



Responsive Web Design

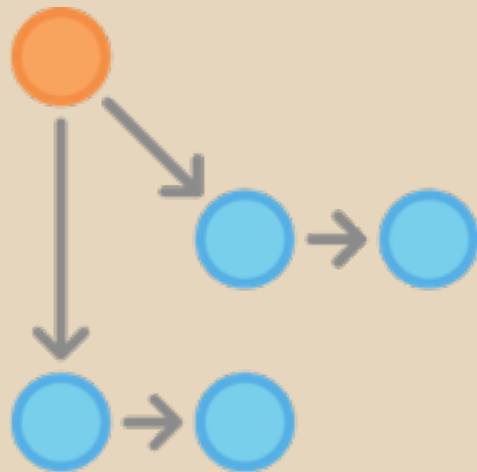
Lecturer: Ming Zhen, Cai

為何要RWD?

1. 提供使用者更好的瀏覽體驗
2. 開發一次，用在各處，讓開發不同裝置的網站變得簡單與迅速



1. Meta tag: Viewport
2. Grid System
3. Media Query
4. Responsive Grid System
5. Responsive Image





Meta tag: Viewport

Problems?

1. 一般的網頁在行動裝置看到的長相
2. 字太小，網頁寬度太寬需要自行縮放
3. 看到這樣只有頭痛！



Solution: Viewport

- 使用meta tag中的viewport
- 定義viewport:
告訴瀏覽器，
內容寬度是裝置的寬度
縮放比例為1.0
- `<meta name="viewport" content="width=device-width, initial-scale=1.0">`



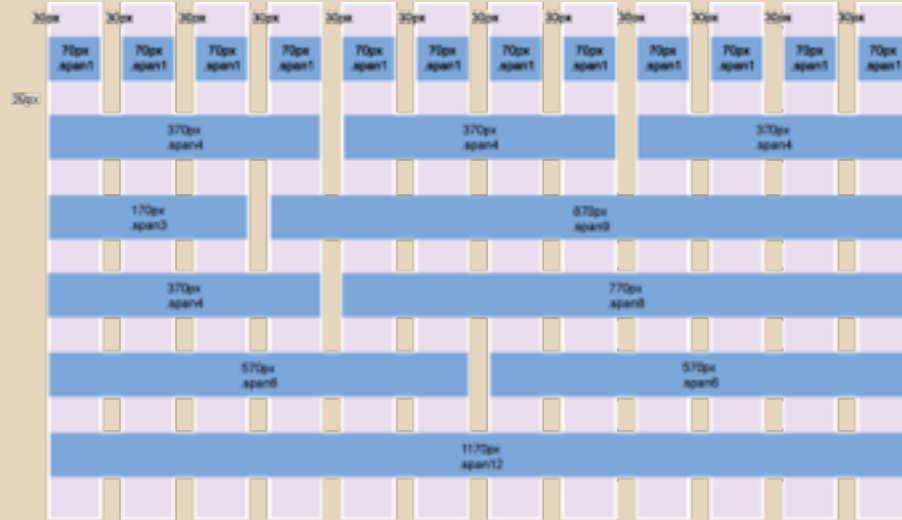
W3C Meta Tag Viewport

屬性	值
<i>width</i>	可以填數字或 <i>device-width</i>
<i>height</i>	可以填數字或 <i>device-height</i>
<i>initial-scale</i>	<i>0.25~5.0</i>
<i>minimum-scale</i>	<i>0.25~5.0</i>
<i>maximum-scale</i>	<i>0.25~5.0</i>
<i>user-scalable</i>	1或 0

W3C Meta Tag Viewport

- 不是上頁所有屬性都被瀏覽器支援，使用上要注意
- width不一定代表device-width
譬如retina的顯示器，width=960，device-width=320





