|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| <hitle> | Html to Wordpress | <chare> | 1 | <pext> | background:url(./images/  url(<?php echo get\_stylesheet\_directory\_uri(); ?>/images/  src="./images/  src="<?php echo get\_stylesheet\_directory\_uri(); ?>/images/  <?php echo get\_site\_url(); ?>/news/  get\_template\_part('template-parts/common-blog');  1. download underscore theme  2. delete unnecessary files  3. delete default styles  4. make header and footer  5. make extra.css, add pmhwp.css  6. make news/detail page | </end> |
| <hitle> | Page hierarchy | <chare> | 1 | <pext> | ------Home page----------  By default, WordPress sets your site’s home page to display your latest blog posts.  This page is called the blog posts index. You can also set your blog posts to display on a separate static page.  The template file home.php is used to render the blog posts index, whether it is being used as the front page or on separate static page.  If home.php does not exist, WordPress will use index.php.  If front-page.php exists, it will override the home.php template.  1. home.php  2. index.php  ---------Front page----------  The front-page.php template file is used to render your site’s front page, whether the front page displays the blog posts index (mentioned above) or a static page.  The front page template takes precedence over the blog posts index (home.php) template.  If the front-page.php file does not exist, WordPress will either use the home.php or page.php files depending on the setup in Settings → Reading. If neither of those files exist, it will use the index.php file.  1. front-page.php  Used for both “your latest posts” or “a static page” as set in the front page displays section of Settings → Reading.  2. home.php  If WordPress cannot find front-page.php and “your latest posts” is set in the front page displays section, it will look for home.php. Additionally, WordPress will look for this file when the posts page is set in the front page displays section.  3. page.php  When “front page” is set in the front page displays section.  4. index.php  When “your latest posts” is set in the front page displays section but home.php does not exist or when front page is set but page.php does not exist.  ------------Privacy Policy Page display---------  The privacy-policy.php template file is used to render your site’s Privacy Policy page.  The Privacy Policy page template takes precedence over the static page (page.php) template.  If the privacy-policy.php file does not exist, WordPress will either use the page.php or singular.php files depending on the available templates.  If neither of those files exist, it will use the index.php file.  1. privacy-policy.php  Used for the Privacy Policy page set in the Change your Privacy Policy page section of Settings → Privacy.  2. custom template file  The page template assigned to the page. See get\_page\_templates().  3. page-{slug}.php  If the page slug is privacy, WordPress will look to use page-privacy.php.  4. page-{id}.php  If the page ID is 6, WordPress will look to use page-6.php.  5. page.php  6. singular.php  7. index.php  --------------------------Single Post--------------------------  The single post template file is used to render a single post. WordPress uses the following path:  1. single-{post-type}-{slug}.php  (Since 4.4) First, WordPress looks for a template for the specific post. For example, if post type is product and the post slug is dmc-12, WordPress would look for single-product-dmc-12.php.  2. single-{post-type}.php  If the post type is product, WordPress would look for single-product.php.  3. single.php  WordPress then falls back to single.php.  4. singular.php  Then it falls back to singular.php.  5. index.php  Finally, as mentioned above, WordPress ultimately falls back to index.php.  --------------------------Single Page--------------------------  The template file used to render a static page (page post-type). Note that unlike other post-types, page is special to WordPress and uses the following path:  1. custom template file  The page template assigned to the page. See get\_page\_templates().  2. page-{slug}.php  If the page slug is recent-news, WordPress will look to use page-recent-news.php.  3. page-{id}.php  If the page ID is 6, WordPress will look to use page-6.php.  4. page.php  5. singular.php  6. index.php  --------------------------Category--------------------------  Rendering category archive index pages uses the following path in WordPress:  1. category-{slug}.php  If the category’s slug is news, WordPress will look for category-news.php.  2. category-{id}.php  If the category’s ID is 6, WordPress will look for category-6.php.  3. category.php  4. archive.php  5. index.php  --------------------------Tag--------------------------  To display a tag archive index page, WordPress uses the following path:  1. tag-{slug}.php  If the tag’s slug is sometag, WordPress will look for tag-sometag.php.  2. tag-{id}.php  If the tag’s ID is 6, WordPress will look for tag-6.php.  3. tag.php  4. archive.php  5. index.php  --------------------------Custom Taxonomies--------------------------  Custom taxonomies use a slightly different template file path:  1. taxonomy-{taxonomy}-{term}.php  If the taxonomy is sometax, and taxonomy’s term is someterm, WordPress will look for taxonomy-sometax-someterm.php. In the case of post formats, the taxonomy is ‘post\_format’ and the terms are ‘post-format-{format}. i.e. taxonomy-post\_format-post-format-link.php for the link post format.  