

# ADVANCED THEMER

Documentation  
Written  
by Eljay Bos

AT Version 1.0.9

Inspired by the tutorials and reviews of

[Nelmedia](#)

[WPTuts](#)

[WPEasy and WGet](#)

[Advanced Themer](#)



<b>Introduction</b>	<b>7</b>
<b>1. After installation</b>	<b>8</b>
<b>2. AT - Color Palettes</b>	<b>9</b>
2.1. Color Palette Settings	11
2.1.1. Option 1 Add prefix	11
2.1.2. Option 2 Enable Shades	11
2.1.3. Option 3 Enable Dark Mode	13
<b>3. AT - Theme Settings</b>	<b>14</b>
3.1. Builder Elements	14
3.2. Permissions	15
3.3. Gutenberg Settings	15
3.3.1. Replace Gutenberg Color Palettes.	15
3.3.2. Remove Default Gutenberg Presets	15
3.4. Miscellaneous	16
3.4.1. Disable the "ACF" menu item in your Dashboard	16
3.4.2. Remove all data when uninstalling the plugin	17
3.5. Import/Export	17
3.5.1. Export Theme Settings	17
3.5.2. Import Theme Settings	18
3.6. Reset Settings	18
3.6.1. Reset Theme Settings	19
<b>4. Theme Settings - General - Global</b>	<b>20</b>
4.1. CSS Variables	20
4.1.1. Theme Settings: Enable CSS Variables Features	20
4.1.2. Add a prefix to your global CSS variables	22
4.1.3. Base Font Size	23
4.1.4. Minimum Viewport Width	23
4.1.5. Maximum Viewport Width	23
4.1.6. Clamp Unit	23

4.2. Theme Settings: Typography	23
4.3. Theme Settings: Spacing	24
4.4. Theme Settings: Borders	26
4.5. Theme Settings: Border Radius	26
4.6. Theme Settings: Box-Shadow	27
4.7. Theme Settings: Width	28
4.8. Theme Settings: Custom Variables	28
4.9. Import your own CSS Variable Framework	29
<b>5. Classes &amp; Styles</b>	<b>31</b>
5.1. Classes & Styles: Grids	31
5.2. Classes & Styles: Class Importer	32
5.3. Classes & Styles: Advanced CSS editor	32
5.3.1. Panels	33
<b>6. Builder Tweaks</b>	<b>35</b>
6.1. Builder Tweaks: General	35
6.2. Topbar	35
6.2.1. Enable Global Features	35
6.2.2. Grid Columns	36
6.2.3. Grid Gap	37
6.3. Structure Panel Icons	37
6.3.1. Default Tag View	38
6.3.2. Contextual Menu	38
6.3.3. General Tweaks	39
6.4. Builder Tweaks for Classes & Styles	41
6.5. Builder Tweaks for the Elements	43
6.5.1. Builder Tweaks for the Elements Details	43
6.5.2. Tabs Shortcuts	44
6.5.3. Pseudo Elements Shortcuts	45
6.5.4. Type of Dummy content	45

6.5.5. Default Elements List Columns	45
6.6. Elements Shortcuts	46
6.7. Keyboard Shortcuts	48
<b>7. Strict Editor View</b>	<b>49</b>
7.1. Strict Editor View: White Label	49
7.2. Strict Editor View: Toolbar	50
7.2.1. Disable Toolbar Icons	50
7.3. Strict Editor View: Elements	50
7.3.1. Enable the following elements on Strict Editor View	50
7.4. Strict Editor View: Miscellaneous	51
7.4.1. Tweaks	51
<b>8. AI Integration</b>	<b>52</b>
<b>9. Frontend Playground</b>	<b>53</b>
9.1. Frontend Playground Theme	54
9.2. Custom Post Type Permissions	54
<b>10. Extra's</b>	<b>55</b>
10.1. Theme Settings: Resources	55
10.2. BricksLabs Panel	56
<b>11. Builder Tweaks: Classes &amp; Styles</b>	<b>57</b>
<b>12. The Enhanced Top Bar(s) Explained</b>	<b>62</b>
12.1. Enhanced Left Top Bar	62
12.2. Element Top Bar	62
12.3. Child Element Top Bar	63
<b>13. The Enhanced Right Top Bar(s)</b>	<b>63</b>
13.1. The Enhanced Context Menu	65
<b>14. Miscellaneous</b>	<b>68</b>
14.1. Breakpoint indicators	68
14.2. Indicators of styles inherited from a class	69

	Advanced Themer
14.3. Locked Class Indicator	69
14.4. Clone Class and Copy Class to Clipboard	70
14.5. Elements that share the same Class are highlighted	71

# Introduction

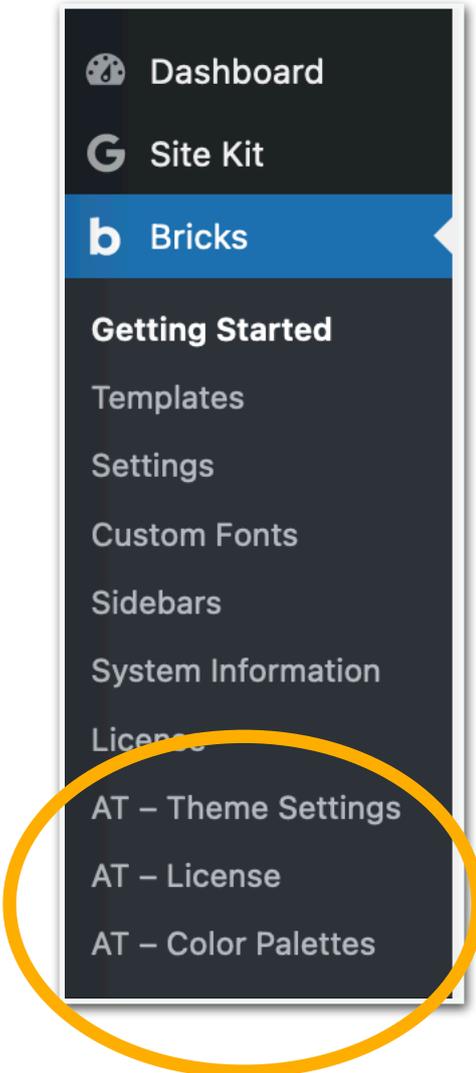
---

To use construction terms, If Bricks Builder is the foundation, then Advanced Themer is the elevation and expansion.

For many professional website builders, who already wholeheartedly accepted Bricks Builder as their professional page builder of choice, the introduction of Advanced Themer is like a Swiss army knife full of unexpected and much needed tweaks and tools.

It will help streamline the website-building process with the unlimited possibilities of colors, and it will, even more, further professionalize the skill level with the use of variables, clamp function, and fluid design.

# 1. After installation



After the plugin is installed, Advanced Themes (AT) can be found under Bricks in the Admin Sidebar, with three different settings. See the orange ellipse in the screenshot left.

## 1.1. AT - License

Let's walk through the different settings, starting with AT - License. This is a one-time event. You'll find the license number in your AT - account, fill it in, and you never have to look at it again (i.e. when you have a LTD account).

Advanced Themer License

Enter your license key. You can manage your license, update payment method and view your invoices right from [your account](#).

Status: **active** (never expires)

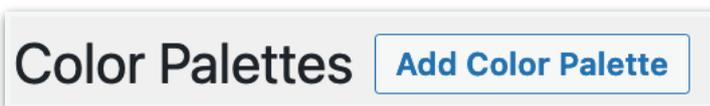
## 2. AT - Color Palettes

---

The second setting is AT - Color Palettes. The first time you enter the below screenshot is what you'll see.

<input type="checkbox"/> Title	Colors	Shades	Darkmode	JSON	Prefix	Date
No Color Palettes found						
<input type="checkbox"/> Title	Colors	Shades	Darkmode	JSON	Prefix	Date

So, you'll have to click at the top of the page:



Which will bring you to the next page "Add new Color Palette":

 A screenshot of the 'Add new Color Palette' form. At the top, there's a text input field labeled 'Add title'. Below it are two tabs: 'Colors' (selected) and 'Settings'. Under the 'Colors' tab, there's a section titled 'Colors' with a sub-instruction: 'Add the colors to your palette here. Choose a unique name for each label in order to avoid CSS conflicts, or make sure to set a prefix value in the settings tab.' There's a list of color entries, with the first one labeled '1'. Each entry has a 'Label' input field and a 'Color' input field with a 'Select Color' button. At the bottom right of the form is a blue button labeled 'Add a New Color'. On the right side of the form, there's a sidebar with buttons for 'Save Draft', 'Status: Draft Edit', 'Visibility: Public Edit', 'Publish immediately Edit', and a 'Publish' button.

Setting the colors of your choice

Give your new palette a title, and label your color (e.g. primary, secondary, or something different), then choose the colors you want to use, and when you're satisfied click Publish.

From then on this palette will be available in Bricks, and you can even make it the default palette. See the examples below.

New color palette in Bricks

**Colors** Settings

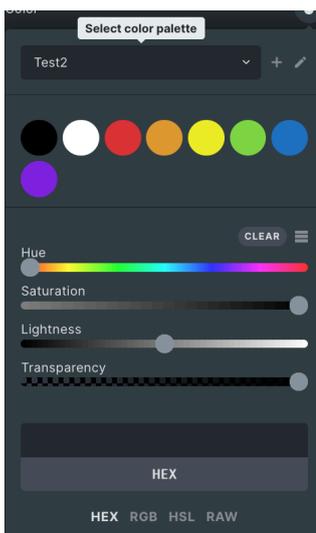
**Colors \***

Add the colors to your palette here. Choose a unique name for each label in order to avoid CSS conflicts, or make sure to set a prefix value in the settings tab.

1	Label * Primary	Color * Select Color
2	Label * Secondary	Color * Select Color
3	Label * Text	Color * Select Color
4	Label * Links	Color * Select Color

Add a New Color

The new basic color palette is immediately available within Bricks (see screenshot). Also, notice that you can choose the palette of your choice.



You may have noticed that there is another tab named Settings. Here are some interesting features to be found.

## 2.1. Color Palette Settings

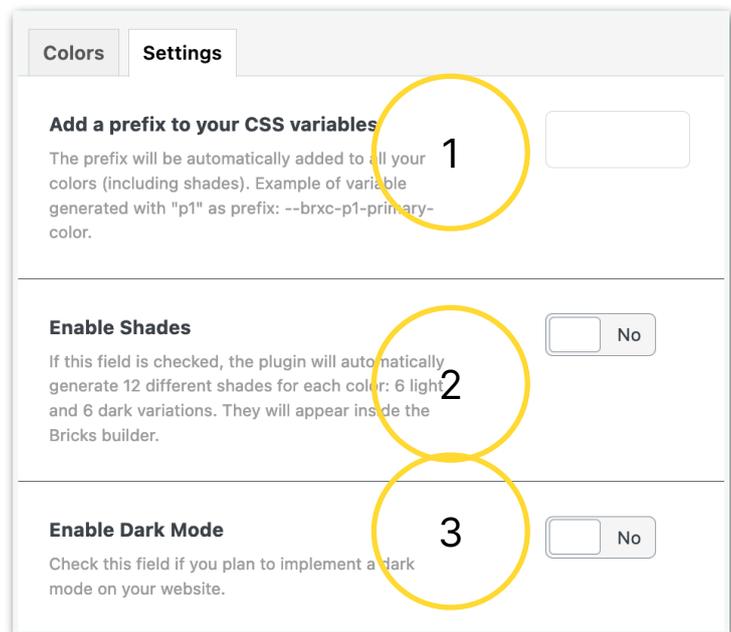
In the Color Palette Settings are three different options.

### 2.1.1. Option 1 Add prefix

With the first option, you can add a prefix to your CSS variables.

#### **Why is that relevant?**

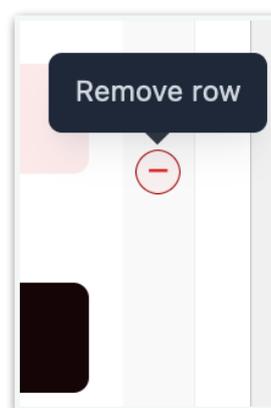
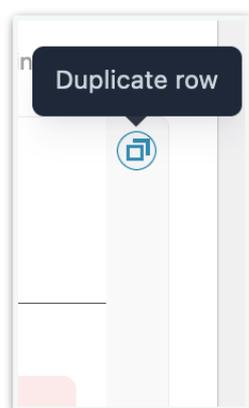
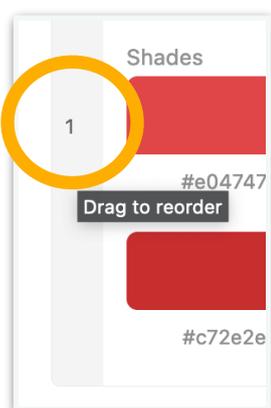
Well, many Bricks users also work with frameworks like ACSS, Winden, or others. The naming convention could be overlapping and confusing, and by putting your own prefix (e.g. AT) before your variables you will always recognize the ones you made with Advanced Themer.



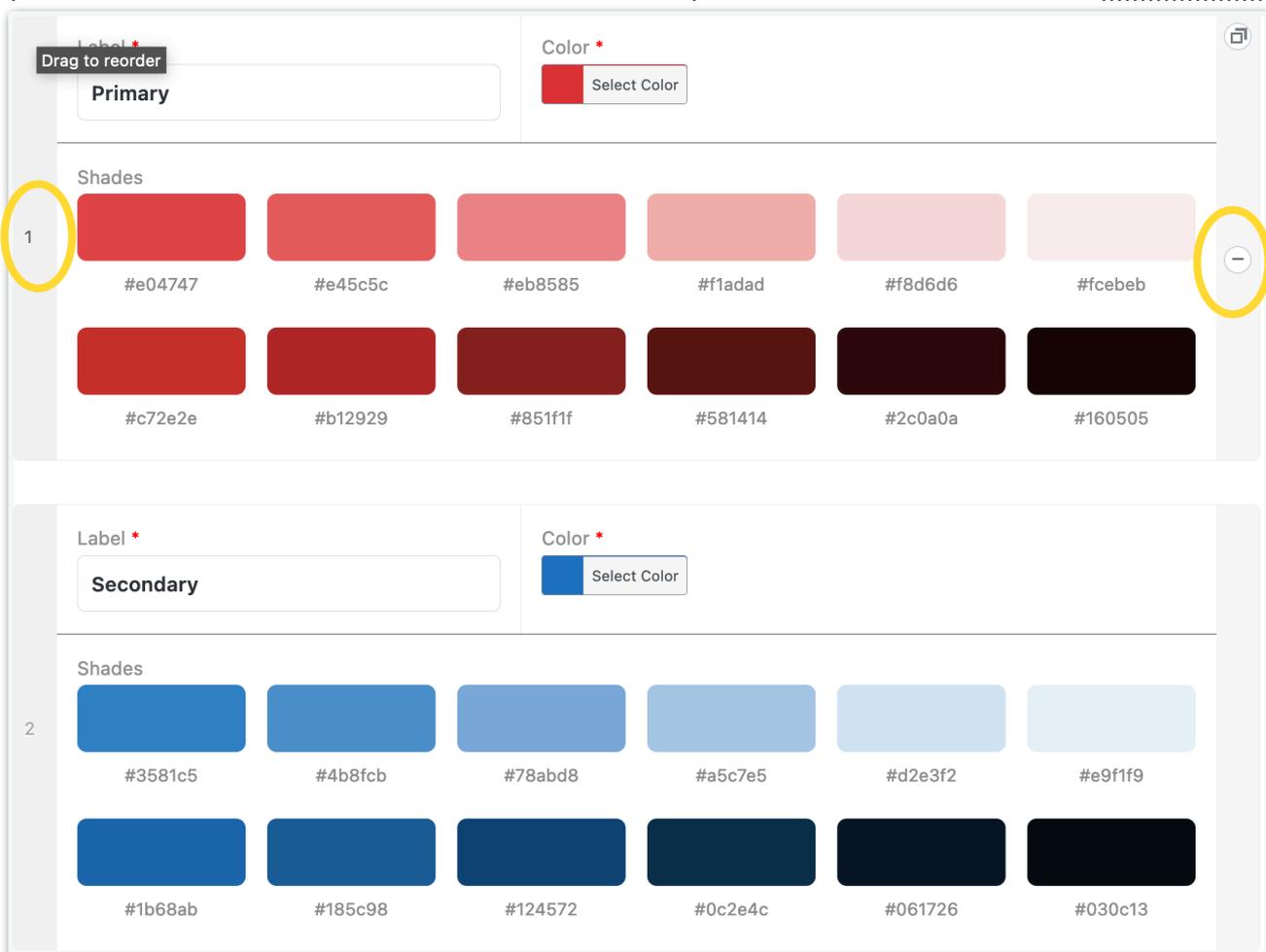
### 2.1.2. Option 2 Enable Shades

When this is enabled AT will produce 12 different shades for each color of your palette. Six light and six dark ones. See the screenshot on the next page. These shades will also appear in the Bricks page builder.

You'll notice the number in front of the shades (see orange circle in screenshot below). When the mouse pointer is hovering over this it will change to indicate

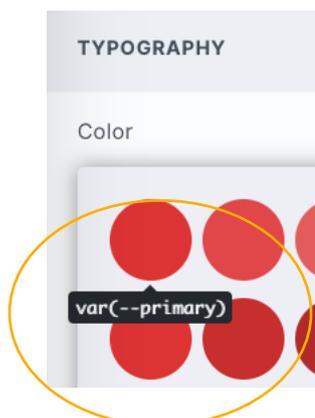


the possibility of movement. Every row of shades can be reordered to your liking.

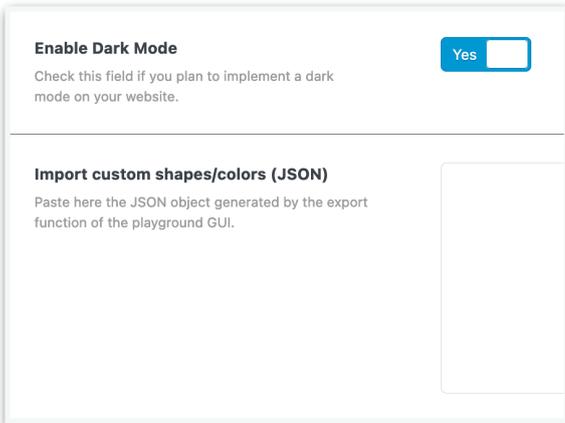


On the right-hand side of each row, you may notice two icons that appear when hovering over the row of shades, it's a '+' icon to duplicate a row, and a '-' sign to remove a row.

In the Bricks workspace, we can now find the shades as well. These colors are based on variables, which means that if changes are made to this color it will happen to all instances where this color has been used.



