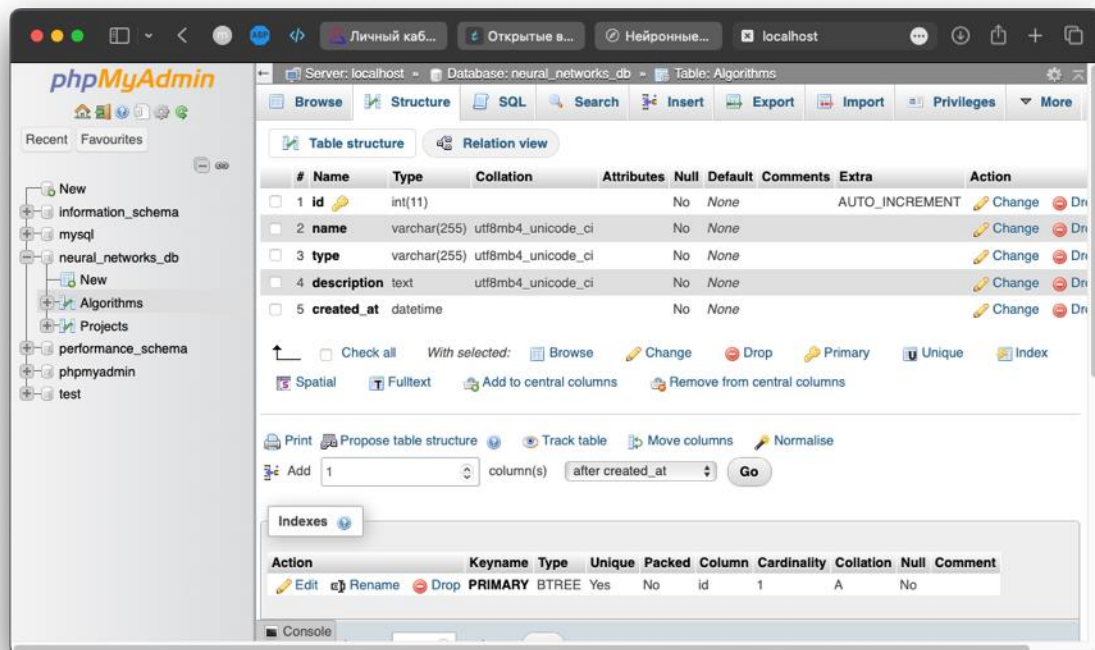


Рисунок 1 – структура таблицы алгоритмов

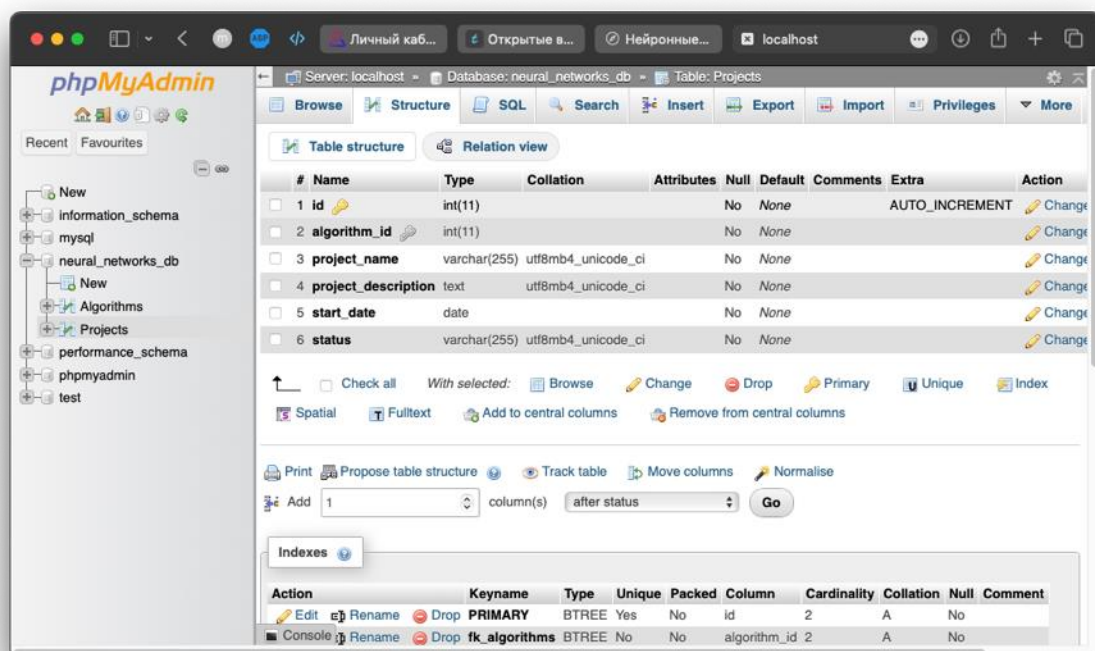


Рисунок 2 – структура таблицы проектов

Скриншоты страниц сайта и таблиц базы данных

Базовое задание:

The screenshot shows a web browser window with the address bar displaying 'localhost'. The main content area is a white card with a green header 'Neural Networks Algorithms and Projects'. Below the header, there are three main sections: 'Add New Algorithm', 'Add New Project', and 'Algorithms'. The 'Add New Algorithm' section has three input fields: 'Name:' (containing '1'), 'Type:' (containing '1'), and 'Description:' (containing '1'). Below these fields is a green button labeled 'Add Algorithm'. The 'Add New Project' section has five input fields: 'Project Name:' (empty), 'Description:' (empty), 'Start Date:' (containing '21/03/2024'), 'Status:' (empty), and 'Algorithm:' (a dropdown menu showing 'No algorithms available'). Below these fields is a green button labeled 'Add Project'. The 'Algorithms' section has the text 'No algorithms found.' Below this, there is a section for 'Projects' with the text 'No projects found.' At the bottom, there is a section for 'Create Matrix' with two input fields: 'Rows:' (containing '1') and 'Columns:' (containing '1'). Below these fields is a green button labeled 'Create Matrix'.

Neural Networks Algorithms and Projects

Add New Algorithm

Name: 1

Type: 1

Description: 1

Add Algorithm

Add New Project

Project Name:

Description:

Start Date: 21/03/2024

Status:

Algorithm: No algorithms available

Add Project

Algorithms

No algorithms found.

Projects

No projects found.

