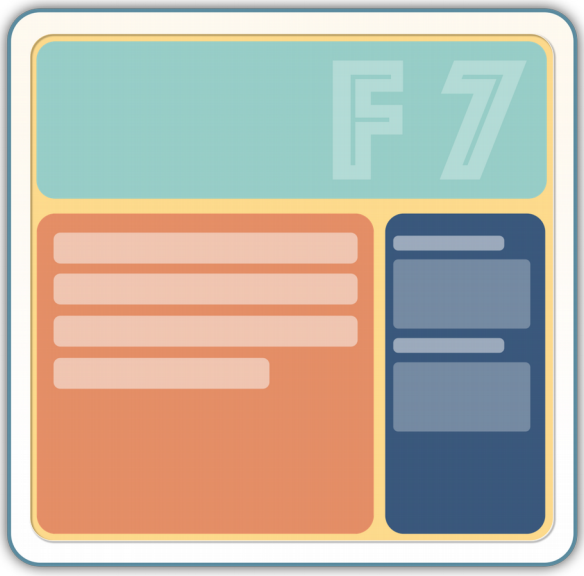
Flux 7 Manual



The Escapers

Contents

Contents 2
Flux 7 7
Using this Manual 7
Document Standards 7
Flux 7 Reference 8
Getting Started 8
Creating a new website 8
Editing an existing website 9
Creating a new website using a Template 9
Workflow 10
The Flux User Interface 10
Suggested Techniques 11
Site Manager 15
Navigating 15
Opening files 15
Creating new files 15
Copying files into your website 15
Finding files 15
Page Editor 16
Opening the Page Editor 16
Understanding the Page Editor 16
What is being edited? 19
CSS 22
External Styles 22

Inline Styles 27
Inline Vs External Styles 29
Converting Inline Styles to External Styles 30
Advanced CSS Selectors 32
CSS Reset 32
ID Selectors 32
Class Selectors 32
Tag Selectors 33
Pseudo Selectors 33
Descendant Selectors 33
Child Selectors 33
Project Settings 34
Project Metadata 34
Code Editing 35
Toggling the Code Editor 35
Making changes 35
Code Context 36
Text Content Editing 37
Content Editor 37
Inline Editing 38
Element Library 39
Finding an Element in the Library 39
Inserting an Element from the Library 39
Advanced Insertion 39
Inserting Text Elements 40
Inserting Elements from the menu 40
Keyboard Shortcuts 40

Navigator 41
Navigating 41
Editing Code Contents 41
Understanding the hierarchy 41
Contextual Toolbar 44
Inline Vs External Styles 44
Mini Navigator 45
Overlays 47
What are Overlays? 47
Toggling Overlays 48
Swatches 49
Importing Swatches 49
Selecting Swatches 49
Rollback 50
Enabling Rollback 50
Using Rollback 50
Google Fonts 52
Importing a Font 52
Using the font 53
Removing the font 54
Licence 54
YouTube 55
Get the YouTube link 55
Paste in the Code 55
Restrictions 56
Validator 57
Master Elements 58

Introduction ... 58
Creating a Master Element definition ... 58
Using a Master Element ... 58
Updating Master Elements ... 59
Snippets ... 60
Using Snippets ... 60
Adding Snippets ... 60
Exporting Snippets ... 61
Importing Snippets ... 61
Images ... 62
Adding an image to your page ... 62
Adding an image from the clipboard ... 62
Changing the Image path ... 63
Scaling the image ... 64
Image formats ... 64
Image Maps ... 65
Creating an image map ... 65
Editing Polygon Areas ... 66
Easily add an Image Map ... 67
Links ... 68
Creating a link on an element ... 68
Creating a link on a text selection ... 68
The Link Dialog ... 69
Google Adsense ... 70
Inserting the code ... 70
Accessibility ... 71
The Image 'alt' attribute ... 71

Code Order ... 71
Flexible Layouts ... 71
DOCTYPEs ... 72
Example DOCTYPE ... 72
SASS ... 73
Creating a new SCSS file ... 73
Adding a SASS file to your page ... 73
Editing SASS ... 74
HTML E-Mail ... 75
Responsive Design ... 77
Remote Working ... 86
Setting up a Site for Remote Working ... 86
Opening a Site for Remote Working ... 86
Creating a bookmark ... 87
Understanding Remote Working ... 87
Backup ... 87
Solving problems ... 88
Preferences ... 89
General ... 89
Browsers ... 90
Bookmarks ... 90
Tips ... 91
Make Flux run quicker ... 91
Troubleshooting ... 91

Flux 7

Using this Manual

Flux is a powerful application, to get the most out of it, you'll need this manual for reference. If there is a topic that is not covered, please let us know.

This manual covers the use of Flux, and some HTML/CSS concepts, if you are not familiar with HTML and CSS, you may also benefit from a book on this subject too.

Document Standards

This manual uses consistent formatting to assist the user.

Flux features are formatted like this ***Site Manager***

Page Elements are formatted like this **DIV**

CSS Properties are formatted like this color

Buttons in the user interface are formatted like this *OK*

Menus in the user interface are formatted like this **File**

Flux 7 Reference

Getting Started

Learn Flux 7 from the very beginning, it is best to start here if you have never used Flux before, or used a previous version.

Creating a new website

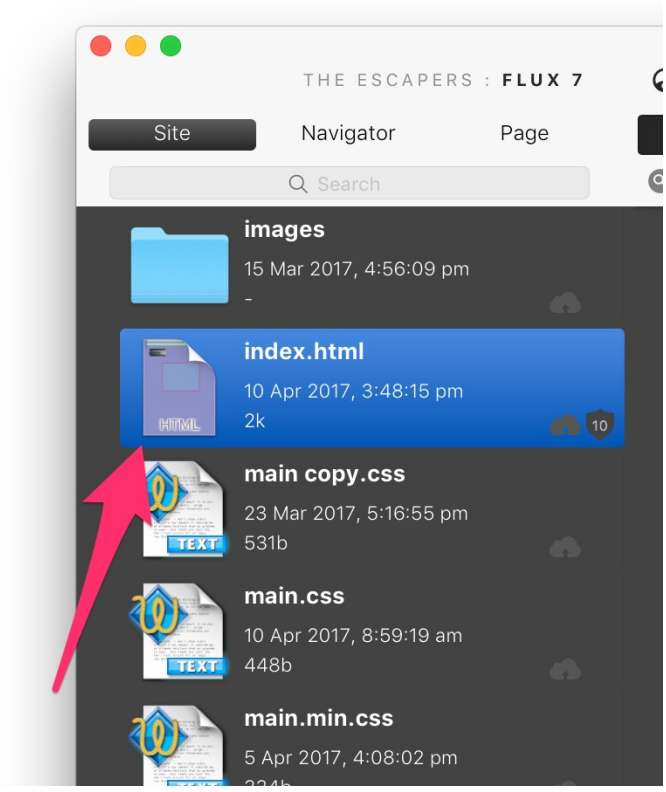
Creating a new site from scratch

Open Flux, and choose **New** from the **File** menu.

Choose a destination and name for your site, you can also add any Finder Tags here.

You will be asked if you want Flux to create a few starter files. For most sites, choose *Yes*, but you can erase or create these files later on if you want.

The new site is now displayed in the ***Site Manager*** on the left side of the window. To start editing, double click the index.html file.



The Escapers recommends you always work on a copy of site, and keep regular backups.

Editing an existing website

Method 1 : If you have the website on your Mac, then simply drag the website folder from the Finder onto the Flux icon on the Dock (or the Finder). Flux will open the site, and display the contained files in the Site Manager on the left side of the window.

Method 2 : You can also open a HTML file in Flux by dragging the HTML file onto the Flux icon in the Dock (or the Finder). Flux will ask you to choose the root folder of the website, once you have done this, Flux will open the HTML file that was dragged.

Flux can open almost any kind of website, including HTML templates downloaded from sites such as www.themeforest.com

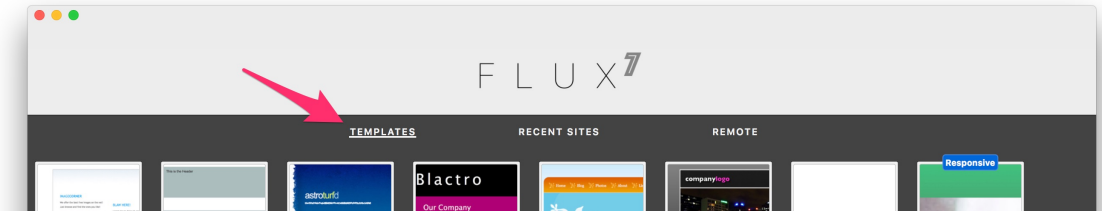
If you find a site that Flux cannot open, please contact us@theescapers.com and we will try our best to help.

The Escapers recommends you always work on a copy of site, and keep regular backups.

Creating a new website using a Template

Open Flux, if the Welcome panel doesn't show automatically, choose **New** from the **File** menu.

Click *Templates* at the top of the Welcome panel.



Double click the Template you want to use. Note, in Flux, Templates are just a starting point, you still have complete control over your website.

Choose a destination and name for your site, you can also add any Finder Tags here.

Your new site has now been created, with the assets from the Template added. To start editing, double click any .html file.

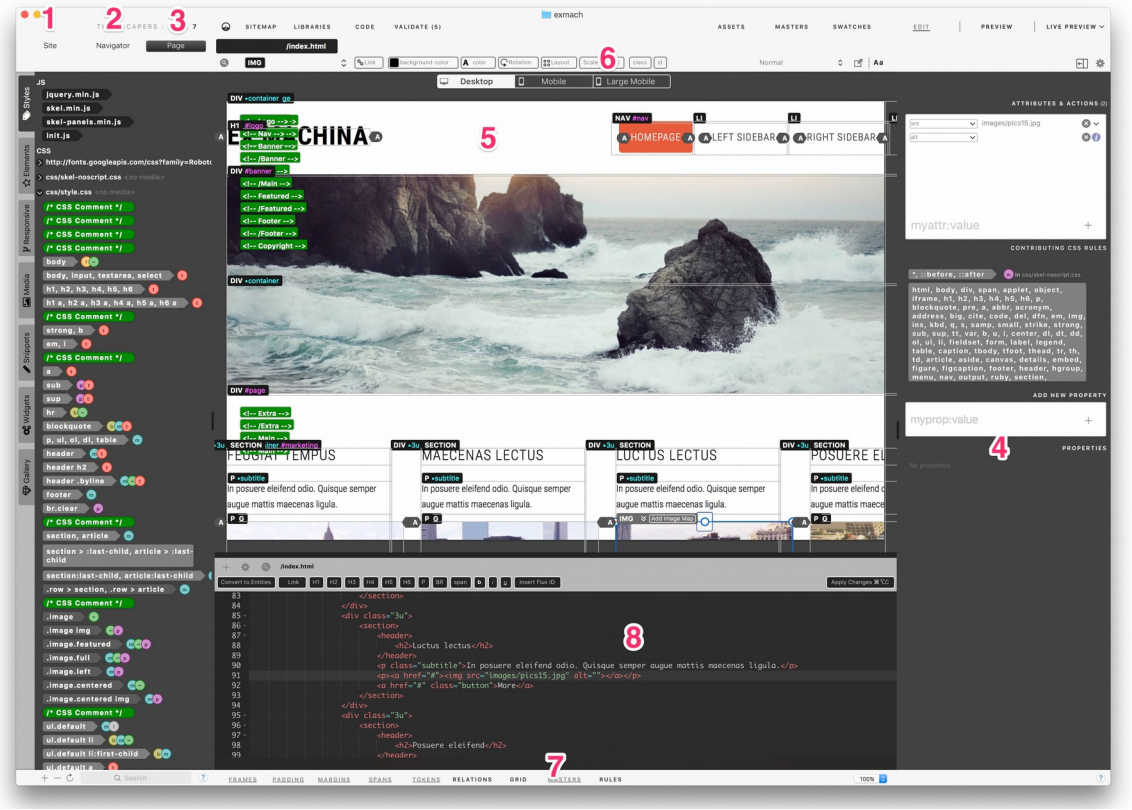
The Escapers recommends you always work on a copy of site, and keep regular backups.

Workflow

Flux 7 is powerful application and to get the most out of it, The Escapers recommends you read this manual, and understand the concepts behind Flux, HTML and CSS.

The Flux User Interface

The best way to discover the user interface is to explore it, create a new site, or create one from a template, and start playing with the interface, it's a new site, so it doesn't matter if you mess anything up!



1. The ***Site Manager***, this is where the assets (files) of your site are managed.

2. The ***Navigator***, this displays the hierarchal tree of the current web page.

3. The ***Page*** tab manages the current page, including that are styles attached to your page, and adding new elements to the page.

4. The ***Inspector*** is where the parameters of all objects are adjusted, such as HTML elements and CSS rules.

5. The ***Page Editor*** is where objects can be visually manipulated and the page layout displayed.

6. The ***Contextual Toolbar*** contains tools that are specific to the currently selected object.

7. The ***Overlays Toolbar*** controls what overlays are seen on the page.

8. The ***Code Editor*** displays the code of the selected element or rule.

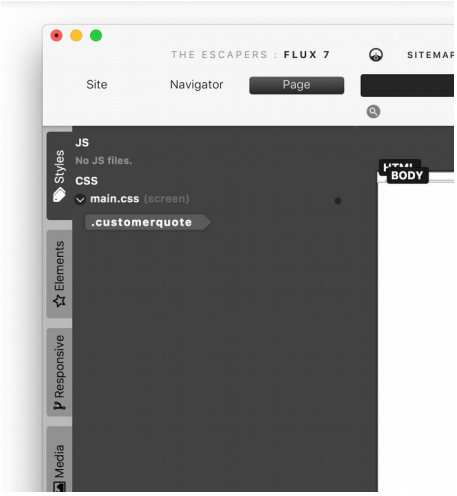
Suggested Techniques

To get the most out of Flux, and HTML/CSS, it's suggested you use the following techniques.

Use External Styles

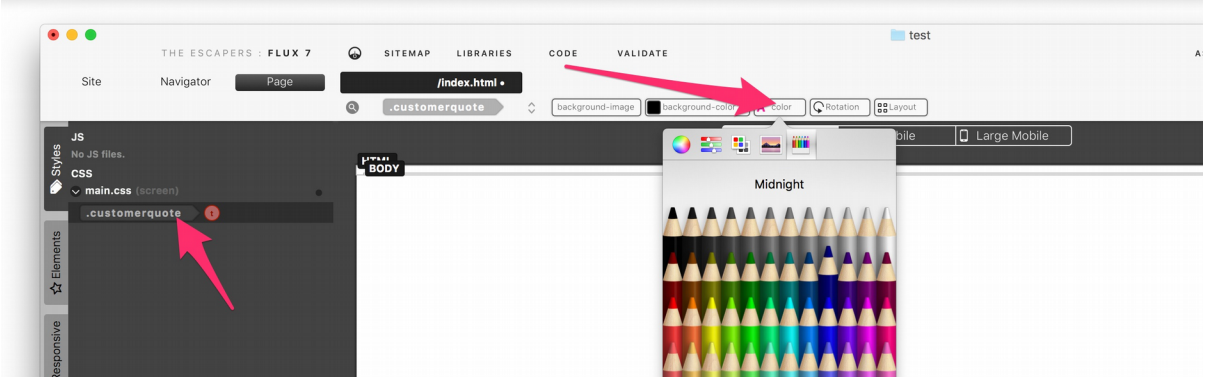
When building a site, or a page, consider what elements will follow the same style, for example, do you want all customer quotes to appear in the same font, colour? Yes, you probably do.

So, rather than creating all the customer quotes individually, and styling them individually, you should create a new CSS style...



Note, the style is preceded by a '.', this indicates that it's a class style, and can be used more than once in in a page.

Once the style has been created, we can now add properties to it, for example, the color.



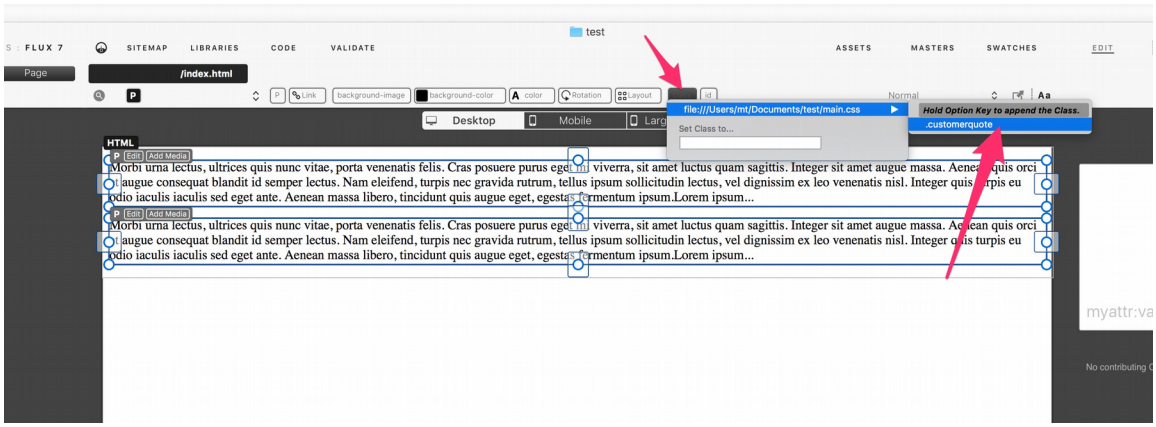
We won't see any changes on the page, why? Because we haven't actually used the class on any element, lets make a few customer quotes.

Click on the parent element that you want to contain the customer quotes, and add a Paragraph P from ***Element Library***. Hold option when doing this to add sample text content.

Reselect the parent element, and insert another Paragraph P.

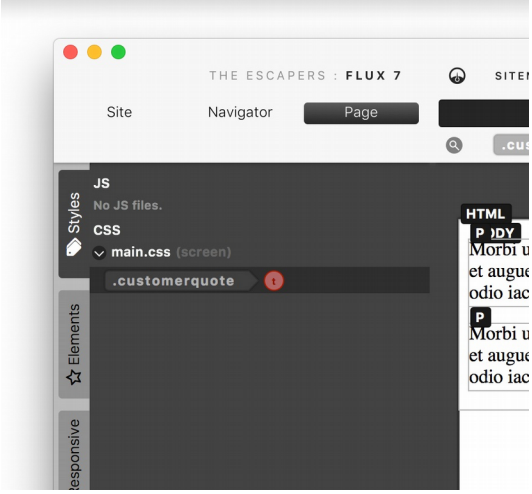
For the purposes of this example, you don't need to change the text, but you can if you want.

To set the class of the two quotes, select both P elements (hold Shift to select more than one element.), then choose the class from the ***Contextual Toolbar***.

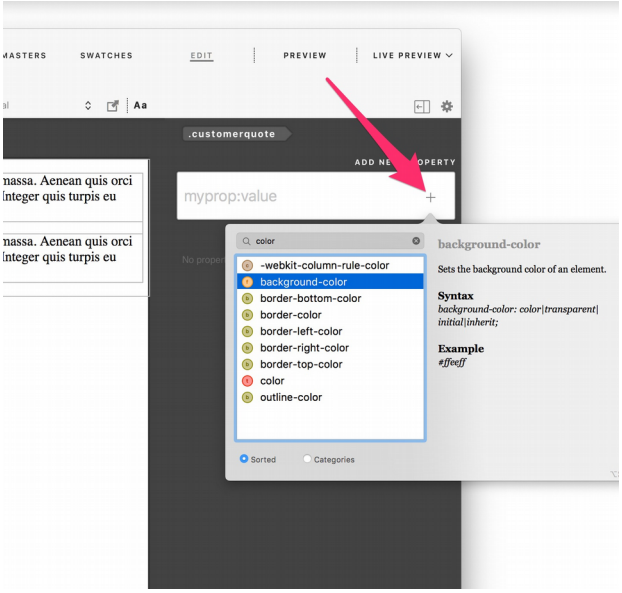


Now that the class has been applied to the elements, any change we make to the class will be shown on those elements.

Select the CSS style, and we'll make some changes.

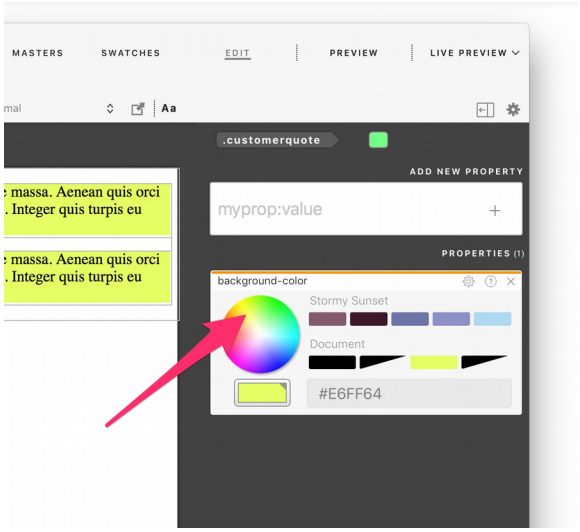


We will add the background-color property, which adjusts the color of the background of every element it applies to.



We see no change, that's because the default value of the background-color property is transparent, once we change this, we'll see the elements change color.

If you want to change the text color, then use the color property.



It's important to note that these changes are not being made inline, but to an external style, which is considered best practice. Using this method, if you make a new page, you can reuse the style, saving a lot of work, and ensuring consistency across pages.

Use Valid Code

Try to keep your code valid, it's not always possible, you may need to use a 3rd party plugin or hack, but generally speaking, it's easy to keep your code valid, using the ***Validator*** to keep a check that the changes you are making are OK.

If you use valid HTML5, every modern browser should render it correctly.

Site Manager

The ***Site Manager*** is where the files that make up a website (assets) are managed in Flux. From the ***Site Manager***, you can create, open, edit, and delete website assets.

Navigating

When a folder is double-clicked, the folder will be opened, and become the current folder in the ***Site Manager***. To return to the previous folder, press the left pointing arrow at the bottom left of the ***Site Manager***.

Opening files

To open a file, simply double-click it. You can also right-click (or Control-Click) the file to open the contextual menu. From here, you can open the file with an external editor in Flux, or another application installed on your Mac.

Creating new files

To create a new HTML file, CSS file, folder or other files, press the '+' button at the bottom of the ***Site Manager***. Double Click the type of file you want to create, and give it a name. The new file will be created in the current folder in the Site Manager.

You can also add files to your site from any other application, including the **Finder**. The ***Site Manager*** will update automatically to reflect these changes.

Copying files into your website

Files such as images can be copied by simply dragging them from the **Finder** to the ***Site Manager*** or copying the files into the correct folder using the Finder. Flux will automatically update to reflect changes made outside of Flux.

Finding files

Method 1 : At the top of the ***Site Manager*** there is a search field. You can enter a search string into this field to use Spotlight to find files.

Method 2 : For advanced Find & Replace, you can press the 'Spyglass' icon at the bottom of the *Site Manager*. Using this feature you can Find files using a variety of methods. This panel allows Find & Replace over your entire site, so take great care when using the 'Replace' feature.

Page Editor

The ***Page Editor*** is central to Flux, it's where your pages are designed, edited, if you are a beginner to web design, you'll use the ***Page Editor*** more than any other area of Flux.

Opening the Page Editor

The ***Page Editor*** is the default editor for HTML files, so simply doubling clicking a .html files in the ***Site Manager*** will open the Page Editor.

