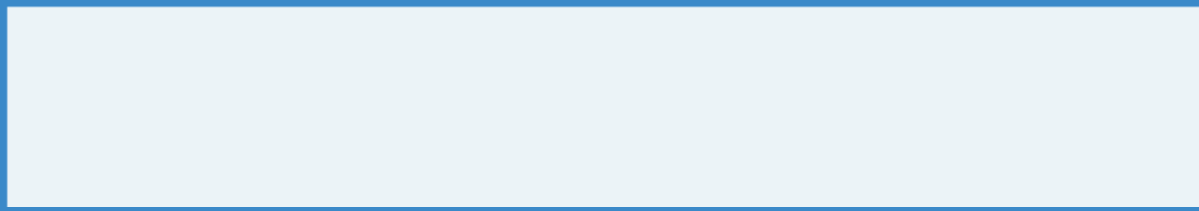
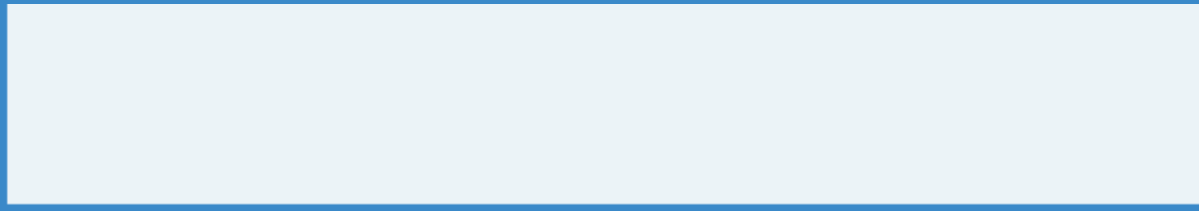
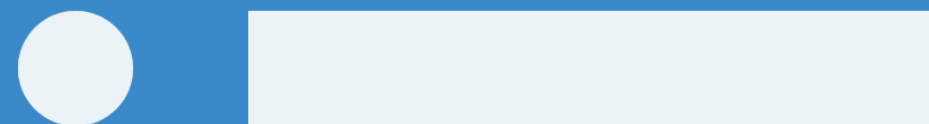
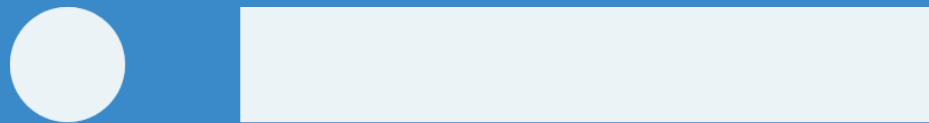


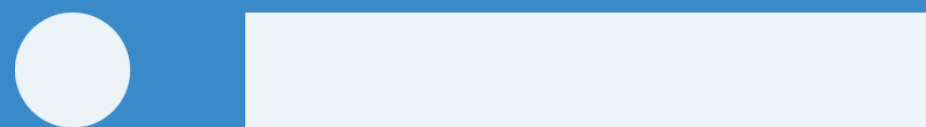
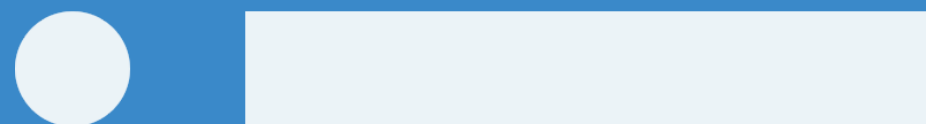
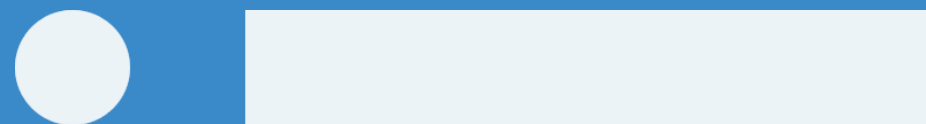
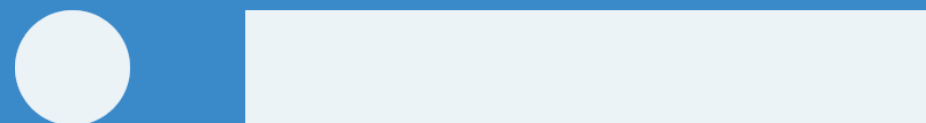
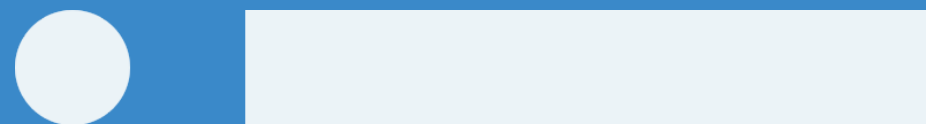
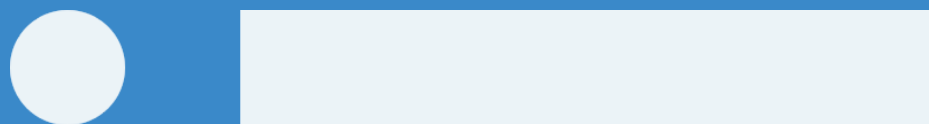
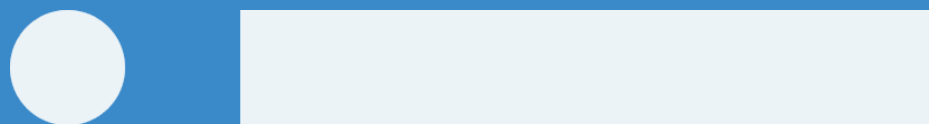
Stop Hitting
Yourself: CSS
Design Patterns
that Scale

