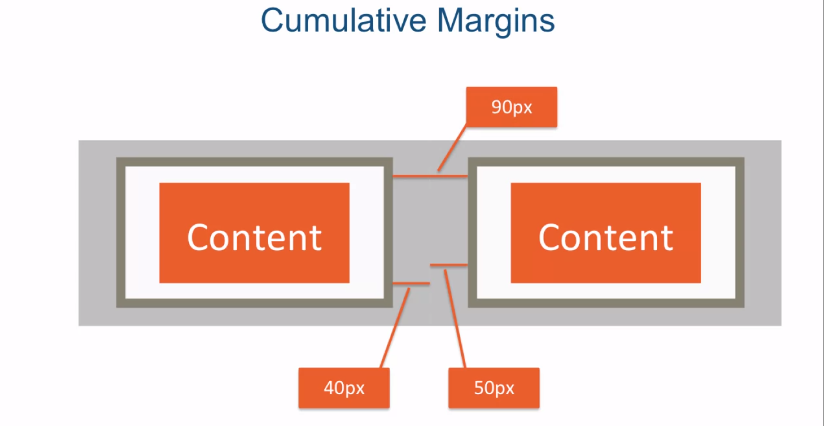


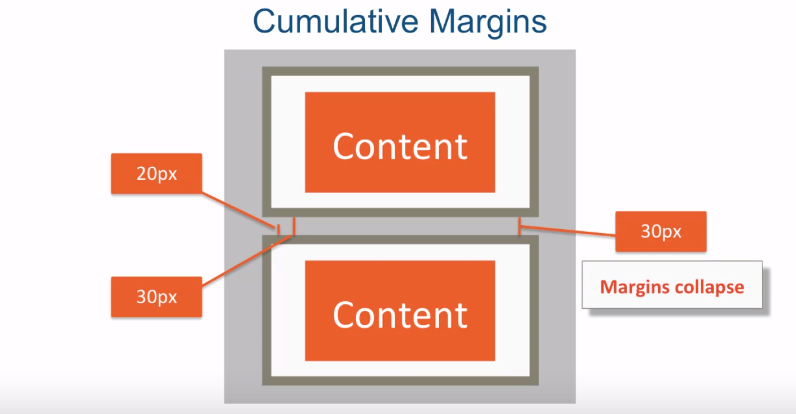
**BOX-SIZING: BORDER-BOX; Sirve para definir que los tamaños que nosotros seleccionamos sean los de la caja y no los del contenido, si usamos esto la caja utiliza los valores que pasamos y el resto se ajusta a eso. Si usamos content-box estamos definiendo el tamaño del contenido, no del borde.**

**BOX SIZING NO ES INHERITABLE.**

**NECESITAMOS USAR EL SELECTOR \***

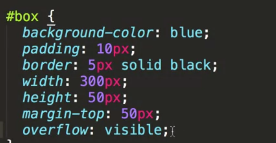
**EL \* TE DICE: NO TE ESTOY DICIENDO QUE HEREDES SINO QUE SELECCIONES CADA ELEMENTO. AGARRA ESTA PROPIEDAD Y VALOR, Y APLICASELA A TODO.**

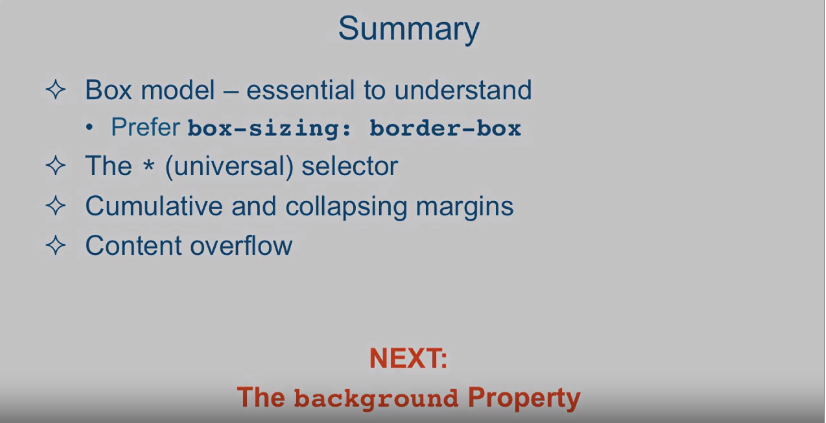
****

****

**THE LARGER MARGIN WINS EN LOS MARGENES VERTICALES.**

OVERFLOW: COMO MANEJAR EL CONTENIDO QUE EXCEDE LA CAJA.