Understanding the Page Editor

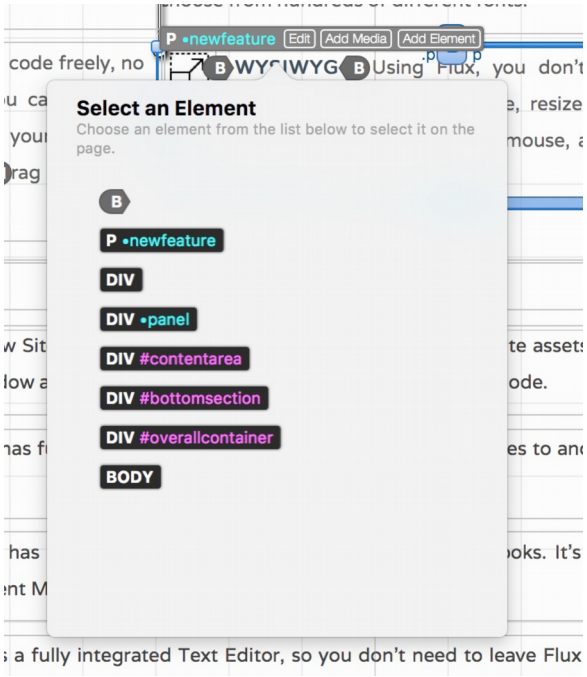
The ***Page Editor*** displays, and allows editing of your web pages, once you have opening a HTML file in the editor, you can edit text, elements, and the structure of your page.

Selecting Elements

Flux 7 has a number of ways to select an element.

The first, and simplest, is to simply click on it. On a simple webpage, with no overlapping elements, this will usually be the only type of selection you will ever need to make.

Where there are overlapping elements, it can be easier to click & hold on the element, and you will be presented with a list of the elements under the mouse pointer, you can then choose which element to select.



Adding Elements

Elements are added using the ***Element Library***, this feature is covered in more depth in a dedicated section in this manual.

You can also add elements using copy and paste, and dragging images from the Site Manager.

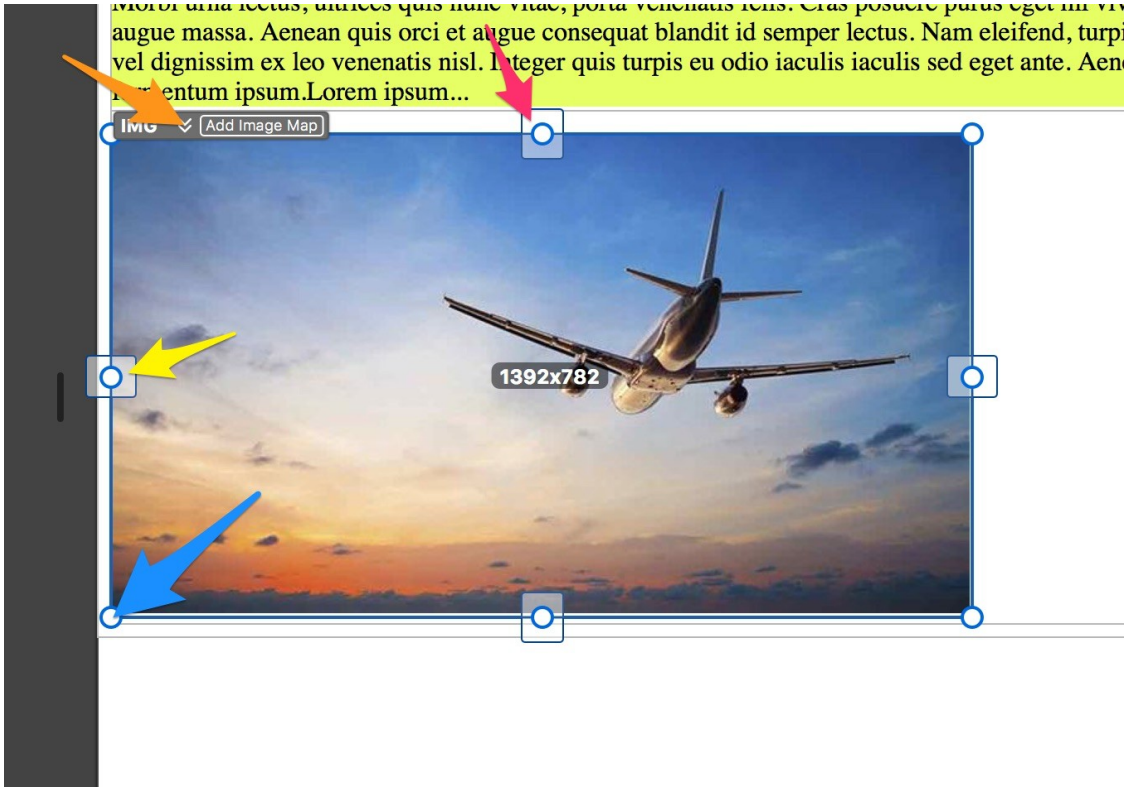
Before adding an element, using any of the above methods, ensure you have selected the correct element to insert into.

Deleting Elements

Select the element you want to delete and hit the Delete key. Remember, a webpage is a 'tree' like structure, if you delete a parent element, it's children will also be deleted.

Editing Elements

When an element on the page is selected, there are number of controls available for it's manipulation.



Margin handles, the handles on the outside of the element are for manipulating the CSS Margin properties. For example, dragging the handle on the

right side of the element will adjust the CSS property. Note, if the element does not have the adjusted property you will be prompted to create it.

If the property is inline, then the change will also be applied inline. If the property is within an external CSS file, then the external CSS style will be modified.

Padding handles, the handles on the inside of the element are for manipulating the CSS Padding properties. For example, dragging the handle on the right side of the element will adjust the CSS property. Note, if the element does not have the adjusted property you will be prompted to create it.

If the property is inline, then the change will also be applied inline. If the property is within an external CSS file, then the external CSS style will be modified.

Resize handles, the handles on the corners and centre of the element borders, are for manipulating the CSS size properties, i.e. top/left/width/height. For example, dragging the handle on the right side of the element will adjust the CSS property. If the left most handle is dragged, then the and properties will be adjusted to create the desired change.

Note, if the element does not have the adjusted property you will be prompted to create it.

If the property is inline, then the change will also be applied inline. If the property is within an external CSS file, then the external CSS style will be modified.

The element tag will display the type of the element, and any class or ID applied to it.

If the element can contain text content, then it will also display a 'gear' button which provides access to the ***Content Editor***, for editing text content.

What is being edited?

HTML and CSS are complicated. For example, if you have a paragraph, with a red background, there are numerous ways that this can be achieved, and there isn't necessarily a correct or wrong way to do it. The change could be inline, i.e.

red text

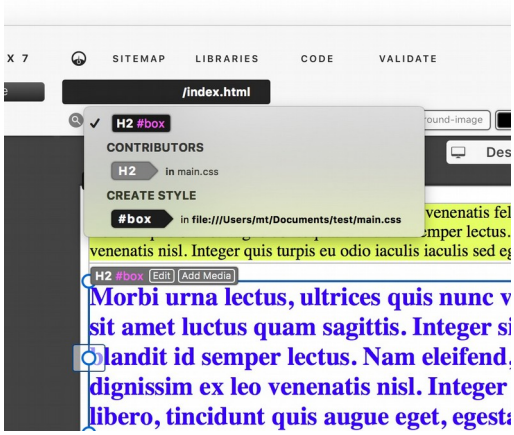
Or it could be in a CSS style called 'P', or it could be referenced by an *id* or *class* attribute.

It's a critically important part of HTML and CSS, and thus critically important to Flux.

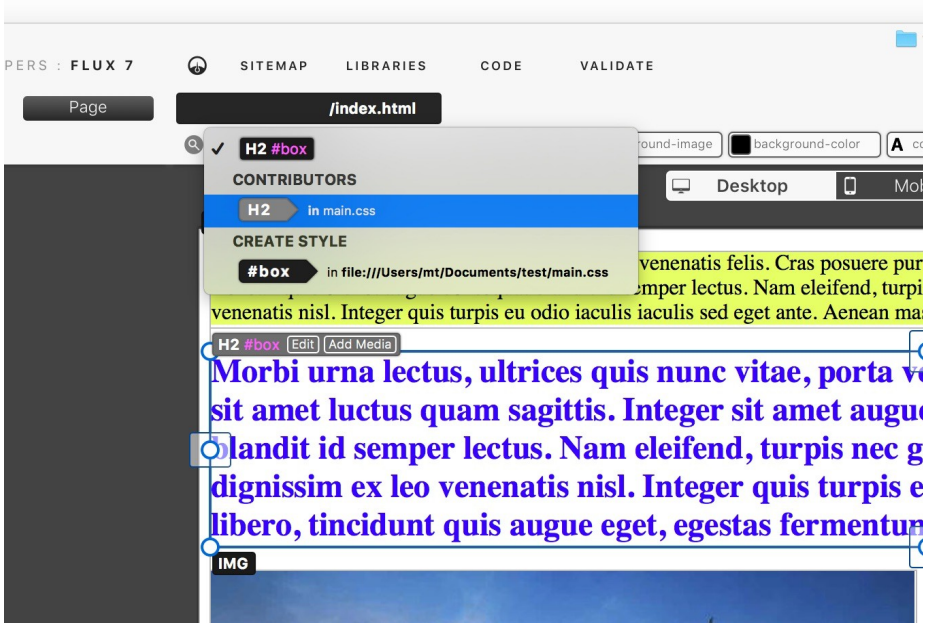
Flux controls where the edits are take place with the popup shown below.

If you click this popup, you will be given the options where you can make the change. In the example below, the top entry is the inline style, and there is one 'contributor', which is the 'h2' style in main.css.

There can be many contributors, and if your element has an *id* or *class* then Flux will offer to create an appropriate style as shown below.

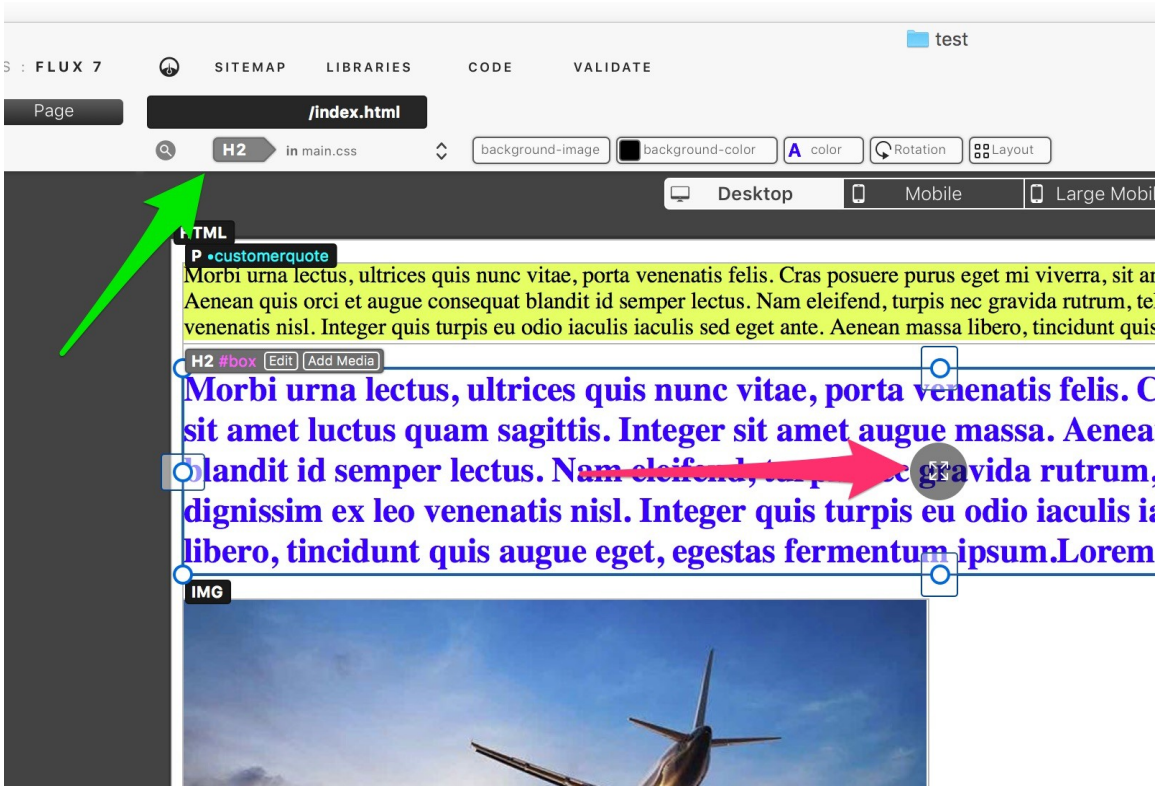


But in this case, we'll use a style that already exists.



If we choose 'h2' in main.css, then any changes are made to that style and *not* inline.

If you want to move the element, make sure you use the circular movement handle in the middle of the element, this will prevent Flux from re-selecting the element.



When ‘h2 in main.css’ is selected, all WYSIWYG changes are made to that style, so if you drag a padding or margin handle, the changes are made to the external CSS style and not inline.

This also applies to the contextual toolbar, tools such as color, Rotation, and Layout shown above will be made to the external style.

There are screencasts available from www.theescapers.com/flux which explain this visually.

CSS

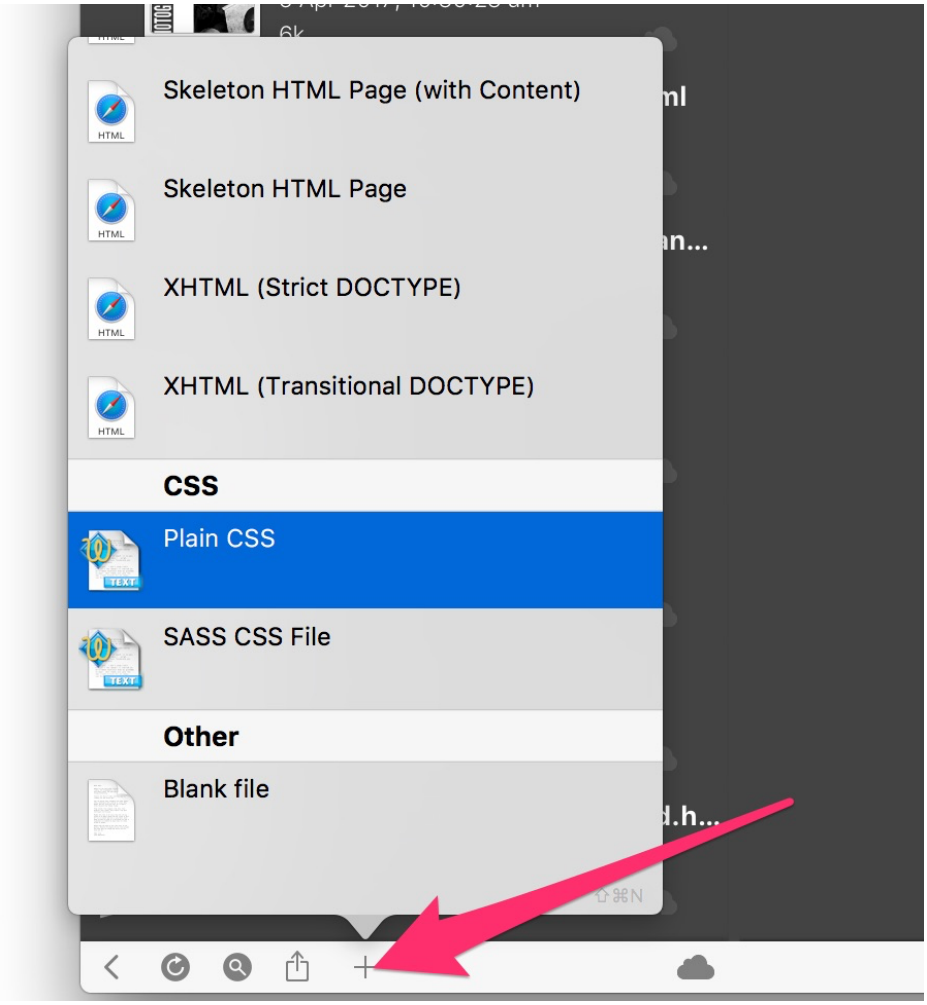
CSS is a fundamental part of modern web design, and therefore, a fundamental part of Flux.

External Styles

Creating a Stylesheet

Stylesheets are created in the ***Site Manager***.

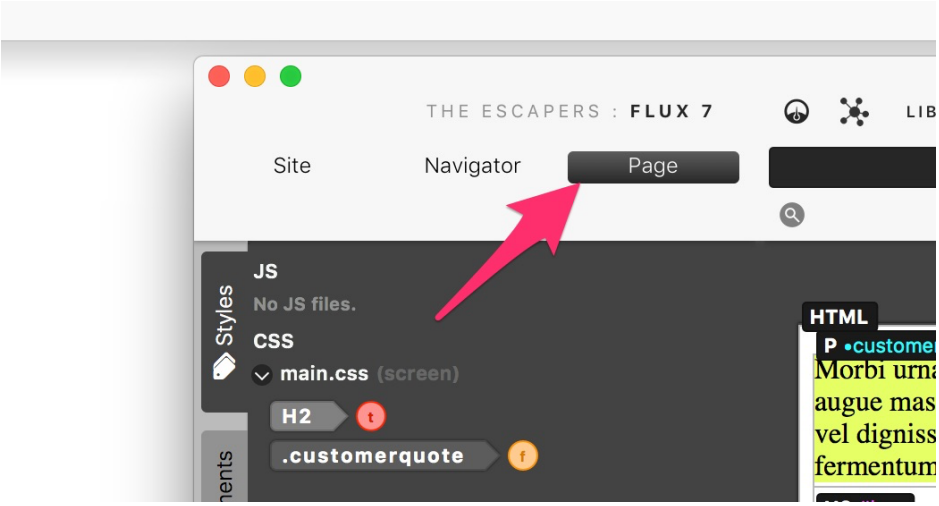
Click the + button to open the Popover, and choose 'Plain CSS'.



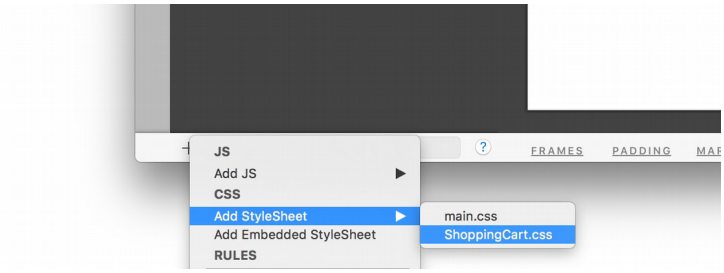
You will be asked to enter a name for your CSS file, you can name this anything you like, but it makes sense to name it something relevant to the content it will contain, for example, you might have a separate CSS file to style your shopping cart, in this case, you could call it 'ShoppingCart.css'.

Attaching a Stylesheet to a Web Page

After opening the Web page, make sure you are on the 'Page' tab, with 'Styles' selected.



Click the + button at the bottom of the ***Style Navigator*** panel, and choose the CSS file you want to add from the popup menu.

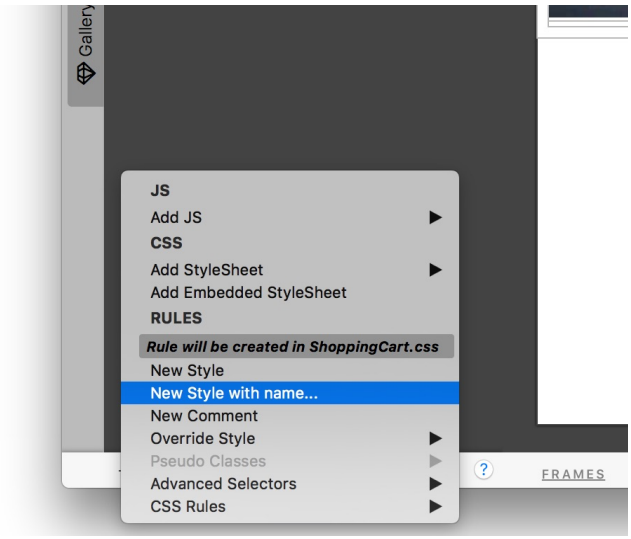


You will be asked what type of media the Stylesheet is intended for. Unless you have specific requirements, choose 'All'.

The Stylesheet will appear in the ***Style Navigator***, ready for use.

Adding styles to a Stylesheet

Select the Stylesheet you want to add the style to by clicking on it in the ***Style Navigator*** panel.

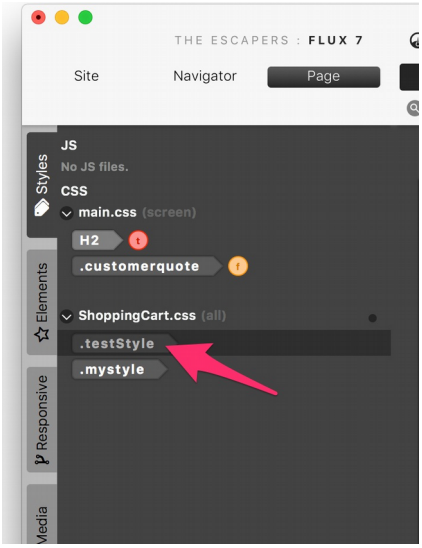


Click the button at the bottom of the ***Style Navigator*** panel, and choose

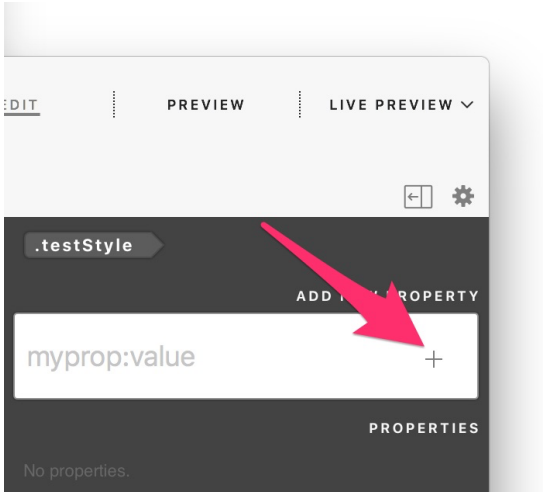
You will be asked to give the style a name, let's call it '.teststyle'.

Adding Properties to Styles

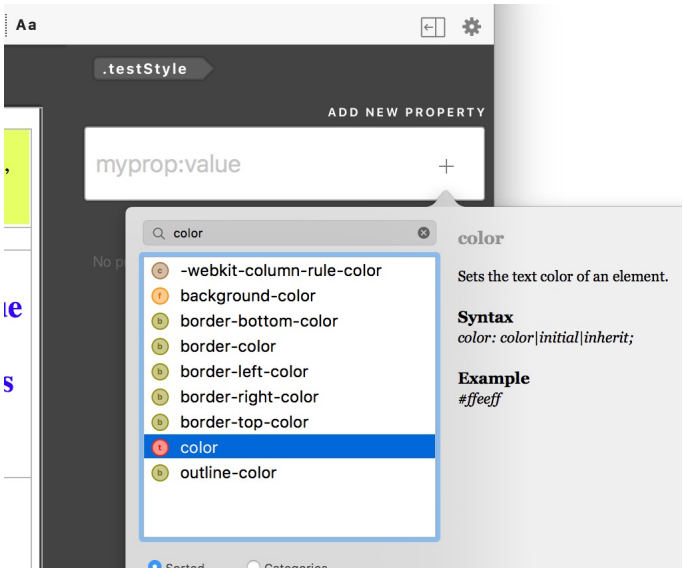
Select the Style you want to edit by clicking on it in the ***Style Navigator*** panel.



