

# DrupalCamp Taipei Workshop: Mobile Website Development

John Albin Wilkins  
@JohnAlbin

# 5 ways to build mobile

- **Native apps** (code compiled to run natively on iPhones, Androids, etc.)
- **Web apps** (HTML5 and JavaScript-based apps running on mobile browsers)
- **Mobile/desktop domain switching** (parallel websites)
- **Responsive design**
- **RESS: Responsive Design + Server-side Components**

<http://palantir.net/blog/drupal-8-mobile-initiative>

# viewport size $\neq$ screen size

- Before iPhone, websites were built for desktop sizes of ~960 pixels
- iPhone was 320px wide
- iPhone reports a viewport width of 960px
- We need a meta tag to reset the viewport width to be the device width

[http://www.quirksmode.org/blog/archives/2010/04/a\\_pixel\\_is\\_not.html](http://www.quirksmode.org/blog/archives/2010/04/a_pixel_is_not.html)

# Mobile device zooming

## 手機上的內容縮放

- Meta tags can control the scaling/zooming of mobile devices  
使用Meta tags控制手機裝置的比例和縮放
- More info: 其他資訊  
<http://davidbcalhoun.com/2010/viewport-metatag>

```
<meta name="viewport"  
  content="width=device-width">  
<meta name="MobileOptimized" content="width">  
<meta name="HandheldFriendly" content="true">
```

# Responsive Design

## 彈性設計

1. Flexible Grids 彈性大小網格
2. Flexible Images 彈性大小圖片  
(let the browser scale the images)
3. CSS3 Media Queries  
CSS3的Media Queries

# How do we target different screen sizes? 如何指定不同的視窗大小？

- With CSS2 media types we could target different classes of devices:  
使用CSS2的media types來指定不同裝置的class:  
print, screen, handheld, all
- With CSS3 media queries we target device capabilities.  
使用CSS3的media queries指定裝置性能
  - [type] and ([query])
  - all and (max-width: 768px)

# CSS3 的 media queries

```
/* Some devices don't understand media
queries. Provide a "mobile first"
design by default. */
```

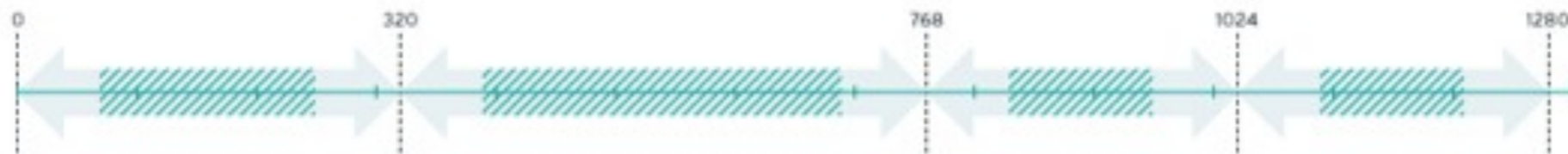
```
.selector { property: value; }
```

```
@media all and (width: 444px) {
  .selector { property: value; }
}
```

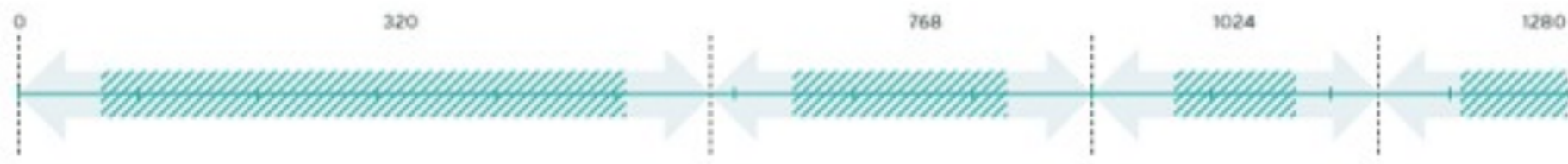
```
@media all and (max-width: 888px) {
  .selector { property: value; }
}
```

# Breakpoints

## Traditional Approach



## Proposed Approach





# Flexible images

## 彈性大小圖片

- Use `max-width` to constrain images within the flexible grid.  
使用 `max-width` 屬性來限制彈性網格裡的圖片大小
- ```
img {  
  max-width: 100%;  
  height: auto;  
}
```

