ГУАП КАФЕДРА № 43

ОТЧЕТ

ЗАЩИЩЕН С ОЦЕНКОЙ ПРЕПОДАВАТЕЛЬ

|  |  |  |
| --- | --- | --- |
| ассистент |  | М.А. Мурашова |
| 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. | 4232 | Е.А.Уткина |
|  | подпись, дата | инициалы, фамилия |

Санкт-Петербург

2025

**Содержание**

[1. Цель работы 3](#_Toc192092890)

[2. Задание 3](#_Toc192092891)

[3. Вариант задания 4](#_Toc192092892)

[4. Названия средств, использованных при выполнении работы 4](#_Toc192092893)

[5. Выполнение задания 4](#_Toc192092894)

[6. Листинг программного кода 6](#_Toc192092895)

[7. Выводы 6](#_Toc192092896)

# 1. Цель работы

Изучение и применение на практике технологии AJAX.

# 2. Задание

**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.

**Расширенное задание**

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.

# 3. Вариант задания

|  |  |  |
| --- | --- | --- |
| **№ варианта** | Тип запроса | Гр 4232 |
| 1 | Get | Уткина Екатерина Алексеевна |

Тема сайта: Язык программирования Python

# 4. Названия средств, использованных при выполнении работы

1. Редактор кода Visual Studio Code.

2. Браузер Google Chrome.

3. Браузер Microsoft Edge.

# 5. Выполнение задания

На рисунках 1-2 представлена страница ajax\_database.php в браузерах Google Chrome и Microsoft Edge.

