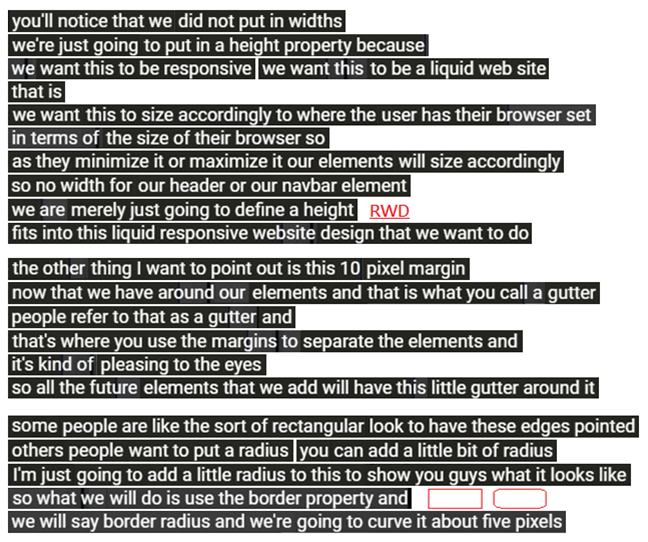
***layout\_A06.png***



you'll notice that we did not put in widths.

we're just going to **put in a height property** because

we want this to **be responsive** we want this to be a liquid web site

that is

we want this to size accordingly to where the user has their browser set

in terms of **the size of their browser** so

**as they minimize it or maximize it our elements will size accordingly**

so **no width for** our header or our navbar element

we are merely just going to define a height

fits into this liquid **responsive website design** (known as **RWD**) that we want to do.

the other thing I want to point out is this 10 pixel margin

now that we have around our elements and that is what **you call a gutter**

people refer to that as a gutter and

that's where you **use the margins to separate the elements** and

it's kind of pleasing to the eyes

so all the future elements that we add will have this little gutter around it .

some people are like the sort of **rectangular** look to have these **edges pointed** ,

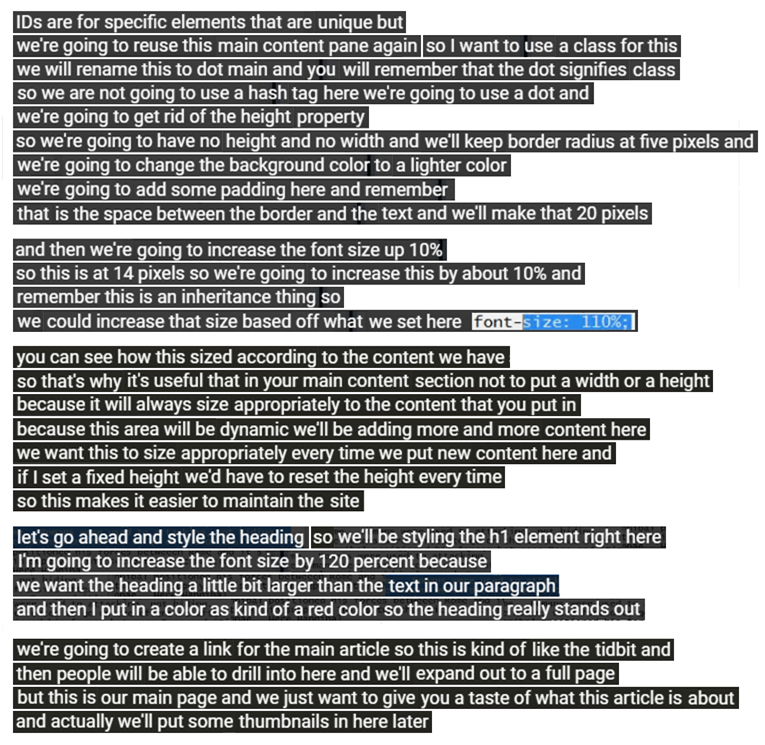
others people want to put a **radius ,** you can add a little bit of radius .