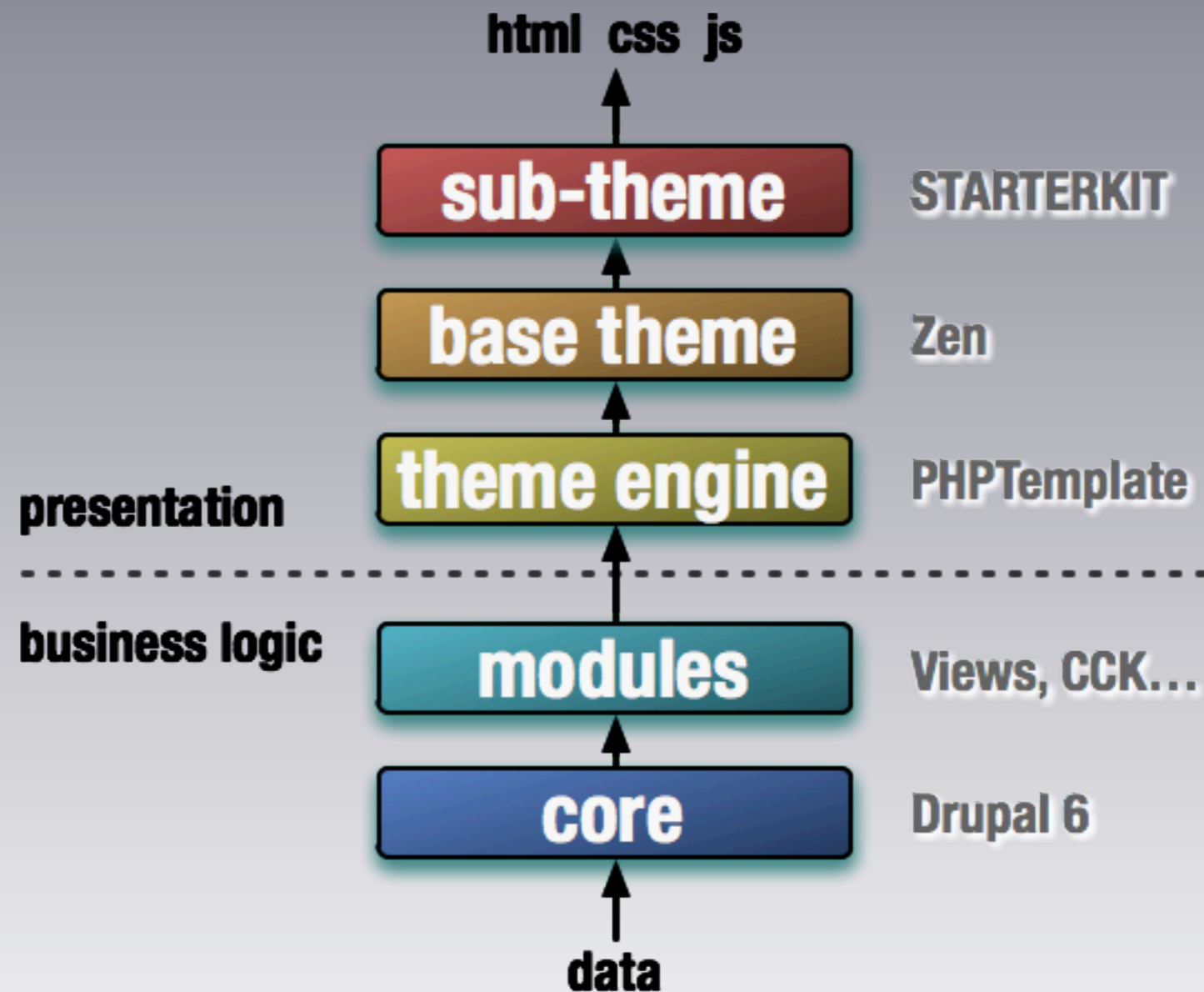
# Drush + Zen =

- > drush dl zen (*download Zen*)
- > drush cc all (*clear the cache*)
- > drush zen "DrupalCamp Taipei" taipei  
--description="Our example theme."  
--without-rtl

Starter kit for "DrupalCamp Taipei" created  
in: /sites/all/themes/taipei

# Drupal's Theme Layer

(the big picture)



# Traditional way to support mobile 手機版傳統開發方式

Build a separate  
mobile website

製作手機版網頁



This is really hard! 這真的很難！

# Responsive Design

## 彈性網頁設計

- Single source of HTML  
單一原始HTML
- Responds to different viewport sizes  
根據瀏覽器大小調整內容



