

ГЛАВА 1

Что такое интернет и как он работает

Добро пожаловать в удивительный и разнообразный мир веб-разработки! Каждое увлекательное путешествие к знаниям должно начинаться с основ. Чтобы понять, как работают веб-технологии, нужно изучить фундамент, на который они опираются: интернет.

В первой главе мы подробно расскажем о принципах его работы. Поначалу некоторые идеи и понятия могут показаться сложными, но не волнуйтесь. В этой книге мы постараемся раскрыть и упростить их, чтобы разобраться мог каждый читатель, независимо от своего опыта и знаний.

1.1. Общее представление об интернете

Интернет — это огромная переплетенная сеть, объединяющая компьютеры, серверы, маршрутизаторы и другие устройства, благодаря которой мы можем беспрепятственно обмениваться информацией. Он похож на запутанную паутину с бесчисленными соединениями и путями, по которым данные могут свободно перемещаться в любую точку мира.

Представьте, что интернет — это международное почтовое отделение, только вместо бумажных писем оно отправляет пакеты цифровых данных от одного компьютера к другому за считанные миллисекунды. Содержимое пакетов может быть разным: в них встречается текст, изображения, видео и многое другое. Просто удивительно, как интернет помогает преодолевать расстояние между

людьми, компаниями и организациями, позволяя им общаться и сотрудничать, где бы они ни находились.

Кроме того, благодаря интернету в корне изменился способ получения информации. Всего несколько нажатий клавиш — и нам доступны огромные базы данных, онлайн-библиотеки и знания по множеству различных тем. Образование стало доступнее: теперь каждый, у кого есть выход в интернет, может постоянно учиться и расширять кругозор.

Однако Сеть стала не только средством общения и обмена информацией, но и важнейшим инструментом торговли. С тех пор как появились онлайн-магазины, платформы электронной коммерции и цифровые маркетплейсы, изменилось представление о покупке и продаже товаров и услуг. Теперь компании могут привлекать клиентов по всему миру, а малый бизнес и частные предприниматели повысили шансы на успех.

Изобретение интернета во многом изменило привычный мир. У нас появилось больше возможностей для общения и развития, а дни стали гораздо насыщеннее. Теперь интернет — неотъемлемая часть современного общества. Он влияет на то, как мы взаимодействуем с окружающим миром в любой сфере, будь то работа, образование, торговля или обмен информацией.

1.1.1. Как работает интернет?

Данные в Сети передаются с помощью широко распространенного **интернет-протокола (IP)**. При этом каждое устройство с выходом в интернет — от смартфона до мощного сервера, на котором размещен ваш любимый сайт, — получает уникальный IP-адрес.

Он играет ту же роль, что и обычные домашние адреса: обеспечивает точную и надежную передачу информации выбранному получателю, но при этом действует в цифровой среде.

Если вы решите перейти на сайт, расположенный по определенному адресу (URL), он откроется в несколько этапов. Давайте рассмотрим их, чтобы лучше понимать общий принцип работы.

1. Поиск на сервере DNS

Первый этап для получения доступа к сайту. Когда вы ввели в адресную строку доменное имя, удобное для человеческого восприятия (например, www.example.com), браузеру нужно узнать соответствующий IP-адрес. Для этого он отправляет запрос на сервер системы доменных имен (DNS-сервер), который действует в роли посредника, преобразуя имена в соответствующие адреса. DNS-сервер ищет в базе IP-адрес, связанный с введенным

доменным именем, и затем возвращает его браузеру. Получив адрес, браузер пытается подключиться к сайту и загрузить нужную веб-страницу.

2. Подключение

Полученный IP-адрес позволяет браузеру подключиться к серверу, на котором расположен сайт. Такое подключение также называется «сетевым» и играет решающую роль: без него невозможно связаться с сервером и загрузить содержимое сайта.