### 2.1.3. Option 3 Enable Dark Mode



The screenshot shows a settings panel with two sections. The top section is titled 'Enable Dark Mode' and contains a checkbox labeled 'Yes' which is currently checked. Below the checkbox is the text: 'Check this field if you plan to implement a dark mode on your website.' The bottom section is titled 'Import custom shapes/colors (JSON)' and contains the text: 'Paste here the JSON object generated by the export function of the playground GUI.' To the right of this text is a large, empty rectangular text area for pasting JSON data.

When Dark Mode is enabled each color palette will present the colors for Dark Mode as well.

It is also possible to import custom shapes/ colors (in JSON format).

## 3. AT - Theme Settings

When you open the Theme settings the very first time, as of version 1.08 you'll see the user interface below with only the tab "Global Settings" opened. You'll find a number of toggle buttons (functions) that can be activated and on the left-hand side a column of vertical tabs. Let's go through the vertical tabs first.

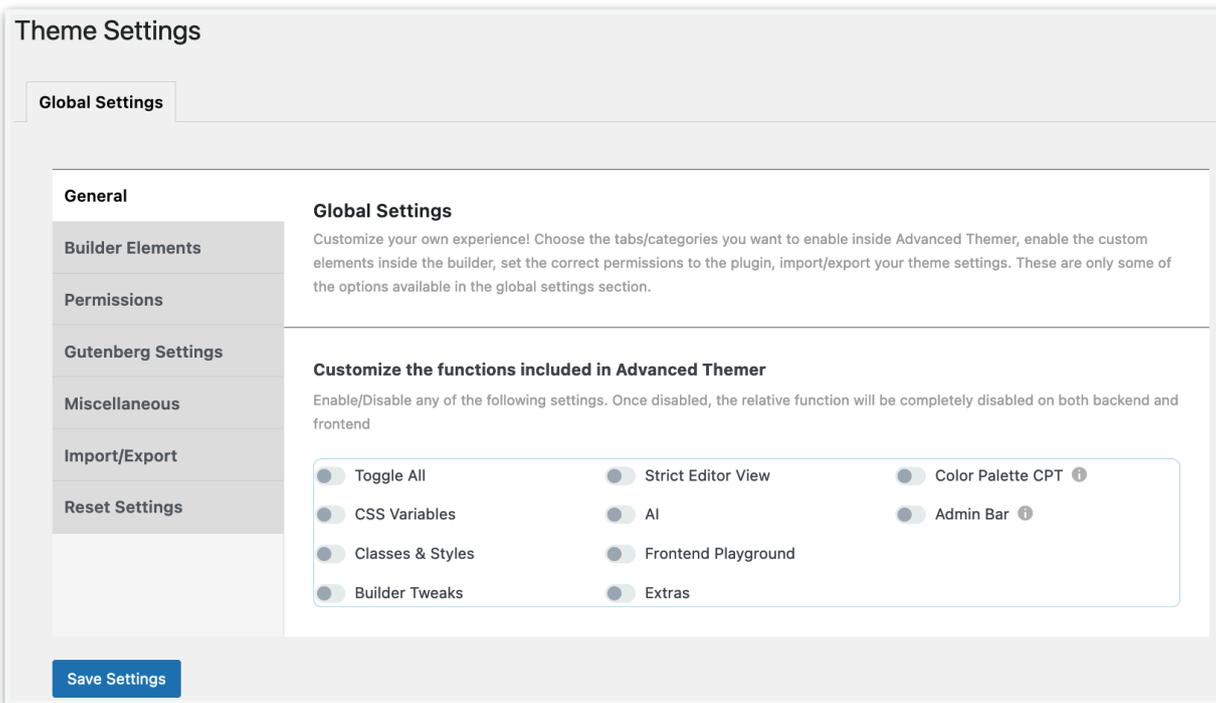
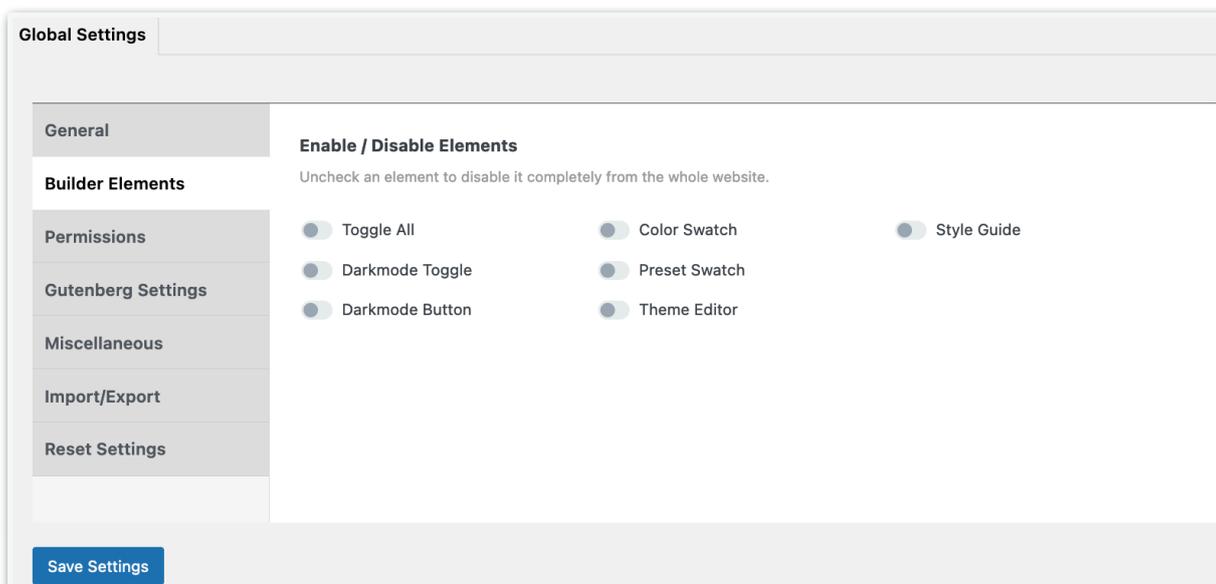
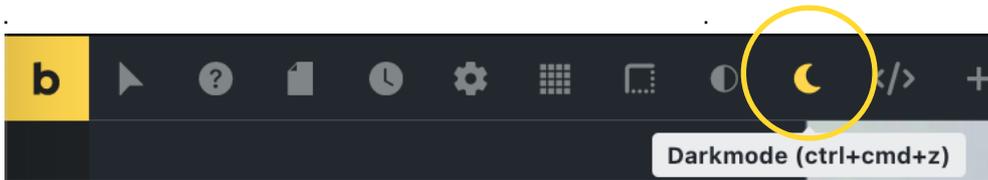


Fig: Opening screen

### 3.1. Builder Elements

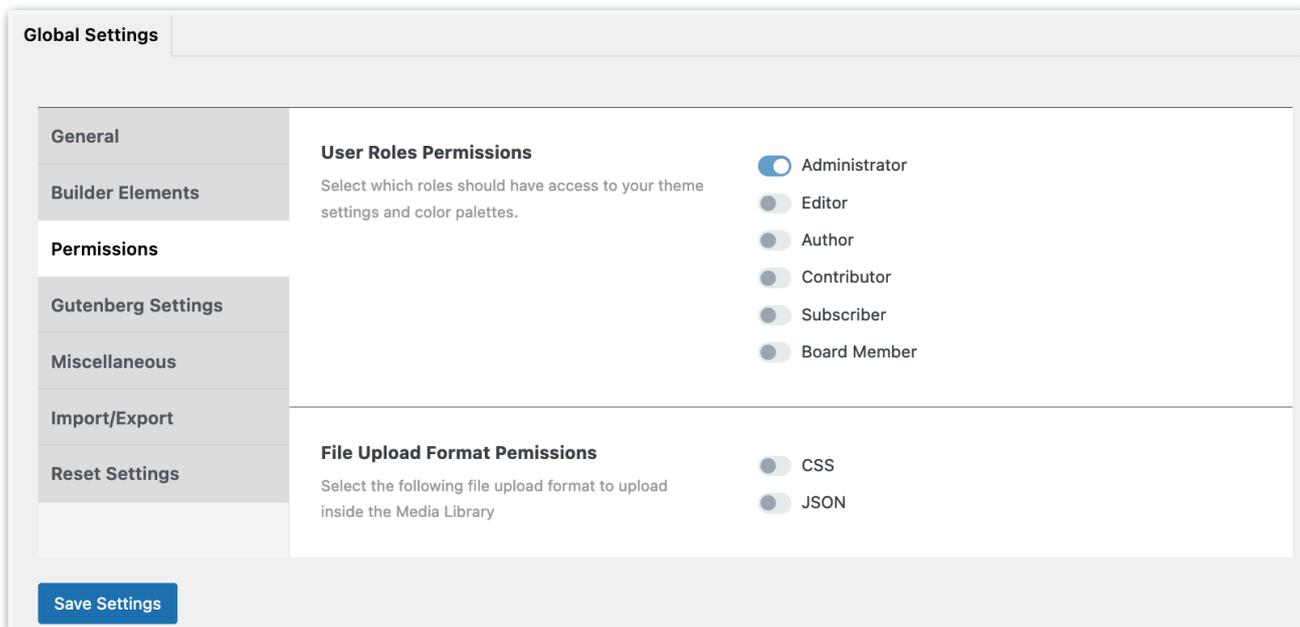
By toggling on these elements their respective icons will turn up in the work area of Bricks. See screenshot of the Bricks left top bar below.





## 3.2. Permissions

Here you determine what access different user roles have to the theme settings and color palettes. You can also set which file format to upload to the media library.



## 3.3. Gutenberg Settings

### 3.3.1. Replace Gutenberg Color Palettes.

When this option is checked, your Bricks color palettes and the Gutenberg color palettes will be synched together. Uncheck this option if you don't plan to use your custom color palettes with Gutenberg.

### 3.3.2. Remove Default Gutenberg Presets

When this option is checked, the default Gutenberg presets CSS variables (like `--wp--preset--color--black`) won't be loaded on the front end anymore.

Global Settings

General	<b>Replace Gutenberg Color Palettes</b> <input type="checkbox"/> No
Builder Elements	When this option is checked, your bricks color palettes and the Gutenberg color palettes will be synched together. Uncheck this option if you don't plan to use your custom color palettes with Gutenberg.
Permissions	
<b>Gutenberg Settings</b>	
Miscellaneous	<b>Remove Default Gutenberg Presets</b> <input type="checkbox"/> No
Import/Export	When this option is checked, the default Gutenberg presets CSS variables (like --wp--preset--color--black) won't be loaded on the frontend anymore.
Reset Settings	

Save Settings

## 3.4. Miscellaneous

### 3.4.1. Disable the "ACF" menu item in your Dashboard

If for some reason you prefer to hide the ACF menu item from your Dashboard, check this toggle. Note that if you have ACF PRO installed, this option will be ignored and the ACF menu will be visible.

Global Settings

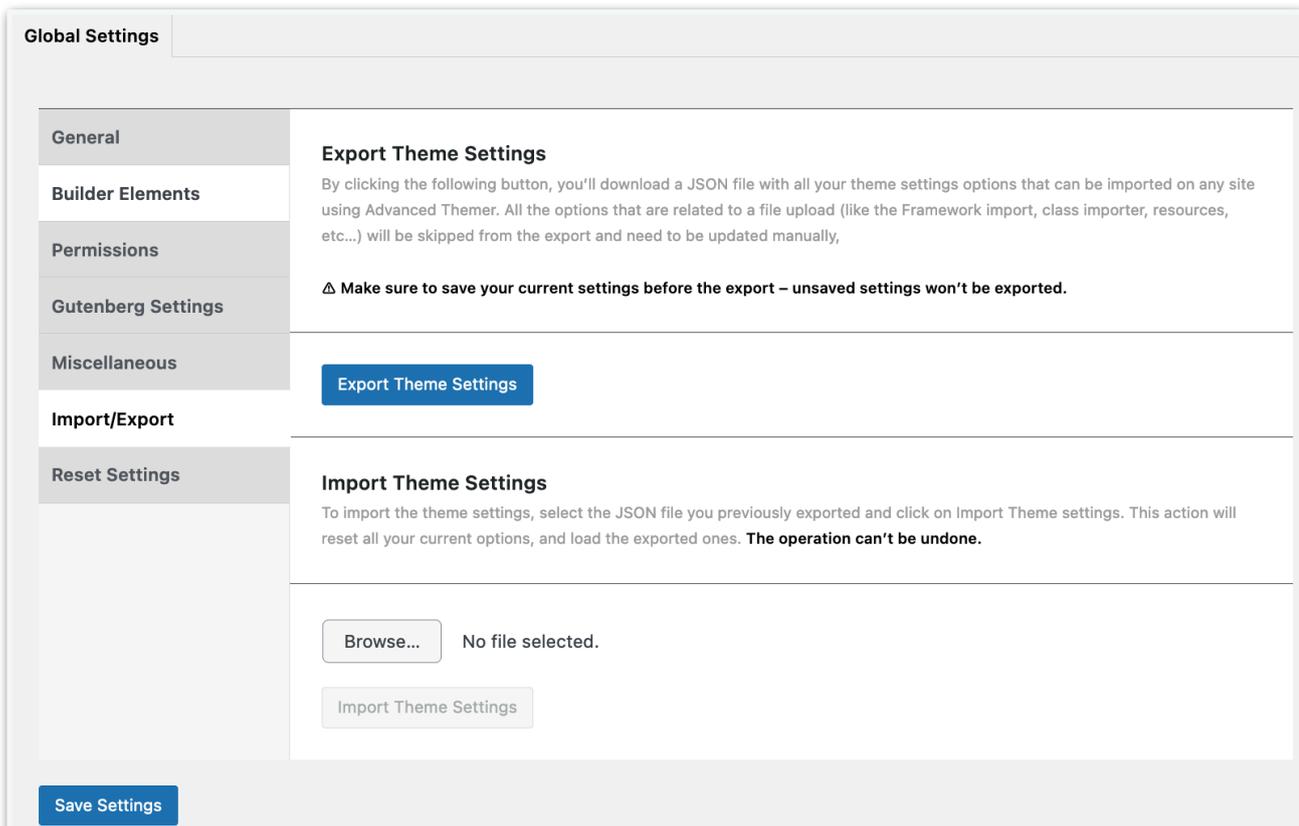
General	<b>Disable the "ACF" menu item in your Dashboard</b> <input checked="" type="checkbox"/> Yes
Builder Elements	If for some reason you prefer to hide the ACF menu item from your Dashboard, check this toggle. Note that if you have ACF PRO installed, this option will be ignored and the ACF menu will be visible.
Permissions	
<b>Gutenberg Settings</b>	
Miscellaneous	<b>Remove all data when uninstalling the plugin</b> <input type="checkbox"/> No
Import/Export	Check this toggle if you want to erase all the data from your database when uninstalling the plugin. This includes all your theme options, your color palettes, and your license.
Reset Settings	

Save Settings

### 3.4.2. Remove all data when uninstalling the plugin

Check this toggle if you want to erase all the data from your database when uninstalling the plugin. This includes all your theme options, your color palettes, and your license.

## 3.5. Import/Export



### 3.5.1. Export Theme Settings

By clicking the blue button, you'll download a JSON file with all your theme settings options that can be imported on any site using Advanced Themer. All the options that are related to a file upload (like the Framework import, class importer, resources, etc...) will be skipped from the export and need to be updated manually,

**⚠ Make sure to save your current settings before the export – unsaved settings won't be exported.**

### 3.5.2. Import Theme Settings

To import the theme settings, select the JSON file you previously exported and click on Import Theme settings. This action will reset all your current options, and load the exported ones. **This operation can't be undone.**

General

Grids

Class Importer

Advanced CSS

**Import your classes from a CSS file**

In the following repeater, you can add/edit/remove your imported Stylesheets. Each row requires a label and a CSS file attached. The version field is optional. Once saved, the CSS file will be automatically enqueued to your website and all the classes in it will be parsed and added inside the Builder.

If you're not allowed to upload CSS files to the Media Library, go to the **Settings tab -> Permissions -> toggle on the CSS option.**

1	<p>Label *</p> <input type="text"/>	<p>Enqueue the CSS</p> <p>Yes <input type="checkbox"/></p>
	<p>Enqueue in *</p> <p>Head <input type="button" value="v"/></p>	<p>Priority *</p> <p>10 <input type="button" value="v"/></p>
	<p>Version</p> <p>1.0.0 <input type="text"/></p>	
	<p>CSS file *</p> <p>No file selected <input type="button" value="Add File"/></p>	

You can choose whether you want to put it in the header or the footer.

### 3.6. Reset Settings

Global Settings

General

Builder Elements

Permissions

Gutenberg Settings

Miscellaneous

Import/Export

Reset Settings

**Reset Theme Settings**

By clicking the following button, you'll reset all the theme options to default. It's recommended to backup your database before proceeding to the theme reset. Note that only the Advanced Themer settings will be erased – all the other ones (including the core Bricks settings) will be untouched.

**⚠ The operation can't be undone.**

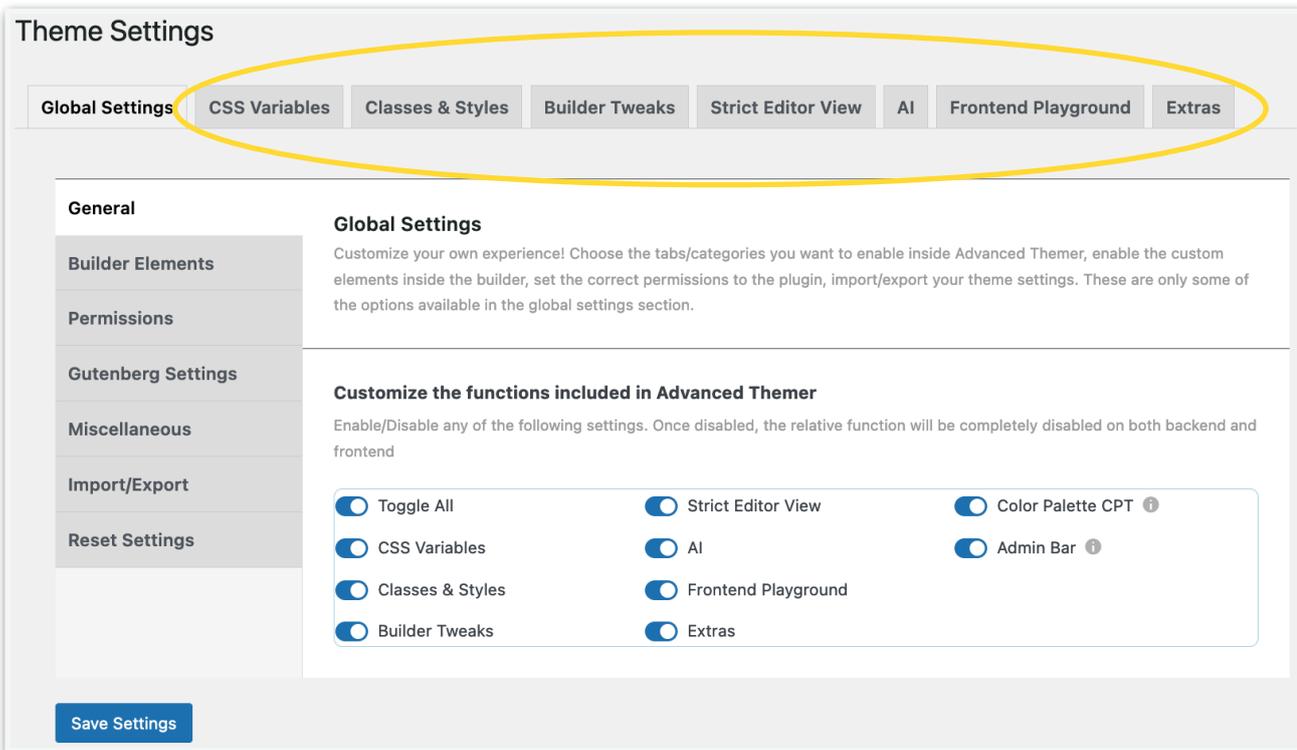
### 3.6.1. Reset Theme Settings

By clicking the blue button, you'll reset all the theme options to default. It's recommended to back up your database before proceeding to the theme reset. Note that only the Advanced Themer settings will be erased – all the other ones (including the core Bricks settings) will be untouched.