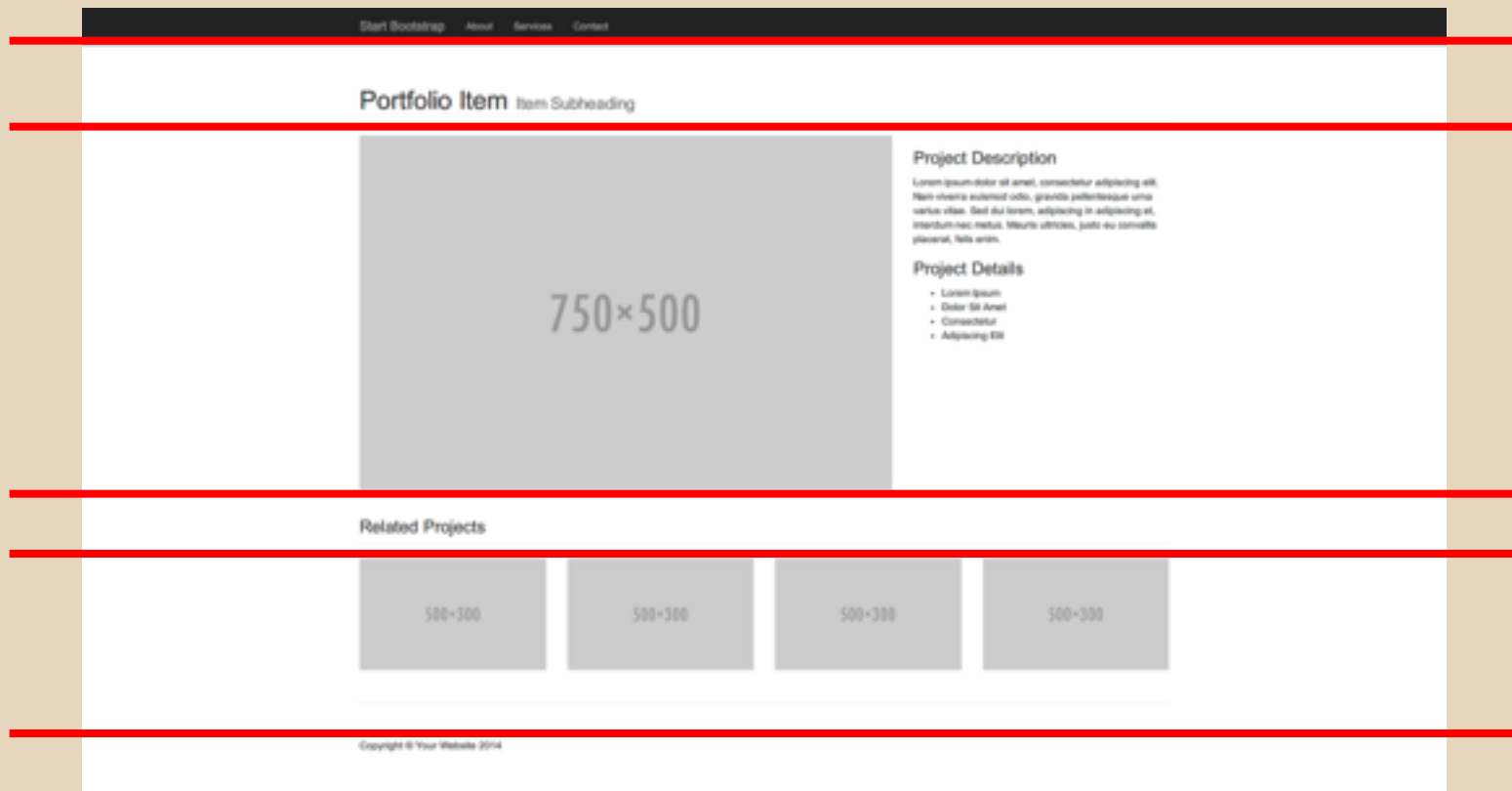
Grid System

Bootstrap Grid System

- 由row與col組成，每個row最多可放總和為12單位的col
- 格式col-[screen size]-[1-12] <http://960.gs/>

.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1
.col-md-8								.col-md-4			
.col-md-4				.col-md-4				.col-md-4			
.col-md-6						.col-md-6					