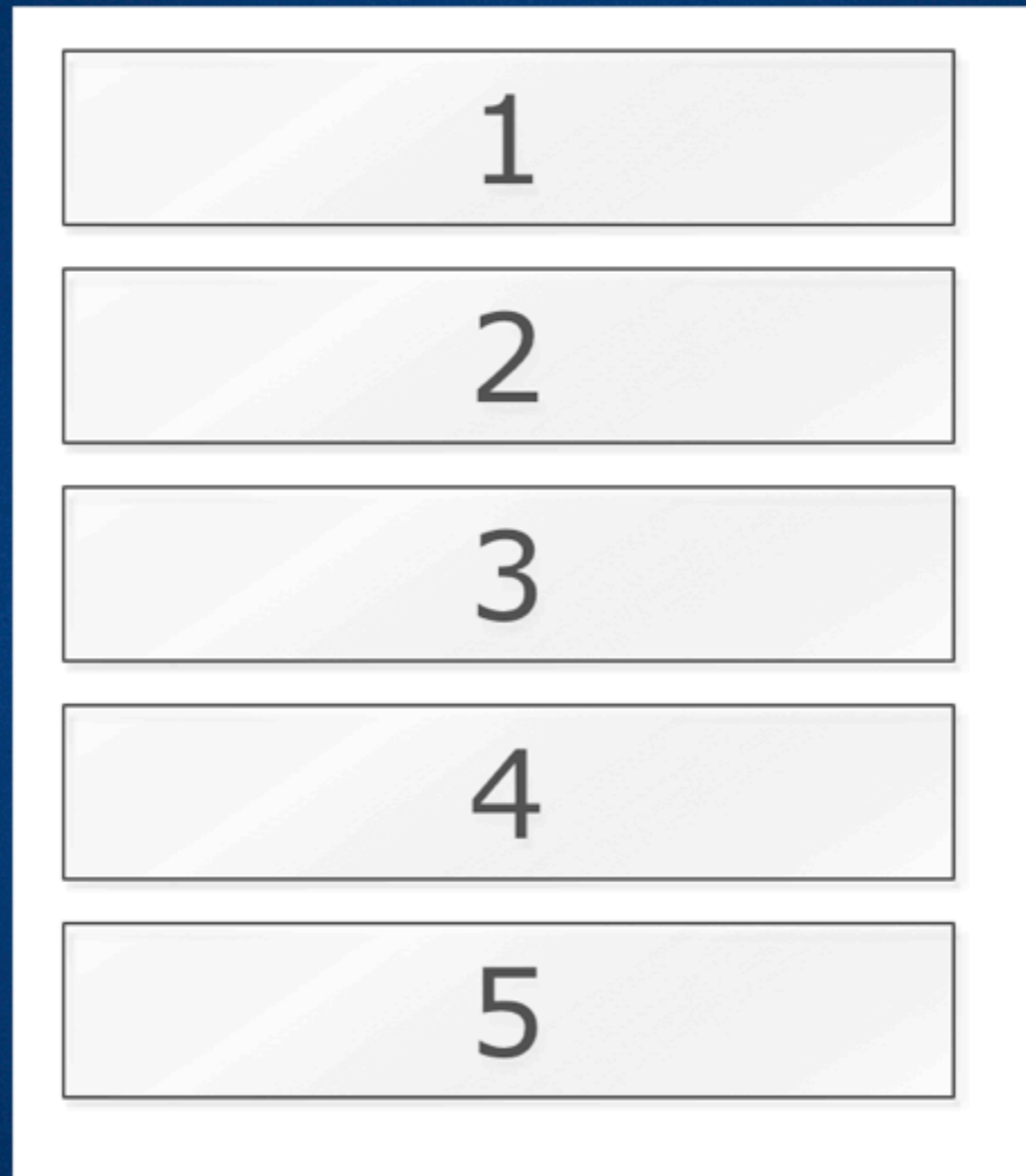
# Creating a layout is like... a game of Tetris



...but upside-down.



Content first



Use your content's natural semantic ordering.

(All other rules depend on it.)





# The Adjacent Sibling Rule



Rows are composed of elements that are next to each other in the HTML source.