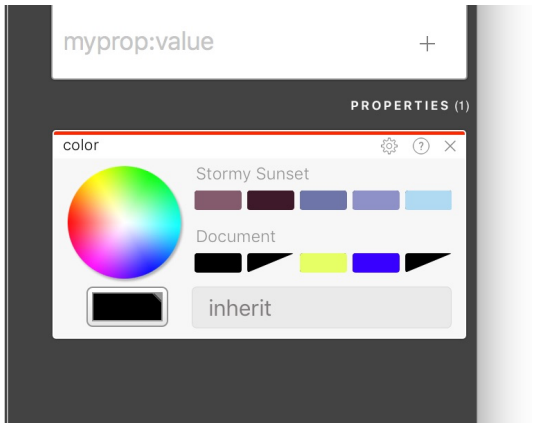
We can now add properties to the style in the ***Inspector*** by clicking the + button.



Double click the property you want to add in the list, in this case, we will choose color, which defines the text color of the style.

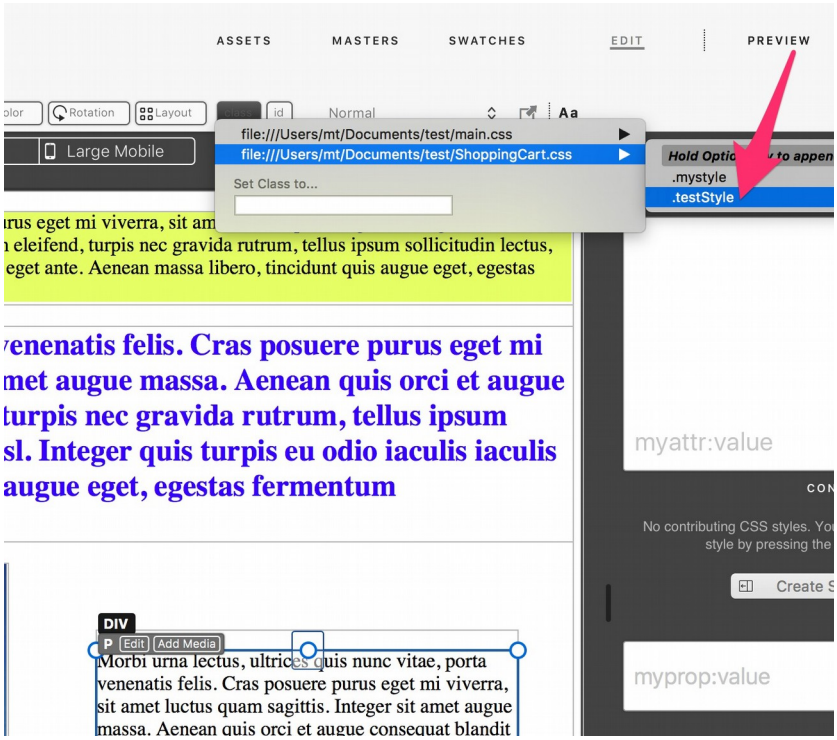


The property will be created, and display within the ***Inspector*** ready for editing.

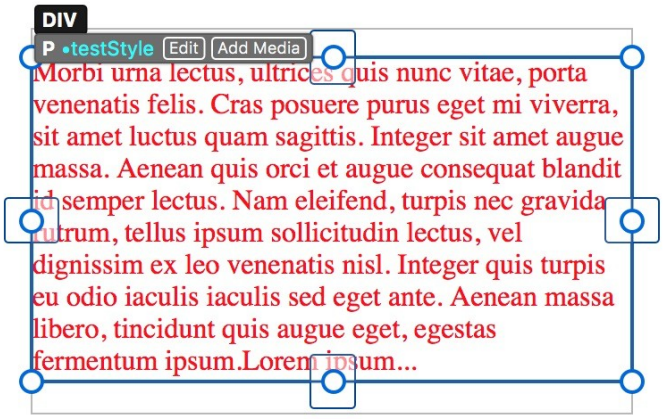


You won't see the changes right away, because your new style isn't being used by any elements yet.

We will now apply the style to an element on the page, by selecting it, and choosing our new style from the *class* button.

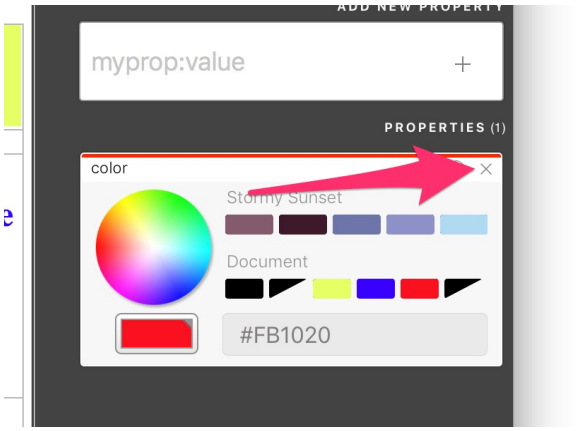


We can now see the style has been applied, and the selected element will display what class it is using.



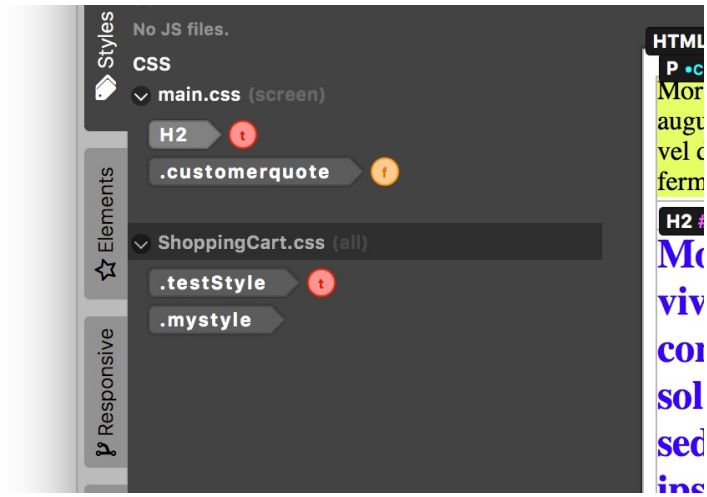
Removing Properties from Styles

To remove a property, simply press the [X] button next to the property you want to remove.



Unlinking Stylesheets from Webpages

To unlink a Stylesheet, select the the StyleSheet in the ***Style Navigator*** and hit the Delete key.



Inline Styles

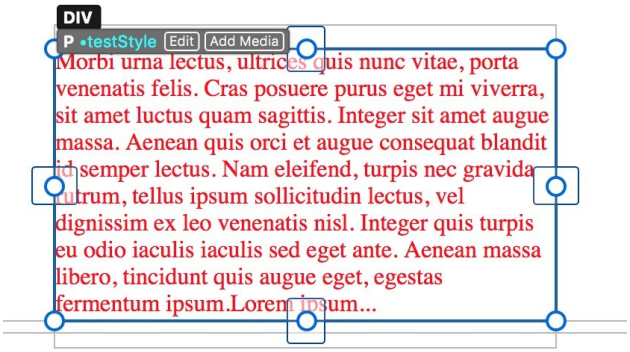
Inline CSS has the highest priority out of the three ways you can use CSS: external, internal, and inline. This means that you can easily override styles that are defined in external or internal by using inline CSS. However, this somewhat defeats the purpose of CSS, which is to separate content from design.

The following element has an inline style, this style will make the text red.

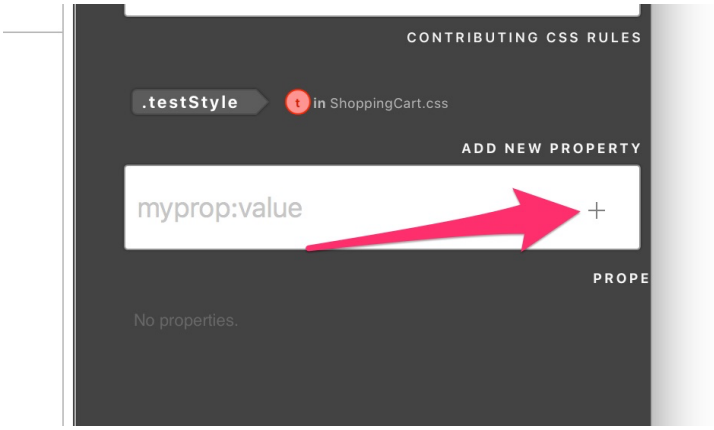
```
<H1 style="color:red">Red Text</H1>
```

Applying Inline Styles to an Element

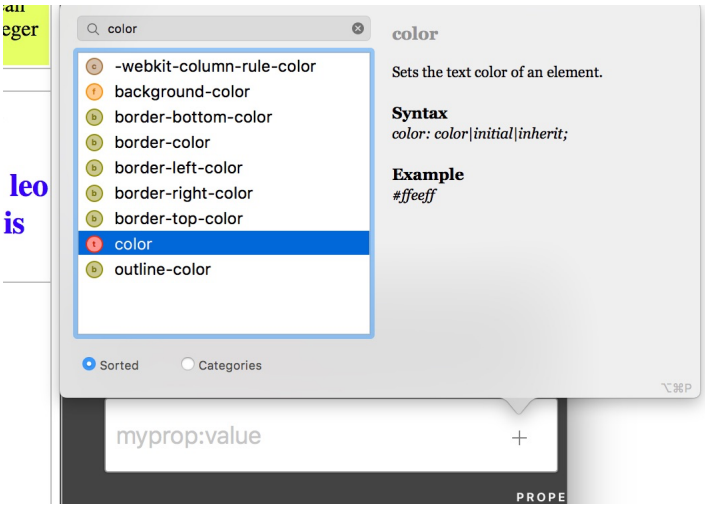
To apply an inline style to an element, simply select it on the page. By default, Flux will edit the inline style. If you want to change the style Flux applies the changes to, then please read the ”What is being edited?” section in the Workflow section of this guide.



You can then add properties from the + button.



In this case, we will add the color property.



And change the value in the ***Inspector***.

The change will apply immediately, as you don't need to 'link' an inline style, as it is directly within the element it is to apply to.

Inline Vs External Styles

Applying Inline styles is easier, less complex than applying an External style, however, it is worth the external effort to create External styles, and only use Inline styles when truly appropriate.

Imagine you had a website with 10 pages, all of those pages had 10 H1 elements with inline styles you wanted to change to be a different font, or different color. That's 100 styles you need to change!

If you used an external style instead, such as

```
H1 {color:blue;}
```

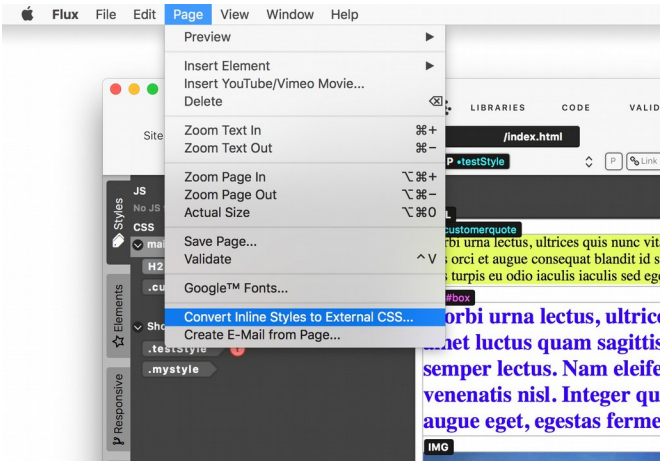
It would be one single change, and all your pages would be updated.

Flux 7 has a tool to help convert pages styled with inline styles into external styles, this is discussed in the next chapter.

Converting Inline Styles to External Styles

Flux 7 has a tool to help convert pages that use Inline styles into pages that use external styles.

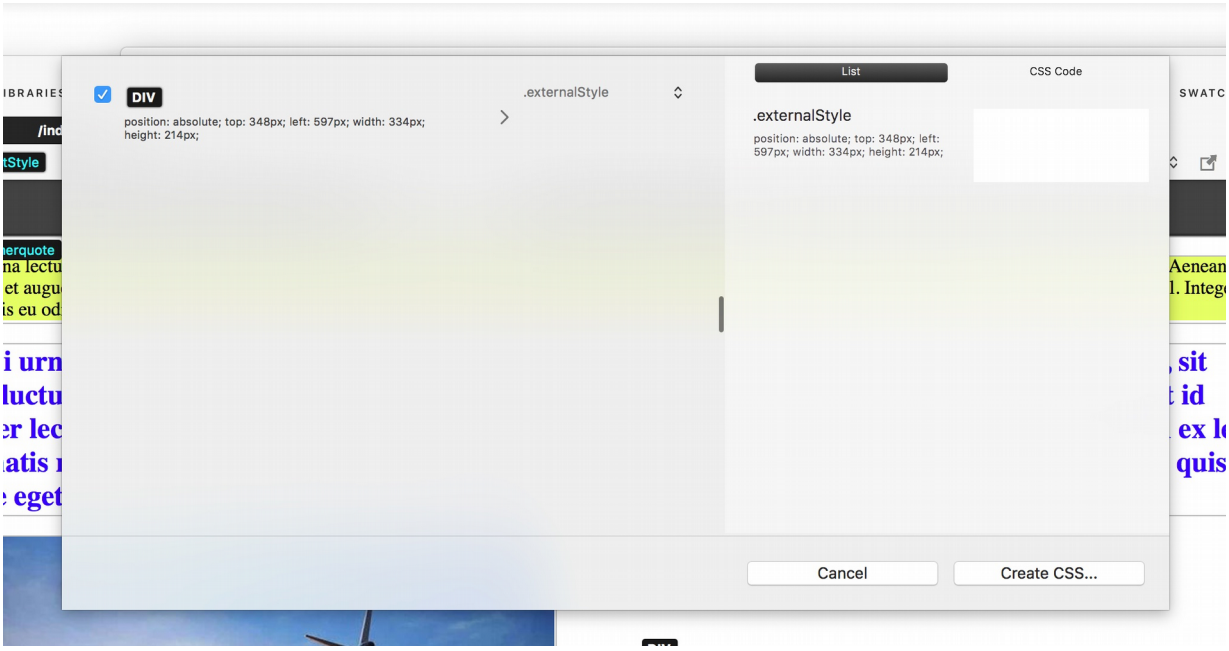
To use this tool, open the page you want to convert, and choose **Convert Inline Styles to External CSS...** from the **Page** menu .



You will be presented with a panel that displays the page elements on the left side, and the styles that will be created on the right.

By default, the styles are not named descriptively, as Flux cannot predict the purpose of those styles, so it makes sense to rename them before creating the external CSS.

To rename a style, select the style, and double click the name to change it.



It's of course possible to change the names in the CSS afterwards, but you'd also need to change the names in the elements where the styles are referenced, which would be very time consuming, so it's best to get it right at this stage.

Once you are happy with stylesheet that will be created, press the *Create CSS…* button.

The styles created will be editable in Flux just like any other external stylesheet.

It's very important to note that this tool makes very substantial changes to your page, and it's crucial to make backups of your site.

Advanced CSS Selectors

CSS is very powerful, it's also very complex, and deserves a book in it's own right, but this article will attempt to explain some of the more commonly used CSS Selectors.

Selectors are one of the most important aspects of CSS as they are used to 'select' elements on an HTML page so that they can be styled.

CSS Reset

An important, but often overlooked selector is the '*', it can be used to reset the parameters of all elements. This is especially important when targeting many different web browsers, as they will all have different defaults for styles, for example, Safari may give H1 a default 2px margin, Firefox may be 5px, Opera may be something else.

```
* {
 margin: 0;
 padding: 0;
}
```

The * can also be used with child selectors.

```
#quote * {
 border: 1px solid black;
}
```

This will target every single element that is a child of the #quote div.

ID Selectors

The following selector will only target the element with an *id* of 'quote'.

Every page can only contain one instance of an id, it is a unique identifier, and should never be used more than once on page. For this reason, it's often used for something that must be uniquely identified on a page.

```
#quote {
 border: 1px solid black;
}
```

Class Selectors

The following code is a class selector. The fundamental difference between ids and classes is that, with class styles, you can target multiple elements. Use classes when you want your styling to apply to a number of elements. For this reason, you will probably find yourself with many times the number of class styles than id styles.

The preceding period specifies a class style.

```
.quote {
 border: 1px solid black;
}
```

Tag Selectors

The following code is a tag selector. In this case, it will 'select' all **H1** elements.

```
h1 {
 border: 1px solid black;
}
```

Pseudo Selectors

The following code is a pseudo selector. In this case, it will 'select' all **a** (link) elements, that have been visited.

```
a:visited { color: red; }
```

Descendant Selectors

The following code will apply to all elements that have a *class* of 'box' that are inside an element with an *id* of 'container'. It doesn't need to be an immediate child, just a descendant.

```
#container .box {
  float: left;
  padding-bottom: 15px;
}
```

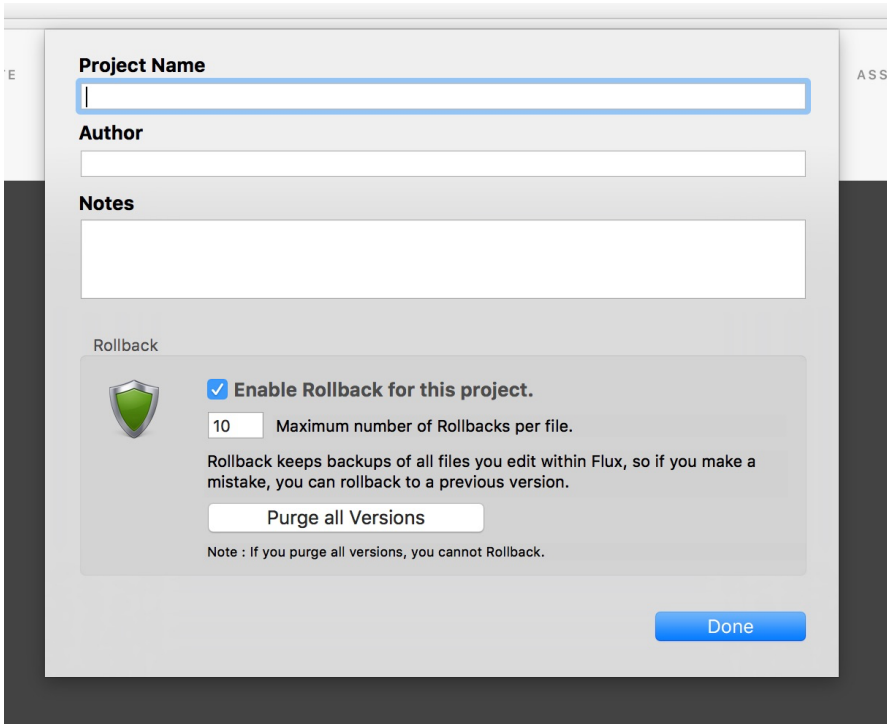
Child Selectors

The following code will apply to all elements that have a *class* of 'box' that are directly inside an element with an *id* of 'container'. It needs to be an immediate child, i.e. parent to child, not grandchild.

```
#container > .box {
  float: left;
  padding-bottom: 15px;
}
```

Project Settings

Project settings is accessed from the **Project Settings...** menu entry in the **File** menu.



Project Metadata

The Project Name, Author and Notes are purely for your own reference, and have no effect on the website.

Rollback

If ***Rollback*** is enabled, then every time a file is saved within Flux, a copy of previous version is kept. This feature is explained in more depth in the ***Rollback*** section of this manual.

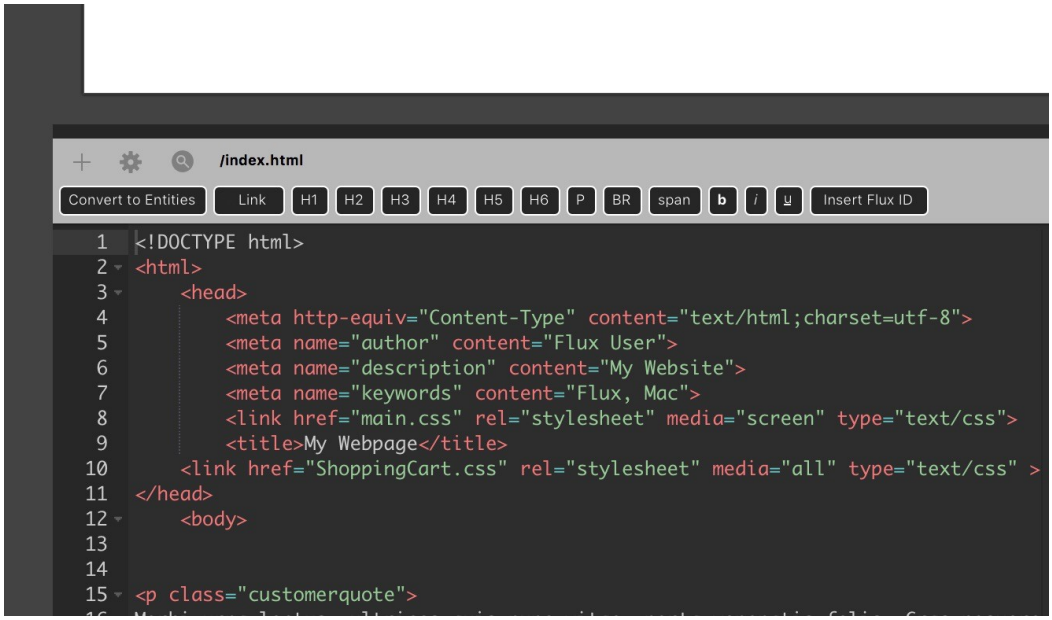
Code Editing

Flux allows complete control of your code, both HTML and CSS. You can edit code directly in the ***Code Editor***, and the changes will be reflected right away on the page.

Flux 7 introduces an entirely new code editor, with improvements in all areas including auto-completion.

Toggling the Code Editor

The ***Code Editor*** is toggled in the **View** menu, under **Code Editor**.



Making changes

Simply edit the code to your requirements, all changes will be applied automatically. Flux can cope with invalid code, but if it's too messy and invalid, then certain features will not work, such as WYSIWYG control over page elements, i.e. if Flux can't find or understand your code, then it can't (and won't attempt) to change it.

Using the Code Editor toolbar, you can selected portions of text, click for example the *H1* button and the selected text will be wrapped in the appropriate tags.