2. taxonomy-{taxonomy}.php  If the taxonomy were sometax, WordPress would look for taxonomy-sometax.php.  3. taxonomy.php  4. archive.php  5. index.php  --------------------------Custom Post Types--------------------------  Custom Post Types use the following path to render the appropriate archive index page.  1. archive-{post\_type}.php  If the post type is product, WordPress will look for archive-product.php.  2. archive.php  3. index.php  (For rendering a single post type template, refer to the single post display section above.)  --------------------------Author display--------------------------  Based on the above examples, rendering author archive index pages is fairly explanatory:  1. author-{nicename}.php – If the author’s nice name is matt, WordPress will look for author-matt.php.  2. author-{id}.php – If the author’s ID were 6, WordPress will look for author-6.php.  3. author.php  4. archive.php  5. index.php  --------------------------Date--------------------------  Date-based archive index pages are rendered as you would expect:  1. date.php  2. archive.php  3. index.php  --------------------------Search Result--------------------------  Search results follow the same pattern as other template types:  1. search.php  2. index.php  --------------------------404 (Not Found) --------------------------  Likewise, 404 template files are called in this order:  404.php  index.php  --------------------------Attachment--------------------------  Rendering an attachment page (attachment post-type) uses the following path:  1. {MIME-type}.php  can be any MIME type (For example: image.php, video.php, pdf.php). For text/plain, the following path is used (in order):  1-1. text-plain.php  1-2. plain.php  1-3. text.php  2. attachment.php  3. single-attachment-{slug}.php  For example, if the attachment slug is holiday, WordPress would look for single- attachment-holiday.php.  4. single-attachment.php  5. single.php  6. singular.php  7. index.php  --------------------------Embeds--------------------------  The embed template file is used to render a post which is being embedded. Since 4.5, WordPress uses the following path:  1. embed-{post-type}-{post\_format}.php  First, WordPress looks for a template for the specific post. For example, if its post type is post and it has the audio format, WordPress would look for embed-post-audio.php.  2. embed-{post-type}.php  If the post type is product, WordPress would look for embed-product.php.  3. embed.php  WordPress then falls back to embed.php.  4. Finally, WordPress ultimately falls back to its own wp-includes/theme-compat/embed.php template.  --------------------------Non-ASCII Character Handling--------------------------  Since WordPress 4.7, any dynamic part of a template name which includes non-ASCII characters in its name actually supports both the un-encoded and the encoded form, in that order. You can choose which to use.  Here’s the page template hierarchy for a page named “Hello World ” with an ID of 6:  1. page-hello-world- .php  2. page-hello-world-%f0%9f%98%80.php  3. page-6.php  4. page.php  5. singular.php  The same behaviour applies to post slugs, term names, and author nicenames.  --------------------------Filter Hierarchy--------------------------  The WordPress template system lets you filter the hierarchy. This means that you can insert and change things at specific points of the hierarchy. The filter (located in the get\_query\_template() function) uses this filter name: "{$type}\_template" where $type is the template type.  Here is a list of all available filters in the template hierarchy:  embed\_template  404\_template  search\_template  frontpage\_template  home\_template  privacypolicy\_template  taxonomy\_template  attachment\_template  single\_template  page\_template  singular\_template  category\_template  tag\_template  author\_template  date\_template  archive\_template  index\_template | </end> |
| <hitle> | post\_meta data | <chare> | 1 | <pext> | get\_post\_meta( get\_the\_ID(), 'thumbnailurl', true )  update\_post\_meta($post->ID, 'views', $views + 1); | </end> |
| <hitle> | Format date | <chare> | 1 | <pext> | date\_format(date\_create($date),"Y年 m月"); | </end> |
| <hitle> | Change excerpt length | <chare> | 1 | <pext> | function my\_excerpt\_length($length){ return 80; }  add\_filter(‘excerpt\_length’, ‘my\_excerpt\_length’); | </end> |
| <hitle> | Order by meta value | <chare> | 1 | <pext> | $args = array(  'post\_type' => 'product',  'orderby' => 'meta\_value\_num',  'meta\_key' => 'price',  );  $query = new WP\_Query( $args ); | </end> |
| <hitle> | Custom pos type monthly archive permalink | <chare> | 1 | <pext> | /\*\*  \* Generate date archive rewrite rules for a given custom post type  \* @param string $cpt slug of the custom post type  \* @return rules returns a set of rewrite rules for Wordpress to handle  \*/  function cpt\_generate\_date\_archives($cpt, $wp\_rewrite) {  $rules = array();  $post\_type = get\_post\_type\_object($cpt);  $slug\_archive = $post\_type->has\_archive;  if ($slug\_archive === false) return $rules;  if ($slug\_archive === true) {  $slug\_archive = $post\_type->name;  }  $dates = array(  array(  'rule' => "([0-9]{4})/([0-9]{1,2})/([0-9]{1,2})",  'vars' => array('year', 'monthnum', 'day')),  array(  'rule' => "([0-9]{4})/([0-9]{1,2})",  'vars' => array('year', 'monthnum')),  array(  'rule' => "([0-9]{4})",  'vars' => array('year'))  );  foreach ($dates as $data) {  $query = 'index.php?post\_type='.$cpt;  $rule = $slug\_archive.'/'.$data['rule'];  $i = 1;  foreach ($data['vars'] as $var) {  $query.= '&'.$var.'='.$wp\_rewrite->preg\_index($i);  $i++;  }  $rules[$rule."/?$"] = $query;  $rules[$rule."/feed/(feed|rdf|rss|rss2|atom)/?$"] = $query."&feed=".$wp\_rewrite->preg\_index($i);  $rules[$rule."/(feed|rdf|rss|rss2|atom)/?$"] = $query."&feed=".$wp\_rewrite->preg\_index($i);  $rules[$rule."/page/([0-9]{1,})/?$"] = $query."&paged=".$wp\_rewrite->preg\_index($i);  }  return $rules;  } | </end> |
