

Color



- [Config methods](#)
 - [set_palette\(\\$palette = array\(\) \)](#)
 - [set_alpha_enabled\(\\$enabled = true \)](#)
-

Renders a color picker with an optional palette.

Colors are represented with six hexadecimal digits prefixed with `#` (e.g. white is `#FFFFFF`) or with 8 hexadecimal digits when alpha is enabled (e.g. white is `#FFFFFFF`).

```
Field::make( 'color', 'crb_box_background', __( 'Background Color' ) )
```

Config methods

`set_palette( $palette = array() )`

Sets the color picker's palette of predefined colors. Must be an array of hexadecimal colors.

```
Field::make( 'color', 'crb_background', 'Background' )  
->set_palette( array( '#FF0000', '#00FF00', '#0000FF' ) );
```

`set_alpha_enabled( $enabled = true )`

Enables alpha selection for the field. Changes the stored value from 6 hexadecimal digits to 8 to account for the extra alpha information.

```
Field::make( 'color', 'crb_background', 'Background' )  
->set_alpha_enabled( true );
```

!> If you need separate RGB(A) values instead of a color's hexadecimal value you can use the built-in `carbon_hex_to_rgba( $hex )` function:

```
$hex = carbon_get_theme_option( 'foobar' ); // Assuming this returns #FF0000BF
$rgba = carbon_hex_to_rgba( $hex );
print_r( $rgba );
```

Output:

```
array(4) {
    ["red"] => int(255), // integer in the 0-255 range
    ["green"] => int(0), // integer in the 0-255 range
    ["blue"] => int(0), // integer in the 0-255 range
    ["alpha"] => float(0.74901960784314), // float in the 0-1 range
}
```

📄 #2

📄 📄 📄 2026-08-29 16:13:57 UTC

📄 📄 📄 2026-08-29 16:13:57 UTC