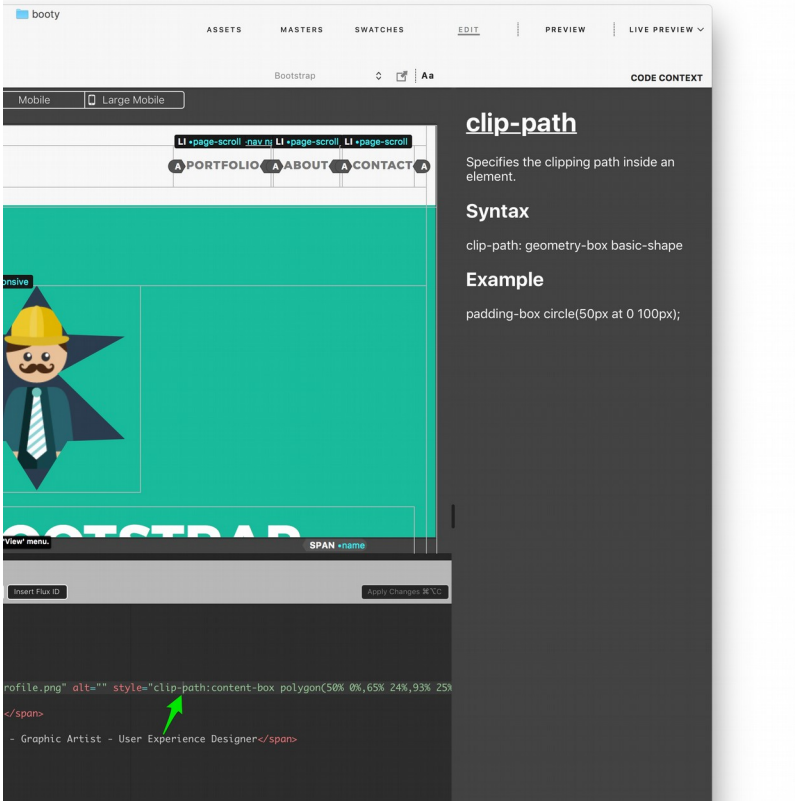
Finding sections of code

When the ***Code Editor*** is displayed, you can click on page elements or CSS styles, and the editor will display and highlight this section. This makes it easy to find, and edit any particular element.

Code Context

New in Flux 7, Code Context introduces even more power to the Code Editor.

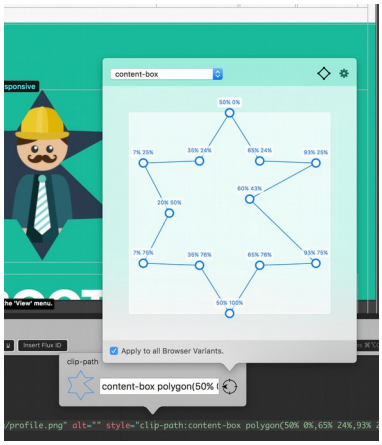
Code Context displays help on the relevant code, and also helps to edit it.



In the example above, the text cursor is within the 'clip-path' CSS property. This is a property that is recognized by Code Context, so useful help is displayed in the right panel of Flux.

In addition to this help, Flux also provides assistant editors for these CSS properties. These are accessed by holding the Command Key, and clicking on the property.

All properties that are supported in the Inspector are also supported by Code Context.

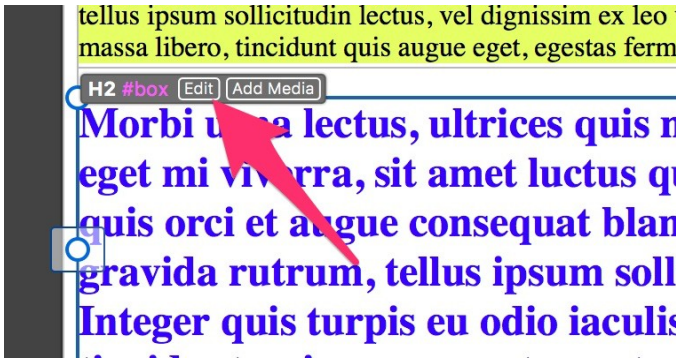


Text Content Editing

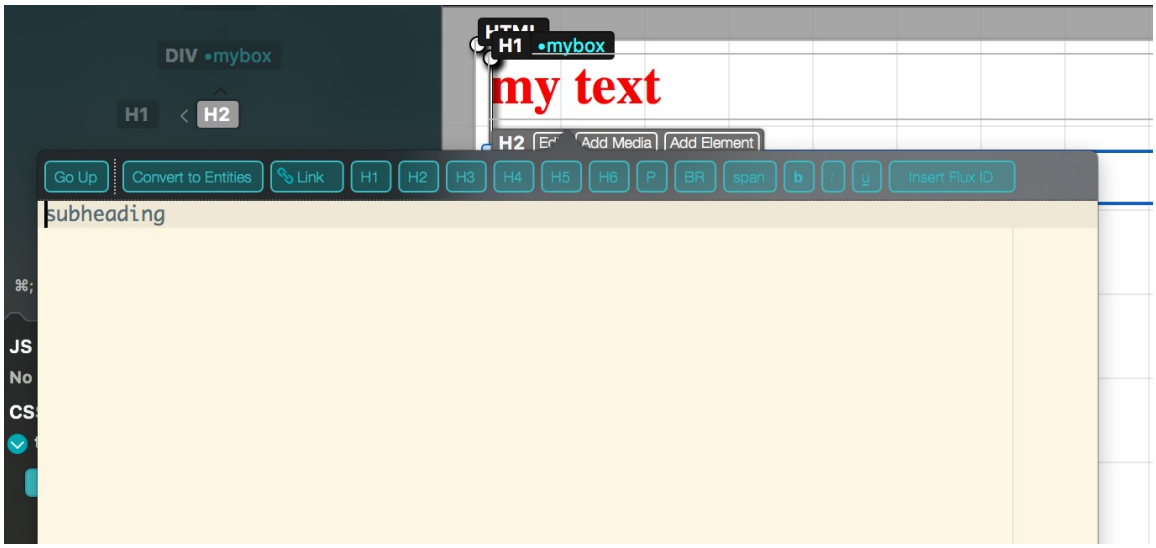
Flux 7 handles text editing differently to previous releases, so even if you are experienced in this area in Flux 4, please still read this article.

Content Editor

Flux now uses the ***Content Editor*** for advanced editing of text content. To access this, press the 'Edit' button on the element tag on the page, or the ***Element Navigator***.



The ***Content Editor*** shows the code contents of the element, and allows you to change this content.



In this window, you are changing actual HTML, so you need to ensure it's valid.

The ***Content Editor*** includes a toolbar, where you can access frequently used tags, simply select the text you want to apply the tag to, and press the appropriate button.

All changes made in the ***Content Editor*** are applied to the page automatically.

Since the ***Content Editor*** is also available from the ***Element Navigator***, you can make text changes to hidden, or difficult to access elements.

Inline Editing

Inline editing can be accessed simply by double-clicking the element you wish to edit, and then simply edit the text like you would in a word processor. You can switch back to the 'Movement' mode using the toggle at the top right of the page.

Take care when using inline editing, as you are editing a hierarchal structure, you may be making changes that are fundamental to the structure of your document. If you are making substantial changes, we recommend that the Content Editor is used.

If multiple elements are on top of each other, you may find it difficult to edit the text in this way, so again, we recommend that the Content Editor is used in this situation.

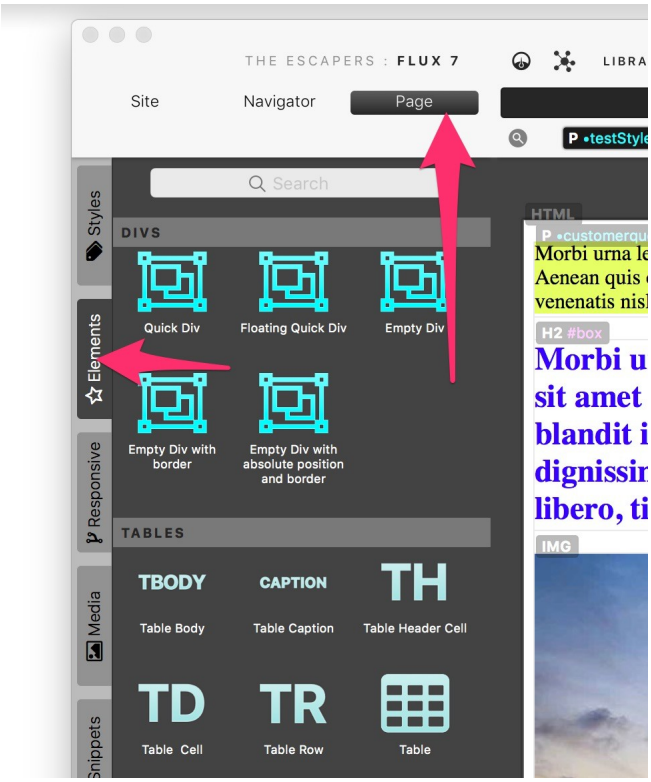
As with all major editing, we strongly recommend backups are kept.

Element Library

The ***Element Library*** is one of the most important areas of Flux, it's where objects are inserted into your page.

Finding an Element in the Library

The ***Element Library*** is accessible in the 'Page' tab, in the 'Elements' section.



To find the element you want, you can scroll down the list to find it. Optionally, this list can be filtered to only show element that are compatible with the selected page element.

Inserting an Element from the Library

Once you have found the Object you want to insert, select the element on the page you want to insert the object into and then double click the element. You can also drag the element to the desired parent element on the page.

Advanced Insertion

By default, Flux will insert the new element at the end of the selected object. This may not always be desirable, so you can also insert an object before or after the selected object using the contextual menu.

For example, if you have two P paragraphs, and you want to insert a H2 header in between them, select the first of the P elements, and choose **Insert after selected Element**.

Inserting Text Elements

When inserting text elements, such as P, SPAN or B (bold), Flux will insert some sample text into the element, making it easier to manipulate on the page.

Inserting Elements from the menu

The ***Element Library*** is also available from the **Page** menu, under the **Insert Element** sub menu.

Keyboard Shortcuts

You can setup keyboard shortcuts for your frequently used elements. This is done in the **Keyboard** section of **System Preferences**. Choose the 'Shortcuts' tab and add a shortcut under 'App Shortcuts'.

Navigator

The ***Navigator*** is a hierarchical view of the page. The ***Navigator*** shows all the elements on the page, and can assist in understanding the structure of a page, and accessing hard to reach elements.

Flux V6 also contains the Mini Navigator for quick access to hard to reach elements.

Navigating

To select an element, simply click on the element in the Navigator list. This has the same effect as clicking the element on the page.

Editing Code Contents

When the mouse pointer is over an element that can be edited, a small *Edit* button will appear. Clicking this button will open the Code Content Editor, where you can manipulate the code directly, changes will be applied automatically.

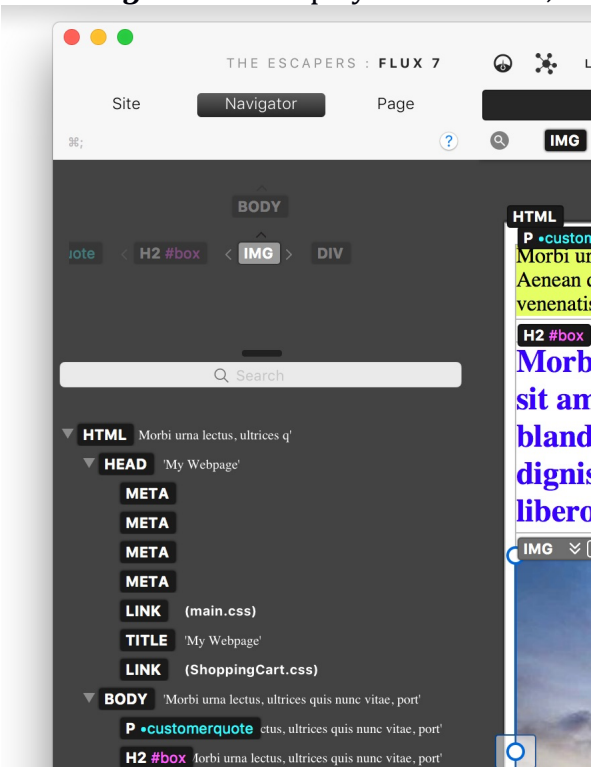
Understanding the hierarchy

A HTML page is a tree of elements, an element (other than the root) will have a parent, and, very often, child elements.

For example, the code below is a very simply page, the **DIV** element has two children, a **H1** , and a . **H2**

```
<HTML>
    <BODY>
        <DIV>
            <H1>my text</H1>
            <H2>subheading</H2>
        </DIV>
    </BODY>
</HTML>
```

The ***Navigator*** will display this as a tree, often called the 'DOM' tree.



The hierarchy of a page is important to the way it looks, for example, if we moved the parent **DIV** element, the **H1** and the **H2** would also move, as they are contained within the **DIV** .

This tree affects CSS profoundly, for example, if we setup a style called '.mybox H1'

```
.mybox H1 {
        color:red;
}
```

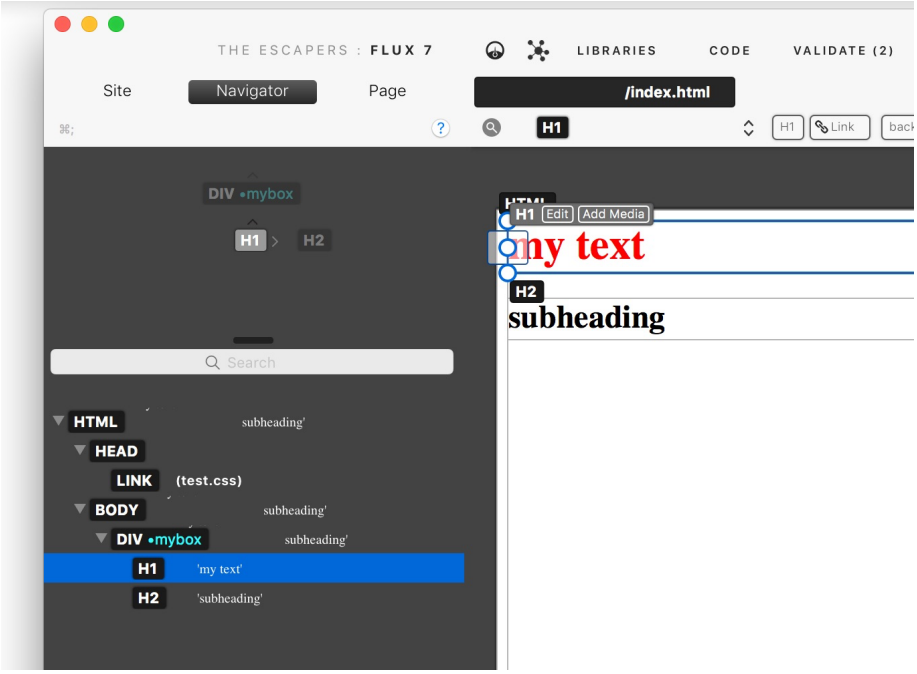
And we altered the HTML to this. (In this example, the new style has been setup in the test.css file.)

```
<HTML>
  <head>
     <link href="test.css" rel="stylesheet" media="all" type="text/css" >
   </head>

        <BODY>
```

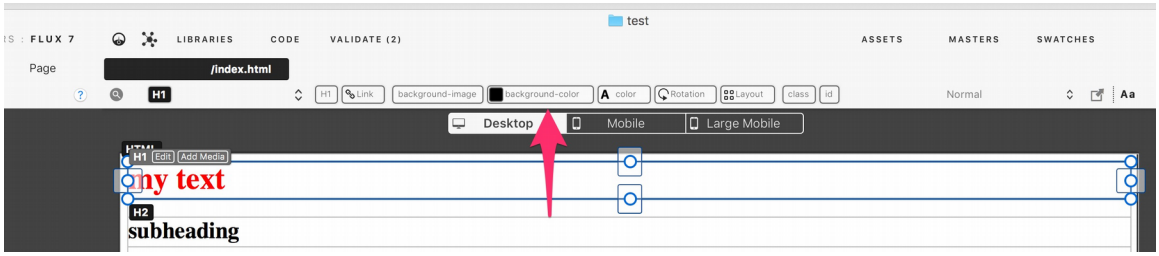
```
            <DIV class="mybox">
                <H1>my text</H1>
                <H2>subheading</H2>
            </DIV>
        </BODY>
</HTML>
```

Then all instances of the **H1** element inside the 'mybox' **DIV** would colored red, all other **H1** elements would be unaffected.



Contextual Toolbar

The ***Contextual Toolbar*** is a toolbar at the top of the ***Page Editor*** which provides tools based on the type of element that is selected.



From the toolbar, you can change a variety of options, including the type of element, in the above image, we could change the element from **SECTION** to **H1**.

We can also change the color of the element(background-color) or the color of the text within (color).

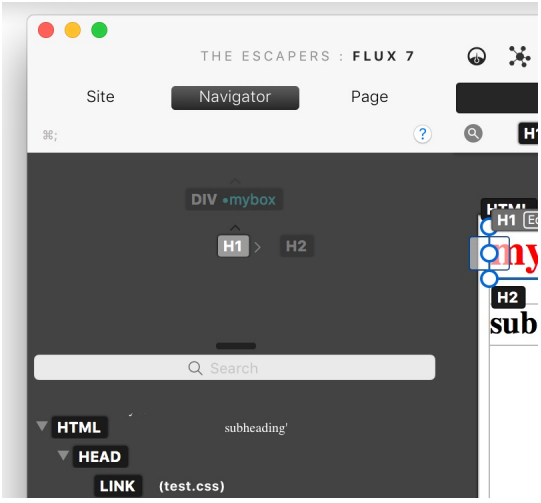
Inline Vs External Styles

In the above example, any changes will be 'inline' to the element, i.e. not in external CSS (which is considered to be best practice). To make changes to the external style, make sure you select the appropriate CSS style, and not the element on the page.

Mini Navigator

The Mini Navigator is a new feature in Version 6 of Flux. It provides easy, and contextual access to elements on the page.

Due to the hierarchal nature of webpages, it's common that objects overlap, so the Mini Navigator allows you to access these elements easily, quickly, and if you like, via the keyboard.



When an element is selected on the page, it's parent objects, child objects and sibling objects will be display around it in the *Mini Navigator*, the parent objects appear above, child objects below, and sibling objects to tbe sides.

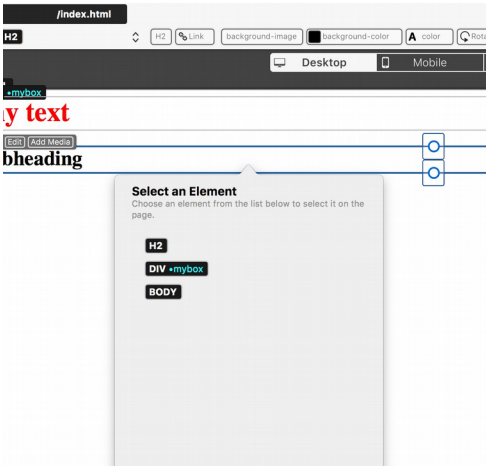
To change the selection, simply click the element you want to select.

The Mini Navigator can be activated by the keyboard shortcut ⌘;

Once the Mini Navigator is activated, you can use the cursor keys on your keyboard to change the selection.

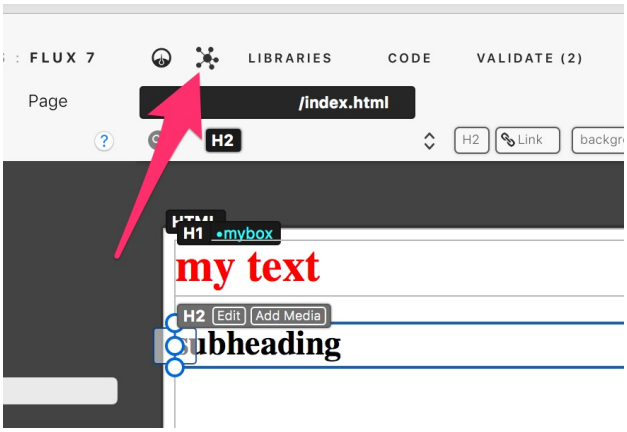
When the selection changes, the element is 'bounced' on the page to make it easier to locate.

As an alternative, you can also click & hold on the page to present a popover containing all elements under the mouse pointer to assist in selecting difficult to access elements.



Site Map

Flux 7 includes a Site Map feature that can be accessed from the main Toolbar.



If a Site Map tab is opened when a page is open, then that page will be used as the starting point. If there is no page open, then the currently selected file in the *Site Manager* will be used.

When a Site Map tab is opened, it will create a 'map' of the current page and it's links.

A Site Map can take a while to build if many of the links are remote, or you a slower internet connection.

A Site Map can be useful to understand the navigation of your site.

If your site has broken links, these links will be shown in red.

The Site Map, optionally, can also display resources used by your pages, such as images, and CSS files.

The Site Map has a contextual menu that allows the following of links, and editing of linked pages (note, only pages in your site can be edited.)

Overlays

At the bottom of the ***Page Editor***, there is the ***Overlays Toolbar***.

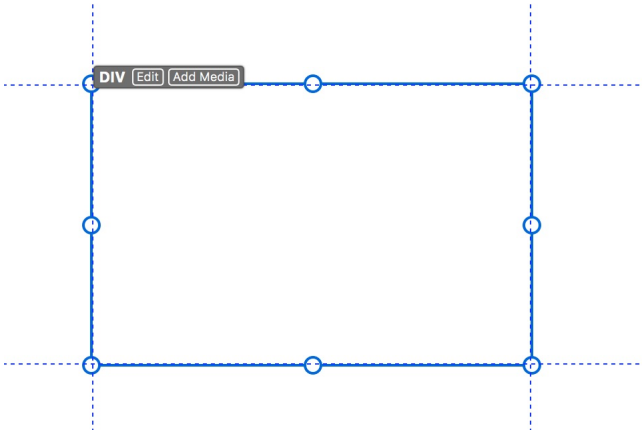
What are Overlays?

Overlays are guides and tokens that are overlaid on your page design to assist in the editing of the page.

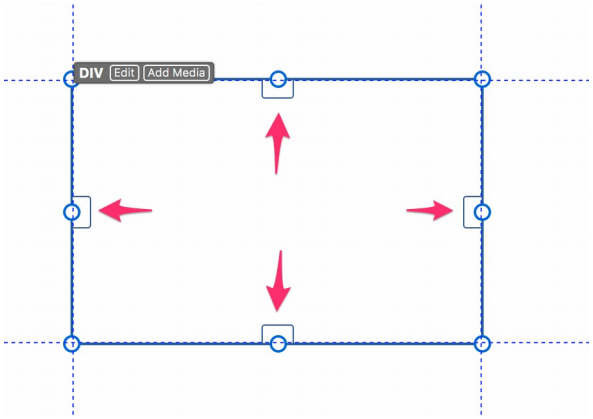
For example, if we enable the 'Padding' overlay, then we are able to drag the padding handles to increase or decrease the padding on an element.

If you are not currently editing padding properties, then it may be beneficial to turn off that overlay, it'll make Flux run quicker, and your page will be less cluttered.

This is how a page could appear with the Padding overlay turned off.



This is how a page could appear with the Padding overlay turned on.



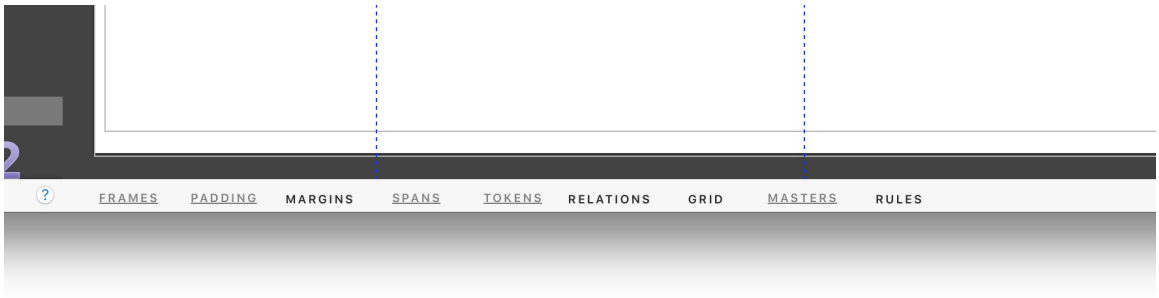
Note : Overlays do not appear on the finished webpage, they are used for editing the page.