⚠ *This operation cannot be undone.*

## 4. Theme Settings - General - Global

Back to the original opening screen, once you click the “Toggle All” button, the screen changes to the following, and the tabs within the orange oval are added:



As you can see there are quite a number of tabs. Let’s go through them one by one, starting with CSS Variables.

### 4.1. CSS Variables

This is the central place to manage all CSS variables easily thanks to the in-built CSS Variables GUI. Now we can create fluid and responsive typography, spacing, border, and width scales in just a few clicks!

More information can be found on the [Official website](#). Let’s dive in...

#### 4.1.1. Theme Settings: Enable CSS Variables Features

This is the full overview of the CSS variables screen. We’ll go over it one by one.

Here you can choose which variables you want to use. Disabling a feature here will also apply to the front end. Below is the overview.

**General**

**CSS Variables**

Manage your CSS variables with ease thanks to our in-built CSS Variables GUI. Create fluid and responsive typography / spacing / border / width scales in few clicks!

🔗 helpful links: [Official website](#)

**Typography**

**Enable CSS Variables Features**

Choose which variables you want to use. Disabling a feature will also apply on frontend.

<input checked="" type="checkbox"/> Toggle All	<input checked="" type="checkbox"/> Border	<input checked="" type="checkbox"/> Width
<input checked="" type="checkbox"/> Typography	<input checked="" type="checkbox"/> Border-Radius	<input checked="" type="checkbox"/> Custom Variables
<input checked="" type="checkbox"/> Spacing	<input checked="" type="checkbox"/> Box-Shadow	<input checked="" type="checkbox"/> Import Framework <span style="font-size: 0.8em;">ⓘ</span>

**Add a prefix to your global CSS variables**

The prefix will be automatically added to all your CSS variables. Example of variable generated with "p1" as prefix: --p1-gap-1.

--AT

**Base Font Size \***

Insert the base font-size you are using on the website. This field is required in order to calculate the correct REM values. Change this value if you know what you're doing!

10 px

The default base font-size in Bricks is 10px.

**Minimum Viewport Width \***

Set the minimum viewport width where the default min value of the clamp function will apply. Above this value, the fluid formula will run until reaching the maximum viewport width.

The default value is set to 360px.

360 px

**Maximum Viewport Width \***

Set the maximum viewport width where the default max value of the clamp function will apply. Below this value, the fluid formula will run until reaching the minimum viewport width.

The default value is set to 1600px.

1600 px

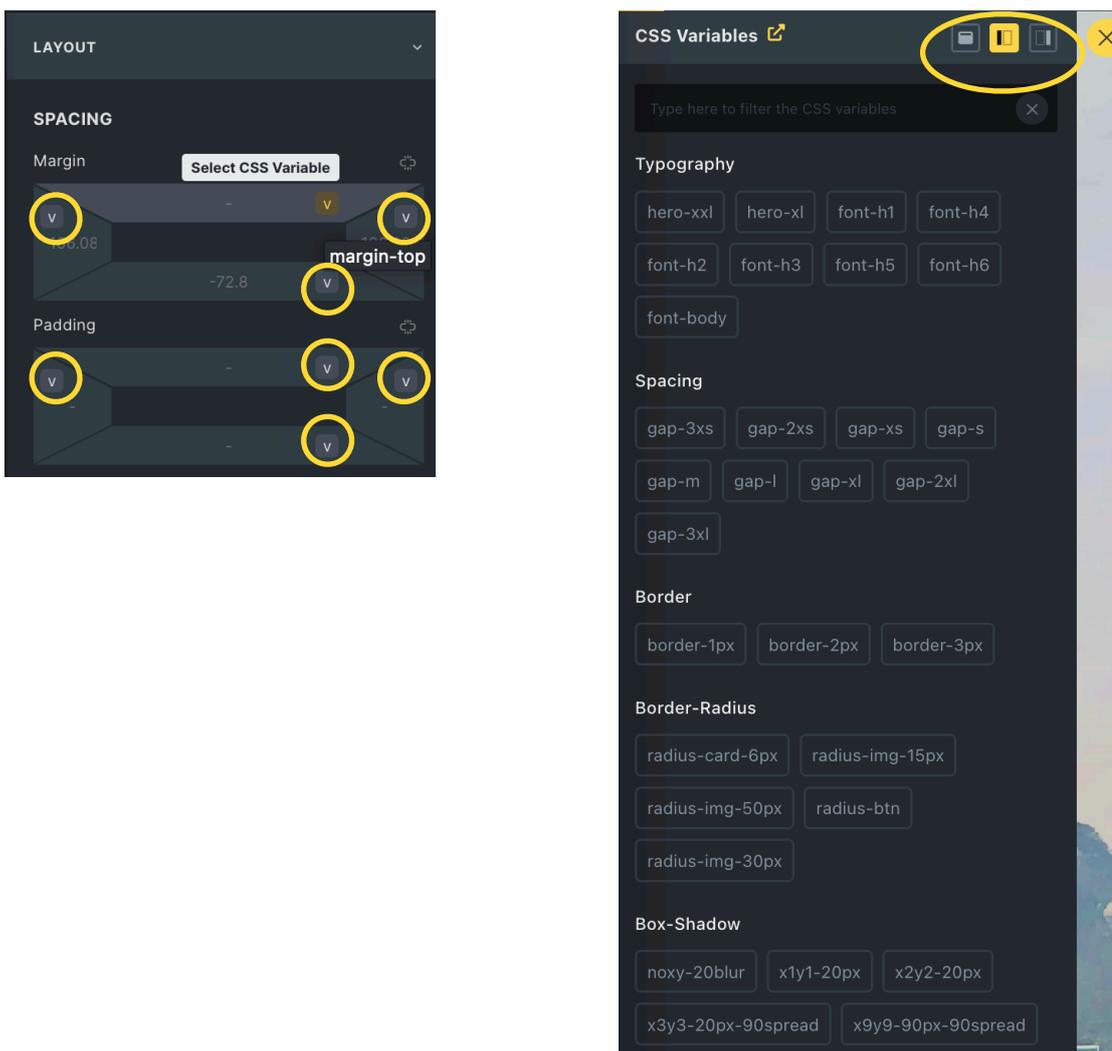
**Clamp Unit**

Choose the CSS unit used inside the clamp function. Note that CQI might not be supported on older browsers.

VW v

Once the CSS variables are enabled some changes happen in the workspace. See the screenshot below on the left of the 'Style' panel in Bricks. Wherever you see the 'v' means that you can use variables. When you click on the 'v', the screen on the right will open with your own defined variables. In the top right corner, you can choose between appearance as a left sidebar, a right sidebar, or a screen-wide modal.

The 'close' button is at the top right.



#### 4.1.2. Add a prefix to your global CSS variables

The prefix will be automatically added to all CSS variables. Example of a variable generated with "p1" as prefix: —p1-gap-1.

***Why is that relevant?***

Well, many Bricks users also work with an (S)CSS framework. The naming convention could be overlapping and confusing, and by putting your own prefix (e.g. AT) before your variables you will always recognize the ones you made.

### 4.1.3. Base Font Size

Insert the base font size you are using on the website. This field is required in order to calculate the correct REM values. Change this value if you know what you're doing!

**The default base font size in Bricks is 10px.**

### 4.1.4. Minimum Viewport Width

Set the minimum viewport width where the default min value of the clamp function will apply. Above this value, the fluid formula will run until reaching the maximum viewport width.

**The default value is set to 360px.**

### 4.1.5. Maximum Viewport Width

Set the maximum viewport width where the default max value of the clamp function will apply. Below this value, the fluid formula will run until reaching the minimum viewport width.

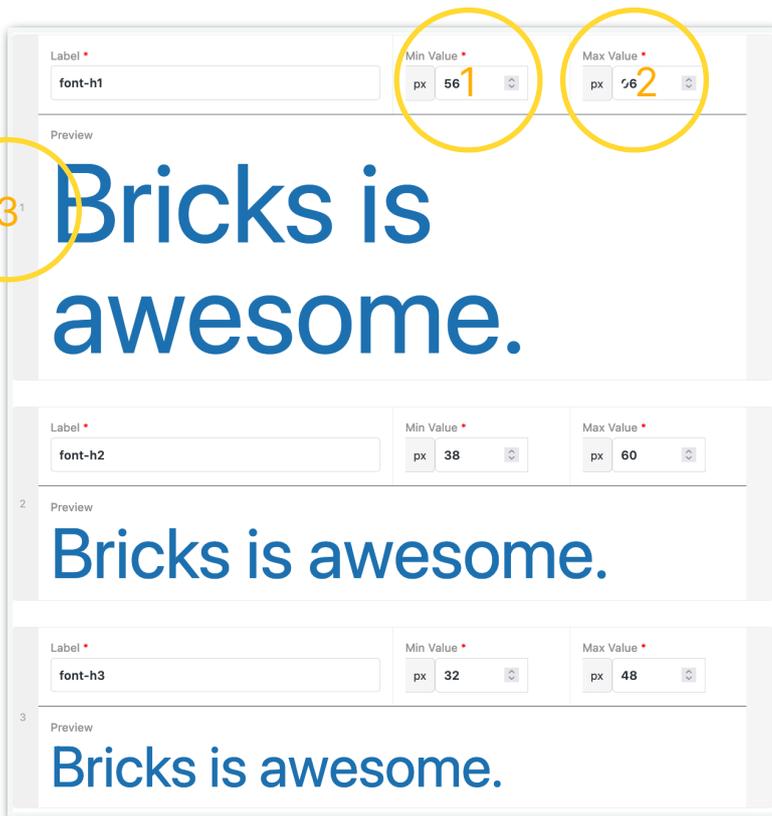
**The default value is set to 1600px.**

### 4.1.6. Clamp Unit

Choose the CSS unit used inside the clamp function. Note that CQI might not be supported on older browsers.

## 4.2. Theme Settings: Typography

In the screenshot, you can see how you can add, edit or remove typography variables. Each row requires a label, a min value, and a max value. The label is used to create the CSS variable like `var(--label)`.



The min value “1” is set in pixels and represents the default value applied when reaching the minimum viewport width set in the Settings tab. The max value “2” is also set in pixels and represents the default max value when reaching the maximum viewport width.

Keep in mind that all pixel values will be converted into REM on the front end.

As we saw in the chapter about Colors, there is a number (see “3”) in front of each row, meaning

you can reorder the different fonts if necessary. The number will change accordingly. At the end of each row are the “+” and “-” buttons to duplicate or delete a row.

Each font is based on a variable and when they interfere with another framework you can easily delete them. The CSS footprint however is very small.

The plugin comes with a variety from “H1” to “H6”, “font subtitles”, and “body”, as well as a “font button”, “font caption”, and “font overline/underline”. You can add as many fonts as you like, creating your own global styles.

The default values are set according to the [Utopia’s fluid type scale calculator](#).

## 4.3. Theme Settings: Spacing

### Spacing Variables

In the screenshot below you can see how you can add, edit, and remove the spacing variables. Each row requires a label (1), a min value (2), and a max value (3). The label is used to create the CSS variable like `var(--label)`. The min value is set in pixels and represents the default value applied when reaching

the minimum viewport width set in the Setting tab. The max value is also set in pixels and represents the default max value when reaching the maximum viewport width.

Keep in mind that all the pixel values will be converted into REM on the front end.

1	Label *	Min Value *	Max Value *
4	gap-xs-1	px 4	px 4
Preview			
[Two blue squares]			
2	Label *	Min Value *	Max Value *
	gap-xs-2	px 8	px 8
Preview			
[Two blue squares]			
3	Label *	Min Value *	Max Value *
	gap-s-1	px 12	px 16
Preview			
[Two blue squares]			
	Label *	Min Value *	Max Value *
	gap-s-2	px 16	px 24

### Gap variables

As we saw with the color shades there is a number (see “4”) in front of each row, meaning you can reorder the different fonts if necessary. The number will change accordingly. At the end of each row are the “+” and “-” buttons to duplicate or delete a row.

The default values are set according to the [Utopia’s fluid space calculator](#).

## 4.4. Theme Settings: Borders

Below you can see how to add, edit, and/or remove border variables. Each row requires a label and a value. The label is used to create your CSS variable like `var(--label)`. The value needs to be a proper CSS border value (example: `1px solid #000000`).

Border variables

## 4.5. Theme Settings: Border Radius

### Border-Radius Variables

You can add, edit, and/or remove the border-radius variables here. Each row requires a label (1), a min value (2), and a max value (3). The label is used to create your CSS variable like `var(--label)`. The min value is set in pixels and represents the default value applied when reaching the minimum viewport width set in the Setting tab. The max value is also set in pixels and represents

the

default max value when reaching the maximum viewport width.

Keep in mind that all the pixel values will be converted into CQI/REM on the front end.

As we saw before there is a number (see “4”) in front of each row, meaning you can reorder the different fonts if necessary. The number will change accordingly. At the end of each row are the “+” and “-” buttons to duplicate or delete a row.

When I add a new radius circle, I give it a label, minimum value, and max value, and voila. See the result below:

The screenshot shows a configuration panel for a new spacing variable. It has three main input fields: 'Label \*' with the value 'radius-circle', 'Min Value \*' with 'px' and '50', and 'Max Value \*' with 'px' and '100'. Below these is a 'Preview' section with a blue circle and the number '4' next to it. At the bottom right, there is a blue button that says 'Add a new spacing variable'.

## 4.6. Theme Settings: Box-Shadow

In the screenshot, you can see how to add, edit, and/or remove box-shadow variables. Each row requires a label and a value. The label is used to create a CSS variable like `var(-label)`. The value needs to be a proper CSS box-shadow value (example: `0px 20px 40px #000`).

The screenshot shows the 'Box-Shadow' settings in a theme editor. On the left is a sidebar with categories: General, Typography, Spacing, Border, Border-radius, Box-Shadow, Width, Custom Variables, and Import Framework. The main area is titled 'Box-Shadow' and contains a list of variables. Each variable has a 'Label \*' and a 'Value \*'. The first variable has the label 'noXY-20blur' and the value '0px 0px 20px #00000030'. Below it is a preview of a blue square with a shadow. The second variable has the label 'X1Y1-20px' and the value '1px 1px 20px #00000030', also with a preview of a blue square with a shadow.

## 4.7. Theme Settings: Width

In the screenshot below, you can see how to add, edit, and/or remove Width variables. Each row requires a label, a min value, and a max value. The label is used to create your CSS variable like `var(-label)`. The min value is set in pixels and represents the default value applied when reaching the minimum viewport width set in the Setting tab. The max value is also set in Pixels and represents the default max value when reaching the maximum viewport width.

Keep in mind that all the pixel values will be converted in CQI/REM on the front end.

**General**

**Typography**

**Spacing**

**Border**

**Border-radius**

**Box-Shadow**

**Width**

**Custom Variables**

**Import Framework**

**Width**

In the following repeater, you can add/edit/remove your width variables. Each row requires a label, a min value, and a max value. The label is used to create your CSS variable like `var(-label)`. The min value is set in Pixels and represents the default value applied when reaching the minimum viewport width set in the Setting tab. The max value is also set in Pixels and represents the default max value when reaching the maximum viewport width. Keep in mind that all the pixels values will be converted in CQI/REM on frontend.

Label *	Min Value *	Max Value *
MinMax376-1366	px 376	px 1366

1 Preview

Add a new width variable

## 4.8. Theme Settings: Custom Variables

You can create, edit or delete your own custom variables here, instead of going into the 'root' and doing your own custom CSS. Here is the place to make it happen.

First, create a category where the variable will be stored. The category label will be shown inside the Variable Picker. Each row requires a label and a value.

The label is used to create your CSS variable like `var(--label)`. Choose between a static (1) or a fluid (2) (clamp) variable.

**Custom Variables**

In the following repeater, you can add/edit/remove your own variables. First, create a category where the variable will be stored. The category label will be shown inside the Variable Picker. Each row requires a label and a value. The label is used to create your CSS variable like `var(--label)`. Choose between a static or a fluid (clamp) variable.

Category Label \*

spacing

CSS Variables

1 Static Variable

Label \*

space-extra-small

Value \*

.5rem

2 Fluid Variable

Label \*

space-small

Min Value \*

px 10

Max Value \*

px 20

Add a Variable

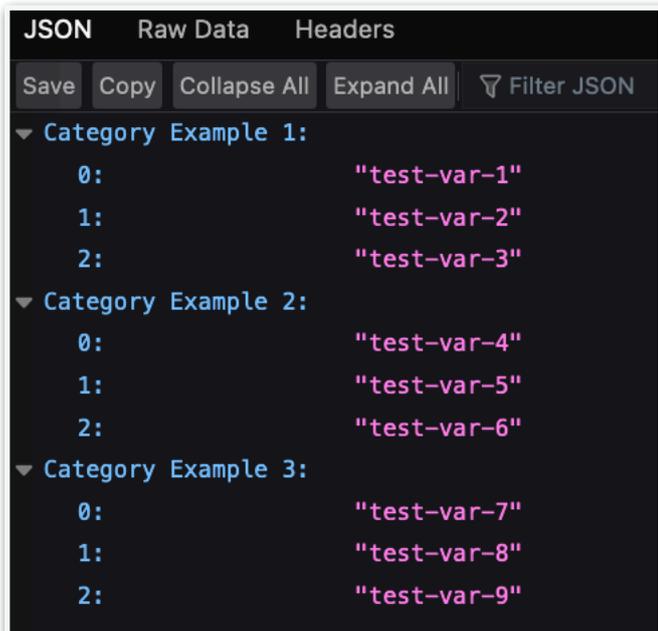
Add a Category

Custom Variables

As mentioned previously there is a number in front of each row, meaning you can reorder the different variables if necessary. The number will change accordingly. At the end of each row are the “+” and “-” buttons to duplicate or delete a row.

