Web Development with HTML&CSS: Publish Your Own Website in a Week!

Heng Y.

Schoolhouse.world

26 August 2024

What this series will cover

In this series, we'll:

- Learn the principles of how the internet works
- Learn how websites are structured
- Learn how to program in the main languages of website development
- Develop a small personal website
- Launch your website to the Internet!

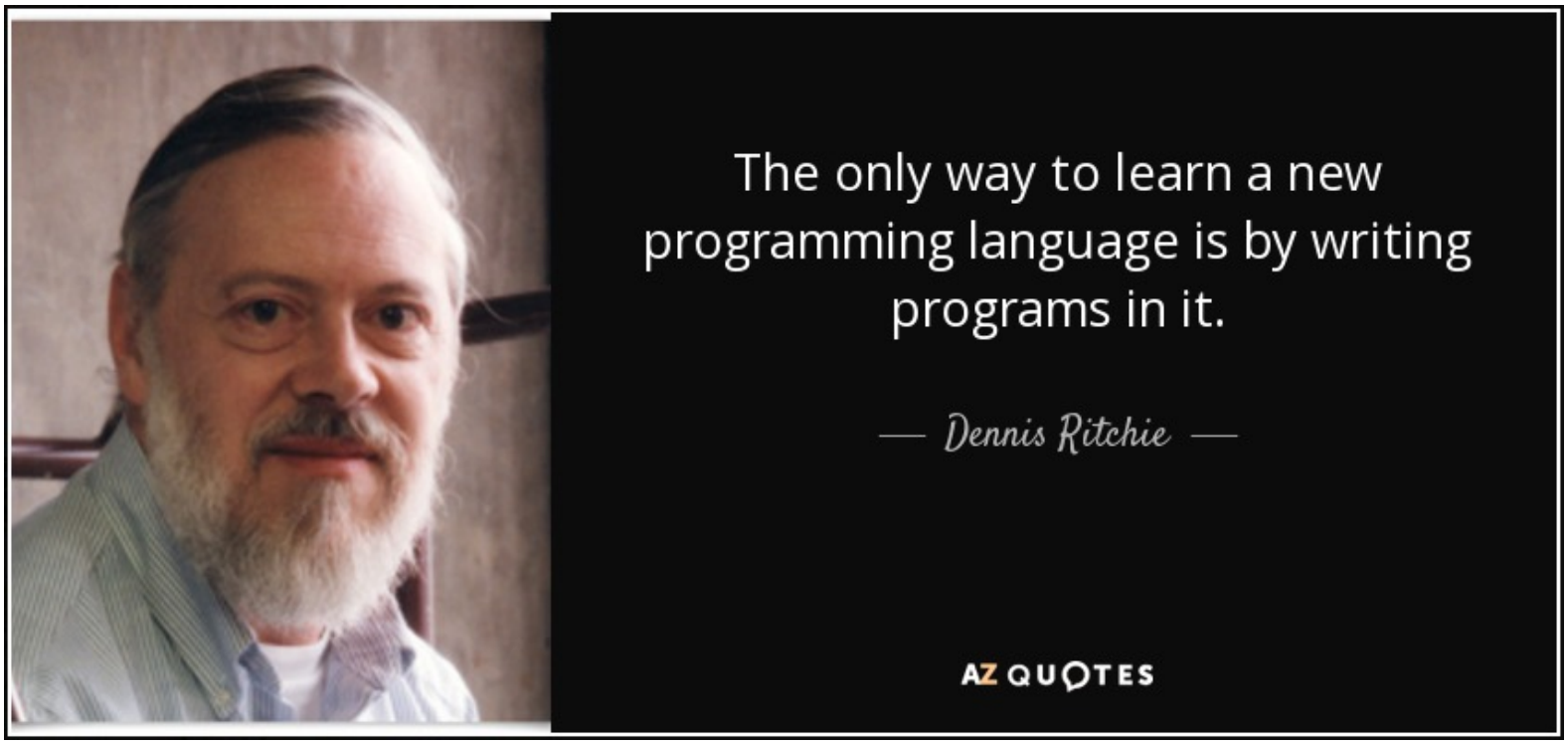
Schedule

There will be one session every day this week from Monday to Friday. Please make sure to attend all sessions.