There are other overlays such as 'Spans' which will toggle the display of text level elements such as (bold) and (italic).

Toggling Overlays

Overlays are toggled by clicking on the buttons on the toolbar, the buttons are highlighted if the option is currently turned on.

Some overlays are affected by the selection state of the element, if an element is selected, the overlay may change to display more information.



Swatches

Flux allows you to manage colour 'swatches', these are simply collections of colours that have usually be setup to complement each other. You can download swatches from www.colourlovers.com

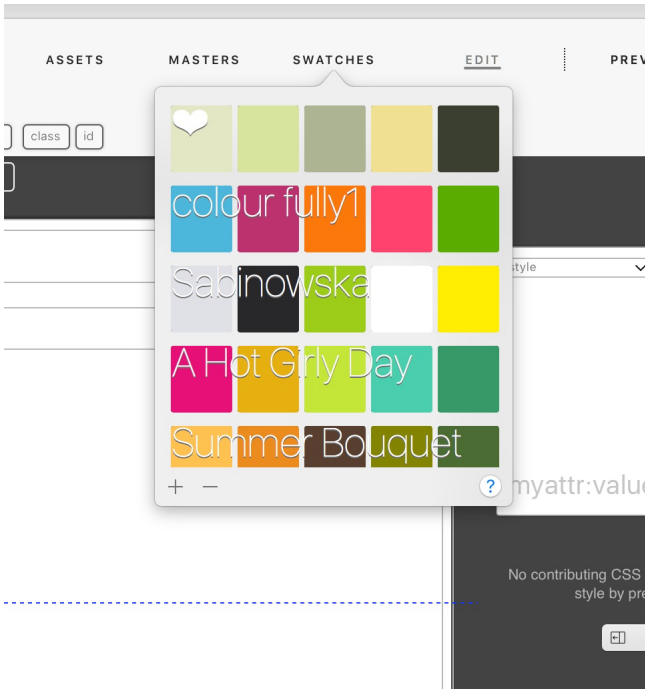
Importing Swatches

To import a swatch into Flux, simply double click a .gpl file. Flux can currently only import '.gpl' files.

Selecting Swatches

Once you have imported or created a swatch, you can activate it by simply selecting it in the list. A tick mark will appear on the current swatch.

The swatch will then become available in areas such as background-color in the Inspector, making in simple to apply complementary colors to areas of your design.



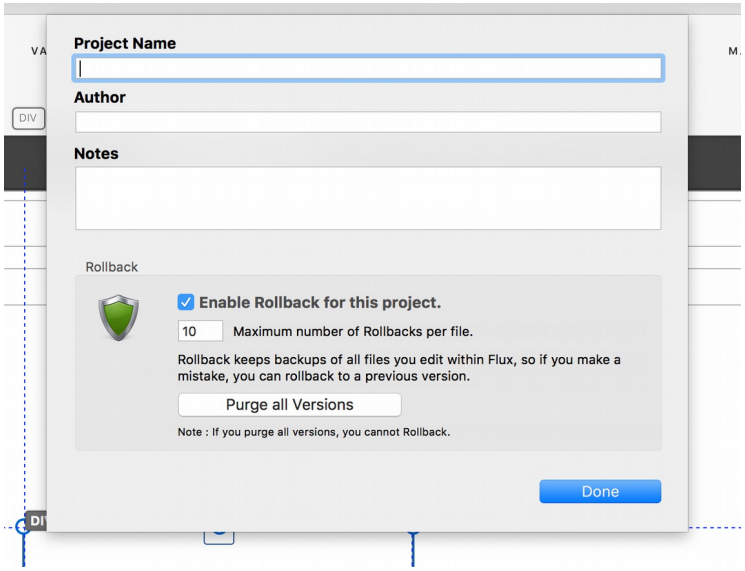
Rollback

Rollback is a feature that allows you to step back through previous versions of files.

Enabling Rollback

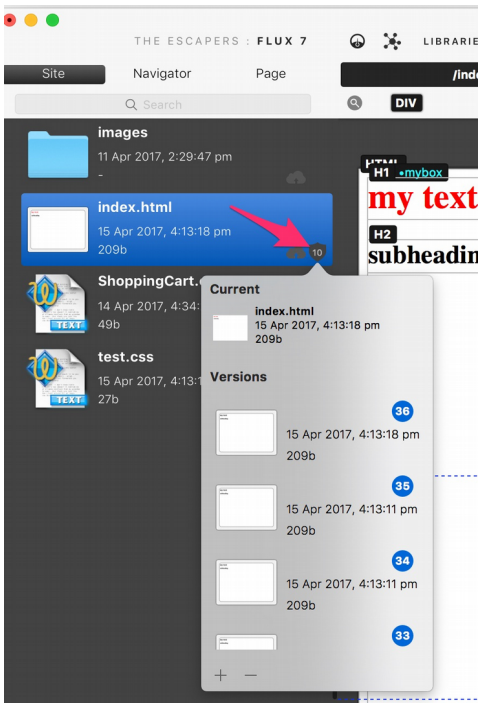
Rollback is enabled on a project by project basis. To enable Rollback, choose **Project Settings...** from the **File** menu.

Check the *Enable Rollback for this project* switch button. Press *Done*.



Using Rollback

Every time you save a file in Flux, the previous version is recorded. To rollback to one of these previous versions, click on the small 'shield' icon next to the file in the ***Site Manager***, this will display a popup list of the available versions. Double click the version you wish to rollback to.



Notes

The icon displayed for the Rollback may not be accurate, as there are many factors that govern how a page looks, such as images and CSS files. The HTML file will be rolled back, but unless you also rollback any other associated files that have been changed, the page may not appear the way you expect.

Google Fonts

Google Fonts is an online directory of hundreds of fonts, which are free for you to use on your website.

Flux 7 support for Google Fonts has changed substantially since V6, so even if you used Flux 6, please still read this part of the manual.

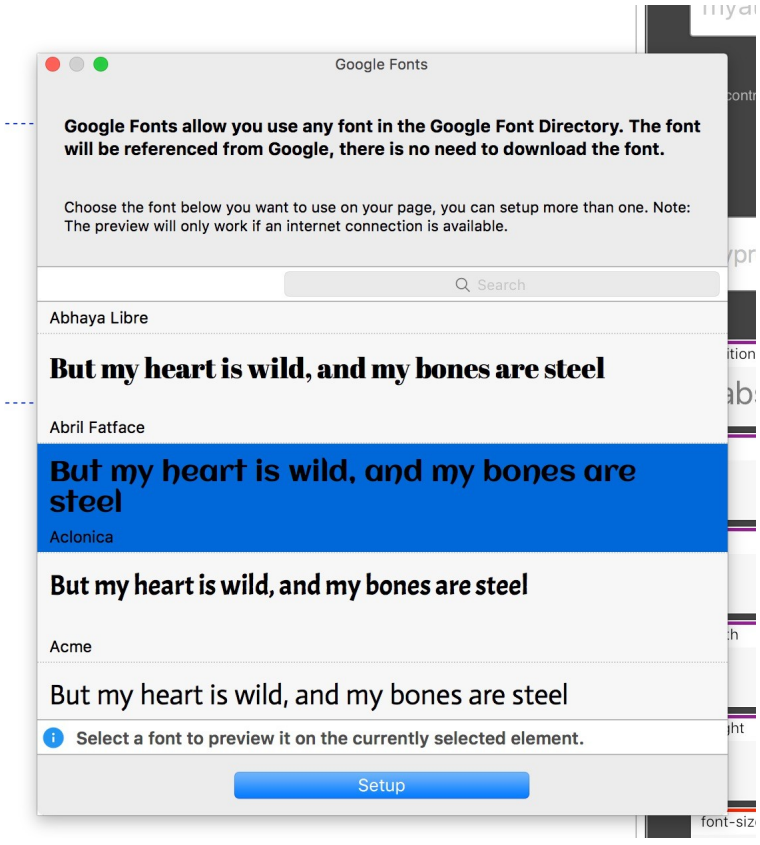
Importing a Font

To use a font from the Google Library, choose the **Google Fonts...** option in **Page** menu.

You will be presented with a large list of fonts.

If you select one of the fonts, and also have an element on the page selected, the font will be previewed on that element. Note, your HTML or CSS are not changed, it's simply a preview.

Once you have chosen the one you want, press the *Setup* button at the bottom of the panel.



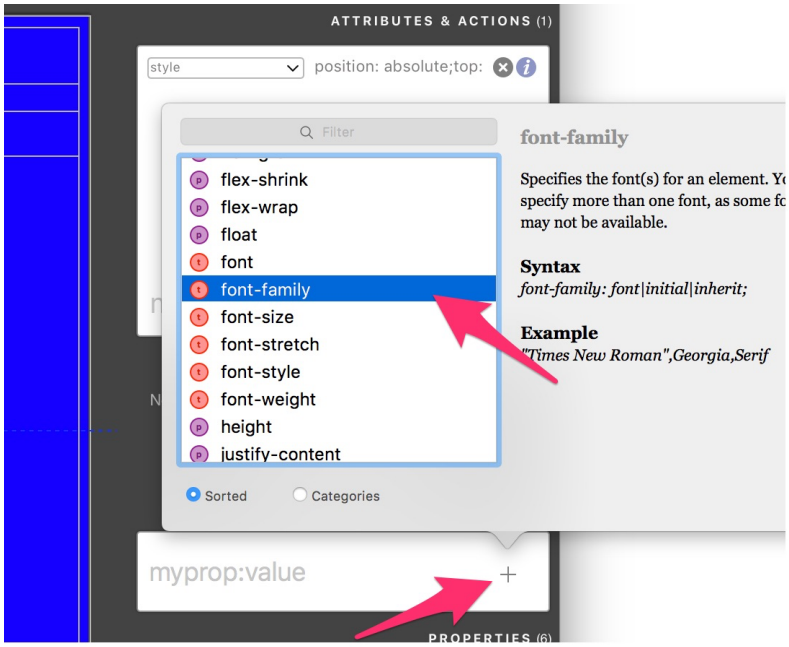
The next dialog asks for a name for a new CSS style to use the font, you don't have to create this, if you don't want to, leave the field blank.

If you do choose to create a CSS style, you will be asked which CSS Stylesheet to put it in.

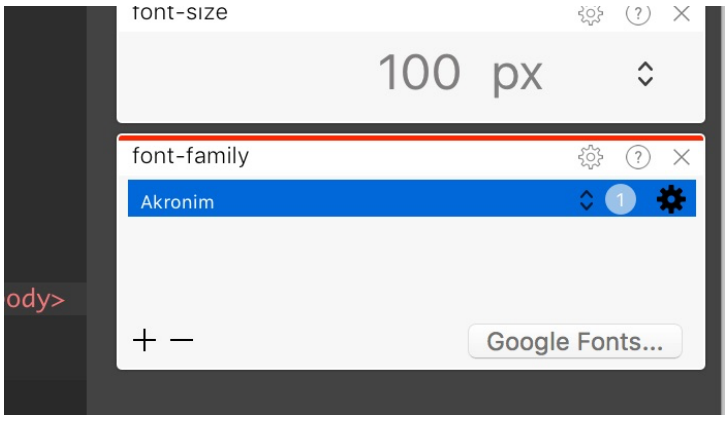
Using the font

Select the CSS style, or page element that you want to apply the font to.

In the Inspector, scroll to the font-family property. If your object doesn't have the font-family property, then add it, using the + button at the top of the Inspector.



In the font-family property, select the font you want from the pop-up menu. Google Fonts are listed at the top of the menu.



Now, all elements that use this style will use the Google Font.

Removing the font

If you no longer want to use the font, you just need to remove the appropriate **LINK** element from your HTML code, for example..

```
<link href="http://fonts.googleapis.com/css?family=Arvo"
rel="stylesheet" media="screen" type="text/css" >
```

Once you've removed this reference, you should also remove any reference to the font in your CSS code,but this isn't strictly required, as the CSS font system is designed to fallback to other fonts if one is not available.

Licence

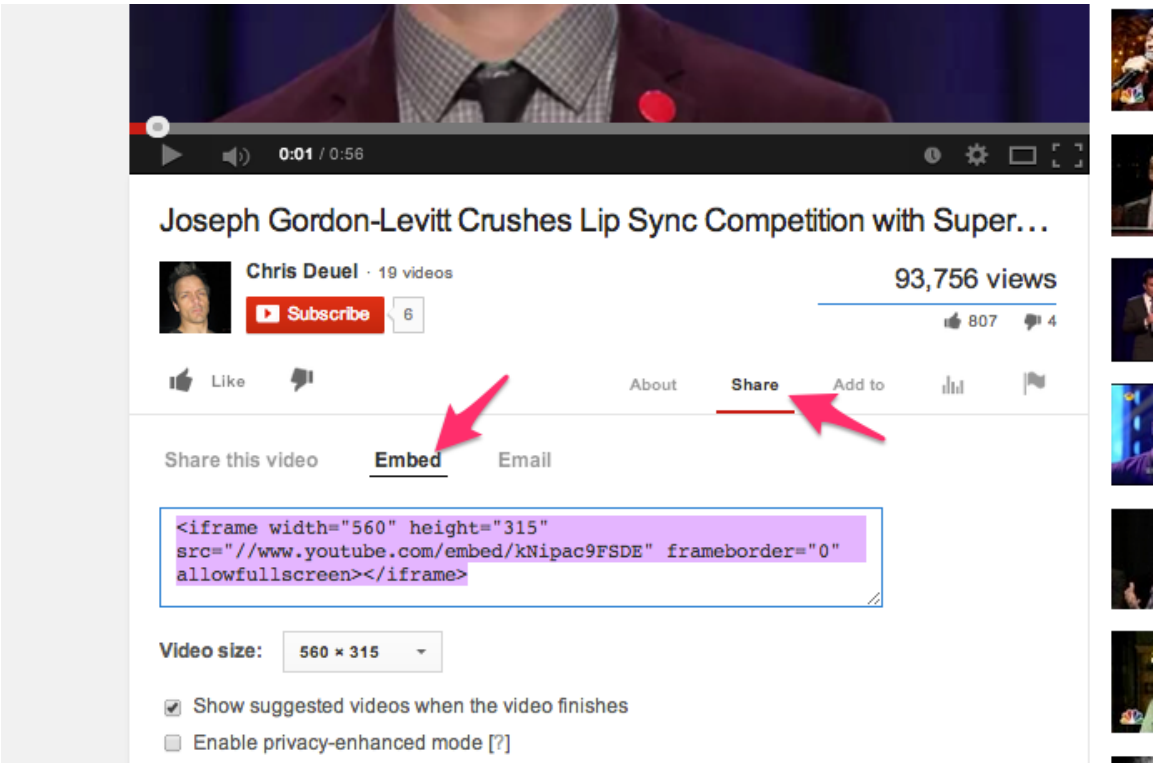
All of the fonts are Open Source. This means that you are free to use them on any site you want, including commercial ones.

YouTube

Flux has integrated support for embedding YouTube videos.

Get the YouTube link

First, get the 'Embed' code, copy it to your clipboard.



Paste in the Code

Select the page element that you want to contain the YouTube video, and hit Command-V (i.e. Paste)

```
<iframe width="560" height="315" src="//www.youtube.com/embed/kNipac9FSDE" frameborder="0" allowfullscreen></iframe>
```

Note : This may not work correctly due to an anomaly in the code provided by YouTube. If it does not, ensure that the attribute starts with a 'http:', so that the resulting code looks like this...

```
<iframe width="560" height="315" src="http://www.youtube.com/embed/kNipac9FSDE" frameborder="0" allowfullscreen></iframe>
```

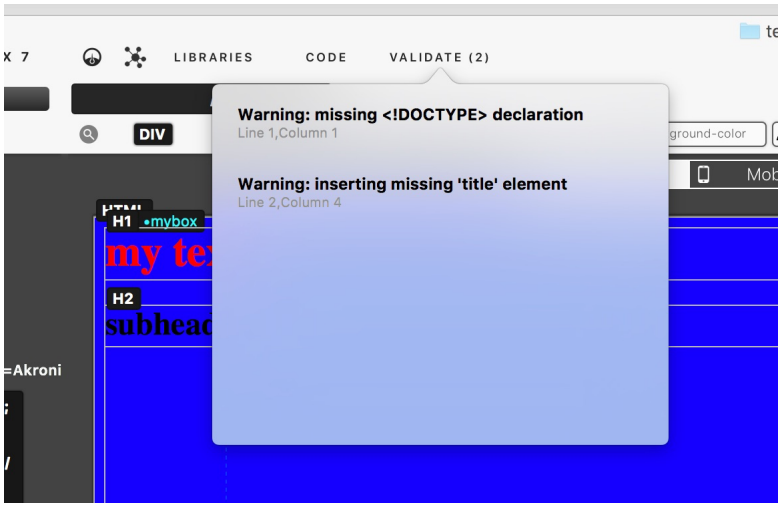
Now, switch to 'Preview', and the YouTube video will be displayed.

Restrictions

Note, some YouTube videos do not allow embedding due to licensing restrictions.

Validator

Flux has a built-in code validator, that will inform you of any mistakes in your Code.



The ***Validator*** is accessed from the *Validate* Toolbar button.

If there are any issues in the HTML code, the Toolbar button will also display the number of issues.

The Validator will not report ALL issues, just ones that conflict with the recommended document structure.

Clicking on the issue within the ***Validator*** will take you to the relevant section of code in the ***Code Editor***.

Master Elements

Master Elements have changed significantly in Flux 7 compared to Flux 4. The new Master Elements system is more flexible, and easier to integrate into an existing workflow, as it uses simple XHTML style comments to delineate the Master Element definitions and where they are in use. The Master Elements system is not backwards compatible with Flux 4, although Flux 4 can open a site using Flux 7 Master Elements, and they will be preserved, and do no harm.

Introduction

Flux 7 Master Elements can be explained using two points:

1) You can make any section of a XHTML or HTML into a named Master Element. 2) You can use this named Master Element in any other page in your site.

This allows you to repeat sections of page anywhere in your site. When you change the content of the Master Element, you can easily and quickly roll these changes out to all other pages using the Master Element.

Creating a Master Element definition

Click on the the *Masters* button on the main toolbar to open the Master Elements palette. Select the tab, *Create Master* . Now select the text in the code editor which you wish to become a Master Element. This will often be a complete **DIV** element, using incomplete elements is completely supported, but might make your HTML invalid if you do not complete the **DIV** where the Master Element is used.

Now your text is selected, enter a name into the text field on the Master Elements palette. This could be something like 'copyright_notice', which might be the copyright notice at the bottom of the page, which you'd like to repeat on all pages.

Now click *Setup as Master*. This will make your selection into a Master Element you can use elsewhere.

Using a Master Element

In the ***Master Elements*** palette, select the tab *Use Master* . Click the Refresh button next to the popup menu. This tells Flux to scan all files for Master Element definitions, and put them into the popup menu.

You will see that the popup menu is now enabled, and contains your Master Element definitions in all your HTML files in the current site.

Put your text cursor where you wish to insert your Master Element definition in the Code Editor, select the Master Element definition you wish to insert, and click *Insert* to use the definition.

You will notice that the code from your definition is inserted and surrounded with comments. These comments are what allows Flux to recognise the code as a definition, and you should not modify them. If you modify the code in between the comments, that code will be lost if you choose to update the Master Elements on that page. A use of Master Element should be considered an 'import' of code from the definition, for use where you wish to repeat code from elsewhere.

Updating Master Elements

In the event that you modify the code in a Master Element definition, you will almost certainly wish to use those changes where that Master Element is in use. To do so, select the tab *Update* in the Master Elements palette.

Here you have two options.

The first option is to update the Master Elements uses in the current page. This is good if you want to quickly test the Master Elements are as you want them. If you decide that you are happy with the changes, you can choose to update all Master Element uses in all pages.

Snippets

The ***Snippets*** panel allows you to collect, organise and use snippets of code.

Snippets can be arranged into collections, and when you want to use them, simply drag them to the code editor to paste them in.

Using Snippets

The ***Snippets*** panel is available from the **Page** tab.

Simply select where on the page, or the Code Editor you want to insert the Snippet, and press the *Insert* button on the Snippet you want to to insert.

Adding Snippets

Select an element on the page you want to add, and choose **Add Element to Snippets...** from the contextual menu.

You can also add Snippets from the ***Code Editor***. Select the text you want to add, and choose **Put selected text in Snippets** from the 'gear' menu.

The Snippets added in this way will appear in the 'Uncategorized' Snippet category.

Exporting Snippets

To export a snippet collection, use the contextual menu on the Snippets panel by right clicking, or Command-Clicking on the list of snippets. Only the current collection will be exported.

Importing Snippets

To import a '.fluxsnippets' file, simply drag it to the Flux icon on your dock.

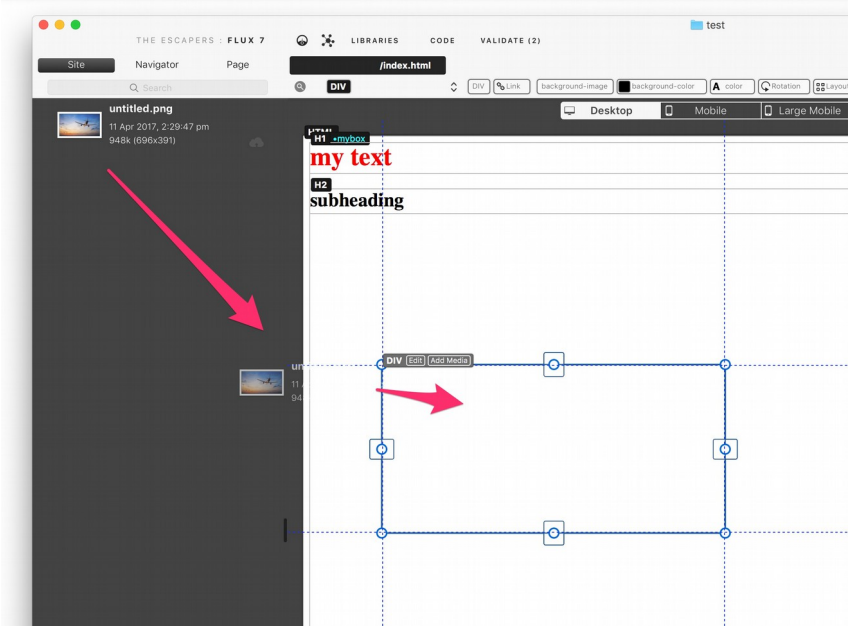
Images

The **IMG** element is one of the most used in web pages, it's the simplest way of displaying an image.

It's worth noting that **IMG** elements do not allow child elements, i.e. you cannot have another element inside an image.

Adding an image to your page

If the image you want to use is already in your project, then simply select the element you want to contain the image, and drag from the ***Site Manager*** to the page.