| <hitle> | Check Contain String | <chare> | 1 | <pext> | if(strpos("PHP string example.", "PHP") !== false){  echo "Found!";  } | </end> |
| <hitle> | Function.php init | <chare> | 1 | <pext> | /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  add\_action( 'after\_setup\_theme', 'custom\_setup\_theme' );  function custom\_setup\_theme() {  $new\_page\_list = [  'about',  'company',  'service',  'contact',  'success',  'failed',  'privacypolicy',  'request',  'request-confirm',  'request-success',  'request-failed',  'blog'  ];  foreach($new\_page\_list as $page\_slug) {  $new\_page = array(  'post\_type' => 'page', // Post Type Slug eg: 'page', 'post'  'post\_title' => $page\_slug, // Title of the Content  'post\_status' => 'publish', // Post Status  'post\_name' => $page\_slug // Slug of the Post  );  if (!get\_page\_by\_path( $page\_slug, OBJECT, 'page')) { // Check If Page Not Exits  $new\_page\_id = wp\_insert\_post($new\_page);  }  }  // for($i=0;$i<10;$i++) {  // $new\_page = array(  // 'post\_type' => 'post', // Post Type Slug eg: 'page', 'post'  // 'post\_title' => 'タイトルテキストがあります。タイトルテキストがあります。タイトルテキストがあります。', // Title of the Content  // 'post\_content' => '内容 テキスト内容 テキスト内容 テキスト内容 テキスト内容 テキスト内容 テキスト内容 テキスト内容 テキスト内容 テキストがあります。内容 テキスト内容 テキスト内容 テキスト内容 テキスト内容 テキスト内容 テキスト内容 テキスト内容 テキスト内容 テキストがあります。',  // 'post\_status' => 'publish', // Post Status  // 'post\_name' => 'post'.$i // Slug of the Post  // );  // wp\_insert\_post($new\_page);  // }  } | </end> |
| <hitle> | Fixed image div width-height | <chare> | 1 | <pext> | .discography\_left\_dp {  width:160px;  padding-top: 160px;  overflow: hidden;  position: relative;  }  .discography\_left\_dp img{  position: absolute;  top: 0;  left: 0;  right: 0;  bottom: 0;  max-width: 100%;  max-height: 100%;  min-width: 100%;  min-height: 100%;  object-fit: cover;  } | </end> |
| <hitle> | Post query and loop | <chare> | 1 | <pext> | $args = array(  'post\_type' => 'post',  'paged' => get\_query\_var('paged'),  'orderby' => 'publish\_date',  'order' => 'DESC',  //'posts\_per\_page' => -1, //page limit  );  $query = new WP\_Query($args);  if ( $query->have\_posts() ) :  while ( $query->have\_posts() ) :  $query->the\_post();  $loop\_id = get\_the\_id();  $loop\_title = get\_the\_title();  $loop\_content = get\_the\_excerpt();  $loop\_date = get\_the\_date('Y/m/d');  $loop\_terms = get\_the\_terms($loop\_id, 'disco\_cat'); //get the custom category  $loop\_cat = $loop\_terms[0]->name;  $loop\_permalink = get\_the\_permalink($loop\_id);  $loop\_thumb\_url = get\_thumb\_url($loop\_id);  endwhile;  endif; | </end> |
| <hitle> | Relative posts loop | <chare> | 2 | <pext> | $terms = get\_the\_terms( get\_the\_ID(), 'category' );  $term\_list = wp\_list\_pluck( $terms, 'slug' );  $related\_args = array(  'post\_type' => 'post',  'posts\_per\_page' => 3,  'post\_status' => 'publish',  'post\_\_not\_in' => array( get\_the\_ID() ),  'orderby' => 'rand',  'tax\_query' => array(  array(  'taxonomy' => 'category',  'field' => 'slug',  'terms' => $term\_list  )  )  );  $query = new WP\_Query( $related\_args );  if ( $query->have\_posts() ) :  while ( $query->have\_posts() ) :  $query->the\_post();  $loop\_id = get\_the\_id();  $loop\_title = get\_the\_title();  $loop\_content = get\_the\_excerpt();  $loop\_date = get\_the\_date('Y.m.d');  $loop\_category\_name = "";  $loop\_category\_objects = get\_the\_category($loop\_id);  foreach($loop\_category\_objects as $cd){  $loop\_category\_name = $cd->cat\_name;  break;  }  $loop\_permalink = get\_the\_permalink($loop\_id);  $loop\_thumb\_url = get\_thumb\_url($loop\_id);  endwhile;  endif; | </end> |
| <hitle> | Get feature/thumbnail image url | <chare> | 1 | <pext> | function get\_thumb\_url($post\_id, $default\_empty = true) {  $thumb\_url = "";  $thumb\_id = get\_post\_thumbnail\_id($post\_id); //get the featured image  $thumb\_url\_array = wp\_get\_attachment\_image\_src($thumb\_id, 'thumbnail-size', true);  if(!empty($thumb\_url\_array)) {  $thumb\_url = $thumb\_url\_array[0];  }  if(strpos($thumb\_url, "default.png") !== false){  $thumb\_url = "";  }  if(empty($thumb\_url) && $default\_empty) {  $thumb\_url = get\_stylesheet\_directory\_uri()."/images/empty.jpg";  }  return $thumb\_url;  } | </end> |
| <hitle> | WP-PageNavi | <chare> | 1 | <pext> | Pagination plugin | </end> |
| <hitle> | Manual post pagenavi | <chare> | 2 | <pext> | $args = array(  'post\_type' => 'news-item',  'posts\_per\_page' => 5,  'paged' => get\_query\_var('paged'),  'orderby' => 'publish\_date',  'order' => 'DESC',  );  $query = new WP\_Query($args);  <?php wp\_pagenavi(array('query' => $query)); ?>  <div class="follow\_code">  <div class="pagination\_wrap">  <?php wp\_pagenavi(array('query' => $query)); ?>  </div>  </div> | </end> |
| <hitle> | Wp-pagenavi css | <chare> | 2 | <pext> | .pagination\_wrap {  display: flex;  align-items: center;  justify-content: center;  padding-top: 50px;  }  .wp-pagenavi {  display: flex;  flex-direction: row;  align-items: center;  justify-content: center;  column-gap: 3px;  }  .wp-pagenavi a,  .wp-pagenavi span {  width: 36px;  height: 36px;  display: flex;  justify-content: center;  align-items: center;  margin: 0;  color: #fff;  border-color: #fff;  font-size: 14px;  }  .wp-pagenavi a.previouspostslink {  width: 94px;  }  .wp-pagenavi a.nextpostslink {  width: 64px;  }  .wp-pagenavi span {  color: #fff;  background-color: #000;  }  .wp-pagenavi a:hover {  color: #000 !important;  background-color: #fff !important;  }  .wp-pagenavi span.current {  background-color: #fff;  color: #000;  }  @media(min-width:0px) and (max-width:767px){  .wp-pagenavi a,  .wp-pagenavi span {  width: 24px;  height: 24px;  font-size: 10px;  }  .wp-pagenavi a.previouspostslink {  width: 70px;  }  .wp-pagenavi a.nextpostslink {  width: 48px;  }  } | </end> |