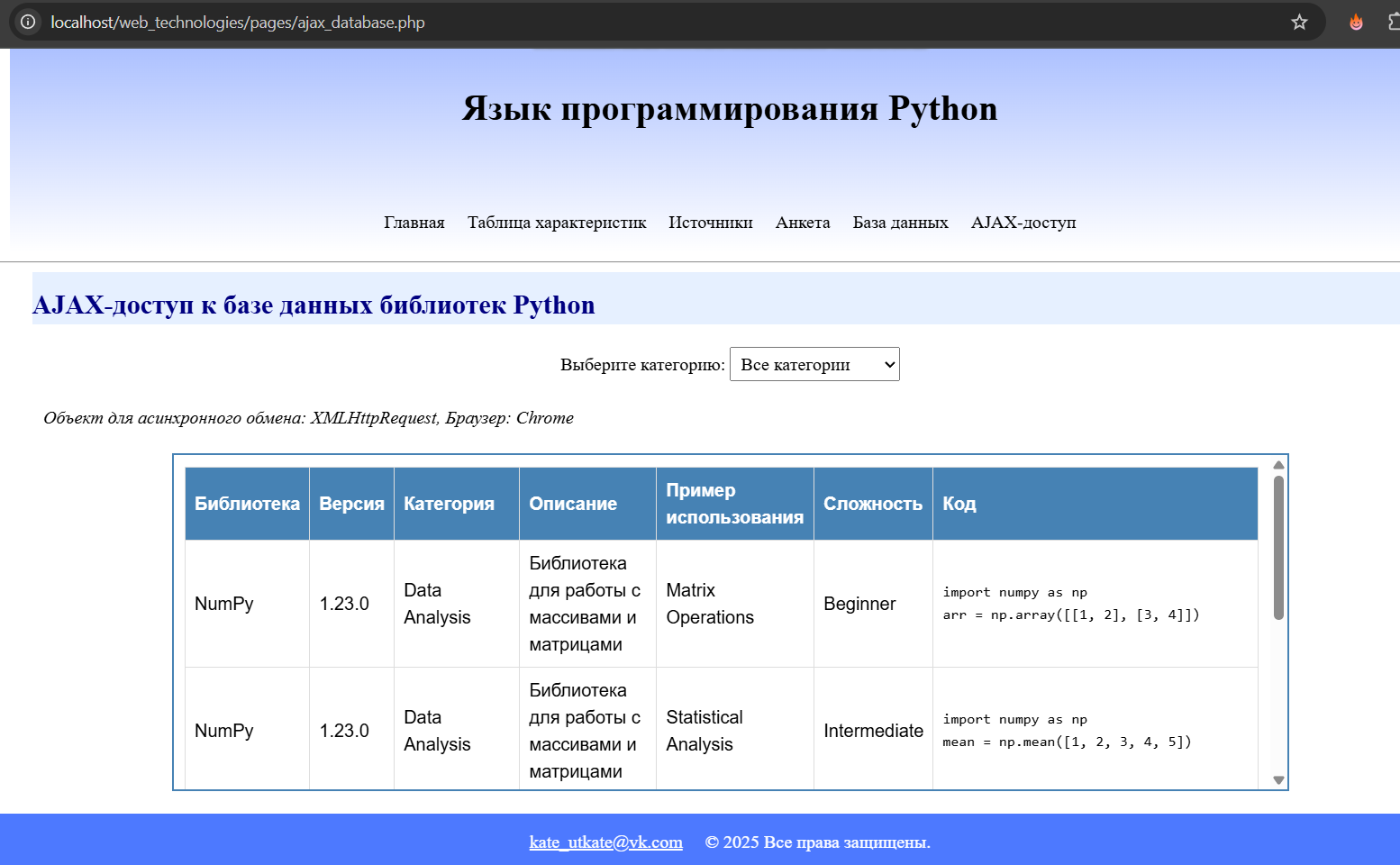


Рисунок 1 – Исходная страница в браузере Google Chrome

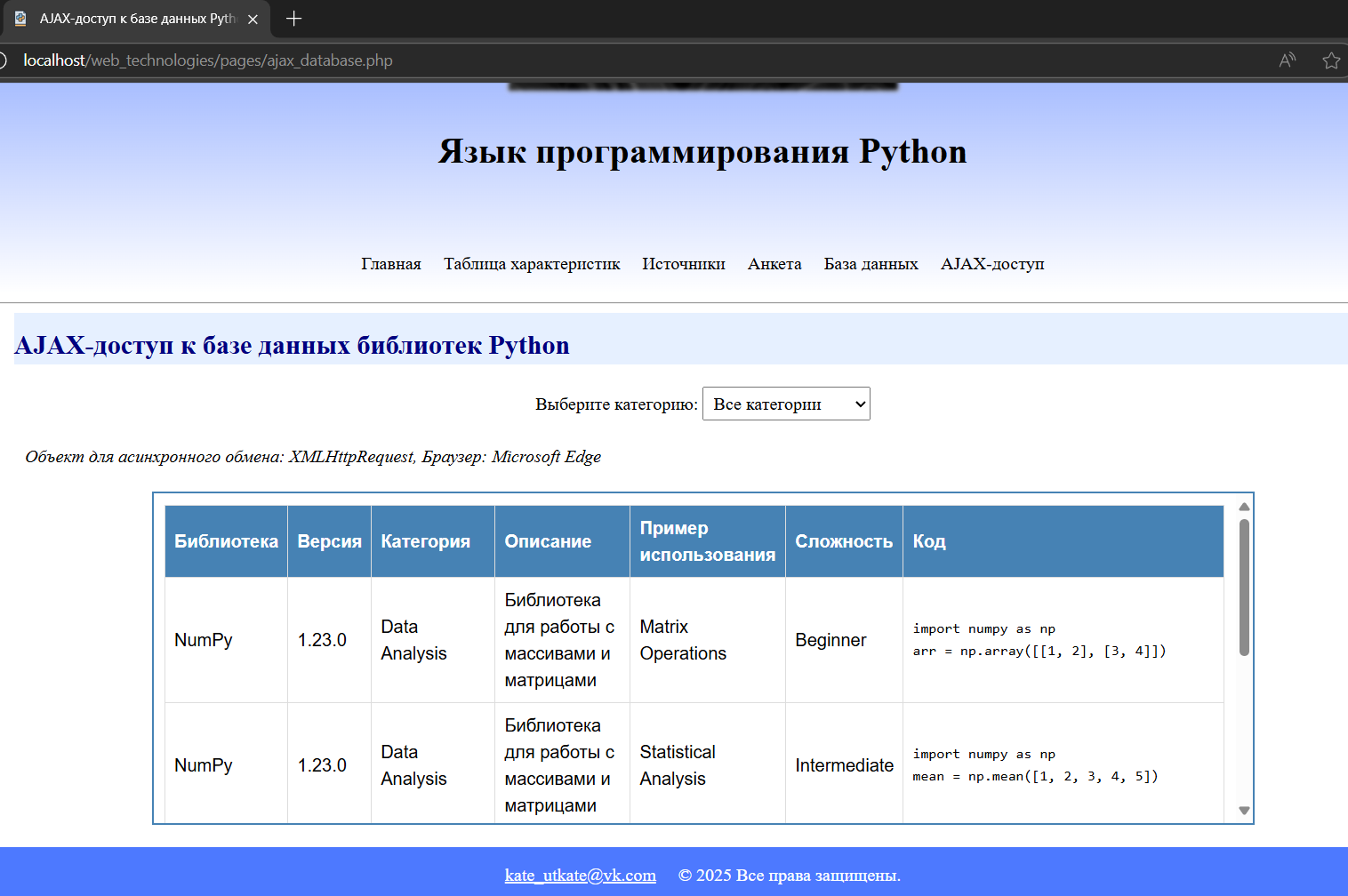


Рисунок 2 – Исходная страница в браузере Microsoft Edge

На рисунке 3 представлен выпадающий список с выбором категории фреймворка языка Python.