При попытке подключиться к серверу браузер отправляет ему запрос на доступ к сайту, сообщая при этом различную информацию, например тип и версию браузера и другие необходимые сведения. Затем сервер обрабатывает запрос и в ответ подтверждает подключение, передавая данные для отображения сайта.

В случае успешного подключения браузер и сервер начинают обмениваться пакетами данных. В них содержатся инструкции и другая необходимая информация, с помощью которой можно извлечь и показать содержимое сайта. Благодаря постоянной связи браузер правильно загружает веб-страницы и обрабатывает действия пользователя, например переход по ссылкам или заполнение форм.

Таким образом, доступ к сайту невозможен без сетевого подключения. Лишь взаимодействуя с сервером, браузер способен загрузить сайт и поддерживать его беспрепятственную работу.

3. Запрос и ответ

Подключившись к серверу, браузер запрашивает у него содержимое сайта. В этом запросе подробно указано, какие именно данные нужны для отображения веб-страницы. Затем сервер обрабатывает запрос и в ответ передает различные файлы, которые отвечают за создание сайта, включая HTML, CSS, JavaScript и другие. Получив их, браузер интерпретирует данные и отображает на экране сайт, с которым можно взаимодействовать.

Таким образом, когда вы вводите в браузере адрес сайта, для загрузки веб-страницы запускается несколько незаметных для вас процессов. Без них браузер не сможет извлечь и правильно отобразить содержание веб-страниц на экране.

Хотя загрузка страницы кажется мгновенной, этот процесс состоит из множества этапов и работает благодаря слаженному взаимодействию веб-технологий.

1.1.2. IP-адреса и DNS

IP-адрес

IP-адрес, или адрес интернет-протокола, — это уникальная цифровая метка, присваиваемая каждому устройству с выходом в интернет. Он служит идентификатором, с помощью которого устройства могут устанавливать связь и обмениваться информацией.

Проще говоря, он похож на почтовый адрес, по которому устройства могут находить друг друга и взаимодействовать в Сети. Без IP-адресов было бы невозможно подключаться к интернету и получать доступ к огромному количеству информации и множеству сервисов.

DNS

Система доменных имен (DNS) играет ключевую роль в инфраструктуре интернета. Она напоминает телефонный справочник: для понятного нам доменного имени в ней найдется соответствующий IP-адрес.

Это значительно упрощает доступ к сайтам и различным онлайн-ресурсам. Только представьте, что вместо знакомых имен нам приходилось бы запоминать и вручную вводить все цифровые IP-адреса. Пользоваться интернетом было бы чрезвычайно неудобно.

Помимо преобразования доменных имен, DNS помогает управлять почтовыми службами, используется для маршрутизации интернет-трафика и обеспечивает безопасное подключение. Благодаря этой системе возможен эффективный обмен информацией и связь между устройствами по всему миру.

Иными словами, DNS создает условия для удобной и беспрепятственной работы в интернете и упрощает доступ к онлайн-ресурсам.

Таким образом, IP-адреса и DNS — это основные составляющие интернета, которые обеспечивают непрерывную связь и доступ к информации. Вместе они помогают нам просто и быстро ориентироваться в онлайн-мире.

1.1.3. Пример кода: из чего состоит URL

URL (унифицированный указатель ресурса) — не просто адрес страницы в интернете. В нем содержится подробная информация о том, как получить доступ к ресурсу. Давайте посмотрим, из чего состоит URL.

`https://www.example.com:80/path/to/page.html?query=123#section`

- **https:** протокол, который используется для доступа к ресурсу. Если в URL указан HTTPS, это означает, что подключение зашифровано для повышения безопасности. Такой протокол позволяет избежать несанкционированного доступа к данным, которыми ваше устройство обменивается с сайтом.
- **www.example.com:** доменное имя сайта. Оно выступает в роли адреса, по которому сайт расположен в интернете.
- **80:** номер порта (порты, используемые по умолчанию, часто опускаются). Порты похожи на точки доступа, через которые различные службы могут обмениваться данными.
- **/path/to/page.html:** путь к определенному ресурсу или странице сайта. Указывает на точное местоположение контента, который вам нужен.
- **?query=123:** строка запроса, с помощью которой можно передать серверу определенные параметры для отображения веб-страницы.
- **#section:** фрагмент — идентификатор для перехода к определенной части страницы. Помогает быстро открывать нужный раздел или элемент страницы, тем самым упрощая поиск информации.

