**Code Snippets**

**Suppress warnings on Local sites:**

ini\_set('display\_errors','Off');

ini\_set('error\_reporting', E\_ALL );

define('WP\_DEBUG', false);

define('WP\_DEBUG\_DISPLAY', false);

-----------------------------------------------------------------------------------------------------

**Include css in a php script:**

// If we get this far, we have custom styles. Let's do this.  
?>  
<style type="text/css">  
<?php  
// Has the text been hidden?  
if ( ! display\_header\_text() ) :  
 ?>  
 .site-title,  
 .site-description {  
 position: absolute;  
 clip: rect(1px, 1px, 1px, 1px);  
 }  
 <?php  
 // If the user has set a custom color for the text use that.  
else :  
 ?>  
 .site-title a,  
 .site-description {  
 color: #<?php echo esc\_attr( $header\_text\_color ); ?>;  
 }  
<?php endif; ?>  
</style>  
<?php

**Download all the images from a page:**

Drop this code into the console.

$$('img').forEach(async (img) => {

try {

const src = img.src;

// Fetch the image as a blob.

const fetchResponse = await fetch(src);

const blob = await fetchResponse.blob();

const mimeType = blob.type;

// Figure out a name for it from the src and the mime-type.

const start = src.lastIndexOf('/') + 1;

const end = src.indexOf('.', start);

let name = src.substring(start, end === -1 ? undefined : end);

name = name.replace(/[^a-zA-Z0-9]+/g, '-');

name += '.' + mimeType.substring(mimeType.lastIndexOf('/') + 1);

// Download the blob using a <a> element.

const a = document.createElement('a');

a.setAttribute('href', URL.createObjectURL(blob));

a.setAttribute('download', name);

a.click();

} catch (e) {}

});

Telephone links with extensions:

<**a** href="tel:123-456-7890p123">CLICK TO CALL</**a**>

**Change the title of the page**

Use the “document\_title\_parts” filter.

function mcnally\_custom\_title( $title\_parts ){  
 if (is\_tax('specialties')):  
 $title\_parts['title'] = mcnally\_tax\_custom\_title();  
 endif;  
 return $title\_parts;  
}  
add\_filter( 'document\_title\_parts', 'mcnally\_custom\_title' );

the $title\_parts parameter is an array of the document title parts:

The document title parts.

* title string

Title of the viewed page.

* page string

Optional. Page number if paginated.

* tagline string

Optional. Site description when on home page.

* site string

Optional. Site title when not on home page.

**Set the image alt tag if it’s empty:**

Set this function in functions.php:

//If no alt tag is set for images, use the title:  
//Credit: https://julienmelissas.com/setting-the-alt-attribute-from-an-acf-image-object/  
function acf\_set\_alt($img) {  
 $alt = $img['alt'];  
 if(!empty($alt)) {  
 return $alt;  
 } else {  
 $title = $img['title'];  
 $title = preg\_replace('%\s\*[-\_\s]+\s\*%', ' ', $title);  
 $title = ucwords(strtolower($title));  
 $img['alt'] = $title;  
 return $title;  
 }  
}

Set the alt variable:

$img = get\_field(‘’);  
if($img) :  
 $src = $img['url'];  
 $alt = acf\_set\_alt($img);  
endif;  
?>

Use in the image tag like this:

<img src="<?php echo $src; ?>" alt="<?php echo $alt; ?>">

Background Image:

style="background-image: url('<?php echo str\_replace( "1024px@2x", "1200px", $slide[ 'image' ] ); ?>?quality=70')"

Image cluster

$img = get\_sub\_field('image');  
if($img) :  
 $src = $img['url'];  
 $alt = acf\_set\_alt($img);  
endif;

Link cluster:

$link = get\_sub\_field('link');  
if($link) :  
 $url = $link['url'];  
 $title = $link['title'];  
 $target = $link['target'] ? : '\_self';  
endif;

Background video:

HTML:

<div class="billboard">  
 <section class="video">  
 <video preload muted autoplay loop playsinline>  
 <source src="https://embed-ssl.wistia.com/deliveries/3074956cf27cad8019658e692657e40a53699125/my-file.mp4" type="video/mp4">  
 </video>  
 </section>  
</div>

CSS:

.billboard {  
 position: relative;  
 width: 100vw;  
 height: 100vh;  
}  
video {  
 display: block;  
 position: absolute;  
 top: 50%;  
 left: 50%;  
 min-width: 100%;  
 min-height: 100%;  
 width: auto;  
 height: auto;  
 transform:translate(-50%, -50%);  
}

Continuous animation of words:

$( ***document*** ).ready( function() {  
 let wordCount;  
 let wordInterval;  
 let currentWord = 0;  
 let previousWord = 0;  
  
 // eslint-disable-next-line prefer-const  
 wordCount = $( '.word-slide' ).length;  
 //use .eq to get the 1st word and only give it a top of 5px.  
 $( '.word-slide:eq(' + currentWord + ')' ).animate( { top: 5 }, 'fast' );  
  
 // eslint-disable-next-line no-unused-vars,prefer-const  
 wordInterval = setInterval( wordRotate, 2000 );  
 // eslint-disable-next-line no-mixed-spaces-and-tabs  
 function wordRotate() {  
 currentWord = ( previousWord + 1 ) % wordCount;  
 // eslint-disable-next-line no-mixed-spaces-and-tabs  
 $( '.word-slide:eq(' + previousWord + ')' ).animate( { top: -205 },  
 'slow',  
 function() {  
 $( this ).css( 'top', '210px' );  
 } );  
 $( '.word-slide:eq(' + currentWord + ')' ).animate( { top: 5 }, 'slow' );  
 previousWord = currentWord;  
 }  
} );

CSS Calc for font-sizes:

(Source: https://www.madebymike.com.au/writing/fluid-type-calc-examples/)

body {

font-size: calc([minimum size] + ([maximum size] - [minimum size]) \* ((100vw - [minimum viewport width]) / ([maximum viewport width] - [minimum viewport width])));

}

Fluid Type Sass mixin: <https://codepen.io/MadeByMike/pen/vNrvdZ>

Fluid Typography Tool: <https://modern-fluid-typography.vercel.app/>

Simple SCSS set up: https://sprucecss.com/blog/the-simplest-sass-compile-setup/