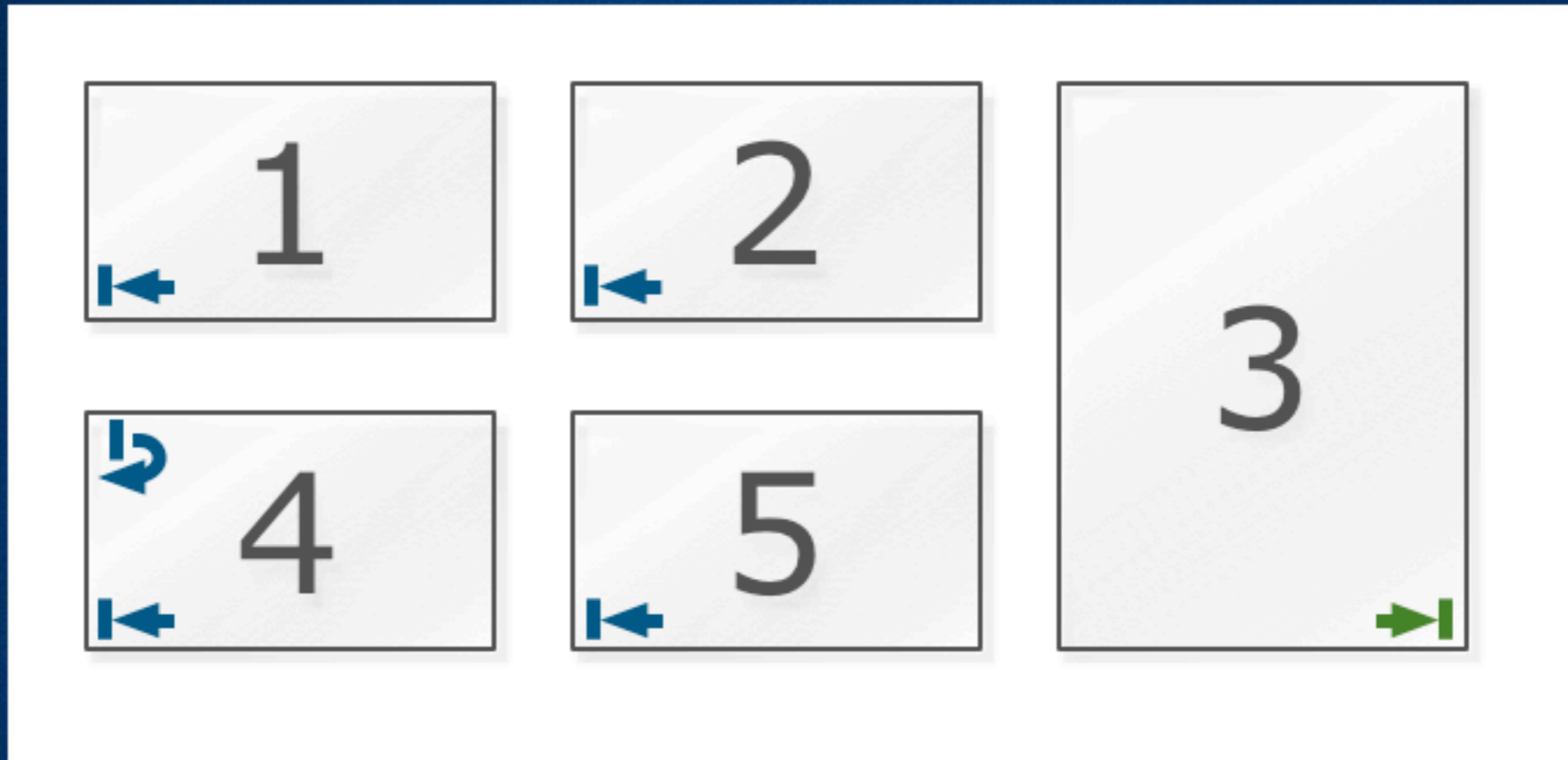
(But they can be re-arranged within the row.)

(To start a new row, add a clear to the first sibling.)

← float: left;    ↶ clear: left;



# The Opposing Float



An element can span multiple rows if it floats in the opposite direction of its siblings.

↩ float: left;   ↶ clear: left;   ➡ float: right;