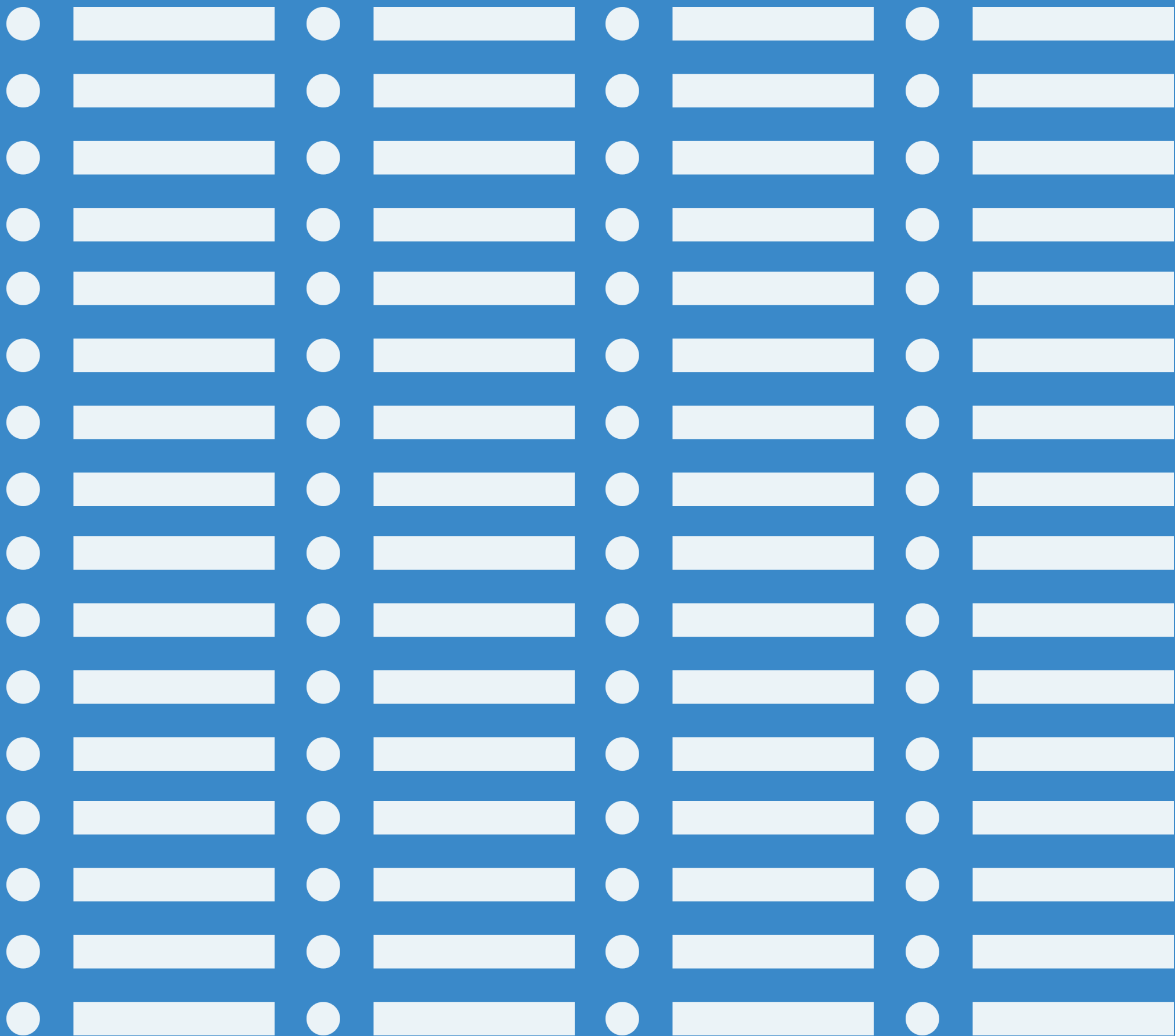


@klamping

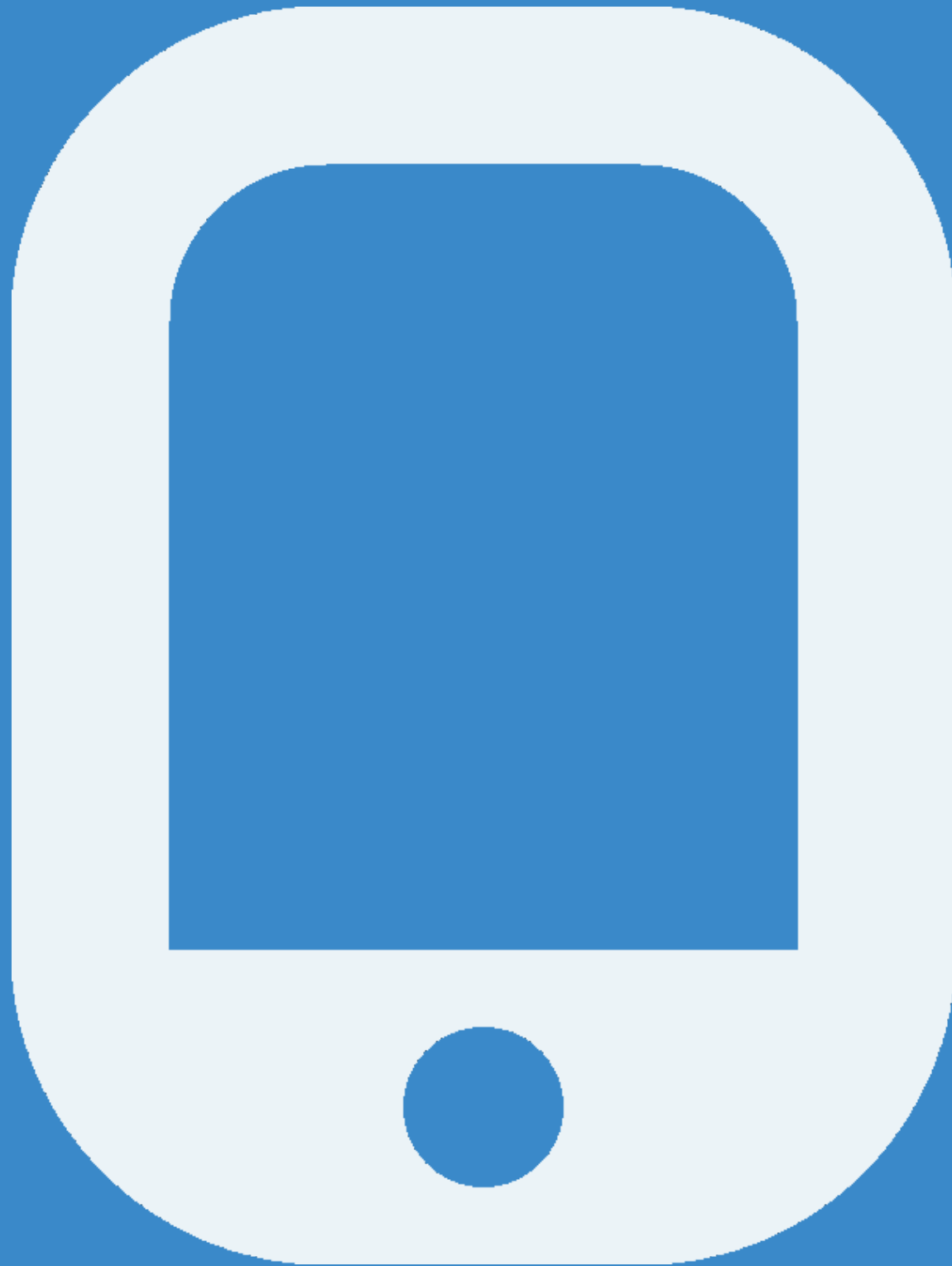