| <hitle> | Get all tag | <chare> | 1 | <pext> | $tags = get\_tags(array(  'hide\_empty' => false  ));  foreach ($tags as $tag) {  $tag->name;  get\_tag\_link($tag->term\_id);  } | </end> |
| <hitle> | Get all category | <chare> | 1 | <pext> | $categories = get\_categories();  foreach($categories as $category) {  echo '<div class="col-md-4"><a href="' . get\_category\_link($category->term\_id) . '">' . $category->name . '</a></div>';  } | </end> |
| <hitle> | Get category and tags from post | <chare> | 1 | <pext> | $loop\_id = $post\_ids[$i];  $loop\_category\_objects = get\_the\_category($loop\_id);  $loop\_category\_name = "";  foreach($loop\_category\_objects as $cd){  $loop\_category\_name = $cd->cat\_name;  break;  }  $loop\_tags = [];  $loop\_tag\_objects = get\_the\_tags();  if ($loop\_tag\_objects) {  foreach($loop\_tag\_objects as $tag) {  array\_push($loop\_tags, array('name' => $tag->name, 'link' => get\_tag\_link($tag->term\_id)));  }  } | </end> |
| <hitle> | Ranking, Favorite post list | <chare> | 1 | <pext> | \*\*\*\*\*\*\*In the single post\*\*\*\*\*  $visit\_count = get\_post\_meta($loop\_id, 'visit\_count', true);  if(empty($visit\_count)) $visit\_count = 0;  $visit\_count++;  update\_post\_meta($loop\_id, 'visit\_count', $visit\_count);  \*\*\*\*\*\*\*To get ranking, favorite post list\*\*\*\*  <?php  $favor\_args = array(  'post\_type' => 'post',  'orderby' => 'meta\_value\_num',  'meta\_key' => 'visit\_count',  'order' => 'DESC',  'posts\_per\_page' => 5  );  $favor\_query = new WP\_Query($favor\_args);  $favor\_count = 0;  if ( $favor\_query->have\_posts() ) :  while ( $favor\_query->have\_posts() ) :  $favor\_query->the\_post();  $loop\_id = get\_the\_id();  $loop\_title = get\_the\_title();  $loop\_date = get\_the\_date('Y.m.d');  $loop\_permalink = get\_the\_permalink($loop\_id);  $loop\_thumb\_url = get\_thumb\_url($loop\_id);  $favor\_count++;  endwhile;  endif;  ?> | </end> |
| <hitle> | Current category in Archive | <chare> | 1 | <pext> | global $wp\_query;  $query = $wp\_query;  $category\_id = 0;  if(is\_category() || is\_single()){  $category\_id = get\_query\_var('cat');  } | </end> |
| <hitle> | Monthly archieve link | <chare> | 1 | <pext> | function get\_monthly\_archive\_array() {  $years = [];  $years\_args = [  'type' => 'monthly',  'format' => 'custom',  'before' => '',  'after' => '|',  'echo' => false,  'post\_type' => 'post',  // 'order' => 'ASC',  ];    // Get Years  $years\_content = wp\_get\_archives( $years\_args );    if ( ! empty( $years\_content ) ) {  $years\_arr = explode( '|', $years\_content );    // Remove empty whitespace item from array.  $years\_arr = array\_filter( $years\_arr, function ( $item ) {  return trim( $item ) !== '';  } );    foreach ( $years\_arr as $year\_item ) {    $year\_row = trim( $year\_item );  preg\_match( '/href=["\']?([^"\'>]+)["\']>(.+)<\/a>/', $year\_row, $year\_vars );    if ( ! empty( $year\_vars ) && is\_array( $year\_vars ) ) {  $year\_url = ! empty( $year\_vars[1] ) ? $year\_vars[1] : ''; // Eg: http://demo.com/2021/11/  $parts = parse\_url( $year\_url );  $path = ! empty( $parts['path'] ) ? $parts['path'] : '';    $years[] = [  'name' => $year\_vars[2], // Eg: November 2021  'value' => $path // Eg: /2021/11/ , so that we can prefix o suffix our own URL.  ];  }  }  }    return $years;  }  function get\_jp\_date($date) {  $date = date\_create($date);  return date\_format($date,"Y年 m月");  } | </end> |
| <hitle> | Search custom post type | <chare> | 1 | <pext> | <https://pagedart.com/blog/wordpress-search-for-custom-post-type/>  <form id="search" action="' . home\_url( '/' ) . '" method="get">  <input type="hidden" name="post\_type" value="book" />  <input id="s" name="s" type="text" value="' . get\_search\_query() . '" />  </form> | </end> |
| <hitle> | Contact form | <chare> | 1 | <pext> | 1. write php code  if(isset($\_POST['submit-confirm']) && ($\_POST['submit-confirm'] == 'submit-confirm')) {  $field1 = isset($\_POST['field1']) ? $\_POST['field1']: '';  $field2 = isset($\_POST['field2']) ? $\_POST['field2']: '';  $field3 = isset($\_POST['field3']) ? $\_POST['field3']: '';  $field4 = isset($\_POST['field4']) ? $\_POST['field4']: '';  $field5 = isset($\_POST['field5']) ? $\_POST['field5']: '';    $to = 'prgfinal216@gmail.com';  $message = "  お名前 : ".$field1."<br>  メールアドレス : ".$field2."<br>  電話番号 : ".$field3."<br>  お問い合わせ内容 : ".$field4."<br>  お問い合わせ詳細 : <br>".$field5."<br>  ";  $subject = 'お問い合わせがありました';  $headers = "From: " . $field2 . "\r\n";  $headers .= "Reply-To: " . $field2 . "\r\n";  $headers .= "MIME-Version: 1.0\r\n";  $headers .= "Content-Type: text/html; charset=UTF-8\r\n";  if(mail($to, $subject, $message, $headers)) {  echo '<script>location.href="'.home\_url().'/success"</script>';  } else {  echo '<script>location.href="'.home\_url().'/failed"</script>';  }    if(mail($to, $subject, $message, $headers)) {  echo '<script>alert("Success!"); location.href="'.home\_url().'"</script>';  } else {  echo '<script>alert("Failed!"); location.href="'.home\_url().'"</script>';  }  }  2. add form tag  <form action="" method="post" enctype="multipart/form-data">  <div class="formBtn">  <input type="hidden" name="submit-confirm" value="submit-confirm">  <button id="submit-btn" class="btn -type5" type="submit">送信</button>  </div>  </div>  </form>  3. change a tag to button and replace style  {border:none}  button:focus {outline:0;}  4. success page template  <div style="width:100%; display:flex; height:70vh; justify-content: center; align-items: center; padding: 16px;">  Success  </div> | </end> |
