

Building Better Websites: 3 Key Principles for Efficiency, Consistency, and Maintainability - Kevin Geary

[00:00:19.370]

With Bluehost, you can get your site up and running fast. It's recommended by Wordpress.org. Setup is simple and personalised. Bluehost's team of experts can also help or build it for you. Start your out-of-the-blue idea today at bluehost.com.

[00:00:34.810]

Welcome to page Builder Summit 6.0. My name is Kevin Geary. I'm the creator of Automatic CSS, and I'm also a WordPress and web design educator. And one of the biggest problems, one of the biggest challenges that I see in modern web design is web designers using page builders to assemble websites as quickly as they can, working from client to client to client. But at the end of the day, what clients are left with is more of a mess than anything else.

[00:01:05.000]

The websites that are built for them are not scalable, they're not maintainable, they're not very professional and they require a lot of manual labour to make changes from page to page to page. What I want to do in this training is I want to give you three principles to approach all of your website builds going forward. And if you follow these three principles, you are going to end up with websites that are more professional, easier to scale and easier to maintain. This is going to be better for your clients and it's going to be better for you as well. So what we're going to do is we're going to hop right in and we're going to talk about principle number one, which is always loop important content. Static content, which this is the majority of content that most people using page builders are using.

[00:01:55.820]

They're putting static content everywhere throughout a website. Static content makes sites impossible to maintain and scale. If, on the other hand, we use what are called custom post types, we can create a single source of truth for every piece of content. That can then be looped and taxonomized and bi-directionally related to other content. It also makes existing content easier to edit, it makes new content easier to add.

[00:02:24.900]

This is the approach that we want to take with all key content. By the way, we are using WordPress, which is a content management system. If we're not using custom post types and dynamic data, custom fields, query loops, things like this, we're not using WordPress as a content management system. We're really using WordPress as basically a junk drawer where we're just tossing everything into posts and pages and nothing is really organised, nothing is well structured and we miss out on a lot of opportunities for scalability and maintainability. So what you see here is a grid of avatar cards, right?

[00:03:03.610]

This could be team members or CEOs or whatever it happens to be. What I'm going to do is I'm going to edit this with Bricks. Now I'm using bricks builder. It doesn't matter what page builder you're using, as long as the page builder you're using allows you to do the things that we're going to be talking about in this training.

[00:03:20.060]

If you're using a page builder that doesn't allow you to do these things, then you are going to be severely limited in the professionalism you can put into the websites and the scalability, the maintainability, all the things that we're going to talk about over and over and over again throughout this training. There's many, many page builders that do allow you to do these techniques. So it's not just bricks, but make sure that you are using a good one. Okay, so what you see right now is a grid of cards. There are three cards in the grid.

[00:03:50.240]

And right now, if I wanted to add team members, this is how many, many people approach web design. They just have the cards and they manually come in. This is static content. They manually come in and they say like, person number one. And then they say person number two, and then they say person number three.

[00:04:09.810]

Okay? And then if the title needs to change like we'll do title number one. And then this will be title number two. And then this will be title number three. And then they'll come in and they'll put in the person's actual photo.

[00:04:23.790]

So what we're going to do in a minute is we're going to build sites for a yoga studio. So I'm just going to throw these photos in here real quick and you can see that. Yeah, it feels kind of quick, right? It feels kind of quick. It feels kind of easy.

[00:04:38.420]

I mean, this is why people do this. And then if another instructor needs to be added, all we're going to do is come in and we're going to duplicate the card, the last card there. And then we're going to come in and swap the I don't have a fourth photo, but we would swap the photo and then we would put person number four. And then we would put their title in. And then if somebody quit, right, so this first person quit, so we're going to come in and we're just going to delete their profile card.

[00:05:04.520]

And that's going to leave these other profile cards here. And this is how I think most websites in the modern era are managed. And you might be thinking to yourself, well, Kevin, what's the problem with that? That seems very easy, very straightforward. In fact, I think I can even teach the client to do that right.

[00:05:22.890]

And if these instructor cards only exist on this one page, and we pretend for a moment that instructors are not related, for example, to the classes that they teach, okay, then this might be an okay approach, but we actually build websites in the real world where these instructor cards are likely going to show up on multiple pages across the website. And it does make sense that an instructor would be related to specific classes that they teach, which means there's going to be references to these instructors when those classes are mentioned and if this is happening on multiple pages throughout a website, this static approach to managing content is not a viable approach. While it seems like it's easy and straightforward, it actually causes a tremendous amount of problems as the site gets bigger and as the site needs to be maintained into the future. When an instructor leaves the studio and a new one joins, we have to suddenly now with this static approach remember, where was that old instructor referenced?

[00:06:34.570]

Everywhere. Right. What posts were they referenced in? Which pages were they referenced in? What cards were they referenced in?

[00:06:40.990]

It's all over the place and there's no way to manage that content that way. There's no way to manually keep track of all that, nor would you want to, because now I have to manually go to every single reference of that person and I have to remove them, and then I have to take the new instructor and I have to manually go reference them on every other page and post and card and everywhere that that person needs to be referenced. And so what seems like a really easy, simple, straightforward approach to managing instructors on a yoga site suddenly becomes an absolute nightmare scenario in the real world on a multi page website, where all of these different things are happening. And that's ultimately what's

going on with most websites that are just thrown together in a page builder. The client ends up with a nightmare scenario and we have to avoid that at all costs.

[00:07:36.490]

Professionals do not create nightmare scenarios. So what we want to do is we want to realise, okay, what I need to do is use WordPress as a content management system. I don't want to statically manage content. What I want to do is I want to loop content and I want to use dynamic data so that a page is more or less dynamically generated. And this sounds very technical and it sounds very complicated.

[00:08:03.600]

The good news is it's actually very easy and the payoff is tremendous relative to the actual work required. You just have to know the approach. And once you know the approach, it's actually very easy. So I'm going to teach you the approach right now. We're going to go into the WordPress dashboard and I'm using something called advanced Custom fields.

[00:08:25.230]

And what Advanced Custom Fields allows me to do is create something called post types. And then I can also create custom fields that are attached to those post types. I'm going to show you all of this. It's going to make perfect sense in just a minute. I can also categorise these things and I can also add relationships from one post to another post, and even one post type to another post type.

[00:08:47.530]

Let's start out with just post types so that you can wrap your head around this and it's going to make it so easy. Okay, so I'm going to hit add new post type and what we're going to add is Instructors. We're going to add a way for us to add instructors to this website. Notice that I have posts up here, I have pages here, but these aren't a type of content. This is just generic.