## 4.9. Import your own CSS Variable Framework

In this section, you can upload your own CSS Variable Framework. To do so, just set a label and select the JSON file that contains your categories and variable values. In order to work correctly, you need to follow the same semantics as in [this example](#) (see screenshot below). If you’re not allowed to upload JSON files to the Media Library, go to the **Settings tab → Permissions → toggle on the JSON option**.



The screenshot shows a JSON viewer interface with a dark theme. At the top, there are three tabs: "JSON", "Raw Data", and "Headers", with "JSON" selected. Below the tabs is a toolbar with buttons for "Save", "Copy", "Collapse All", "Expand All", and "Filter JSON". The main content area displays a JSON structure with three categories, each expanded to show three indexed items:

```
▼ Category Example 1:  
  0: "test-var-1"  
  1: "test-var-2"  
  2: "test-var-3"  
▼ Category Example 2:  
  0: "test-var-4"  
  1: "test-var-5"  
  2: "test-var-6"  
▼ Category Example 3:  
  0: "test-var-7"  
  1: "test-var-8"  
  2: "test-var-9"
```

Example semantics

## 5. Classes & Styles

Improve the way of handling styles and classes inside the Bricks Builder.

① helpful links: [Official website](#)

**General**

**Classes & Styles**  
Improve the way you're handling styles and classes inside the Bricks Builder.

① helpful links: [Official website](#)

**Enable Classes & Styles Features**  
Enable the following features related to your classes & styles inside the Bricks Builder.

Toggle All  Class Importer ⓘ

Grids ⓘ  Advanced CSS ⓘ

### Classes & Styles

## 5.1. Classes & Styles: Grids

As shown in the screenshot below, you can add, edit, and/or remove your grid classes. Each row requires a class name (without dots), a gap value, a

**Grid Classes**

In the following repeater, you can add/edit/remove your grid classes. Each row requires a class name (without dots), a gap value, a maximum number of columns, and a minimum column width (expressed in pixels). Once saved, the classes will be available inside the Builder. Note that grids are already fully responsive.

1	Class * grid-3	Gap * 2rem	Max N° of Cols * 3	Min Col Width * px 280
2	Class * grid-4	Gap * 2rem	Max N° of Cols * 4	Min Col Width * px 250
3	Class * grid-6	Gap * 2rem	Max N° of Cols * 6	Min Col Width * px 200

[Add a new grid class](#)

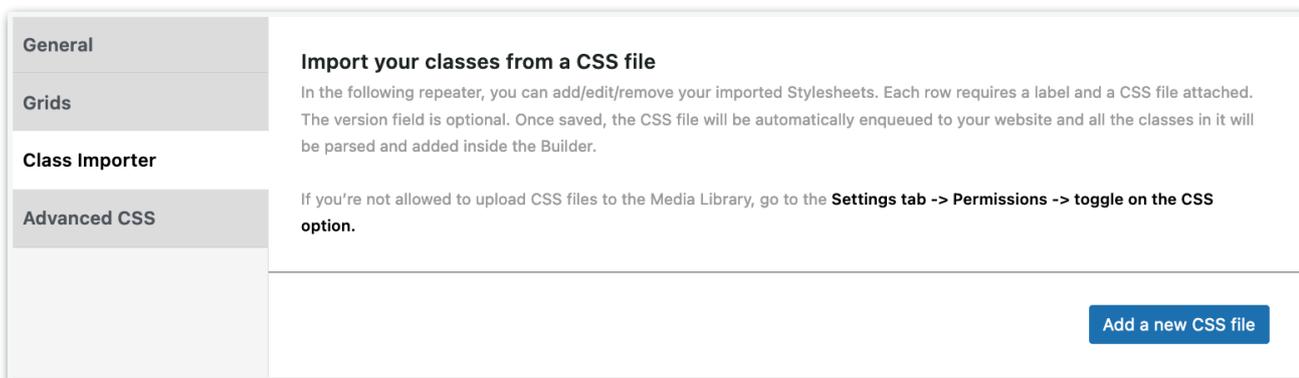
maximum number of columns, and a minimum column width (expressed in pixels). Once saved, the classes will be available inside the Builder. Note that grids are already fully responsive.

Here as well is a number in front of each row, meaning you can reorder the different variables if necessary. The number will change accordingly. At the end of each row are the “+” and “-” buttons to duplicate or delete a row.

## 5.2. Classes & Styles: Class Importer

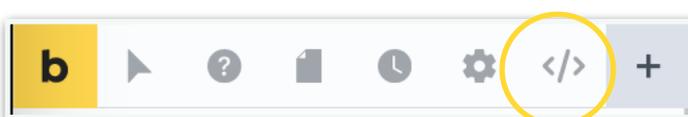
In the screenshot below you’ll see a repeater, which you can add, edit, and/or remove your imported Stylesheets. Each row requires a label and a CSS file attached. The version field is optional. Once saved, the CSS file will be automatically enqueued to your website, and all the classes in it will be parsed and added inside the Builder.

If you’re not allowed to upload CSS files to the Media Library, go to the **Settings tab → Permissions → toggle on the CSS option.**



## 5.3. Classes & Styles: Advanced CSS editor

Advanced CSS is a powerful CSS editor integrated into the Bricks builder. It comes with many improvements compared to the native CSS editor of Bricks. To start using it, click on the corresponding icon inside the builder’s left top bar (see screenshot).

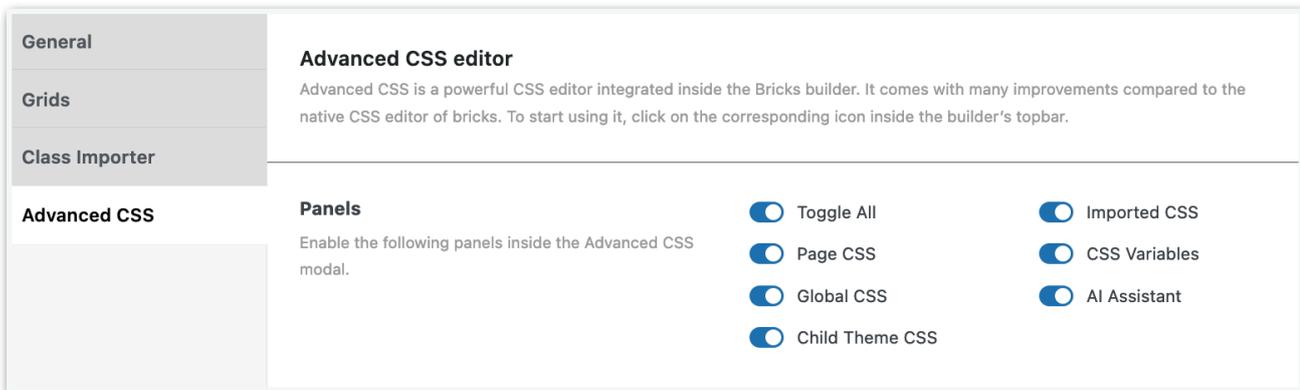


Bricks Builder Left Top Bar with Advanced CSS icon.

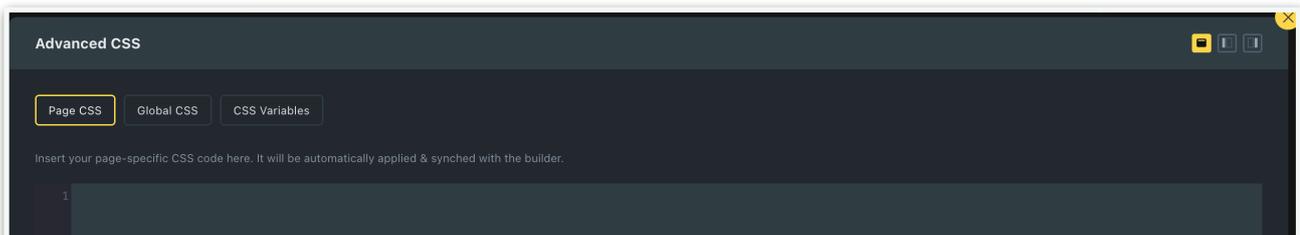
### 5.3.1. Panels

Toggle On/Off will enable/disable the following panels inside the Advanced CSS modal at the bottom of the page.

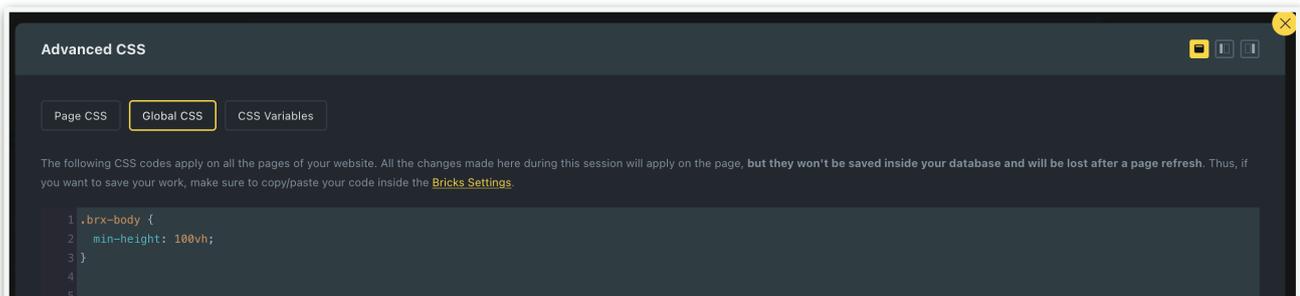
- Toggle All
- Page CSS
- Global CSS
- Child Theme CSS
- Imported CSS
- CSS Variables
- AI Assistant

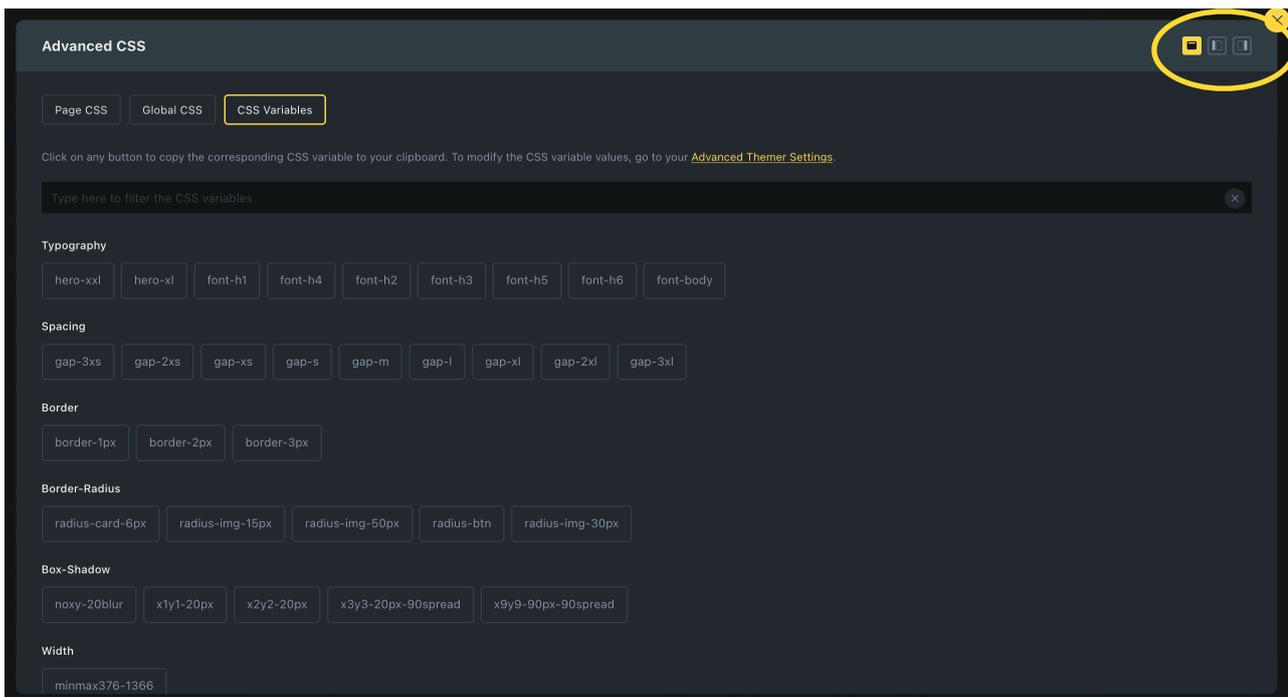


When you click the 'advanced CSS' button in the topbar the screenshot with Page CSS below is what you'll see.



The other two use cases are as follows:





and the CSS variables. With the buttons in the top right corner you can choose the have the panel as a left sidebar, a right sidebar or a full screen modal.

# 6. Builder Tweaks

In this chapter, you'll find a number of options for boosting productivity inside the Bricks builder. Helpful links: [Official website](#)

## 6.1. Builder Tweaks: General

**General**

**Builder Tweaks**  
In this section, you'll find numerous options for boosting your productivity inside the Bricks builder.

helpful links: [Official website](#)

**Enable Builder Tweaks Features**  
Enable the following tweaks inside the Bricks Builder.

- Toggle All
- Topbar
- Structure Panel
- Classes & Styles
- Elements
- Keyboard Shortcuts

**Annotations:**

- These tweaks represents features that you can enable globally inside the builder. They usually create a new icon inside the topbar. (Points to Toggle All and Topbar)
- These tweaks help you to increase your productivity when dealing with classes and styles. (Points to Structure Panel and Classes & Styles)
- These tweaks help you to increase your productivity when dealing with classes and styles. (Points to Elements and Keyboard Shortcuts)
- These tweaks add new features when handling the Bricks elements. (Points to Elements)

## 6.2. Topbar

### 6.2.1. Enable Global Features

**General**

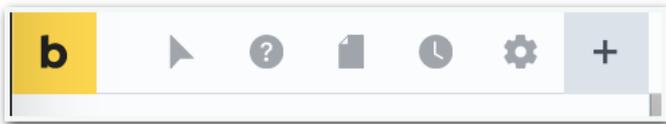
**Enable Global Features**  
Enable the following features inside the Bricks Builder. Once activated, a dedicated icon will be shown inside the Bricks Builder Toolbar.

- Toggle All
- X-Mode
- Darkmode
- Grid Guide
- Contrast Checker
- Open in a New Tab

**Grid Columns \***  
Specify the number of columns for the Grid Guide feature. Default: 12.  col

**Grid Gap \***  
Specify the gap between the columns for the Grid Guide feature. Default: 20px.  px

Enabling the different features inside the Bricks Builder will show additional dedicated icons. See the Topbar 'comparison' below.

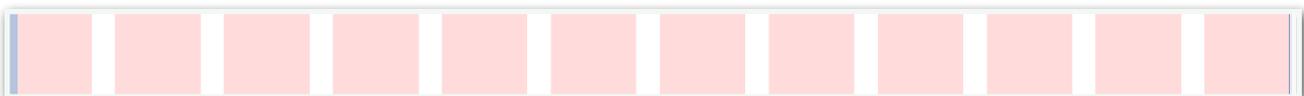


The 'normal' left top bar.

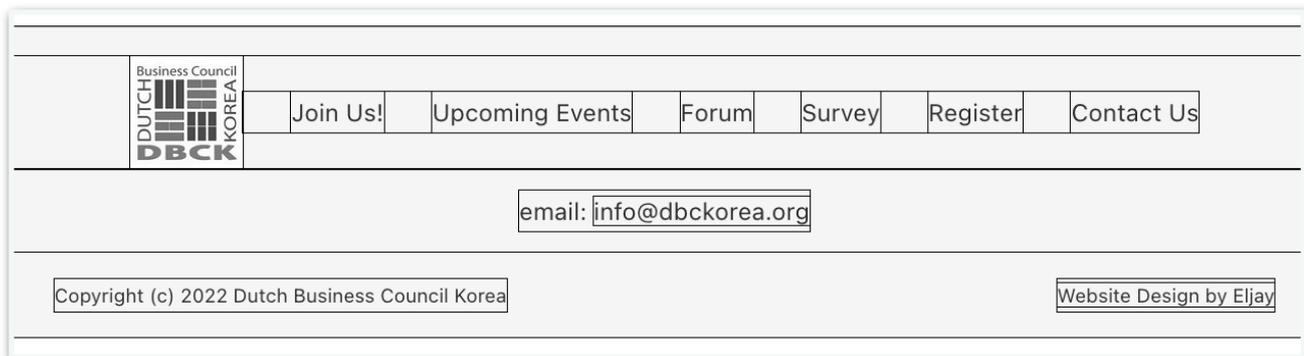


The Enhanced left top bar with 5 additional icons

1. Grid Guide: when you click this icon the whole screen shows the guide.



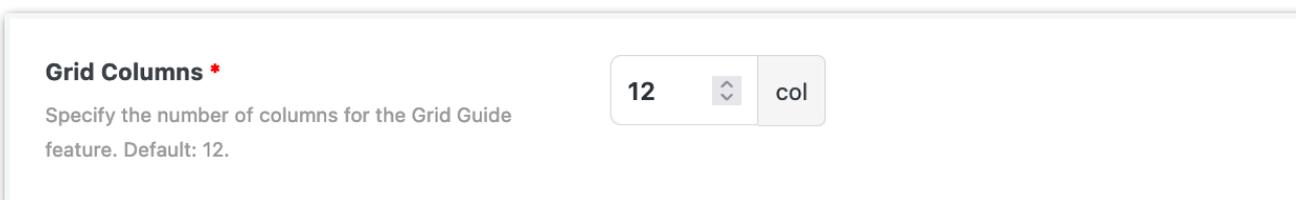
2. X-mode: choosing this basically changes the site into wireframe mode.



- 3. Contrast Checker
- 4. Dark Mode
- 5. Advanced CSS

### 6.2.2. Grid Columns

Here you can specify the number of columns for the Grid Guide feature. Default = 12.



### 6.2.3. Grid Gap

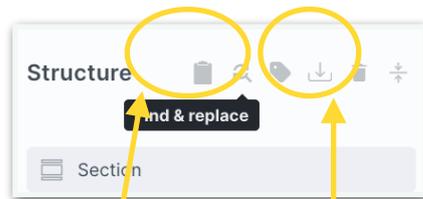
Here you can specify the gap between the columns for the Grid Guide feature. Default = 20px.

**Grid Gap \***

Specify the gap between the columns for the Grid Guide feature. Default: 20px.