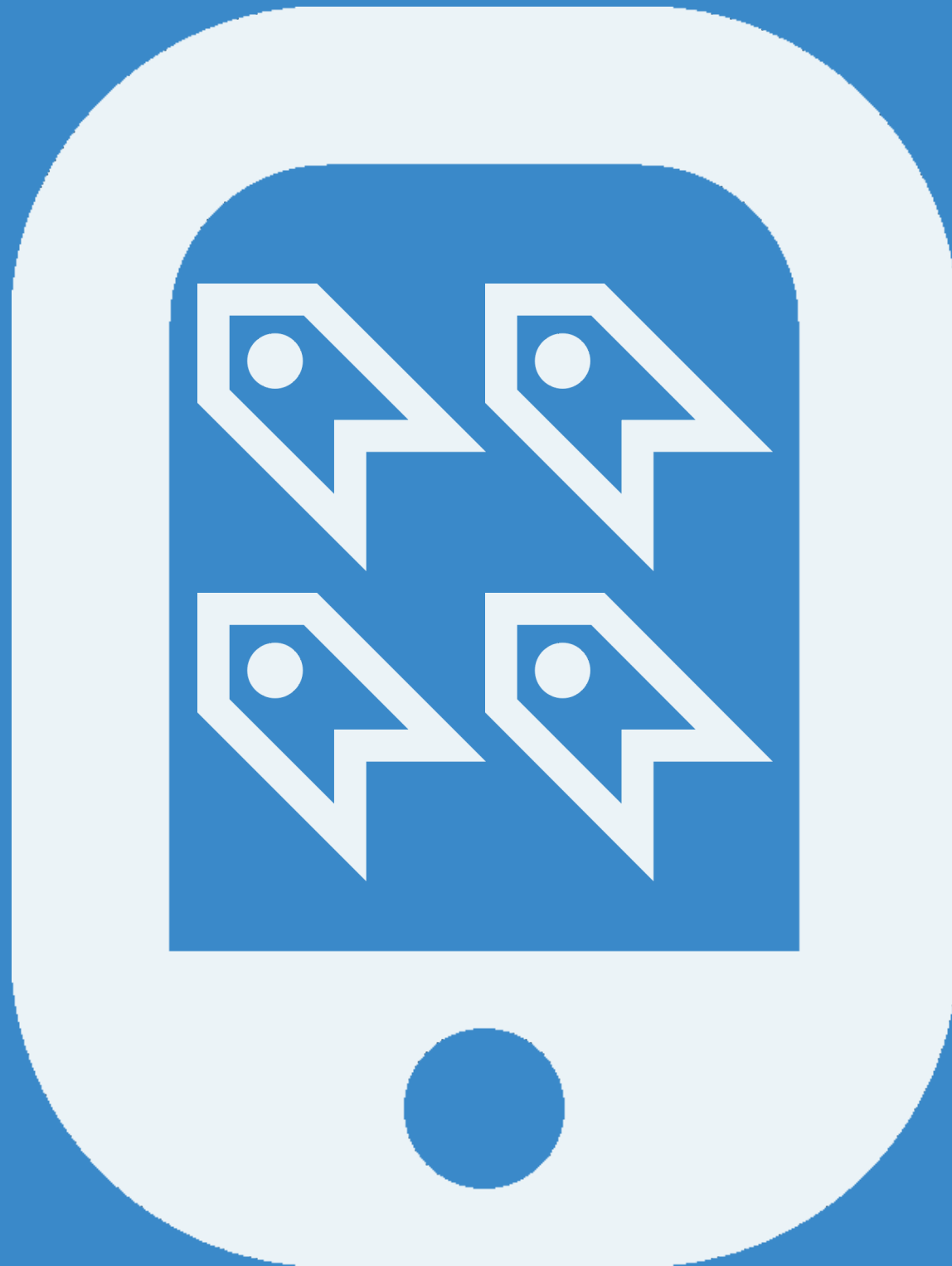