Поняв принципы работы интернета, вы сможете правильно оценить многогранность и потенциал веб-разработки. В этой книге мы расскажем, как с помощью HTML и CSS создавать и оформлять веб-страницы, которыми вы будете пользоваться каждый день. Кроме того, мы рассмотрим различные технологии и фреймворки, например JavaScript и адаптивный дизайн, без которых немыслима современная веб-разработка.

Овладев основами, вы сможете создавать сайты, которые не только отлично выглядят, но и хорошо функционируют на устройствах с экранами любых размеров. Не забывайте, что каждый профессионал был когда-то новичком, а ваш путь в веб-разработку только начинается. Давайте вместе отправимся в это увлекательное путешествие и преодолеем весь путь шаг за шагом.

1.1.4. Роль веб-серверов и браузеров

Веб-серверы

Серверами называются мощные компьютеры, выделенные для специальных задач и постоянно подключенные к интернету. Они спроектированы для хранения и обработки веб-страниц, а также их быстрой и надежной передачи на устройства пользователей.

Получив запрос от браузера на содержимое определенной веб-страницы, сервер немедленно отправляет все файлы и другие данные для ее правильного отобра-

жения. Благодаря этому контент становится доступен без каких-либо задержек. Чтобы пользователю было удобно взаимодействовать с сайтом и просматривать страницы, сервер должен быстро доставлять запрашиваемую информацию.

Продвинутые технологии и инфраструктура современных серверов позволяют им обрабатывать множество запросов, поступающих одновременно от разных пользователей. Благодаря этому сервер эффективно справляется с большой нагрузкой и своевременно передает данные всем пользователям.

Однако серверы не просто отправляют браузерам файлы — они играют важную роль в хранении и обработке данных. Их оборудуют чрезвычайно вместительными накопителями, что позволяет обеспечить сайты и онлайн-приложения необходимыми ресурсами для нормальной работы. Кроме того, в серверах установлены мощные процессоры и надежные системы программного обеспечения (ПО). Благодаря этому можно проводить многосложные операции, управлять базами данных, передавать файлы для отображения веб-страниц, обеспечивать защиту сайтов с помощью протоколов безопасности и выполнять другие задачи.

Таким образом, именно серверы поддерживают работу всей Сети и играют ключевую роль в беспрепятственной доставке контента. В результате пользователи могут без труда просматривать веб-страницы. Благодаря своей эффективности, надежности и постоянному подключению к интернету серверы стали неотъемлемой частью цифровой инфраструктуры, на которую опирается современный онлайн-мир.

Веб-браузеры

Браузеры — это приложения, которые интерпретируют и отображают веб-страницы, полученные от сервера. Они открывают пользователям доступ к огромному объему информации в интернете и предлагают инструменты для работы с полученными знаниями и ресурсами.

Браузеры, например Google Chrome, Firefox или Safari, играют важнейшую роль в области веб-разработки. Они служат связующим звеном между сервером и конечным пользователем, обеспечивая беспрепятственный доступ к созданному контенту.

Значение браузеров трудно переоценить, ведь с их помощью пользователи могут просматривать сайты и мультимедиа, а также работать с различными веб-приложениями. Для полноценной работы с интернет-ресурсами нужен специальный инструмент, и без браузеров наши возможности были бы сильно ограничены.

В связи с этим веб-разработчики должны хорошо понимать, как различные браузеры обрабатывают веб-страницы и с какими ограничениями придется столкнуться. Только так можно создавать привлекательные, удобные и интуи-

тивно понятные сайты, тем самым улучшая пользовательский опыт аудитории в увлекательном путешествии по глобальной Сети.