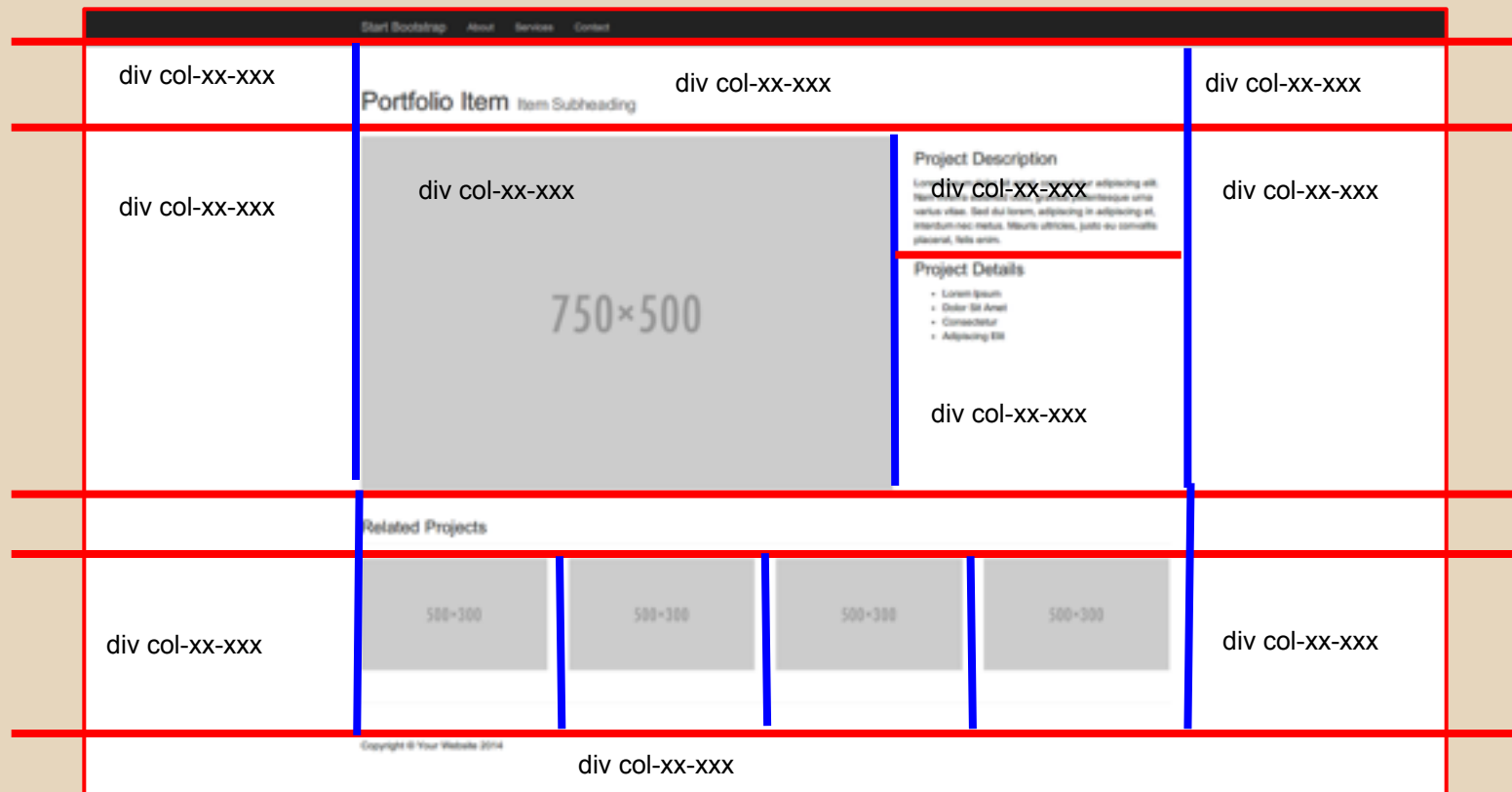
Step1: 規劃 Row



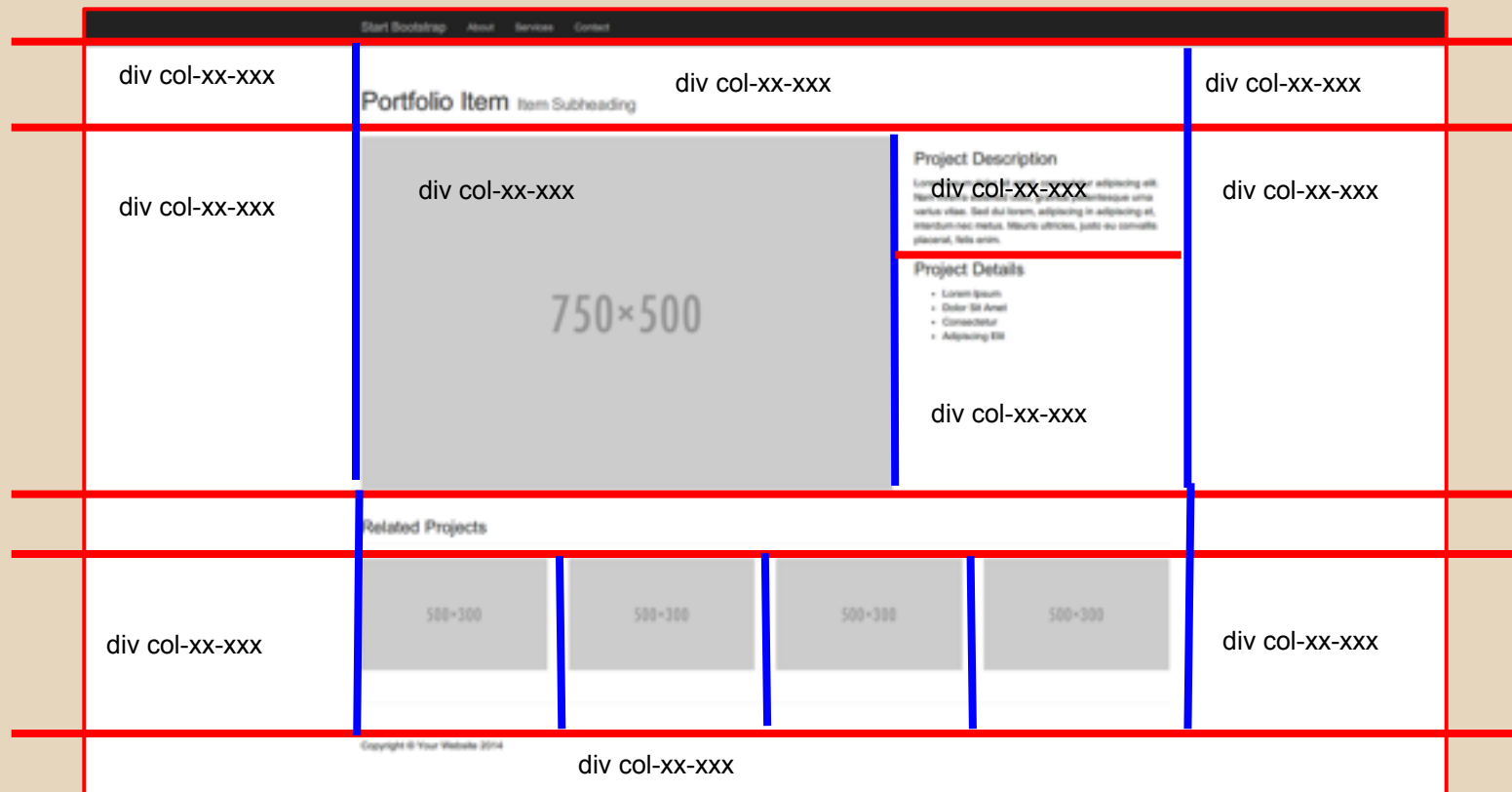
Step2: 規劃 Column



Step3: 規劃div與設定比例



Step4: 根據div 寫CSS 廢話！



- 標籤的屬性(Attribute)也可以選擇
CSS3中提供了新的選擇法
- 記號：
^ 代表開頭，\$ 代表結尾，* 代表包含
- [attribute^=value]: 選取屬性以value開頭的所有標籤
- [attribute\$=value]: 選取屬性以value結尾的所有標籤
- [attribute*=value]: 選取屬性包含value的所有標籤

Attribute Selector Example

1. `a[href^="https"]`: 選取所有 href 以 https 開頭的 a
2. `a[href$=".pdf"]`: 選取所有 href 以 .pdf 結尾的 a
3. `a[href*="ntu"]`: 選取所有 href 包含 ntu 的 a

