# /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*jQuery Examples\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*alert message for testing\*\*\*\*\*/

jQuery(document).ready(function(){

alert('Hello!');

});

/\*\*\*\*\*\*\*\*\*\*\*/

//////////////////////////Sample Script Format////////////////////

jQuery(document).ready(function () {

//alert('Hello!');

});

(function ($) {

$(document).ready(function () {

//alert('Hello!');

});

})(this.jQuery);

(function() {

$(document).ready(function() {

// your code goes here..

});

})(jQuery);

(function ($) {

//var $allVideos = $(this).find(selectors.join(','));

//$allVideos = $allVideos.not("object object"); // SwfObj conflict patch

})(window.jQuery || window.Zepto);

jQuery(function ($) {

// Quantity buttons

$('div.quantity:not(.buttons\_added), td.quantity:not(.buttons\_added)').addClass('buttons\_added').append('<input type="button" value="+" class="plus" />').prepend('<input type="button" value="-" class="minus" />');

$('.plus').val('\uf067');

$('.minus').val('\uf068');

$(document).on('click', '.plus, .minus', function () {

// Get values

var $qty = $(this).closest('.quantity').find('.qty'),

currentVal = parseFloat($qty.val()),

max = parseFloat($qty.attr('max')),

min = parseFloat($qty.attr('min')),

step = $qty.attr('step');

// Format values

if (!currentVal || currentVal === '' || currentVal === 'NaN') currentVal = 0;

if (max === '' || max === 'NaN') max = '';

if (min === '' || min === 'NaN') min = 0;

if (step === 'any' || step === '' || step === undefined || parseFloat(step) === 'NaN') step = 1;

// Change the value

if ($(this).is('.plus')) {

if (max && (max == currentVal || currentVal > max)) {

$qty.val(max);

} else {

$qty.val(currentVal + parseFloat(step));

}

} else {

if (min && (min == currentVal || currentVal < min)) {

$qty.val(min);

} else if (currentVal > 0) {

$qty.val(currentVal - parseFloat(step));

}

}

// Trigger change event

$qty.trigger('change');

});

});

//////////////////////////Sample Script Format////////////////////

var courseids=[];

jQuery(document).ready(function() {

//jQuery('#content').removeAttr('style');

if (jQuery('.pmpro-confirmation').length != 0) {

setTimeout(function () {

window.location.href = 'all-courses';

}, 2000);

}

var widthscreen = screen.width;

if (widthscreen > 1050)

{

//var header = jQuery('.fusion-header').height();

//jQuery('.loggedin').css('top',header);

}

//window.alert(header);

jQuery('body.nonpaid-member form.purchase-course').remove('form');

jQuery('body.expired-member form.purchase-course').remove('form');

jQuery('#\_lp\_retake\_count').attr('disabled', 'true');

jQuery('#\_lp\_duration\_select').attr('disabled', 'true');

jQuery('#\_lp\_duration').css('pointer-events', 'none');

jQuery('#\_lp\_duration').css('opacity', '0.5');

jQuery('#\_lp\_duration').val('0');

//jQuery('#acf-material\_url\_osoftz').css('display', 'none');

//jQuery('#acf-lesson\_url\_osoftz').css('display', 'none');

jQuery("#user\_login").attr("placeholder", "USERNAME");

jQuery("#user\_pass").attr("placeholder", "PASSWORD");

});

jQuery(document).on("change", '#bemail', function () {

var min = 0000;

var max = 9999;

var randno=Math.floor(Math.random() \* (+max - +min)) + min;

jQuery('#user\_activation\_key').val(Date.now() + randno);

jQuery('#user\_status').val(2);

});

/\*\*\*\*\*\*\*\*\*\*\*\*\*/

jQuery(document).ready(function () {

jQuery('.fusion-events-shortcode .fusion-events-content p').append('...')

