Links:

1. <https://html.spec.whatwg.org/multipage/>
2. <https://caniuse.com/>
3. <https://validator.w3.org/>
4. <https://www.w3schools.com/browsers/default.asp>

[SOUND]

Let's briefly talk about the history of HTML. Its usually the case that

the history of any technology is not particularly interesting or

exciting and I can't claim that the history of

HTML is that interesting either. But there are certain parts of this

history that not only give you an understanding of how this

technology was developed, but also gives you an appreciation

of certain aspects of HTML that are still relevant and

applicable today. So before 1997,

there were no community standards, so browsers basically did

whatever they wanted. They invented new tags. They implemented the same

tags differently. I was kind of the wild west of the web. And you could go to a website and end up being told that your browser is not

compatible with this website so you have to go get a different browser in order

to even view the website to begin with. Now around 1997 the World Wide

Web Consortium, the W3C, came up with the first standard browsers actually

started to pay somewhat attention to, which is HTML4 and

they very quickly updated it to HTML4.01. That standard was, pretty loose,

and, the browser's had, still had, way too much leeway inside that standard, as to how they implemented it, and

how pages we're being rendered. So, around 2000, the W3C came up

with another specification called XHTML 1.0 and

that specification was based on XML. And XML is a very rigid but

very clear markup language. In W3C wanted to kind of pick that up and

keep going, with it and produce XHTML 2.0. The problem is that the browser vendors

which at this point already had a history of not really listening a 100%

to any standard, they decided that this whole W3C thing is moving way

too slowly, and on top of that, they felt that the specifications

are really moving in the wrong direction. So the browsers banded together, the

browser vendors banded together and they created yet another group that produced

specifications that was called WHATWG. And WHATWG, that's yet another abbreviation because

we don't have enough of them. So that group is called WHATWG and it stands for Web Hypertext

Application Technology Group. So it's a bunch of browser

vendors that got together. And this group is much less

democratic than the W3C. In other words they have one central, one

editor that makes the final decisions so all the browser vendor representatives

can argue all they want but at the end the appointed one editor

that kind of makes the final decisions. And they're the ones that

are driving the entire HTML5, what we have now,

they're the drivers behind it. So for a long time the two organizations

didn't really see eye to eye much at all and they didn't work together, so they were really going to

two different directions. But I think W3C eventually realized

that hey, you know, that organization is actually being driven by the people and

by the companies that actually matter. Because they are the guys who

are developing the browsers themselves, so maybe it's a good idea

to kind of make peace and try to see if we can

work together somehow. And this is what happened around 2007,

2009 time frame. So WHATWG and W3C started sort

of kind of working together. And what they produced in the end

is what we have now, HTML5. So what does any of that matter to you? Well what matters now is that there are

two organizations out there, one is called W3C and the other one is WHATWG and

they're both in charge of HTML somewhat. So it could be that you

could get into a situation, it's certainly possible that a browser

potentially can be compliant with a specification which isn't yet

even formalized by the official W3C. So, it makes your head spin a little bit,

doesn't it? So the way they divided it up is

that W3C is going to be in charge of the standards since this is what their

bread and butter is, its standards. And they are going to be in

charge of the HTML5 standard. And WHATWG is not even going to give

a version to their HTML standard, they're just going to say it's HTML and

it's constantly evolving. There is no version. We never stop. We keep evolving it. And that WHATWG, they're the ones

that really implement the browsers. So what W3C does is it kind of cherry

picks some of the more successful things that are actually kind of start

to be implemented by the browsers and they slowly bring it into

the official standard. So the point of this history,

this quick history lesson, is that this whole thing

is a bit of a muddy waters. And it's kind of hard

to keep track of well, what is the browser supposed

to be compliant with? But unfortunately it is our job as

developers to keep track of these capabilities. And it's particularly important

nowadays because all the major modern browsers are all evergreen browsers. And what that means is is that

they silently update themselves on the users computer. So you no longer even

need to have the user acknowledge the fact

that they want to update. That means that,

which is a good thing actually, right. Because now every browser is kind of

patched as much as they can be patched for security reasons. And also it kind of automatically