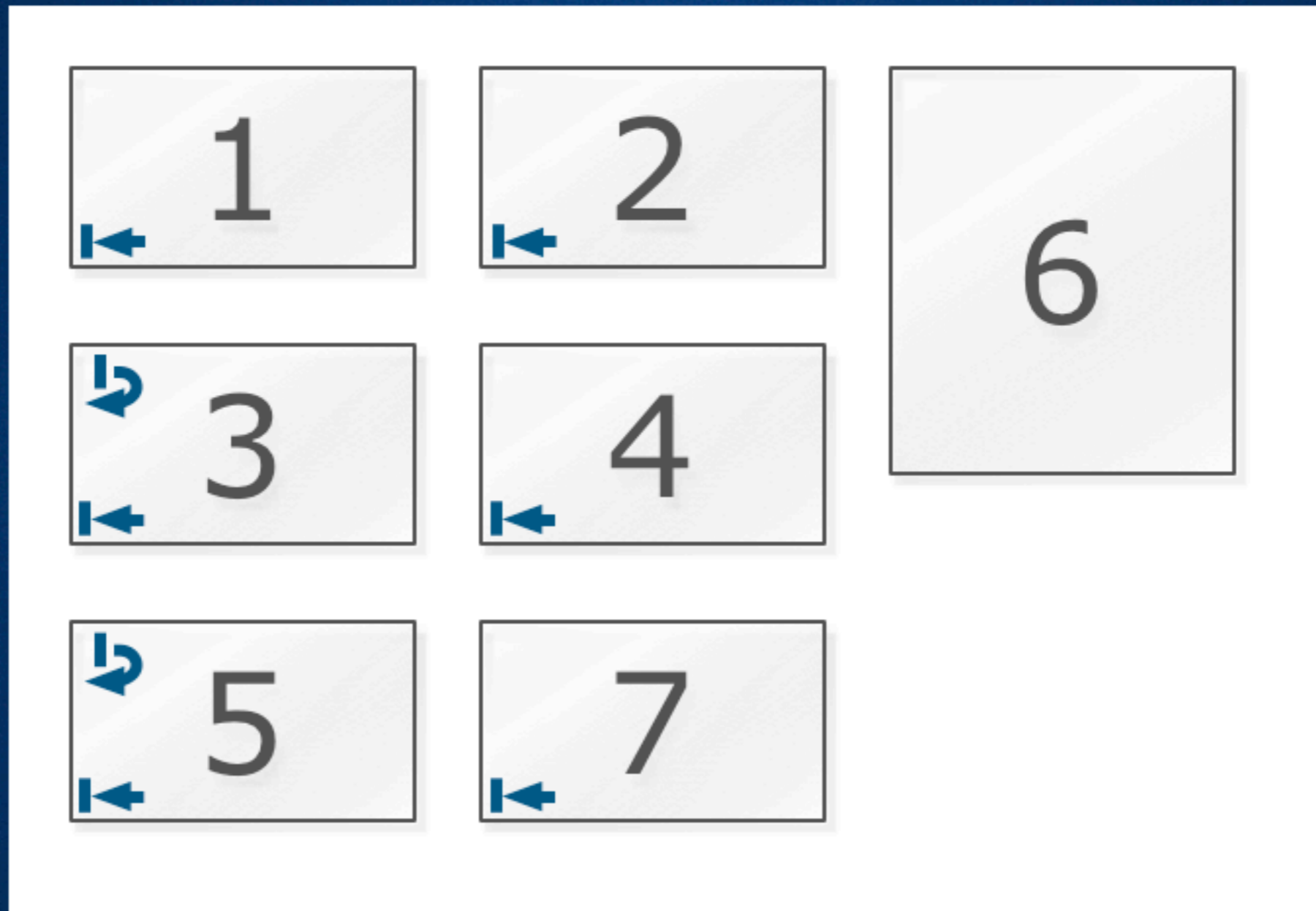
More complicated layouts are possible

← float: left;    ↵ clear: left;    → float: right;