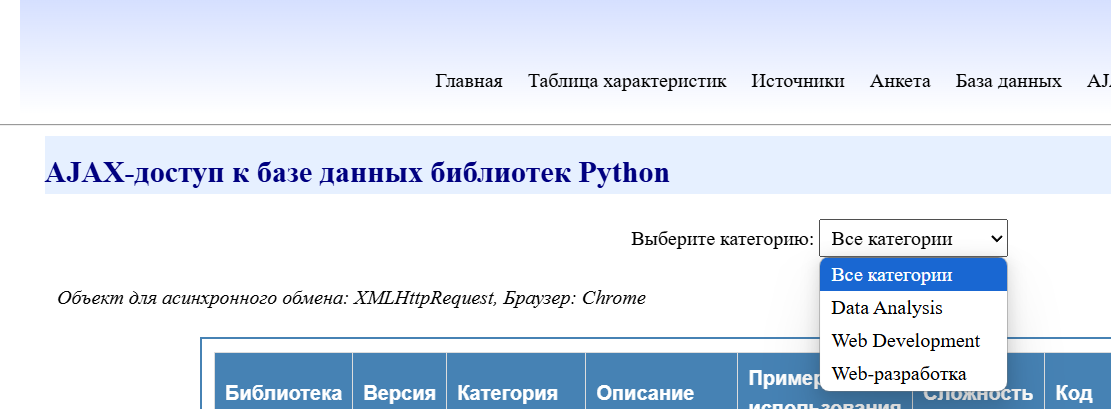


Рисунок 3 – Выбор категории

На рисунке 4 представлены данные, выводящиеся на страницу асинхронно (без обновления страницы) при выборе конкретной категории.