In HTML every element is considered a box. However, there's more to that box than

just the content that the element wraps around. Let's take a quick look

at a simple diagram. Besides their [INAUDIBLE] content,

each box consists of padding, border as well as margins. The box model refers to the components

that make up an HTML box as well as the rules that govern how

these box components affect the layout as well as how width and

height of the box are calculated. Let's jump into the code and

take a look at some examples. Okay, so here I am in subline text and I'm looking at the file

called box-model-before.html. And it's located in

the examples Lecture19 folder. Let's quickly examine the HTML

structure of this document. It's actually pretty simple. It just has one h1 here. Has a div that has an ID box and

inside of that div there is another div with an ID content and has some Loren

ipsum, basically some dummy text in it. In addition to that I colored the box

itself with a background color blue and the content which is the div right here. This content. I colored it with this color

which is basically light green. And just so we have some more

interesting colors in the background I colored the body to be gray. Okay, so here is the representation

of our page in the browser. And since div is a block level element, it tries to fill in the entire

parent in terms of width. So it's trying to fill up the entire

width here in the browser. But there's something strange

going on if you take a look. There's some space that's gone

in here right before Lorem and here at the end of our content

beginning and there's spacing going on. So let's take a look at the chrome

developer tools real quick and figure out what that space is about. Okay so we'll go ahead and

choose the box, that's our box. And we'll take a look

as to what's going on. There doesn't seem to be any spacing

that's set on the box itself. Well how about the body

itself in the body tag? Go and here we go. See we'd realized that the body tag here

has a margin all around of eight pixels. Now where is that eight pixel coming from? Or if we take a look here,

it's telling you where it's coming from. It's coming from the user agent style. Style sheet. Which means it's the browser itself. It's the default browser styles. So that's actually pretty common that

the browsers do stuff like that. And what we're going to do is in order for it not to affect what we're

doing we're going to reset it. And it actually resets

on the web out there, plenty of resets on the web how

to reset default browser setting. So we'll go ahead and do margin 0 and we'll also do padding 0 so

we won't be affected by any of that. And when we do that and we save and

refresh, you see that now our content is flush with the actual borders

of the browser window. Okay, so moving on. So we have our content. And you can see the context is green. Now we're not seeing the background of

the box the content is actually sitting in because we would have seen something blue

in the background and always see green. And that is because the inner box,

the inner content div is covering up the outer one completely

because they are the same size. They both try to fill up their parent

which is in this case the body tag. Both try to fill it up all the way. And since this one basically sit

it on top or inside of this one, it's covering up the blue background. Let's go ahead and

set some padding on our box. Let's do padding, and we'll do 10px top. 10px right, 10px bottom, and 10px left. Now we set that and by the way,

there's obviously a shortcut for this. You could actually remove

all this 10px right here and that would mean exactly the same thing and

I let you actually look up as to all the different shortcuts that you could

take, but I'll leave this here for now. So point is, it's always goes top, right, bottom, left and let's go ahead and

save that and refresh the browser. And voila, now we've seen the padding. And what the padding is doing is basically

giving some padding around our content and our content being this div tag right here

with the content text, is the one that's being kind of squeezed all around and

the padding is showing up all around and now you can see the blue because that's

the background color of our box. Okay so next,

let's give our box some border. Let's go ahead and say border, and we'll give that border that's

going to be 3px thick. And it's going to be solid as opposed

to dashed or something like that. And there's other options here

you're welcome to look up. And we want to give it the color black. Okay.