## 6.3. Structure Panel Icons

Two more icons will show in Bricks Builder's right top bar. Find & replace (1), and show Elements Tag (2).



**Structure Panel Icons**

Enable the following icons inside the header of the structure panel

Toggle All     Find & replace     Elements Tag ⓘ

**Default Tag View**

Select the default view of the elements tag when loading the builder.

Developer Mode - With colors and dropdowns ▾

**Contextual Menu**

Enable the following options to be shown inside the contextual menu

Toggle All     Extend Classes & Styles ⓘ     Class Converter ⓘ

Hide Element ⓘ     Find & Replace ⓘ

**General Tweaks**

Enable the following global features inside the structure panel

Toggle All     Styles & Classes indicators ⓘ

Enable shortcuts for creating new elements. ⓘ     Resizable Structure Panel ⓘ

**Elements Shortcuts**

Select the elements shortcuts you want to display inside the right sidebar

Toggle All     Heading     Image

Section     Text basic     Video

Container     Rich Text     Icon box

Block     Button     Social icons

Div     Icon     List

When this option is enabled, a new icon will show up inside the structure panel's header. When clicked on it, the HTML tag of each element will show up inside your structure.

When clicked on this contextual menu item, the element will be set as display:none.

When clicked on this contextual menu item, a modal will show up and let you automatically convert the ID styles of an element (and his children) to an autogenerated class.

When clicked on this contextual menu item, a modal will show up and let you extend the classes & styles of the selected element to others elements inside the structure.

When clicked on this contextual menu item, a modal will show up and let you automatically convert the ID styles of an element (and his children) to an autogenerated class.

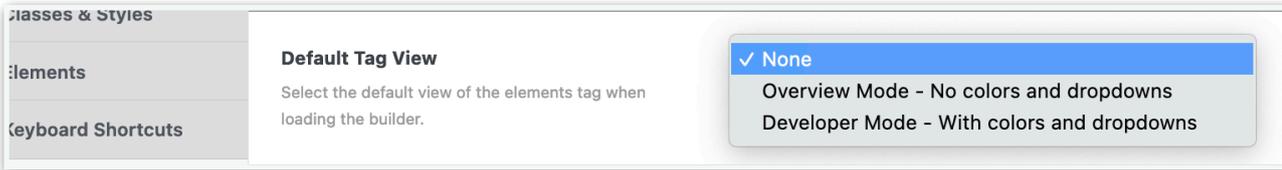
When this option is checked, you'll be able to resize the structure panel the exact same way you can do it with the element panel.

When this option is checked, a right sidebar will be created inside the structure panel with shortcuts of the most-used elements that you can add to your structure on the fly.

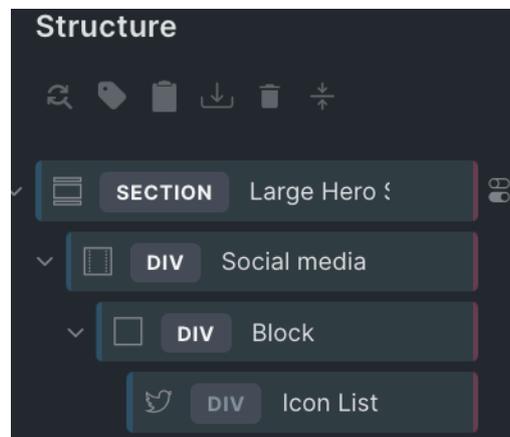
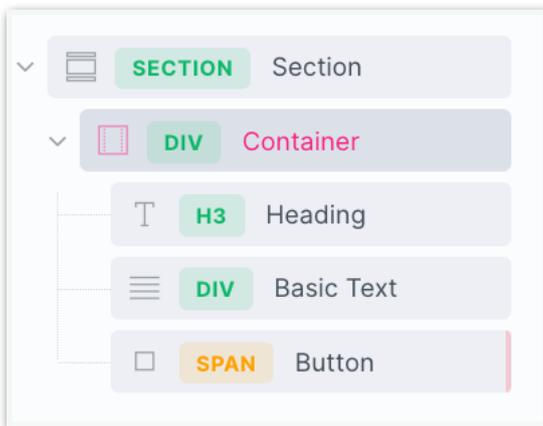
When this option is checked, you'll see new bar indicators inside your structure panel elements. The left bar indicates that the element has at least one global class applied. The right one indicates that it contain styles on the ID level.

### 6.3.1. Default Tag View

Default Tag View has a dropdown menu. The choice will enhance the Structure Panel in Bricks Builder.

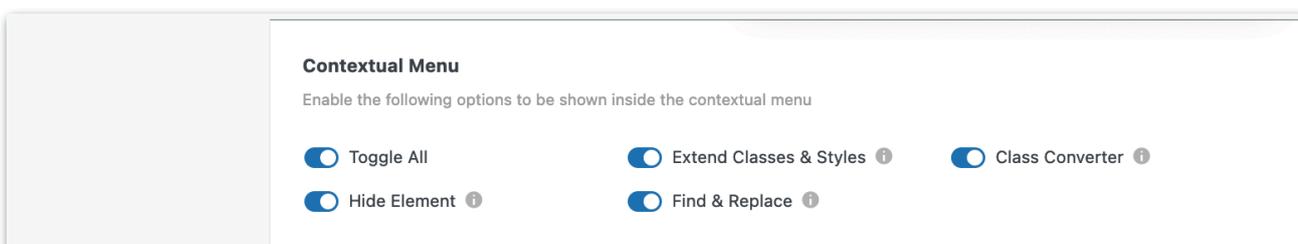


In Developer Mode - With colors and drop-downs the structure panel will change into the screenshot below left. Overview Mode - No colors see screenshot below right.



*The red indicator on the right side of the element means that there is some styling done at ID-level. The blue bar on the left side indicates that at least one global class has been created.*

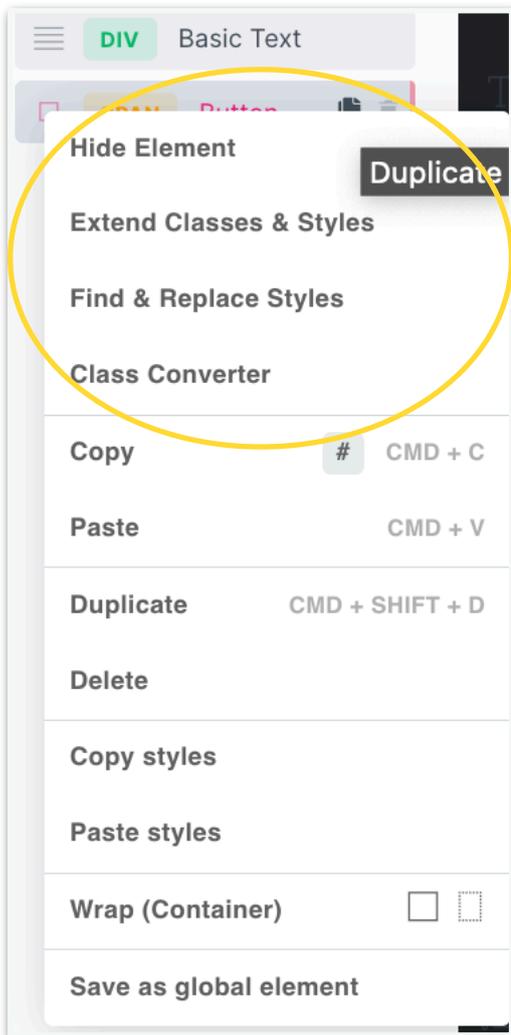
### 6.3.2. Contextual Menu



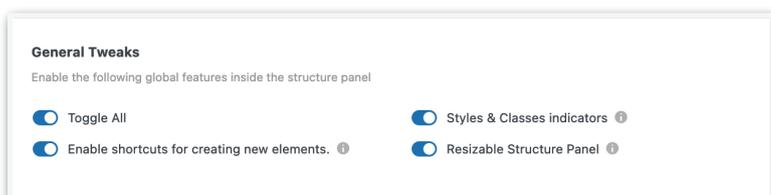
When all buttons are toggled added to the contextual menu (right mouse click) in the structure panel are:

1. Hide Element: the element will be set as display: none
2. Extend Classes & Styles,
3. Find & Replace Styles

#### 4. Class Converter



### 6.3.3. General Tweaks



The Styles & Classes indicators can be found in the structure panel.



## 6.4. Builder Tweaks for Classes & Styles

Enable/Disable any of the following builder tweaks related to classes and styles. [Learn more about the builder tweaks for classes & styles](#)

The image shows a settings panel for 'Builder Tweaks for Classes & Styles' with ten toggle options. Each option has an information icon (i) to its right. Arrows point from callout boxes on the right to the information icons of the corresponding options.

- Toggle All**
- Extend Global Classes and Styles.** *i*

This feature will consent you to extend the classes & styles from an element to his parents/children
- Find & Replace Styles.** *i*

This feature will consent you to replace any style value assigned to any element inside the builder.
- Reorder the global classes alphabetically.** *i*

Check this option if you want your global classes reordered alphabetically inside the Builder.
- Lock Styles on element ID level.** *i*

Check this option if you want to lock the ability to style your elements on an ID level. In order to style your elements, you'll need to either create/activate a class or click on lock icon to unlock the style tab.
- CSS Variables Picker.** *i*

When this option is checked, you'll see a new icon popping up on the relevant style fields inside the Bricks builder. When clicked on it, the script will open a modal where you can pick the CSS variable of your choice.
- Autocomplete for CSS Variables.** *i*

When this option is checked, a bottom popup will show up at the bottom of each field when typing with the list of all the matching CSS variables.
- Plain Classes.** *i*

When this option is checked, a new icon will show up next to the element's class field. When you click on it, a popup window will appear where you can type the classes you want to add/remove in bulk.
- Export ID Styles to Classes.** *i*

When this option is checked, a new export icon will show up next to the element's class field. When you'll click on it, you'll be able to insert a class name and export all your ID styles to it. Note that you can also import the ID styles if a class is activated.
- Highlight Classes.** *i*

When this option is checked, a blue outline will appear on all the elements that share the same class when you select it inside the builder. It's a great way to localize where your classes are applied.

When this option is checked, a new counter will show up next to the class name that indicates the number of times the class is used on the page. Clicking on the counter will scroll the page to each element that is using the active class.

When this option is checked and the color grid of any element is open, hovering on each color will temporarily apply the color to the element. This is a great way to preview your colors inside the builder.

When this option is checked and the class dropdown of any element is open, hovering on each class will temporarily apply the class to the element. This is a great way to preview the impact of a class to your elements inside the builder.

When this option is checked, you'll see a new bar indicator on each media query icon where styles are applied. It works both for ID styles and styles applied on a class.

When this option is checked, you'll see a new blue dot on the left of all the controls that have a style generated from an active class.

When this option is checked, you'll see a new small device icon next to each group that has style set on different breakpoint inside the style tab.

When this option is checked, the locked classes will appear with red background inside the builder. The unlocked ones will be displayed with a green background.

When this option is checked, a new clone icon will show up once you activate a class. Once clicked, an input will be visible with the current class name prefiled. Quickly change the name of the class and save it. All the styles will be copied to the new class.

When this option is checked, a new clone icon will show up once you activate a class. Once clicked, the active class's name will be copied to the clipboard.

**Count Classes & Navigation.** ⓘ

**Color Preview on hover.** ⓘ

**Class Preview on hover.** ⓘ

**Style Indicators in the media queries panel.** ⓘ

**Indicators of styles inherited from a class.** ⓘ

**Breakpoint Indicator.** ⓘ

**Locked Class Indicator.** ⓘ

**Clone Class.** ⓘ

**Copy Class to Clipboard.** ⓘ

## 6.5. Builder Tweaks for the Elements

### Overview

- General
- Topbar
- Structure Panel
- Classes & Styles
- Elements**
- Keyboard Shortcuts

#### Builder Tweaks for the Elements

Enable/Disable any of the following builder tweaks related to the elements. [Learn more about the general builder tweaks](#)

- Toggle All
- Enable Tabs Shortcuts. ⓘ
- Activate the Pseudo-Element Shortcuts. ⓘ
- Enable Go to Parent Shortcut. ⓘ
- Enable Lorem Ipsum Generator. ⓘ
- Disable the PIN Icon on the elements list. ⓘ
- Close all open Style accordeons by default. ⓘ
- Disable element's outline when styling Borders and Box-shadow. ⓘ
- Change the columns of the elements list panel. ⓘ

#### Tabs Shortcuts

Select the shortcut icons you want to display inside each element panel. This will create an icon for each Content/Style Tab in order to quickly access the accordion tab when styling an element inside the Builder

- Toggle All
- content
- layout
- typography
- background
- borders
- gradient
- transform
- css
- attributes

#### Pseudo Elements Shortcuts

Select the shortcut icons you want to display inside each element panel. This will create an icon for each status in order to quickly activate/deactivate your pseudo-classes when styling an element inside the Builder

- Toggle All
- hover
- before
- after
- active
- focus

#### Type of dummy content

Choose between the classic Latin Lorem Ipsus text or the human-readable Website Ipsum created by [Kyle Van Deusen](#)

#### Default Elements List Columns

Set the default number of columns of the elements list panel when the page is loaded.

- 2 columns
- 3 columns
- 4 columns

### 6.5.1. Builder Tweaks for the Elements Details

Enable or Disable any of the following builder tweaks related to the elements. [Learn more about the general builder tweaks](#)

### Builder Tweaks for the Elements

Enable/Disable any of the following builder tweaks related to the Builder.

- Toggle All
- Enable Tabs Shortcuts. ⓘ
- Activate the Pseudo-Element Shortcuts. ⓘ
- Enable Go to Parent Shortcut. ⓘ
- Enable Lorem Ipsum Generator. ⓘ

---

- Disable the PIN Icon on the elements list. ⓘ
- Close all open Style accordeons by default. ⓘ
- Disable element's outline when styling Borders and Box-shadow. ⓘ
- Change the columns of the elements list panel. ⓘ

Check this option to enable the left-panel shortcuts to quickly access to your style groups.

When this option is checked, new icon shortcuts will display next to the Conditions and Interactions icons.

When this option is checked, you'll see a new icon popping up on the left panel of each element. Clicking on this icon will activate the parent element.

When this option is checked, you'll see a new icon popping up on the relevant text/textarea fields inside the Bricks builder. When clicked on it, the script will automatically generate dummy content for that specific field.

When this option is checked, you'll see a new icon popping up on the left panel of each element. Clicking on this icon will activate the parent element.

When this option is checked, all the tabs of the Style panel will be closed by default. This allows you to avoid closing the layout tab continuously when styling an element.

When this option is checked, you'll see 3 new icons on the top-right of the global elements panel that will consent you to control the grid's column number.

When this option is checked, the green outline that surrounds the active element will be removed to consent you to easily style both borders and box-shadows.

## 6.5.2. Tabs Shortcuts

### Tabs Shortcuts

Select the shortcut icons you want to display inside each element panel. This will create an icon for each Content/Style Tab in order to quickly access the accordion tab when styling an element inside the Builder

<input checked="" type="radio"/> Toggle All	<input checked="" type="radio"/> background	<input checked="" type="radio"/> css
<input checked="" type="radio"/> content	<input checked="" type="radio"/> borders	<input checked="" type="radio"/> attributes
<input checked="" type="radio"/> layout	<input checked="" type="radio"/> gradient	
<input checked="" type="radio"/> typography	<input checked="" type="radio"/> transform	

---

### Pseudo Elements Shortcuts

Select the shortcut icons you want to display inside each element panel. This will create an icon for each status in order to quickly activate/deactivate your pseudo-classes when styling an element inside the Builder

<input checked="" type="radio"/> Toggle All	<input checked="" type="radio"/> before	<input checked="" type="radio"/> active
<input checked="" type="radio"/> hover	<input checked="" type="radio"/> after	<input checked="" type="radio"/> focus

---

### Type of dummy content

Choose between the classic Latin Lorem Ipsus text or the human-readable Website Ipsum created by [Kyle Van Deusen](#)

Lorem Ipsum ▼

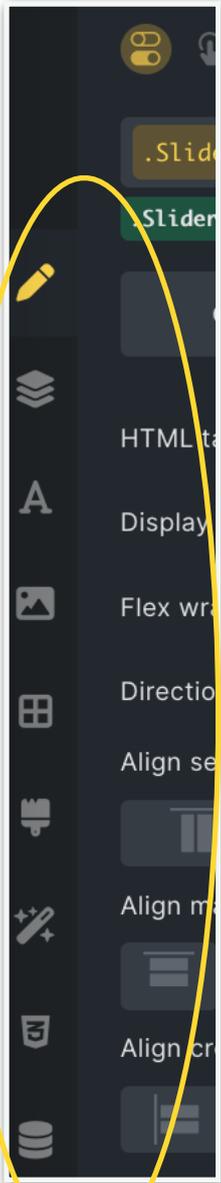
---

### Default Elements List Columns

Set the default number of columns of the elements list panel when the page is loaded.

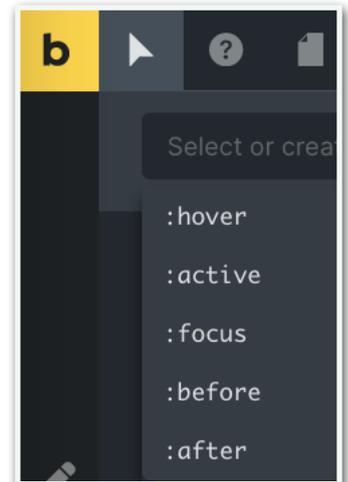
✓ 2 columns  
 3 columns  
 4 columns

Select the shortcut icons you want to display on the left of the element panel. This will create an icon for each Content/Style Tab in order to quickly access the accordion tab when styling an element inside the Builder (see screenshot below left). You will see the icons when you have a class selected.



### 6.5.3.Pseudo Elements Shortcuts

Select the shortcut icons you want to display inside each element panel. This will create an icon for each status in order to quickly activate/deactivate your pseudo-classes when styling an element inside the Builder.

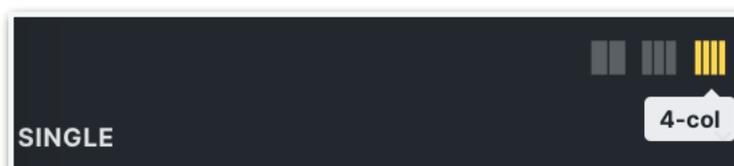


### 6.5.4.Type of Dummy content

Now you can choose between the classic Latin Lorem Ipsum text or the human-readable Website Ipsum created by [Kyle Van Deusen](#).