[00:09:08.580]

Like I said earlier, it's a junk drawer, right? What we want is a specific drawer, like an organised drawer, like a drawer with a label that says Instructors. And then we put cards in there and it's like think you're at the library, right? Those little index card organiser things. You just take your instructor cards, you organise them in there and now they're all in one place.

[00:09:28.520]

All of your instructors are in one nice organised drawer. That's exactly what we're doing here, okay? So we have instructors. It's asking for a singular label, which would be Instructor. And then it's going to make this post type key for me.

[00:09:41.610]

I'm just going to go ahead and make it plural and then I'm going to go ahead and we're going to do nothing else here because I want to make this as easy as it can possibly be. We're just going to hit save changes. And you can see right now we get this new area called Instructors. And I'm free to just go in here and say, add new Instructor, just like I was adding a new post, just like I was adding a new page, right? You saw that.

[00:10:00.850]

That is not difficult in the slightest, okay? So what I'm going to do is I'm going to go over to Featured Image here and I'm going to add the featured image for one of my instructors and I'm going to note their name. This is Daisy Lou right here. Okay. And I'm going to say set featured Image And then I'm just going to say, hey, this is Daisy Lou, all right?

[00:10:18.100]

And then what I'm going to do is I'm going to hit Publish and I'm going to go back and you're going to see that I have an instructor now called Daisy Lou. Now I'm going to add another one just like I'm adding another post, and we're going to be adding Clara Brown. So I'm going to come in here and say Clara Brown and I'm going to go ahead and hit Publish and I'm going to come back. And now you see, I have two instructors in my drawer, right? I'm going to hit add new here.

[00:10:39.560]

We're going to go over and we're going to add our last instructor. This is going to be Caroline Grace, all right? Caroline Grace. And I'm going to go ahead and hit Publish here. That's perfect.

[00:10:49.650]

And then I'm going to go back and you can see all of my instructors are added. Now what I really want is I want a short bio for each instructor. And so what I'm going to do now is I'm going to add a custom field to my instructor cards that allows me to put in a quick little Bio. So what I'm going to do is go back to advanced custom fields. This time I'm going to go to field groups and I'm going to click Add New.

[00:11:15.270]

And I'm going to say that this is going to be assigned to our instructors. So all the fields that I want to put here are going to be assigned to our instructors. Now imagine physical cards, right? You have the instructor's name, you have their photo over here that you can just paste on the card. And then you're just going to create a little area for a quick bio, like a short bio.

[00:11:35.170]

So I'm going to come in here and I'm just going to use a text area field, or I can use a WYSIWYG field, a what you see is what you get field. And I'm going to name this the WYSIWYG, by the way, will allow me to bold things and italicise things very easily. So it's a decent field to use here. And I'll just say Instructor Bio right here, okay? And I'm going to go ahead and assign.

[00:11:59.270]

I've said that this is going to be assigned to our instructors and I added an Instructor Bio field, but I haven't actually told the fields to be assigned to instructors. That happens right here on this row. So it says if post type is equal to instructor, then I can go ahead and hit save changes. Now if you're familiar with advanced custom fields and you're familiar with custom post types and all this, don't leave the training. There's two more very important principles that we're going to cover.

[00:12:26.020]

This is just principle number one. Okay? So I have assigned Instructor Bio, really all of this field group, I've assigned to our instructor. So what, I can go over here and I can go to Daisy and I can see now, wow, look at this. I have a new field where I can paste Daisy's Bio.

[00:12:42.760]

So what I'm going to do here is I'm just going to go to website Ipsum for now. We're not going to take the time to write real Instructor BIOS and I'm just going to paste this in and we're going to hit update, and then we're going to go back. And then what I'm going to do is go to Clara around and I'm going to give her a Bio and we're going to update and we're going to go back. And then what I'm going to do is go to Caroline Grace and I'm going to give her a Bio and then we're going to go back. So now each instructor has a name, they have a Bio and they have a photo.

[00:13:08.740]

And I think that's good enough for right now. So now you're going to see, I have these things in the database as custom fields or as custom post types. And I have custom fields attached to those custom post types. The question everybody has now is how do I get those to actually display on the page. Now, every page builder is going to take a different approach to this, but again, you should be using a page builder that allows you to do this.

[00:13:34.340]

Bricks is one of the easiest. What I'm going to do is I'm going to come in and I'm going to see that I have three profile cards. I'm going to delete two of them. I only need one card. And this card is going to serve as like the template for our instructor cards.

[00:13:50.710]

And then I'm just going to loop them, which means it's going to duplicate them equal to the number of times as there are instructors. And then what we're going to do is we're going to dynamically insert the data that goes with each card. Okay? So let's take a look at how easy this is. So I'm grabbing profile card foxtrot right here and I'm going to come in and I'm going to just tick this little box that says use query loop or it's a toggle, okay?

[00:14:16.530]

And immediately you see that it just hey, it starts duplicating the cards, but now I have to tell it exactly what it's looping through. What data are you going to loop through? So we're going to open this little query loop icon and we're going to do a post, okay? These are posts. The post type, however, is not just posts, okay?

[00:14:36.630]

And you see that I don't have any other types here, even though I created one called Instructors. Why is that? It's because this Builder has been sitting open while I went in the back end and did this other stuff. The Builder doesn't have access to that new data yet. So what I'm going to do is save our progress and I'm just going to refresh the Builder really fast.

[00:14:54.700]

And now when I go click on the query loop indicator right here, it's going to allow me to select the new post type instructors because now the Builder actually has access to the new information that's in the database. So I'm going to say, hey, we're going to loop through Instructors right here and now. By the way, I have the ability to order them. Like, I can alphabetize them according to their name. I can order them by a lot of different things.

[00:15:17.580]

So I'm instantly given more power as to how this grid is filtered and organised because I'm using a loop, right? That's why I say always loop important content. I'm not going to worry about the order right now. I'm not even going to worry about the post per page. I'm going to leave pretty much everything at the default and I'm going to hit save.

[00:15:36.220]

So that's about as easy as it can get, right? We're not doing anything super technical here. Now the only problem is it's looping, but it's using all static content still. It's not using any of the content from the actual

post themselves, like the person's avatar the person's name and their title right here. And the title we don't even have a custom field for yet.

[00:15:58.840]

So we'll take a look at what we need to do for that in just a second. So what I'm going to do is I'm going to delete the static content. That's always step one, get rid of the static content. We need to replace that now with dynamic content. So I'm going to come in and click this little lightning bolt.

[00:16:12.140]

Again, this is different in every page builder, but the page builder you're using should allow you to do this. I'm going to click the little lightning bolt here and I'm going to say that we are going to dynamically pull in the featured image and look at this. My grid of instructors is building itself. Okay? And then I'm going to go in and I'm going to say, all right, this doesn't really need to be a 1920 image.