- Mon: Basics of the internet
- Tue: Delve into HTML tags
- Wed: Delve into CSS rules
- Thu: Advanced HTML and CSS, finishing up website
- Fri: Deployment of Website

Class expectations

- Don't be hesitant to ask questions - try to engage!
- Try to keep up with the content



What we will cover today

- Basic definitions of the Internet
- Fundamental protocols: IP, DNS, HTTP
- Components of websites: HTML, CSS, JS

What you'll need for today

Please make sure you have access to a computer of any type (with a keyboard) and an email address.

What is the internet?

- The Internet is a global network that connects many devices.
- Devices are connected using hardware links (fiber-optic, coax, etc)
- Personal computers are not connected directly to the internet: the outside internet only sees one device from your network: the router.

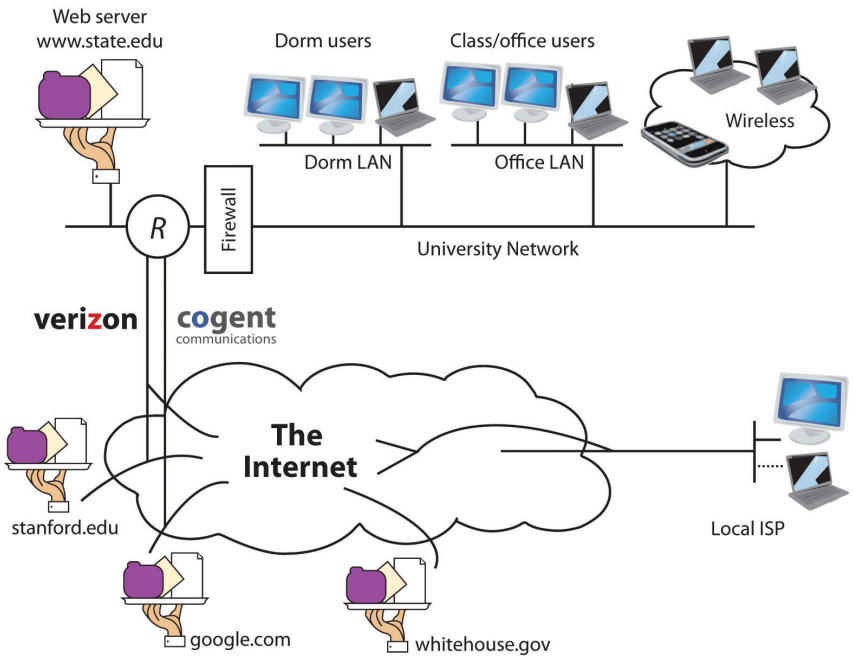


Figure: Basic structure of the internet (Gallaugher)

What is a browser?

A browser is a piece of software that is responsible for communicating with Web servers. Examples of browsers include Chrome, Firefox, and Edge.

Domain Name System

When you try to access a webpage through a browser, such as Chrome or Firefox, your browser makes a request to the server.

- First, the browser must know the IP address of the web server.
- For example, the IP address of "google.com" is 8.8.8.8
- This is obtained through the domain name system (DNS)

Figure: How DNS works (cloudacademy.com)

History of HTTP and HTML

HTTP and HTML were developed by Tim Berners-Lee, an English computer scientist.