1.1.5. Протоколы, отличные от HTTP и HTTPS

Чаще всего веб-разработчики используют HTTP (HyperText Transfer Protocol, протокол передачи гипертекста) и HTTPS (HyperText Transfer Protocol Secure — HTTP с поддержкой шифрования). Однако важно также познакомиться с другими протоколами, которые встречаются в интернете.

FTP

FTP (File Transfer Protocol, протокол передачи файлов) широко используется для эффективной и безопасной передачи файлов между клиентом и сервером в одной сети. Он обладает большим набором функций и отлично подойдет для тех, кому нужно обмениваться разнообразными файлами.

Благодаря FTP можно передавать документы, изображения или видео одинаково надежно и безопасно. Протокол обеспечит их сохранность и конфиденциальность на всех этапах обмена.

SMTP

SMTP (Simple Mail Transfer Protocol, простой протокол передачи почты) — важный протокол для бесперебойной передачи электронной почты. Он выступает в качестве надежного и эффективного канала связи, обеспечивая мгновенную и точную доставку писем адресатам.

Изучив работу SMTP, мы сможем понять, как отправляются и принимаются электронные письма, а также какие механизмы в этом задействованы. Это позволит эффективно устранять проблемы в работе электронной почты, оптимизировать отправку писем через маршрутизатор и защищать учетные записи от спама и несанкционированного доступа.

Кроме того, в SMTP есть важнейшие средства для аутентификации отправителей. С их помощью можно проверять подлинность источников и снижать риск фишинга и других видов интернет-мошенничества. Таким образом, знать принципы работы SMTP чрезвычайно полезно всем, кто взаимодействует с электронной почтой, будь то отдельные пользователи или сотрудники компаний и организаций.

IMAP и POP3

IMAP (Internet Message Access Protocol, протокол доступа к интернет-сообщениям) и POP3 (Post Office Protocol Version 3, протокол почтового отделения) — два важнейших протокола для доступа к электронной почте. Благодаря им пользователи могут легко работать с электронными письмами.

ИМАР предлагает различные средства для управления электронной почтой и доступа к ней с нескольких устройств. Это означает, что пользователи могут переключаться между устройствами и взаимодействовать с письмами, не думая о проблемах синхронизации.

POP3, напротив, предназначен для загрузки электронной почты только на одном устройстве. Он подойдет сторонникам более традиционной системы, которые предпочитают хранить письма на определенном устройстве и иметь к ним доступ без подключения к интернету. ИМАР и POP3 играют важную роль в обработке электронной почты и могут удовлетворять различные потребности пользователей.

Изучив эти интернет-протоколы, мы лучше узнаем инфраструктуру, которая лежит в основе различных онлайн-сервисов и приложений.

1.1.6. Безопасность и конфиденциальность в интернете

Объем конфиденциальной информации, которой пользователи регулярно обмениваются в интернете, постоянно увеличивается. Поэтому мы считаем необходимым как можно подробнее рассказать читателям о том, как защитить персональные данные.

Хотя в книге уделяется внимание защищенному соединению HTTPS, следует подробнее остановиться на теме безопасности и рассказать о том, почему важно понимать и соблюдать политику конфиденциальности. Мы также хотим познакомить вас с принципами шифрования данных, которые помогают защитить конфиденциальную информацию от несанкционированного доступа и потенциальных угроз.

Стоит отметить, насколько важно регулярно обновлять операционные системы и программное обеспечение. Это позволит устранять уязвимости, которыми могли бы воспользоваться злоумышленники. Помните, что надежные и уникальные пароли для каждой учетной записи помогают снизить риск взлома.

Будьте осторожны, сообщая в интернете личные данные: киберпреступники могут выманивать у пользователей конфиденциальную информацию с помощью фишинга и социальной инженерии. Не теряйте бдительности и следите за новостями в сфере кибербезопасности — так вы сможете защититься от интернет-мошенничества.