Create Matrix

Rows: 1

Columns: 1

Create Matrix

Рисунок 3 – добавляем тестовые данные

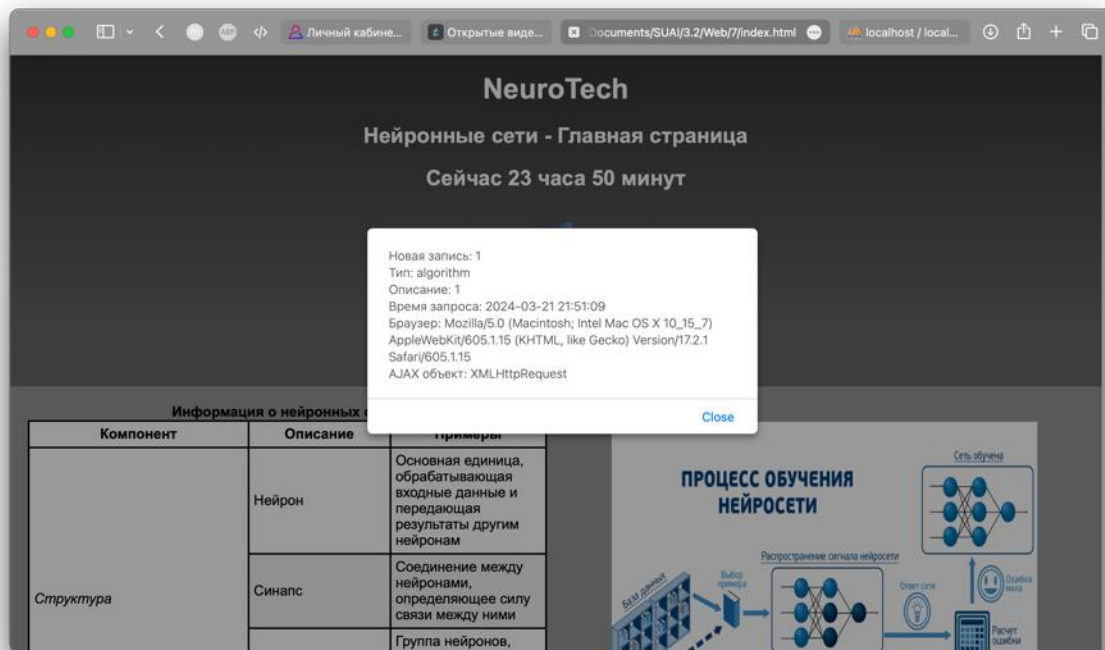


Рисунок 4 – уведомление на сайте

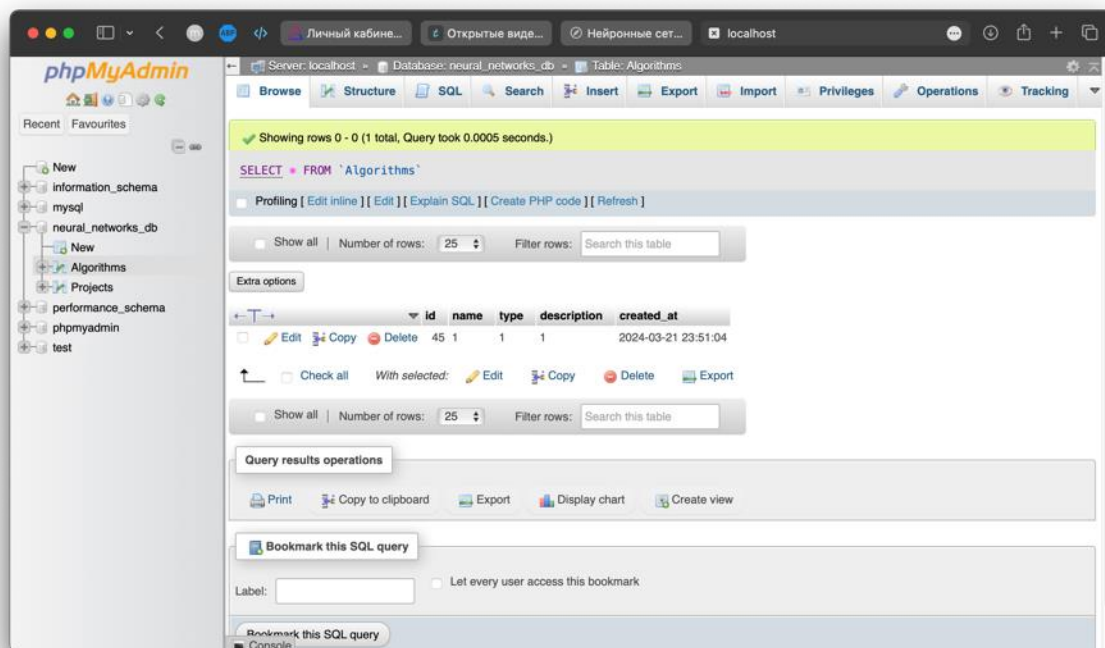


Рисунок 5 – обновление в БД

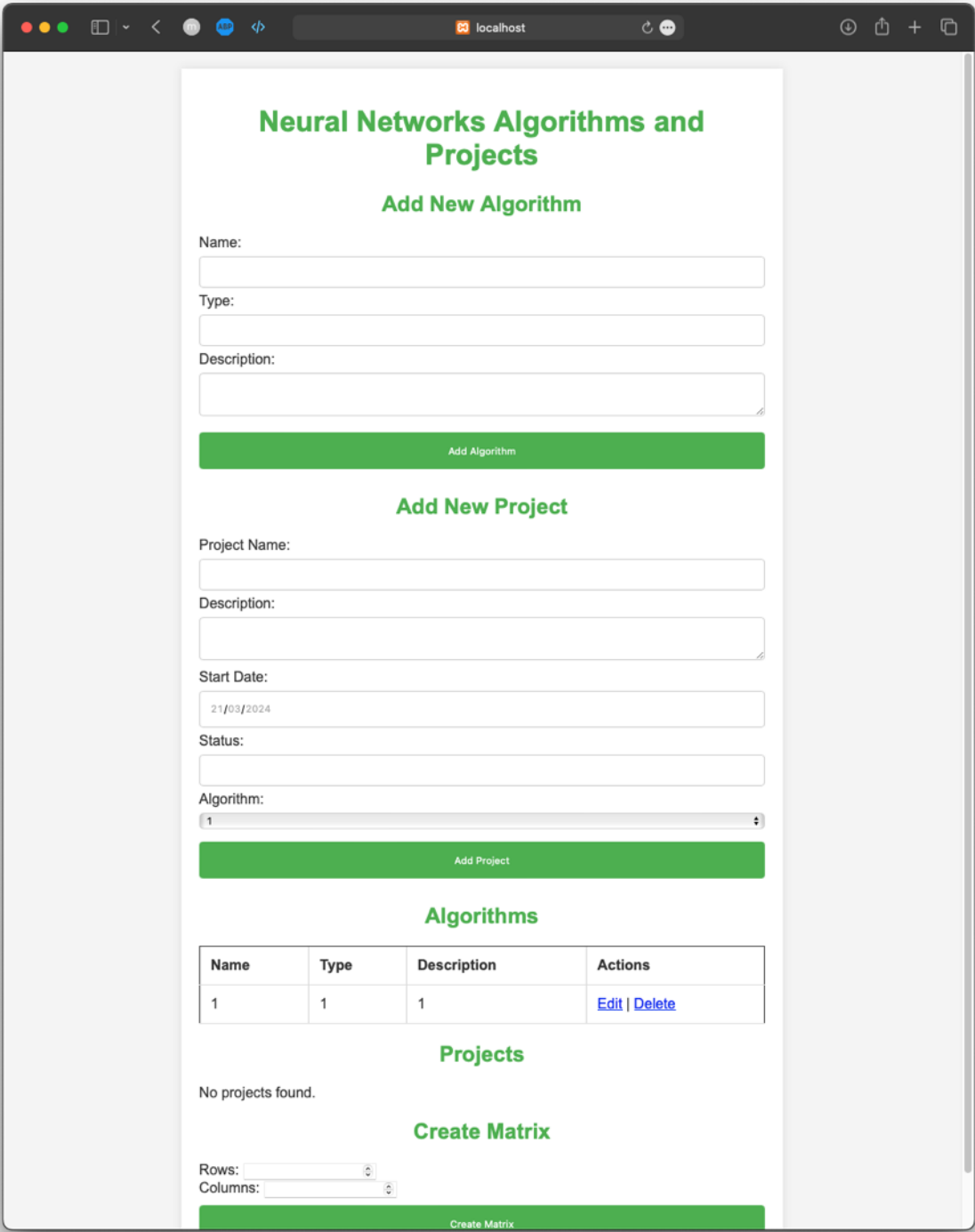


Рисунок 6 – обновление в форме

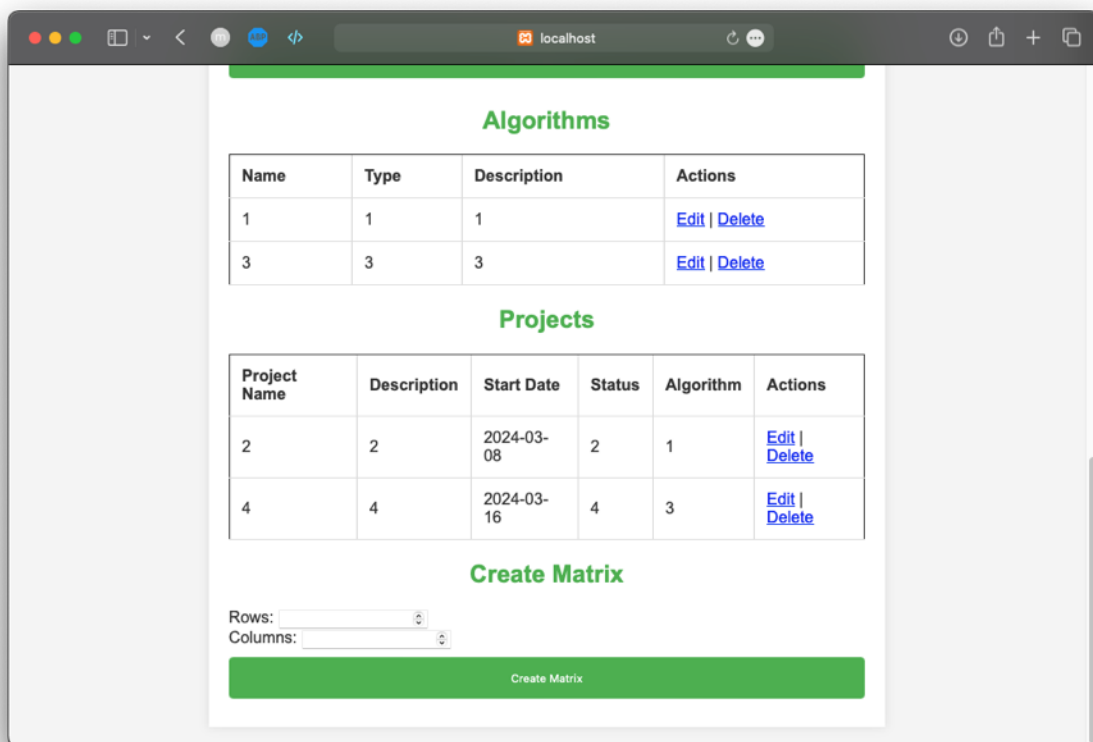


Рисунок 7 – дополнительные тестовые данные

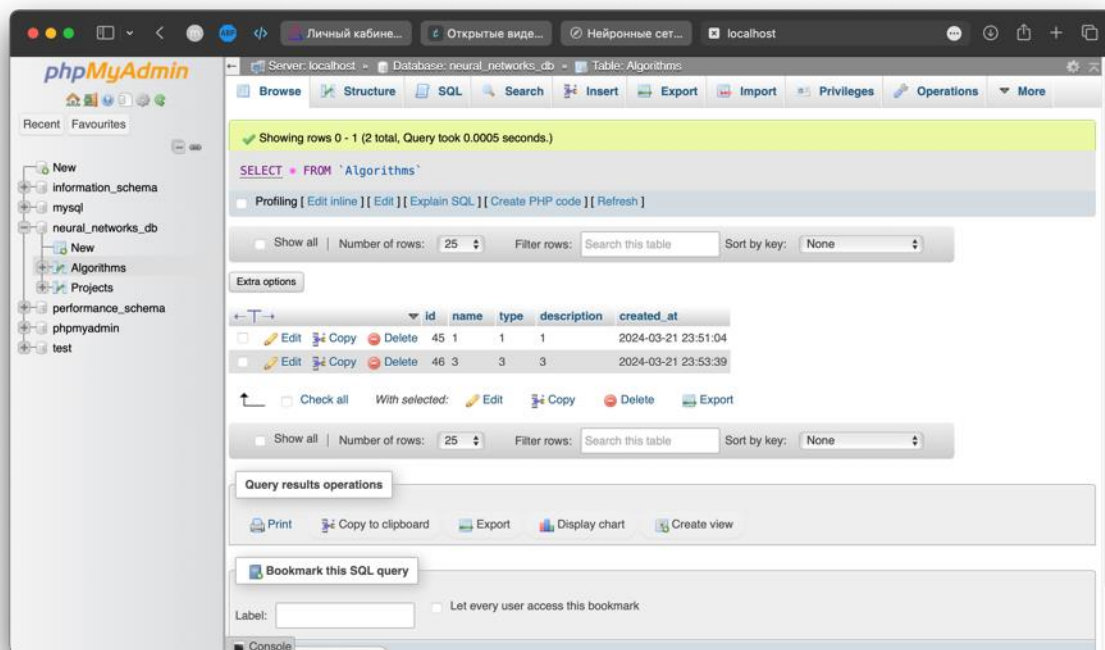


Рисунок 8 – таблица алгоритмов

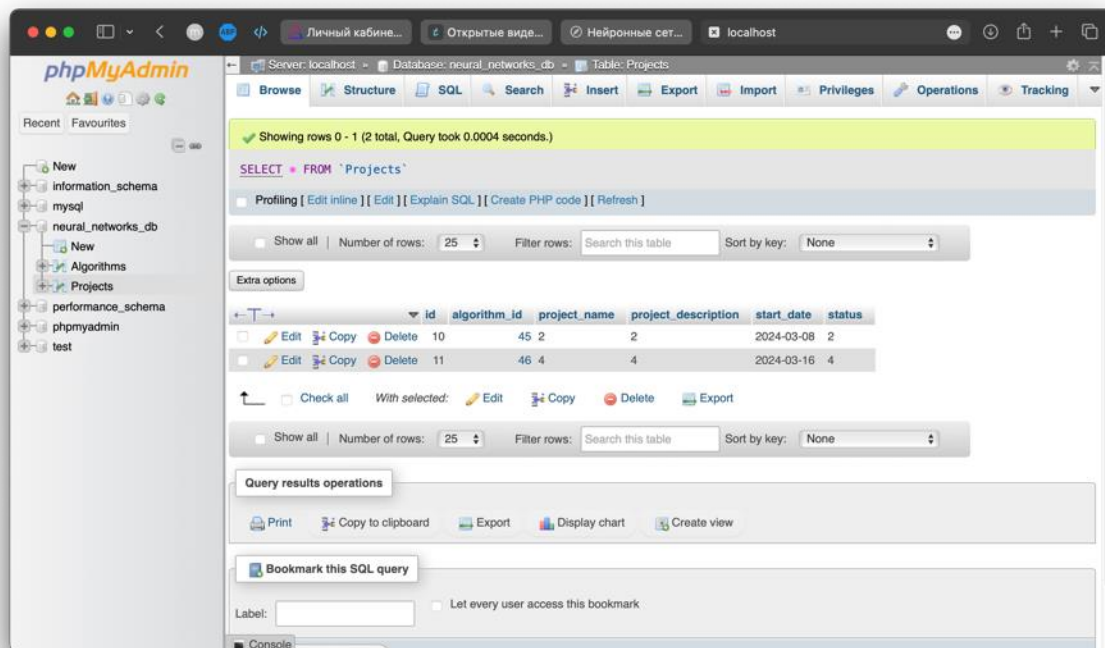


Рисунок 9 – таблица проектов

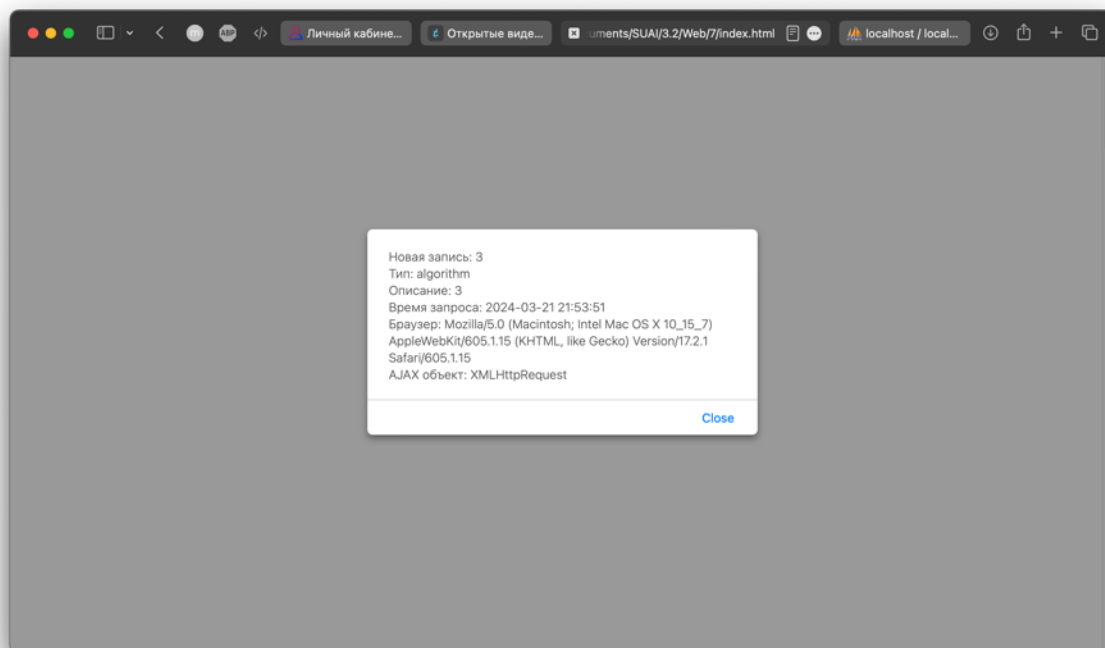


Рисунок 10

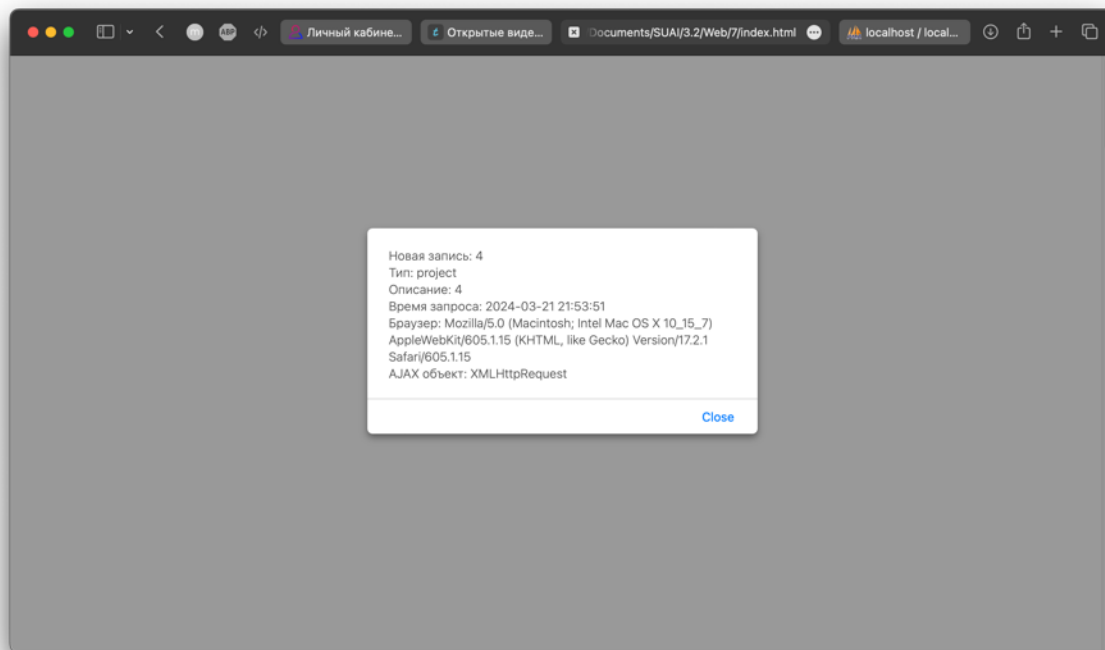


