ГУАП КАФЕДРА № 43

ОТЧЕТ

ЗАЩИЩЕН С ОЦЕНКОЙ ПРЕПОДАВАТЕЛЬ

|  |  |  |
| --- | --- | --- |
| ассистент |  | М.А.Мурашова |
| должность, уч. степень, звание | подпись, дата | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №2 |
| ПРИМЕНЕНИЕ КАСКАДНЫХ ТАБЛИЦ СТИЛЕЙ |
| по дисциплине: WEB-ТЕХНОЛОГИИ |

РАБОТУ ВЫПОЛНИЛА

|  |  |  |
| --- | --- | --- |
| СТУДЕНТКА ГР. | 4232 | Е.А.Уткина |
|  | подпись, дата | инициалы, фамилия |

Санкт-Петербург

2025

**Содержание**

[1. Цель работы: 3](#_Toc190809475)

[2. Задание: 3](#_Toc190809476)

[3. Вариант задания 5](#_Toc190809477)

[4. Названия средств, использованных при выполнении работы 5](#_Toc190809478)

[5. Выполнение задания 5](#_Toc190809479)

[6. Листинг программного кода 9](#_Toc190809480)

[7. Вывод 16](#_Toc190809481)

# 1. Цель работы:

Применение каскадных таблиц стилей (CSS) при оформлении HTML-страниц.

# 2. Задание:

Базовое задание

HTML-страницы, разработанные в рамках лабораторной работы № 1, оформить с применением каскадных таблиц стилей. Выполнить задания:

1) Использовать три варианта подключения таблиц CSS:

- связные таблицы стилей (отдельный внешний файл)

- глобальные таблицы стилей (блок css в файле html (тег style))

- локальные таблицы стилей (локально для одного тега (атрибут style)).

2) В таблицах:

- оформить границы;

- в одну из ячеек вставить картинку, сохранив при этом выравнивание в таблице.

3) Использовать следующие технические средства:

- селекторы: тегов, классов, идентификаторов, составной;

- псевдоклассы (:hover, :visited, :link);

- указание размера: в пикселях, в миллиметрах, через процент;

- указание цвета: слово, шестнадцатеричный формат, десятичный формат.

4) Выполнить задание по индивидуальному варианту (Таблица 1).

Расширенное задание

1. Для одного и того же элемента применить правила, расположенные в блоках разного уровня (связанные, глобальные, локальные). Объяснить результат.

2. Использовать символ «+» для объединения селекторов. В отчете показать ситуацию, когда этот инструмент работает и когда не работает.

3. Использовать псевдоэлемент (:first-letter, :first-line и т.д.)

4. В оформлении применить и показать разницу между margin, border, padding

5. Скруглить углы прямоугольного элемента (свойство border-radius)

6. Сделать фон с градиентом (свойство background-image: xxx-gradient)

7. Использовать свойство text-decoration

8. Применить абсолютное позиционирование

9. Для изображения использовать свойство filter

10. Использовать селектор атрибута. В отчете показать ситуацию, когда этот инструмент работает и когда не работает.

11. Использовать свойство transform

# 3. Вариант задания

Оформление таблицы и списка:

|  |  |  |
| --- | --- | --- |
| 1 | Прокрутка в таблице | Разные типы цифр для разных уровней списка |

Тема сайта:

|  |  |
| --- | --- |
| 5 | Язык Python |

# 4. Названия средств, использованных при выполнении работы

1. Редактор кода Visual Studio Code.

2. Браузер Google Chrome.