Рекомендуем регулярно проверять настройки конфиденциальности в социальных сетях и других онлайн-сервисах, чтобы определять, какая информация о вас доступна другим пользователям. Внимательно следите за тем, что вы публикуете, и ограничивайте доступ к личным данным — так вы значительно повысите свою конфиденциальность и безопасность в интернете.

1.1.7. Эволюция интернета

Чтобы понять непростую природу интернета и выяснить, какие изменения нас ждут, нужно познакомиться с историческими этапами его развития. В первую очередь отметим, что интернет изменился со времен статических веб-страниц (Web 1.0) — до динамических и интерактивных платформ, на которые мы полагаемся сегодня (Web 2.0). Зная об этом, проще представить будущее цифрового мира.

Например, теперь потенциал и дальнейшее развитие интернета во многом обусловлены стандартом Web 3.0 и такой концепцией, как «интернет вещей» (IoT). Узнав о них больше, вы сможете лучше ориентироваться в переменчивой сфере цифровых технологий.

Знания о том, как функционирует интернет, помогут заложить прочный фундамент для дальнейшего изучения веб-разработки.

Продолжив изучение этого руководства, вы поймете, что мы постарались изложить материал просто и понятно. Так информация будет доступна каждому читателю — даже тем, кто только начинает изучать веб-разработку и еще не имеет большого опыта в программировании.

Разъяснив сложные концепции, мы поможем вам взаимодействовать с постоянно развивающимся миром веб-разработки.

1.2. Введение в веб-разработку

Ваша цель может быть любой: поделиться ценными знаниями, продемонстрировать впечатляющее портфолио или создать процветающий бизнес в Сети. Чего бы вы ни пожелали, мощнейшие инструменты веб-разработки позволят вам установить и расширить цифровое присутствие. Давайте исследуем глубины веб-разработки с неиссякаемым оптимизмом и неутолимой жаждой знаний.

1.2.1. Что такое веб-разработка?

Веб-разработка — это многосложный процесс создания и поддержки сайтов, включающий в себя широкий спектр задач. Он охватывает не только видимую, но и скрытую, «закулисную» часть работы, необходимой для правильного функционирования веб-страниц.

Веб-разработчики должны хорошо разбираться в различных языках программирования, инструментах и технологиях, а также уметь с ними работать. Нужно постоянно учиться новому и быстро адаптироваться в условиях стремительно меняющейся цифровой среды.

Когда речь заходит о веб-разработке, выделяют два основных направления, которые отличаются своими задачами.

Фронтенд-разработка

Фронтенд-разработка (также «разработка клиентской части», «разработка интерфейса») сосредоточена на пользовательском интерфейсе. Фронтенд-разработчики решают, как будут выглядеть все элементы, с которыми взаимодействует пользователь, и делают сайт привлекательным и интуитивно понятным.

Они применяют HTML (Hypertext Markup Language, язык гипертекстовой разметки), CSS (Cascading Style Sheets, каскадные таблицы стилей) и язык JavaScript, а также различные фреймворки и библиотеки (например, React, Angular или Vue.js), которые позволяют повысить функциональность и интерактивность сайта.

Фронтенд-разработчики тесно сотрудничают с UX-дизайнерами¹ и вместе создают бесшовный интерфейс, ориентированный на удобство пользователей. Зная потребности целевой аудитории, они могут оптимизировать сайт, повысить его доступность для людей с ограниченными возможностями, а также быстрее вносить изменения на различных устройствах и в браузерах.

Кроме того, фронтенд-разработчики следят за последними тенденциями в своей области, чтобы страницы всегда выглядели современно и отвечали требованиям пользователя. Они постоянно изучают новые технологии и совершенствуют свои навыки, благодаря чему создают инновационные сайты высокого качества.

Таким образом, фронтенд-разработка — динамичная, постоянно развивающаяся область, в которой нужны творческий подход, знание технологий и умение ориентироваться на пользователя, что позволит создавать удобные и привлекательные сайты.