Рисунок 11

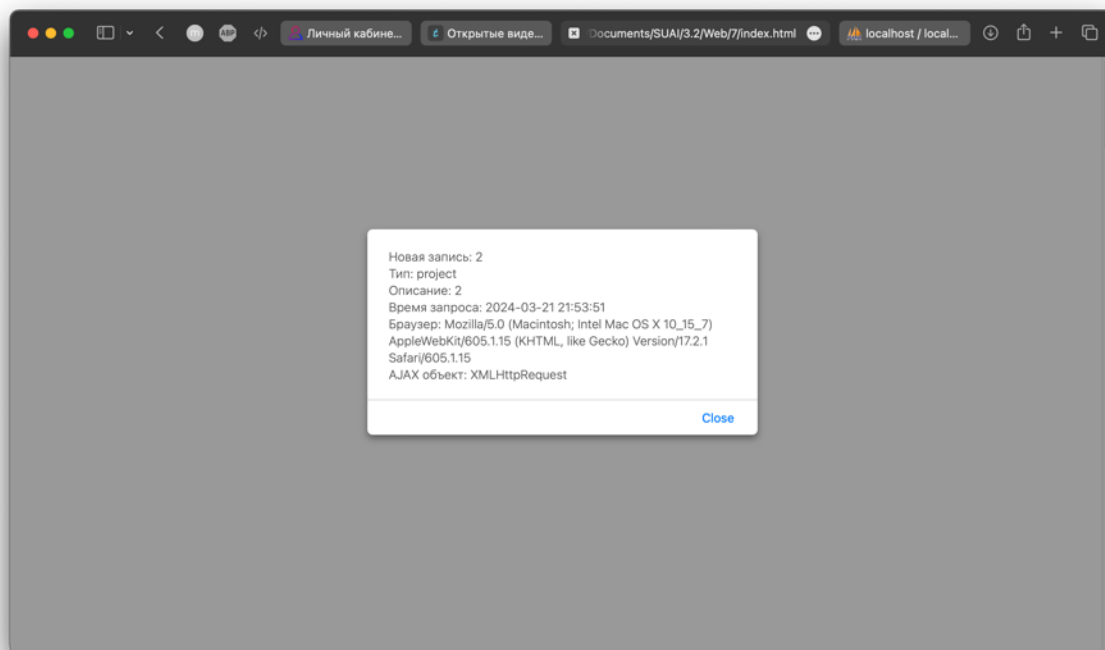


Рисунок 12

Рисунки 10-12 – примеры уведомлений

Расширенное задание

1.. использовать json

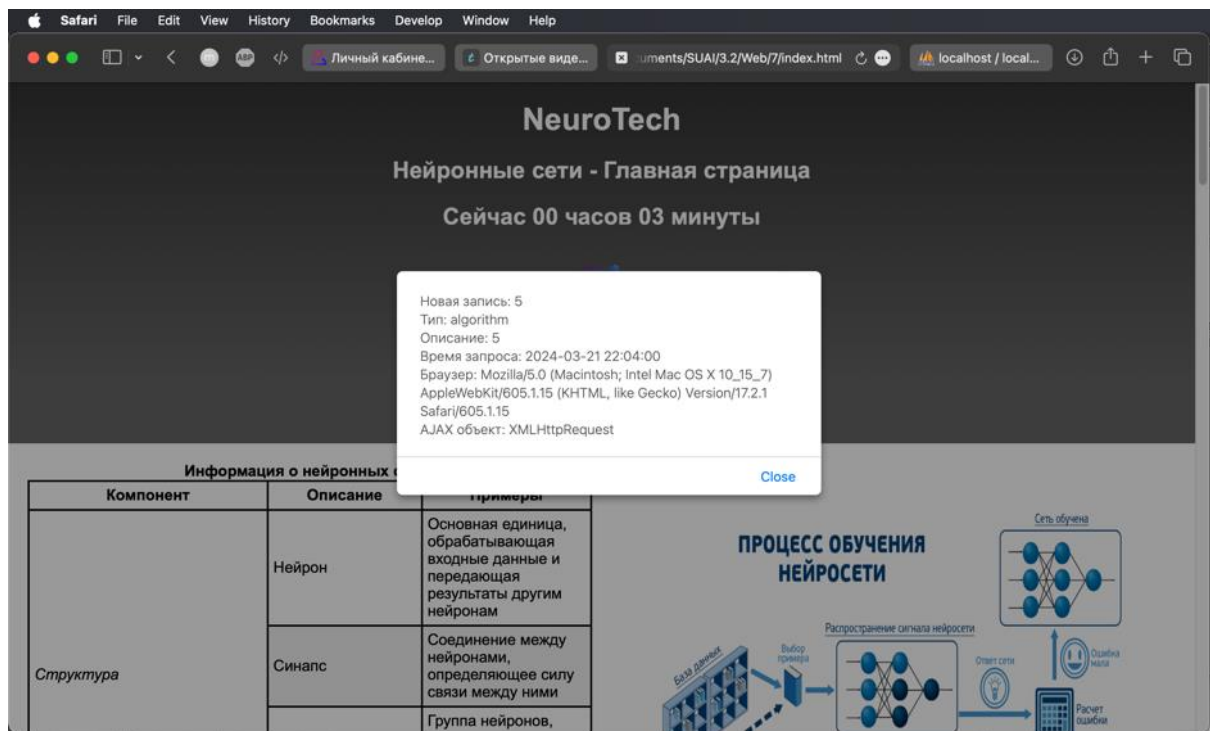


Рисунок 13 – использование JSON, используем дополнительные поля

2.. проверить работу в разных браузерах. В сценарии на javascript добавить вывод на страницу названия объекта, выполняющего асинхронный обмен, и название браузера. В отчете привести доказательства этой работы.

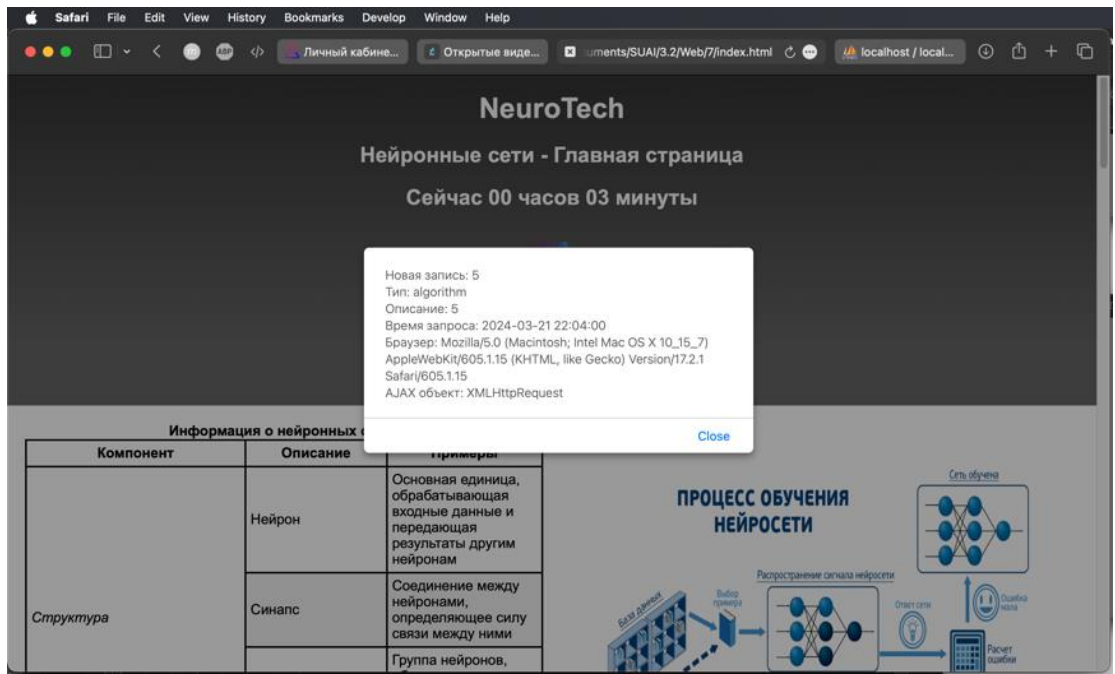


Рисунок 14 - Safari

Слева сверху на Рисунке 14 выведено меню с активным приложением – это Браузер Сафари