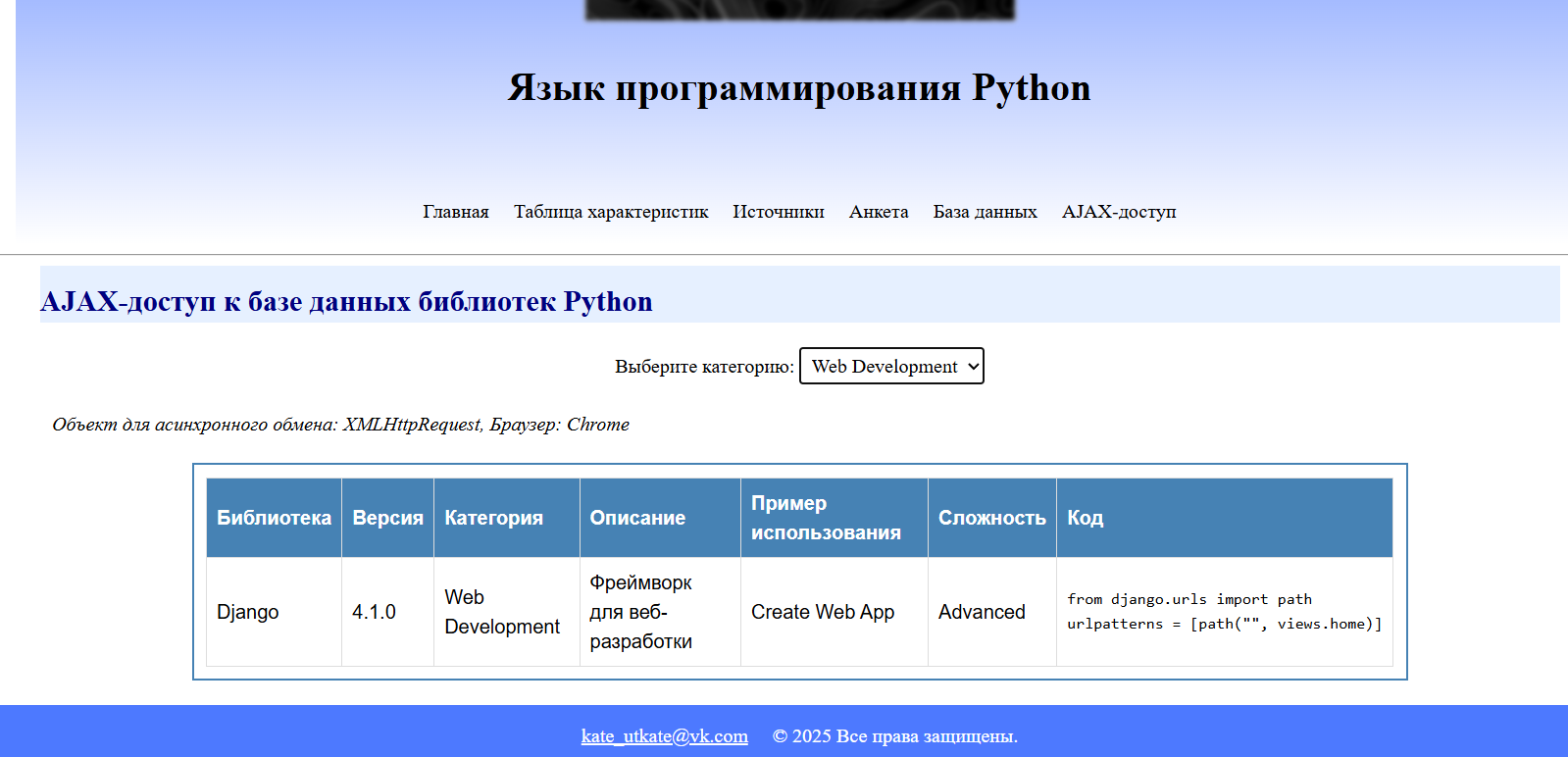


Рисунок 4 – Строки категории «Web Development»

# 6. Листинг программного кода

**get\_data.php для получения json-ответа**

<?php

header('Content-Type: application/json; charset=utf-8');

$host = 'localhost';

$user = 'root';

$pass = '';

$dbname = 'python\_db';

$conn = new mysqli($host, $user, $pass, $dbname);

if ($conn->connect\_error) {

    echo json\_encode(['error' => 'Ошибка подключения к БД: ' . $conn->connect\_error]);

    exit();

}

$conn->set\_charset("utf8");

// Получаем категорию из GET-запроса

$category = isset($\_GET['category']) ? $\_GET['category'] : '';

// Запрос к БД

if ($category) {

    $sql = "SELECT l.name, l.version, l.category, l.description, u.use\_case\_name, u.difficulty, u.example\_code

            FROM libraries l

            INNER JOIN use\_cases u ON l.id = u.library\_id

            WHERE l.category = ?";

    $stmt = $conn->prepare($sql);

    $stmt->bind\_param("s", $category);

    $stmt->execute();

    $result = $stmt->get\_result();

} else {

    $sql = "SELECT l.name, l.version, l.category, l.description, u.use\_case\_name, u.difficulty, u.example\_code

            FROM libraries l

            INNER JOIN use\_cases u ON l.id = u.library\_id";

    $result = $conn->query($sql);

}

// Формируем массив данных

$data = [];

while ($row = $result->fetch\_assoc()) {

    $data[] = $row;

}

// Возвращаем данные в формате JSON

echo json\_encode($data);

$conn->close();

?>

**ajax\_database.php:**

<!DOCTYPE html>

<html lang="ru">

<head>

    <meta charset="UTF-8">

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

    <title>AJAX-доступ к базе данных Python</title>

    <link rel="icon" type="image/x-icon" href="../images/favicon.ico">

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

    <style>

        table { border-collapse: collapse; width: 80%; margin: 20px auto; }

        th, td { border: 1px solid #ddd; padding: 8px; text-align: left; }

        th { background-color: #f2f2f2; }

        .filter-container { margin: 20px; text-align: center; }

        select { padding: 5px; }

        #browserInfo { margin: 10px; font-style: italic; }

    </style>

</head>

<body>

    <header class="header">

        <div class="header\_\_inner container">

            <div class="header\_\_logo-title">

                <img class="header\_\_logo" src="../images/header\_hat.jpg" alt="Шапка сайта">

                <h1 class="header\_\_title">Язык программирования Python</h1>

            </div>

            <nav class="header\_\_menu">

                <ul class="header\_\_menu-list">

                    <li class="header\_\_menu-item"><a class="header\_\_menu-link" href="../index.html">Главная</a></li>

                    <li class="header\_\_menu-item"><a class="header\_\_menu-link" href="characteristics.html">Таблица характеристик</a></li>

                    <li class="header\_\_menu-item"><a class="header\_\_menu-link" href="sources.html">Источники</a></li>

                    <li class="header\_\_menu-item"><a class="header\_\_menu-link" href="anketa.html">Анкета</a></li>

                    <li class="header\_\_menu-item"><a class="header\_\_menu-link" href="database.php">База данных</a></li>

                    <li class="header\_\_menu-item"><a class="header\_\_menu-link" href="ajax\_database.php">AJAX-доступ</a></li>