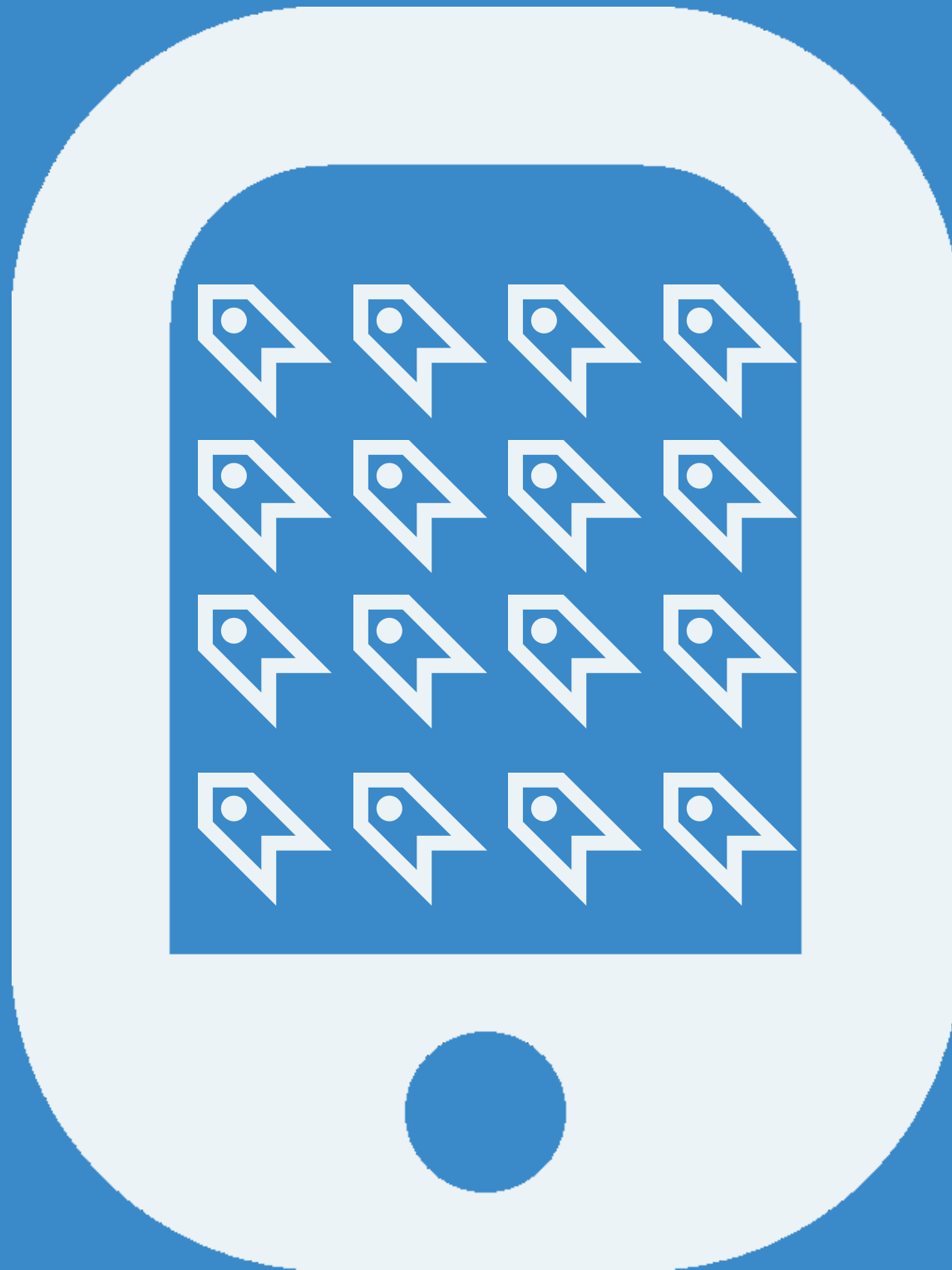


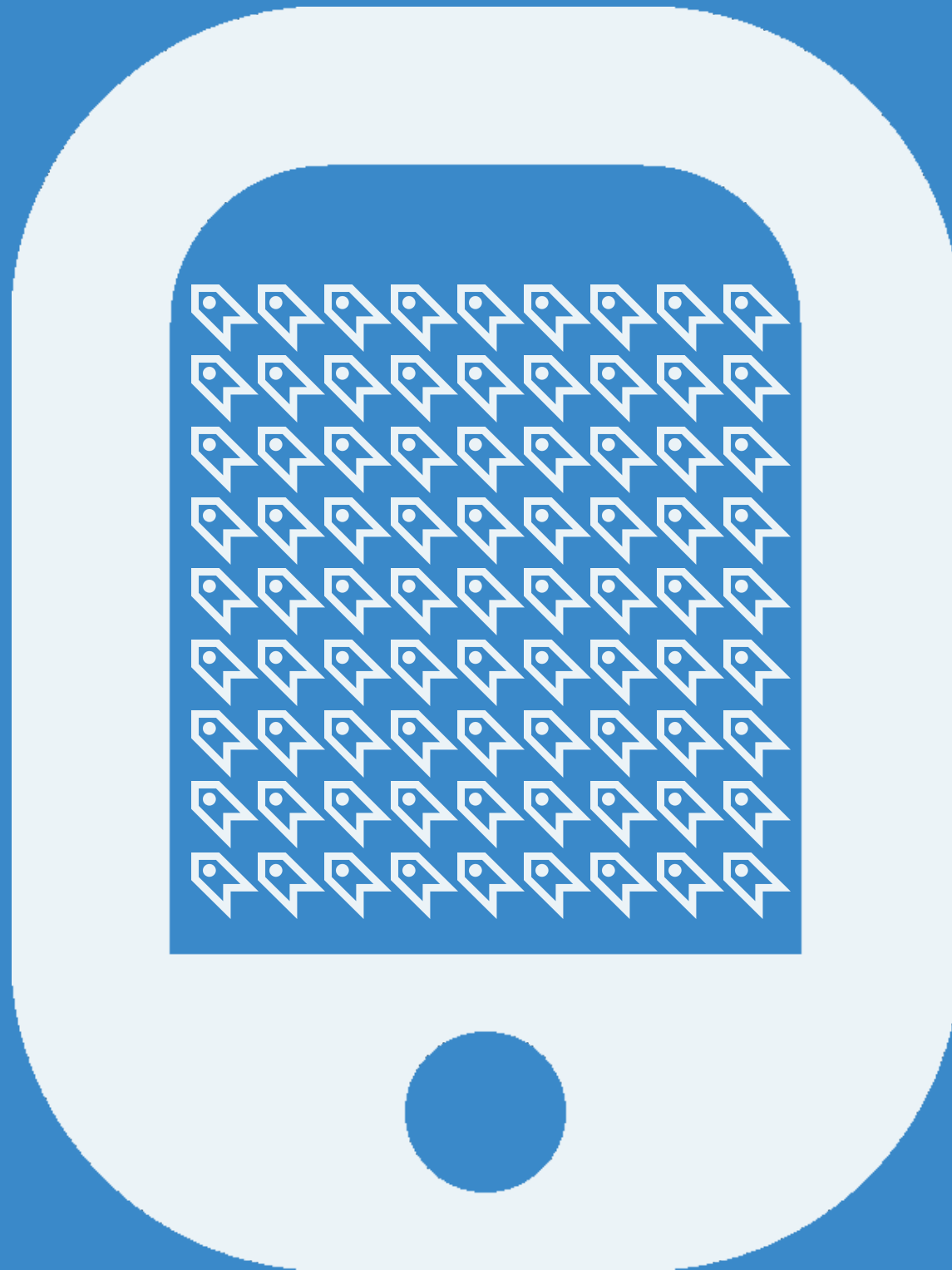