I'm just going to **add a little radius** to this to show you guys what it looks like

so what we will do is use the border property and

we will say **border radius** and we're going to curve it about five pixels

***layout\_B06.png***



IDs are for specific elements that are unique but

we're going to reuse this main content pane again so I want to use a class for this

we will rename this to dot main and you will remember that the dot signifies class

so we are not going to use a hash tag here we're going to use a dot and

we're going to get rid of the height property

so we're going to have no height and no width and we'll keep border radius at five pixels

we're going to change the background color to a lighter color

we're going to add some padding here and remember

that is the space between the border and the text and we'll make that 20 pixels

and then we're going to increase the font size up 10%

so this is at 14 pixels so we're going to increase this by about 10% and

remember this is an inheritance thing so

we could increase that size based off what we set here font

**-SIEO.**

you can see how this sized according to the content we have

so that's why it's useful that in your main content section not to put a width or a height

because it will always size appropriately to the content that you put in

because this area will be dynamic we'll be adding more and more content here

we want this to size appropriately every time we put new content here and

if I set a fixed height we'd have to reset the height every time

so this makes it easier to maintain the site

let’s go ahead and style the heading so We'll be styling the h1 element right here.

I'm going to increase the font size by 120 percent because

we want the heading a little bit larger than the text in our paragraph

and then I put in a color as kind of a red color so the heading really stands out

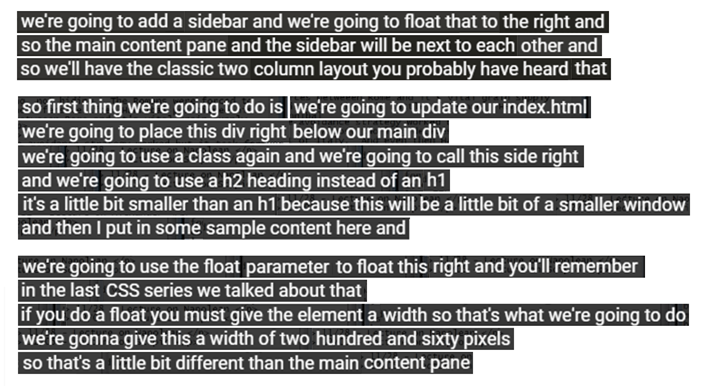
we're going to create a link for the main article so this is kind of like the tidbit and

then people will be able to drill into here and well expand out to a full page

but this is our main page and we just want to give you a taste of what this article is about

and actually we'll put some thumbnails in here later .

***layout\_C00.png***

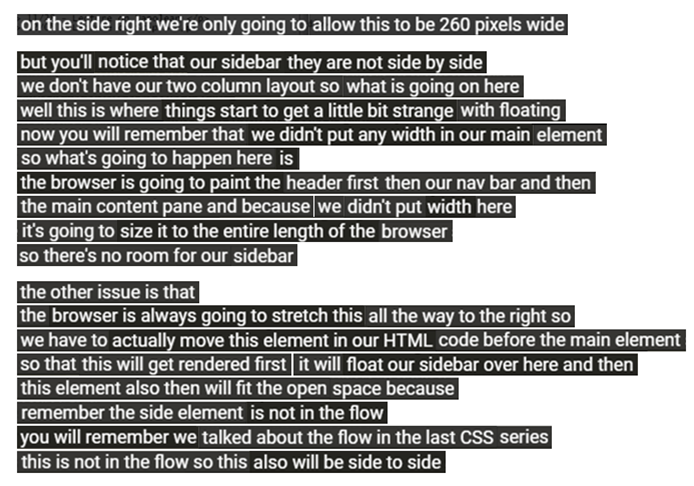


we're going to add a sidebar and float that to the right, so the main content pane and the sidebar will be next to each other. so we will have the classic two column layout you probably have heard that. so the first thing we're going to do is we're going to update our index.html and we're going to place this div right below our main div and use a class again and we're going to call this “sideright”.

we're going to use a h2 heading instead of an h1. it's a little bit smaller than an h1 because this will be a little bit of a smaller window and then I put in some sample content here and we're going to use the float parameter to float this to the right.

you'll remember in the last CSS series we talked about that, if you do a float you must give the element a width so that's what we're going to do. we're gonna give this a width of 260 pixels, so that's a little bit different than the main content pane.

***layout\_C09.png***



on the side right we're only going to allow this to be 260 pixels wide but you'll notice that our sidebar they are not side by side, we don't have our two column layout so what is going on here. well this is where things start to get a little bit strange with floating.

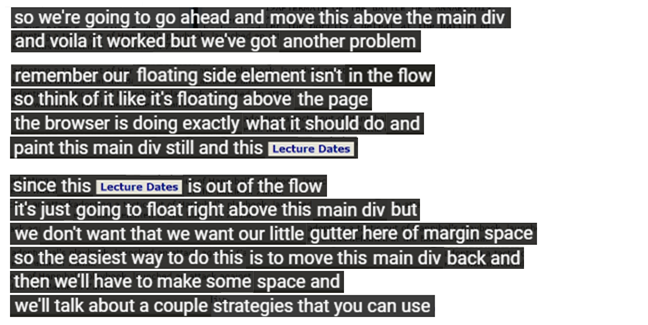
now you will remember that we didn't put any width in our main element, so what's going to happen here is the browser is going to paint the header first then our nav bar and then the main content pane and because we didn't put width here, it's going to size it to the entire length of the browser so there's no room for our sidebar.

the other issue is that the browser is always going to stretch this all the way to the right.