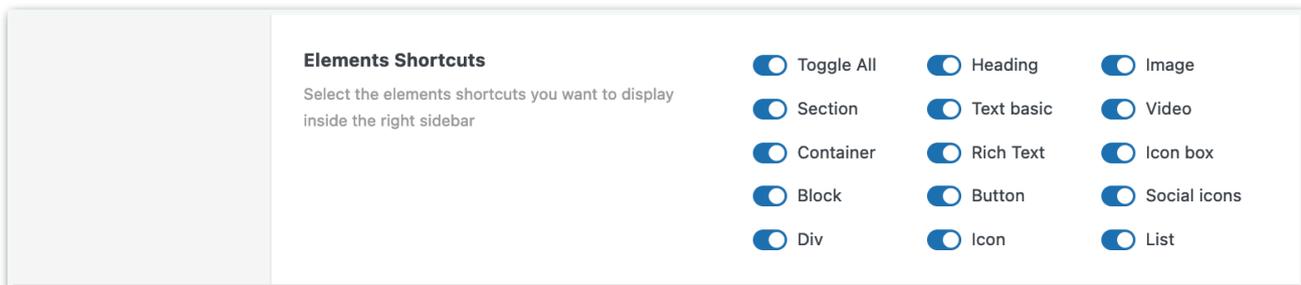
### 6.5.5.Default Elements List Columns

Set the default number of columns of the elements list panel when the page is loaded. Choose between a two, three, or four-column layout for your elements.



## 6.6. Elements Shortcuts

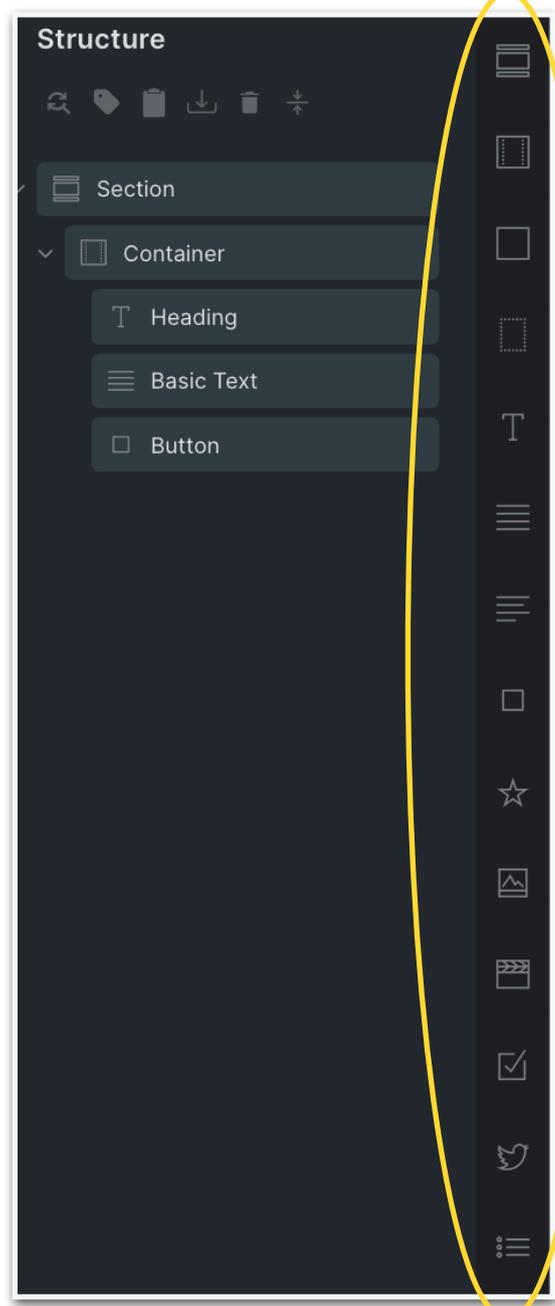
When toggling the shortcuts below the result can be found on the right-hand side of the Structure Panel. Now you can lightning-fast adding of the most used elements.



Shortcuts right-hand side of the Structure Panel.

### Hold SHIFT

*When you hold the SHIFT button and click one of the elements, there will be autofocus on that element. And as long as you hold the shift button all other elements will be nested inside the parent element.*





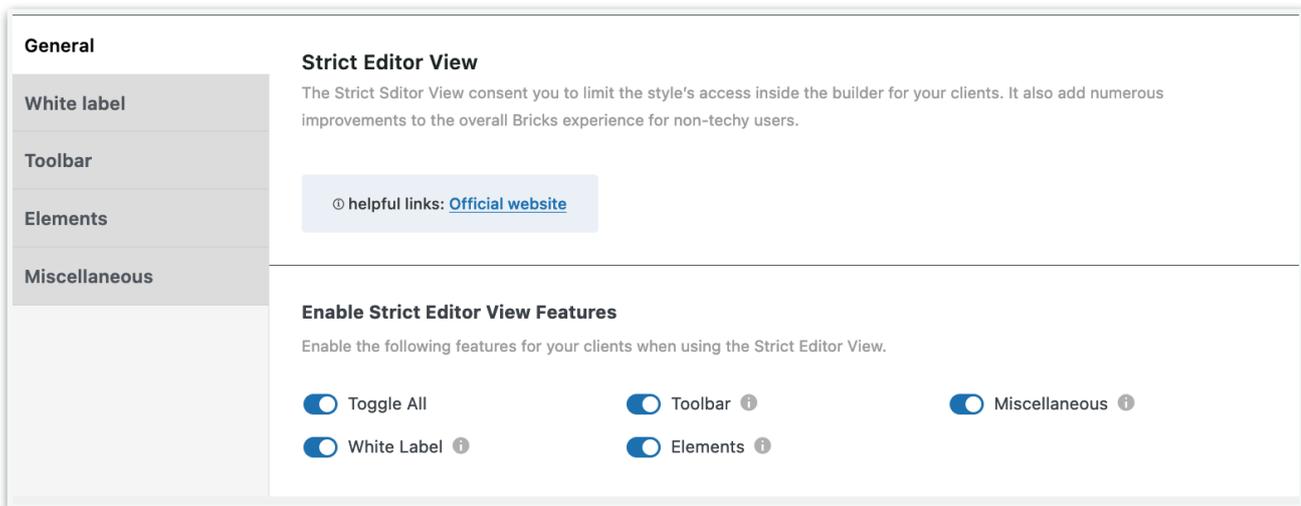
## 6.7. Keyboard Shortcuts

### Overview

General	Enable/Disable Grid Guides	CTRL + CMD + i
Topbar		
Structure Panel	Enable/Disable X-Mode	CTRL + CMD + j
Classes & Styles		
Elements	Enable/Disable Contrast Checker	CTRL + CMD + k
Keyboard Shortcuts		
	Enable/Disable Darkmode	CTRL + CMD + z
	Open the Advanced CSS Modal	CTRL + CMD + l
	Open the Resources Modal	CTRL + CMD + x
	Open the OpenAI Assistant Modal	CTRL + CMD + o
	Open the BricksLabs Modal	CTRL + CMD + n

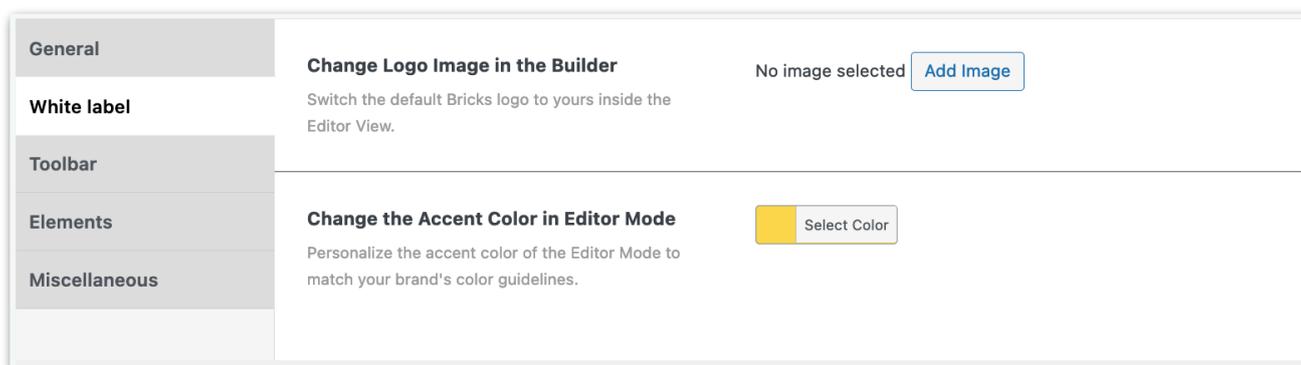
## 7. Strict Editor View

The Strict Editor View allows you to limit the style's access inside the builder for your clients. It also adds numerous improvements to the overall Bricks experience for non-tech users. Helpful links: [Official website](#)



### 7.1. Strict Editor View: White Label

You can change the Logo Image in the Builder and switch the default Bricks logo to a different one inside the Editor View.



You can also change the Accent Color in Editor Mode, and personalize the accent color of the Editor Mode to match your brand's color guidelines.

## 7.2. Strict Editor View: Toolbar

### 7.2.1. Disable Toolbar Icons

Click on any of the following icons to hide them from the Strict Editor View's Toolbar.

General	<b>Disable Toolbar Icons</b>		
White label	Click on any of the following icons to hide them from the Strict Editor View's Toolbar.		
Toolbar	<input type="checkbox"/> Toggle All	<input type="checkbox"/> Revisions	<input type="checkbox"/> Undo / Redo
Elements	<input type="checkbox"/> Logo	<input type="checkbox"/> Settings	<input type="checkbox"/> Edit with WordPress
Miscellaneous	<input type="checkbox"/> Help	<input type="checkbox"/> Breakpoints	<input type="checkbox"/> Preview
	<input type="checkbox"/> Pages	<input type="checkbox"/> Dimensions	

## 7.3. Strict Editor View: Elements

### 7.3.1. Enable the following elements on Strict Editor View

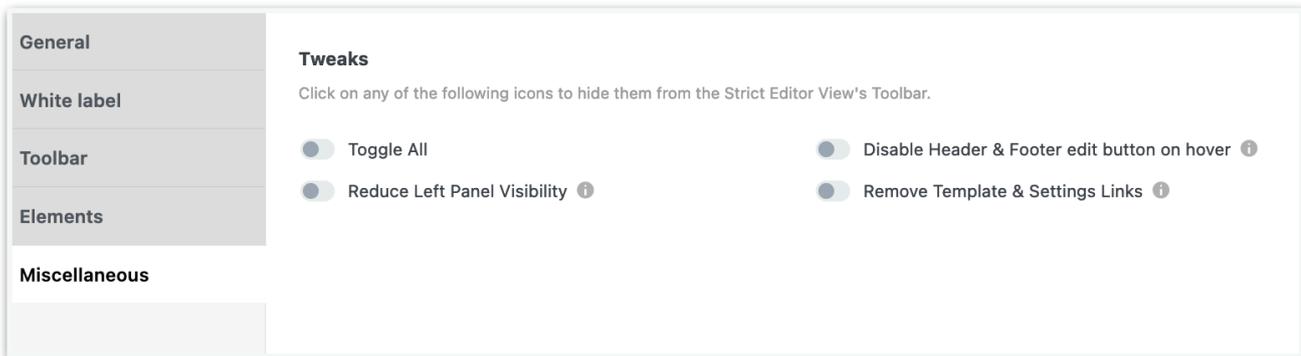
All the following checked elements will be selectable by your clients inside the editor and, thus, partially editable. All the others will be in read-only mode.

General	<b>Enable the following elements on Strict Editor View</b>		
White label	All the following checked elements will be selectable by your clients inside the editor and, thus, partially editable. All the others will be in read-only mode.		
Toolbar	<input checked="" type="checkbox"/> Toggle All	<input checked="" type="checkbox"/> Accordion	<input checked="" type="checkbox"/> Testimonials
Elements	<input checked="" type="checkbox"/> Heading	<input checked="" type="checkbox"/> Accordion nested	<input checked="" type="checkbox"/> Logo
Miscellaneous	<input checked="" type="checkbox"/> Text basic	<input checked="" type="checkbox"/> Tabs	<input checked="" type="checkbox"/> Facebook page
	<input checked="" type="checkbox"/> Text	<input checked="" type="checkbox"/> Tabs nested	<input checked="" type="checkbox"/> Image gallery
	<input checked="" type="checkbox"/> Button	<input checked="" type="checkbox"/> Animated typing	<input checked="" type="checkbox"/> Audio
	<input checked="" type="checkbox"/> Icon	<input checked="" type="checkbox"/> Countdown	<input checked="" type="checkbox"/> Carousel
	<input checked="" type="checkbox"/> Image	<input checked="" type="checkbox"/> Counter	<input checked="" type="checkbox"/> Slider
	<input checked="" type="checkbox"/> Video	<input checked="" type="checkbox"/> Pricing tables	<input checked="" type="checkbox"/> Slider nested
	<input checked="" type="checkbox"/> Icon box	<input checked="" type="checkbox"/> Progress bar	<input checked="" type="checkbox"/> Svg
	<input checked="" type="checkbox"/> Social icons	<input checked="" type="checkbox"/> Pie chart	<input checked="" type="checkbox"/> Nav menu
	<input checked="" type="checkbox"/> List	<input checked="" type="checkbox"/> Team members	

## 7.4. Strict Editor View: Miscellaneous

### 7.4.1. Tweaks

Click on any of the following icons to hide them from the Strict Editor View's Toolbar.



The screenshot shows a settings panel with a sidebar on the left containing menu items: General, White label, Toolbar, Elements, and Miscellaneous. The main content area is titled 'Tweaks' and contains the following text and options:

**Tweaks**  
Click on any of the following icons to hide them from the Strict Editor View's Toolbar.

- Toggle All
- Reduce Left Panel Visibility ⓘ
- Disable Header & Footer edit button on hover ⓘ
- Remove Template & Settings Links ⓘ

# 8. AI Integration

## AI Integration

In this section, you can enable the OpenAI integration inside the Bricks builder (by creating AI-generated text, images, codes, etc...). Make sure to insert a valid OpenAI API Key. Helpful links: [Official website](#)

The screenshot shows the 'General' settings panel for 'AI Integration'. It includes a title 'AI Integration', a descriptive paragraph, a helpful link button, and an 'OpenAI API KEY' field with instructions and a warning.

**General**

**AI Integration**  
In this section, you can enable the OpenAI integration inside the Bricks builder (create AI generated text, images, codes, etc...). Make sure to insert a valid OpenAI API Key.

🔗 helpful links: [Official website](#)

**OpenAI API KEY**  
Insert here your OpenAI API key that you can find in your [OpenAI account](#). The key will be stored in your database using a 128-bit AES encryption method.  
This field is mandatory if you plan to use the AI integration.

You can also Insert here your OpenAI API key that is in your [OpenAI account](#). The key will be stored in your database using a 128-bit AES encryption method. This field is mandatory if you plan to use AI integration. Don't forget this is a paid service.

The screenshot shows the 'OpenAI Assistant' interface. A red error banner at the top reads: 'OpenAI API returned an error with the following message: "Billing hard limit has been reached"'. Below the banner are four buttons: 'Completion / Chat', 'Edit', 'Images', and 'History'.

**OpenAI Assistant**

OpenAI API returned an error with the following message: "Billing hard limit has been reached" ✕

Completion / Chat Edit Images History

# 9. Frontend Playground

The Frontend Playground allows you to play with your global settings and colors on the front end without impacting the whole site. It also includes a ton of helpers and advanced functions to help you in your design journey.

Once enabled, navigate to any page of the website and activate it by opening the Advanced Themer menu inside the Admin Bar and clicking “Enable the Frontend Playground”.

Note that URLs that contain a query are not supported (such as <https://example.com/?bricks=run>). Helpful links: [Official website](#)

The screenshot shows the 'General' settings page for the Frontend Playground. It includes a description of the feature, instructions on how to enable it, and a note about unsupported URLs. There are three main sections: 'Frontend Playground Theme' with a dropdown menu set to 'Dark Theme', and 'Custom Post Types Permissions' with a list of post types and their corresponding toggle switches.

**General**

### Frontend Playground

The Frontend Playground allows you to play with your global settings and colors on frontend – without impacting the whole site. It also includes ton of helpers and advanced functions to help you in your design journey.

Once enabled, navigate to any page of the website and activate it by opening the Advanced Themer menu inside the Admin Bar and clicking “Enable the Frontend Playground”.

**Note that url's that contains a query are not supported (such as <https://example.com/?bricks=run>).**

© helpful links: [Official website](#)

---

**Frontend Playground Theme** Dark Theme

Choose the Theme of the Frontend Playground

---

**Custom Post Types Permissions**

Enable the Frontend Playground on specific custom post types.

- post
- page
- attachment
- announcement
- minutes
- verslag

## 9.1. Frontend Playground Theme

Choose between a light or dark theme for the Frontend Playground.

**Frontend Playground Theme**

Choose the Theme of the Frontend Playground

Dark Theme

Light Theme

## 9.2. Custom Post Type Permissions

You can enable the Frontend Playground on specific custom post types.

**Custom Post Types Permissions**

Enable the Frontend Playground on specific custom post types.

post

page

attachment

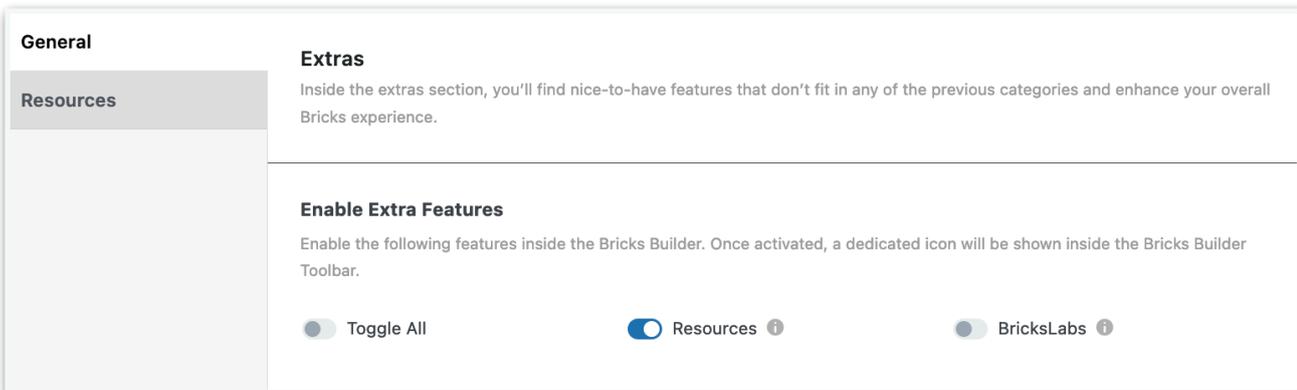
announcement

minutes

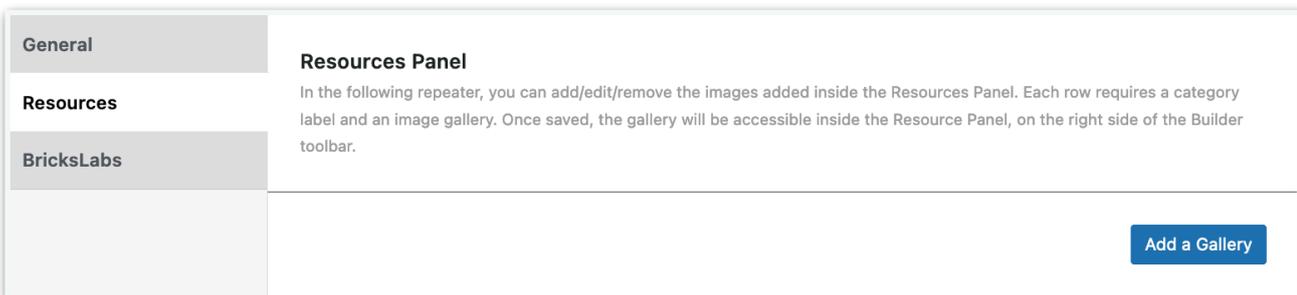
verslag

# 10. Extra's

Inside the extras section, you'll find nice-to-have features that don't fit in any of the previous categories and will enhance your overall Bricks experience.