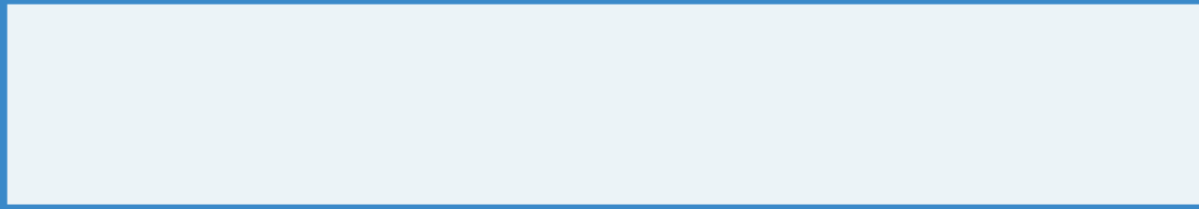
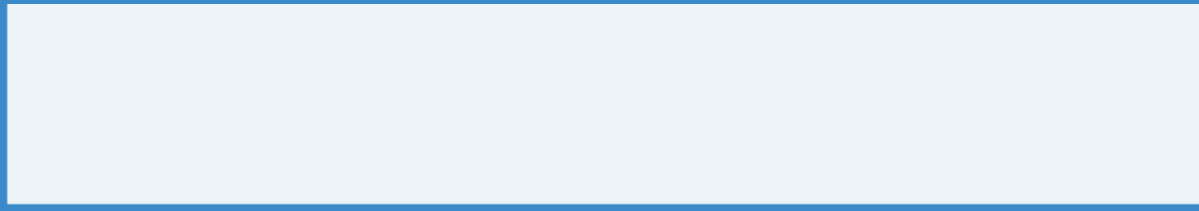