So we have to actually move this element in our HTML code and put it before the main element. so that this will get rendered first, it will float our sidebar over here(the main element) and then this element also then will fit the open space because remember the side element is not in the flow.

you will remember we talked about the flow in the last CSs series, this is not in the flow so this also will be side to side.

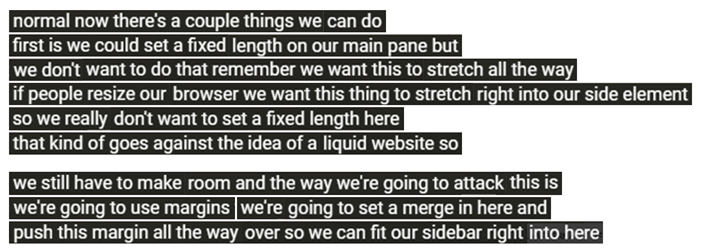
***layout\_C13.png***

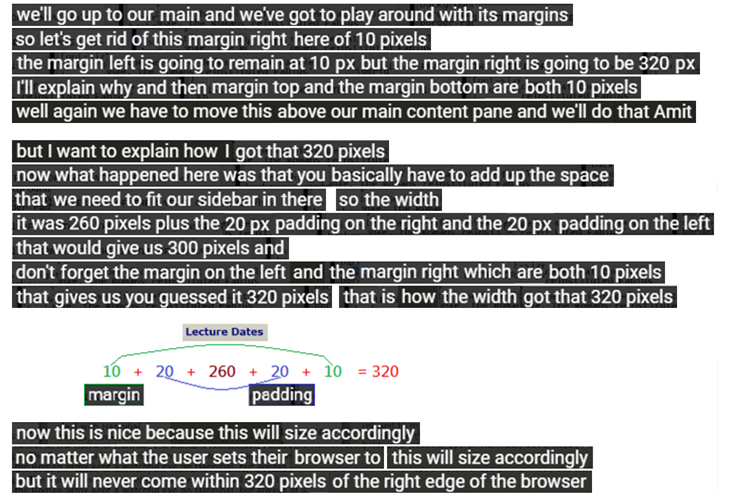


so we're going to go ahead and move this “sideright (Lecture Dates)” above the main div and voila it worked but we've got another problem, remember our floating sideright element isn't in the flow. so think of it like **it's floating above the page**. the browser is doing exactly what it should do and paint this main div still and this “sideright (Lecture Dates)” element.

since this “sideright (Lecture Dates)” is out of the flow, it's just going to float right above this main div but we don't want that, **we want our little gutter here of margin space.** so the easiest way to do this is to move this main div back and then we'll have to make some space and we'll talk about a couple strategies that you can use.

***layout\_C26.png***





Now there're a couple things we can do. first is we could **set a fixed length on our main pane** **but we don't want to do that**, remember we want this to stretch all the way. If people resize our browser we want this thing to stretch right into our “sideright” element. So **we really don't want to set a fixed length** here that kind of **goes against the idea of a liquid website**.

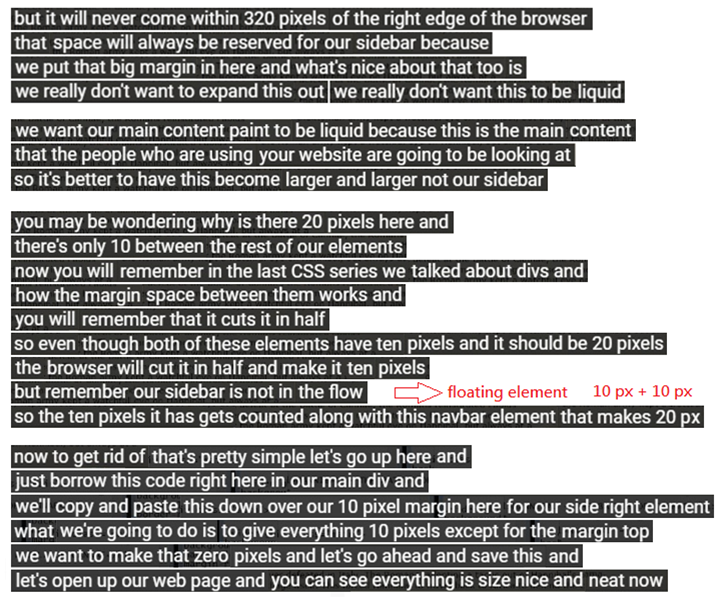
So we still have to make room and the way we're going to attack this is, we're going to use margins. **We're going to set a merge in here and push this margin all the way over**, so we can fit our sidebar right into here. We'll go up to our main and we've got to play around with its margins.