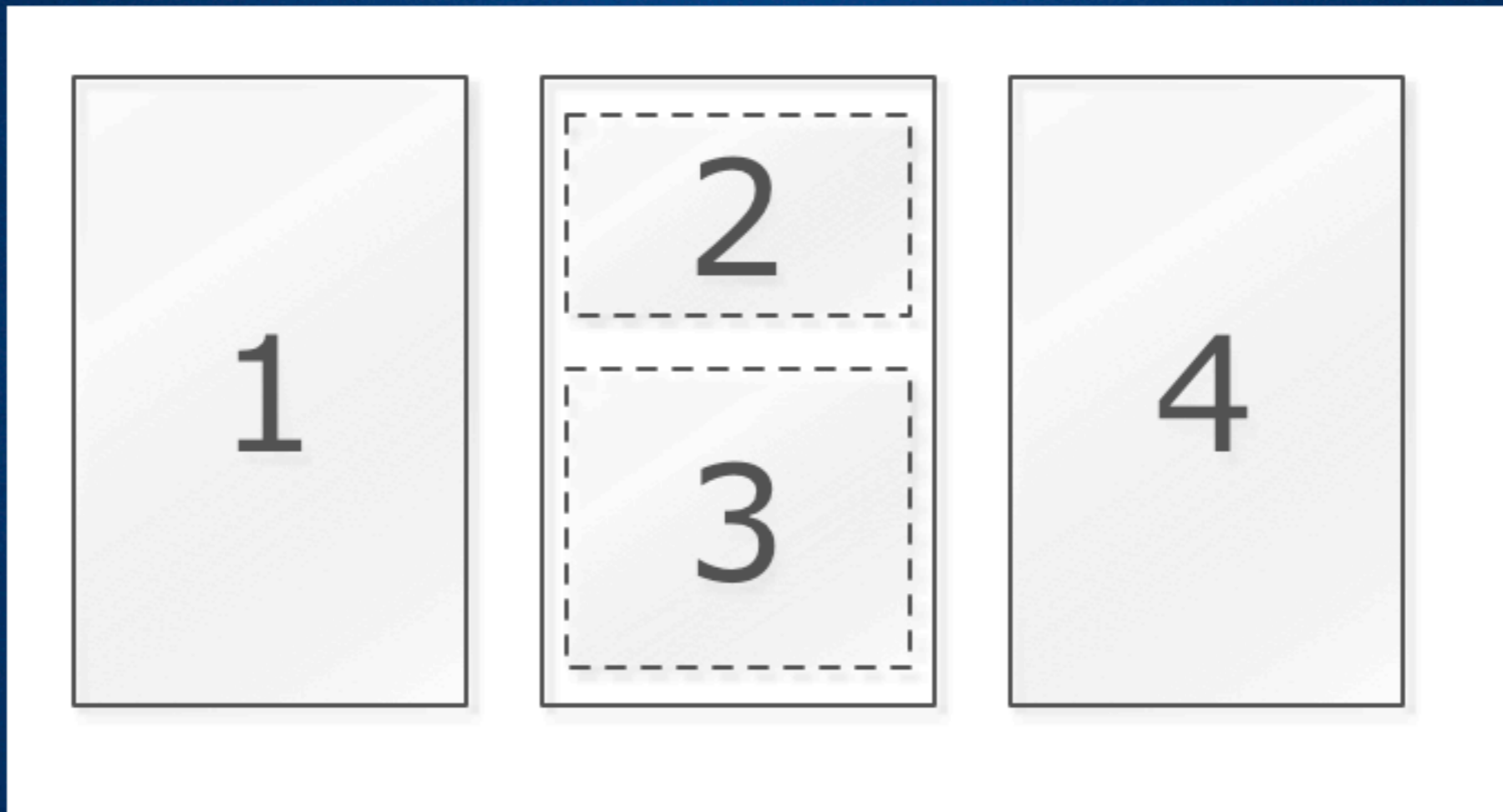


# The Absolute Exception

( to the Adjacent Sibling rule )