});

/\*\*\*\*\*\*\*\*\*\*\*change input tag attribute\*\*\*\*\*\*\*\*/

jQuery(document).ready(function () {

jQuery(".advance\_search\_for\_woocommerce\_save\_btn").attr("value", "Search");

});

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

jQuery(document).ready(function () {

jQuery("p:first").addClass("intro1");

});

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

# /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Script Examples\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

//////////////// Example to include HTML code after some unique class

jQuery(document).ready(function () {

jQuery('.tribe-events-style-full .fusion-footer').before('<div></div>');

});

//////////////// Example to Set Attribute value.

//http s://learn.jquery.com/using-jquery-core/document-ready/

jQuery( document ).ready(function() {

jQuery(".advance\_search\_for\_woocommerce\_save\_btn" ).attr("value","Search");

});

//(or)

document.getElementsByClassName("advance\_search\_for\_woocommerce\_save\_btn").value = "Search";

// Some other examples

function readyFn(jQuery) {

jQuery(".advance\_search\_for\_woocommerce\_save\_btn").attr("value", "Search");

}

$(document).ready(readyFn);

// or:

$(window).on("load", readyFn);

/////////////////////////I.jQuery attribute methods starts

//type1: attr( properties ): We have already seen it’s usage, however we can use this method to set multiple key-value properties too.

//Syntax: selector.attr({ FirstProperty : First value , second Property: Second value})

$("img").attr({

src: "/images/jquery.gif",

title: "jQuery Image",

alt: "Sample jQuery Image"

});

//type2: removeAttr(name): This method is used to remove attribute from the matched elements.

//Syntax: selector.removeAttr( name ) where name is the name of the element’s attribute.

$("table").removeAttr("border"); //Here the attribute border of the table is removed using this method.

//type3: hasClass(class ): This method is used to find a specified class is present on the any of the matched elements.It will return a Boolean value.

//Syntax: selector.hasClass(class )

$("p#pID").hasClass("myClass"); //– Here myClass is the CSS class name.

//type4:removeClass(class ): This method is used to remove all the specified classes.

//Syntax: selector.removeClass(class )

$("p#pID").removeClass("myClass"); // – Here myClass is the CSS class name

//type5:html(): This method is used to find and get the first HTML element from the set of matched HTML elements.

//Syntax: selector.html(), for example $("div").html();

//

//type6: text(): This method is used get the combined text contents of matched elements.This method cannot be used for input or form elements.

//Syntax: selector.text()

$(document).ready(function () {

var title = $("em").attr("title");

$("#divid").text(title);

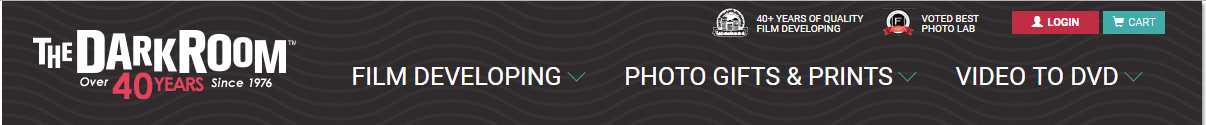
});

//In this example text method is used to get the title of the element.

//These are the commonly used attribute methods in jQuery.There are many more methods than we have in this list.You can use these methods in a variety of ways in different situations.

/////////////////////////I.jQuery attribute methods ends

# /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Word press Script Examples\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/



///\*\*\*function.php\*\*\*\*\*\*\*\*\*///

<?php

function getproductbycategory()

{

$args = array( 'post\_type' => 'product', 'product\_cat' => $\_POST['product\_cat'] , 'orderby' => 'rand' );

$loop = new WP\_Query( $args );

$product='<ul>';

if ($loop->have\_posts()) {

while ($loop->have\_posts()){

$loop->the\_post();

$product.='<li>'.get\_the\_title().'</li>';

}

wp\_reset\_query();

}

else

{

$product.='<li>No product found</li>';

}

$product.='</ul>';

echo $product;

wp\_die();

}

add\_action( 'wp\_ajax\_getproductbycategory', 'getproductbycategory' );