## 10.1. Theme Settings: Resources



Here you can add, edit, and/or remove images added inside the Resources Panel. Each row requires a category label and an **image gallery**. Once saved, the gallery will be accessible inside the Resource Panel, on the right side of the Builder top toolbar.

This works flawlessly. Once a category is made, one click on the 'Add to gallery' button brings you to the media library where you can select multiple images.

Category \*

Gallery \*






Add to gallery

Bulk actions ▼

Add Row

Resources Panel

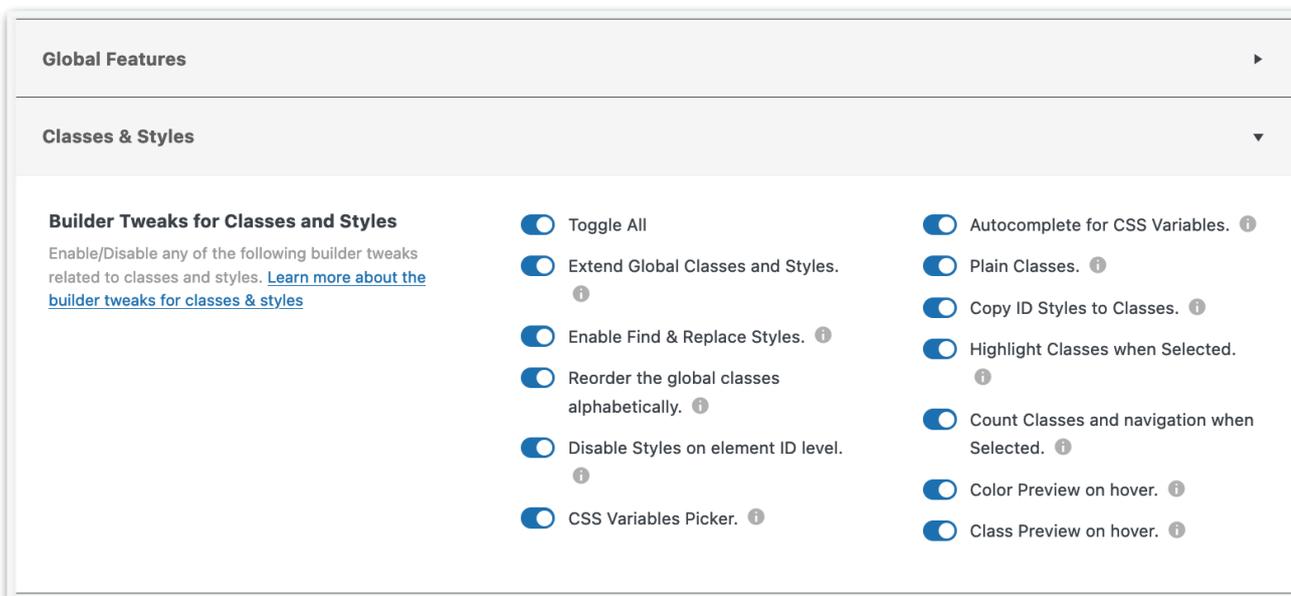
## 10.2. BricksLabs Panel

The BricksLabs feed is activated on the builder. Just click on the “lab” icon inside the builder’s top bar to see the latest articles published on Bricklabs and filter your results by any given keyword.

<b>General</b>	<b>BricksLabs Panel</b>
<b>Resources</b>	The BricksLabs feed is activated on the builder. Just click on the “lab” icon inside the builder’s topbar to see the last articles published on Bricklabs and filter your results by any given keyword.
<b>BricksLabs</b>	

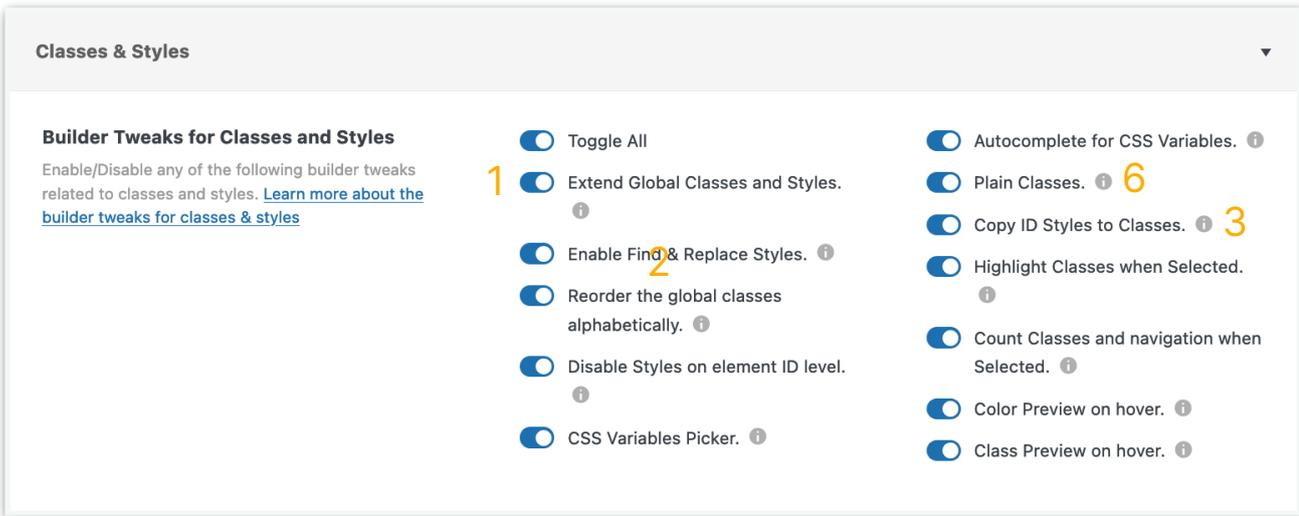
# 11. Builder Tweaks: Classes & Styles

[Learn more about the builder tweaks for classes & styles](#)



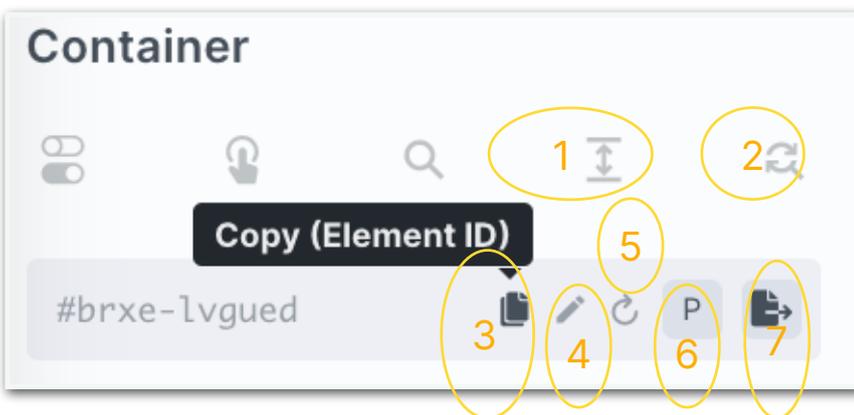
- **Toggle All:** This is self-explanatory.
- **Extend Global Classes and Styles.** This will give you the ability to extend the global classes and styles of an element to its parent or children. For instance, you start styling an element at the ID level and then discover you forgot to create a class.
- **Enable Find & Replace Styles.** With this, you can replace any style, e.g., a specific font with another font.
- **Reorder the global classes alphabetically.** When enabled all global classes will be reordered alphabetically in Bricks Builder.
- **Disable Styles on the element ID level.** Styling an element is always on the ID level unless you create a class. When you use this option you will be forced to create a class and apply the styles on it.
- **CSS Variables Picker.** When this option is checked, a new icon will pop up on the relevant style fields inside Bricks Builder. When clicked, a modal will be opened in which you can pick the desired CSS variable.

- **Autocomplete for CSS Variables.** When typing, a bottom popup will open at the bottom of each field with the list of all matching CSS variables.
- **Plain Classes.** A new icon will show up next to the element's class field. When clicked, a popup window will appear where you can type the classes you want to add or remove in bulk.
- **Copy ID Styles to Classes.** When checked, a new export icon can be seen next to the element's class field. When selected, you're able to insert a class name and put all your ID styles into it, because sometimes we start styling an element (on the ID level) without making a class, and when we find out we have to undo all the changes at the element ID level, create a class and redo the styling. But not anymore with this tool.
- **Highlight Classes when Selected.** When checked, a blue outline will appear on all elements that share the same class when selected inside the builder. Ideal to localize where your classes are applied, and when you make changes they will be applied to all elements with the same class.
- **Count Classes and navigation when Selected.** A new counter will show up next to the class name that indicates the number of times the class is used on the page. Clicking on the counter will scroll the page to each element that is using the active class.
- **Color Preview on hover.** When checked and the color grid of an element is open, hovering on a color will temporarily apply the color to the element, providing a nice preview.
- **Class Preview on hover.** When this option is checked, and the class dropdown of an element is open, hovering on each class will temporarily apply that class to the element, providing a nice preview of the impact of a class on the element.



When everything is toggled “ON” the following changes will take place in the workspace. See the screenshots below.

Changes in the left menu bar of the Bricks workspace.

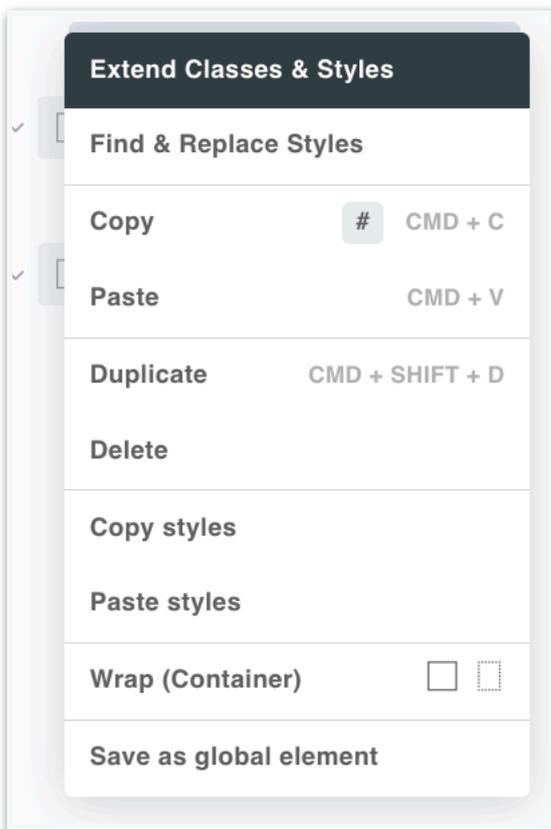


Left menu changes

### 1: Extend Global Classes and Styles

Ok, so you styled this one element, and you put it in a class. What if you have many of the same elements? Do you have to give everyone the same class? No, not with this toggle.

Right-click on the element (see screenshot left), and choose Extend Classes & Styles or choose button ‘1’ in the left menu (see fig. Left menu changes).



You'll be presented with the choices in the default screen below, i.e. Classes or Styles.

Choose Classes and determine whether it must be applied inside the same parent's Container, Section, or the whole page.

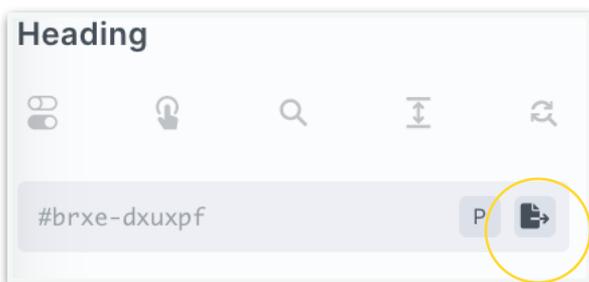
A screenshot of a dialog box titled "Extend Classes & Styles". It contains the following elements:

- "I want to extend the:" followed by two buttons: "Global Class(es)" (highlighted with a red border) and "Style(s)".
- "To the following elements category" followed by a dropdown menu showing "heading".
- "That are positioned:" followed by seven buttons: "On the same DOM level (Siblings)", "Inside the same parent's DIV", "Inside the same parent's Block", "Inside the same parent's Container" (highlighted with a red border), "Inside the same parent's Section", and "On the whole Page".
- At the bottom right, "Cancel" and "Extend" buttons.

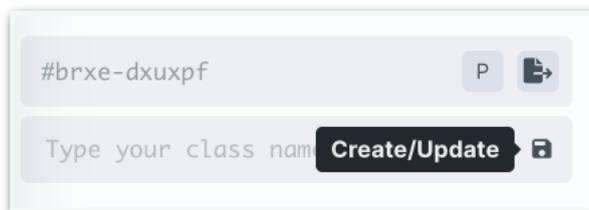
Since I already had selected the heading that is what you'll see in the elements category.

- 2: Find and replace style
- 3: Copy ID styles to classes
- 4: Rename the element ID
- 5: Reset the styles
- 6: Plain classes

## 7: Export the styles to a class



With this button, you can export the styles to a class. It will open a text field to name the class.



After giving it a name and pressing 'Create/Update', the styling at the ID level is gone and has moved to the new class.

## 12. The Enhanced Top Bar(s) Explained

### 12.1. Enhanced Left Top Bar

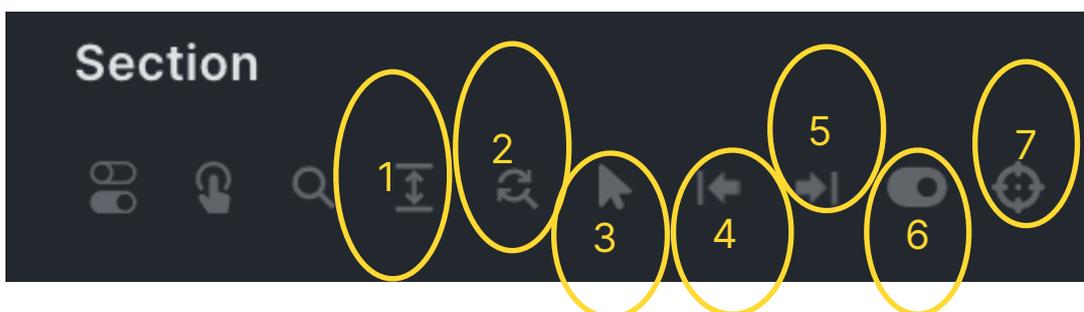
Bricks Builder has multiple sections in the top bar, which we will handle one by one. Let's start on the left with the Enhanced Top Left Bar. We'll see a number of new icons/buttons.



1. Grid guides (ctrl+cmd+i)
2. X-mode (ctrl+cmd+j)
3. Contrast Checker (ctrl+cmd+k)
4. Dark Mode (ctrl+cmd+z)
5. Advanced CSS (ctrl+cmd+l)
6. Pseudo States: :hover, :active, :focus, :before, :after

### 12.2. Element Top Bar

The top bar of the individual elements is enhanced with shortcuts as well. In this case, it is a Section, which is a parent element.

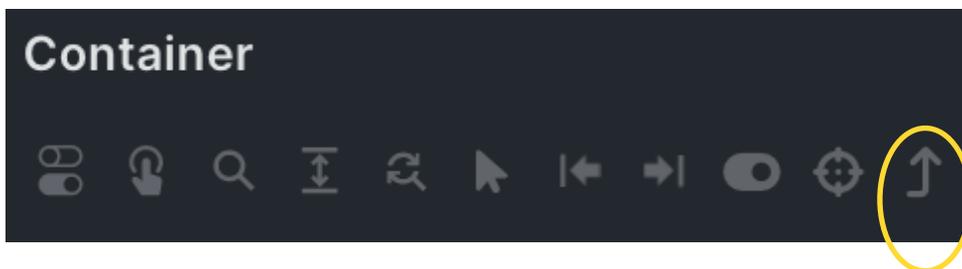


1. Extend Classes & Styles
2. Find and Replace Styles

3. :Hover
4. :Before
5. :After
6. :Active
7. :Focus

### 12.3. Child Element Top Bar

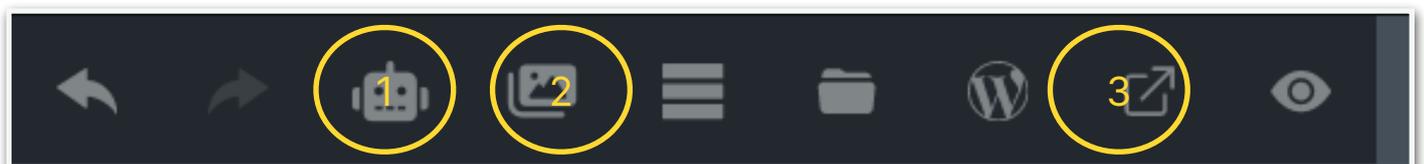
When we change to a child element another extra icon is added, “Go to Parent Element”.



## 13. The Enhanced Right Top Bar(s)

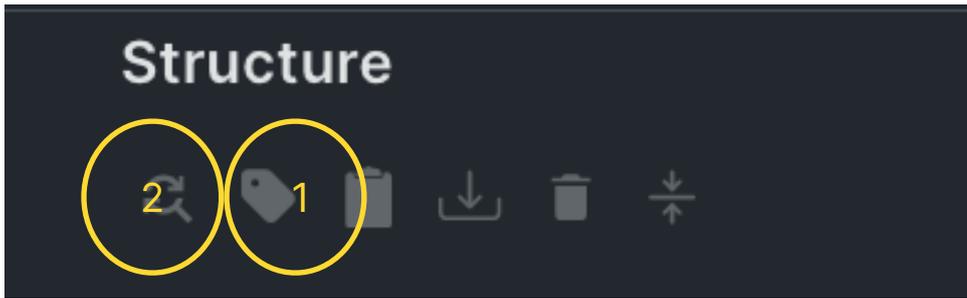
---

When we switch over to the enhanced right Top Bar we'll find three additional icons.