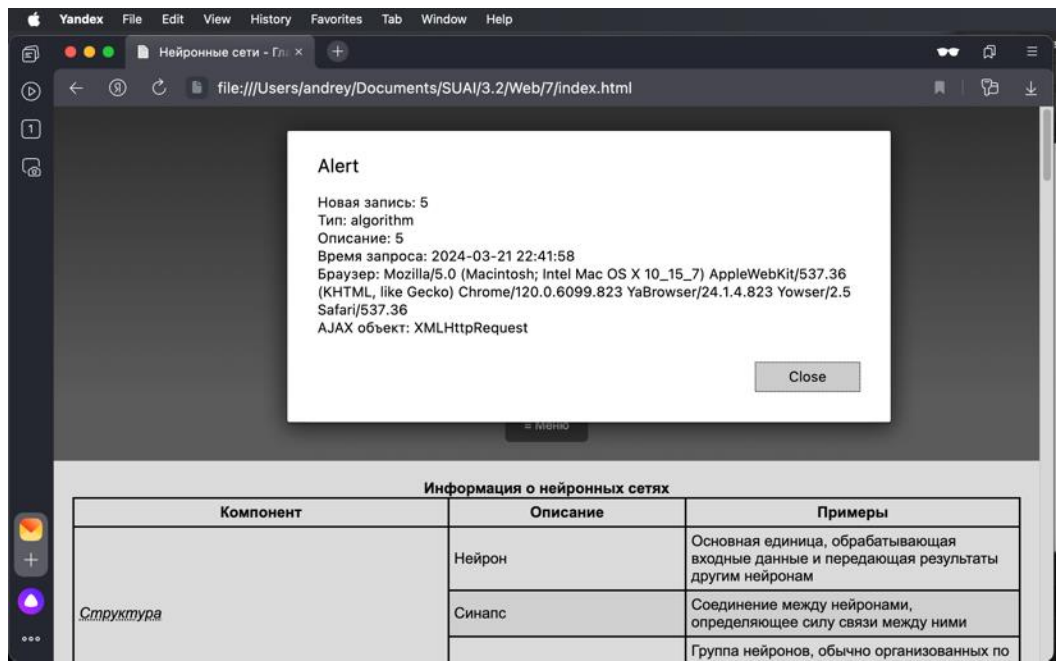


Рисунок 15

Слева сверху на Рисунке 15 выведено меню с активным приложением – это Яндекс Браузер

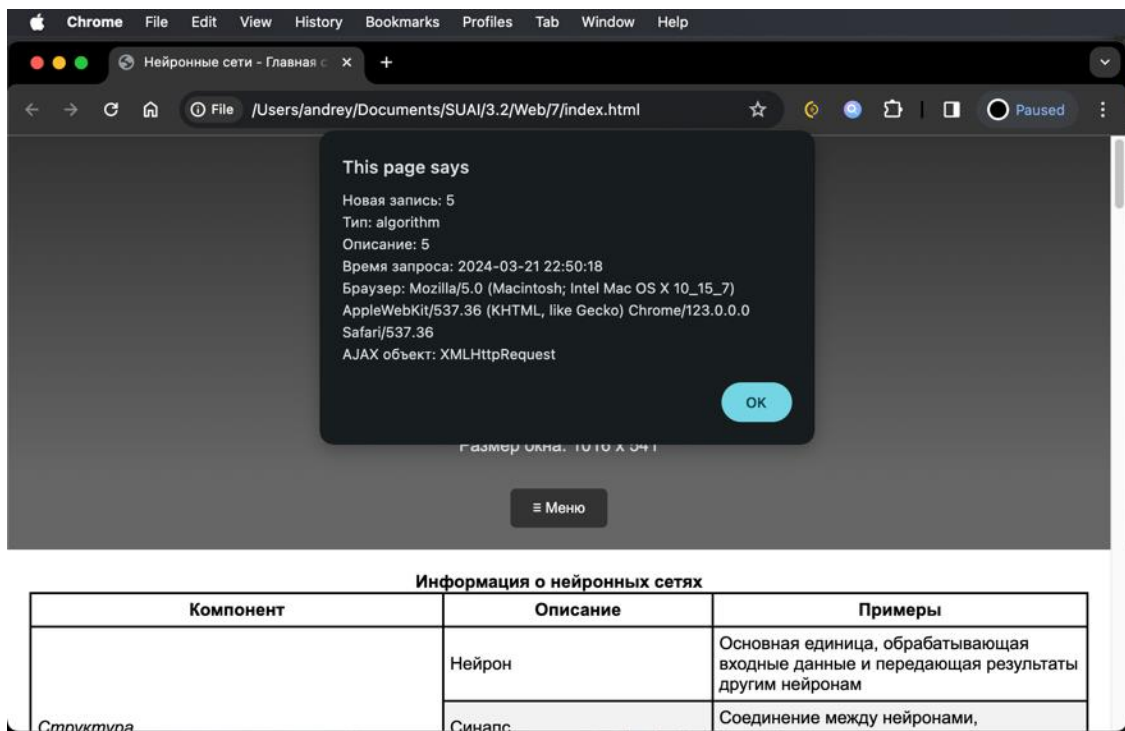


Рисунок 16

Слева сверху на Рисунке 14 выведено меню с активным приложением – это Гугл Хром

Код javascript

```
<!-- ЛР 7 -->
```

```
<script>
```

```
    let shownEntries = new Set(); // Множество для хранения уже показанных записей
```

```
    // Функция для проверки наличия новых записей
```

```
    function checkForNewEntries() {
```

```
        var xhr = new XMLHttpRequest();
```

```
        var browserInfo = navigator.userAgent; // Получаем информацию о браузере пользователя
```

```
        var ajaxObject = "XMLHttpRequest"; // Объект, выполняющий асинхронный обмен
```

```
        xhr.onreadystatechange = function() {
```

```
            if (xhr.readyState === 4 && xhr.status === 200) {
```

```
                var response = JSON.parse(xhr.responseText);
```

```
                if (response.status === "success") {
```

```
                    response.data.forEach(function(entry) {
```

```
                        if (!shownEntries.has(entry.id)) {
```

```
                            // Добавляем ID новой записи в множество показанных записей
```

```
                            shownEntries.add(entry.id);
```

```
                        var message = "Новая запись: " + entry.name + "\n" +
```

```
                            "Тип: " + entry.type + "\n" +
```

```

        "Описание: " + entry.description.substring(0, 30) + "\n" +
        "Время запроса: " + response.timestamp + "\n" +
        "Браузер: " + browserInfo + "\n" +
        "AJAX объект: " + ajaxObject;
    alert(message);
    }
    });
    }
    }
    };
    xhr.open("POST", "http://localhost/check_new_entries.php", true);
    xhr.send();
}

// Периодическая проверка новых записей каждые 5 секунд
setInterval(checkForNewEntries, 5000);
</script>

```

php код

```

<?php
header('Access-Control-Allow-Origin: *'); // Разрешает доступ для всех доменов

$host = 'localhost';
$user = 'root';
$pass = '';
$db = 'neural_networks_db';

// Создаем подключение
$conn = new mysqli($host, $user, $pass, $db);

// Проверяем подключение
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

// Запрос для получения последних записей алгоритмов и проектов
$sql = "(SELECT 'algorithm' as type, name, description, id FROM algorithms ORDER BY id
DESC LIMIT 5)
UNION
(SELECT 'project' as type, project_name as name, project_description as description, id
FROM projects ORDER BY id DESC LIMIT 5)
ORDER BY id DESC LIMIT 5";

$result = $conn->query($sql);

$new_entries = array();

if ($result->num_rows > 0) {
    // Добавляем каждую новую запись в массив
    while($row = $result->fetch_assoc()) {
        array_push($new_entries, $row);
    }
}

```

```

    }
}

// Возвращаем результат в формате JSON
$response = array(
    "status" => "success",
    "timestamp" => date("Y-m-d H:i:s"),
    "data" => $new_entries
);

echo json_encode($response);

// Закрываем подключение
$conn->close();
?>

```

HTML код

Index.html

```

<!DOCTYPE html>
<html lang="ru">

<head>
    <!-- Секция, содержащая метаданные и ссылки на внешние ресурсы -->
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta name="keywords" content="нейронные сети, искусственный интеллект,
машинное обучение, распознавание образов, обработка естественного языка">
    <meta name="description" content="На странице рассматривается тема нейронных
сетей, их применение в современном мире, а также основные типы и структура.">
    <meta name="author" content="Захаров Андрей 4133K">
    <meta name="language" content="ru">
    <title>Нейронные сети - Главная страница</title>
    <link rel="stylesheet" href="index.css">
    <!-- -->

    <style>
    .header {

```



```

/* Стили для заголовка */
position: relative;
background-color: #333;
color: #fff;
text-align: center;
padding: 20px 0;
}

body {
    transition: background-color 0.5s; /* Плавное изменение цвета фона */
}

.logo {
display: block;
margin: 0 auto;
width: 150px; /* Начальный размер логотипа */
}

</style>

</head>
<body>
<!-- Тело документа, содержит видимое содержание страницы -->

<!-- Блок с заголовком страницы -->
<div class="header" name="page-header">
    <h1>NeuroTech</h1>
    <h2>Нейронные сети - Главная страница</h2>
    <h2 id="time"></h2>
    
    <div id="windowSizeInfo">
        Размер окна: <span id="windowWidth"></span> x <span id="windowHeight"></span>
    </div>

</div>

```

```

<!-- Блок с меню -->
<div class="menu">
    <button class="menu-btn" onclick="toggleMenu()" style="background-color: #333; color:
white;">≡ Меню</button>
    <div id="menu-items" class="menu-items">
        <a href="index.html">Главная страница</a>
        <a href="page2.html">Вторая страница</a>
        <a href="sources.html">Использованные источники</a>
        <a href="forma.html">Анкета</a> <!-- Новый пункт меню -->
        <a href="http://localhost/neural_networks.php">Neural Networks</a> <!-- Новый пункт
меню -->
    </div>
</div>

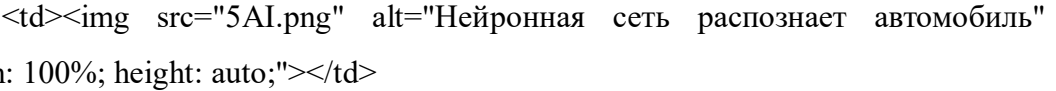
```

```

<!-- Главный контент страницы -->
<div class="content">
    <div class="left-column"> <!-- Левая колонка контента -->
        <!-- Таблица с информацией о нейронных сетях -->
        <table name="neural-network-table" border="1">
            <caption><strong>Информация о нейронных сетях</strong></caption>
            <thead>
                <tr>
                    <th>Компонент</th>
                    <th>Описание</th>
                    <th>Примеры</th>
                </tr>
            </thead>
            <tbody>
                <tr>
                    <td rowspan="3"><em><abbr title="Организация нейронов и их
связей">Структура</abbr></em></td>
                    <td>Нейрон</td>
                    <td>Основная единица, обрабатывающая входные данные и передающая
результаты другим нейронам</td>

```

| | |
|--|---|
| | |
| | Синапс |
| | Соединение между нейронами, определяющее силу связи между ними |
| | |
| | Слой |
| | Группа нейронов, обычно организованных по уровням, которые выполняют определенные функции |
| | |
| | <div> <div>Разновидности архитектур и функциональных характеристик</div> <div>Типы</div> </div> |
| | Однонаправленные |
| | Нейронные сети, в которых данные передаются только в одном направлении, от входных к выходным |
| | |
| | Рекуррентные |
| | Сети с циклическими связями, позволяющими сохранять информацию о предыдущих состояниях |
| | |
| | Свёрточные |
| | Специализированные для обработки данных с пространственной структурой, такими как изображения |
| | |
| | <div> <div>Области использования в технологии и науке</div> <div>Применение</div> </div> |
| | Распознавание образов |
| | Используется для автоматического распознавания и классификации объектов и образов |
| | |
| | |

| | |
|--|---|
| Обработка естественного языка | Применяется для анализа, понимания и генерации текста естественного языка |
| Прогнозирование временных рядов | Используется для прогнозирования будущих значений на основе последовательности данных |
|  | |
| Распознавание объектов с использованием нейронных сетей | Нейронная сеть анализирует изображение и определяет на нем автомобиль |

| | | |
|--|--|--|
| Таблица о нейронных сетях с дополнительной информацией о структуре, типах и применении | | |
|--|--|--|