[00:16:35.160]

I think we can get away with 960 here. And I'm going to go ahead and hit save and let's just go to the front end and see if this is working in real time. And it is. I've got all three of my instructors. The problem is it doesn't say their name, it says person number two.

[00:16:49.810]

So we effectively need to go do the same thing here. What's step one? Get rid of the static content. Step two is to get dynamic content inserted. And now I'm going to search for Instructor.

[00:17:01.150]

Actually, Instructor name is actually the title of the post. Okay, so this is the post title. I didn't make a custom field for this, I'm just using the post title. So what I'm going to do with my dynamic data is I'm just going to reference the post title and look at this. Let's go back to our front end of our website.

[00:17:21.810]

Again, my whole grid of instructors is building itself. I'm not building it, I just built one card and I'm just telling it, hey, dynamic data for the image, dynamic data for the post type. Now what about this title right here? All we need to do is we need a custom field for the Instructor title, right? So I'm going to go back to custom field groups, open my instructors group, and I'm going to add a field.

[00:17:44.770]

It's just going to be a plain text field called Instructor and we'll say Title. Let's make it nice and easy. Go ahead and hit save changes. Now I can go into my instructors, I can go to Caroline Grace, I can give her a title. Now it is easier if you plan this ahead of time, right?

[00:18:02.600]

So when they say, hey, we've got instructors, so I ask the client, okay, what information do we need to have about instructors on the website? Maybe their Bio, the classes that they teach. Then we have their title, their headshot, what else are we going to need? Right? And we plan all this stuff out ahead of time so that when we're putting instructors into the database, we can fill out all of these fields at once, so we don't have to constantly go back between the builder and the back end and all of that.

[00:18:27.910]

You want to make it as efficient as possible, but remember, for those of you who are saying, wow, look at this, this work. You've got to add custom fields and you got to put it in there, and then you got to use Dynamic Day. Remember the big problem that we talked about earlier? This is not a mess, this is organisation. This is a professional process.

[00:18:46.740]

If you don't do these things, what you're going to end up with is websites that are an absolute nightmare to scale and maintain. We already talked about that it is impossible to maintain static content properly. So even though you might think it's faster and easier and more straightforward to use static content and build out static pages, it creates a nightmare scenario and we have to avoid that at all costs. Okay, so I'll say her title right here. I'm going to go ahead and hit Update.

[00:19:16.790]

I don't know what yoga instructor titles are. Well, maybe we'll say like Lead instructor. Okay, try to get something somewhat relevant in there. And then I'll bring in Daisy Liu and we're going to go to her instructor title. Now, keep in mind, if a client needs to manage the instructors of the site or the instructors of the studio on the website, which is highly likely, do you want them in the builder changing fields, moving cards around, things like that?

[00:19:43.080]

Or do you want them to just update these little custom fields right here, open an instructor, change their title, change their bio, swap their image, hit Publish. They never have to touch the builder. They never have to go into builder land and mess with any of that stuff. And the chance of breaking things is slim to none with this method, whereas chance of breaking things when they're working in the builder very high. Okay, so instructor Title, another title right here.

[00:20:08.370]

We just need some different content. Okay? All right, so what I'm going to do is I'm going to go back into the builder, we're going to refresh so that we have access to that new stuff in the database. And then for title two again, what am I going to do? Get rid of the static content.

[00:20:21.340]

Lightning bolt. And I'm going to come into title right here, ACF instructor Title. And it's going to immediately bring that in. Let's go to the front end. I see her title lead instructor.

[00:20:32.350]

Another title right here. So I've got my three instructors. All of this is really good. Now, the other thing that is happening basically on the back end is each of these things is a post or a page. They have their own URLs, so I can create an entire bio page for Caroline and a Bio page for Clara and a Bio page for Daisy Lou.

[00:20:53.450]

And in my builder right here, I can go to my H three and the URL, I can say again, dynamic data. I can say the post URL. And it's immediately going to make all of these link to the instructor's bio page. Now let's go to the front right here and you can see these are all going to go to the same place, which is just a blank page for right now, but cheque the URL up top. Okay, let me see if I can zoom in.

[00:21:17.240]

Instructors, Daisy Lou, look at the organisation in our URLs automatically by using this methodology. Look how everything is the way we need it to be for a professional, scalable, maintainable website. And now imagine if I needed to change an instructor, I needed to swap them out, remove an instructor, add an instructor, what have you, automatically my website, regardless of how many times these cards show up, where they show up, what they're related to, doesn't matter. It all takes care of itself. All I have to do is go over here, we're going to add a new instructor, we're going to say new instructor.

[00:21:53.470]

And then this is their actually, their Bio goes down here. This is their bio. And then this title right here. And then I'm just going to use another one of basically the same photo we used before. I'm going to hit publish.

[00:22:08.290]

Publish. And then we're going to refresh the front end. I haven't touched the builder. I'm not doing anything in the builder. Remember, these cards could be used a gazillion times across the entire website.

[00:22:17.720]

I'm going to hit refresh the grids, build themselves. The cards build themselves, the website builds itself. Why? Because I'm following principle number one. Always loop important content.

[00:22:31.180]

Now, this is not all, okay? We're going to go down and we're going to talk about two more very important principles for scalability, maintainability and professionalism. And then at the end, we are going to put it all together and I'm going to show you exactly what this all looks like, okay? Now let's cheque right here before we move on to principle number two, what should be managed with custom post types and custom fields and query loops and all of that? And I say the answer is guys almost and girls almost everything, okay?

[00:23:04.650]

Default posts should only be used for blog posts and pages should be used as little as possible. So custom post types, team members, services, products, reviews, unless you're using WooCommerce. WooCommerce already does that for you. But if you're putting in products manually and you're not using something like WooCommerce, absolutely. Custom post type, reviews, locations, service areas, FAQs, freebies courses and trainings videos, events, downloads, landing pages.

[00:23:34.610]

I put downloads twice because it's a bonus. Resources, legal pages, press releases, white papers, all of these things should be custom post types. Again, this is using WordPress as a content management system, not a junk drawer. So many websites use WordPress, so many web designers, I should say use WordPress as a junk drawer. You can't do that any longer.

[00:24:00.510]

Professionalism, scalability, maintainability says use WordPress as a content management system, organise everything, have a good infrastructure and then loop your important content. Build pages dynamically, not statically. Let's move on to principle number two. You need to tokenize your global styles. Tokenized styling is the future of web design and it's been happening for years and years now.