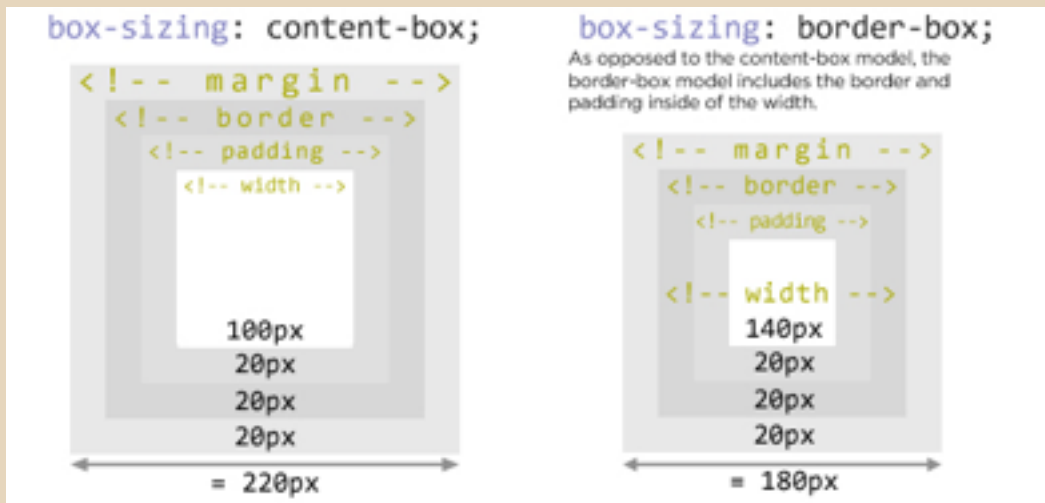


- 在被選取的tag後面插入內容
- Example:

```
p:after {  
    content:"test";  
    background-color:yellow  
}
```

AFTER

1. box-sizing: **content-box** (原本的box model)
2. box-sizing: **border-box**
(決定好大小，大小就不會因為padding等屬性改變)



1. 仿造Bootstrap，我們使用12等分column的網格系統
2. column應該是float的，並且不使用pixel作為單位
3. column與row盡量以CSS3的box-sizing做設定



基礎設定

1. 所有的tag都使用**border-box**

```
* {  
    box-sizing: border-box;  
}
```

1.

```
[class*="col-"] {  
    float: left;  
    padding: 15px;  
    border: 1px solid red;  
}
```

.col-[1-12]

```
.col-1 {width: 8.33%;}  
.col-2 {width: 16.66%;}  
.col-3 {width: 25%;}  
.col-4 {width: 33.33%;}  
.col-5 {width: 41.66%;}  
.col-6 {width: 50%;}  
.col-7 {width: 58.33%;}  
.col-8 {width: 66.66%;}  
.col-9 {width: 75%;}  
.col-10 {width: 83.33%;}  
.col-11 {width: 91.66%;}  
.col-12 {width: 100%;}
```

將全螢幕分割為12等份

1代表橫跨1等份，2代表橫跨2等份

意即：col-1為1/12，同理col-2為2/12，依此類推

.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1	.col-md-1
.col-md-8								.col-md-4			
.col-md-4				.col-md-4				.col-md-4			
.col-md-6						.col-md-6					

.row

- 每個row包含許多col，下一個row會被上一個row中的col蓋住，因為col皆為float。
- 因此需要clear屬性來使得row的左右不會有float元素
- 使用after，在row後面插入一個空白元素，並將其clear:both

```
.row:after {  
    content: "";  
    clear: both;  
    display: block;  
}
```

小試身手

- 1.把熊變成圓形圖片，
排成六張一行，且置中
- 2.把png檔的圖片用紅色虛線圈起來
- 3.在連得到的超連結圖片下面
加上“I am real!” 字樣
(注意看超連結中的網址開頭！)





Media Query

Media Query

1. CSS3中加入了Media query，可惜並不是所有瀏覽器都支援
2. 語法：**@media** 開頭， min表示 \geq ， max表示 \leq

```
@media screen and (min-width: 1200px) {  
    // 視窗寬度  $\geq$  1200px，將使用這裡的 CSS。  
}  
  
@media screen and (min-width: 768px) and (max-width: 1200px){  
    // 視窗寬度  $\geq$  768px且寬度  $\leq$  1200px將使用這裡的 CSS。  
}
```