Компонент

Описание

Примеры

Добавить строку

```

<div>
  <input type="text" id="listItemInput" placeholder="Текст элемента списка">
  <button onclick="addItemToList()">Добавить элемент</button>
  <button onclick="removeLastItem()">Удалить последний элемент</button>
</div>
<ul id="dynamicList">
  <!-- Список, в который будут добавляться элементы -->
</ul>

```

```

<hr>

```

```

<!-- Начало упорядоченного списка с тремя основными разделами -->
<ol type="1" start="1">
  <li>Структура нейронной сети
    <!-- Вложенный упорядоченный список-->
    <ol type="o">
      <li>Нейрон</li>
      <li>Синапс</li>
      <li>Слой</li>
    </ol>
  </li>
  <li>Типы нейронных сетей
    <!-- Вложенный упорядоченный список-->
    <ol type="A">
      <li>Однонаправленные</li>
      <li>Рекуррентные</li>
      <li>Свёрточные</li>
    </ol>
  </li>
  <li>Применение нейронных сетей
    <!-- Вложенный упорядоченный список-->
    <ol type="I">

```

- Распознавание образов
 - Обработка естественного языка
 - Прогнозирование временных рядов
-
-
-
- <hr>

<p>Идея и ранние этапы развития нейронных сетей</p>

<p>Идея создания <dfn>нейронных сетей</dfn> возникла ещё в середине 20 века, вдохновленная работами ученых по изучению мозга и его функций. Первые концепции нейронных сетей были основаны на <abbr title="биологических принципах работы нейронов и сетей нервных клеток в мозге живых существ">биологических принципах</abbr> работы нейронов и сетей нервных клеток в мозге живых существ.</p>

<p>Одним из первых важных этапов в развитии нейронных сетей было создание модели <cite>искусственного нейрона</cite>, предложенной Уорреном Маккаллохом и Уолтером Питтсом в 1943 году. Их модель, названная "моделью Маккаллоха-Питтса", была первым шагом к созданию искусственных нейронных сетей. Этот искусственный нейрон был представлен как <var>математическая модель</var> биологического нейрона, способного принимать входные сигналы, обрабатывать их и выдавать выходной сигнал.</p>

<p>Дальнейшие исследования привели к разработке более сложных моделей и алгоритмов обучения, включая <mark>перцептрон Розенблатта</mark> в 1957 году, который стал первым примером обучаемой нейронной сети. Перцептрон имел возможность обучаться на примерах и корректировать свои веса, чтобы правильно классифицировать входные данные.</p>

<p>В конце 1960-х и начале 1970-х годов нейронные сети столкнулись с некоторыми ограничениями, такими как <q>невозможность обучения сложных функций</q> и проблемы, связанные с "затуханием градиента" при обучении глубоких сетей. Это привело к затишью в развитии нейронных сетей, из-за чего многие исследователи обратили своё внимание на другие методы машинного обучения.</p>

<p>Развитие нейронных сетей в современную эпоху</p>

<p>После затишья в развитии нейронных сетей в конце 20 века интерес к ним возрос снова благодаря нескольким ключевым факторам. Одним из них был рост

доступности данных и вычислительной мощности. С появлением интернета и цифровых технологий объем данных, доступных для обучения нейронных сетей, резко увеличился. Это позволило использовать более сложные модели и алгоритмы для решения различных задач, таких как *распознавание образов*, *обработка естественного языка*, *медицинская диагностика* и многое другое.

Вторым ключевым фактором был прорыв в области алгоритмов глубокого обучения, который произошел в начале 2010-х годов. Алгоритм глубокого обучения, известный как *глубокое обучение на основе нейронных сетей* или *глубокое обучение с обратным распространением ошибки*, смог решить проблему *затухания градиента*, что позволило эффективно обучать нейронные сети с большим количеством слоев (*глубокие нейронные сети*). Этот прорыв привел к бурному развитию области и применению нейронных сетей в самых разных областях.

С появлением графических процессоров (*GPU*), которые обладают параллельными вычислительными возможностями, обучение глубоких нейронных сетей стало более эффективным и быстрым. *GPU* позволяют выполнять параллельные вычисления на больших объемах данных, что идеально подходит для обработки и обучения нейронных сетей.

Важным моментом в развитии нейронных сетей было также улучшение архитектур и методов оптимизации. Сети с различными архитектурами, такими как *сверточные нейронные сети* (*CNN*) для обработки изображений и *рекуррентные нейронные сети* (*RNN*) для работы с последовательными данными, стали стандартом в обработке различных типов данных.

Кроме того, появились новые методы оптимизации, такие как *стохастический градиентный спуск с моментом* (*SGD with momentum*), *адам* (*Adam*) и другие, которые помогли ускорить и улучшить процесс обучения нейронных сетей, сделав его более стабильным и эффективным.

Таким образом, современное развитие нейронных сетей отличается от их ранних этапов значительными прорывами в области алгоритмов, архитектур и

вычислительных ресурсов, что привело к широкому применению этой технологии в самых разных областях науки и промышленности.</p>

<p>Перспективы и вызовы в эволюции нейронных сетей</p>

<p>Сегодня нейронные сети играют ключевую роль в современной технологической революции, принося новые возможности в области искусственного интеллекта, анализа данных и автоматизации процессов. Однако перед этой технологией стоят и вызовы и перспективы.</p>

<p>Одним из вызовов является обеспечение безопасности и надежности нейронных сетей. С увеличением их сложности и применения в критических областях, таких как медицина и автономные транспортные средства, важно обеспечить защиту от атак, ошибок и сбоев.</p>

<p>Другим вызовом является интерпретируемость нейронных сетей. Понимание того, как и почему нейронные сети принимают определенные решения, особенно в случае сложных моделей глубокого обучения, остается активной областью исследований.</p>

<p>С развитием технологий и исследований, мы можем ожидать дальнейшего развития нейронных сетей в будущем. Это включает в себя создание более эффективных алгоритмов обучения, разработку новых архитектур сетей, а также интеграцию нейронных сетей с другими технологиями, такими как квантовые вычисления и бионика.</p>

<p>В целом, эволюция нейронных сетей продолжает вести нас к новым горизонтам в области искусственного интеллекта и обработки информации. Сохранение баланса между инновациями и этическими вопросами, а также продолжение исследований в области безопасности и интерпретируемости, будет важным фактором в обеспечении устойчивого и продуктивного развития этой захватывающей области технологии.</p>

<hr>

</div>

<div class="right-column">

<div style="text-align: center;">

<source srcset="1AI.png" type="image/webp">


```
        
```

```
        <source srcset="2AI.png" type="image/webp">
        
    </div>
```

```
</div>
</div>
```

```
<hr>
```

```
<hr>
<!-- Нижний колонтитул с авторскими правами и контактной информацией -->
<div class="footer">
    <p>© 2024 NeuroTech. Все права защищены.</p>
    <!-- Добавление тега address -->
    <address>
        <p>Адрес: ул. Нейронная, д. 1, г. Сетевск</p>
        <p>Телефон: +7 (123) 456-78-90</p>
    </address>
</div>
```

```
</body>
```

```
<!-- Базовая часть ЛР 4 -->
```

<!-- 1) Вызов всплывающего окна с ФИО и группой автора сайта -->

<script>

```
document.body.addEventListener('keydown', function(event) {  
    if (event.keyCode === 13) {  
        alert("Автор сайта: Захаров Андрей, группа 4133К");  
    }  
});
```

</script>

<!-- 2) Добавить сценарий по варианту из Таблицы 1 (столбец «сценарий») -->

<script>

```
// Функция для добавления ведущего нуля к числу, если оно меньше 10  
function addLeadingZero(num) {  
    return (num < 10 ? "0" : "") + num;  
}  
  
// Функция для получения окончания слова в зависимости от числа  
function getWordEnding(number, wordForms) {  
    var remainder10 = number % 10;  
    var remainder100 = number % 100;  
    if (remainder10 === 1 && remainder100 !== 11) {  
        return wordForms[0];  
    } else if ([2, 3, 4].includes(remainder10) && ![12, 13, 14].includes(remainder100)) {  
        return wordForms[1];  
    } else {  
        return wordForms[2];  
    }  
}  
  
// Функция для обновления времени на странице  
function updateTime() {  
    var now = new Date();  
    var hours = addLeadingZero(now.getHours());  
    var minutes = addLeadingZero(now.getMinutes());  
  
    // Определяем окончания для слов "час" и "минута"  
    var hoursEnding = getWordEnding(hours, ['час', 'часа', 'часов']);  
    var minutesEnding = getWordEnding(minutes, ['минута', 'минуты', 'минут']);
```

```

        // Формируем строку с учетом окончаний
        document.getElementById("time").innerText = "Сейчас " + hours + " " + hoursEnding +
" " + minutes + " " + minutesEnding;
    }
    // Функция инициализации, вызываемая при загрузке окна
    function init() {
        updateTime();
        window.dispatchEvent(new Event('resize'));
    }
    // Устанавливаем общий обработчик события загрузки
    window.onload = init;
    // Обновляем время каждую минуту
    setInterval(updateTime, 60000);
</script>

```

<!-- 3) Добавить пустую строку в таблицу, подготовленную в работе 1 -->

```

<script>
    document.body.addEventListener('keydown', function(event) {
        if (event.keyCode === 13) {
            var table = document.getElementsByName('neural-network-table')[0];
            var newRow = table.insertRow(-1); // Вставляем строку в конец таблицы

            // Добавляем пустые ячейки
            for(var i = 0; i < table.rows[0].cells.length; i++) {
                newRow.insertCell(i);
            }
        }
    });
</script>

```

<!-- 4) Изменить цвет элемента -->

```

<script>
    document.body.addEventListener('keydown', function(event) {
        if (event.keyCode === 13) {
            var header = document.getElementsByName('page-header')[0];

```

```

        header.style.color = 'rgb(' + Math.floor(Math.random()*256) + ',' +
Math.floor(Math.random()*256) + ',' + Math.floor(Math.random()*256) + ');
    }
});
</script>

```

<!-- 5) Изменить размер элемента -->

```

<script>
    var logoSize = 150; // Начальный размер логотипа в пикселях

    document.body.addEventListener('keydown', function(event) {
        if (event.keyCode === 13) {
            var logo = document.getElementsByName('logo-image')[0];
            logoSize += 10; // Увеличиваем размер на 10 пикселей
            logo.style.width = logoSize + 'px';
        }
    });
</script>

```

<!-- Расширенная часть ЛР 4 -->

<!-- Использование addEventListener для Скроллинга -->

```

<script>
    window.addEventListener('scroll', () => {
        const height = document.documentElement.scrollHeight -
document.documentElement.clientHeight;

```

```

const scrollTop = window.pageYOffset || document.documentElement.scrollTop;
const scrolled = scrollTop / height; // Процент прокрутки страницы

// Определяем диапазон, в котором будет белый цвет
const startTransition = 0.05; // Начало перехода в белый цвет (5% прокрутки)
const endTransition = 0.99; // Конец перехода в белый цвет (90% прокрутки)

let colorValue;

if (scrolled < startTransition) {
  // Внешний диапазон: цвет серый
  colorValue = 150;
} else if (scrolled > endTransition) {
  // Внешний диапазон: цвет серый
  colorValue = 150;
} else {
  // Внутри диапазона перехода: меняем оттенок от серого к белому
  const range = endTransition - startTransition;
  const positionInRange = (scrolled - startTransition) / range;
  colorValue = Math.max(150, 255 - (Math.abs(positionInRange - 0.5) * 2 * 105)); //
Быстрый переход от серого к белому и обратно
}

document.body.style.backgroundColor = `rgb(${colorValue}, ${colorValue},
${colorValue})`;
});
</script>