/\*\*\*\*\*\*\*\*\*childscript.js\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

jQuery( document ).ready(function() {

jQuery('#top-area').find('.top-area-items').after('<div class="top-area-items inline-inside custom-product-list"></div>');

});

// custom-product-cat- which is from existing page

jQuery(document).on("click",".custom-product-cat a",function() {

var thisval=jQuery(this);

if(jQuery(thisval).hasClass('toggle\_on')==true)

{

jQuery(thisval).removeClass('toggle\_on');

jQuery('.custom-product-list').html('');

}

else

{

var data = {

'action': 'getproductbycategory',

'product\_cat': jQuery(this).attr('title')

};

// since 2.8 ajaxurl is always defined in the admin header and points to admin-ajax.php

jQuery.post(the\_ajax\_scripts.ajaxurl, data, function(response) {

jQuery('.custom-product-list').html(response);

jQuery('.custom-product-cat a').removeClass('toggle\_on');

jQuery(thisval).addClass('toggle\_on');

});

}

});

/\*\*\*\*\*\*\*\* append function anywhere in word press page using below code starts\*\*\*\*\*\*\*\*/

//////function.php

function test(){

echo "<div class='test1' style='display:none'><h1>TEST</h1></div>";

}

//add\_action('get\_footer', 'test'); //append function before footer without display:none

//add\_action('wp\_head', 'test'); //append function after header without display:none

add\_action('test', 'test'); //where ever want to append using jQuery with display:none

//////child.js

jQuery( document ).ready(function() {

jQuery('.gem-slideshow').after(jQuery('.test1').html());

$("#container").before($('.test1').html());

});

/\*\*\*\*\*\*\*\* append function anywhere in word press page using below code ends\*\*\*\*\*\*\*\*/

# /\*\*\*\*\*\*\*\*\*\*\*Wordpress child theme creation\*\*\*\*\*\*\*\*\*\*/

<https://developer.wordpress.org/themes/advanced-topics/child-themes/>

WORDPRESS SAMPLEs

create short code for events:

<?php

/\*\*\*\*\*\*\*\*\* events in table format starts \*\*\*\*\*\*\*\*\*\*\*\*\*\*/

function wpc\_event\_details(){

global $wpdb;

$query="SELECT \* FROM (SELECT DISTINCT p.ID, p.post\_title, p.post\_content,

(SELECT meta\_value FROM **$wpdb->postmeta** where meta\_key ='\_EventStartDate' AND post\_id = p.ID) as EventDate

FROM **$wpdb->posts** p

WHERE p.post\_type ='tribe\_events' AND p.post\_status ='publish'

AND

p.ID IN (SELECT post\_id FROM **$wpdb->postmeta**

where meta\_key ='\_EventStartDate' and meta\_value >= NOW())) e

order by EventDate asc";

$result = $wpdb->get\_results($query);

return $result;

}

?>

<?php

/\*\*\*\*\*\*\*\*\*\* latest events excerpt grid starts \*\*\*\*\*\*\*\*\*/

add\_shortcode('list\_events\_detailstest','custom\_events\_detailsnew');

function custom\_events\_detailsnew(){

global $wpdb;

$list = '';

/\*\*\*get query result \*\*\*\*\*/

$query="SELECT \* FROM **$wpdb->posts**

where post\_type = 'post' and post\_status = 'publish'

AND ID IN (SELECT object\_id FROM **$wpdb->term\_relationships**

WHERE term\_taxonomy\_id IN (SELECT term\_id FROM **$wpdb->terms** where name='tribe\_event'))

order by post\_date desc limit 0,2";

$result = $wpdb->get\_results($query);

/\*or\*/

$result = wpc\_event\_details();

/\*\*\*get query result \*\*\*\*\*/

$list .='<div class="table-2 csd\_eventlisttable1">

<table width="100%">

<thead>

<tr>

<th align="center"></th>

<th align="left">Date</th>

<th align="left">Description</th>

<th align="left">Time</th>

<th align="left">Link</th>

</tr>

</thead>

<tbody>';

if(!empty($result)){

foreach($result as $event){

$post\_id = $event->ID;