| <hitle> | Confirm email/password | <chare> | 2 | <pext> | $("#contact\_form").submit(function(e){  //var email = $('#field3').val();  //var conf\_email = $('#field4').val();  if($("#input\_mail").val() != $("#input\_confirm\_mail").val()) {  alert('確認メールアドレスが間違っています');  $("#input\_confirm\_mail").focus();  e.preventDefault();  }  }); | </end> |
| <hitle> | Textarea linebreak BR | <chare> | 2 | <pext> | <?php echo nl2br($field5); ?> | </end> |
| <hitle> | Button focus style none | <chare> | 2 | <pext> | .button:focus {  outline: none;  } | </end> |
| <hitle> | Button history back | <chare> | 2 | <pext> | <button onclick="history.back()">Go Back</button> | </end> |
| <hitle> | Japanese Address auto input | <chare> | 2 | <pext> | <?php  get\_header();  ?>  <script src="https://yubinbango.github.io/yubinbango/yubinbango.js" charset="UTF-8"></script>  <!-- <form class="h-adr">  <span class="p-country-name" style="display:none;">Japan</span>  郵便番号：<input type="text" class="p-postal-code" size="8" maxlength="8" value=”520-0514”><br>  都道府県：<input type="text" class="p-region" /><br>  地区町村：<input type="text" class="p-locality" /><br>  町名番地：<input type="text" class="p-street-address p-extended-address" />  </form> -->  1. script  <script src="https://yubinbango.github.io/yubinbango/yubinbango.js" charset="UTF-8"></script>  2. form tag  <form class="h-adr" action="" method="post" enctype="multipart/form-data">  <span class="p-country-name" style="display:none;">Japan</span>  <input type="hidden" class="p-region" />  <input type="hidden" class="p-locality p-street-address" />  <input type="hidden" class="p-extended-address" />  3. change auto input button  <a class="btn\_autoaddress" href="javascript:void(0)">住所自動入力</a>  4. 郵便番号  <input type="text" name="field5" class="p-postal-code" size="8" maxlength="8" placeholder="例）000-0000">  5. 都道府県  <input type="text" class="my\_region" name="field6" placeholder="例）東京都">  6. 市区町村・番地  <input type="text" class="my\_address" name="field7" placeholder="例）〇〇市〇区00-000">  7. 建物名・部屋番号  <input type="text" name="field8" class="my\_extended\_address" placeholder="">  8. custom.js  $('.btn\_autoaddress'). click(function() {  $('.my\_region').val($('.p-region').val());  $('.my\_address').val($('.p-locality').val());  $('.my\_extended\_address').val($('.p-extended-address').val())  }); | </end> |
| <hitle> | datepicker | <chare> | 2 | <pext> | 1. copy css files and images folder into css folder  Jquery-ui.css  Jquery-ui.structure.css  Jquery-ui.theme.css  [and images folder]  2. copy js files into js folder  Jquery-ui.js  Datepicer-ja.js  3. import css  <link href="css/jquery-ui.css" rel="stylesheet" type="text/css">  <link href="css/jquery-ui.structure.css" rel="stylesheet" type="text/css">  <link href="css/jquery-ui.theme.css" rel="stylesheet" type="text/css">  4. import js  <script type="text/javascript" src="js/jquery-ui.js"></script>  <script type="text/javascript" src="js/datepicker-ja.js"></script>  5. make datepicker input  <input type="text" class="datepicker" name="希望日" value="" placeholder="クリックして日付を入力"></input>  6. write js code into main/custom.js  $(function(){  $(".datepicker").datepicker();  }); | </end> |
| <hitle> | Hamburger menu not closing | <chare> | 1 | <pext> | /\*----- MOBILE\_MENU\_START -----\*/  $(".mobile-menu-hp").click(function() {  $(".menu-toggle-btn-hp").toggleClass("open");  $(".navigation-main-hp").slideToggle();  $("body").toggleClass("hide-scroll");  });  $(".navigation-main-hp a").click(function() {  $(".menu-toggle-btn-hp").toggleClass("open");  $(".navigation-main-hp").slideToggle();  $("body").toggleClass("hide-scroll");  });  /\*----- MOBILE\_MENU\_END -----\*/ | </end> |
| <hitle> | Js\_button | <chare> | 1 | <pext> | $('.js\_button').click(function() {  if($(this).attr('data-href')) {  if($(this).attr('data-target') == "\_blank") {  window.open($(this).attr('data-href'), '\_blank');  } else {  location.href = $(this).attr('data-href');  }  }  });  .js\_button {  cursor: pointer;  transition: all .3s;  }  .js\_button:hover {  opacity: .7;  } | </end> |
| <hitle> | Default Post | <chare> | 1 | <pext> | Wordpress default post | </end> |
| <hitle> | Common Blog/Sidebar | <chare> | 2 | <pext> | ------In the index/archieve/category page-----  global $wp\_query;  $query = $wp\_query;  $category\_id = 0;  if(is\_category() || is\_single()){  $category\_id = get\_query\_var('cat');  }  get\_template\_part('template-parts/common-blog');  --------template-parts/common-blog.php----------  global $query;  global $category\_id;  $post\_ids = [];  if ( $query->have\_posts() ) :  while ( $query->have\_posts() ) :  $query->the\_post();  $loop\_id = get\_the\_id();  array\_push($post\_ids, $loop\_id);  endwhile;  endif; | </end> |
| <hitle> | Post prev/next navigation | <chare> | 2 | <pext> | the\_post\_navigation(  array(  'prev\_text' => '<span class="nav-subtitle">' . esc\_html\_\_( 'Previous:', 'hayapen' ) . '</span> <span class="nav-title">%title</span>',  'next\_text' => '<span class="nav-subtitle">' . esc\_html\_\_( 'Next:', 'hayapen' ) . '</span> <span class="nav-title">%title</span>',  )  ); | </end> |