without the user intervention gets the latest greatest features of HTML5

and all of the browser software itself. But every major browser Chrome,

Firefox, Safari, Internet Explorer, they all are evergreen browsers, so

they update themselves automatically. So what I would like to do now is

give you a few resources on the web that will make this job keeping track

of these changes a little bit easier. The first resource that I

really should share is the W3C. The W3C HTML5 standard, and this is

a very academic type of reading but you can still look stuff up in

it that is somewhat useful. And certainly if you want to kind of

know what the actual standard is that everybody agreed to,

this is a great document to start with. Another excellent site that you

should definitely check out, and kind of keep in your

arsenal is caniuse.com. And this is a site that basically

itself keeps track of HTML5 standards, SVG standards, CSS standards,

JavaScript APIs and on, and on, and on in the web space to be able to

tell you which browser supports what. So, for example if I search for

Srcset, which is a pretty new attribute that is going to hopefully

come soon to the image tags. Which we'll talk about it later. But you can take a look and see if when I type Srcset that it will

tell you well this is an attribute. It will also tell you which browsers at

this point is supported and which ones, like IE, do not. And you can yourself take a look at and

see whether or not something that a browser let's say you're not

really targeting, you're not really, don't care so much about, if it doesn't

support it, doesn't support it fully, you can make the decision whether or not

you want to use that particular feature. Another great resource

to find out whether or not your HTML actually is going to work

in browsers is to use a validator and there's a great validator

provided by the W3.org website. And you can basically cut and paste

all of your website straight here, or you could upload a particular HTML file,

or at this point you could even copy and paste your HTML straight in here and

check whether or not it's valid. And if it is valid chances are,

very high chances are, that it will work very

well in the browser. So let's take a look and

see let's tested out and see if we could validate the W3C's

own page and if we'd click check and sure enough it tells us that it

is valid or everything is green. Now when you write webpages in the real

world, you can't always say that I'm just going to go ahead and try to support

every browser that exists in the entire webisphere, and that's not going to work

because there's just so many of them and some of them are so outdated. So it's always a good idea to check

the browser statistics and to see. Certainly if you are in an organization

that you know all the browsers that are in that organization

that are being used, all the browsers and

browser versions, that helps. But if you're writing an application for the entire web, it's good to know which

browsers are actually popular, and which browsers are basically

falling off the edge. And if you take a look here,

this is a website, w3schools.com, gives you the browser statistics, and if

you take a look here in September of 2015, Chrome was basically

crushing it's competition. And you can see the next one is almost

three times less market share than Chrome. And you can tell that 65, 66 almost percent of users on

the web are using Chrome so that seems like a pretty important browser

to make sure that your webpage works in. And if you're all the way

at 1.4% of market share. If there some feature that you really,

really want to implement and all the other browsers implemented but one of them that has such a little, such

little market share, doesn't implement. Well, you might want to

decide to go with it anyway. This is something that you

can make a decision about. And last, but not least,

is Google.com or any other, really, search engine that you would like to use. I don't think there's a day that

goes by that, in web development, that you don't have to look

something up on the web. And this is just part of

the day-to-day business. And if you're going to be in web

development, you kind of have to get used to the fact that you have

to constantly look things up. Things are constantly changing,

and obviously, no human being can know

it all by themselves. So to summarize, we went over somewhat

relevant, hopefully, history of HTML, and kind of saw how it effects

us as developers, even today. So I gave you some resources for

validation of HTML. Which kind of gives you a little bit more of a comfort factor that your HTML

will work in browsers properly. And I gave you some resources for feature investigation like the website of

canIuse.com which lets you know whether or not a particular feature is implemented

in a particular version of a browser. And we also went over where you

could look up browser statistics so you can make an informed decision of

what browsers you should even target for your web application to begin with. Next we're going to talk about

the anatomy of an HTML tag. We'll take it apart and see what an HTML

tag is and how to properly code it.