[00:24:30.630]

So if you're not on the tokenization bus, my friend, you are a bit behind and I want to catch you up. I want to get you to be tokenizing your styling. So once again, remember static verse dynamic, right? Using

static values for website styling creates hundreds of problematic disconnections. Tokens on the other hand, create a single source of truth for style values that you are going to use over and over again.

[00:25:00.020]

Tokenized styling helps reduce decision making while increasing efficiency, consistency, scalability and maintainability. So the first thing we need to talk about is what is a token? A token is effectively a placeholder for a value and in CSS, these are officially called custom properties. You'll also hear people refer to them as variables. I refer to them and many people refer to them as tokens.

[00:25:30.190]

Doesn't matter what the terminology is, the concept is the same. And so if we look at how do you create a token? Tokens cannot really be created in any page builder. They have to be created in a CSS style sheet. But this is very easy to do.

[00:25:45.420]

You can use WP code box, you can use a CSS third party CSS editor, you can use a child theme CSS style sheet. You can do it. A lot of builders have style sheets in the builder. You can do it there, you can do it any place, okay? You just need to make sure that it happens all in one place typically.

[00:26:03.790]

So it's easier to maintain and organise. But to create a token, it's very, very simple. So I'm going to do this in bricks's, child theme. So I'm going to go to appearance and I'm going to go to theme file editor right down here. And then I'm going to say, hey, I understand, right?

[00:26:20.570]

And we're in the style sheet, style CSS. I'm going to zoom in here to make it very easy for you guys to see. I'm going to type colon. I'm going to type root. And then I'm going to open the curly brackets right here.

[00:26:31.550]

And let's say I'm going to create a token for medium font size on a website, like my website's base font size, right? So I'm going to do a double dash. That's how you define a token name. And then you just name the token. I'm going to say text medium, okay?

[00:26:48.790]

And then you put a value for text. Medium. The value is going to be three REM for right now. Of course, you should already know we're avoiding pixels, right? We don't want to do something like 30 pixels.

[00:26:59.690]

We want to do three REM for maximum accessibility. Okay? Now I'm going to go ahead and hit update file. Now that I've updated that, I can refresh WP builds. And let's take a look at, for example, this content right here.

[00:27:14.250]

Actually, let's do it on this card, okay? So I can grab this card right here, and I'm going to go to Style typography and the font size. I'm going to say, hey, I want this font size to be my Medium text of my website, right? So I'm going to write a variable expression, okay? So we can see down here, how do you use a token?

[00:27:34.360]

You use a token by wrapping the token name in a variable expression VAR and then open close parentheses. Okay? So I'm doing that right here. Now, you can do this in custom CSS, or you can do it in your page builder, assuming your page builder supports variables and tokens and custom properties, which it absolutely should if it's a professional page builder. If it doesn't, that's a red flag.

[00:27:58.530]

Okay, we need to come in, write our VAR, and then put our variable text Medium. And look at this. Well, it's too big, right? It's too big, but that's our Medium text. Now, here's the thing.

[00:28:09.360]

I'm going to go ahead and hit save here. What if it is too big, right? I can come in and say, oh, I actually want that to be 1.4 REM, okay? And I'm going to hit Update File and watch the power of the token when I refresh. They're all updated.

[00:28:25.180]

Every one of them just updated, right? And I'm using a class, right? So it's also extending to every single card. We're going to talk about that in just a minute. But I now have a single source of truth.

[00:28:36.950]

I'm not using 1.4 REM everywhere, right? What a lot of people are doing is they're going, here 1.4 REM, here 1.4 REM, here 1.4 Ram, here 1.4 REM, here 1.4 Ram, here 1.4. It's just 1.4 Ram as a static value

referenced all over the place. And this is once again creating an absolute nightmare if you use that 1.4 rem value 100 different times. And then someone comes along and says, you know what, we really need our Medium text size, our base text size, to be a little bit bigger.

[00:29:08.250]

And by the way, keep in mind that typically you're not using one font size, okay? The other point here is that we have something called Text large, and maybe that's 1.8 REM. And then we have something called I love the child theme area. Actually, I don't. We're going to go to Text extra large, something like this.

[00:29:25.960]

And that might be 2.2 REM. Okay. So I'm going to update the file. So now we have different options. And now we're in a situation where obviously someone comes along later and says, hey, that large text we've been using everywhere, we need to dial it back just a little bit, or we need to make it a little bit bigger.

[00:29:41.600]

And then our extra large, we need to dial it back or make it a little see, the point is, when I use these raw values everywhere, like if I just went around littering the website with 2.2 REM here, 2.2 REM here, two point every time I want large text, I'm creating a lot of breakage. A lot of breakage and a situation that's going to require a tonne of manual labour to go update when we decide that things need to change, that is not scalable, that is not maintainable, that is not professional. A professional knows, hey, I can't be using raw values for things everywhere. I need to use tokens so that I have a single source of truth for that value. Just like the professional says, hey, I need a single source of truth for important content in the database.

[00:30:31.650]

I can't be going and littering this website with static content everywhere. Once again you're going to hear this phrase over and over again. Single source of truth. I need a single source of truth for my key content. I need a single source of truth for my global styles that I use over and over and over again.

[00:30:49.250]

And then I need a single source of truth for element styling. We're going to talk about that in just a second. Now let's talk about these tokens, how to make them even more powerful. What if you want these to be automatically responsive, fluidly responsive, so that you don't have to manage breakpoints and all of this other stuff? Well that requires a Clamp function.

[00:31:09.370]

Okay, if you go to Geary.co and you search for Clamp Calculator, I have a free Clamp Calculator that you can use. And you can come in. You could set your root font size, and you can set your device sizes, your content width of your website, and then you could set the minimum value, let's say, like, 1.2 REM. And then our maximum value is going to be 1.8 REM, something like that.

[00:31:30.890]

And then I'm just going to grab this whole clamp function. If you're not familiar with clamp functions, don't worry about it too much right now. Again, page building 101. My free course goes over all of this stuff in detail. I'm going to go ahead and put this clamp function in here.

[00:31:45.650]

Let's make that the large text. Okay, so we're going to say large text is that clamp function. Now update file. Here's the thing. If I use this text large token all over the website, that's phenomenal because now that I've made this change, this Clamp function is now applying to all of those places like that.

[00:32:05.080]

I didn't even have to go in the builder right I'm just changing one thing into style sheet. And perhaps I've updated 1000 different elements across the website just from that one single source of truth. I've now taken all of that text and I've made it automatically responsive. This is fantastic. So text large, let's go use that somewhere.

[00:32:23.410]

Let's go use that right here. So I'm going to say that this is going to be text large instead of text medium. And I'm going to go ahead and hit save. And then we're going to go to the front end and we're going to inspect this. Large text right here.

