**What qualifications do you need to become a web developer?**

While there are no formal or specific qualifications required to become a web developer, a numerate degree in a subject such as maths or science will be useful.

You should also ideally have an aptitude for - or experience of - elements such as:

* User experience (UX)
* User interface (UI)
* Visual design
* Coding languages including HTML and CSS
* Frontend web programing languages and skills such as JavaScript, Ajax and web animation techniques
* Backend web programing languages such as C# or Java, PHP and Ruby
* Design software like Photoshop and Illustrator and Sketch
* An understanding of SEO
* Web servers and how they function

A career as a web developer is very specialised. While programs such as Adobe Dreamweaver and platforms such as WordPress are often perceived to take the place of core coding skills - and they do indeed allow novice users to create sites with basic knowledge - developing sites for corporate clients requires high levels of customisation which cannot be met with the use of programs such as Adobe Dreamweaver, as web developers are required to code in the raw language of the web.

A portfolio of your work is an ideal way to demonstrate your skills as a developer. Aim to include:

1. Examples of websites you've worked on - this allows you to share the work you have completed and helps to show prospective clients what you can do
2. Testimonials from clients you have previously worked with - this will reassure prospective clients that you have completed work for other clients and they were happy with what you delivered. Always ask any clients you work for to provide a testimonial once you complete a project
3. Your USPs - this is the ideal way to really sell yourself to prospective clients, showing what makes you stand out and why they should choose you over other developers
4. Your contact details - your name, email address, telephone number and social media handles, plus a link to your portfolio

Resources you may find helpful in building your portfolio include:

* [Sitepoint](https://www.sitepoint.com/how-to-create-a-portfolio-site-that-will-get-you-hired/)- tips on how to create a portfolio site to get you hired
* [Codementor](https://www.codementor.io/learn-programming/12-important-things-to-include-in-web-dev-portfolios) - advice on elements you should always include in your portfolio

If you are considering which subjects may stand you in good stead for a career in web development, consider numeracy-based subjects such as maths or science, plus subjects such as computer science.

**How does the web works?**

How the web works provides a simplified view of what happens when you view a webpage in a web browser on your computer or phone.

This theory is not essential to writing web code in the short term, but before long you'll really start to benefit from understanding what's happening in the background.

**Your internet connection**: Allows you to send and receive data on the web. It's basically like the street between your house and the shop.

**TCP/IP**: Transmission Control Protocol and Internet Protocol are communication protocols that define how data should travel across the internet. This is like the transport mechanisms that let you place an order, go to the shop, and buy your goods. In our example, this is like a car or a bike (or however else you might get around).

**DNS**: Domain Name Servers are like an address book for websites. When you type a web address in your browser, the browser looks at the DNS to find the website's real address before it can retrieve the website. The browser needs to find out which server the website lives on, so it can send HTTP messages to the right place (see below). This is like looking up the address of the shop so you can access it.

**HTTP**: Hypertext Transfer Protocol is an application [protocol](https://developer.mozilla.org/en-US/docs/Glossary/Protocol) that defines a language for clients and servers to speak to each other. This is like the language you use to order your goods.

**Component files**: A website is made up of many different files, which are like the different parts of the goods you buy from the shop. These files come in two main types:

**Code files**: Websites are built primarily from HTML, CSS, and JavaScript, though you'll meet other technologies a bit later.

**Assets**: This is a collective name for all the other stuff that makes up a website, such as images, music, video, Word documents, and PDFs.

## [So what happens, exactly?](https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/How_the_Web_works#so_what_happens_exactly)

When you type a web address into your browser (for our analogy that's like walking to the shop):

1. The browser goes to the DNS server, and finds the real address of the server that the website lives on (you find the address of the shop).
2. The browser sends an HTTP request message to the server, asking it to send a copy of the website to the client (you go to the shop and order your goods). This message, and all other data sent between the client and the server, is sent across your internet connection using TCP/IP.
3. If the server approves the client's request, the server sends the client a "200 OK" message, which means "Of course you can look at that website! Here it is", and then starts sending the website's files to the browser as a series of small chunks called data packets (the shop gives you your goods, and you bring them back to your house).
4. The browser assembles the small chunks into a complete web page and displays it to you (the goods arrive at your door — new shiny stuff, awesome!).

**-What’s the role of a web developer?**

A web developer is responsible for programming the code that “tells” a website how to function. A developer builds a website from the bottom up, which means designing it in such a way that end users have no difficulty navigating the site. The website should not be so simple that it does not appeal to advanced users, nor should it be so complicated that the beginner is easily lost in the process.

Web development can be divided into three parts: code that executes in a web browser and determines what customers or clients will see when they land on a website (client-side scripting); code that executes on a web server and powers the behind-the-scenes mechanics of how a website works (server-side scripting); and database technology, which helps to keep a website running smoothly and efficiently. Large-scale web projects often divide these tasks among multiple web developers.