$startdate = date('d/m/Y', strtotime($event->EventDate));

$starttime = date('h:i a', strtotime($event->EventDate));

$eventend = get\_post\_meta($post\_id, '\_EventEndDate', true);

$endtime = date('h:i a', strtotime($eventend));

$link = get\_post\_meta($post\_id, '\_event\_bookinglink', true);

//$link = get\_permalink($post\_id);

$img\_url = get\_the\_post\_thumbnail\_url($post\_id);

if(!$img\_url){

$img\_url = '/wp-content/uploads/2019/03/no-image.jpg';

}

$shortdesc = get\_post\_meta($post\_id, '\_event\_shortdesc', true);

if($shortdesc){

$shortdesc = ' - '.$shortdesc;

}

$content = $event->post\_content;

//$content = apply\_filters('the\_content', $content);

$content = str\_replace(']]>', ']]&gt;', $content);

//$content = wp\_trim\_words($content,'10');

$content=substr(strip\_tags($content), 0, 90);

/\*$eventdesc = $event->post\_excerpt;

if(!$eventdesc){

$eventdesc = $event->post\_excerpt;

}\*/

$list .=' <tr class="event\_'.$post\_id.'">

<td><a class="hover-type-none" > <img src="'.$img\_url.'" alt="" draggable="false" width=125px> </a></td>

<td>'.$startdate.'</td>

<td>'.$event->post\_title.$shortdesc.'</td>

<td>'.$starttime.' - '.$endtime.'</td>

<td><a href="'.$link.'">'.$link.'</a></td>

</tr>';

}

}

$list .=' </tbody>

</table>

</div>';

return $list;

}

/\*\*\*\*\*\*\*\*\*\* latest events excerpt grid ends \*\*\*\*\*\*\*\*\*/

Insert and update meta

<?php

/\*\*\*\*\*\*\*\*\*\*\* custom field for eventlist short description \*\*\*\*\*\*\*\*\*\*\*\*\*/

add\_action( 'add\_meta\_boxes', 'custom\_add\_events\_metaboxes' );

function custom\_add\_events\_metaboxes() {

add\_meta\_box(

'wpc\_events\_short\_desc',

'Event Short Description',

'wpc\_events\_short\_desc',

'tribe\_events',

'normal',

'high'

);

add\_meta\_box(

'wpc\_events\_booking\_link',

'Event Booking Link',

'wpc\_events\_booking\_link',

'tribe\_events',

'normal',

'high'

);

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\* to display inputbox in admin page event booking link design \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

function wpc\_events\_booking\_link($post){

$link = get\_post\_meta( $post->ID, '\_event\_bookinglink', true );

echo '<table class="eventtable">

<tbody>

<tr>

<td style="width:172px;">URL:</td>

<td>

<input tabindex="2014" type="text" id="event\_booking\_link" name="event\_booking\_link" size="25" value="'.$link.'" placeholder="example.com" style="width: 100%;">

</td>

</tr>

</tbody>

</table>';

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* to display inputbox in admin page event short desc design \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

function wpc\_events\_short\_desc($post){

$short\_desc = get\_post\_meta( $post->ID, '\_event\_shortdesc', true );

echo '<table class="eventtable">

<tbody>

<tr>

<td style="width:172px;">Description:</td>

<td>

<input tabindex="2014" type="text" id="event\_short\_desc" name="event\_short\_desc" size="25" value="'.$short\_desc.'" placeholder="Short description." style="width: 100%;">

</td>

</tr>

</tbody>

</table>';

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* insert or update field data \*\*\*\*\*\*\*\*Action Example\*\*\*\*\*\*\*\*\*\*\*/

add\_action('save\_post', 'wpc\_events\_save\_details');

function wpc\_events\_save\_details(){

global $post;

$post\_id = $post->ID;

if(!current\_user\_can( 'edit\_post', $post\_id ) ) {

return;

