Термин *Web-сервер* может относиться к аппаратного или программного обеспечения, или их обоих работать вместе.

1. С аппаратной точки зрения веб-сервер - это компьютер, на котором хранится программное обеспечение веб-сервера и файлы компонентов веб-сайта (например, HTML-документы, изображения, таблицы стилей CSS и файлы JavaScript). Веб-сервер подключается к Интернету и поддерживает физический обмен данными с другими устройствами, подключенными к Интернету.
2. Что касается программного обеспечения, веб-сервер включает в себя несколько частей, которые управляют тем, как веб-пользователи получают доступ к размещенным файлам. Как минимум, это *HTTP-сервер*. HTTP-сервер - это программное обеспечение, которое понимает [URL](https://developer.mozilla.org/en-US/docs/Glossary/URL) (веб-адреса) и [HTTP](https://developer.mozilla.org/en-US/docs/Glossary/HTTP) (протокол, который ваш браузер использует для просмотра веб-страниц). Доступ к HTTP-серверу можно получить через доменные имена веб-сайтов, которые он хранит, и он доставляет содержимое этих размещенных веб-сайтов на устройство конечного пользователя.

На самом базовом уровне всякий раз, когда браузеру требуется файл, размещенный на веб-сервере, браузер запрашивает файл по протоколу HTTP. Когда запрос поступает на правильный (аппаратный) веб-сервер, (программный) *HTTP-сервер* принимает запрос, находит запрошенный документ и отправляет его обратно в браузер, также через HTTP. (Если сервер не находит запрошенный документ, вместо этого он возвращает [404](https://developer.mozilla.org/en-US/docs/Web/HTTP/Status/404) ответа.)

U,{7adc1add-c298-4822-b92e-328218c8c8ba}{77},12.5,4.166666666666667

Для публикации веб-сайта вам нужен либо статический, либо динамический веб-сервер.

**Статический веб-сервер**, или стек, состоит из компьютера (аппаратного обеспечения) с HTTP-сервером (программным обеспечением). Мы называем его "статическим", потому что сервер отправляет размещенные файлы как есть в ваш браузер.

**Динамический веб-сервер** состоит из статического веб-сервера плюс дополнительного программного обеспечения, чаще всего *сервера приложений* и *базы данных*. Мы называем его "динамическим", потому что сервер приложений обновляет размещенные файлы перед отправкой содержимого в ваш браузер через HTTP-сервер.

Например, для создания конечных веб-страниц, которые вы видите в браузере, сервер приложений может заполнить HTML-шаблон содержимым из базы данных. Такие сайты, как MDN или Википедия, содержат тысячи веб-страниц. Обычно сайты такого типа состоят всего из нескольких HTML-шаблонов и гигантской базы данных, а не из тысяч статических HTML-документов. Такая настройка упрощает обслуживание и доставку контента.

## [Более глубокое погружение](https://developer.mozilla.org/en-US/docs/Learn/Common_questions/Web_mechanics/What_is_a_web_server#deeper_dive)

Для ознакомления: чтобы получить веб-страницу, ваш браузер отправляет запрос веб-серверу, который выполняет поиск запрошенного файла в своем собственном хранилище. Найдя файл, сервер считывает его, обрабатывает по мере необходимости и отправляет в браузер. Давайте рассмотрим эти шаги более подробно.

### [Хостинг файлов](https://developer.mozilla.org/en-US/docs/Learn/Common_questions/Web_mechanics/What_is_a_web_server#hosting_files)

Во-первых, веб-сервер должен хранить файлы веб-сайта, а именно все HTML-документы и связанные с ними ресурсы, включая изображения, таблицы стилей CSS, файлы JavaScript, шрифты и видео.

Технически вы могли бы разместить все эти файлы на своем компьютере, но гораздо удобнее хранить все файлы на выделенном веб-сервере, потому что:

* Выделенный веб-сервер, как правило, более доступен (запущен).
* За исключением простоев и системных неполадок, выделенный веб-сервер всегда подключен к Интернету.
* Выделенный веб-сервер может иметь один и тот же IP-адрес все время. Это называется *выделенным IP-адресом*. (Не все [интернет-провайдеры](https://developer.mozilla.org/en-US/docs/Glossary/ISP) предоставляют фиксированный IP-адрес для домашних линий.)
* Выделенный веб-сервер обычно поддерживается третьей стороной.

По всем этим причинам поиск хорошего хостинг-провайдера является ключевой частью создания вашего веб-сайта. Изучите различные услуги, предлагаемые компаниями. Выберите ту, которая соответствует вашим потребностям и бюджету. (Услуги варьируются от бесплатных до тысяч долларов в месяц.) Более подробную информацию вы можете найти [в этой статье](https://developer.mozilla.org/en-US/docs/Learn/Common_questions/Tools_and_setup/How_much_does_it_cost#hosting).

После того, как у вас появится услуга веб-хостинга, вы должны [загрузить свои файлы на свой веб-сервер](https://developer.mozilla.org/en-US/docs/Learn/Common_questions/Tools_and_setup/Upload_files_to_a_web_server).