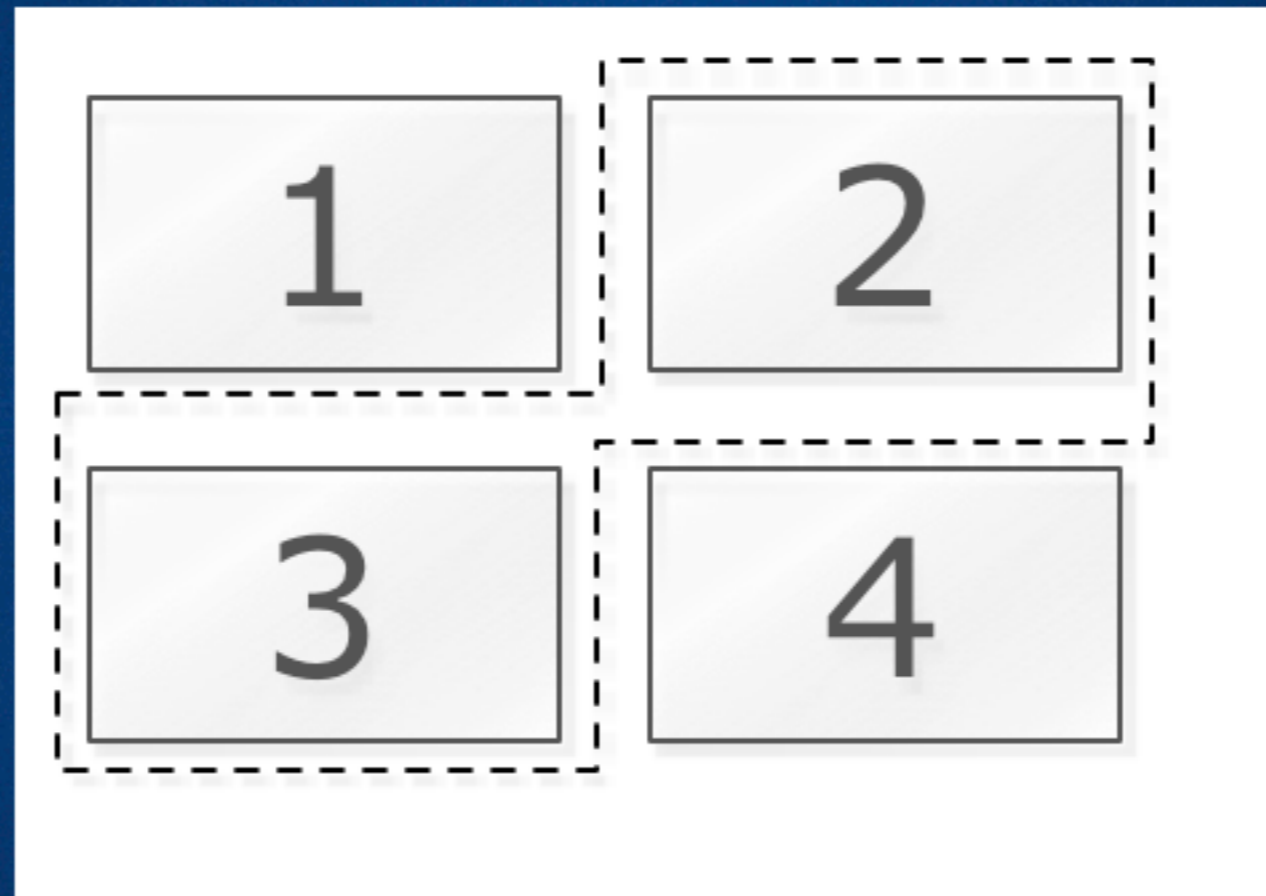


If you leave space below and can be sure no other content will overlap, you can apply `position: absolute;`



Sometimes the “lasso” is positioned...

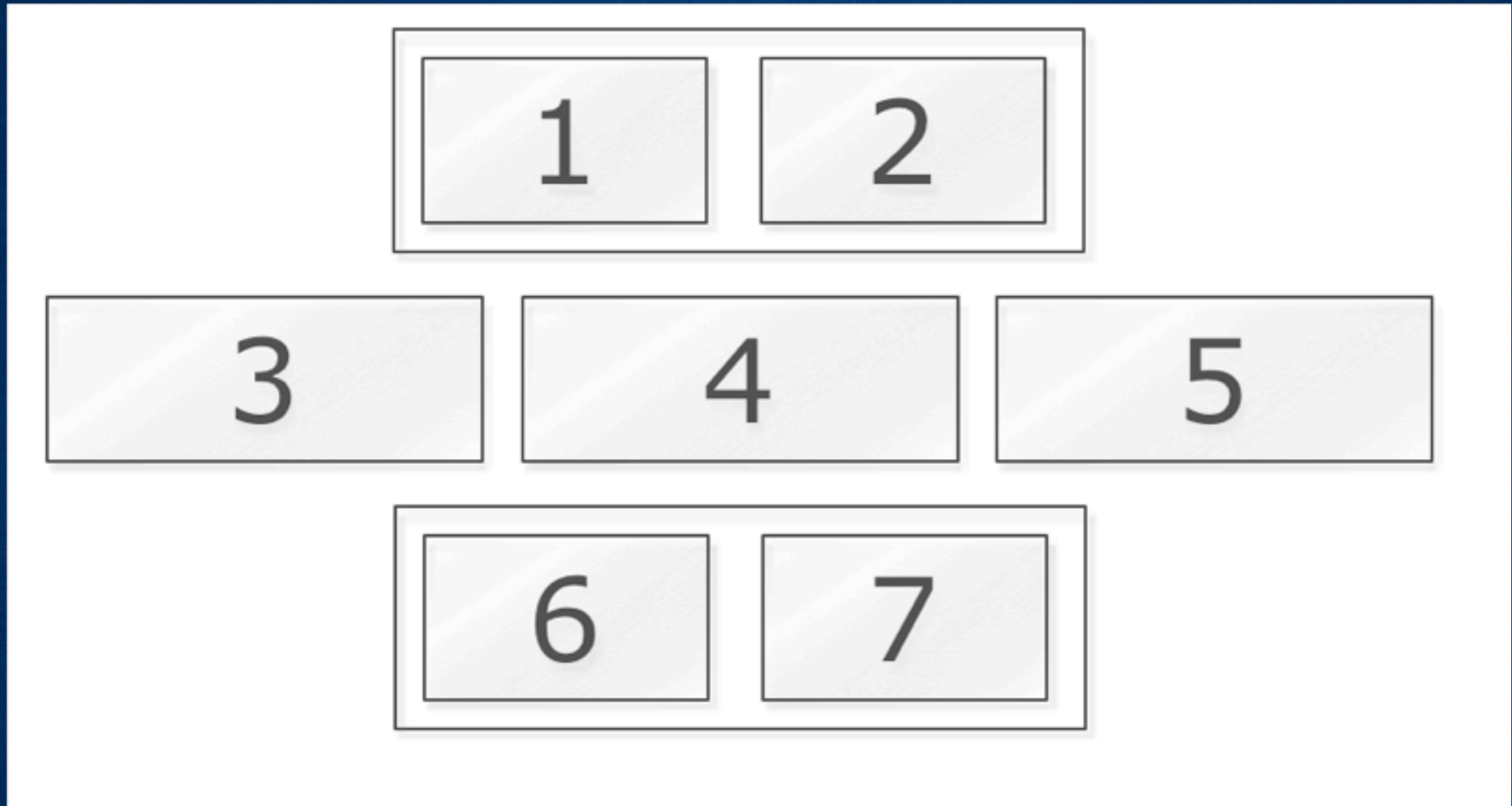




Sometimes the “lasso” is positioned...  
...and sometimes not.