```

<!-- При добавлении строки в таблицу для заполнения ячеек брать данные, введенные пользователем в элементы интерфейса (input) -->

```

<script>
function addTableRow() {
  var table = document.getElementsByName('neural-network-table')[0];
  var newRow = table.insertRow(-1); // Вставляем строку в конец таблицы

  // Получаем данные из текстовых полей

```

```

var component = document.getElementById('componentInput').value;
var description = document.getElementById('descriptionInput').value;
var example = document.getElementById('exampleInput').value;

// Создаем ячейки и добавляем в них текст
var cell1 = newRow.insertCell(0);
var cell2 = newRow.insertCell(1);
var cell3 = newRow.insertCell(2);
cell1.textContent = component;
cell2.textContent = description;
cell3.textContent = example;

// Очищаем текстовые поля
document.getElementById('componentInput').value = "";
document.getElementById('descriptionInput').value = "";
document.getElementById('exampleInput').value = "";
}
</script>

<!-- Добавить и удалить элемент списка -->
<script>
// Функция для добавления элемента в список
function addItemToList() {
    var itemText = document.getElementById('listItemInput').value;
    if (itemText.trim() === "") {
        alert('Пожалуйста, введите текст для элемента списка.');
        return;
    }

    var list = document.getElementById('dynamicList');
    var listItem = document.createElement('li');
    listItem.textContent = itemText;
    list.appendChild(listItem);

    // Очищаем поле ввода после добавления элемента
    document.getElementById('listItemInput').value = "";

```

```

    }

    // Функция для удаления последнего элемента из списка
    function removeLastItem() {
        var list = document.getElementById('dynamicList');
        if (list.children.length > 0) {
            list.removeChild(list.lastChild);
        } else {
            alert('Список уже пуст.');
```

```

    }
</script>

```

```

<!-- Использовать события окна браузера (загрузка, перемещение, скроллинг) -->

```

```

<script>
    window.addEventListener('resize', () => {
        const width = window.innerWidth;
        const height = window.innerHeight;

        document.getElementById('windowWidth').textContent = width;
        document.getElementById('windowHeight').textContent = height;
    });
</script>

```

```

<!-- ЛР 7 -->

```

```

<script>
    let shownEntries = new Set(); // Множество для хранения уже показанных записей

    // Функция для проверки наличия новых записей

```

```

function checkForNewEntries() {
    var xhr = new XMLHttpRequest();
    var browserInfo = navigator.userAgent; // Получаем информацию о браузере
    пользователя

    var ajaxObject = "XMLHttpRequest"; // Объект, выполняющий асинхронный обмен

    xhr.onreadystatechange = function() {
        if (xhr.readyState == 4 && xhr.status == 200) {
            var response = JSON.parse(xhr.responseText);
            if (response.status === "success") {
                response.data.forEach(function(entry) {
                    if (!shownEntries.has(entry.id)) {
                        // Добавляем ID новой записи в множество показанных записей
                        shownEntries.add(entry.id);

                        var message = "Новая запись: " + entry.name + "\n" +
                            "Тип: " + entry.type + "\n" +
                            "Описание: " + entry.description.substring(0, 30) + "\n" +
                            "Время запроса: " + response.timestamp + "\n" +
                            "Браузер: " + browserInfo + "\n" +
                            "AJAX объект: " + ajaxObject;
                        alert(message);
                    }
                });
            }
        }
    };

    xhr.open("POST", "http://localhost/check_new_entries.php", true);
    xhr.send();
}

// Периодическая проверка новых записей каждые 5 секунд
setInterval(checkForNewEntries, 5000);
</script>

```


</html>

page2.html

<!DOCTYPE html>

<html lang="ru">

<head>

<meta charset="UTF-8">

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

<meta name="keywords" content="нейронные сети, искусственный интеллект, машинное обучение, распознавание образов, обработка естественного языка">

<meta name="description" content="На странице рассматривается тема нейронных сетей, их применение в современном мире, а также основные типы и структура.">

<meta name="author" content="Захаров Андрей 4133K">

<meta name="language" content="ru">

<title>Нейронные сети - Вторая страница</title>

<link rel="stylesheet" href="page2.css">

<!-- -->

<style>

header,

footer {

/* Стили для заголовка и подвала*/

text-align: center;

background-image: linear-gradient(to bottom, #333, #666);

background-color: #333;

padding: 20px 0;

color: #fff;

}

body {

```
    transition: background-color 0.5s; /* Плавное изменение цвета фона */
}
```

```
.logo {
display: block;
margin: 0 auto;
width: 150px; /* Начальный размер логотипа */
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
    <!-- Шапка сайта -->
```

```
<header name="page-header">
```

```
    <h1 >NeuroTech</h1>
```

```
    <h2 >Нейронные сети - Вторая страница</h2>
```

```
    <h2 id="time"></h2>
```

```
    
```

```
    <div id="windowSizeInfo">
```

```
        Размер окна: <span id="windowWidth"></span> x <span id="windowHeight"></span>
```

```
    </div>
```

```
</header>
```

```
<div class="menu">
```

```
    <button class="menu-btn">≡ Меню</button>
```

```
    <div class="menu-items">
```

```
        <a href="index.html">Главная страница</a>
```

```
        <a href="page2.html">Вторая страница</a>
```

```
        <a href="sources.html">Использованные источники</a>
```

```
        <a href="forma.html">Анкета</a> <!-- Новый пункт меню -->
```

```
        <a href="http://localhost/neural_networks.php">Neural Networks</a> <!-- Новый пункт
меню -->
```

```
    </div>
```

</div>

<main>

<section class="content">

<article class="left-column">

<!-- Таблица с границей толщиной в 1 пиксель -->

<table name="neural-network-table" border="1">

<caption>Будущее развитие нейронных сетей</caption>

<thead>

<tr>

<th>Аспект</th>

<th>Прогноз</th>

<th>Возможные направления</th>

</tr>

</thead>

<tbody>

<tr>

<td rowspan="3"><abbr title="Технологические инновации">Инновации</abbr></td>

<td>Улучшение эффективности</td>

<td>Разработка более эффективных алгоритмов обучения и оптимизации для ускорения процессов обучения и работы нейронных сетей.</td>

</tr>

<tr>

<td>Развитие гибридных моделей</td>

<td>Интеграция нейронных сетей с другими технологиями, такими как квантовые вычисления или биоинспирированные алгоритмы, для создания более эффективных и универсальных систем.</td>

</tr>

<tr>

<td>Создание автономных систем</td>

<td>Разработка нейронных сетей с возможностью самостоятельного обучения и принятия решений без постоянного вмешательства человека.</td>

</tr>

<tr>

| | | | |
|--|---|---|--------------------|
| | | | Прикладные области |
| | Медицина и здравоохранение | Применение нейронных сетей для более точного диагностирования заболеваний и персонализированного лечения. | |
| | Умные города и транспорт | Использование нейронных сетей для оптимизации управления городскими системами, такими как транспорт и энергосбережение. | |
| | Кибербезопасность | Применение нейронных сетей для обнаружения и предотвращения кибератак и угроз безопасности в сети. | |
| | | Улучшение взаимодействия между человеком и машиной для совместного решения сложных задач. | |
| | Разработка гибридных интеллектуальных систем, где человеческий интеллект и искусственный интеллект дополняют друг друга, увеличивая эффективность и точность принятия решений | | |

| | | | |
|--|--|--|--|
| Таблица предполагаемого развития нейронных сетей, их инноваций и применений в будущем. | | | |
|--|--|--|--|

```

<div>
  <input type="text" id="componentInput" placeholder="Аспект">
  <input type="text" id="descriptionInput" placeholder="Прогноз">
  <input type="text" id="exampleInput" placeholder="Возможные направления">
  <button onclick="addTableRow()">Добавить строку</button>
</div>
<br>
<hr>
<div>
  <input type="text" id="listItemInput" placeholder="Текст элемента списка">
  <button onclick="addItemToList()">Добавить элемент</button>
  <button onclick="removeLastItem()">Удалить последний элемент</button>
</div>
<ul id="dynamicList">
  <!-- Список, в который будут добавляться элементы -->
</ul>
<hr>
<!-- Нумерованный список с перечислением ключевых тем -->
<ol type="1">
  <li>Перспективы развития нейронных сетей
    <!-- Вложенный нумерованный список-->
    <ol type="o">
      <li>Интеграция с квантовыми вычислениями
      </li>
      <li>Применение в космической индустрии
      </li>
      <li>Улучшение обучения с подкреплением
      </li>
    </ol>
  </li>
  <li>Прорывные области применения
    <!-- Вложенный нумерованный список-->
    <ol type="A">
      <li>Медицинская диагностика и терапия
      </li>
      <li>Энергетика и электроника

```

```

</li>
<li>Финансовые технологии
</li>
</ol>
</li>
<li>Этические и социальные вопросы
<!-- Вложенный нумерованный список-->
<ol type="I">
<li>Прозрачность и объяснимость алгоритмов
</li>
<li>Контроль за приватностью данных
</li>
<li>Регулирование и законодательство
</li>
</ol>
</li>
</ol>

```

```

<hr>

```

```

<br>

```

```

<p><strong>Новые горизонты в развитии нейронных сетей</strong></p>

```

```

<p><strong>Парадигменные сдвиги в архитектуре нейронных сетей</strong></p>

```

```

<p>Современное развитие нейронных сетей привнесло в мир компьютерных наук
ряд важных парадигменных сдвигов в архитектуре и функционировании искусственных
нейронных сетей.</p>

```

```

<p>Первый сдвиг связан с внедрением <dfn>архитектур, основанных на
внимании</dfn>. Это новаторский подход, позволяющий нейронным сетям
фокусироваться на наиболее важных аспектах входных данных, что приводит к более
точным и глубоким анализам. Модели с вниманием открывают новые перспективы в
области <abbr title="обработки естественного языка">обработки естественного
языка</abbr>, анализа изображений и других задач, требующих сложного понимания
контекста.</p>

```

<p>Второй важный сдвиг - это расширение области применения нейронных сетей в <dfn>обучении с подкреплением</dfn>. Этот подход позволяет создавать нейронные сети, способные самостоятельно осваивать и улучшать свои навыки взаимодействуя с окружающей средой. Такие системы могут адаптироваться к новым задачам и условиям, что делает их более гибкими и эффективными в решении широкого круга задач.</p>

<p>Третий сдвиг связан с развитием нейронных сетей с <dfn>автоэнкодерами</dfn>. Автоэнкодеры представляют собой модели, способные сжимать информацию и восстанавливать ее обратно. Это открывает новые возможности в области сжатия данных, реконструкции изображений и генерации новых данных, что является важным в контексте обработки информации в различных областях, таких как <abbr title="медицинская диагностика">медицинская диагностика</abbr> и анализ текста.</p>

<p>Таким образом, современные парадигмы развития нейронных сетей открывают новые горизонты в области искусственного интеллекта и компьютерных наук, расширяя возможности применения нейронных сетей в различных областях жизни и деятельности человека.</p>

<p>Эволюция алгоритмов и методов обучения нейронных сетей</p>

<p>Новые подходы к оптимизации и обучению нейронных сетей</p>

<p>С появлением новых архитектур нейронных сетей возникла необходимость разработки более эффективных и устойчивых методов их обучения. В этом контексте значительное внимание уделяется разработке новых алгоритмов оптимизации, способных обеспечить более стабильное и быстрое обучение нейронных сетей.</p>

<p>Один из таких подходов - это применение <dfn>мета-обучения</dfn> для автоматической настройки параметров нейронных сетей. Мета-обучение позволяет создавать алгоритмы, способные быстро адаптироваться к новым задачам и условиям обучения, что повышает гибкость и эффективность нейронных сетей.</p>

<p>Другим важным направлением является разработка методов оптимизации, устойчивых к шуму и выбросам. Одним из примеров таких методов является оптимизация с использованием <dfn>эволюционных алгоритмов</dfn>, которые основаны на принципах естественного отбора и мутаций. Это позволяет создавать более устойчивые и адаптивные нейронные сети, способные эффективно работать в различных условиях и с разнообразными данными.</p>

Кроме того, активно развиваются методы обучения с подкреплением, позволяющие создавать нейронные сети, способные самостоятельно учиться на основе полученного опыта. Это открывает новые перспективы для создания автономных систем, способных адаптироваться к изменяющимся условиям и решать сложные задачи в реальном времени.