}

if ( "tribe\_events" != $post->post\_type ){

return;

}

if ( isset( $\_POST['event\_short\_desc'] ) ) {

if( get\_post\_meta( $post\_id, '\_event\_shortdesc', true )){

update\_post\_meta($post\_id, "\_event\_shortdesc", $\_POST['event\_short\_desc']);

}else{

add\_post\_meta($post\_id, "\_event\_shortdesc", $\_POST['event\_short\_desc']);

}

}

if ( isset( $\_POST['event\_booking\_link'] ) ) {

if( get\_post\_meta( $post\_id, '\_event\_bookinglink', true )){

update\_post\_meta($post\_id, "\_event\_bookinglink", $\_POST['event\_booking\_link']);

}else{

add\_post\_meta($post\_id, "\_event\_bookinglink", $\_POST['event\_booking\_link']);

}

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Filter Example\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* This changes the event link to the event website URL if that is set.

\* NOTE: Comment out the add\_filter() line to disable this function. \*/

function tribe\_set\_link\_website ( $link, $postId ) {

$website\_url = tribe\_get\_event\_website\_url( $postId );

// Only swaps link if set

if ( !empty( $website\_url ) ) {

$link = $website\_url;

}

return $link;

}

add\_filter( 'tribe\_get\_event\_link', 'tribe\_set\_link\_website', 100, 2 );

?>

/\*event page get event cat and term\*/

add\_shortcode('custom\_events\_cat\_terms','custom\_events\_cat\_terms');

add\_action('get\_footer', 'custom\_events\_cat\_terms');

function custom\_events\_cat\_terms(){

global $wpdb;

$list = '';

$result = wpc\_event\_details();

$list .='<div class="table-2 csd\_eventlisttable1">

<div>Important Event Dates</div>

<table width="100%">

<tbody>';

if(!empty($result)){

foreach($result as $event){

$post\_id = $event->ID;

$startdate = date('d/m/Y', strtotime($event->EventDate));

$starttime = date('h:i a', strtotime($event->EventDate));

$eventend = get\_post\_meta($post\_id, '\_EventEndDate', true);

$endtime = date('h:i a', strtotime($eventend));

$link = get\_post\_meta($post\_id, '\_event\_bookinglink', true);

//$link = get\_permalink($post\_id);

$img\_url = get\_the\_post\_thumbnail\_url($post\_id);

if(!$img\_url){

$img\_url = '/wp-content/uploads/2019/03/upcoming-dates-filter-image.jpg';

}

$shortdesc = get\_post\_meta($post\_id, '\_event\_shortdesc', true);

if($shortdesc){

$shortdesc = ' - '.$shortdesc;

}

/\*$eventdesc = $event->post\_excerpt;

if(!$eventdesc){

$eventdesc = $event->post\_excerpt;

}\*/

$eventcat\_id=get\_post\_meta($post\_id, '\_yoast\_wpseo\_primary\_tribe\_events\_cat', true);

$terms=get\_term( $eventcat\_id, 'tribe\_events\_cat' );

//print\_r(get\_term( $eventcat\_id, 'tribe\_events\_cat' ));

//echo $terms->name;

$event\_cat\_color=get\_option('teccc\_options',true);

//var\_dump( $event\_cat\_color );

//echo $event\_cat\_color->$eventcat\_id;

//echo $event\_cat\_color[$terms->name.'-background'];

$color\_id=$event\_cat\_color[$terms->name.'-background'];

$list .=' <tr class="event\_'.$post\_id.'">

<td style="background: '.$color\_id.'; padding:20px;"></td>

<td>'.$event->post\_title.$shortdesc.'</td>

</tr>';

}

}

$list .=' </tbody>

</table>

</div>';

return $list;

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* create short code in different way \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

<?php

// Avoid direct calls to this file

if ( !defined( 'ABSPATH' ) ) {

header( 'Status: 403 Forbidden' );

header( 'HTTP/1.1 403 Forbidden' );

exit();