Add Break Point

```
/* Desktop middle size */
@media screen and ( min-width:992 px ){
.col-md-1 {width: 8.33%;}
.col-md-2 {width: 16.66%;}
.col-md-3 {width: 25%;}
.col-md-4 {width: 33.33%;}
.col-md-5 {width: 41.66%;}
.col-md-6 {width: 50%;}
.col-md-7 {width: 58.33%;}
.col-md-8 {width: 66.66%;}
.col-md-9 {width: 75%;}
.col-md-10 {width: 83.33%;}
.col-md-11 {width: 91.66%;}
.col-md-12 {width: 100%;}
}
```

```
/* Mobile Device small size */
@media screen and ( min-width:768px ) and ( max-
width: 991px ){
.col-md-1 {width: 8.33%;}
.col-md-2 {width: 16.66%;}
.col-md-3 {width: 25%;}
.col-md-4 {width: 33.33%;}
.col-md-5 {width: 41.66%;}
.col-md-6 {width: 50%;}
.col-md-7 {width: 58.33%;}
.col-md-8 {width: 66.66%;}
.col-md-9 {width: 75%;}
.col-md-10 {width: 83.33%;}
.col-md-11 {width: 91.66%;}
.col-md-12 {width: 100%;}
}
```

不想學Bootstrap，還可以這樣

```
/* For desktop: */  
.col-1 {width: 8.33%;}  
.col-2 {width: 16.66%;}  
.col-3 {width: 25%;}  
.col-4 {width: 33.33%;}  
.col-5 {width: 41.66%;}  
.col-6 {width: 50%;}  
.col-7 {width: 58.33%;}  
.col-8 {width: 66.66%;}  
.col-9 {width: 75%;}  
.col-10 {width: 83.33%;}  
.col-11 {width: 91.66%;}  
.col-12 {width: 100%;}
```

```
@media only screen and (max-width: 768px) {  
  /* For mobile phones: */  
  [class*="col-"] {  
    width: 100%;  
  }  
}
```



col都要算比例超麻煩

- 有沒有類似程式的寫法，請愛用SASS、SCSS（進階班再談）

SCSS

```
1  section {  
2    height: 100px;  
3    width: 100px;  
4  
5    .class-one {  
6      height: 50px;  
7      width: 50px;  
8  
9      .button {  
10       color: #074e68;  
11     }  
12   }  
13 }
```

CSS

```
1  section {  
2    height: 100px;  
3    width: 100px;  
4  
5  
6  section .class-one {  
7    height: 50px;  
8    width: 50px;  
9  
10  
11  section .class-one .button {  
12    color: #074e68;  
13  }
```

不支援 media query 怎辦？

- 用別人寫好的js來處理

```
<!--[if lt IE 9]>
```

```
<script src="http://css3-mediaqueries-js.googlecode.com/svn/trunk/css3-mediaqueries.js"></script> <![endif]-->
```

或其他

- <http://css-tricks.com/resolution-specific-stylesheets/>
- <http://www.themaninblue.com/experiment/ResolutionLayout/>
- <http://plugins.jquery.com/project/MediaQueries>



Responsive Image

max-width

- 盡量使用max-width取代width: 螢幕小於max-width就用螢幕width
- 使用max-width:100% 和 height:auto，圖片能自適應父元素的寬度。

```
img {  
    max-width: 100%;  
    height: auto;  
}
```

- 遇到IE8，可使用width:auto。

```
@media screen {  
    img {  
        width: auto; /* for ie 8 */  
    }  
}
```