# The Corset Variant

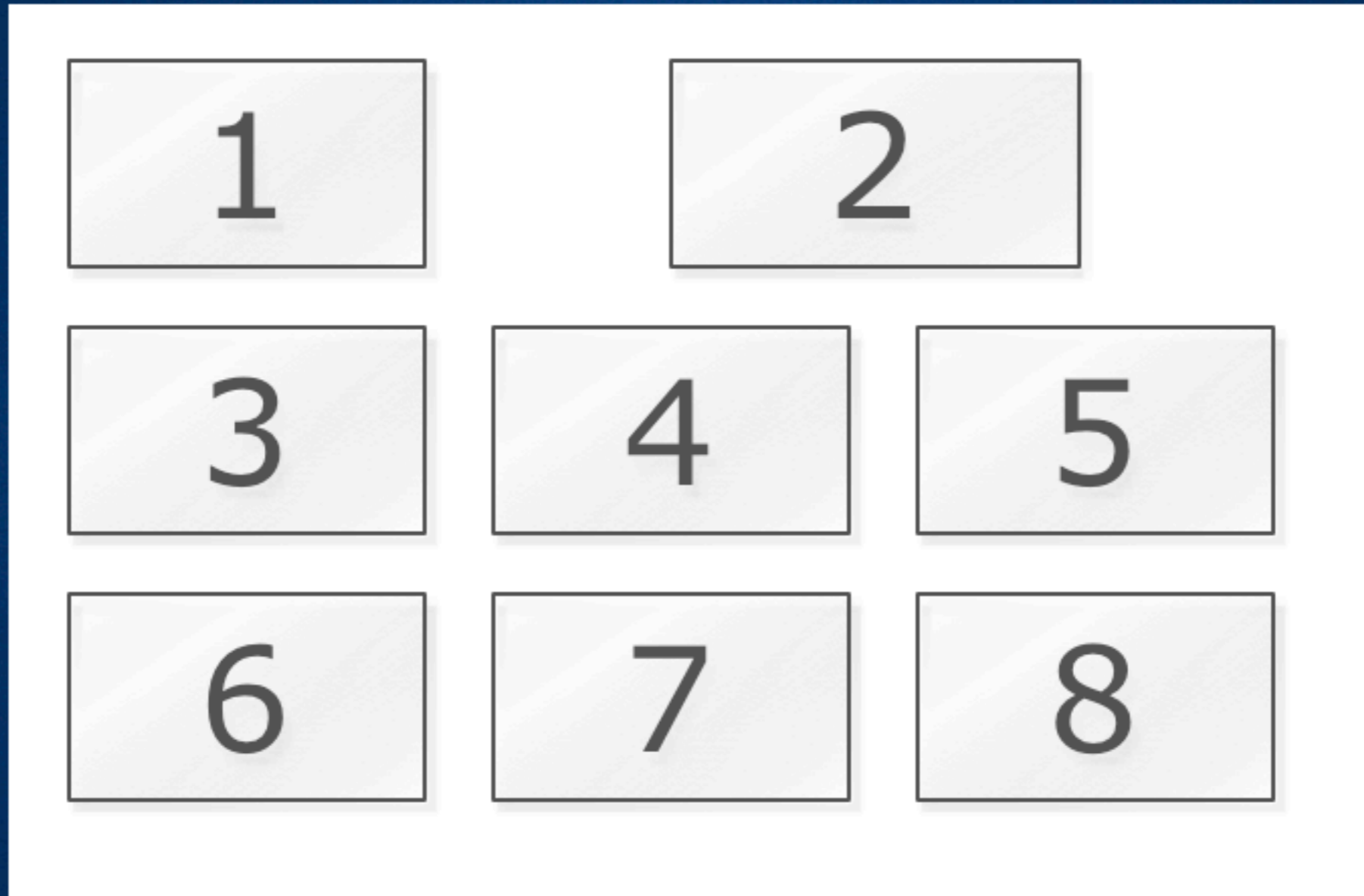


Two “lassos” make up the corset.

One more thing...



# The Violator



Violate the grid judiciously.

# More Tools: Sass to the rescue!



<http://zengrids.com/>

**Thank You!**