What works for
small

Doesn't work for
BIG

Because we build for small

We struggle
with **BIG**

Because we build for small

Scale The ability to

Scale The ability to

reuse

CODE

Scale The ability to

reuse add

CODE

Scale The ability to

reuse add fix

CODE

Reuse Code

Specific code
is too specific

Error

Name Required

#E9F2F9

Error

Name Required

#E9F2F9

Error

Name Required

#9CC4E4

```
<div class="error">  
  <h1>Error:</h1>  
  <p>Name Required</p>  
</div>
```

```
.error h1 {  
  color: #E9F2F9;  
}
```

```
.error p {  
  color: #9CC4E4;  
}
```

Simple and Semantic

But is it reusable?

Think





Think

My Life

Things I hate:

- Lists
- Irony
- Clichés

My Life

Things I hate:

- Lists
- Irony
- Clichés

#E9F2F9



#9CC4E4




```
<div id="content">
  <h1>My Life</h1>
  <aside>
    <h2>Things I Hate</h2>
    <ul>
      <li>Lists</li>
      <li>Irony</li>
      <li>Clichés</li>
    </ul>
  </aside>
</div>
```

```
.error h1,  
#content h2 {...}  
.error p,  
#content li {...}
```

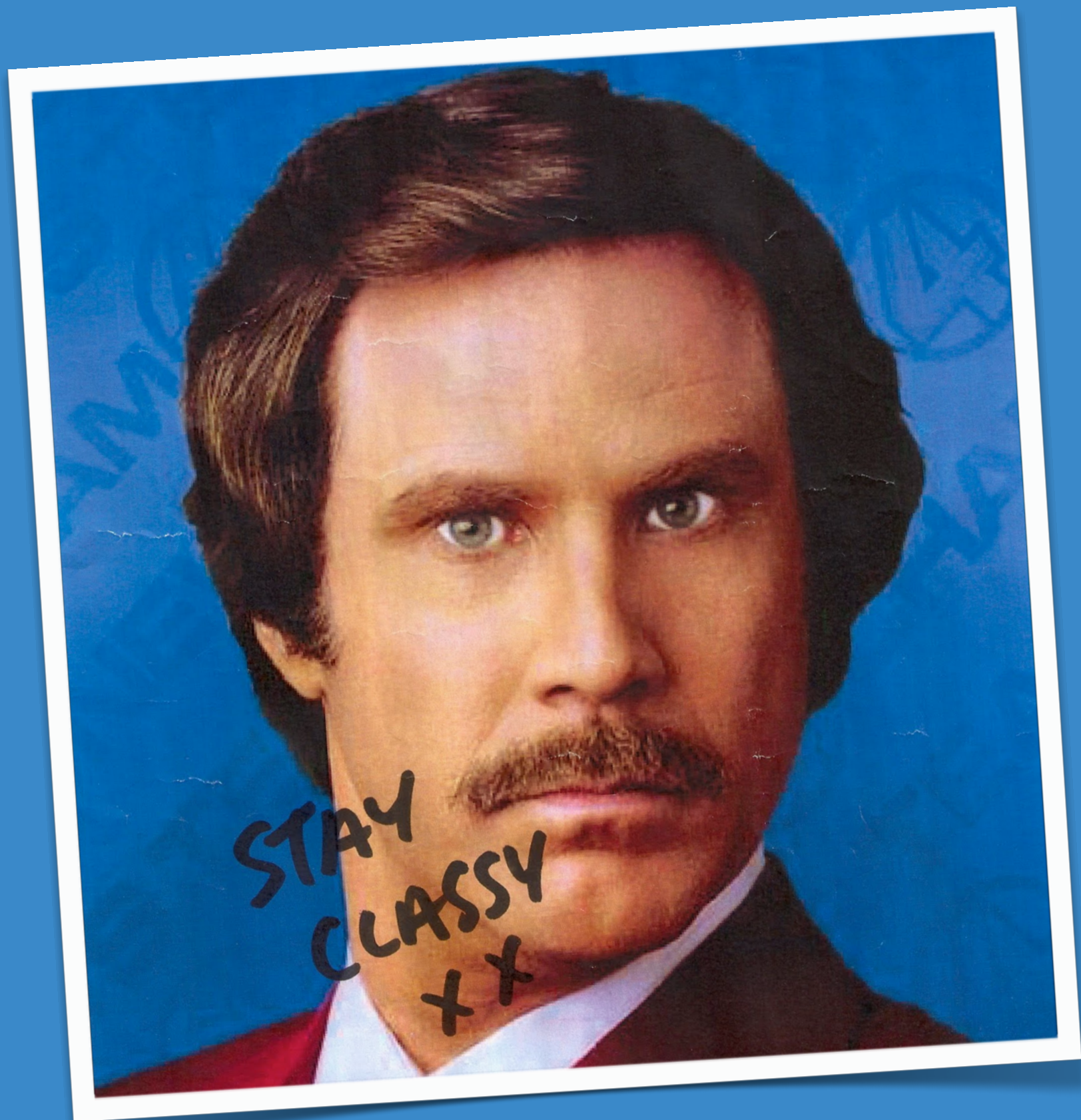
```
<div class="error box">  
  <h1 class="box-title">...</h1>  
  <p class="box-content">...</p>  
</div>
```

```
.box .box-title {...}
```

```
.box .box-content {...}
```

```
<div id="content">  
  <h1>My Life</h1>  
  <aside class="box">  
    <h2 class="box-title">...</h2>  
    <ul class="box-content">  
      <li>Lists</li>  
      <li>Irony</li>  
      <li>Clichés</li>  
    </ul>  
  </aside>  
</div>
```

```
<div id="content">
  <h1>My Life</h1>
  <aside class="box">
    <h2 class="box-title">...</h2>
    <ul class="box-content">
      <li>Lists</li>
      <li>Irony</li>
      <li>Clichés</li>
    </ul>
  </aside>
</div>
```