# 5. Выполнение задания

В шапке главной страницы был применен градиент, размытие логотипа сайта, увеличение и изменение цвета ссылок в навигации:

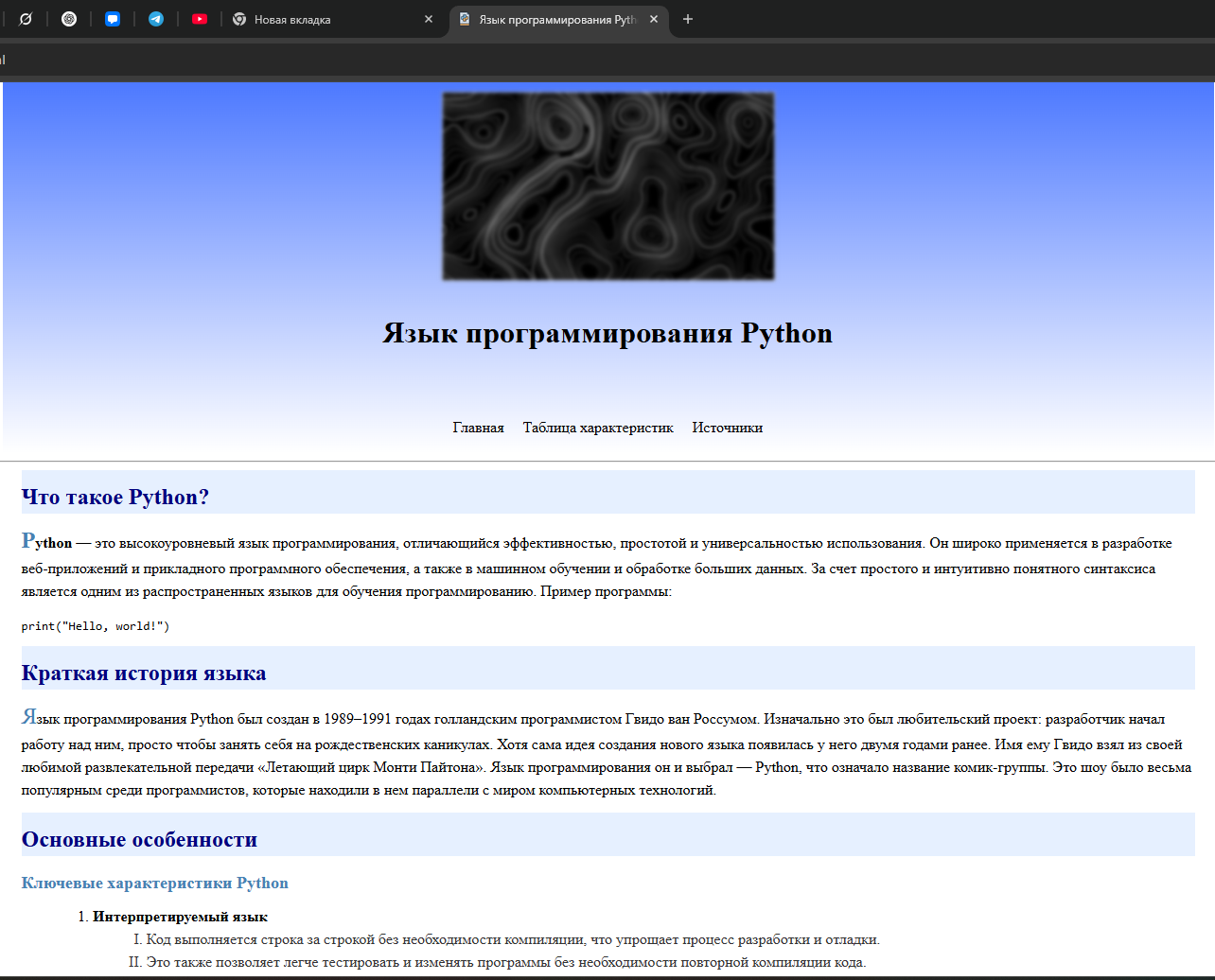


Рис. 1. Шапка и основное содержание сайта

В основном контенте были изменены цвета заголовков, добавлен фоновый цвет и изменена первая буква каждого абзаца.

В секции «Основные особенности» были использованы разные цифры для разных уровней вложенного списка, добавлено выделение красным цветом при наведении на пункт.