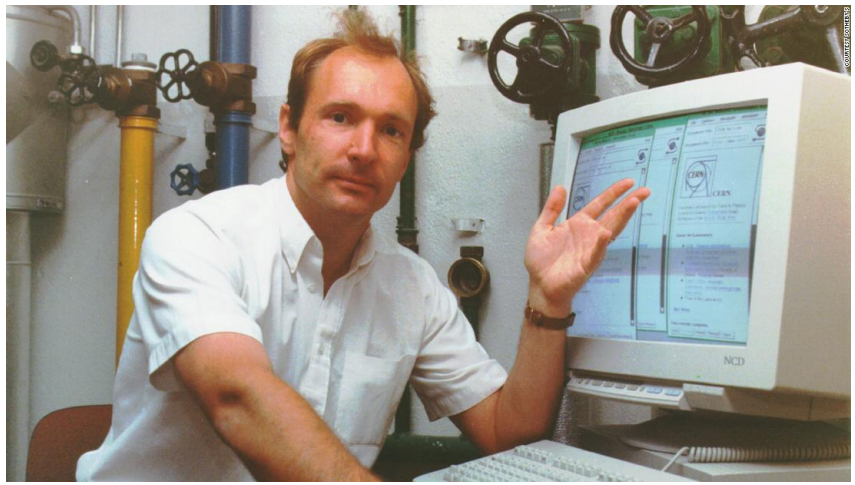


Figure: Tim Berners-Lee

HTTP Requests

Now, since we have the IP address, we can communicate with the web server.

- The web server is responsible for providing the browser with the information needed to display the webpage.
- This includes the content, styling, and logic of the webpage.
- The HTTP protocol is used to do this in a format both the browser and server recognize.

Example of HTTP Request

```
GET / HTTP/3
Host: www.google.com
User-Agent: Mozilla/5.0 (X11; Linux x86_64...
Accept: text/html,application/xhtml+xml,...
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
```

Example of HTTP Response

```
HTTP/3 200
date: Mon, 17 Jun 2024 20:14:54 GMT
content-type: text/html; charset=UTF-8
...
<!doctype html>...
```

What's inside the response?

- After the headers, we have the content of the webpage. (i.e. the text, links, anything that's displayed)
- For webpages designed to be seen in a browser, this is in HTML format.
- HTML stands for Hypertext Markup Language. It is used to give structure to the content on the webpage. For example, the text of the page and what content is bold, italic, etc.
- HTML is also used to integrate other features such as CSS and JS...more on those later.

Where does the content come from?

The HTML has to be loaded from somewhere...

- The HTML files are stored on the disk of the web server.
- A program such as Apache or Nginx is running on the server. This server program is responsible for answering the web requests and giving the browser the correct page.
- When a browser sends an HTTP request, the server looks on its disk for the correct file in a certain directory and responds with the file content as the body of the HTTP response.

Components of Websites

Modern websites are made up of many parts, as follows.

- HTML: used to provide the content and structure of each webpage.
- Cascading Style Sheets (CSS): used to style the webpage. For example, fonts, backgrounds, colors, etc.
- JavaScript (JS): used to add logic to webpages. For example, buttons and popups.
- Other files such as images and videos that are included from HTML, CSS, or JS

Web Development Tools

There are many ways to make websites, such as Google Sites, Wix, Wordpress, etc...what do these tools do?

- These tools generate HTML, CSS, and JS!
- Those are the only types of file a browser can display as a web page!
- In this course, we will get hands-on with the core technologies, so even if you decide use tools later on, you'll know how and why they work!

Creating HTML files

- HTML files are nothing special: just plain text files with a .html extension.
- Therefore, we may create them in any text editor!
- For this course, we will use JSFiddle. This tool allows us to edit HTML, CSS, and JS in one place in a web-based environment.
- JSFiddle may be used on any type of device with a web browser...which you definitely have since it wouldn't be a good idea to do web design without a web browser :)

Registering for a JSFiddle account

You will need to create a JSFiddle account to store your webpage. Navigate to https://jsfiddle.net/user/signup/ to create an account.