STAY
CLASSY
xx

Design
outside
your needs

Abstract your CSS,
just like other code

Design
outside
your needs

Build components,
not implementations

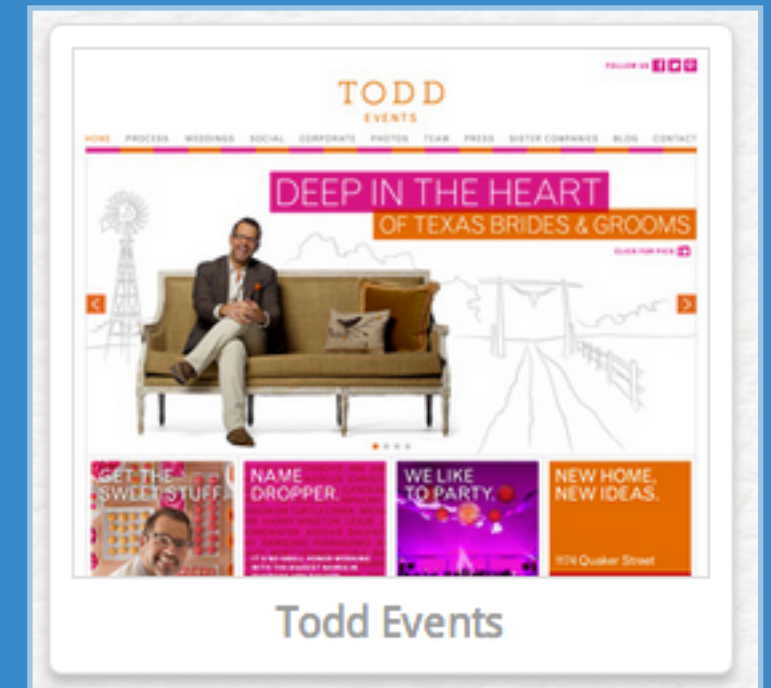
Design
outside
your needs

Literally,
think outside the box

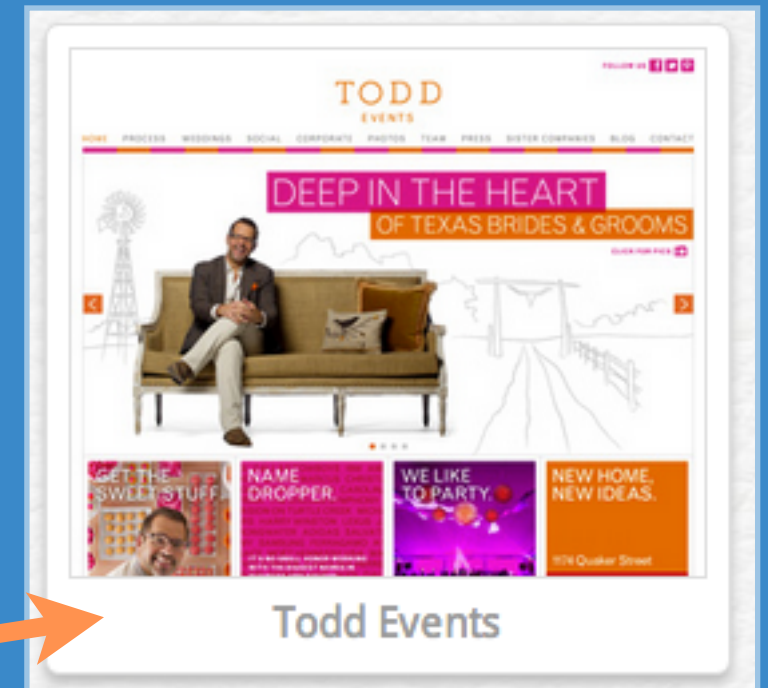
Design
outside
your needs

No “Magic Numbers”