}

$mydb = new wpdb('m8\_classic','root',' ','localhost');

$rows = $mydb->get\_results("

SELECT P.ID, P.post\_title, P.post\_content, P.post\_author, meta\_value, thumb

FROM wp\_posts AS P

LEFT JOIN wp\_postmeta AS PM on PM.post\_id = P.ID

WHERE P.post\_type = 'post' and P.post\_status = 'publish'

and meta\_key = 'job\_title'

and meta\_key = 'company\_email' //2nd meta\_key I need to work

ORDER BY P.post\_date DESC

");

foreach ($rows as $row) :

echo $row->meta\_value;

endforeach;

/\*\*

\* Events calendar shortcode addon main class

\*

\* @package events-calendar-shortcode

\* @author Dandelion Web Design Inc.

\* @version 1.0.11

\*/

class Events\_Calendar\_Shortcode

{

/\*\*

\* Current version of the plugin.

\*

\* @since 1.0.0

\*/

const VERSION = '1.0.11';

/\*\*

\* Constructor. Hooks all interactions to initialize the class.

\*

\* @since 1.0.0

\* @access public

\*

\* @see add\_shortcode()

\*/

public function \_\_construct()

{

add\_shortcode('ecs-list-events', array($this, 'ecs\_fetch\_events') ); // link new function to shortcode name

} // END \_\_construct()

public function ecs\_fetch\_events( $atts )

{

/\*\*

\* Check if events calendar plugin method exists

\*/

if ( !function\_exists( 'tribe\_get\_events' ) ) {

return;

}

global $wp\_query, $post;

$output = '';

$atts = shortcode\_atts( array(

'cat' => '',

'month' => '',

'limit' => 5,

'eventdetails' => 'true',

'time' => null,

'past' => null,

'venue' => 'false',

'author' => null,

'message' => sprintf(

\_\_( 'There are no upcoming %s at this time.', 'tribe-events-calendar' ),

tribe\_get\_event\_label\_plural() ),

'key' => 'End Date',

'order' => 'ASC',

'viewall' => 'false',

'excerpt' => 'false',

'thumb' => 'false',

'thumbwidth' => '',

'thumbheight' => '',

'contentorder' => 'title, thumbnail, excerpt, date, venue',

'event\_tax' => '',

), $atts, 'ecs-list-events' );

// Category

if ( $atts['cat'] ) {

if ( strpos( $atts['cat'], "," ) !== false ) {

$atts['cats'] = explode( ",", $atts['cat'] );

$atts['cats'] = array\_map( 'trim', $atts['cats'] );

} else {

$atts['cats'] = $atts['cat'];

}

$atts['event\_tax'] = array(

'relation' => 'OR',

array(

'taxonomy' => 'tribe\_events\_cat',

'field' => 'name',

'terms' => $atts['cats'],

),

array(

'taxonomy' => 'tribe\_events\_cat',

'field' => 'slug',

'terms' => $atts['cats'],

)

);

}

// Past Event

$meta\_date\_compare = '>=';

$meta\_date\_date = date( 'Y-m-d' );

if ( $atts['time'] == 'past' || !empty( $atts['past'] ) ) {

$meta\_date\_compare = '<';

}

// Key

if ( str\_replace( ' ', '', trim( strtolower( $atts['key'] ) ) ) == 'startdate' ) {

$atts['key'] = '\_EventStartDate';

} else {

$atts['key'] = '\_EventEndDate';

}

// Date

$atts['meta\_date'] = array(

array(

'key' => $atts['key'],

'value' => $meta\_date\_date,

'compare' => $meta\_date\_compare,

'type' => 'DATETIME'

)

);

// Specific Month

if ( $atts['month'] == 'current' ) {

$atts['month'] = date( 'Y-m' );

}

if ($atts['month']) {

$month\_array = explode("-", $atts['month']);

$month\_yearstr = $month\_array[0];

$month\_monthstr = $month\_array[1];