| <hitle> | Category/tags in post | <chare> | 2 | <pext> | $loop\_category\_objects = get\_the\_category($loop\_id);  $loop\_category\_name = "";  foreach($loop\_category\_objects as $cd){  $loop\_category\_name = $cd->cat\_name;  break;  }  $loop\_tags = [];  $loop\_tag\_objects = get\_the\_tags();  if ($loop\_tag\_objects) {  foreach($loop\_tag\_objects as $tag) {  array\_push($loop\_tags, array('name' => $tag->name, 'link' => get\_tag\_link($tag->term\_id)));  }  } | </end> |
| <hitle> | Custom Post Type | <chare> | 1 | <pext> | -----Custom Category-----------  CPT UI/Texanomy/  階層 또는 hierarchy true로 설정하여야 기존카테고리처럼 사용할수 있다.  -----Custom Tag-----  CPT UI/Texanomy  階層 또는 hierarchy false로 설정하여야 기존태그처럼 사용할수 있다. | </end> |
| <hitle> | Category/tags in current-custom-post | <chare> | 2 | <pext> | $category\_id = get\_queried\_object()->term\_id; | </end> |
| <hitle> | All custom category/tags | <chare> | 2 | <pext> | $terms = get\_terms([  'taxonomy' => ‘custom\_category’,  'hide\_empty' => false,  ]);  foreach ($terms as $term){  echo $term->slug." : ";  echo $term->name;  echo $get\_term\_link($term);  echo $term->term\_id;  } | </end> |
| <hitle> | Text Only from the\_content() | <chare> | 2 | <pext> | echo wp\_strip\_all\_tags( get\_the\_content() ); | </end> |
| <hitle> | Image rotate 360 infinite | <chare> | 1 | <pext> | img.rot360 {  transform: rotate(0deg);  animation: anim\_rotate360 10s linear infinite;  }  @keyframes anim\_rotate360 {  0% {  transform: rotate(0deg);  }  100% {  transform: rotate(360deg);  }  } | </end> |
| <hitle> | Phone call, mailto | <chare> | 1 | <pext> | <a href="tel:123-456-7890">123-456-7890</a>  <a href="mailto:email@example.com, secondemail@example.com">Send Email</a> | </end> |
| <hitle> | Contact Form 7 | <chare> | 1 | <pext> | <?php echo do\_shortcode("[your\_shortcode]"); ?> | </end> |
| <hitle> | Anchor point | <chare> | 1 | <pext> | <div class="id\_line" id="greeting"></div>  .id\_line {  width: 100%;  height: 0;  transform: translateY(-100px);  }  @media (max-width:767px) {  .id\_line {  transform: translateY(-60px);  height: 0;  }  } | </end> |
| <hitle> | Study Sites | <chare> | 1 | <pext> | <https://igie.jp/service/>  scss, normalize/reset.scss | </end> |
| <hitle> | Magic scrolling effect | <chare> | 1 | <pext> | /js/ScrollMagic.min.js  /js/TweenMax.min.js  let controller = new ScrollMagic.Controller();  let target = document.querySelectorAll('.js\_tgt');  for (var i = 0; i < target.length; i++) {  let classToggle = new ScrollMagic.Scene({  triggerElement: target[i],  //reverse: false  }).triggerHook('0.85').setClassToggle(target[i], 'on').addTo(controller);  } | </end> |
| <hitle> | OwlCarousel | <chare> | 1 | <pext> | Import, Owl.theme.min.css, owl.theme.default.css, owl.theme.default  <link href="<?php echo get\_stylesheet\_directory\_uri(); ?>/css/owl.carousel.css" rel="stylesheet">  <link href="<?php echo get\_stylesheet\_directory\_uri(); ?>/css/owl.theme.default.css" rel="stylesheet">  <script src="<?php echo get\_stylesheet\_directory\_uri(); ?>/js/owl.carousel.min.js"></script>  ---------set items wrap div with owl classes-------  <div class="works\_top\_hp owl-carousel owl-theme" id="works\_top\_hp">  ---------use owl js--------  if($('#works\_top\_hp')) {  $(".works\_top\_hp").owlCarousel({  dots: true,  dotsEach: true,  nav: false,  margin: 20,  smartSpeed: 450,  loop: true,  autoplay: true,  responsive: {  0: {  items: 1,  },  768: {  items: 2,  },  992: {  items: 3,  },  1401: {  items: 4,  },  }  });  } | </end> |
| <hitle> | Disable Empty href | <chare> | 1 | <pext> | $('a[href="#"]').click(function(e){  e.preventDefault();  console.log("hello");  }) | </end> |
| <hitle> | Japanese Fonts | <chare> | 1 | <pext> | @import url('https://fonts.googleapis.com/css2?family=Noto+Sans+JP:wght@100;300;400;500;700;900&display=swap');  @import url('https://fonts.googleapis.com/css2?family=Poppins:wght@100;200;300;400;500;600;700;800;900&display=swap');  @import url('https://fonts.googleapis.com/css2?family=Montserrat:wght@100;200;300;400;500;600;700;800;900&display=swap');  @import url('https://fonts.googleapis.com/css2?family=Josefin%20Sans:wght@100;200;300;400;500;600;700;800;900&display=swap');  font-family: 'Noto Sans JP', "ヒラギノ角ゴ ProN", Hiragino Kaku Gothic ProN, "メイリオ", Meiryo, "游ゴシック Medium", YuGothic, YuGothicM, sans-serif;  .font-mon {  font-family: 'Montserrat', sans-serif;  }  .font-pop {  font-family: 'Poppins', sans-serif;  }  .font-jose {  font-family: 'Josefin Sans', sans-serif;  } | </end> |
| <hitle> | animation SpecialColImage | <chare> | 1 | <pext> | <script src="js/jquery-3.5.1.slim.min.js" type="text/javascript"></script>  <script src="js/popper.min.js" type="text/javascript"></script>  <script src="js/bootstrap.min.js" type="text/javascript"></script>  <script src="js/jquery.min.js"></script>  <script src="js/ScrollMagic.min.js"></script>  <script src="js/TweenMax.min.js"></script>  <script src="js/custom.js"></script>  /\*-- Magic animation start --\*/  $(function(){  /\*  \* Scroll Function  \*/    // 全体制御コントローラ  var controller = new ScrollMagic.Controller();    // トリガー設定  const hook = '0.8';    // 速度設定  var duration = 0.6;      var fadeUpText = new TimelineMax();  fadeUpText.fromTo('.tw-ttl', duration, { opacity: 0, y: '60px', x:'0'}, {opacity: 1, y: '0', x: '0', ease: Cubic.easeOut})  .fromTo('.tw-txt', duration, { opacity: 0, y: '60px', x:'0'}, {opacity: 1, y: '0', x: '0', ease: Cubic.easeOut}, 0.1);      var showImg;      });    let controller = new ScrollMagic.Controller();  let target = document.querySelectorAll('.js\_tgt');  for (var i = 0; i < target.length; i++) {  let classToggle = new ScrollMagic.Scene({  triggerElement: target[i],  //reverse: false  }).triggerHook('0.85').setClassToggle(target[i], 'on').addTo(controller);  }    /\*-- Magic animation