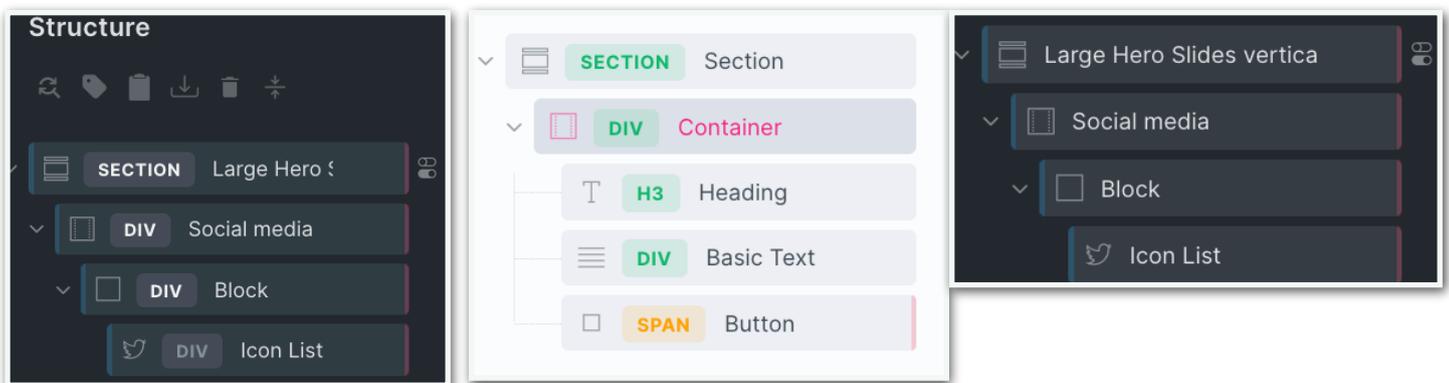


1. Open AI assistance (ctrl+cmd+o)
2. Resources (ctrl+cmd+x)
3. View on the front end (directly)

The Structure bar is also enhanced with two more icons



1. **Show Elements Tag.** Clicking this icon will cycle you through the black & grey version with tags, or the colored version, or the original Bricks version.



When you click your pointer on the “TAG” name, there will be a dropdown with options to make changes to the tag. Any choice made here will be immediately reflected on the other panel on the left.

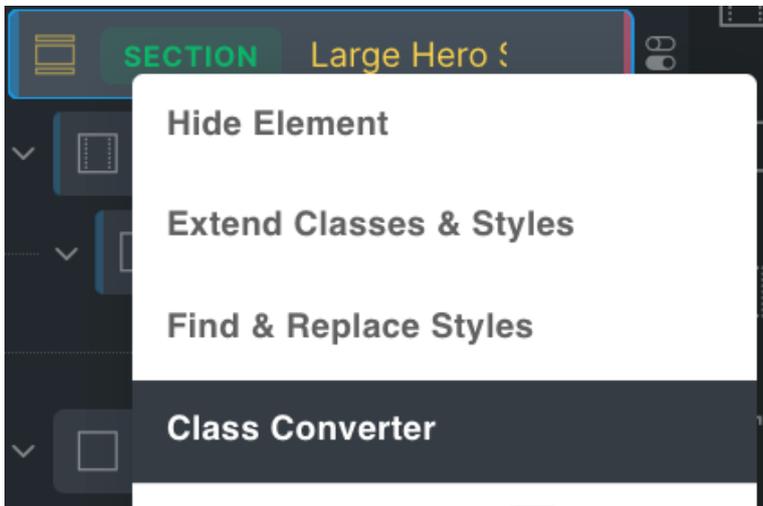


So, if you choose to do so, you can change the section into a Div or whatever you want. As an example, the size of a Heading will also show, click on it and you can make the change immediately from a default H3 to any other size. Or if you have basic text shown as a div, you can change it to a paragraph (see picture to the left).

When using the colored Tags, you may notice 3 different colors. Green means the Tag can be changed as described above. Orange means you can make a custom change manually in the left panel. And Red means you can't change.

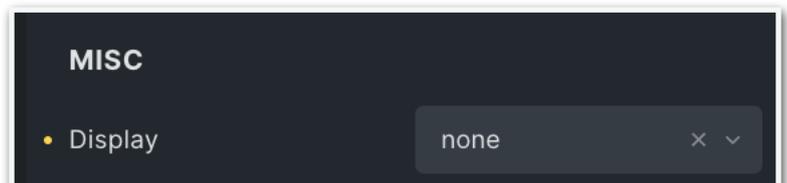
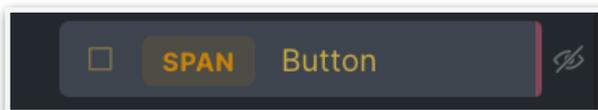
## 13.1. The Enhanced Context Menu

Right-click the mouse on one of the elements in the Structure Panel and four new options are shown.



### 1. Hide Element

When you choose “Hide Element” an icon will show right from the Element in the Structure Panel.



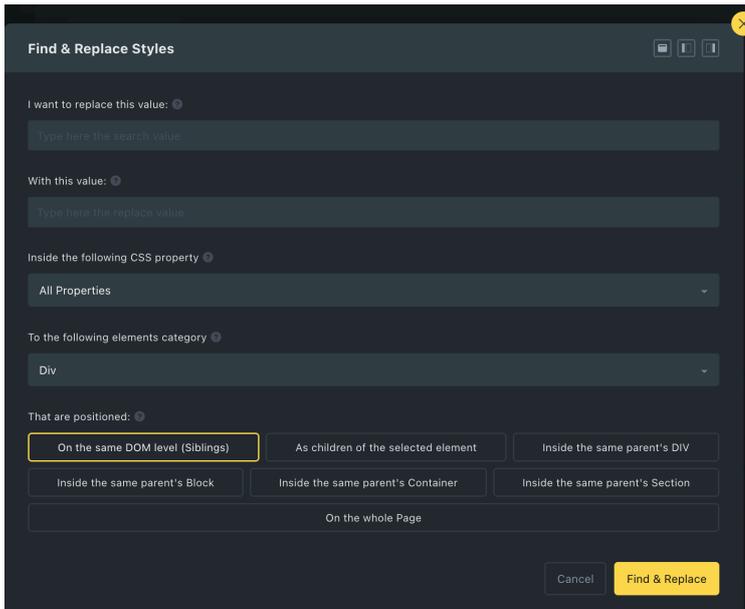
In the Style Panel - Miscellaneous, Bricks will change the display to None.

To undo simply click the ‘hide’ icon.

### 2. Extend Classes & Styles

This will give you the ability to extend the global classes and styles of an element to its parent or children. For instance, you start styling an element at the ID level and then discover you forgot to create a class.

### 3. Find & Replace Styles

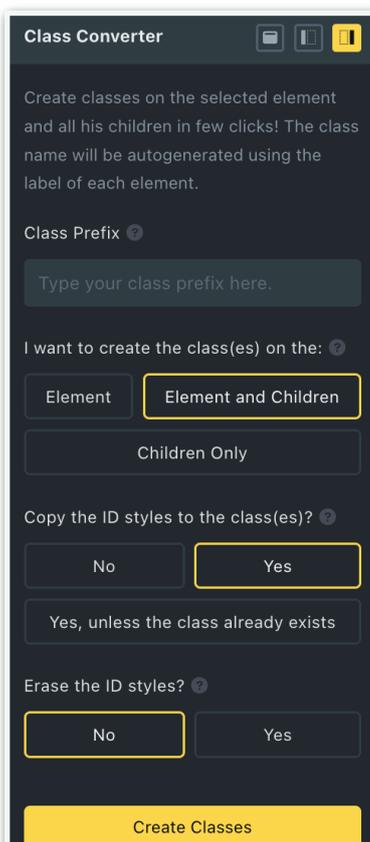


By clicking on the “Find & Replace Styles” option a modal will show, either as a left sidebar, a right sidebar, or a full-screen modal.

### 4. Class Converter

By clicking on the Class Converter option a modal will show, either as a left sidebar, a right sidebar, or a full-screen modal.

The aim is to remove the classes on the ID level and bring them to the parent level as this is good practice.

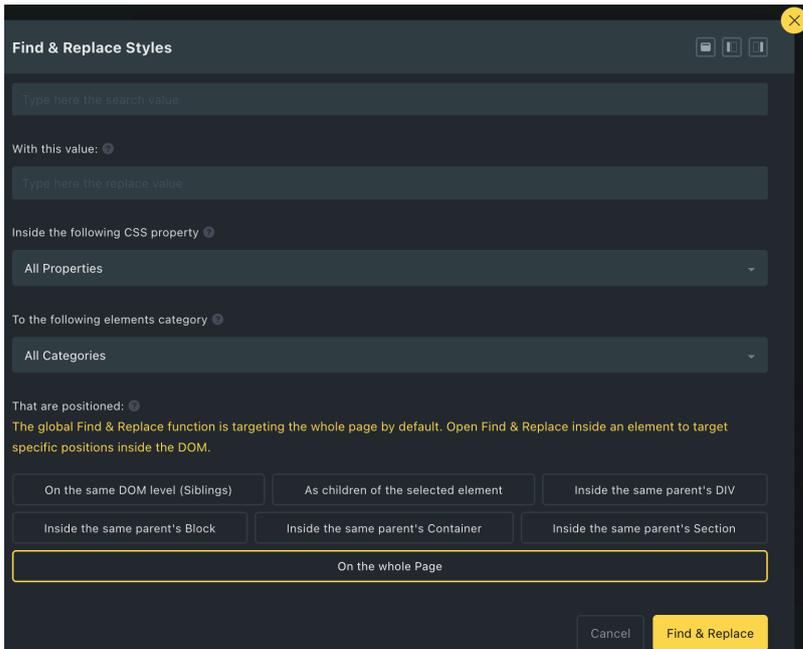


Create classes on the selected element and all its children in just a few clicks! The class name will be autogenerated using the label of each element.

Use the Class Prefix e.g. when you use the BEM naming or any other framework naming convention. The other options are quite self-explanatory.

Do you want to create the class(es) on:

- 1.The Element itself, or the
- 2.Element and Children, or the
- 3.Children Only



So, you want to create the class on the Element and Children. You want to copy the ID styles to the class (unless the class already exists) and erase the ID styles. Then press the button “Create Classes”.

All styles on the ID level are now gone, and in the left panel, you’ll find the new class with the Prefix you’ve chosen.

The fun part is that your new classes will take the name that you gave the elements, e.g. if you named your text “Description”, that will also become the class name.

# 14. Miscellaneous

## 14.1. Breakpoint indicators

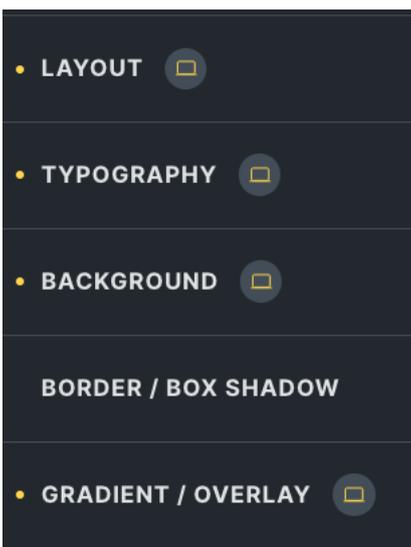
Breakpoint indicators can be found in both the left Style Panel as well as in the Top Bar. When relevant multiple indicators are shown and when you click a specific one (e.g. the mobile) the work screen shows immediately the relevant size. See screenshots of the indicators below



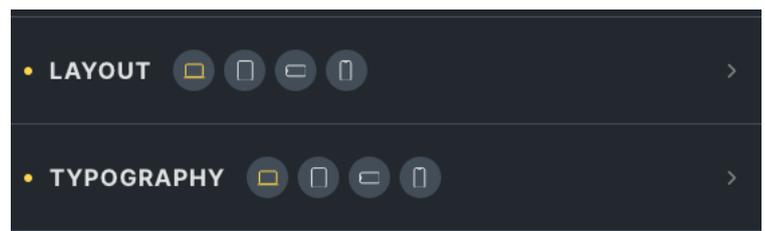
Single Indicator Top Bar



Multiple Indicators Top Bar



Single Indicator Style Panel



Multiple Indicators Style Panel

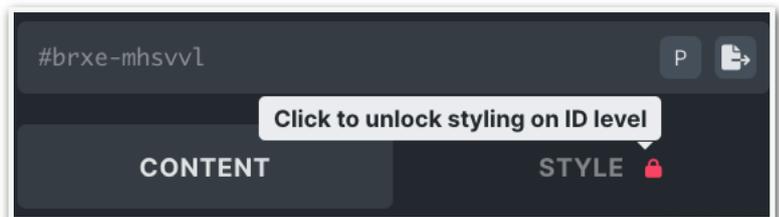
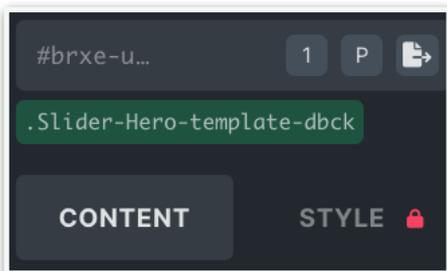
In the Style Panel screenshot you can see by the yellow dot changes have been made to the layout group panel, and you can see the changes are relevant only to the desktop. In the Top Bar screenshot, you can see changes have been made at each breakpoint level.

## 14.2. Indicators of styles inherited from a class

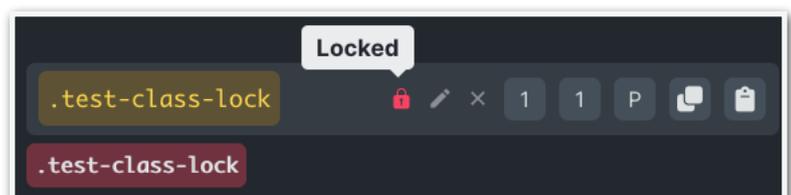
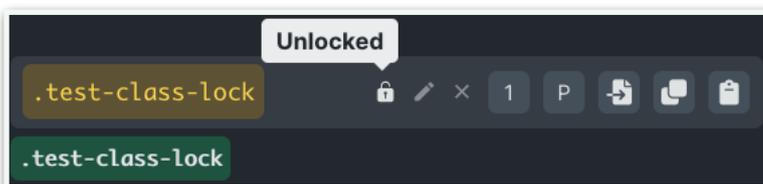
A blue dot instead of an orange dot indicates that a Class already has styled an element at the ID level.

## 14.3. Locked Class Indicator

Classes have different colors too. Green means changes can be made, a red one means it is locked and you have to unlock it first in order to make changes.

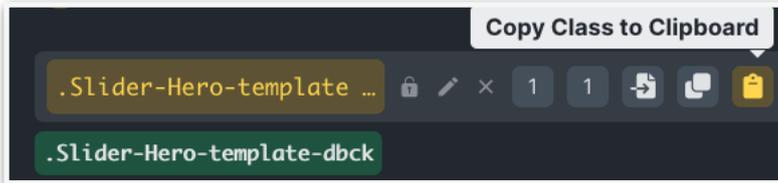


When I make a new class it is green by default. But I can lock it at will. See the two screenshots below.



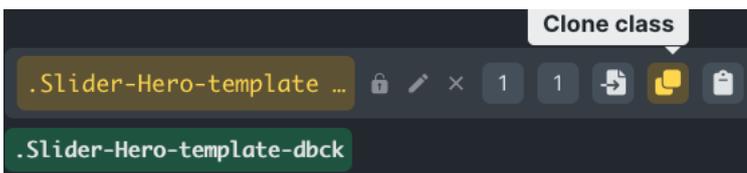
## 14.4. Clone Class and Copy Class to Clipboard

Sometimes you really want to copy a certain class and apply elsewhere. Until now there was no quick way of doing that. Now, there is a new item. The icon at the very right of the Class builder.

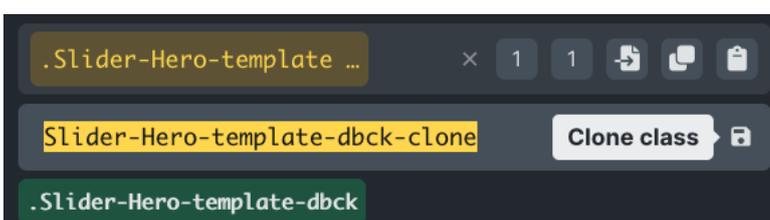
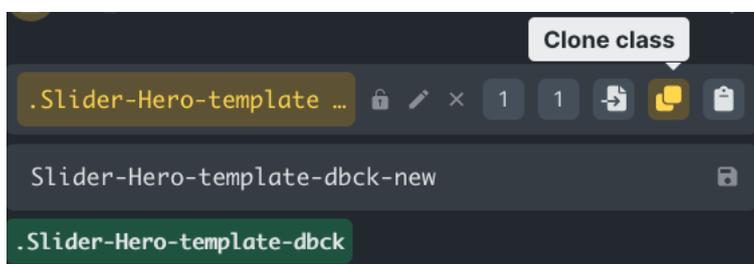


When you click that icon the Class is copied to the clipboard, and then when you create a new Class the content of the clipboard is immediately pasted in the new Class the moment you hit the return key.

The Clone Class idea comes from the Facebook group.

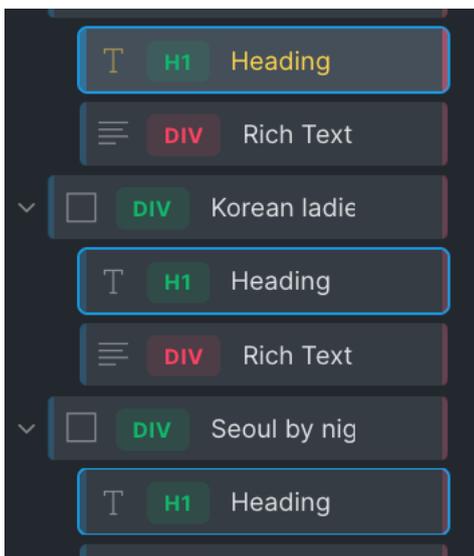


Sometimes you may have a very nice Class with many styles you want to use in other elements as well, maybe with just a few tweaks or changing a few parameters. As you can see in the below screenshot the name of the 'old' class is used and you can change the name to your liking or need.



When you click the button Clone Class the process is done and you can tweak the desired changes in the Style Panel.

## 14.5. Elements that share the same Class are highlighted



Video 44:48