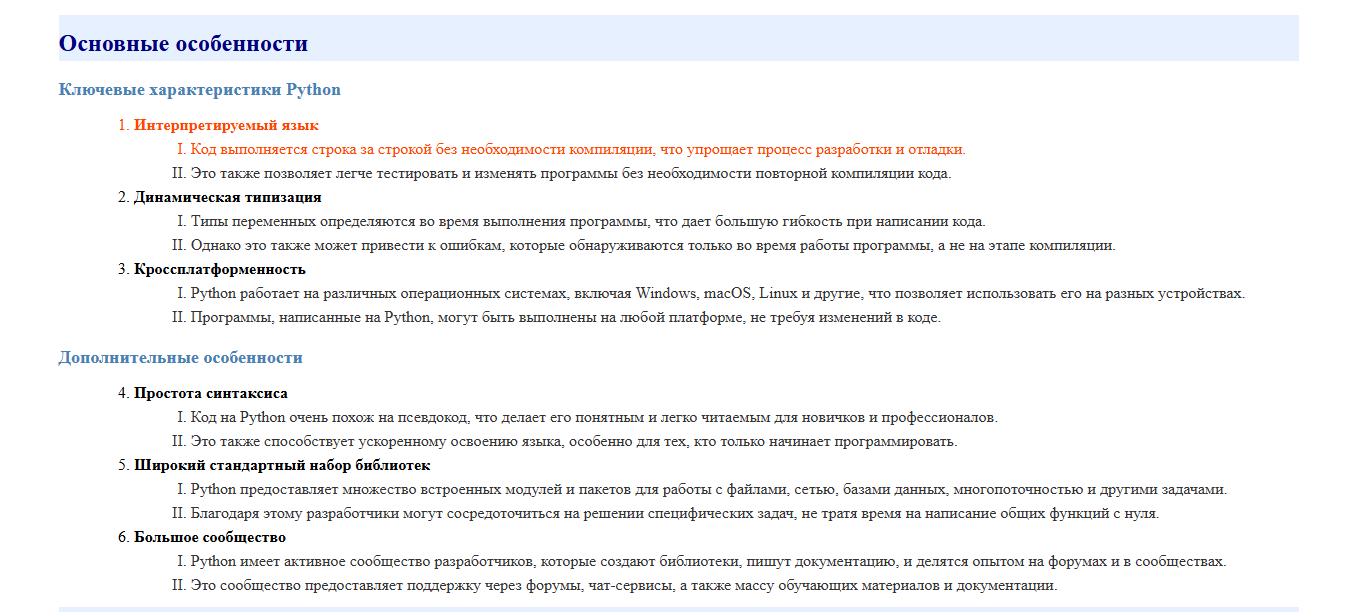


Рис. 2. Основные особенности языка

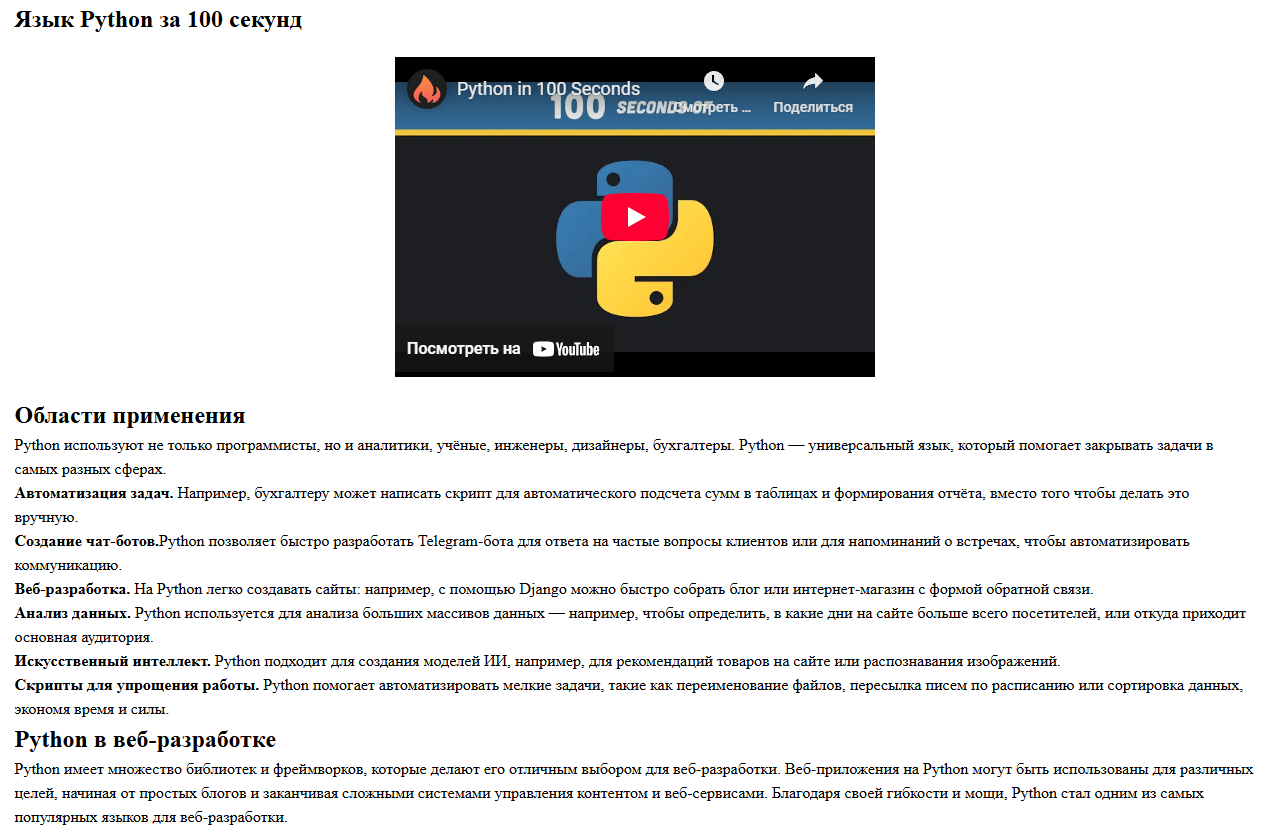


Рис. 3. Основное содержание главной страницы, часть 2

В таблицах было добавлено прокручивание, изменены цвета границ, добавлена картинка и подсветка ячеек, на которые наведена мышь.



Рис. 4. Вторая страница с таблицей

В футере были скруглены границы.

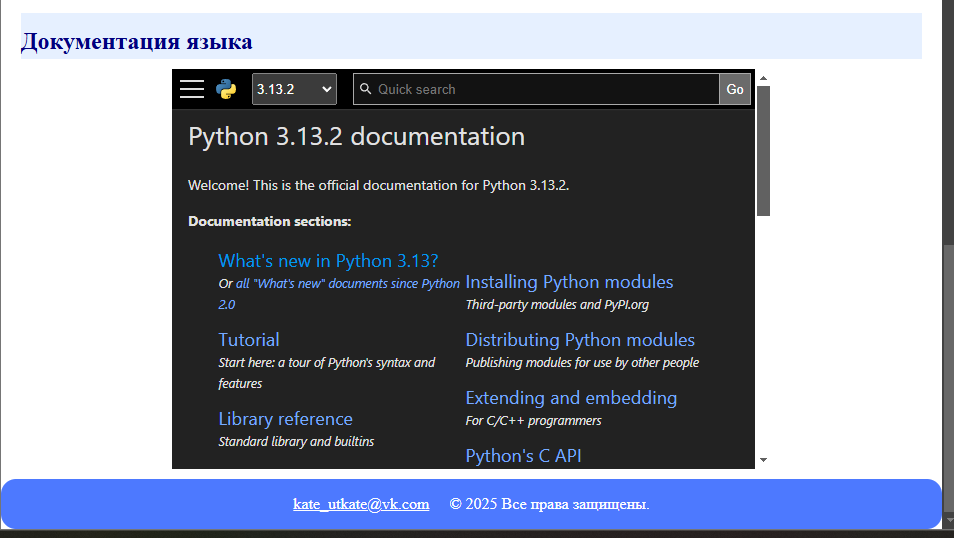


Рис. 5. Подвал сайта

# 6. Листинг программного кода

index.html

<!DOCTYPE html>

<html lang="ru">

<head>

    <meta charset="UTF-8">

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

    <title>Язык программирования Python</title>

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

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

</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="./pages/characteristics.html">Таблица характеристик</a>