$month\_startdate = date($month\_yearstr . "-" . $month\_monthstr . "-1");

$month\_enddate = date($month\_yearstr . "-" . $month\_monthstr . "-t");

$atts['meta\_date'] = array(

array(

'key' => $atts['key'],

'value' => array($month\_startdate, $month\_enddate),

'compare' => 'BETWEEN',

'type' => 'DATETIME'

)

);

}

$posts = get\_posts( array(

'post\_type' => 'tribe\_events',

'posts\_per\_page' => $atts['limit'],

'tax\_query'=> $atts['event\_tax'],

'meta\_key' => $atts['key'],

'orderby' => 'meta\_value',

'author' => $atts['author'],

'order' => $atts['order'],

'meta\_query' => array( $atts['meta\_date'] )

) );

if ($posts) {

$output .= '<ul class="ecs-event-list">';

$atts['contentorder'] = explode( ',', $atts['contentorder'] );

foreach( $posts as $post ) :

setup\_postdata( $post );

$li\_classes = 'ecs-event';

$start\_date = tribe\_get\_start\_date($post, false, 'Y-m-d');

$today = date( 'Y-m-d', current\_time( 'timestamp' ) );

$tomorrow = date( 'Y-m-d', strtotime( 'tomorrow', current\_time( 'timestamp' ) ) );

if ($start\_date == $today) {

$li\_classes .= ' ecs-today';

} else if ($start\_date == $tomorrow) {

$li\_classes .= ' ecs-tomorrow';

}

$output .= '<li class="' . $li\_classes .'">';

// Put Values into $output

foreach ( $atts['contentorder'] as $contentorder ) {

switch ( trim( $contentorder ) ) {

case 'title' :

$output .= '<h4 class="entry-title summary">' .

'<a href="' . tribe\_get\_event\_link() . '" rel="bookmark">' . apply\_filters( 'ecs\_event\_list\_title', get\_the\_title(), $atts ) . '</a>

</h4>';

break;

case 'thumbnail' :

if( self::isValid($atts['thumb']) ) {

$thumbWidth = is\_numeric($atts['thumbwidth']) ? $atts['thumbwidth'] : '';

$thumbHeight = is\_numeric($atts['thumbheight']) ? $atts['thumbheight'] : '';

if( !empty($thumbWidth) && !empty($thumbHeight) ) {

$output .= get\_the\_post\_thumbnail($post->ID, array($thumbWidth, $thumbHeight) );

} else {

$size = ( !empty($thumbWidth) && !empty($thumbHeight) ) ? array( $thumbWidth, $thumbHeight ) : 'medium';

if ( $thumb = get\_the\_post\_thumbnail( $post->ID, $size ) ) {

$output .= '<a href="' . tribe\_get\_event\_link() . '">';

$output .= $thumb;

$output .= '</a>';

}

}

}

break;

case 'excerpt' :

if( self::isValid($atts['excerpt']) ) {

$excerptLength = is\_numeric($atts['excerpt']) ? $atts['excerpt'] : 100;

$output .= '<p class="ecs-excerpt">' .

self::get\_excerpt($excerptLength) .

'</p>';

}

break;

case 'date' :

if( self::isValid($atts['eventdetails']) ) {

$output .= '<span class="duration time">' . apply\_filters( 'ecs\_event\_list\_details', tribe\_events\_event\_schedule\_details(), $atts ) . '</span>';

}

break;

case 'venue' :

if( self::isValid($atts['venue']) ) {

$output .= '<span class="duration venue">' . apply\_filters( 'ecs\_event\_list\_venue', ' <em>at</em> ' . tribe\_get\_venue(), $atts ) . '</span>';

}

break;

}

}

$output .= '</li>';

endforeach;

$output .= '</ul>';

if( self::isValid($atts['viewall']) ) {

$output .= '<span class="ecs-all-events"><a href="' . apply\_filters( 'ecs\_event\_list\_viewall\_link', tribe\_get\_events\_link(), $atts ) .'" rel="bookmark">' . translate( apply\_filters( 'ecs\_event\_list\_viewall\_text', 'View All Events', $atts ), 'tribe-events-calendar' ) . '</a></span>';