[00:32:36.870]

Says font size right here. VAR text large. Now, more importantly, let's go to computed. Remember I told you it's going to be automatically responsive. Now it computes to 25 pixels.

[00:32:48.440]

Okay, what we're going to do, I'm going to click this and I'm going to start bringing this in. Now, this page has not been made to be super responsive yet, but look at this font size number. As the screen gets smaller, the font size is getting smaller, right? And so that font is automatically responsive. Now think about what if I did this to every font size that I use across the website?

[00:33:11.390]

What if every one of them was represented with a token? And what if every one of them had a clamp function that made them automatically responsive? My typography across this website would be insanely easy to maintain, insanely easy to use. Let's go back here for a second and let's talk about what styles should be tokenized, okay? Any style that is used repeatedly, it's not just font sizes, my friends.

[00:33:39.030]

Colours, shades and transparencies text sizes. Heading sizes. I have text sizes here, heading sizes over here. I keep those separate on my website. Built spacing values section, spacing values, button styling, website header height, that comes in tremendously handy.

[00:33:55.460]

Border radius, website content width, your website's gutter, the side padding on your gutter, grid spacing content spacing, container spacing, common grid structures, box shadows, gradients, all of this stuff should be tokenized. And again, I go into much more detail in Page building 101 and what I will just say about this as well, it is a lot of work to set up ahead of time. Okay? My pro tip, and this is why I created Automatic CSS, why I use Automatic CSS on every website, why thousands and thousands of developers using page builders are using Automatic CSS. Automatic CSS gives you the tokens for all of these things out of the box.

[00:34:37.610]

And then we give you a UI dashboard. So you go back here, you're like, well, how do I change the values of those tokens? Well, that's what's amazing. You can change the values with simple input fields, right? So go to your typography.

[00:34:49.410]

You can change your base heading size, you can change your base text size. All of your other sizes are calculated based on a mathematical ratio. But if you want to override them. Very easy text size overrides. You just say, hey, here's what I want.

[00:35:03.400]

Let's say I want 60 pixels for my XXL size. 60 pixels on desktop, 24 pixels on mobile. It's going to be fluidly responsive automatically. I hit save changes. All I have to do is go use my variables, use my tokens whenever I'm building things.

[00:35:19.720]

And by the way, when you're using tokens in your builds, the decision making all goes away. It's so easy and efficient to build because you're just dropping tokens everywhere. You're not having to think, oh, what's that value again? What was that? 2.2 rim?

[00:35:34.880]

Is that what we were using for this? And then what were we using at the breakpoints? No, there's no more breakpoints. Everything's automatically fluid responsive and there's no more thinking about what was the value there. It's just dropping tokens.

[00:35:45.620]

Oh, I just need my Medium text there. Oh, I need my large text there. I need my Medium spacing there. I need my large spacing there. And you're going to see this come to life in just a second.

[00:35:53.870]

I'm going to walk you through the entire workflow so that you can see exactly how easy it is, okay? But we need to move on. So we know tokenizing, global styles, very important. What else do we need to do? What's principle number three, the final principle.

[00:36:08.880]

We need to use a class based workflow. Why? Because styling elements individually is time consuming and it makes websites impossible to maintain and scale. Assigning styles to classes. On the other hand, right, just a little short teaser here just to cover this really fast.

[00:36:32.930]

When you use a page builder, you just open a page builder up and you start styling something, okay? Like, let's just add a button here, okay? So a button has been added right here. And if I just go start styling things like font size, 2.2 rim and I come in here and I say my padding is going to be 18 pixels and 18 pixels and 24 pixels on the sides. This is what so many people do.

[00:36:56.490]

They just come in and start littering static values everywhere. They're not using tokens. And even worse, they're styling at the ID level. This is called an ID. And the ID styling only applies to this individual element, okay?

[00:37:14.220]

It doesn't apply to any other buttons. Well, a button is something that we use a gazillion times across a website, right? These cards, look at these cards right here. I'm using these cards multiple times in this grid. And then I'm using this grid of cards multiple times across the website on different pages of the website.

[00:37:31.950]

And so I need to have a single source of truth. There's that phrase again, a single source of truth for how these cards are styled. And the way that I do that is with classes. Look on here. The reason these cards all have the exact same styling is they all have classes on them and their elements all have classes as well.

[00:37:53.700]

So for example, if I wanted to round the corners of these cards, watch this. I'm going to go into Border and notice I'm editing the class. Okay? I'm going to go into Border and I'm going to have a there's actually a radius on here already. I think what the problem is is there's no overflow set.

[00:38:09.870]

So I'm going to go and that's good. This goes to show you, hey, we set a radius on all these cards but we're not seeing a radius. So that's a problem that we have to fix. So we're going to go into Layout overflow and I'm going to set the overflow to Hidden. And now we see the rounded corners.

[00:38:23.290]

And we don't just see the rounded corners on the card I just fixed. We see the rounded corners on every card for two reasons. One, these cards are generated by a loop. So there's actually only one instance, but also because I'm doing this on a class, meaning that style is going to apply to every instance of that class. And so with one change, one fix, I'm going to refresh right here.

[00:38:44.920]

You see that? That applied to every single card. That's the value of looping content and that's the value of assigning classes to elements. Now I'm going to show you down here there is a situation where these cards are not looped. There's actually three cards.

[00:39:02.480]

These grids, this grid looks exactly the same. These are actually two grids. This is grid one. This is grid number two right here in grid one. Everything has a class down here, no classes.

[00:39:14.370]

Okay, so we're going to see something. What I need to do, see this red colour we're using and a grey background. And I need to make these cards look like this green theme over here, okay? This green light green and then dark green for the text, I need to make these cards look exactly like that. So what I'm going to do is I'm going to activate the class.

[00:39:33.420]

Remember, this is not a loop. So all three of these cards are manually. They're statically placed here. But I am using classes. So let's take a look at how this goes.

[00:39:42.860]

So I'm going to go background colour. What I want to do here is I want to use, and I'm going to use a variable. I'm using tokens. Okay, I'm going to use my website's Success Colour. We're going to say success light. Look how every single card just updated. I made that one change to one card but all of them updated. Okay, next thing I'm going to do is I'm going to click on my icon right here and I'm going to come over here and I'm going to say Token success, Dark, success. Look, they all updated.

[00:40:14.150]

I'm going to come over here and I'm going to grab that class right here. Once again, I'm going to come in. This needs to be success dark. Every single one of those checkmarks just updated. Now I'm going to go back to my card, I'm going to go to border and I'm going to say, okay, instead of a one pixel border on all sides, we need a five pixel border on all sides.