                    </li>

                    <li class="header\_\_menu-item">

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

                    </li>

                </ul>

            </nav>

        </div>

    </header>

    <hr>

    <main class="main">

        <section class="section container">

            <article class="section\_\_body">

                <section class="section-1">

                    <h2 class="section\_\_title">Что такое Python?</h2>

                    <div class="section\_\_text">

                        <p>

                            <strong>Python</strong> — это высокоуровневый язык программирования, отличающийся

                            эффективностью, простотой и универсальностью использования. Он широко применяется в

                            разработке веб-приложений и

                            прикладного программного обеспечения, а также в машинном обучении и обработке больших

                            данных. За счет простого и интуитивно понятного синтаксиса является одним из

                            распространенных языков для обучения программированию. Пример программы:

                        </p>

                        <code>print("Hello, world!")</code>

                    </div>

                </section>

                <section class="section-2">

                    <h2 class="section\_\_title">Краткая история языка</h2>

                    <div class="section\_\_text">

                        <p>

                            Язык программирования Python был создан в 1989–1991 годах голландским программистом Гвидо

                            ван Россумом. Изначально это был любительский проект: разработчик начал работу над ним,

                            просто чтобы занять себя на рождественских каникулах. Хотя сама идея создания нового языка

                            появилась у него двумя годами ранее. Имя ему Гвидо взял из своей любимой развлекательной

                            передачи «Летающий цирк Монти Пайтона». Язык программирования он и выбрал — Python, что

                            означало название комик-группы. Это шоу было весьма популярным среди программистов, которые

                            находили в нем параллели с миром компьютерных технологий.

                        </p>

                    </div>

                </section>

                <section class="section-3">

                    <h2 class="section\_\_title">Основные особенности</h2>

                    <h3 class="section\_\_subtitle">Ключевые характеристики Python</h3>

                    <ol class="features-list">

                        <li><strong>Интерпретируемый язык</strong>

                            <ol class="sub-list">

                                <li>Код выполняется строка за строкой без необходимости компиляции, что упрощает процесс

                                    разработки и отладки.</li>

                                <li>Это также позволяет легче тестировать и изменять программы без необходимости

                                    повторной компиляции кода.</li>

                            </ol>

                        </li>

                        <li><strong>Динамическая типизация</strong>

                            <ol class="sub-list">

                                <li>Типы переменных определяются во время выполнения программы, что дает большую

                                    гибкость при написании кода.</li>

                                <li>Однако это также может привести к ошибкам, которые обнаруживаются только во время

                                    работы программы, а не на этапе компиляции.</li>

                            </ol>

                        </li>

                        <li><strong>Кроссплатформенность</strong>

                            <ol class="sub-list">

                                <li>Python работает на различных операционных системах, включая Windows, macOS, Linux и

                                    другие, что позволяет использовать его на разных устройствах.</li>

                                <li>Программы, написанные на Python, могут быть выполнены на любой платформе, не требуя

                                    изменений в коде.</li>

                            </ol>

                        </li>

                    </ol>

                    <h3 class="section\_\_subtitle">Дополнительные особенности</h3>

                    <ol class="features-list" start="4">

                        <li><strong>Простота синтаксиса</strong>

                            <ol class="sub-list">

                                <li>Код на Python очень похож на псевдокод, что делает его понятным и легко читаемым для

                                    новичков и профессионалов.</li>

                                <li>Это также способствует ускоренному освоению языка, особенно для тех, кто только

                                    начинает программировать.</li>

                            </ol>

                        </li>

                        <li><strong>Широкий стандартный набор библиотек</strong>

                            <ol class="sub-list">

                                <li>Python предоставляет множество встроенных модулей и пакетов для работы с файлами,

                                    сетью, базами данных, многопоточностью и другими задачами.</li>

                                <li>Благодаря этому разработчики могут сосредоточиться на решении специфических задач,