Бэкенд-разработка

В отличие от фронтенд-разработки, которая в основном посвящена внешнему виду сайта, в бэкенд-разработке (также «разработке серверной части») стараются обеспечить его бесперебойную и эффективную работу.

Бэкенд-разработчики отвечают за функционирование сайта на стороне сервера и занимаются целым рядом важнейших задач, включая управление хранилищем данных, обработку пользовательских запросов и поддержку бесперебойной связи между сайтом и базой данных. Благодаря им сайт работает эффективно и радует пользователей.

¹ UX (User Experience) — пользовательский опыт; UX-дизайн — проектирование интерфейсов с учетом удобства пользователя. — *Примеч. пер.*

Для выполнения своих задач бэкенд-разработчики используют несколько языков программирования, наиболее распространенные из которых — Python, PHP, Ruby и Java.

Но разработчики должны не только хорошо знать эти языки, но и уметь создавать инфраструктуру на стороне сервера и реализовывать многосложную логику работы сайта. Благодаря их знаниям и навыкам сайт может достичь поставленных целей.

Помимо фронтенд- и бэкенд-разработки, существуют и другие специализации, например разработка полного цикла, которая включает в себя оба направления и требует цельного представления о процессе создания сайта.

Также выделяют мобильную веб-разработку, которая сосредоточена на создании сайтов, оптимизированных для мобильных устройств, и учитывает соответствующие особенности просмотра веб-страниц и требования пользователей.

Кроме того, в веб-разработке необходимо не только иметь высокий уровень технической подготовки, но и творчески подходить к созданию страниц. Чтобы сайт был удобным и привлекательным, нужны специалисты с широким набором навыков, включая дизайн, принятие решений и коммуникацию. Сфера веб-разработки постоянно развивается и учитывает последние тенденции и технологии. Она формирует современный цифровой мир и определяет то, как мы будем пользоваться интернетом в дальнейшем.

1.2.2. Знакомство с HTML и CSS

HTML и CSS — основные инструменты веб-разработки. HTML, язык гипертекстовой разметки, отвечает за структуру и контент веб-страниц. С его помощью определяют заголовки, абзацы, списки, изображения, ссылки и другие элементы.

Кроме того, HTML предлагает широкий набор атрибутов и тегов для дополнительной настройки поведения и функций веб-страницы. Например, можно использовать атрибут `href` для создания кликабельных гиперссылок или `src`, чтобы добавить изображение или видео.

В свою очередь, технология каскадных таблиц стилей CSS отвечает за внешний вид сайта и макеты страниц. Она позволяет выбирать цвета, шрифты, настраивать внешние отступы, внутренние поля и другие параметры, чтобы сайт выглядел привлекательнее.

С помощью CSS можно создавать красивые дизайны и макеты. Благодаря различным стилям заголовков, абзацев и других элементов сайт будет выглядеть уникально и при этом гармонично.

Также CSS предлагает множество функций для создания интерактивных и динамических эффектов. CSS-анимации, переходы и трансформации помогут оживить веб-страницы и повысить вовлеченность пользователей.

Объединив возможности HTML и CSS, вы сможете создавать привлекательные интерактивные страницы, с которыми пользователям будет приятно взаимодействовать. Инструменты, заложенные в HTML и CSS, подходят для разработки как личного блога, так и сайта электронной коммерции.

Основы HTML

HTML — сокращение от Hypertext Markup Language, язык гипертекстовой разметки — ключевой компонент веб-разработки. Стоит упомянуть, что он не относится к традиционным языкам программирования, а является именно языком разметки. Он сообщает браузеру, как выстраивать веб-страницы и представлять их пользователям, которые посещают сайт во время путешествия по интернету.

Благодаря широкому набору элементов HTML позволяет организовывать и структурировать разделы сайта, чтобы вы могли настраивать и персонализировать их функции и внешний вид в соответствии со своими требованиями.