[00:40:34.820]

And instead of it being shade light, I want it to be success dark. Okay, so success dark. Now, that doesn't actually look so good. Let's just use the success colour by itself. That's fine, whatever.

[00:40:46.260]

I'm not trying to make the most beautiful card on planet Earth. What I want to show you, I'm going to refresh. What I want to show you is that by changing the styles on one card, I changed all of them. And it doesn't matter if there's three of them or if there's 30,000 of them, all of them just updated. I'm not having to copy and paste styles.

[00:41:05.390]

I'm not having to go page to page to page to page to page. This is what scalability and maintainability look like, and this is what professionalism looks like in modern web design. Now, I'm going to come down here

to where we have no classes, and I'm going to make the exact same changes. And I want you to see what a nightmare scenario this is. Because classes were not used, we're not going to use tokens either, my friends.

[00:41:32.320]

We got to know what the hex colours are. Okay? So check this out. I'm going to first change the background colour. So I'm going to come in, I'm going to get rid of this shade ultralight.

[00:41:42.070]

And I'm not using tokens, right? I'm just doing it the old manual static way. What's the hex code of my Success Light colour? I don't even know. So what I have to do is I have to come up here and I got to pick a colour for how many of you have lived this life?

[00:41:54.770]

I got to choose the hex code right here. I got to bring this in and I got to put it in. Okay, finally it's in. Now I want to make this success dark. Okay?

[00:42:02.870]

So I've got to go back to my colour picker right here. I got to come in here and I got to get that colour right there. Then I got to get out of my colour picker. Let's go back in here. Where were we?

[00:42:12.830]

We're here. We're here, okay. And I'm going to say that this is going to be that hex code right there. Okay, that's looking fantastic. Now, I'm going to use the same one on these icons.

[00:42:22.710]

So I'm going to paste that in there manually. Come in here, paste that in there, come in here. Get this. Paste this in here. Okay, perfect.

[00:42:33.150]

Now I need the border. Okay, let's do the border. All right, so we're going to go to style, we're going to go to border. This is so painful, guys. Five on each side.

[00:42:43.080]

Okay. And then I'm going to go into colour. And then I've got oh, I need this action colour. Let's go back to the colour picker. Because I'm not using tokens, I can't memorise these hex codes, right?

[00:42:54.280]

So I'm going to grab that. I'm going to come in colour. Let's go ahead and paste that in. Okay. And then oh, border radius.

[00:43:01.280]

The border radius has got to match. Okay, what are we using for the border radius here? I don't even know. I guess I got to make up a value or something. So I'm going to go border radius.

[00:43:10.110]

I mean, what's going to look good? Let's do like ten pixels. I'm using pixels because I don't know what I'm doing. I'm eyeballing it. That looks decent.

[00:43:18.970]

Okay, let's put that on all sides. Amazing. Now save refresh. I've made the changes. These cards look exactly the same.

[00:43:27.250]

But guys, look at these two cards. They did not change, they did not update. What am I going to do now? I've got to either copy paste these styles and then, by the way, copy paste styles does not work across pages of a website. So I need to know everywhere this card has been used.

[00:43:43.260]

How am I going to know that? How am I going to know that? Just what, some sort of miracle that my brain is for all of my clients? I know where all of their cards have been used on a website and I'm just going to go manually change them one by one by one by one. No, this is impossible.

[00:43:57.450]

This is a nightmare scenario you saw with classes and tokens. All I needed to do was know the names of the tokens, which is very easy because they have logical names. And then I assign the styles to classes and magically, like I said, doesn't matter if there's three cards or 30 million cards, they're all getting the update from one single source of truth, professionalism, scalability and maintainability. Okay? In action.

[00:44:24.970]

You see the problem right here? Glaring issue. No issue whatsoever up here. Now, how do we keep our classes organised? You can use something called BEM B-E-M, block element modifier.

[00:44:38.570]

If you want to go deep into this. On my YouTube channel, I have a whole BEM deep dive. Okay, but to keep it nice and simple, you name your card something like, hey, we're going to make a card for our services. Let's call it a service card. Or for our classes, let's name it a class card, for our instructors, let's name it an instructor card.

[00:44:54.560]

And then you put a class on every element inside of that block. Okay? So if there's an icon in the block, you name it. Service card double underscore icon if there's a heading in the block. Service card double underscore heading if there's text in the block.

[00:45:11.000]

Service card double underscore text if that text needs to be more descriptive, like bio name it. Service card double underscore bio. And what this does is it makes all of these classes live together and visually just by reading. I know on this website there's something called a service card and it's got an icon in it, it's got a heading in it and it's got text in it. And these are the classes for those, right?

[00:45:34.380]

And then we just apply the styles to the classes and we use tokens to apply those styles. And what we have at the end of the day is a website that is insanely professional, insanely scalable, very easy to maintain, okay? Super efficient, super logical, very organised. Multiple team members can work on it. Very, very easy to do.

[00:45:55.490]

So let's take a look at how this workflow actually comes together. So, putting it all together, we are tasked with building a website for a yoga studio. They have six different classes and a few different instructors. We want to easily show the classes that are offered right here. We want to show the instructors and we want to show which instructors lead which classes.

[00:46:19.360]

Let's make sure that we do this in a way that's well organised, scalable and maintainable. Okay, so here we go. We're going to dive in and we're going to try to do this quickly as possible because I'm running out

of time. In fact, to save time, what we're going to do is we're going to come up here and I'm going to duplicate this grid of instructor cards right here and we are going to steal that grid and we're going to bring it all the way down here and we're going to drop it into our instructors area. So our cards are already done.

[00:46:49.810]

You guys kind of already saw me handle those, right? And I want to get rid of one of our instructors. So we're going to go back into instructors here and we are going to get rid of new instructor right there. Okay? I'm going to go ahead and hit save and we're going to refresh.

[00:47:02.850]

So now we've got our instructor cards in here and I actually want to fix our text size for these. We want to use our real. We're going to use like text S as our font size there. Let's go ahead and refresh here. That looks much better.

[00:47:17.160]

Okay. And in fact, instead of a title, I think we're going to use that area to show what class she teaches. Okay? But we got to get the classes in there first. So we don't have any classes yet.

[00:47:28.290]

This is once again an area where we need a custom post type. So we're going to go back into ACF. We're going to hit Add New and we're going to say Classes, singular class, post type, key classes. Go ahead and hit save. There it is.

[00:47:42.690]

It's as easy as that. I'm going to go into classes, I'm going to add a new one. I'm going to set a featured image. Here's the photo for it. It's called vinyasa yoga.

[00:47:52.320]

So I'm going to say vinyasa yoga. I'm going to go ahead and hit Publish. Now, before we add any of the other classes, we need to start thinking about custom fields. What do we need? We need two things.