min-width

- 透過min-width設定最小的寬度，使得元素寬度小於min-width時，強迫用min-width

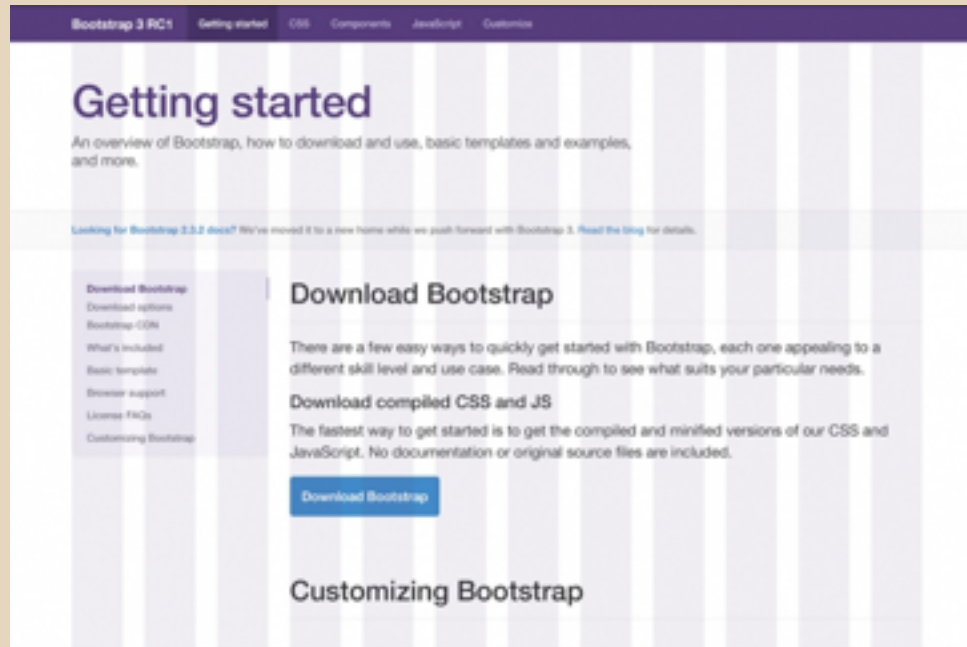




Useful Tools for Designers

Bootstrap PSD

- <http://goo.gl/NLpr6N>

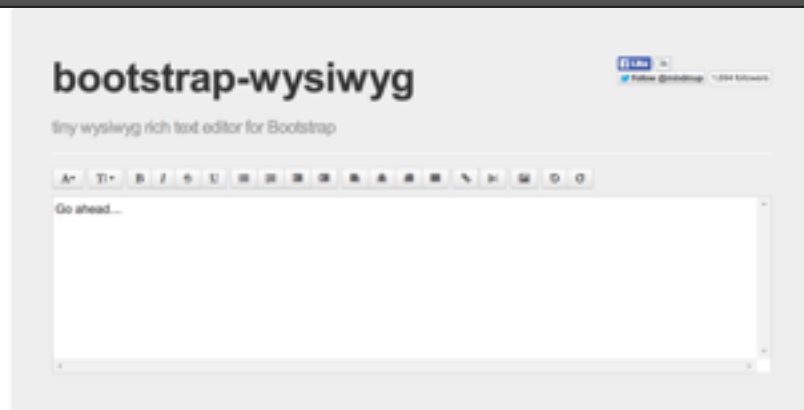


Bootstrap 3 Illustrator Template

<http://goo.gl/yVg8ZY>



bootstrap-wysiwyg



About

This [tiny \(2KB, < 200 lines\) jQuery Bootstrap plugin](#) turns any DIV into a WYSIWYG rich-text editor, inspired by [CKEditor](#) and [bootstrap-wysihtml5](#). Here are the key features:

- Automatically binds standard hotkeys for common operations on Mac and Windows
- Drag and drop files to insert images, support image upload (also taking photos on mobile devices)
- Voice dictation input (only in Chrome)
- Allows a custom built toolbar, no magic markup generators, enabling the web site to use all the goodness of Bootstrap, Font Awesome and so on...
- Does not force any styling - it's all up to you
- Uses standard browser features, no magic non-standard code, toolbar and keyboard configurables to include any supported browser command
- Does not create a separate frame, backup text areas etc - instead keeps it simple and runs everything inline in a DIV
- (Optionally) cleans up trailing whitespace and empty divs and spans
- Requires a modern browser (tested in Chrome 26, Firefox 19, Safari 6, reported by users to work in IE10)
- Supports mobile devices (tested on iOS 6 iPad/iPhone and Android 4.1.1 Chrome)

Why?



Usage

```
<div id="editor"> </div> </div>
```

Don't forget to style your editor div:

```
<div id="editor" style="border: 1px solid #ccc; min-height: 300px;">
```

If you want to use this for a mobile web site, make sure to read about [how to style it](#) to optimize mobile screen-usage and experience (please note that this demo page isn't optimized for mobile screens).

Optionally, also create a toolbar (see the source of this page for an example):

```
<div class="tiny-toolbar" data-role="editor-toolbar">
  <div target="#editor">
    ...
  </div>
</div>
```

In the toolbar, execute simple commands by adding a data-will attribute to a link.

```
<a href="#editor" data-will="bold">...</a>
```

execute more complex commands by adding an argument after a blank or providing an input with a data-will command (the input value is used as an argument). In case of the inputs, the file contents are read in using the FileReader API and used as the command value.

```
<a href="#editor" data-will="fontName" data-will-value="Arial">...</a>
<input type="text" data-will="fontSize" data-will-value="12px" value="12px">
```



Practice

現場實戰