### [Общение по HTTP](https://developer.mozilla.org/en-US/docs/Learn/Common_questions/Web_mechanics/What_is_a_web_server#communicating_through_http)

Во-вторых, веб-сервер обеспечивает поддержку [HTTP](https://developer.mozilla.org/en-US/docs/Glossary/HTTP) (**H**yper**t**ext **T**ransfer **P**rotocol). Как следует из названия, HTTP определяет способ передачи гипертекста (связанных веб-документов) между двумя компьютерами.

[Протокол](https://developer.mozilla.org/en-US/docs/Glossary/Protocol) - это набор правил для обмена данными между двумя компьютерами. HTTP - текстовый протокол без сохранения состояния.

[Текстовый](https://developer.mozilla.org/en-US/docs/Learn/Common_questions/Web_mechanics/What_is_a_web_server#textual)

Все команды представлены в виде обычного текста и понятны человеку.

[Без состояния](https://developer.mozilla.org/en-US/docs/Learn/Common_questions/Web_mechanics/What_is_a_web_server#stateless)

Ни сервер, ни клиент не помнят предыдущие сообщения. Например, полагаясь только на HTTP, сервер не может запомнить введенный вами пароль или ваш прогресс в незавершенной транзакции. Для подобных задач вам нужен сервер приложений. (Мы рассмотрим такого рода технологии в других статьях.)

HTTP предоставляет четкие правила взаимодействия клиента и сервера. Позже мы рассмотрим сам HTTP в [технической статье](https://developer.mozilla.org/en-US/docs/Web/HTTP). А пока просто помните об этих вещах:

* Обычно HTTP-запросы делают только *клиенты* и только к *серверам*. Серверы *отвечают* на HTTP-запрос *клиента*. Сервер также может загружать данные в кэш клиента до их запроса с помощью механизма, называемого [server push](https://en.wikipedia.org/wiki/HTTP/2_Server_Push).
* При запросе файла по HTTP клиенты должны указать [URL](https://developer.mozilla.org/en-US/docs/Glossary/URL) файла.
* Веб-сервер *должен отвечать* на каждый HTTP-запрос, по крайней мере, сообщением об ошибке.

На веб-сервере HTTP-сервер отвечает за обработку входящих запросов и ответы на них.

1. При получении запроса HTTP-сервер проверяет, соответствует ли запрошенный URL существующему файлу.
2. Если это так, веб-сервер отправляет содержимое файла обратно в браузер. Если нет, сервер проверит, должен ли он динамически генерировать файл для запроса (см. [Статическое или динамическое содержимое](https://developer.mozilla.org/en-US/docs/Learn/Common_questions/Web_mechanics/What_is_a_web_server#static_vs._dynamic_content)).
3. Если ни один из этих вариантов невозможен, веб-сервер чаще всего возвращает браузеру сообщение об ошибке [404 Not Found](https://developer.mozilla.org/en-US/docs/Web/HTTP/Status/404). Ошибка 404 настолько распространена, что некоторые веб-дизайнеры тратят значительное время и усилия на разработку страниц с ошибкой 404.U,{8026a2c8-0455-4dc8-a03b-7e680b555000}{122},6.25,6.25

### [Статический и динамический контент](https://developer.mozilla.org/en-US/docs/Learn/Common_questions/Web_mechanics/What_is_a_web_server#static_vs._dynamic_content)

Грубо говоря, сервер может обслуживать как статический, так и динамический контент. Помните, что термин "*статический*" означает "обслуживаемый как есть". Статические веб-сайты настраивать проще всего, поэтому мы предлагаем вам сделать свой первый сайт статичным.

Термин *динамический* означает, что сервер обрабатывает контент или даже генерирует его "на лету" из базы данных. Этот подход обеспечивает большую гибкость, но технический стек более сложный, что значительно усложняет создание веб-сайта.

Невозможно предложить один готовый сервер приложений, который будет правильным решением для каждого возможного варианта использования. Некоторые серверы приложений предназначены для размещения блогов, вики-сайтов или решений для электронной коммерции и управления ими, в то время как другие являются более универсальными. Если вы создаете динамичный веб-сайт, найдите время, чтобы изучить свои требования и найти технологию, которая наилучшим образом соответствует вашим потребностям.

Большинству разработчиков веб-сайтов не нужно создавать сервер приложений с нуля, потому что существует множество готовых решений, многие из которых легко настраиваются. Но если вам действительно нужно создать свой собственный сервер, то вы, вероятно, захотите использовать серверную платформу, используя ее существующий код и библиотеки и расширяя только те части, которые вам нужны, чтобы соответствовать вашему варианту использования. Только относительно небольшому числу разработчиков может потребоваться разработка сервера полностью с нуля: например, для того, чтобы соответствовать жестким ограничениям ресурсов во встроенной системе. Если вы хотите поэкспериментировать со сборкой сервера, ознакомьтесь с ресурсами программы обучения [программированию веб-сайтов на стороне сервера](https://developer.mozilla.org/en-US/docs/Learn/Server-side).