[00:48:02.830]

We need a class description and then we need a relationship. So we need to relate classes to instructors, okay? Very easy to do. So I'm going to go into ACF and we're going to go into field groups. Add new field group.

[00:48:17.250]

This is going to be for classes. I'm going to come down and assign this to classes, okay? We're going to go ahead and hit save. And then the first field that I want is a WYSIWYG field, and that's going to be called class description. Class description.

[00:48:34.830]

No default value needed. Add a field. This is going to be called Relationship. Okay? I can actually just search for it, it'll find it.

[00:48:42.410]

There it is relationship. And I can say because I'm doing classes, I want related instructors, okay? Related instructors. I'm going to go ahead and hit Save for right now, what I need to do, because I just added classes and I'm creating these relationships, I need to go back to my other field group called instructors. And I need to create a relationship field here called what?

[00:49:04.670]

Related classes, right? This allows me to edit a class and say which instructors go with it or edit an instructor and say which classes go with that instructor, right? So this is going to be related classes. Related classes. And I'm going to go ahead and hit save.

[00:49:22.600]

Now I'm going to open let me just get rid of all these tabs, okay? I'm going to open both of these at the same time just because this makes it a little bit easier to understand. So I'm going to edit instructors and I'm going to edit classes. So here's my instructors with related classes, and here's my classes with related instructors. Okay?

[00:49:40.650]

So I'm going to go down here and I'm going to say what post type are we looking for when we want to relate a class? Well, we're looking for the class post type. Okay? Then I'm going to come in and when I'm on classes doing related instructors, what post type do I want to choose from? I want to choose from instructors, and I'm going to go ahead and hit save there.

[00:49:59.200]

All right? Now one more thing. What I need to do on this Relationship field is go to Advanced, okay? Bi-directional, right? It says enabling the bi-directional setting allows you to update a value in the target

fields for each value selected in this field, adding or removing the post ID, yada yada, blah, blah, blah, blah, blah.

[00:50:17.530]

Okay? So target fields, I'm going to say under instructors we want classes are going to be related instructors and good. So I want to choose that. And I'm going to hit save. And then over here and the one thing that will help you here is when you're trying to figure out which one does it actually want me to use?

[00:50:37.210]

Don't use the one that says this field because you're referencing itself. So I'm going to go ahead and hit save. Changes here. And now we just want to test this out. So if I go to Vinyasa Yoga, can I choose related instructors?

[00:50:49.540]

And yes I can. Look at here's our instructor just dynamically saying, hey, here's your instructors. Why? Because we organised our database. We don't have a junk drawer.

[00:50:58.630]

We have organised drawers called custom post types. And here is the data. Okay, now what happens if I go edit an instructor so I can go to Caroline and can I assign her to classes? Yes, I can. I can assign her to separate classes right here.

[00:51:16.040]

And this is what we want. So let's go ahead and finish adding our classes. Then let's build some class cards and then let's show you how the relationships really work. Okay, so I'm going to add a new class here, featured Image. And we are going to use this one for, this is called let's do this one, meditation Yoga.

[00:51:34.250]

So set featured image, meditation Yoga. And let's say Caroline is going to teach this one. So I'm going to sign Caroline to here and then I'm just going to use website Ipsum for our card description or a class description. Go ahead and hit publish here. Let's go back for Vinyasa yoga.

[00:51:53.320]

I want a class description right there and I want Clara to teach that. I'm going to go back, okay? And let's just do three classes. Let's not go crazy. What else can we do?

[00:52:02.550]

We can do basic yoga. That sounds good. Let's do a basic yoga. Let's give it a class description and say Daisy teaches Basic Yoga. Let's go ahead and publish here.

[00:52:14.270]

So now I've got my classes in here, I've got my instructors in here, I've got these things related to each other. What I need to do now is I need to go edit my page. Okay? And again, we would be doing this across multiple pages. So for example, we'd have a page with classes, we'd have a page with instructors, but we're still doing dynamic data, we're still doing loops and we're still building with classes and variables.

[00:52:36.380]

And that's what we're going to do from scratch right here is build cards for our classes. First thing we're going to need to do is create a class for our grid. So this is going to be a classes grid just like that. And I'm going to go display grid. And then remember I said tokens for everything.

[00:52:53.700]

Look how easy this makes it. I don't even need to know how grid works. I just need to drop tokens places. I want a grid gap for my grid gap. It makes sense, right?

[00:53:03.880]

And I want. A three column grid. So I'm going to say grid three. I'm using a token called grid three that's going to immediately give me a three column grid. So I'm going to add a block from my card and then I'm going to come in.

[00:53:15.010]

I'm just showing you, I'm not going to explain every step in every detail, okay? This is going to be our card. This is going to be our grid. Here's our card. And I'm going to call this a class card, all right?

[00:53:24.780]

Because that's what it is, it's showing our classes. So it's a class card. So I'm going to add two wrappers right here. This is going to be my class card. Media wrapper.

[00:53:34.060]

This is going to be my class card. Body wrapper, class card double underscore, body wrapper, just like that. Then in my media wrapper, I'm going to add my media. That's going to be my image, right? So what I'm going to say, class card double underscore, media, just like that.

[00:53:49.530]

And then I'm going to say, hey, dynamic data here. Actually, let's not do the dynamic data yet because I haven't done the loop yet. Let's just go ahead and build the card. Let's build it with a static image just like this, just so we can see what we're actually doing. Okay, there we go.

[00:54:03.540]

And then in my body wrapper, I'm going to need a heading here. I'm going to need text here and I am going to need that's pretty much it. Let's keep it nice and simple. So what is this heading going to be? Class card double underscore.

[00:54:16.010]

Heading. Then what's this text going to be? Class card double underscore. We'll just call it a description because that's exactly what's going to go there. And then what I want to do is style this up a little bit.

[00:54:25.060]

So my body wrapper, I'm going to need to add some padding to that. How much padding? Don't know. Let's just try space M. That's medium spacing.

[00:54:33.630]

Does that look pretty good? It looks pretty good for now. I might change it in a minute. Card description, what we're going to say, and I'm doing all this on our classes, right? I want this to be small text.

[00:54:43.500]

Next thing I'm going to do is I am going to add a gap here. What's our gap going to be? The great thing about tokens is that you can actually do math with them. So I have a token called content gap, but it's going to be a little bit much because this is in a card format. So I want to do content gap divided by two.

[00:55:00.320]

I want half of my content gap and I can easily get that. I'm not worried about Rims and Ms and this and that. I'm just dropping tokens places, that's all I'm doing. Very, very efficient. Then in my body wrapper, I want this to have a background colour.