end --\*/  <div class="about\_right">  <div class="special\_col js\_tgt">  <div class="special\_img\_wrap">  <div class="special\_img\_div">  <img src="images/top\_about.jpg" alt="">  <div class="special\_img\_cover"></div>  </div>  <div class="special\_img\_back"></div>  </div>  </div>  </div>  .about\_right { width: 45%; position: relative; }  .special\_col { width: 100%; position: relative; }  .special\_img\_wrap { transition: transform .35s ease-in; transform: translateX(100%); position: relative; }  .special\_col.js\_tgt.on .special\_img\_wrap { transform: translateX(0); }  .special\_img\_div { position: relative; overflow: hidden; }  .special\_img\_div img { position: relative; z-index: 1; width: 100%; height: auto; }  .special\_img\_back { content: " "; position: absolute; width: 100%; height: 100%; top: 40px; right: 40px; z-index: 0; opacity: .8; background: #F3F2EF; }  .special\_img\_cover { width: 150%; height: 100%; background: #ccc; position: absolute; top: 0; left: -25%; z-index: 2;  transition: transform .65s ease-out; transition-delay: .35s; transform: translateX(0) skewX(-35deg);  }  .special\_col.js\_tgt.on .special\_img\_cover { transform: translateX(100%) skewX(-35deg); } | </end> |
| <hitle> | panil1 left | <chare> | 2 | <pext> | \*\*\*parent should have { position: relative;}\*\*\*  .panil1\_set { width: 100%; position: relative; }  .panil1\_drive { transition: transform .35s ease-in; transform: translateX(-100%); position: relative; }  .panil1\_set.js\_tgt.on .panil1\_drive { transform: translateX(0); }  .panil1\_imgdiv { position: relative; overflow: hidden; }  .panil1\_imgdiv img { position: relative; z-index: 1; width: 100%; height: auto; }  .panil1\_shadow { content: " "; position: absolute; width: 100%; height: 100%; top: 40px; left: 40px; z-index: 0; opacity: .8; background: #F3F2EF; }  .panil1\_cover { width: 150%; height: 100%; background: #ccc; position: absolute; top: 0; left: -25%; z-index: 2;  transition: transform .65s ease-out; transition-delay: .35s; transform: translateX(0) skewX(35deg);  }  .panil1\_set.js\_tgt.on .panil1\_cover { transform: translateX(-100%) skewX(35deg); }  \*\*\*html\*\*\*  <div class="parent">  <div class="panil2\_set js\_tgt">  <div class="panil2\_drive">  <div class="panil2\_imgdiv">  <img src=””>  <div class="panil2\_cover"></div>  </div>  <div class="panil2\_shadow"></div>  </div>  </div>  </div> | </end> |
| <hitle> | panir1 right | <chare> | 2 | <pext> | \*\*\*parent should have { position: relative;}\*\*\*  .panir1\_set { width: 100%; position: relative; }  .panir1\_drive { transition: transform .35s ease-in; transform: translateX(100%); position: relative; }  .panir1\_set.js\_tgt.on .panir1\_drive { transform: translateX(0); }  .panir1\_imgdiv { position: relative; overflow: hidden; }  .panir1\_imgdiv img { position: relative; z-index: 1; width: 100%; height: auto; }  .panir1\_shadow { content: " "; position: absolute; width: 100%; height: 100%; top: 40px; right: 40px; z-index: 0; opacity: .8; background: #F3F2EF; }  .panir1\_cover { width: 150%; height: 100%; background: #ccc; position: absolute; top: 0; left: -25%; z-index: 2;  transition: transform .65s ease-out; transition-delay: .35s; transform: translateX(0) skewX(-35deg);  }  .panir1\_set.js\_tgt.on .panir1\_cover { transform: translateX(100%) skewX(-35deg); }  \*\*\*html\*\*\*  <div class="parent">  <div class="panir2\_set js\_tgt">  <div class="panir2\_drive">  <div class="panir2\_imgdiv">  <img src=””>  <div class="panir2\_cover"></div>  </div>  <div class="panir2\_shadow"></div>  </div>  </div>  </div> | </end> |
| <hitle> | panil2 left | <chare> | 2 | <pext> | \*\*\*parent should have { position: relative;}\*\*\*  .panil2\_set { height:calc(100%); width: 100%; position: relative; }  .panil2\_drive { width: calc(100%); height: calc(100%); transition: transform .35s ease-in; transform: translateX(-100%); position: relative; }  .panil2\_set.js\_tgt.on .panil2\_drive { transform: translateX(0); }  .panil2\_imgdiv { width: calc(100%); height: calc(100%); position: relative; overflow: hidden; }  .panil2\_face {width: 100%; height: calc(100%); background-color: red; position: relative; z-index: 1;}  .panil2\_shadow { content: " "; position: absolute; width: 100%; height: 100%; top: 40px; left: 40px; z-index: 0; opacity: .8; background: #F3F2EF; }  .panil2\_cover { width: 150%; height: 100%; background: #ccc; position: absolute; top: 0; left: -25%; z-index: 2;  transition: transform .65s ease-out; transition-delay: .35s; transform: translateX(0) skewX(35deg);  }  .panil2\_set.js\_tgt.on .panil2\_cover { transform: translateX(-100%) skewX(35deg); }  \*\*\*html\*\*\*  <div class="parent">  <div class="panil2\_set js\_tgt">  <div class="panil2\_drive">  <div class="panil2\_imgdiv">  <div class="panil2\_face"></div>  <div class="panil2\_cover"></div>  </div>  <div class="panil2\_shadow"></div>  </div>  </div>  </div> | </end> |
| <hitle> | panir2 right | <chare> | 2 | <pext> | \*\*\*parent should have { position: relative;}\*\*\*  .panir2\_set { height:calc(100%); width: 100%; position: relative; }  .panir2\_drive { width: calc(100%); height: calc(100%); transition: transform .35s ease-in; transform: translateX(100%); position: relative; }  .panir2\_set.js\_tgt.on .panir2\_drive { transform: translateX(0); }  .panir2\_imgdiv { width: calc(100%); height: calc(100%); position: relative; overflow: hidden; }  .panir2\_face {width: 100%; height: calc(100%); background-color: red; position: relative; z-index: 1;}  .panir2\_shadow { content: " "; position: absolute; width: 100%; height: 100%; top: 40px; right: 40px; z-index: 0; opacity: .8; background: #F3F2EF; }  .panir2\_cover { width: 150%; height: 100%; background: #ccc; position: absolute; top: 0; left: -25%; z-index: 2;  transition: transform .65s ease-out; transition-delay: .35s; transform: translateX(0) skewX(-35deg);  }  .panir2\_set.js\_tgt.on .panir2\_cover { transform: translateX(100%) skewX(-35deg); }  \*\*\*html\*\*\*  <div class="parent">  <div class="panir2\_set js\_tgt">  <div class="panir2\_drive">  <div class="panir2\_imgdiv">  <div class="panir2\_face"></div>  <div class="panir2\_cover"></div>  </div>  <div class="panir2\_shadow"></div>  </div>  </div>  </div> | </end> |