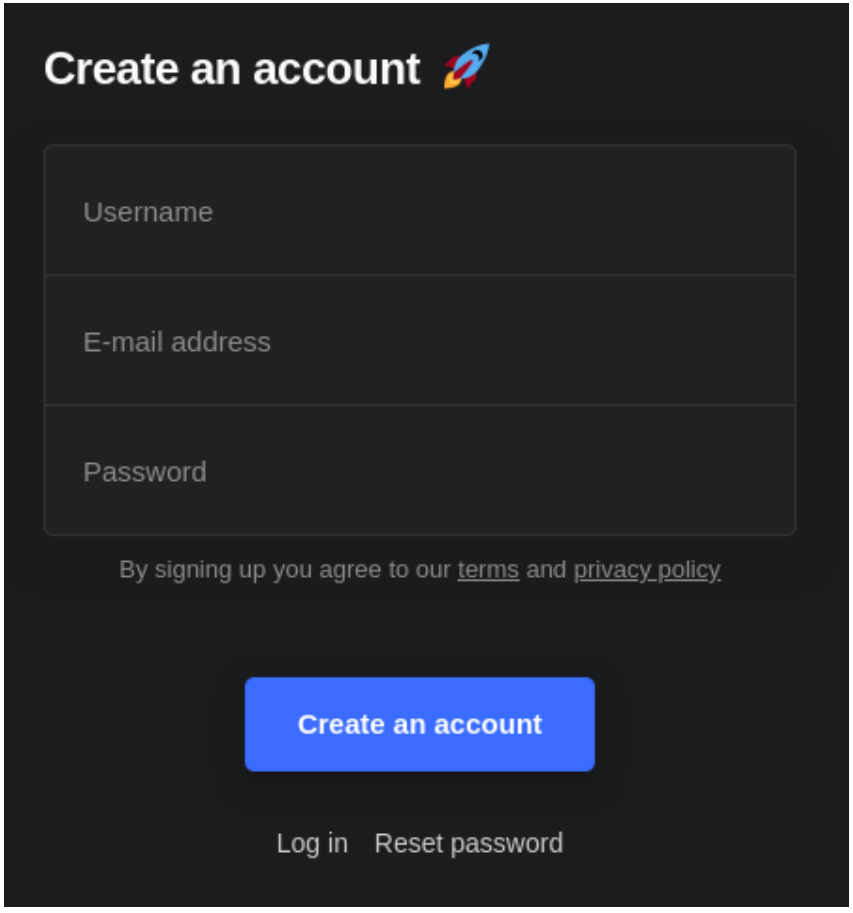


Figure: Creating JSFiddle account

Creating your first HTML page using JSFiddle

The JSFiddle page consists of four parts for HTML, CSS, JS, and the output (what the webpage will look like).
Let's make a basic "Hello World" webpage. Make sure to save!

Basic HTML page

```
<!doctype html>
<html>
    <head>
        <title>My First Page</title>
    </head>
    <body>
        <h1>Hello, World!</h1>
    </body>
</html>
```

HTML tags

- In HTML, tags represent the beginning and end of an element.
- Elements are the building blocks of HTML pages.
- Start tags are in angle brackets: `<body>`
- End tags have a slash before the tag name: `</body>`
- Anything between the start and end tags is the content of the element. For example, in `<h1>Hello</h1>` "Hello" is the content of the H1 tag.

HTML basic structure

- `<!doctype html>` Tells the browser that the file is HTML
- `<head>...</head>` contains the metadata of the page
- `<body>...</body>` contains the content of the page

HTML basic textual tags

Text tags are used to give structure to text, in a similar fashion to the heading levels in Word.

- Headings are represented by `<h1>My Heading</h1>` and the level of heading is changed by the number. In HTML, we have heading levels 1-6, so there are `<h1>, <h2>...<h6>`
- Paragraphs are represented by the `<p>` tag. For example, `<p>This is my paragraph.</p>`

What we covered today

- Principles of the internet
- Facts about HTTP requests
- Basics of HTML

Thank you!

Tomorrow, we will delve into more HTML tags and learn how to incorporate images and other elements into our webpage.
Please leave feedback to let me know how I'm doing!