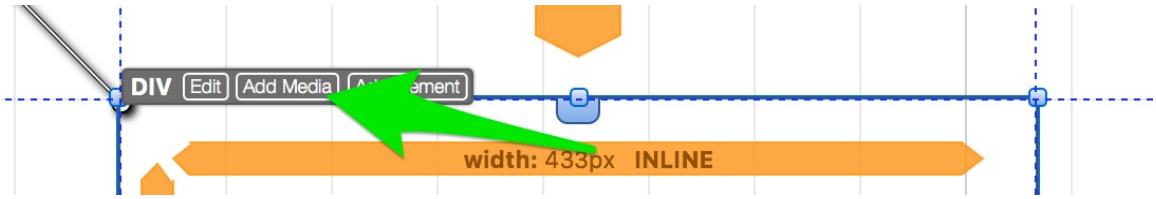
This will create the follow code for you...

```
<img src="images/aus.png" alt="" >
```

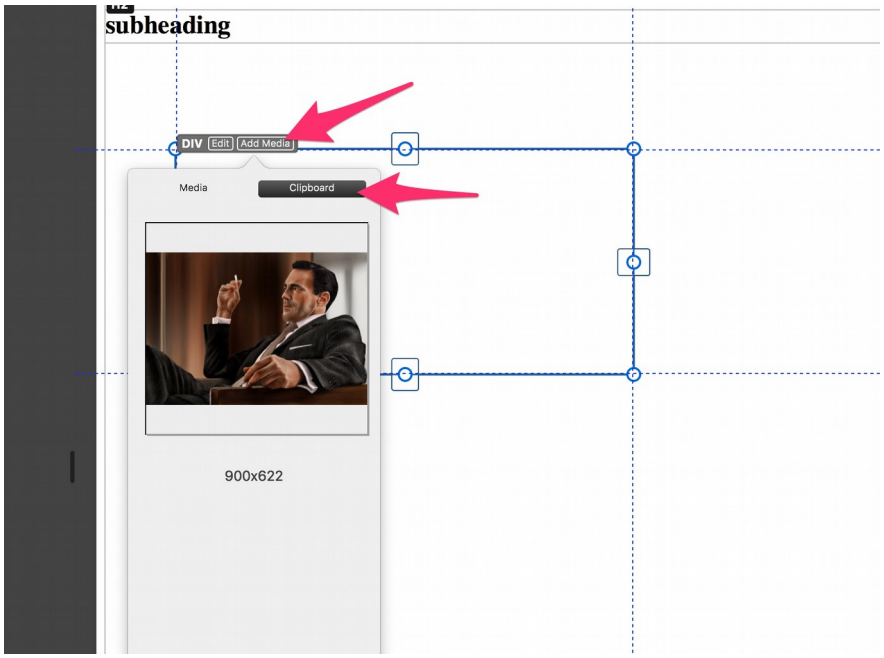
The most important attribute is *src*, which specifies the location of the image to display.

Adding an image from the clipboard

If the image you want to use is in your clipboard, copied from another app, then simply select the element you want to contain the image, and click the *Add Media* button on the the tag.



You can then use the 'Clipboard' section of the popover to insert an image from the clipboard. You will then be asked for a location to save that image. Make sure you choose a location that is within your project.



Changing the Image path

To change the image path, you need to change the 'src' attribute, which can be done in the ***Code Editor***, the ***Inspector*** and the ***Contextual Toolbar***.

To change the image using the ***Contextual Toolbar***, select the image, and press the button on the ***Contextual Toolbar***, and browse for the new image.

If you want to use a remote image, you need to specify a complete URL for the image, for example...

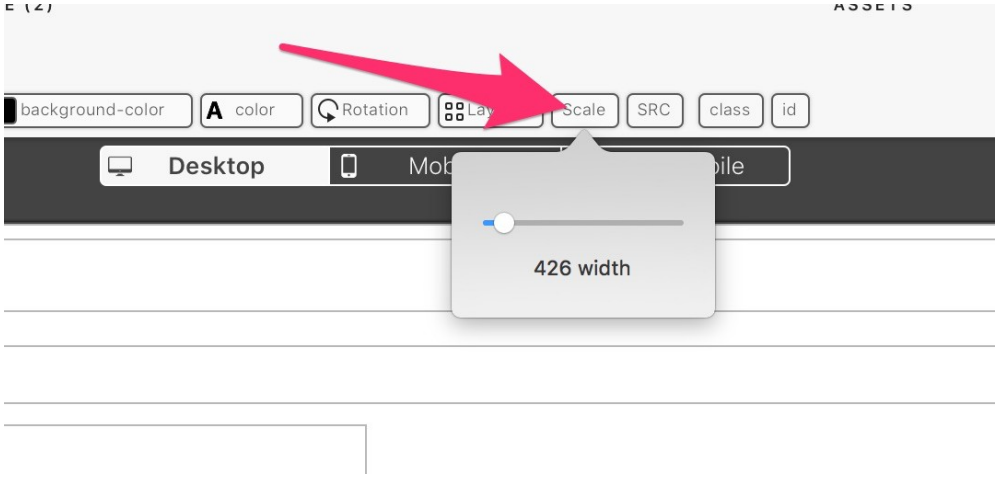
```
<img src="http://upload.wikimedia.org/wikipedia/en/3/3b/Dark_Side_of_the_Moon.png" alt="" >
```

The easiest way to do this, is to select the image, and alter the src attribute using the ***Inspector***.

Scaling the image

Flux 7 allow simple image scaling via the contextual toolbar. Please note, this doesn't alter the underlying image, it only changes the size at which the image is shown.

When an image element is selected, then a 'Scale' button becomes available.



The scaling feature uses the width attribute of the IMG element, and not CSS.

Image formats

The most flexible image format is PNG, and other than very old versions of Internet Explorer, all web browsers support it.

JPEG is also a widely compatible format, with very good compression, however, it does not support an alpha channel, i.e. transparency, so can only be used for images where you don't need to 'see through' parts of the image.

Avoid using image formats such as TIFF which are not widely support outside of the Mac.

Image Maps

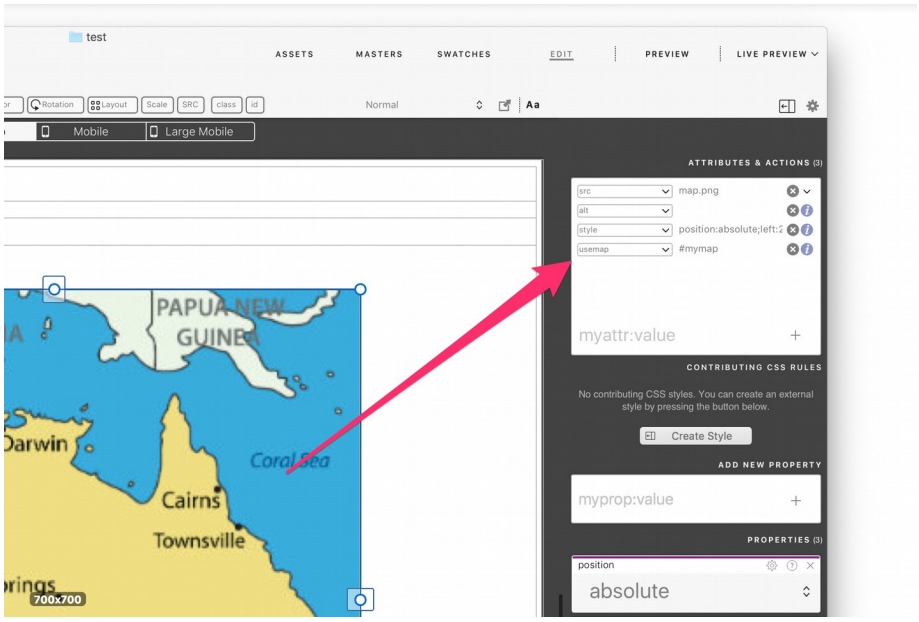
An image map usually consists of a **MAP** element, which contains one or more **AREA** elements.

The intention of an image map is to provide an easy way of linking various parts of an image without dividing the image into separate image files, they can also be used to create links which are non-rectangular.

Creating an image map

The first thing to do is to insert an image, this is explained in the 'Images' section of this manual. Maps can only work with **IMG** elements, not **DIVs** with background images.

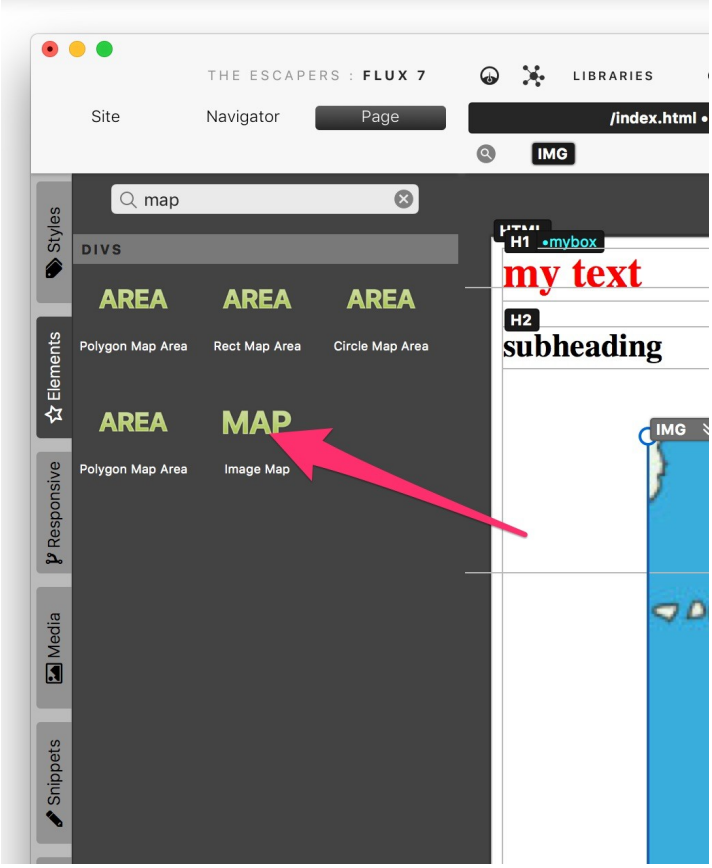
We now need to set the 'usemap' attribute of the **IMG** element using the Inspector.



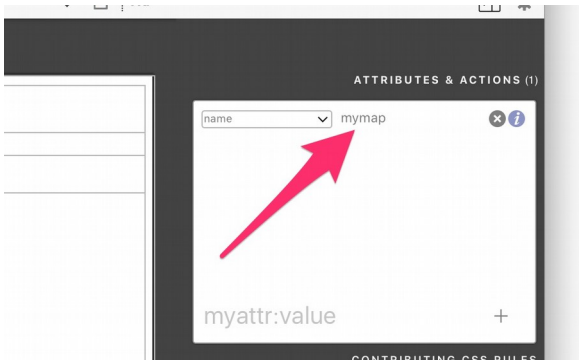
Ensure your name starts with a '#' character. In this case, let's call it '#mymap'.

We now need to create a **MAP** element. It doesn't matter where we create it on the page, **IMG** elements do not permit child elements, so although it may seem logical to put the **MAP** element inside the image, we cannot. For this example, you can insert the element into the body element.

Select the **BODY** element, and insert the Image Map from the Element Library.



Once you have created the **MAP** element, we need to set the name of the **MAP** to 'mymap' (no '#' prefix this time.).



We have now created the **MAP** element, and it has example Areas within it. You can add more areas by selecting the **MAP** element and adding another **AREA** from the Element Library.

Editing Polygon Areas

Circles and Rectangles are, by nature, easy to edit (Just drag them around, and resize like any other element), polygons can be a bit more complicated.

There are two types of handles on polygon areas, the resize ones, which are solid, and the 'add' ones which are white, and have a '+' in them.

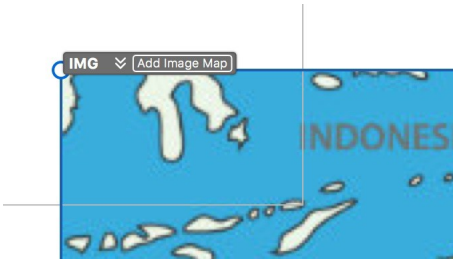
The solid ones can be dragged to reshape the polygon, and they can also be doubled clicked, which gives you the option to remove that point of the polygon.

The '+' handles are simply clicked, to add a new point to the polygon.

By adding and manipulating the points, it's easy to create very advanced polygons with no knowledge of coding.

Easily add an Image Map

Since Flux 7, you can also add an Image Map by clicking the 'Add Image Map' toolbar item that is available on all IMG elements.



Links

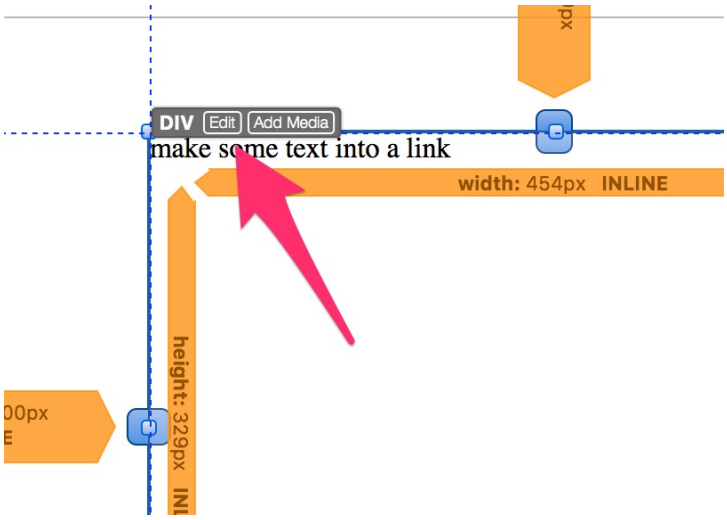
The A element is used to create links to other parts of your website or different remote URLs.

Creating a link on an element

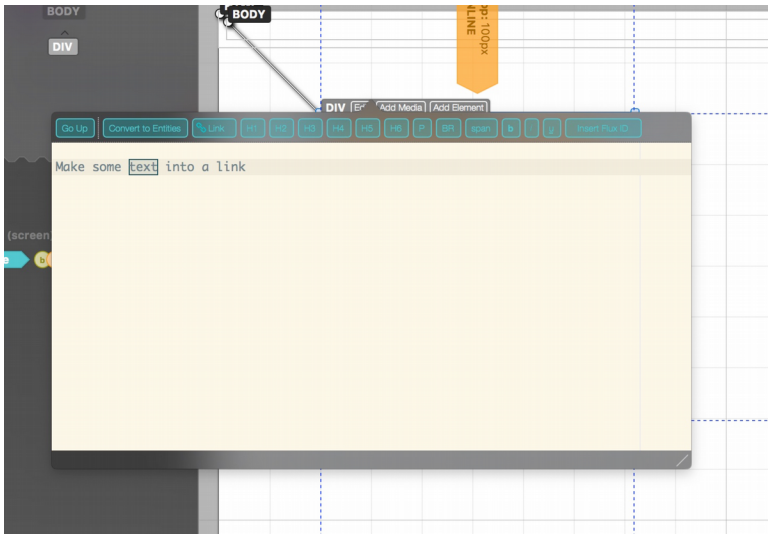
If you want to make an image, or any other element into a link, click on the element to select it, and press the *Link* button on the ***Contextual Toolbar***.

Creating a link on a text selection

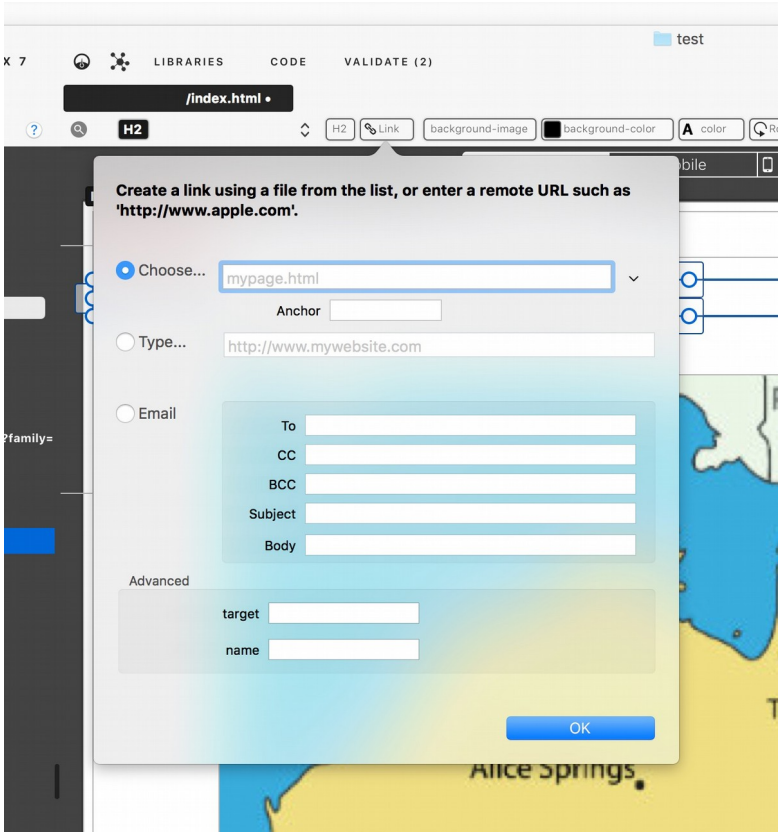
Select the parent element of the text you want to link.



Press the *Edit* button, select the text you want to make into a link and press the *Link* button.



The Link Dialog



The dialog above gives you three different options...

Choose a destination

The pop up allows you to choose any document in your project to link to, you can link to another HTML page, or any other type of file.

Type a destination

The second option allows you to type a link destination, i.e. 'http://www.apple.com'

Email

The third option allows you to specify a template for sending an email. When this link is clicked, the email template will be opened in the default Mail application.

Note : Many users do not use a Desktop email application, so remember to provide an alternative when using this option.

Google Adsense

AdSense is an ad serving application run by Google Inc. Website owners can enroll in this program to enable text, image, and video advertisements on their websites.

Inserting the code

Obtain your advertising code from your Adsense account, it'll look something like this...

```
<script type="text/javascript" ><!--

<google_ad_client = "pub-9752748337765590";
/* 300x250, created 09/02/10 */
google_ad_slot = "4445662943";
google_ad_width = 300;
google_ad_height = 250;
--><
</script>


<script type="text/javascript"
src="http://pagead2.googlesyndication.com/pagead/show_ads.js">
<script>
```

Simply paste this code in your element using Flux's Code Editor. It's a good idea to use a to contain the code, to keep it seperate, and to allow you to move the ad around the page.

You won't see the ad display in Flux's Edit view.

Accessibility

People visiting your web pages may be partially sited or blind here are some ways you can improve your pages to give them a better user experience. Here is a little insight into how Screen Readers work and what you can do as a designer to help.

The Image 'alt' attribute

By default when you add an image to the page in Flux it Generates an attribute for you. All you need to do is add some text.

```
<img src="images/FluxScreenSnapz040.jpg" alt="Image showing where to find alt tag." >
```

A little thought about the text you use can make a lot of difference to a partially sighted or blind person. Img 001 says nothing where as happy clown riding unicycle whilst juggling paints a picture. The Screen Reader reads this text aloud so it is worth putting a little thought into what you write.

Code Order

In short, code order matters when building web pages.

Beginners using absolute positioned containers tend to build the page out of order, Screen Readers read text in code order. If the page is constructed with little or no thought about the order of the page then what is read to a blind person can make little sense. Using Floating Quick Div,rather the Quick Div can really help the beginner keep page order on track and more importantly help with an elastic design, see next section.

Flexible Layouts

Within most browsers the user can choose the minimum font size and can increase font size across any page. In Safari set 'Zoom Text Only' then zoom in and out. Partially sighted people use this to help them view your page. Designing a page without elasticity to accommodate this is failing to take into account other peoples needs, if you really need an element to be an exact size then you can set the overflow to scroll.

DOCTYPEs

The DOCTYPE declaration must be the very first thing in your HTML document, before the 'html' tag.

The DOCTYPE declaration is not an HTML tag; it is an instruction to the web browser about what version of HTML the page is written in.

Example DOCTYPE

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
```

The are many DOCTYPEs in the ***Snippets*** library ready for use.

A DOCTYPE (short for "document type declaration") informs the HTML validator which version of HTML you're using, and should appear at the very top of every web page. DOCTYPEs are important, without one, the results across web browsers will be quite varied.

The DOCTYPE below is for HTML5, unless you are creating a website targetting old browsers, there is rarely any reason to use anything else.