}

} else { //No Events were Found

$output .= translate( $atts['message'], 'tribe-events-calendar' );

} // endif

wp\_reset\_query();

return $output;

}

/\*\*

\* Checks if the plugin attribute is valid

\*

\* @since 1.0.5

\*

\* @param string **$prop**

\* @return boolean

\*/

private function isValid( $prop )

{

return ($prop !== 'false');

}

/\*\*

\* Fetch and trims the excerpt to specified length

\*

\* @param integer **$limit** Characters to show

\* @param string **$source** content or excerpt

\*

\* @return string

\*/

private function get\_excerpt( $limit, $source = null )

{

$excerpt = get\_the\_excerpt();

if( $source == "content" ) {

$excerpt = get\_the\_content();

}

$excerpt = preg\_replace(" (\[.\*?\])", '', $excerpt);

$excerpt = strip\_tags( strip\_shortcodes($excerpt) );

$excerpt = substr($excerpt, 0, $limit);

$excerpt = trim(preg\_replace( '/\s+/', ' ', $excerpt));

if ( strlen( $excerpt ) >= $limit ) {

$excerpt .= '...';

}

return $excerpt;

}

}

/\*\*

\* Instantiate the main class

\*

\* @since 1.0.0

\* @access public

\*

\* @var object **$events\_calendar\_shortcode** holds the instantiated class {@uses Events\_Calendar\_Shortcode}

\*/

global $events\_calendar\_shortcode;

$events\_calendar\_shortcode = new Events\_Calendar\_Shortcode();

?>

<https://developer.wordpress.org/reference/classes/wp_query/#post-type-parameters>

jQuery(window).scroll(function() {

if (jQuery(this).scrollTop() > 1) {

jQuery('#masthead').addClass("sticky-header");

jQuery(".custom-logo").attr("src", "http://resortphotos.com.au/wp-content/uploads/2019/07/RP\_logo\_colour-1.png");

}

else

{

jQuery('#masthead').removeClass("sticky-header");

jQuery(".custom-logo").attr("src", "http://resortphotos.com.au/wp-content/uploads/2019/07/RP\_logo\_white.png");

}

});

if (document.documentElement.clientWidth < 900) {

jQuery( document ).ready(function() {

jQuery('#menu-toggle').addClass("fa fa-bars");

});

}

if (document.documentElement.clientWidth < 900) {

jQuery(window).scroll(function() {

if (jQuery(this).scrollTop() < 1) {

jQuery(".custom-logo").attr("src", "http://resortphotos.com.au/wp-content/uploads/2019/07/RP\_logo\_colour-1.png");

}

});

}

jQuery(document).ready(function ($) {

$('<div class="freecall"><span>FREECALL US ON</span><a href="tel:1300735796"><i class="fa fa-phone" aria-hidden="true"></i><span>1300 735 796</span></a><span class="bookonline"><a href="/book-online">Book Online</a></span></div>').prependTo('.main\_header');

$('<div id="whoweare"></div>').prependTo('.home-who.whoweare .row');

$(".btn\_large.btn\_type5.left").click(function () {

$('html,body').animate({

scrollTop: $("#whoweare").offset().top - 150

},

1600);

});

$('<a href="/requestaquote/" target="\_self" class="shortcode\_button btn\_normal btn\_type5"><i class="fa fa-none"></i>Request Prices</a>').appendTo('.package .iconbox\_body');

var maxHeight = 0;

var scrnwidth = jQuery(window).width();

if (scrnwidth > 1023 && scrnwidth < 1441) {

jQuery('.home-specialities .iconbox\_wrapper').each(function () {

if (jQuery(this).height() > maxHeight) { maxHeight = jQuery(this).height(); }

});

maxHeight = parseInt(maxHeight) - 50;

jQuery('.iconbox\_wrapper').height(maxHeight);

}

});