                                    не тратя время на написание общих функций с нуля.</li>

                            </ol>

                        </li>

                        <li><strong>Большое сообщество</strong>

                            <ol class="sub-list">

                                <li>Python имеет активное сообщество разработчиков, которые создают библиотеки, пишут

                                    документацию, и делятся опытом на форумах и в сообществах.</li>

                                <li>Это сообщество предоставляет поддержку через форумы, чат-сервисы, а также массу

                                    обучающих материалов и документации.</li>

                            </ol>

                        </li>

                    </ol>

                </section>

                <section class="section-4">

                    <h2 class="section\_\_title">Язык Python за 100 секунд</h2>

                    <div class="section\_\_video">

                        <iframe width="480" height="320" src="https://www.youtube.com/embed/x7X9w\_GIm1s"

                            title="Python in 100 Seconds" frameborder="0"

                            allow="accelerometer; autoplay; clipboard-write; encrypted-media; gyroscope; picture-in-picture; web-share"

                            referrerpolicy="strict-origin-when-cross-origin" allowfullscreen></iframe>

                    </div>

                </section>

                <section class="section-5">

                    <h2 class="section\_\_title">Области применения</h2>

                    <p>

                        Python используют не только программисты, но и аналитики, учёные, инженеры, дизайнеры,

                        бухгалтеры. Python — универсальный язык, который помогает закрывать задачи в самых

                        разных сферах.

                    </p>

                    <div class="section-content">

                        <div class="left-column">

                            <p>

                                <strong>Автоматизация задач.</strong> Например, бухгалтеру может написать скрипт для

                                автомат вместо того чтобы делать это вручную.

                            </p>

                            <p>

                                <strong>Создание чат-ботов.</strong> Python позволяет быстро разработать Telegram-бота

                                для ответа на частые вопросы клиентов или для напоминаний о встречах, чтобы

                                автоматизировать

                                коммуникацию.

                            </p>

                            <p>

                                <strong>Веб-разработка.</strong> На Python легко создавать сайты: например, с помощью

                                Django можно быстро собрать блог или интернет-магазин с формой обратной связи.

                            </p>

                        </div>

                        <div class="right-column">

                            <p>

                                <strong>Анализ данных.</strong> Python используется для анализа больших массивов данных

                                — например, чтобы определить, в какие дни на сайте больше всего посетителей, или откуда

                                приходит основная

                                аудитория.

                            </p>

                            <p>

                                <strong>Искусственный интеллект.</strong> Python подходит для создания моделей ИИ,

                                например, для рекомендаций товаров на сайте или распознавания изображений.

                            </p>

                            <p>

                                <strong>Скрипты для упрощения работы.</strong> Python помогает автоматизировать мелкие

                                задачи, такие как переименование файлов, пересылка писем по расписанию или сортировка

                                данных, экономя

                                время и силы.

                            </p>

                        </div>

                    </div>

                </section>

                <section class="section-6">

                    <h2 class="section\_\_title">Python в веб-разработке</h2>

                    <div class="section\_\_text">

                        <p>

                            Python имеет множество библиотек и фреймворков, которые делают его отличным выбором для

                            веб-разработки. Веб-приложения на Python могут быть использованы для различных целей,

                            начиная от простых блогов и заканчивая сложными системами управления контентом и

                            веб-сервисами. Благодаря своей гибкости и мощи, Python стал одним из самых популярных языков

                            для веб-разработки.