                </ul>

            </nav>

        </div>

    </header>

    <hr>

    <main class="main">

        <section class="section container">

            <h2 class="section\_\_title">AJAX-доступ к базе данных библиотек Python</h2>

            <div class="filter-container">

                <label for="categoryFilter">Выберите категорию:</label>

                <select id="categoryFilter" onchange="loadData()">

                    <option value="">Все категории</option>

                    <?php

                    // Подключение к БД

                    $host = 'localhost';

                    $user = 'root';

                    $pass = '';

                    $dbname = 'python\_db';

                    $conn = new mysqli($host, $user, $pass, $dbname);

                    if ($conn->connect\_error) {

                        echo "<option value=''>Ошибка подключения к БД</option>";

                    } else {

                        $conn->set\_charset("utf8");

                        // Получаем уникальные категории только для библиотек с примерами использования

                        $sql = "SELECT DISTINCT l.category

                                FROM libraries l

                                INNER JOIN use\_cases u ON l.id = u.library\_id

                                ORDER BY l.category";

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

                        if ($result->num\_rows > 0) {

                            while ($row = $result->fetch\_assoc()) {

                                $category = htmlspecialchars($row['category']);

                                echo "<option value='$category'>$category</option>";

                            }

                        }

                        $conn->close();

                    }

                    ?>

                </select>

            </div>

            <div id="browserInfo"></div>

            <table id="dataTable">

                <thead>

                    <tr>

                        <th>Библиотека</th>

                        <th>Версия</th>

                        <th>Категория</th>

                        <th>Описание</th>

                        <th>Пример использования</th>

                        <th>Сложность</th>

                        <th>Код</th>

                    </tr>

                </thead>

                <tbody id="dataBody"></tbody>

            </table>

        </section>

    </main>

    <footer class="footer">

        <address class="footer\_\_address">

            <a href="mailto:kate\_utkate@vk.com">kate\_utkate@vk.com</a>

            <p>© 2025 Все права защищены.</p>

        </address>

    </footer>

    <script>

        function loadData() {

            // Определяем объект для AJAX

            let xhr;

            if (window.XMLHttpRequest) {

                xhr = new XMLHttpRequest();

            } else {

                xhr = new ActiveXObject("Microsoft.XMLHTTP");

            }

            // Определяем браузер

            let browser = navigator.userAgent;

            let browserName = "Unknown";

            if (browser.includes("Edg")) browserName = "Microsoft Edge";

            else if (browser.includes("Chrome")) browserName = "Chrome";

            else if (browser.includes("Firefox")) browserName = "Firefox";

            else if (browser.includes("Safari")) browserName = "Safari";

            // Выводим информацию о браузере и объекте

            document.getElementById("browserInfo").innerHTML =

                "Объект для асинхронного обмена: " + xhr.constructor.name +

                ", Браузер: " + browserName;

            // Получаем выбранную категорию

            let category = document.getElementById("categoryFilter").value;

            let url = "../get\_data.php" + (category ? "?category=" + encodeURIComponent(category) : "");

            // Настраиваем запрос

            xhr.open("GET", url, true);

            xhr.onreadystatechange = function() {

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

                    let data = JSON.parse(xhr.responseText);

                    let tbody = document.getElementById("dataBody");

                    tbody.innerHTML = ""; // Очищаем таблицу

                    if (data.error) {

                        tbody.innerHTML = "<tr><td colspan='7'>" + data.error + "</td></tr>";

                        return;

                    }

                    // Заполняем таблицу

                    data.forEach(row => {

                        let tr = document.createElement("tr");

                        tr.innerHTML =

                            "<td>" + row.name + "</td>" +

                            "<td>" + row.version + "</td>" +

                            "<td>" + row.category + "</td>" +

                            "<td>" + row.description + "</td>" +

                            "<td>" + (row.use\_case\_name || "Нет") + "</td>" +

                            "<td>" + (row.difficulty || "Нет") + "</td>" +

                            "<td><pre>" + (row.example\_code || "Нет") + "</pre></td>";

                        tbody.appendChild(tr);

                    });

                }

            };

            xhr.send();

        }

        // Загружаем данные при загрузке страницы

        window.onload = loadData;

    </script>

</body>

</html>

# 7. Выводы

В ходе выполнения работы было разработано AJAX-приложение на странице ajax\_database.php, позволяющее асинхронно получать данные из базы данных с фильтрацией по категориям библиотек с использованием JavaScript и JSON. Проведено тестирование работы приложения в различных браузерах (Chrome, Microsoft Edge), подтверждено корректное определение браузера и объекта AJAX.