Так вы сможете создавать уникальные запоминающиеся страницы, которые привлекают аудиторию и удерживают ее внимание. В результате ваш сайт будет красиво выглядеть и отлично работать.

Рассмотрим небольшой пример документа на языке HTML.

```
<!DOCTYPE html>
<html>
<head>
  <title>Моя первая веб-страница</title>
</head>
<body>
  <h1>Привет, мир!</h1>
  <p>Это моя первая страница. Я изучаю веб-разработку!</p>
</body>
</html>
```

ОБЪЯСНЕНИЕ КОДА

- `<!DOCTYPE html>` — важный элемент, который объявляет тип документа и версию HTML.
- `<html>` — корневой элемент, который охватывает все содержимое страницы.

- `<head>` играет ключевую роль, так как предоставляет метаинформацию в документе, включая заголовок.
- `<title>` определяет название документа, которое отображается в строке заголовков окна или на вкладке браузера.
- `<body>` содержит основное наполнение документа, например текст или изображения.

Основы CSS

CSS — сокращение от Cascading Style Sheets, каскадные таблицы стилей — невероятно мощный и функциональный язык стиливого оформления, который позволяет изменить внешний вид вашего сайта.

Благодаря CSS дизайнеры могут задать оформление элементов для различных условий, например на экране компьютера, при печати на бумаге или в других медиаформатах.

Так сайт будет единообразен на огромном количестве платформ и устройств, а пользователи смогут оценить его уникальный интерфейс и бесшовную работу. Кроме того, CSS предлагает широкий набор функций и свойств для дополнительной оптимизации и настройки внешнего вида веб-страниц. Адаптивные дизайны, созданные с помощью этого языка, могут автоматически реагировать на изменения в размере и ориентации устройства, чтобы пользователям всегда было приятно просматривать ваш сайт.

Итак, CSS — незаменимый инструмент веб-дизайнеров, с помощью которого можно создавать красивые и удобные сайты для просмотра на различных платформах и устройствах.

Рассмотрим, как добавить стили CSS в предыдущий пример страницы HTML.

```
<!DOCTYPE html>
<html>
<head>
  <title>Моя первая веб-страница</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      margin: 20px;
      padding: 20px;
      background-color: #f0f0f2;
    }
  </style>
</head>
```

```
    h1 {
        color: #333366;
    }
    p {
        color: #666666;
    }
</style>
</head>
<body>
    <h1>Привет, мир!</h1>
    <p>Это моя первая страница. Я изучаю веб-разработку!</p>
</body>
</html>
```

ОБЪЯСНЕНИЕ КОДА

Тег `<style>`, расположенный в разделе `<head>`, содержит правила CSS, которые применяются к различным элементам раздела `<body>`. Эти правила задают внешний вид и стиль веб-страницы.

Без тега `<style>` было бы сложно настроить разметку, цвета, шрифты и дизайн других элементов `<body>`. Включая его в раздел `<head>`, веб-разработчики могут эффективнее организовывать CSS-код и создавать привлекательные страницы в едином стиле.

Подведем итоги

Знакомство с HTML и CSS — отправная точка вашего путешествия в мир веб-разработки. По мере изучения языков вы будете приобретать знания и умения, необходимые для создания привлекательных интерактивных и динамических веб-страниц.

Чтобы освоить веб-разработку, вам потребуются время, усердие, упорство и желание преодолевать трудности. Каждая концепция опирается на предыдущую, создавая прочный фундамент ваших знаний. Но по-настоящему закрепить изученный материал и достичь мастерства поможет практика.

Поэтому любопытство и интерес к веб-разработке — ваши верные помощники. Экспериментируйте с кодом, исследуйте различные пути достижения целей. Не бойтесь рисковать и ошибаться, ведь так вы получите ценные уроки и станете успешным веб-разработчиком. Идите вперед шаг за шагом. Именно непрерывное обучение позволит вам по-настоящему преуспеть в этой сфере.