Также стоит отметить постоянное развитие методов оптимизации градиентного спуска. В частности, алгоритмы, такие как стохастический градиентный спуск с моментом (SGD with momentum) и адам (Adam), активно исследуются и применяются для повышения скорости и стабильности обучения.

Таким образом, развитие новых алгоритмов и методов обучения нейронных сетей играет ключевую роль в повышении их эффективности и применимости в различных областях науки и техники.

Этические и социальные аспекты развития нейронных сетей

Роль этических и социальных аспектов в развитии нейронных сетей

Помимо технических и научных аспектов, важно также обращать внимание на этические и социальные вопросы, связанные с развитием нейронных сетей. С развитием этой технологии возникают новые вызовы и проблемы, которые требуют внимательного и взвешенного подхода.

Один из главных этических вопросов связан с прозрачностью и объяснимостью решений, принимаемых нейронными сетями. В связи с тем, что некоторые модели могут давать точные прогнозы, не обладая при этом объяснимостью своих выводов, становится сложно понять, каким образом они пришли к определенным результатам. Это может привести к недоверию к системам их использования в критических областях, таких как медицинская диагностика и судебные решения.

Другим важным аспектом является проблема дискриминации и справедливости. Поскольку нейронные сети обучаются на основе данных, собранных из реального мира, они могут усугублять и распространять существующие социальные неравенства. Например, если система обучается на данных, содержащих предвзятость или дискриминацию, она может повторять эти недостатки в своих выводах. Поэтому важно активно работать над созданием и использованием данных, которые отражают разнообразие и справедливость общества.

<p>Также необходимо учитывать <dfn>проблему безопасности и конфиденциальности данных</dfn>. Поскольку нейронные сети могут обрабатывать большие объемы информации, включая чувствительные данные о людях, существует риск нарушения конфиденциальности и злоупотребления этой информацией. Поэтому важно разрабатывать соответствующие механизмы защиты данных и обеспечения их безопасности.</p>

<p>И наконец, стоит уделить внимание вопросу <dfn>влияния на рынок труда и образование</dfn>. Внедрение нейронных сетей может изменить требования к рынку труда, что может повлиять на занятость и профессиональное развитие людей. Поэтому важно обеспечить доступность образовательных программ и поддержку для тех, кто нуждается в переквалификации или обновлении своих навыков в связи с развитием новых технологий.</p>

<p>Все эти аспекты подчеркивают необходимость комплексного подхода к развитию нейронных сетей, учитывающего их потенциальное воздействие на общество и человека в целом. Только такой подход позволит использовать эту мощную технологию в соответствии с общественными ценностями и принципами справедливости и устойчивости.</p>

<hr>

</article>

<aside class="right-column">

<!-- Боковая колонка с дополнительным контентом, мультимедиа -->

<div>

<div style="text-align: center;">

<!-- Блок с изображениями -->

<picture>

<source srcset="3AI.png" type="image/webp">

</picture>

<picture>

<source srcset="4AI.jpg" type="image/webp">

```
        
```

```
    </picture>
```

```
</div>
```

```
<div style="text-align: center;">
```

```
    <!-- Блок с видео -->
```

```
    <h3>Видео - Experts Predict the Future of Technology, AI & Humanity | Tech Support | WIRED</h3>
```

```
    <video controls width="480" height="320">
```

```
        <source src="video2.mp4" type="video/mp4">
```

```
    </video>
```

```
</div>
```

```
<!-- Блок с аудио -->
```

```
    <h3 style="text-align: center;">Аудио - Experts Predict the Future of Technology, AI & Humanity | Tech Support | WIRED</h3>
```

```
    <audio controls style="display: block; margin: 0 auto;">
```

```
        <source src="audio2.mp3" type="audio/mpeg">
```

```
    </audio>
```

```
</div>
```

```
</aside>
```

```
</section>
```

```
</main>
```

```
<footer>
```

```
    <p>© 2024 NeuroTech. Все права защищены.</p>
```

```
    <address>
```

```
        <p>Адрес: ул. Нейронная, д. 1, г. Сетевск</p>
```

```
        <p>Телефон: +7 (123) 456-78-90</p>
```

```
    </address>
```

```
</footer>
```

```
</body>
```

<!-- Базовая часть ЛР 4 -->

<!-- 1) Вызов всплывающего окна с ФИО и группой автора сайта -->

```
<script>
    document.body.addEventListener('keydown', function(event) {
        if (event.keyCode === 13) {
            alert("Автор сайта: Захаров Андрей, группа 4133К");
        }
    });
</script>
```

<!-- 2) Добавить сценарий по варианту из Таблицы 1 (столбец «сценарий») -->

```
<script>
    // Функция для добавления ведущего нуля к числу, если оно меньше 10
    function addLeadingZero(num) {
        return (num < 10 ? "0" : "") + num;
    }
    // Функция для получения окончания слова в зависимости от числа
    function getWordEnding(number, wordForms) {
        var remainder10 = number % 10;
        var remainder100 = number % 100;
        if (remainder10 === 1 && remainder100 !== 11) {
            return wordForms[0];
        } else if ([2, 3, 4].includes(remainder10) && ![12, 13, 14].includes(remainder100)) {
            return wordForms[1];
        } else {
            return wordForms[2];
        }
    }
</script>
```

```

    }
}
// Функция для обновления времени на странице
function updateTime() {
    var now = new Date();
    var hours = addLeadingZero(now.getHours());
    var minutes = addLeadingZero(now.getMinutes());

    // Определяем окончания для слов "час" и "минута"
    var hoursEnding = getWordEnding(hours, ['час', 'часа', 'часов']);
    var minutesEnding = getWordEnding(minutes, ['минута', 'минуты', 'минут']);

    // Формируем строку с учетом окончаний
    document.getElementById("time").innerText = "Сейчас " + hours + " " + hoursEnding +
    " " + minutes + " " + minutesEnding;
}
// Функция инициализации, вызываемая при загрузке окна
function init() {
    updateTime();
    window.dispatchEvent(new Event('resize'));
}
// Устанавливаем общий обработчик события загрузки
window.onload = init;
// Обновляем время каждую минуту
setInterval(updateTime, 60000);
</script>

```

<!-- 3) Добавить пустую строку в таблицу, подготовленную в работе 1 -->

```

<script>
document.body.addEventListener('keydown', function(event) {
    if (event.keyCode === 13) {
        var table = document.getElementsByName('neural-network-table')[0];
        var newRow = table.insertRow(-1); // Вставляем строку в конец таблицы

        // Добавляем пустые ячейки
        for(var i = 0; i < table.rows[0].cells.length; i++) {

```

```

        newRow.insertCell(i);
    }
}
});
</script>

```

<!-- 4) Изменить цвет элемента -->

```

<script>
    document.body.addEventListener('keydown', function(event) {
        if (event.keyCode === 13) {
            var header = document.getElementsByName('page-header')[0];
            header.style.color = 'rgb(' + Math.floor(Math.random()*256) + ',' +
Math.floor(Math.random()*256) + ',' + Math.floor(Math.random()*256) + ')';
        }
    });
</script>

```

<!-- 5) Изменить размер элемента -->

```

<script>
    var logoSize = 150; // Начальный размер логотипа в пикселях

    document.body.addEventListener('keydown', function(event) {
        if (event.keyCode === 13) {
            var logo = document.getElementsByName('logo-image')[0];
            logoSize += 10; // Увеличиваем размер на 10 пикселей
            logo.style.width = logoSize + 'px';
        }
    });
</script>

```

<!-- Расширенная часть ЛР 4 -->

<!-- Использование addEventListener для Скроллинга -->

<script>

```
window.addEventListener('scroll', () => {
```

```
    const height = document.documentElement.scrollHeight -
```

```
    document.documentElement.clientHeight;
```

```
    const scrollTop = window.pageYOffset || document.documentElement.scrollTop;
```

```
    const scrolled = scrollTop / height; // Процент прокрутки страницы
```

```
    // Определяем диапазон, в котором будет белый цвет
```

```
    const startTransition = 0.05; // Начало перехода в белый цвет (5% прокрутки)
```

```
    const endTransition = 0.99; // Конец перехода в белый цвет (90% прокрутки)
```

```
    let colorValue;
```

```
    if (scrolled < startTransition) {
```

```
        // Внешний диапазон: цвет серый
```

```
        colorValue = 150;
```

```
    } else if (scrolled > endTransition) {
```

```
        // Внешний диапазон: цвет серый
```

```
        colorValue = 150;
```

```
    } else {
```

```
        // Внутри диапазона перехода: меняем оттенок от серого к белому
```

```
        const range = endTransition - startTransition;
```

```
        const positionInRange = (scrolled - startTransition) / range;
```

```
        colorValue = Math.max(150, 255 - (Math.abs(positionInRange - 0.5) * 2 * 105)); //
```

```
        Быстрый переход от серого к белому и обратно
```

```
    }
```

```
    document.body.style.backgroundColor = `rgb(${colorValue}, ${colorValue},
```

```
    ${colorValue})`;
```

```
    });
```

```
</script>
```

<!-- При добавлении строки в таблицу для заполнения ячеек брать данные, введенные пользователем в элементы интерфейса (input) -->

<script>

```
function addTableRow() {
    var table = document.getElementsByName('neural-network-table')[0];
    var newRow = table.insertRow(-1); // Вставляем строку в конец таблицы

    // Получаем данные из текстовых полей
    var component = document.getElementById('componentInput').value;
    var description = document.getElementById('descriptionInput').value;
    var example = document.getElementById('exampleInput').value;

    // Создаем ячейки и добавляем в них текст
    var cell1 = newRow.insertCell(0);
    var cell2 = newRow.insertCell(1);
    var cell3 = newRow.insertCell(2);
    cell1.textContent = component;
    cell2.textContent = description;
    cell3.textContent = example;

    // Очищаем текстовые поля
    document.getElementById('componentInput').value = "";
    document.getElementById('descriptionInput').value = "";
    document.getElementById('exampleInput').value = "";
}
```

</script>

<!-- Добавить и удалить элемент списка -->

<script>

```
// Функция для добавления элемента в список
function addItemToList() {
    var itemText = document.getElementById('listItemInput').value;
    if (itemText.trim() === "") {
        alert('Пожалуйста, введите текст для элемента списка.');
```

```
        return;
    }
}
```

```

    }

    var list = document.getElementById('dynamicList');
    var listItem = document.createElement('li');
    listItem.textContent = itemText;
    list.appendChild(listItem);

    // Очищаем поле ввода после добавления элемента
    document.getElementById('listItemInput').value = "";
}

// Функция для удаления последнего элемента из списка
function removeLastItem() {
    var list = document.getElementById('dynamicList');
    if (list.children.length > 0) {
        list.removeChild(list.lastChild);
    } else {
        alert('Список уже пуст.');
```

```

    }
}
</script>

```

<!-- Использовать события окна браузера (загрузка, перемещение, скроллинг) -->

```

<script>
    window.addEventListener('resize', () => {
        const width = window.innerWidth;
        const height = window.innerHeight;

        document.getElementById('windowWidth').textContent = width;
        document.getElementById('windowHeight').textContent = height;
    });
</script>