So **let's get rid of this margin right here of 10 pixels** and **the margin left is going to remain at 10 pixels** but **the margin right is going to be 320 pixels**. I'll explain why and then **margin top and the margin bottom are both 10 pixels**.

Well again we have to move this above our main content pane and we'll do that , but I want to explain how I got that 320 pixels. Now what happened here was that you basically have to add up the space that we need to fit our sidebar in there, **so the width**, **it was 260 pixels plus the 20 pixels padding on the right and the 20 pixels padding on the left that would give us 300 pixels** and don't forget the margin on the left and the margin right which are both 10 pixels. **That gives us you quested it 320 pixels,** that is how the width got that 320 pixels.

Now this is nice because this will size accordingly, no matter what the user sets their browser to this will size accordingly **but it will never come within 320 pixels of the right edge of the browser**.

***layout\_C39.png***



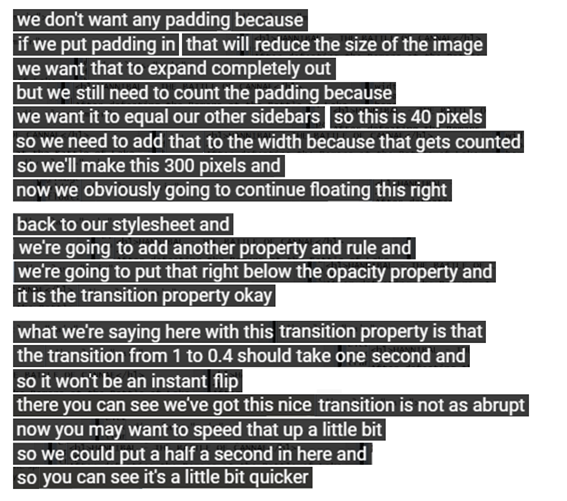
**That space will always be reserved for our sidebar** because **we put that big 320 pixels margin right in here** and what's nice about that too. **we really don't want to expand this sidebar out**.

**we want our main content paint to be liquid because this is the main content that the people who are using your website are going to be looking at**. So it's better to have this become larger and larger not our sidebar.

you may be wondering **why is there 20 pixels here and there's only 10 between the rest of our elements**. Now you will remember in the last CSS series we talked about divs and how the margin space between them works and you will remember that it cuts it in half.

So **even though both of these elements have 10 pixels and it should be 20 pixels**. **The browser will cut it in half and make it 10 pixels but remember our sidebar is not in the flow**, so the 10 pixels it has gets counted along with this **navbar element (10 pixels+ sidebar 10 pixels)** that makes 20 pixels.

Now to get rid of that's pretty simple let's go up here and just borrow this code right here in our main div. We'll copy and paste this down over our 10 pixel margin here for our side right, **what we're going to do is to give everything 10 pixels except for the margin top**. we want to make that margin top zero pixels and let's go ahead and save this and let's open up our web page and you can see everything is size nice and neat now.



We don't want any padding because if we put padding in, that will reduce the size of the image.

We want that to expand completely out but we still need to count the padding because we want it to equal our other sidebars, so this is 40 pixels.