```
<!DOCTYPE html>
```

Flux will put in DOCTYPE for you in new HTML files, and it's easy to change using the ***Code Editor***. If you don't enter a DOCTYPE, a web browser will usually default to 'Quirks mode', which can result in strange bugs, and odd rendering of pages.

SASS

SASS is a scripting language that is interpreted into Cascading Style Sheets (CSS).

SASS is a pre-processor that creates CSS stylesheets.

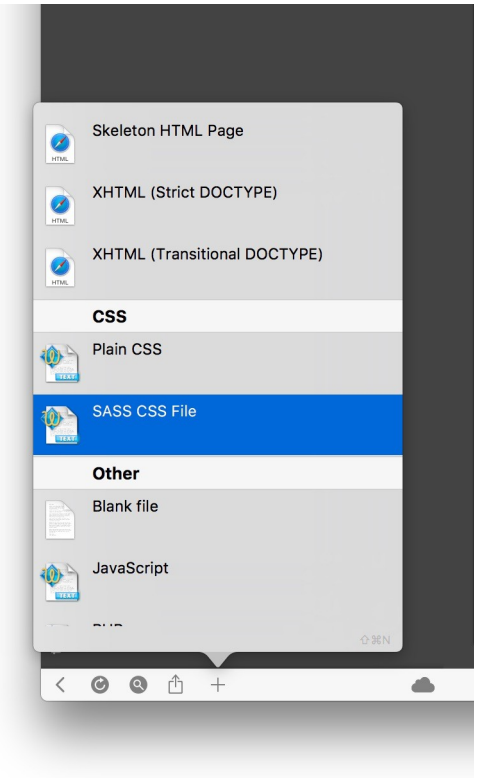
SASS is useful for creating very complex CSS stylesheets, it supports useful programming constructs such as variables.

Flux uses the .scss style of SASS stylesheets.

Please note SASS is for advanced CSS users, it's complex and adds another learning curve to HTML and CSS, if you are a beginner to Flux or CSS, then it's best to learn CSS first.

Creating a new SCSS file

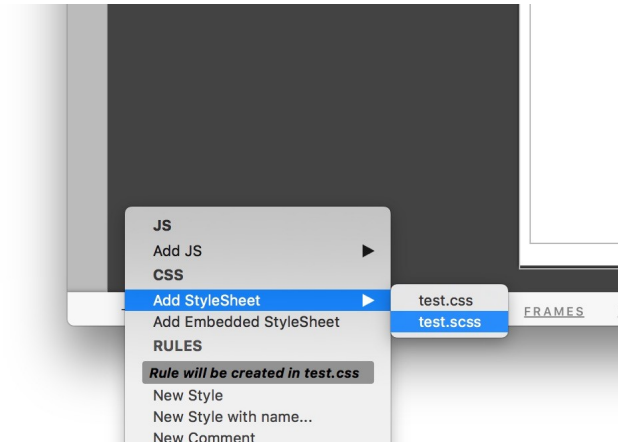
To create a .scss file, either copy an existing file into your site, just like any other file, or create a new one using the + button at the bottom of the ***Site Manager***, and choose 'SASS CSS File' from the list.



Adding a SASS file to your page

Adding a .scss file is a little more complex than adding a .css file, but Flux can do the hard work for you.

Simply chooses your .scss file from the 'Add Stylesheet' submenu, which is acessible from the button at the bottom of the ***Style Navigator***



Flux will add code like this to your page...

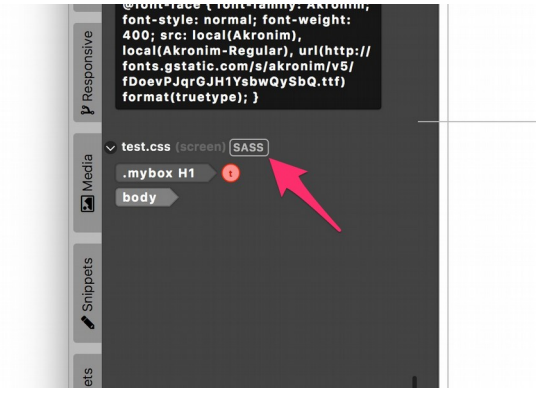
<link href="test.css" rel="stylesheet" media="screen" type="text/css" data-flux-sass="test.scss" >

The attribute is just like a regular .css import, but the CSS file will be generated by SASS, and not created by you, the author.

The important part of this code is the attribute, it tells Flux that the CSS is to be generated by the specified .scss file. If this attribute is removed, Flux will not acknowledge that the CSS is SASS controlled.

Editing SASS

Once the SASS link is added to your page, the stylesheet will appear like the one below.



Clicking on the stylesheet will select the .css stylesheet like any other stylesheet, however, the code will not editable, as the .CSS file is auto-generated, so any changes would be overwritten the next time the SASS is compiled.

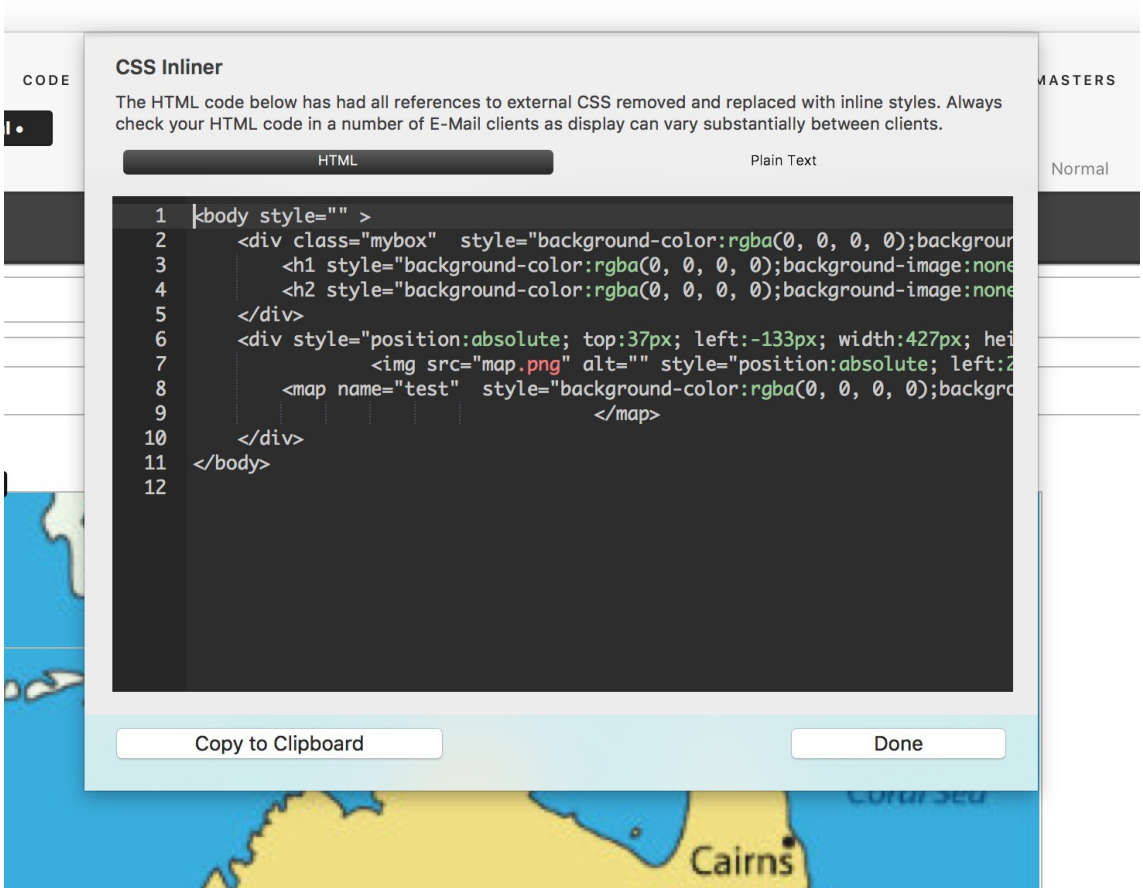
To edit the SASS file, click in the SASS icon.

HTML E-Mail

Flux 7 includes a feature that can help preparing HTML E-Mails. To prepare your page for E-Mail, choose ‘Create E-Mail from Page…’ from the ‘Page’ menu.

You will be presented with a panel that includes a HTML, and Plain text representation of your page.

The HTML in this panel is different from a regular page, as it has all external styling removed, as this is not permitted in HTML E-Mail. All the external styles have been converted to inline styles. This is not usually desirable in HTML, but is essential for sending HTML E-Mail.



It’s very important to check how your HTML E-Mail will be received on a number of E-Mail clients, as many have different capabilities.

Images can be especially troublesome, and it’s often best to avoid using images where possible. When using images, it’s best to host the images on an external server, and reference them directly from there. For example

<img src="http://www.theescapers.com/images/flux.png">

It's critical that this image exists on your server.

Responsive Design

Flux 7 supports Responsive design using the following libraries

Bootstrap

Skeleton

You can use any other responsive framework if you want , but you won't be able to take advantage of the new responsive features in Flux 7.

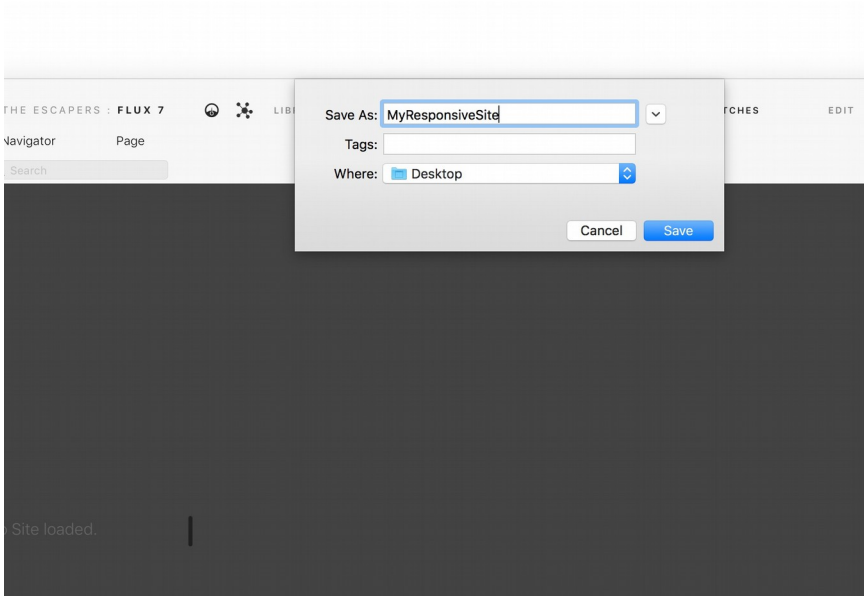
Skeleton is the simpler of the two, and The Escapers recommend it for most purposes.

What is Responsive Design?

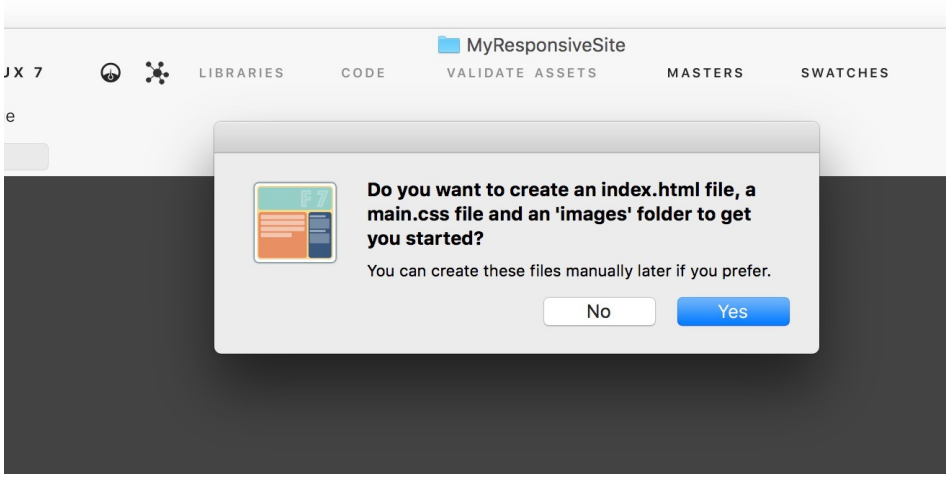
Responsive Design is an approach to creating web pages aimed at crafting sites to provide an optimal viewing and interaction experience across a wide range of devices (from desktop computer monitors to mobile phones).

Creating a Responsive Site

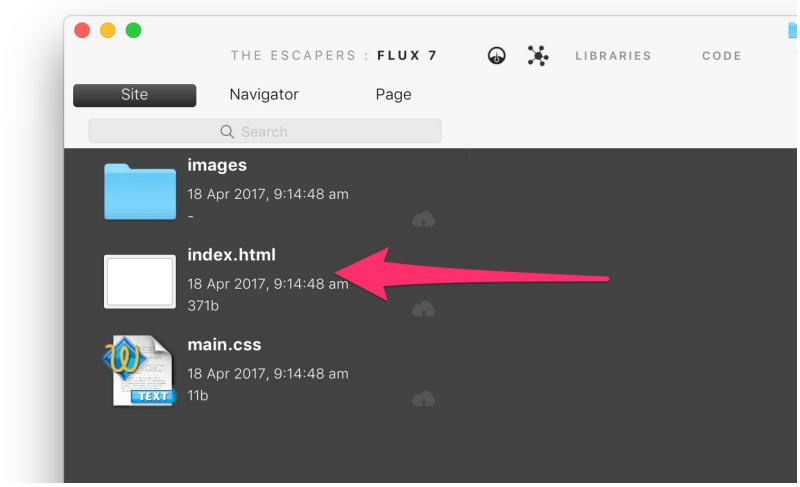
Create a new project in Flux by choosing 'New' from the 'File' menu. Enter a name for the Site when requested.



Flux will offer to create a few files to get you started. Choose 'Yes'.

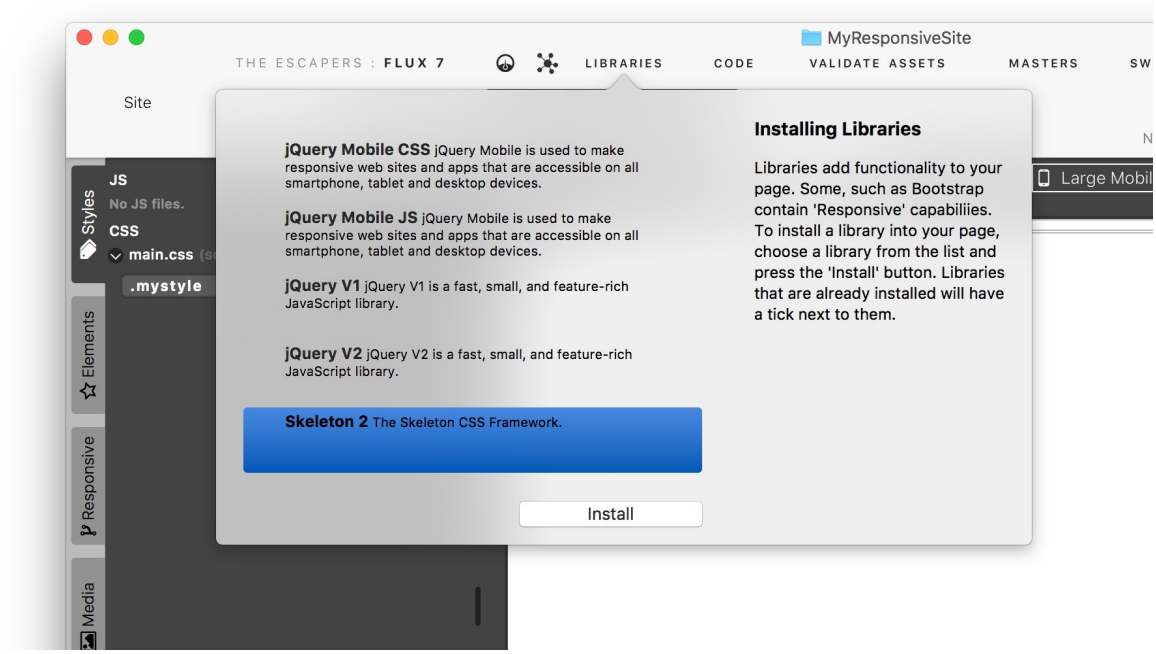


Double click the index.html file in the Site Manager to open the page in either Code or WYSIWYG mode, for this example, choose WYSIWYG.



Currently, the page has no responsive capability.

To add a Responsive library, click the ‘Libraries’ button in the main Flux toolbar. From the popover that appears, install the ‘Skeleton’ library by selecting it in the list and pressing the ‘Install’ button.

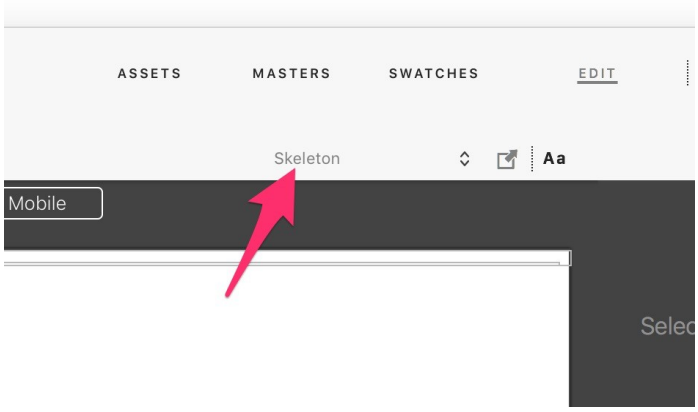


Once you have installed Skeleton, you'll see a lot of CSS styles in the Styles list. These are provided by Skeleton.

Your HTML code has been modified to include the Skeleton library.