When we save that, and when we refresh the browser,

you could see that the border showed up. And in fact,

let's make it a little bit thicker. Let's make it a five pixels border. And let's make it like that. Okay and then last but not least,

let's give it some margins as well. Let's go to margin and let's go ahead and

use the shortcut notation which is giving you 40 pixels which means I'm

saying I want 40 pixels all around. So top, right, bottom, and left. So if I go ahead and refresh that. You could see now there's

40 pixels on every side. And, well, you could see it on the bottom,

per say, and you can't really necessarily see it here. But there's 40 pixels on every side. Okay.

Well, up until now, we've sort of let the content and

some of our margins, and so on dictate how big the actual box is or what about

if we actually set this width our self. Let's set the width to be let's

say 300 pixels and we save and refresh the box became smaller but

let's go ahead and examine that box. Let's click on the inside Chrome Developer

tools, now let's take a look. So it's showing us that our box

is in fact 300 pixels wide and since we didn't restrain our height,

it went ahead and wrapped around the text. And basically the height will

adjust unless we restrain it. A manually by specifying the height, it will adjust to however

much it needs to fit. So if we make it a little bit small,

let's make it let's say 100 pixels, we'll refresh you'll see the height will

become a little bit bigger, the height will become 72, because now we squeezed

it enough that extra lines got added. So now the content width is 100 but

the height is larger. Let's go ahead and

put it back to 300 and refresh. And now if you look at this breakdown we come to our first

pretty interesting point. But first of all let's go

through all the components. Here this one is the pure content right

here that I'm mousing over right here. And then if I go up a little bit,

it'll highlight for me in the browser the padding

that's going on there. And if I highlight, if I go out a little

bit more, I'll go straight up to the border, right, and if you can't really

see the margins if you roll over to the margin side voila now you see what

the margins are gone and even though to the right side showing you a little bit

more than really margin is going on. But the 40 pixels all around still stands. Now what's interesting to

find is well how big is this box exactly as we going to start

putting more and more boxes right. We going to start making layouts

of different HTML components and to get our UI just right,

we need to find out how big is each box. Well, we sort of specified that we

want the box to be 300 pixels wide. The problem is that it's

not really 300 pixels wide. Let's take a look. If we take a look here it says 300 pixels

wide, but we're forgetting there's a border around it or the padding around

it, and the padding is 10 pixels on one side And ten pixels on the other side but

that's not all, there's also a border around it as well that is 5 pixels

wide and 5 pixels wide on this side. So really, it's 300 plus 10 plus 10

that's 20, 320, plus 5 plus 5 that's 330. So if you look at the actual visible

border of this box, this box, this distance is not actually 300. This distance is 330. And the interesting part is,

or kind of annoying part, is that, depending on what border and

padding you set, this width will change. So let's take a look. Let's say if I make the border

something really crazy like 20 pixels. So go ahead and refresh. Now the border is 20 pixels so

now my box end all the way there. And if I make the padding even bigger, let's say I make the padding this one is

the right so I make the padding 30 and on the left I make the padding 30 as well,

and I take a look at the box. Take a look at how it's going to whoa,

it became bigger again. So even though we said I want

the width of the box to be 300, it turns out that we aren't really

setting the width of the box, we're actually setting

the width of the content. And that's this piece

right here in the green. Well it turns out for layouts, and just in

general sanity this is kind of annoying. You really want to change that and

CSS3 actually did change that, it turns out that by default,

the box sizing is actually a property of every HTML element and by default

it's actually set to content box, which means when you specify

it's height and width, you're specifying the height and width of

the content box, not the entire thing. Well CSS 3 came out with a new value for

that property, and that's called border-box. Let's go ahead and set that right here. We'll say box-sizing, and

we'll say border-box. And when we refresh let's take a look,

look at that, it became smaller. So now,

if you take a look at the breakdown, you could see that even though it was

specified the width would be 300. That width is guaranteed to be

the width of this box right here from the edge of this border. To the edge of this border. So if you take a look here,

200 plus 30 plus 30, that's 260, plus 20 plus 20,

that's another 40, that's 300. And if we change it back to content-box, which is the default,

it will jump back to its original size. But the truth of the matter is, all

the modern frameworks like Bootstrap and others use box,

border box as its sizing mode, and as you plan and work with your

layouts this is the same choice to make. So, make sure you always stay

with a box sizing of border box.