```



```
<!-- ЛР 7 -->
```

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
```

```
<script>
```

```
    let shownEntries = new Set(); // Множество для хранения уже показанных записей
```

```
    // Функция для проверки наличия новых записей, используя jQuery для AJAX-запроса
```

```
    function checkForNewEntries() {
```

```
        $.ajax({
```

```
            url: "http://localhost/check_new_entries.php",
```

```
            type: "POST",
```

```
            success: function(response) {
```

```
                var data = JSON.parse(response);
```

```
                if (data.status === "success") {
```

```
                    data.data.forEach(function(entry) {
```

```
                        if (!shownEntries.has(entry.id)) {
```

```
                            // Добавляем ID новой записи в множество показанных записей
```

```
                            shownEntries.add(entry.id);
```

```
                            var message = "Новая запись: " + entry.name + "\n" +
```

```
                                "Тип: " + entry.type + "\n" +
```

```
                                "Описание: " + entry.description.substring(0, 30) + "\n" +
```

```
                                "Время запроса: " + data.timestamp + "\n" +
```

```
                                "Браузер: " + navigator.userAgent + "\n" +
```

```
                                "AJAX объект: jQuery.ajax";
```

```
                            alert(message);
```

```
                        }
```

```
                    });
```

```
                }
```

```
            },
```

```
            error: function(xhr, status, error) {
```

```

        console.error("Ошибка AJAX-запроса: ", status, error);
    }
});
}

// Периодическая проверка новых записей каждые 5 секунд
setInterval(checkForNewEntries, 5000);
</script>

</html>

```

sources.html

```

<!DOCTYPE html>
<html lang="ru">
<head>
    <!-- Секция head, содержащая мета-данные -->
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta name="keywords" content="нейронные сети, искусственный интеллект,
машинное обучение, распознавание образов, обработка естественного языка">
    <meta name="description" content="На странице рассматривается тема нейронных
сетей, их применение в современном мире, а также основные типы и структура.">
    <meta name="author" content="Захаров Андрей 4133K">
    <meta name="language" content="ru">
    <title>Нейронные сети - Используемые источники</title>
    <link rel="stylesheet" href="sources.css">
    <!-- -->

<style>

    header,
    footer {
        /* Стили для заголовка и подвала*/
        text-align: center;
        background-image: linear-gradient(to bottom, #333, #666);
        background-color: #333;
        padding: 20px 0;
        color: #fff;

    }

```

```

body {
transition: background-color 0.5s; /* Плавное изменение цвета фона */
}

.logo {
display: block;
margin: 0 auto;
width: 150px; /* Начальный размер логотипа */
}

</style>

</head>
<body>
  <!-- Шапка сайта -->
  <header name="page-header">
    <h1>NeuroTech</h1>
    <h2>Нейронные сети - Используемые источники</h2>
    <h2 id="time"></h2>
    
    <div id="windowSizeInfo">
      Размер окна: <span id="windowWidth"></span> x <span id="windowHeight"></span>
    </div>
  </header>
  <!-- Блок меню -->
  <div class="menu">
    <button class="menu-btn">≡ Меню</button>
    <div class="menu-items">
      <a href="index.html">Главная страница</a>
      <a href="page2.html">Вторая страница</a>
      <a href="sources.html">Используемые источники</a>
      <a href="forma.html">Анкета</a> <!-- Новый пункт меню -->
      <a href="http://localhost/neural_networks.php">Neural Networks</a> <!-- Новый пункт
меню -->
    </div>
  </div>
  <br>
  <br>
  <br>
  <!-- Основное содержимое страницы -->
  <main>
    <section class="content">
      <!-- Заголовок раздела -->
      <h2>Используемые источники:</h2>
      <ul class="sources">
        <!-- Ссылки открываются в новой вкладке благодаря атрибуту target="_blank" -->
        <li><a
href="https://ru.wikipedia.org/wiki/%D0%9D%D0%B5%D0%B9%D1%80%D0%BE%D0%
BD%D0%BD%D1%8B%D0%B5_%D1%81%D0%B5%D1%82%D0%B8"
target="_blank">Википедия - Нейронные сети</a> - История и основы нейронных
сетей.</li>

```

```

    <li><a href="https://habr.com/ru/hub/neuro/" target="_blank">Хабр - Раздел про
нейронные сети</a> - Статьи о современных тенденциях в развитии нейронных
сетей.</li>
    <li><a href="https://www.tensorflow.org/" target="_blank">Официальный сайт
TensorFlow</a> - Информация о популярной библиотеке машинного обучения и
нейронных сетей.</li>
    <li><a href="https://www.deeplearningbook.org/" target="_blank">Deep Learning
Book</a> - Основы глубокого обучения и нейронных сетей.</li>
    <li><a href="https://arxiv.org/archive/cs" target="_blank">arXiv.org - раздел по
компьютерным наукам</a> - Научные статьи и публикации о нейронных сетях.</li>
    <li><a href="https://towardsdatascience.com/" target="_blank">Towards Data
Science</a> - Статьи и tutorиалы по применению нейронных сетей в различных
областях.</li>
    <li><a href="https://machinelearningmastery.com/" target="_blank">Machine
Learning Mastery</a> - Обзоры и руководства по машинному обучению, включая
нейронные сети.</li>
    <li><a href="https://paperswithcode.com/" target="_blank">Papers With Code</a> -
Научные статьи и реализации алгоритмов, в том числе нейронных сетей.</li>
    <li><a href="https://www.youtube.com/user/GoogleDevelopers"
target="_blank">YouTube: Google Developers</a> - Видеоуроки и презентации от Google
о нейронных сетях и машинном обучении.</li>
    <li><a href="https://www.coursera.org/" target="_blank">Coursera</a> - Онлайн-
курсы по нейронным сетям и машинному обучению от ведущих университетов.</li>
</ul>
</section>
</main>
<!-- Подвал сайта с контактной информацией и копирайтом -->
<footer>
    <p>© 2024 NeuroTech. Все права защищены.</p>
    <address>
        <p>Адрес: ул. Нейронная, д. 1, г. Сетевск</p>
        <p>Телефон: +7 (123) 456-78-90</p>
    </address>
</footer>

</body>

```

```

<!-- Базовая часть ЛР 4 -->

```

```

<!-- 1) Вызов всплывающего окна с ФИО и группой автора сайта -->
<script>
    document.body.addEventListener('keydown', function(event) {
        if (event.keyCode === 13) {
            alert("Автор сайта: Захаров Андрей, группа 4133К");
        }
    });
</script>

```

```

<!-- 2) Добавить сценарий по варианту из Таблицы 1 (столбец «сценарий») -->
<script>
    // Функция для добавления ведущего нуля к числу, если оно меньше 10
    function addLeadingZero(num) {
        return (num < 10 ? "0" : "") + num;
    }
    // Функция для получения окончания слова в зависимости от числа
    function getWordEnding(number, wordForms) {
        var remainder10 = number % 10;
        var remainder100 = number % 100;
        if (remainder10 === 1 && remainder100 !== 11) {
            return wordForms[0];
        } else if ([2, 3, 4].includes(remainder10) && ![12, 13, 14].includes(remainder100)) {
            return wordForms[1];
        } else {
            return wordForms[2];
        }
    }
    // Функция для обновления времени на странице
    function updateTime() {
        var now = new Date();
        var hours = addLeadingZero(now.getHours());
        var minutes = addLeadingZero(now.getMinutes());

        // Определяем окончания для слов "час" и "минута"
        var hoursEnding = getWordEnding(hours, ['час', 'часа', 'часов']);
        var minutesEnding = getWordEnding(minutes, ['минута', 'минуты', 'минут']);

        // Формируем строку с учетом окончаний
        document.getElementById("time").innerHTML = "Сейчас " + hours + " " + hoursEnding +
        " " + minutes + " " + minutesEnding;
    }
    // Функция инициализации, вызываемая при загрузке окна
    function init() {
        updateTime();
        window.dispatchEvent(new Event('resize'));
    }
    // Устанавливаем общий обработчик события загрузки
    window.onload = init;
    // Обновляем время каждую минуту
    setInterval(updateTime, 60000);
</script>

<!-- 4) Изменить цвет элемента -->
<script>
    document.body.addEventListener('keydown', function(event) {
        if (event.keyCode === 13) {
            var header = document.getElementsByName('page-header')[0];
            header.style.color = 'rgb(' + Math.floor(Math.random()*256) + ',' +
            Math.floor(Math.random()*256) + ',' + Math.floor(Math.random()*256) + ')';
        }
    });
</script>

```

```

</script>

<!-- 5) Изменить размер элемента -->
<script>
    var logoSize = 150; // Начальный размер логотипа в пикселях

    document.body.addEventListener('keydown', function(event) {
        if (event.keyCode === 13) {
            var logo = document.getElementsByName('logo-image')[0];
            logoSize += 10; // Увеличиваем размер на 10 пикселей
            logo.style.width = logoSize + 'px';
        }
    });
</script>

```

```

<!-- Расширенная часть ЛР 4 -->

<!-- Использование addEventListener для Скроллинга -->
<script>
    window.addEventListener('scroll', () => {
        const height = document.documentElement.scrollHeight -
document.documentElement.clientHeight;
        const scrollTop = window.pageYOffset || document.documentElement.scrollTop;
        const scrolled = scrollTop / height; // Процент прокрутки страницы

        // Определяем диапазон, в котором будет белый цвет
        const startTransition = 0.05; // Начало перехода в белый цвет (5% прокрутки)
        const endTransition = 0.99; // Конец перехода в белый цвет (90% прокрутки)

        let colorValue;

        if (scrolled < startTransition) {
            // Внешний диапазон: цвет серый
            colorValue = 150;
        } else if (scrolled > endTransition) {
            // Внешний диапазон: цвет серый
            colorValue = 150;
        } else {
            // Внутри диапазона перехода: меняем оттенок от серого к белому
            const range = endTransition - startTransition;
            const positionInRange = (scrolled - startTransition) / range;

```

```

        colorValue = Math.max(150, 255 - (Math.abs(positionInRange - 0.5) * 2 * 105)); //
Быстрый переход от серого к белому и обратно
    }

```

```

        document.body.style.backgroundColor    =    `rgb(${colorValue},    ${colorValue},
    ${colorValue})`;
    });
</script>

```

```

<!-- Использовать события окна браузера (загрузка, перемещение, скроллинг) -->
<script>
    window.addEventListener('resize', () => {
        const width = window.innerWidth;
        const height = window.innerHeight;

        document.getElementById('windowWidth').textContent = width;
        document.getElementById('windowHeight').textContent = height;
    });
</script>

```

```

<!-- ЛР 7 -->

```

```

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>

```

```

<script>
    let shownEntries = new Set(); // Множество для хранения уже показанных записей

    // Функция для проверки наличия новых записей, используя jQuery для AJAX-запроса
    function checkForNewEntries() {
        $.ajax({
            url: "http://localhost/check_new_entries.php",
            type: "POST",
            success: function(response) {
                var data = JSON.parse(response);
                if (data.status === "success") {
                    data.data.forEach(function(entry) {
                        if (!shownEntries.has(entry.id)) {
                            // Добавляем ID новой записи в множество показанных записей
                            shownEntries.add(entry.id);

                            var message = "Новая запись: " + entry.name + "\n" +
                                "Тип: " + entry.type + "\n" +

```

```

        "Описание: " + entry.description.substring(0, 30) + "\n" +
        "Время запроса: " + data.timestamp + "\n" +
        "Браузер: " + navigator.userAgent + "\n" +
        "AJAX объект: jQuery.ajax";
        alert(message);
    }
    });
}
},
error: function(xhr, status, error) {
    console.error("Ошибка AJAX-запроса: ", status, error);
}
});
}

// Периодическая проверка новых записей каждые 5 секунд
setInterval(checkForNewEntries, 5000);
</script>

```

```

</html>

```