[00:55:15.080]

What's my background colour going to be? I want it to be success. Light. Okay? Remember we were doing our Success colour, our success theme on here.

[00:55:23.140]

And then for this heading, I want this to be Success Dark. Okay? So I'm going to say success. Dark. I'm going to grab that and I'm going to make this on my text as well.

[00:55:33.210]

Typography colour can be success. Dark. Okay? And then what I want to do is on my entire card, I want to add some border radius. So I'm going to go border radius and I'm just going to drop a token called my medium radius.

[00:55:48.920]

And then I'm going to go set the overflow to hidden just like this. Now card is coming along nicely. I'm going to grab my image and I'm going to set an aspect ratio on this for this because Bricks doesn't really do aspect ratio very well with the builder. I'm just going to say 16 x nine and I'm going to get a nice 16 x nine format for that card. Then I'm just going to come to object fit and I'm going to cover it and that's going to look pretty sent.

[00:56:15.370]

So I'm going to come into our card and I'm going to say, hey, this is going to be a query loop. What are we going to loop through? Well, we're going to loop through posts. What kind of post? Oh, we're going to loop through our classes.

[00:56:24.270]

Okay. Then I need to go through my process of get rid of the static data, drop in the dynamic data. So here's my featured image. Let's make sure we choose the medium size. I can come into the heading Rid of the static data, come in and say that's going to be our post title.

[00:56:39.170]

Get rid of this. This is going to be the class description right here. ACF class description. And suddenly let's go here. Let's look suddenly guys, it's just built itself out for me, right?

[00:56:52.400]

The whole grid is like, it's done. And remember I said, hey, I might need to change the padding on this, right? Maybe we want large padding instead of medium padding. I'm just going to change the M to L and that's going to give us a little bit more breathing room in our card, right? Which looks good.

[00:57:09.500]

Breathing room for a yoga website. Makes perfect sense. What about this line height on this text right here? What if I want to come in here and I want to say Typography line height 1.6, maybe 1.7, something like that. I'm going to hit save.

[00:57:23.390]

All of the cards are going to update to have 1.7 line height just like that. So this is very scalable, very maintainable. Let's do the relationship now. Okay, we need another field. And here's where the loop really comes in handy, right?

[00:57:38.340]

I'm going to add a piece of information to this loop and all the cards are going to get this new information. So I'm going to add a text field, basic text right here. And this is going to be called Class card Double Underscore class. Okay. Just like that.

[00:57:54.620]

Class card, double underscore. Actually, sorry, we're doing the class card. This needs to be instructor. Okay? So instructor, good.

[00:58:02.090]

And we're going to add this for our instructor relationship, right? How do we do this? Get rid of the static info, come in here. And what we need is a relationship, related instructors. Check this out.

[00:58:15.940]

Related instructors, hit save refresh. It's just sticking the related instructors here. It's even linking to them so that people can click and say, hey, there's the instructor, right? Okay, so we're going to go and we are going to style this up. So I'm going to say font size here is text s and our line height is going to be one just like that.

[00:58:39.160]

And I'm actually going to make this into a little bit of a badge. So for my padding, I'm going to say I can even use Ms here if I want to. You don't always have to use tokens if it's a situation where just very easy

to use Ms, go ahead and do that. I'm going to say 0.5 m on the top and the bottom one, m on the left and the right. And then I'm just going to add a background colour.

[00:59:01.140]

What's our background colour going to be? Success Dark. There we go. And then for Typography, I want the colour to be success Light. Okay?

[00:59:09.810]

Success Light. Just like that. And then I want to round the corners of that. So we're going to go into border radius and I am going to give this a radius of M to match the card. I don't even know what the radius is, but it's going to match the radius on the card refresh.

[00:59:26.800]

And now we've got the instructor on each class card and then on the instructor cards, we can basically do the opposite. So I can come in here. Instead of this being a title, we're going to say that this is the class that they teach, okay, class, let me edit this class. Okay? And then I am going to say, no, this is not their title anymore.

[00:59:49.080]

This is the related classes. Okay? So I go ahead and hit save here. Let's refresh. Awesome.

[00:59:55.940]

So now you can see Caroline teaches meditation yoga, clara teaches Vinyasa yoga, Daisy teaches basic yoga. This is the power of relationships. And again, let me go back to camera. Remember what I just did, what I just showed you? It doesn't matter if there's two instances of that element or style or whatever we happen to be working on or 2 million.

[01:00:21.750]

I have a single source of truth. That was the key phrase that I used over and over and over again. Single source of truth. Let's review. Number one, you need to loop your critical information.

[01:00:37.760]

We talked about custom post types, we talked about custom fields, bi directional relationships, not having a junk drawer of content in your WordPress installation, but having an organised section by section content type by content type methodology for organising your content that creates a single source of truth for each piece of content and each group of content. That's principle number one. Principle number two

was to tokenize your styling. So all of your key styles that you use all over the place are derived and controlled by tokens a single source of truth for each key style. And if you don't want to do all of the front loaded work of creating all of these tokens, you have a framework like Automatic CSS you just install it gives you the whole library of all these tokens that you're going to be using all across your website.

[01:01:35.460]

And then you have a UI dashboard where you can just change the values of all the tokens. Everything is automatically responsive, all of your grids are there. Everything that you could possibly need to build a professional, scalable, maintainable website is already there for you. So you're tokenizing all of your styling, creating that single source of truth for all of your key values. And then last but not least, you have to use a class first workflow.

[01:02:00.010]

Don't have static styles everywhere. Assign those styles to classes. Those classes are assigned to all the elements and therefore you have a single source of truth for blocks elements and then you can modify those as well. We didn't talk about the modifier portion. This once again was skimming the surface of all of these topics.

[01:02:20.710]

I could go hours and hours and hours and hours and hours. In fact, I have gone hours and hours and hours and hours. What I wanted to do here in this presentation is open your eyes to a better methodology, open your eyes to a better workflow. And now I want to invite you to go deeper on that. Like I said, I have a free course, 100% free called Page Building 101 on my YouTube channel.

[01:02:46.320]

That's where you can go much deeper on these concepts as well as other concepts. I hope you got a lot out of this. If you have any questions, I am happy to jump in and help you. I respond to every single comment. But more importantly, look deeper into this.

[01:03:02.520]

Start playing around with it, start experimenting, start seeing these things come to life on the websites that you are building and you will be sold 100% on this new workflow and then your life will be better going forwards, your websites will be better going forwards and everybody will be happy again. I am Kevin Geary. This has been Three key principles for building better websites, scalable websites and maintainable websites. Hope to see you again very soon.