



[\$tag]

[\$tag]

```
[wporg title="WordPress.org"]
Having fun with WordPress.org shortcodes.
[/wporg]
```

3

- \$atts - [\$tag]
- \$content - " WordPress.org "
- \$tag - string - [\$tag]

```
function wporg_shortcode( $atts = array(), $content = null, $tag = '' ) {}
```



-
- [array_change_key_case\(\)](#)
- [shortcode_atts\(\)](#) \$atts
-



[wporg] CSS

```

/**
 * The [wporg] shortcode.
 *
 * Accepts a title and will display a box.
 *
 * @param array $atts Shortcode attributes. Default empty.
 * @param string $content Shortcode content. Default null.
 * @param string $tag Shortcode tag (name). Default empty.
 * @return string Shortcode output.
 */
function wporg_shortcode( $atts = [], $content = null, $tag = '' ) {
    // normalize attribute keys, lowercase
    $atts = array_change_key_case( (array) $atts, CASE_LOWER );

    // override default attributes with user attributes
    $wporg_atts = shortcode_atts(
        array(
            'title' => 'WordPress.org',
        ), $atts, $tag
    );

    // start box
    $o = '<div class="wporg-box">';

    // title
    $o .= '<h2>' . esc_html( $wporg_atts['title'] ) . '</h2>';

    // enclosing tags
    if ( ! is_null( $content ) ) {
        // $content here holds everything in between the opening and the closing tags of your shortcode. eg.g [my-shortcode]content[/my-shortcode].
        // Depending on what your shortcode supports, you will parse and append the content to your output in different ways.
        // In this example, we just secure output by executing the_content filter hook on $content.
        $o .= apply_filters( 'the_content', $content );
    }

    // end box
    $o .= '</div>';

```

```
// return output
return $o;
}

/**
 * Central location to create all shortcodes.
 */
function wporg_shortcodes_init() {
    add_shortcode( 'wporg', 'wporg_shortcode' );
}

add_action( 'init', 'wporg_shortcodes_init' );
```

#1

Vcanson ## 25 2023 09:57:34

Vcanson ## 26 2023 15:34:10