```
12
13  <link data-fluxlibrary="Skeleton 2" rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/skeleton/2.0.4/skeleton.min.css">
14
```

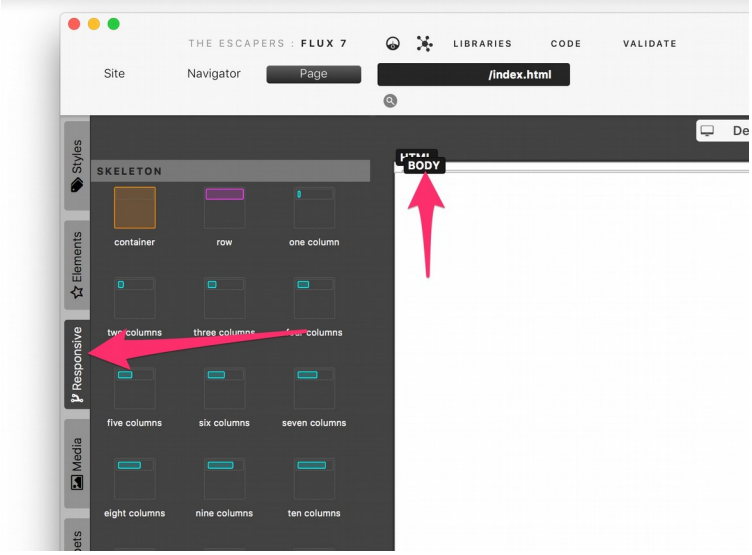
You will also see the control at the top right of the page now says 'Skeleton' to let us know we are working on a Skeleton page.



Most Responsive frameworks work with a few basic types, *containers, rows, and columns.*

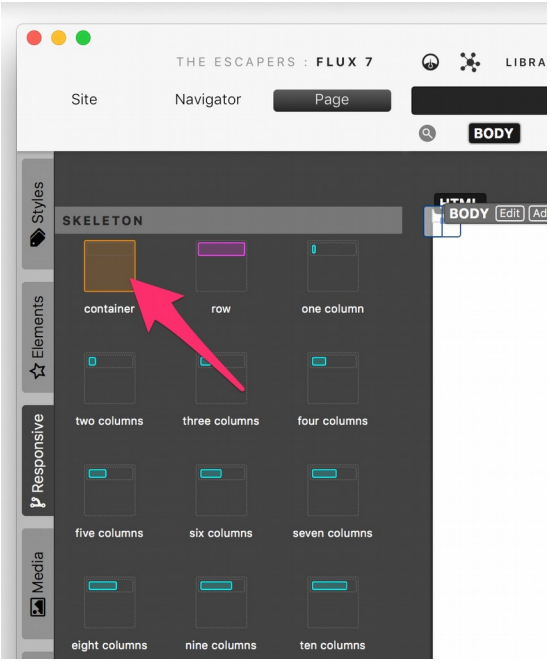
First, we'll insert a *container*.

We need to insert the container into the **BODY** element, so select the **BODY** element on the page and press the 'Add Element' button.



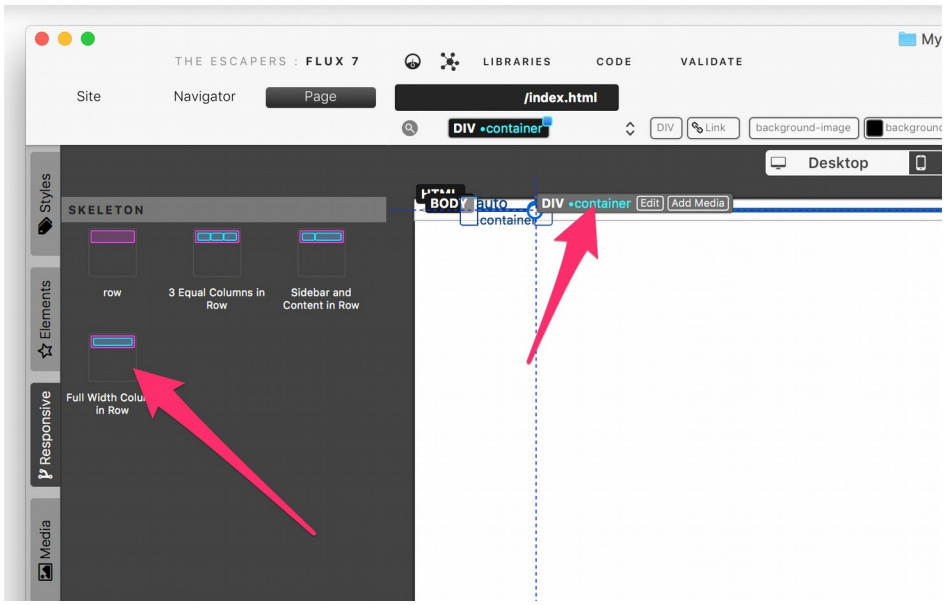
Make sure you are in the 'Responsive' area of the 'Page' tab.

Although Flux can insert a full sample layout, as a learning exercise, we'll do it manually, so double-click 'container'.



Flux will insert a Skeleton container into the **BODY** element.

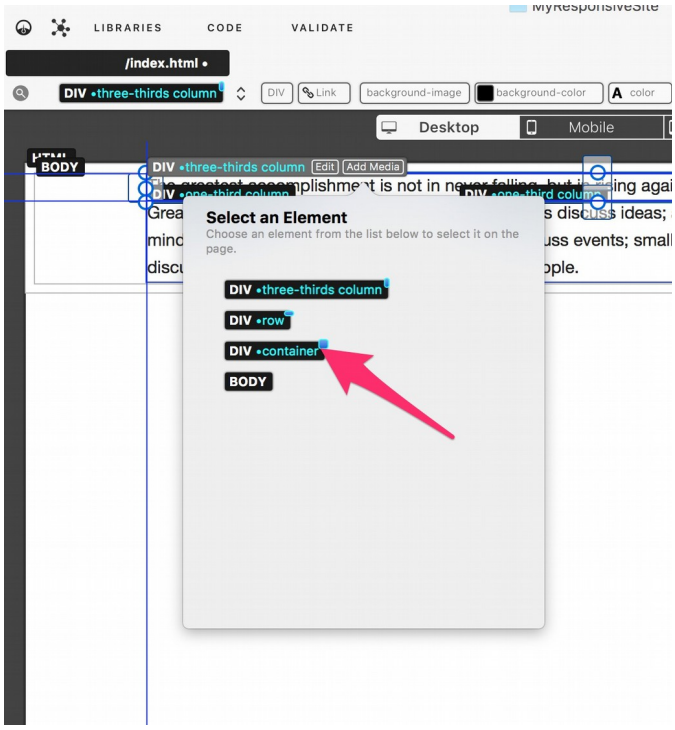
Let's now insert a *row* into that container. Select the newly created container, and double click the desired row from the Responsive panel.



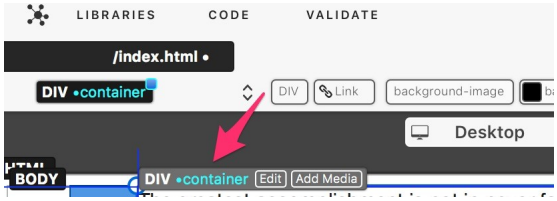
We will now insert another *row*, so re-select the *container* object**. It's very important that you select the container object to insert the next row.**

You may find it difficult to select the container object now, as the row is obscuring it from view. Don't worry, Flux provides a solution to this issue.

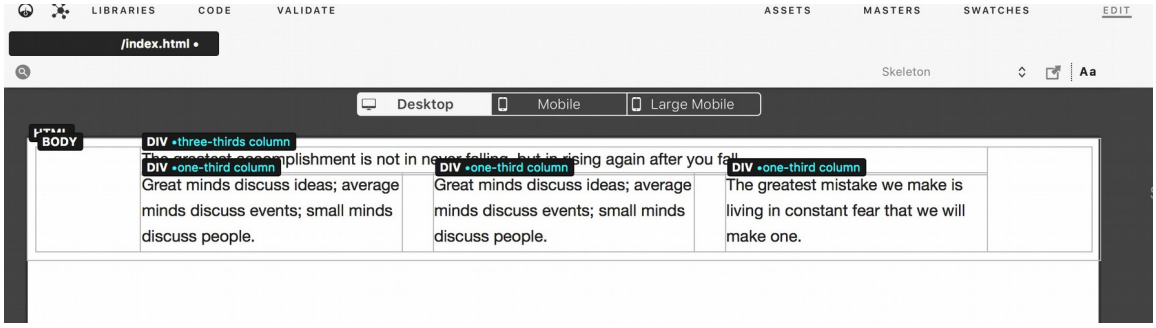
Simply click and hold on the newly created row, and Flux will display a popup containing all elements under the pointer. From this popup, choose the 'container'.



You will now see the container is selected again.

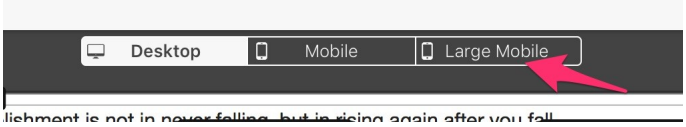


If we add another row, but this time, one with three columns, we now have a page that looks like the one below.

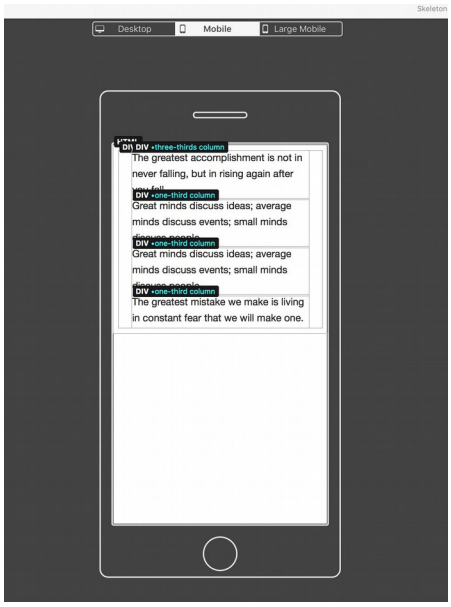


We now have a very basic responsive page.

To see how this page appears on a mobile phone, we can choose a ‘Device’ from the toolbar above the page.

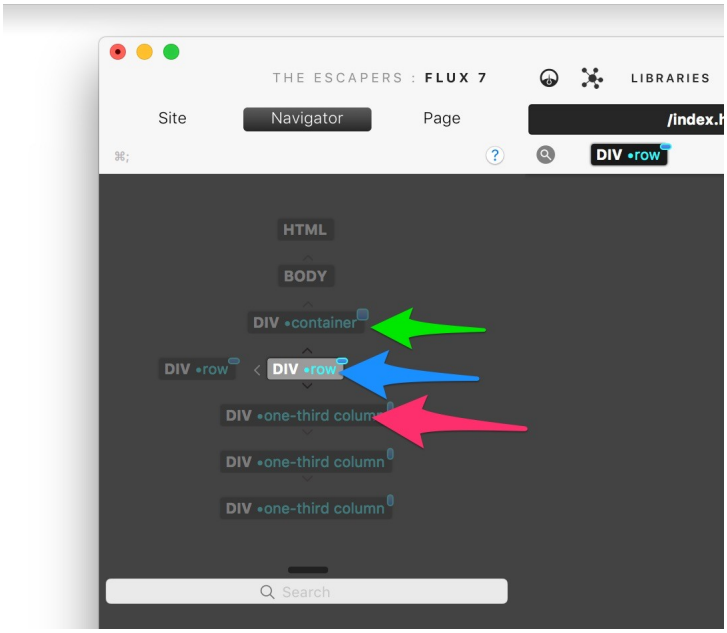


The page will tile itself correctly to fit on a mobile phone screen, and still render correctly on a full size browser. You can also edit the page in this mode.



Responsive elements can be visually identified in Flux with a little token on the main tags.

Containers are identified with a small square (green arrow), rows are identified with a horizontal box (blue arrow), and columns with a vertical box. (pink arrow).

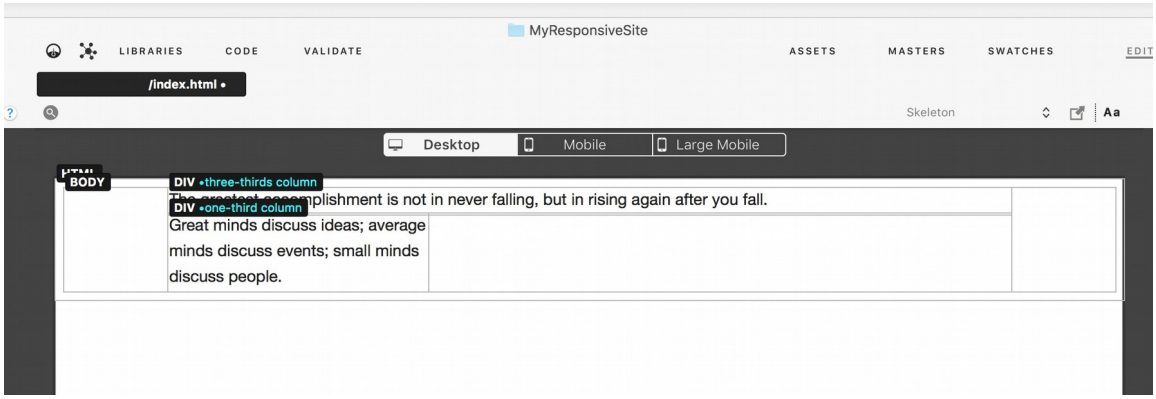


Using Offsets

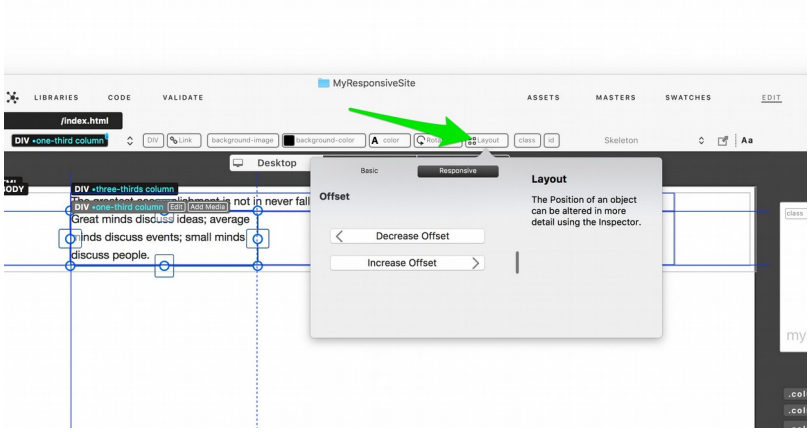
Most responsive frameworks support offsets for use when a row full of columns doesn't suit the design. Flux supports offsets for Bootstrap and Skeleton.

To try this, delete two of the three columns from the second row, it doesn't matter which ones, but make sure you leave one.

We are now left with one column, in the second row, and we can offset it to appear in a different location within that row.



Select the *column* object, and press the 'Layout' button in the toolbar.

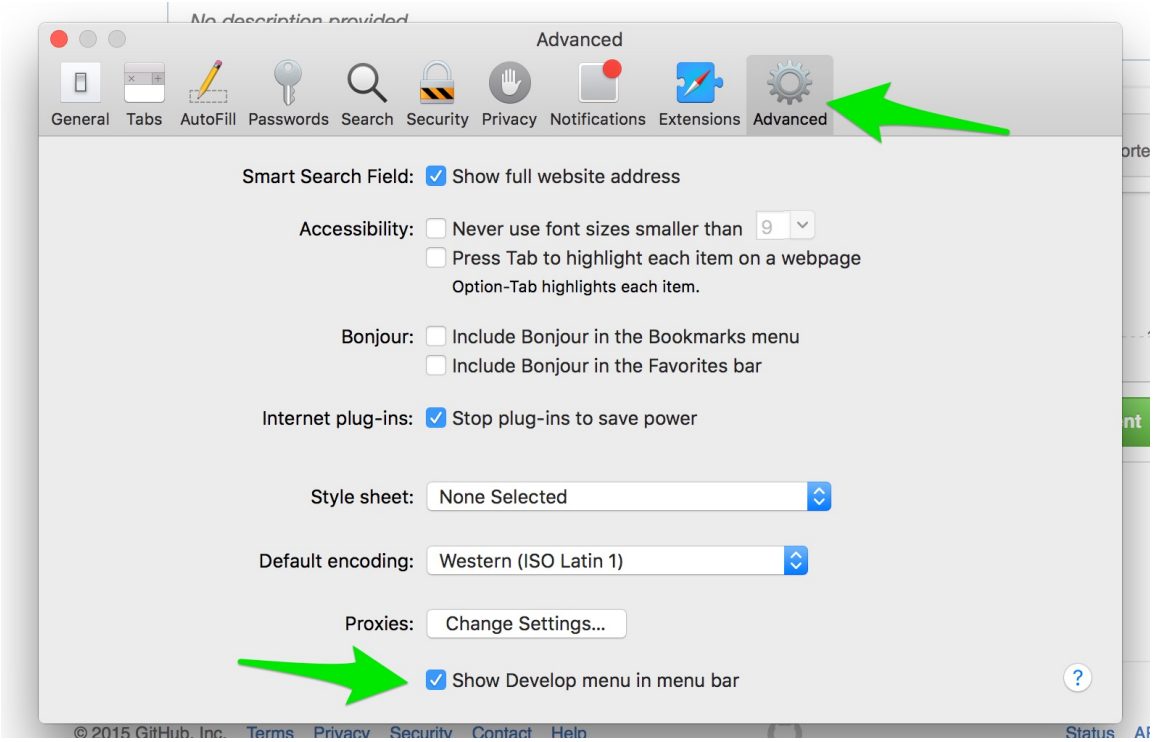


In the 'Responsive' area of the popover, you can increase or decrease the offset to alter the position of the column.

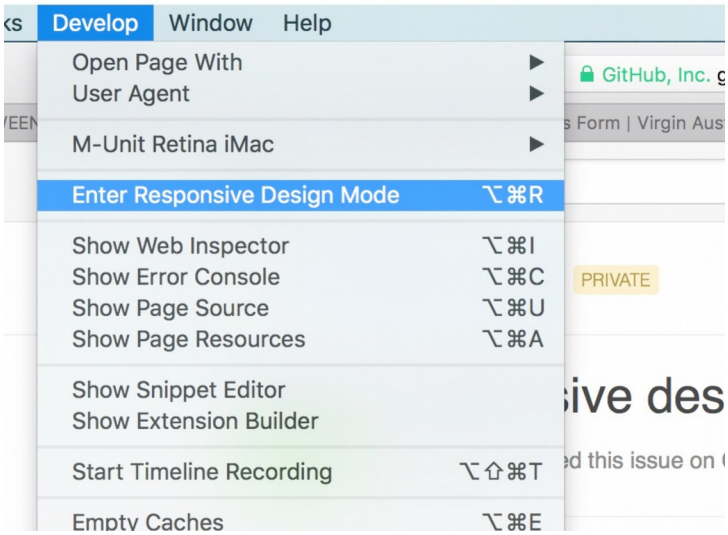
With careful use of responsive frameworks, it's possible to create websites that are first class citizens on both desktop and mobile platforms without needing to create separate pages for each.

Previewing Responsive sites

In addition to the built-in Responsive Preview in Flux, Safari has a very good Responsive preview, but it needs to be enabled first in Safari Preferences.



Once this has been enabled, you can use the 'Enter Responsive Design Mode', in the 'Develop' menu.



Remote Working

Flux supports ***Remote Working***, which allows you to edit sites directly over FTP or SFTP. You don't need to download or mirror the site first.

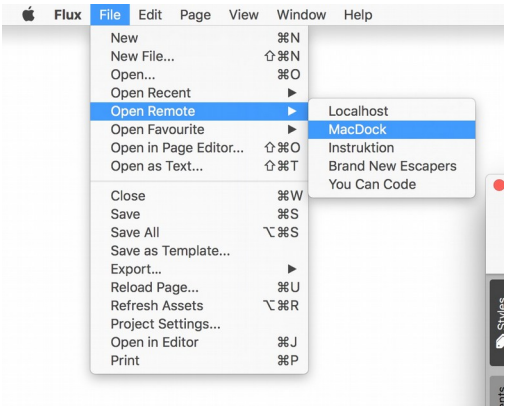
Setting up a Site for Remote Working

To setup a site for remote working, you need to setup a Bookmark in the Flux Preferences window. If you have already setup a bookmark for the site, then you don't need to create another one.

Opening a Site for Remote Working

There are two ways to open a site for remote working.

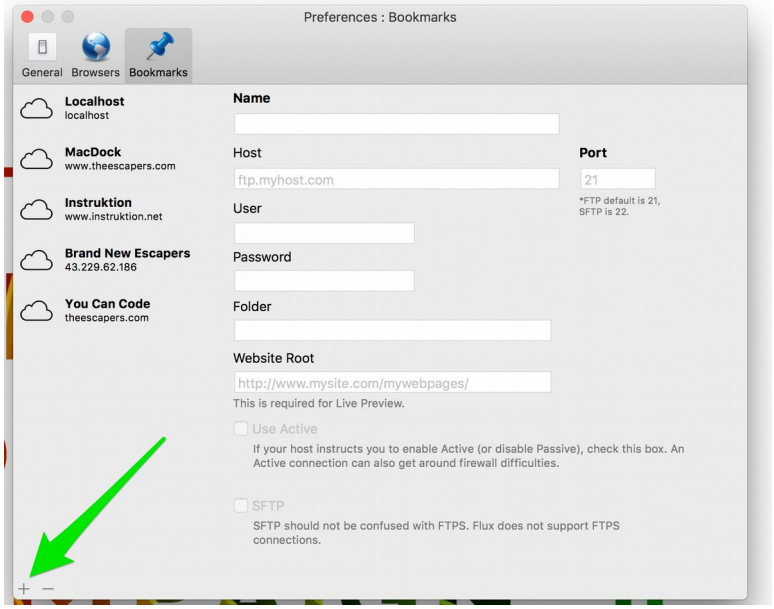
You can open a remote site from the **File** menu.



Or from the Welcome panel…

Simply double-click the site you wish to open. You can also create a bookmark from this window.

Creating a bookmark



Add a bookmark in Preferences using the '+' button pictured in the image above.

Complete the details, your ISP can supply this information, please note Flux does not support FTPS connections, only FTP and SFTP.

Understanding Remote Working

Remote Working edits files directly on your website, so changes are happening 'live'. Once you save a file, your remote file has been changed, there is no auto-backup, and it's important to note that the ***Rollback*** does not work in Remote Working, any backups need to be made manually.

You are working off a remote site, this is by magnitude slower than working from a local disk. Opening files will take much longer as Flux not only has to read the HTML file from the remote site, but also all associated files such as CSS files and images. Saving files will also take longer.

If you use the 'Open With' feature from the Flux ***Site Manager***. Flux will endeavour to upload any changes you make in the external application, but cannot guarantee the functionality of 3rd party applications.

Backup

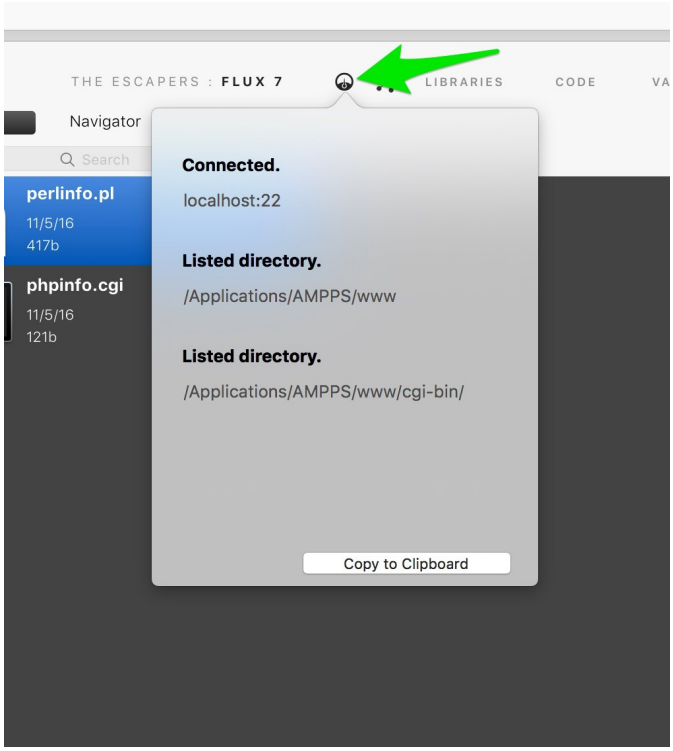
Flux will not backup files for you, you are working on a live site, it is crucial you have a disaster recovery plan, which should include a 'worst case scenario'. The Remote

Working feature has been in testing for a long time, but things can and will go wrong, a power cut during upload can easily wipe out a file.

Backup.

No really. Do it.

Solving problems

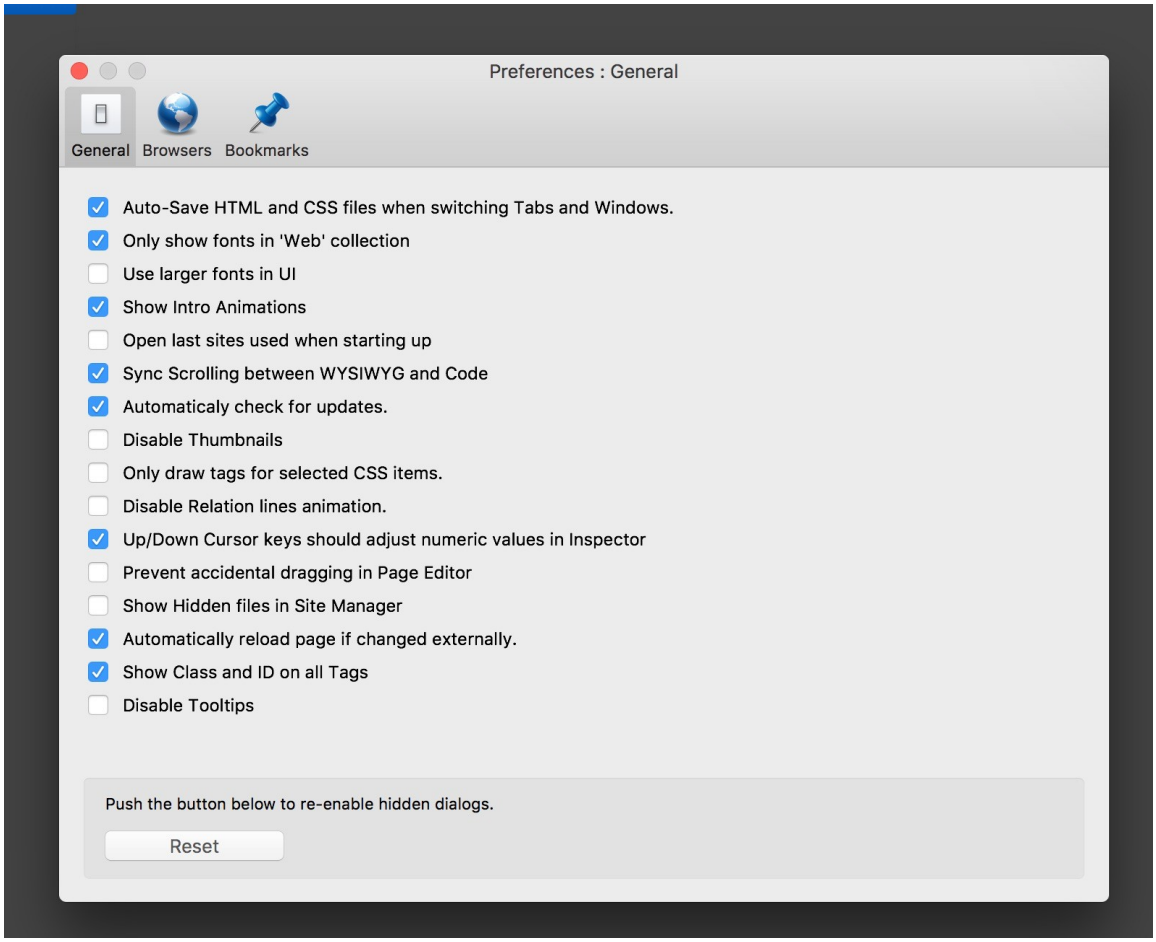


Take care if you send the log, as it may include sensitive information such as usernames and passwords. If the Debug option is not selected, no sensitive information will be included.

If you need help from The Escapers with Remote working with a specific problem, please include a copy of log in your email, a ***Copy to Clipboard*** feature is included for your convenience.

Preferences

Preferences is accessed from the Preferences... menu entry in the File menu.



General

Flux is configurable to your needs, it has many options for customization.

Autosave HTML and CSS files when switching tabs and windows Since a web page is built of many files, it's important that when you are previewing pages, that all the files you are working on are saved, this option will do this automatically.

Although Flux 7 keeps revisions of files automatically using the ***Rollback*** feature, we still strongly recommend you keep regular backups of your sites.

Only show fonts in Web collection To help manage the size of your font menus, you can choose to only show fonts in the 'Web' collection, rather than all fonts on your Mac.

To manage the 'Web' collection, use the 'Font Book' application on your Mac.

Open last sites used when starting up If this option is checked, the site open when Flux was quit will be opened automatically the next time Flux is launched.

Automatically check for updates Flux will periodically check for updates, we don't recommend you turn this option off.

Only draw tags for selected CSS items With this option check, the ***Style Navigator*** will only draw a tag icon for the selected rule. This can assist in readability for the visually impaired.

Disable Relation lines animation If you find Flux running a little slowly, or if you find the relation animation distraction, it can be disabled here.

Up/Down Cursor keys should adjust numeric values in Inspector If this option is selected, the cursor keys will increment or decrement values in the ***Inspector*** fields.

Prevent accidental dragging in Page Editor If this option is selected, page elements will be 'sticky', to prevent accidental movement.

Show hidden files in Site Manager When this option is selected, files that begin with a '.' will be shown in the ***Site Manager***.

Automatically reload page if changed externally When this option is selected, if a page open in Flux is changed using an external editor, Flux will recognize this and reload the page automatically to reflect those changes.

Show Class and ID on all tags When this option is selected, all page elements will display the class and id. If it is not checked, then only the selected element will display this information.

Disable Tooltips When this option is selected, Tooltips will not be used within Flux.

Browsers

You can setup your browsers for previewing pages here. Flux can automatically find all the Browsers on your Mac, but if you need to add some manually, simply drag and drop the Browser from the Finder onto this list.

Bookmarks

This is where your Bookmarks for FTP/SFTP are managed. If you want to publish, upload or download a file from a remote site in Flux, you'll need to configure it here.

It's very important to enter all the details correctly, if you cannot connect to a site, ensure it's all correct. If you can connect with a different client, but the same details, then please contact us@theescapers.com for assistance.

Tips

Make Flux run quicker

With complex sites, large CSS files, or large HTML files. Flux can sometimes run a little slowly.

Collapse unused Stylesheets

If you have very large CSS files, collapsing them in the Style Navigator can make Flux run substantially quicker. You can collapse (or expand) them all at once by holding the Option key when clicking the Expand/Collapse arrow.

Google Fonts

Google Fonts are downloaded remotely when previewing pages and opening new pages, which means a 'lag' can happen. If using Google Fonts, Flux will run faster if you temporarily change the font to something local, so the font doesn't need to be downloaded, and only change it back when ready to publish your site.

Turn off unused overlays

With large/complex HTML files, turning off features such as 'Frames','Spans' on the toolbar at the bottom of the window will make Flux run faster.

Close unused tabs

Every tab open in Flux uses valuable RAM. Close tabs you are not using.

Troubleshooting

Flux can't find the elements in my code

Sometimes Flux will report that it's unable to find the corresponding code for a selected element. For Flux to perform WYSIWYG alterations, it needs to be able to do this.

To find elements, Flux uses a variety of methods, such as looking for unique 'id' attributes, another method is to look for 'data-flux-id' attributes.

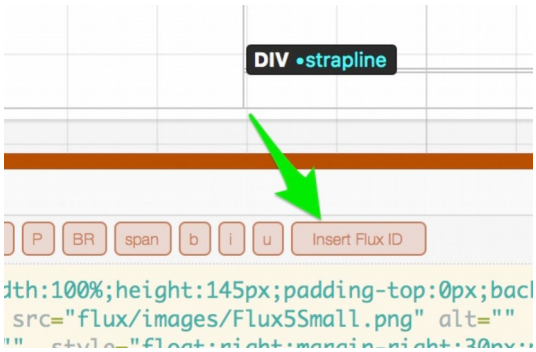
So, if you have an element, such as…

```
<p>my unfindable element</p>
```

You could add this attribute, like this….

```
<p data-flux-id="uniquecode">my unfindable element</p>
```

Once you have done this, Flux will be able to find the code.

The advantage of using this attribute, instead of, say, ‘id’, is that it has no other impact on your page.

The Code Editor has a feature to generate these attributes for you.