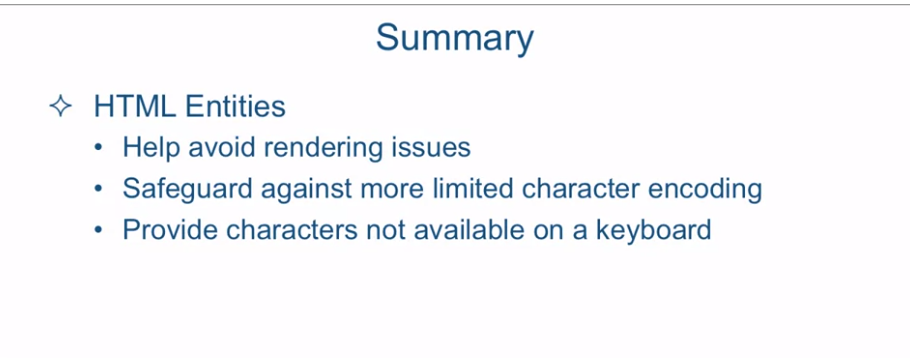


Copyright = &copy;

No breaking space = &nbsp; to avoid breaking of phrases

Quotes = &quot;



[SOUND]

There are many applications for HTML character entity references, however

in this lecture, I will concentrate only on a particular problem that

the entity references can solve for us. Since HTML uses certain characters for

its syntax, we need a way to differentiate between those characters as HTML and

those same characters as content. If we want the browser to

interpret special HTML characters as regular content,

we need a way to escape them. In other words, we need a way to tell

the browser not to interpret them as HTML. Specifically, there are three characters

that should always be escaped to make sure they don't cause rendering issues,

either right away or down the line. And these characters are the following. The < character,

the > character and the &. Instead of using the < character,

you should use the HTML entity which starts with & and

then followed by lt;. So if you put in your HTML, &lt;, what the browser will interpret

this as a < character. And similarly, for

the > character, it's &gt;. And for the & it is &amp;. So let's take a look at some HTML

to see this concept in action. Okay, so we're looking at the document

called html-entities-before.html which is located in

the examples Lectures08 folder. And this document contains a quote

from one of the US presidents, Theodore Roosevelt, which happens to

be one of my favorite quotes of his and a very weird looking h1 content. So we can take a look at it,

it says don't be afraid to be <then a 100% success & >more;. So let's take a look at what

this looks like in the browser. As you can see,

our heading didn't really do so well. It says, Don't be afraid to be more and there's a whole bunch

of words missing here. Well the reason this is going on is

because the browser is interpreting this left angle bracket as

a beginning of a tag. And then it looks at the word then. It doesn't quite understand

what kind of a tag that is and basically skips just about everything

between the left angle bracket and the right angle bracket and

just says the word more. So we could fix that very

easily with substituting the < angle bracket with

the HTML entity reference. And once we're at it, we might as well

substitute the other two characters, the & and the >. We'll save the document,

go reload our webpage, and as you can see, the entire h1 tag content

is displayed on the screen. In reality, HTML contains a whole ton

of different HTML entity references. And obviously we're not going to be

able to go through most of them. However, one particularly common

one is the copyright symbol. And the main reason why we use

an HTML entity reference for it is because it's not really very

readily found on any keyboards out there. However, we could very easily put that in. So let's go ahead and

put it right here after the year. And the copyright entity

reference is just &copy;. So let's save it and

reload it in our browser. And here's the copyright symbol that's

appearing right after the year and before the word copyright. There's another HTML entity reference

that is very commonly used and unfortunately, a lot of

time misused as well. Let's say for

example that in the last sentence here that ends timid souls who

neither know victory nor defeat. Let's say we wanted the words victory nor defeat never to wrap, but

always stay the same. So, what do I mean by that? Let me pull the browser and

make it a little bit less wide. And as you can see, the word victory nor

defeat is split up into different lines. Let's say I didn't want that. For whatever visual reason I

wanted the words victory nor defeat to always stay the same. So if it is going to wrap,

it should either not wrap at all or wrap all together. Well, the way you do that is

with a non-breaking space. And the way you use it is nbsp;. Not breaking space, or

removing all spaces between the words. So now let's take a look what

it looks like in the browser. We'll go ahead and refresh it. And now you'll see that the words victory

nor defeat as we squeeze the browser and make it a little bit less wide, they will

either both drop to the next line or they will stay in the same line, but

they will not be separated anymore. But let me caution about misuse

of this entity reference. A lot of people use this entity reference

if they want to make a space between, let's say the word critic and the word

who, and they will put a few spaces. Let's say they need a few spaces in it. And if I now refresh the browser, I see now got a few spaces between

the word critic and the word who. And that's a total misuse

of this entity reference. If you ever wanted to have spaces,

or extra spaces, between some words in the text, what you would do is you

would probably wrap around some text in a span tag and then apply some margin,

in this case left margin, to the span tag to remove it

the further from the word critic. But you should never use

the non-breaking space HTML and the reference For that purpose. Let me show you another HTML entity

reference that is very commonly used. And it's especially useful when somebody's

trying to write an HTML based email. Since the email clients are notorious for

using a much more limited character set than UTF-8, some of the characters

sometimes get messed up. So let me show you what I mean. Let's go to the web page and

instead of viewing this web page in UTF-8, we'll change the encoding to something

more limited like Windows-1252. And if you notice, some of the quotes

became these very weird characters. So how do we solve that? Well, the way we could solve it is by

using an HTML entity references of quote. And if we preview this in a browser again,

and let's refresh the browser. And now there are quotes. Maybe they are not the same type

of curly quotes as before, but they're quotes nevertheless. So in summary,

we looked at HTML entities, and they help avoid rendering issues, and

especially with those three characters that HTML can try to render instead

of thinking of it as content. We also saw that sometimes

you can safeguard with these HTML HTML entities safe guard against

more limited character set encodings. And you could provide characters

not available in the keyboard. For example, like we provided

the character of copyright that was at the end of the document. That certainly is not on any of

the keyboard that I've seen, but you could still display that

character using the HTML entity. Next we're going to talk about

making text hyper with linking.