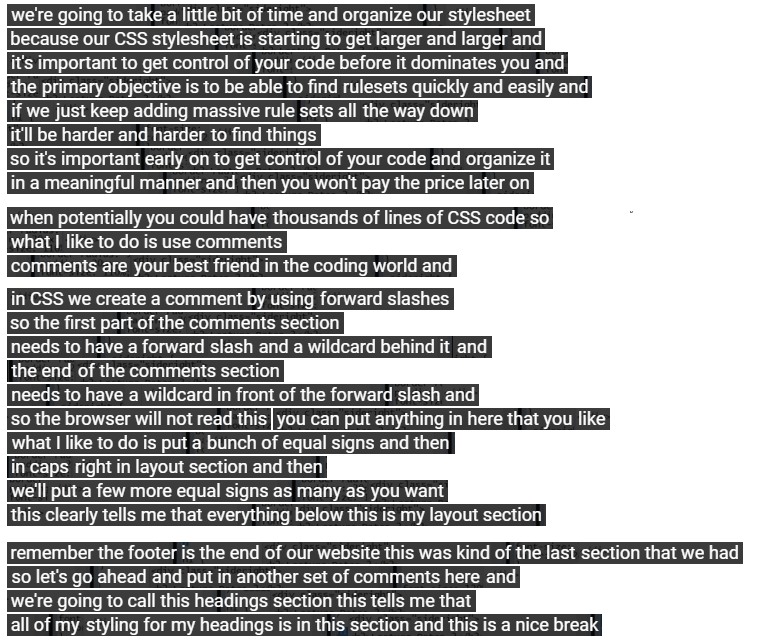
So we need to add that to the width because that gets counted, we'll make this 300 pixels and now we obviously going to continue floating this right.

Back to our stylesheet and we're going to add another property and rule and we're going to put that right below the opacity property and it is the transition property.

Okay, what we're saying here with this transition property is that the transition from 1 to 0.4 should take one second and so it won't be an instant flip.

There you can see we've got this nice transition is not as abrupt.

Now you may want to speed that up a little bit, so we could put a half a second in here and so you can see it's a little bit quicker.



We're going to take a little bit of time and **organize our stylesheet** because our **CSS stylesheet is starting to get larger and larger** and it's important to get control of your code before it dominates you.

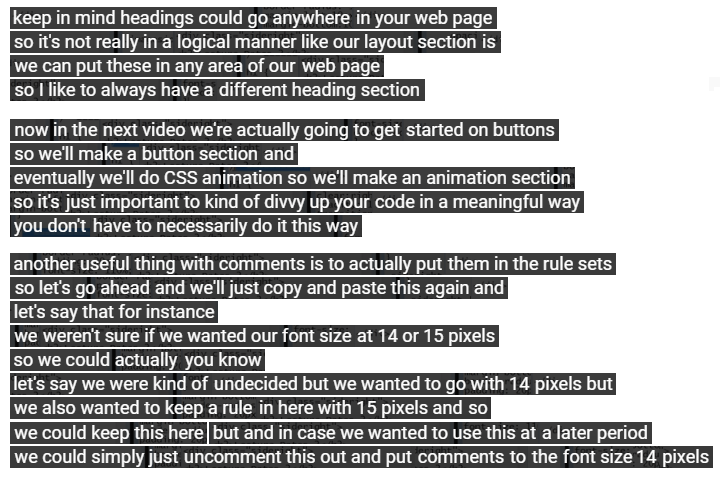
**The primary objective is to** **be able to find rulesets quickly and easily** and if we just keep adding massive rule sets all the way down, it'll be harder and harder to find things.

So it's important early on to get control of your code and organize it in a meaningful manner and then you won't pay the price later on. When potentially you could have thousands of lines of CSS code, **so what I like to do is use comments**. **Comments are your best friend in the coding world and in CSS we create a comment by using forward slashes.**

So **the first part of the comments section** needs to have **a forward slash and a wildcard behind it (/\*)** and **the end of the comments section** needs to have **a wildcard in front of the forward slash(\*/)** and so the browser will not read this you can put anything in here that you like.

What I like to do is put a bunch of **equal signs** and then in caps right in layout section and then we'll put a few more equal signs as many as you want, **this clearly tells me that everything below this is my layout section**.

Remember **the footer is the end of our website** this was kind of the last section that we had. So let's go ahead and put in another set of comments here and we're going to call this headings section this tells me that all of my styling for my headings is in this section and this is a nice break.



Keep in mind headings could go anywhere in your web page, so it's not really in a logical manner like our layout section is. We can put these in any area of our web page, so I like to always have a different heading section.

Now in the next video we're actually going to get started on buttons, so we'll make a button section and eventually we'll do CSS animation, so we'll make an animation section. It's just important to kind of divvy up your code in a meaningful way. you don't have to necessarily do it this way.

Another useful thing with comments is to actually put them in the rule sets, so let's go ahead and we'll just copy and paste this again and let's say that for instance, we weren't sure if we wanted our font size at 14 or 15 pixels, so we could actually you know, let's say we were kind of undecided but we wanted to go with 14 pixels but we also wanted to keep a rule in here with 15 pixels and so we could keep this here but and in case we wanted to use this at a later period, we could simply just uncomment this out and put comments to the font size 14 pixels.