                        </p>

                    </div>

                </section>

                <section class="section-7">

                    <h2 class="section\_\_title">Документация языка</h2>

                    <div class="documentation">

                        <embed src="https://docs.python.org/3/" type="application/pdf" width="600" height="400">

                    </div>

                </section>

            </article>

        </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>

    <style>

        .section-content {

            display: flex;

            justify-content: space-between;

            gap: 20px;

        }

        .left-column,

        .right-column {

            width: 48%;

        }

        .left-column p,

        .right-column p {

            margin-bottom: 15px;

        }

    </style>

</body>

</html>

characteristics.html

<!DOCTYPE html>

<html lang="ru">

<head>

    <meta charset="UTF-8">

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

    <title>Таблица характеристик</title>

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

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

</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">Таблица характеристик языка 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="../pages/characteristics.html">Таблица характеристик</a>

                    </li>

                    <li class="header\_\_menu-item">

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

                    </li>

                </ul>

            </nav>

        </div>

    </header>

    <hr>

    <main class="main">

        <section class="section container">

            <article class="section\_\_body">

                <h2 class="section\_\_title">Язык Python в таблицах</h2>

                <picture class="picture">

                    <img src="../images/python\_opportunities.png" alt="Python careers">

                </picture>

                <div class="columns">

                    <div class="column">

                        <table style="width: 100%; text-align: center; border-collapse: collapse;">

                            <caption>Характеристики языка Python</caption>

                            <thead>

                                <tr>

                                    <th rowspan="2">Категория</th>

                                    <th colspan="2">Описание</th>

                                </tr>

                                <tr>

                                    <th>Характеристика</th>

                                    <th>Значение</th>

                                </tr>

                            </thead>

                            <tbody>

                                <tr>

                                    <td rowspan="2">Типизация</td>

                                    <td>Динамическая</td>

                                    <td>Тип переменной определяется во время исполнения</td>

                                </tr>

                                <tr>

                                    <td>Строгая</td>

                                    <td>Нельзя выполнять операции с несовместимыми типами</td>

                                </tr>

                                <tr>

                                    <td rowspan="2">Парадигма</td>

                                    <td>Процедурная</td>

                                    <td>Возможность писать код без классов</td>

                                </tr>

                                <tr>

                                    <td>Объектно-ориентированная</td>

                                    <td class="image-cell">

                                        <img src="https://www.python.org/static/img/python-logo.png" alt="Python Logo">

                                    </td>

                                </tr>

                            </tbody>

                            <tfoot>

                                <tr>

                                    <td colspan="3">Источник: официальная документация Python</td>

                                </tr>

                            </tfoot>

                        </table>

                    </div>

                    <div class="column">

                        <table style="width: 100%; text-align: center; border-collapse: collapse;">

                            <caption>Применение Python</caption>

                            <thead>

                                <tr>

                                    <th>Область</th>

                                    <th>Описание</th>

                                    <th>Популярность (1-5)</th>

                                </tr>

                            </thead>

                            <tbody>

                                <tr>

                                    <td>Веб-разработка</td>

                                    <td>Django, Flask</td>

                                    <td>5</td>

                                </tr>

                                <tr>

                                    <td>Машинное обучение</td>

                                    <td>TensorFlow, PyTorch</td>

                                    <td>5</td>

                                </tr>

                                <tr>

                                    <td>Анализ данных</td>

                                    <td>Pandas, NumPy</td>

                                    <td>5</td>

                                </tr>

                            </tbody>

                            <tfoot>

                                <tr>

                                    <td colspan="3">Источник: исследование 2025</td>

                                </tr>

                            </tfoot>

                        </table>

                    </div>

                </div>

            </article>

        </section>

        <section class="section container">

            <article class="section\_\_body">

                <h2 class="section\_\_title">Асинхронное программирование в Python</h2>

                <p>Асинхронное программирование позволяет выполнять несколько задач одновременно, не блокируя основной

                    поток выполнения программы. Это особенно полезно для операций ввода-вывода, таких как сетевые

                    запросы или чтение/запись файлов, где время ожидания может быть значительным. В Python асинхронное

                    программирование становится все более популярным благодаря библиотеке <code>asyncio</code>.