| <hitle> | Panil3 left clip-path | <chare> | 2 | <pext> | .panil3 {  display: inline-block;  clip-path: polygon(0% 0%, 0% 0%, 0% 100%, 0 100%);  transition: all 1s cubic-bezier(0.5, 1, 0.89, 1);  vertical-align: middle;  }  .panil3.on {  clip-path: polygon(0% 0%, 100% 0%, 100% 100%, 0 100%);  } | </end> |
| <hitle> | Animation typing effect | <chare> | 1 | <pext> | 1. copy theatherJS.js  2. include theatherJS  <script src="js/theaterJS.js"></script>  3. css  .parent {  Position: relative;  }  .parent .target\_text {  visibility: hidden;  font-size: 50px;  line-height: 1.8;  }  .com\_actor\_\_content {  font-size: 50px;  font-weight: bold;  line-height: 1.8;  position: absolute;  top: 0;  left: 0;  }  @-webkit-keyframes blink {  from {  opacity: 0;  }  to {  opacity: 1;  }  }  @keyframes blink {  from {  opacity: 0;  }  to {  opacity: 1;  }  }  .com\_actor\_\_content--typing::after {  content: "|";  -webkit-animation: blink 500ms infinite;  animation: blink 500ms infinite;  }  4. html  <div class="banner1">  <div id="elactor" class="com\_actor\_\_content"></div>  <h1>With Human Power!</h1>  </div>  5. javascript  /\*--TheaterJS typing effect --\*/  $(function(){  $(window).load(function () {  setTimeout(function () {  var theater = theaterJS()    theater  .on('type:start, erase:start', function () {  theater.getCurrentActor().$element.classList.add('com\_actor\_\_content--typing')  })  .on('type:end, erase:end', function () {  theater.getCurrentActor().$element.classList.remove('com\_actor\_\_content--typing')  })    theater  .addActor('elactor', { speed: 0.8, accuracy: 1 })  .addScene('elactor:With Power? ', -16, 'Human Power')  }, 1000);  });  });  /\*--TheaterJS typing effect end--\*/  6. referenced code  var theater = theaterJS()  theater  .on('type:start, erase:start', function () {  theater.getCurrentActor().$element.classList.add('actor\_\_content--typing')  })  .on('type:end, erase:end', function () {  theater.getCurrentActor().$element.classList.remove('actor\_\_content--typing')  })  .on('type:start, erase:start', function () {  if (theater.getCurrentActor().name === 'vader') {  document.body.classList.add('dark')  } else {  document.body.classList.remove('dark')  }  })  theater  .addActor('vader', { speed: 0.8, accuracy: 0.6 })  .addActor('luke')  .addScene('vader:Luke.', 600)  .addScene('luke:What?', 400)  .addScene('vader:I am your father.', 400)  .addScene('luke:Nooo...', -3, '!!! ', 600, 'No! ', 600)  .addScene('luke:That\'s not true!', 600)  .addScene('luke:That\'s impossible!', 400)  .addScene('vader:Search your feelings.', 1600)  .addScene('vader:You know it to be true.', 1000)  .addScene('luke:Noooooooo! ', 600, 'No!', 400)  .addScene('vader:Luke.', 600)  .addScene('vader:You can destroy the Emperor.', 1600)  .addScene('vader:He has foreseen this. ', 800)  .addScene('vader:It is your destiny.', 1600)  .addScene('vader:Join me.', 800)  .addScene('vader:Together we can rule the galaxy.', 800)  .addScene('vader:As father and son.', 1600)  .addScene('vader:Come with me. ', 800)  .addScene('vader:It is the only way.', 2000)  .addScene(theater.replay.bind(theater)) | </end> |
| <hitle> | Windows10Update | <chare> | 1 | <pext> | 1. run services.msc  2. double-click Windows Update  3. Manual/Disable | </end> |
| <hitle> | VSCode Fomatters | <chare> | 1 | <pext> | auto close tag  auto rename tag  html format  JS-CSS-HTML Formatter | </end> |
| <hitle> | Option page by ACF | <chare> | 1 | <pext> | 1. create ACF field group  2. Assigin to defined option page.  //function.php  add\_action('acf/init', function () {  if( function\_exists('acf\_add\_options\_page') ) {  acf\_add\_options\_page([  'page\_title' => 'よくある質問',  'menu\_title' => 'よくある質問',  'menu\_slug' => 'faq-settings',  'capability' => 'manage\_options',  'position' => 30,  'icon\_url' => 'dashicons-admin-generic',  ]);  }  }); | </end> |
| <hitle> | UpdatingRecords | <chare> | 1 | <pext> | $('a[href="#"]').click(function(e){  e.preventDefault();  console.log("hello");  }) | </end> |
| <hitle> | UpdatingRecords | <chare> | 1 | <pext> | $('a[href="#"]').click(function(e){  e.preventDefault();  console.log("hello");  }) | </end> |
| <hitle> | UpdatingRecords | <chare> | 1 | <pext> | $('a[href="#"]').click(function(e){  e.preventDefault();  console.log("hello");  }) | </end> |
| <hitle> | UpdatingRecords | <chare> | 1 | <pext> | $('a[href="#"]').click(function(e){  e.preventDefault();  console.log("hello");  }) | </end> |
| <hitle> | UpdatingRecords | <chare> | 1 | <pext> | $('a[href="#"]').click(function(e){  e.preventDefault();  console.log("hello");  }) | </end> |
| <hitle> | UpdatingRecords | <chare> | 1 | <pext> | $('a[href="#"]').click(function(e){  e.preventDefault();  console.log("hello");  }) | </end> |
| <hitle> | UpdatingRecords | <chare> | 1 | <pext> | $('a[href="#"]').click(function(e){  e.preventDefault();  console.log("hello");  }) | </end> |
| <hitle> | UpdatingRecords | <chare> | 1 | <pext> | $('a[href="#"]').click(function(e){  e.preventDefault();  console.log("hello");  }) | </end> |