                    Асинхронное

                    программирование позволяет улучшить производительность и отзывчивость приложений, особенно в

                    сценариях с большим количеством операций ввода-вывода. Это достигается за счет того, что программа

                    не простаивает в ожидании завершения одной задачи, а продолжает выполнять другие задачи.</p>

                <p>Пример асинхронной программы с использованием <code>asyncio</code>:</p>

                <pre><code>

import asyncio

async def fetch\_data():

    print("Start fetching data...")

    await asyncio.sleep(2)  # Имитация долгой операции

    print("Data fetched!")

async def main():

    await asyncio.gather(fetch\_data(), fetch\_data(), fetch\_data())

# Запуск программы

asyncio.run(main())

                </code></pre>

                <p>В этом примере функция <code>fetch\_data</code> является корутиной, которая приостановится на 2

                    секунды, имитируя задержку при получении данных. Функции <code>await</code> и

                    <code>asyncio.gather()</code> позволяют параллельно выполнять несколько операций, не блокируя

                    основной поток программы.

                </p>

                <p>Асинхронное программирование позволяет значительно повысить производительность, особенно в сетевых

                    приложениях, таких как веб-серверы, где множество запросов обрабатываются одновременно без

                    блокировки.</p>

                <h2 class="section\_\_title">Оптимизация производительности Python</h2>

                <p>Существует несколько методов, с помощью которых можно улучшить производительность программ на Python.

                    Например, для оптимизации работы с большими объемами данных можно использовать такие библиотеки, как

                    <code>NumPy</code> и <code>Pandas</code>, которые реализуют вычисления на основе низкоуровневых

                    C-библиотек и обеспечивают высокую скорость обработки данных.

                </p>

                <p>Кроме того, оптимизация времени выполнения кода может быть достигнута с помощью профилирования и

                    оптимизации конкретных участков кода. Python предоставляет стандартный модуль <code>cProfile</code>

                    для профилирования кода, который помогает выявить «узкие места» в программе и улучшить

                    производительность.</p>

                <p>Пример использования <code>cProfile</code> для анализа производительности функции:</p>

                <pre><code>

import cProfile

def slow\_function():

    total = 0

    for i in range(1000000):

        total += i

    return total

cProfile.run('slow\_function()')

                </code></pre>

                <p>Этот код позволяет понять, сколько времени занимает выполнение функции и какие части кода требуют

                    оптимизации.</p>

            </article>

        </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>

</body>

</html>

sources.html

<!DOCTYPE html>

<html lang="ru">

<head>

    <meta charset="UTF-8">

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

    <title>Источники</title>

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

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

</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">Использованные источники</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="../pages/characteristics.html">Таблица характеристик</a>

                    </li>

                    <li class="header\_\_menu-item">

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

                    </li>

                </ul>

            </nav>

        </div>

    </header>

    <hr>

    <main class="main">

        <section class="section container">

            <article class="section\_\_body">

                <h2 class="section\_\_title">Ресурсы</h2>

                <div class="section\_\_text">

                    <ul>

                        <li class="list">

                            <a class="contacts" href="https://blog.skillfactory.ru/glossary/python/" target="\_blank">Python: что это за язык программирования</a>

                        </li>

                        <li class="list">

                            <a class="contacts" href="https://metanit.com/python/tutorial/1.1.php" target="\_blank">Python | введение</a>

                        </li>

                        <li class="list">

                            <a class="contacts" href="https://sky.pro/wiki/python/vvedenie-v-veb-razrabotku-na-python/" target="\_blank">Введение в веб-разработку на Python</a>

                        </li>

                        <li class="list">

                            <a class="contacts" href="https://docs.python.org/3/" target="\_blank">Официальная документация языка Python</a>

                        </li>

                        <li class="list">

                            <a class="contacts" href="https://docs.python.org/3/library/asyncio.html" target="\_blank">Официальная документация по asyncio</a>

                        </li>

                        <li class="list">

                            <a class="contacts" href="https://sky.pro/wiki/python/asinhronnoe-programmirovanie-na-python-osnovy-i-primery/" target="\_blank">Асинхронное программирование на Python: основы и примеры</a>

                        </li>

                    </ul>

                </div>

            </article>

        </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>

</body>

</html>

# 7. Вывод

В ходе выполнения лабораторной работы были изучены возможности языка стилей CSS. Применены стили к различным элементам страниц: таблицы, списки, шапка, подвал сайта. Изменены цвета, отступы, размеры и поведение при взаимодействие с элементами.