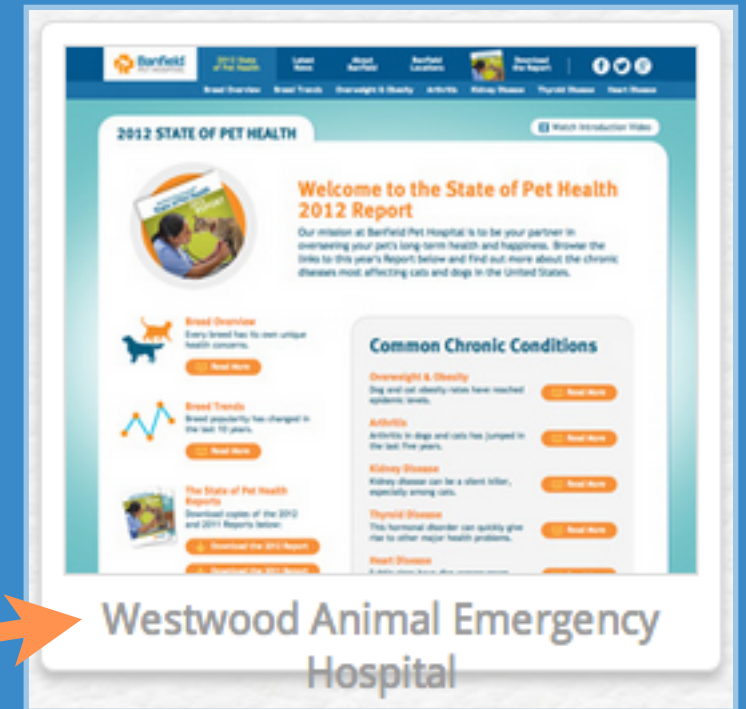
Design
outside
your needs



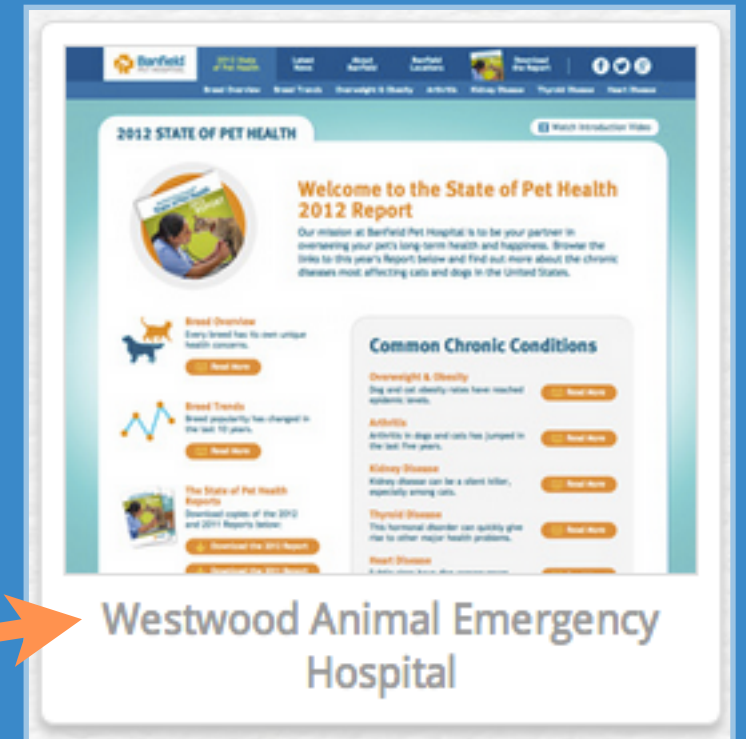
```
.site-title {  
  height: 30px;  
}
```



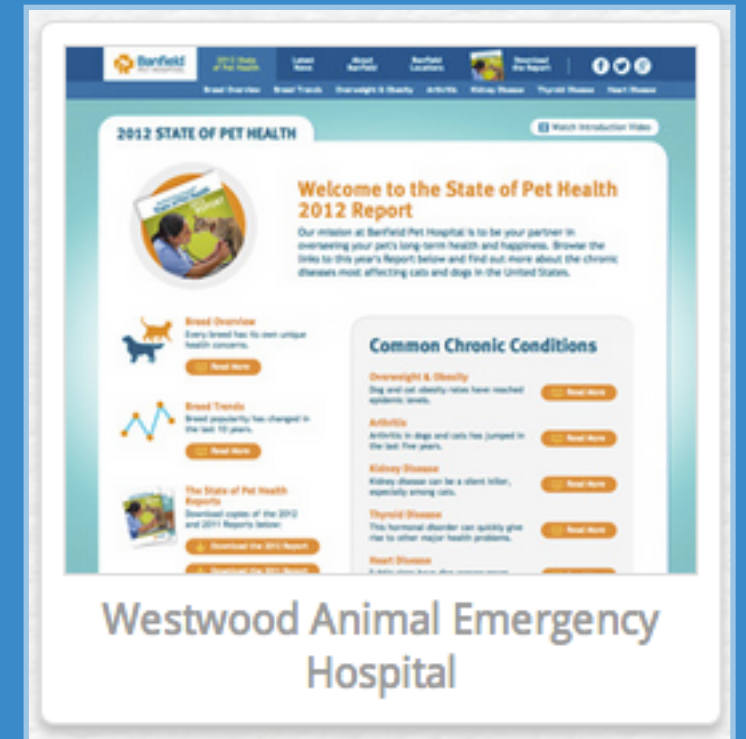
```
.site-title {  
  height: 30px;  
}
```



```
.site-title {  
  min-height: 30px;  
}
```

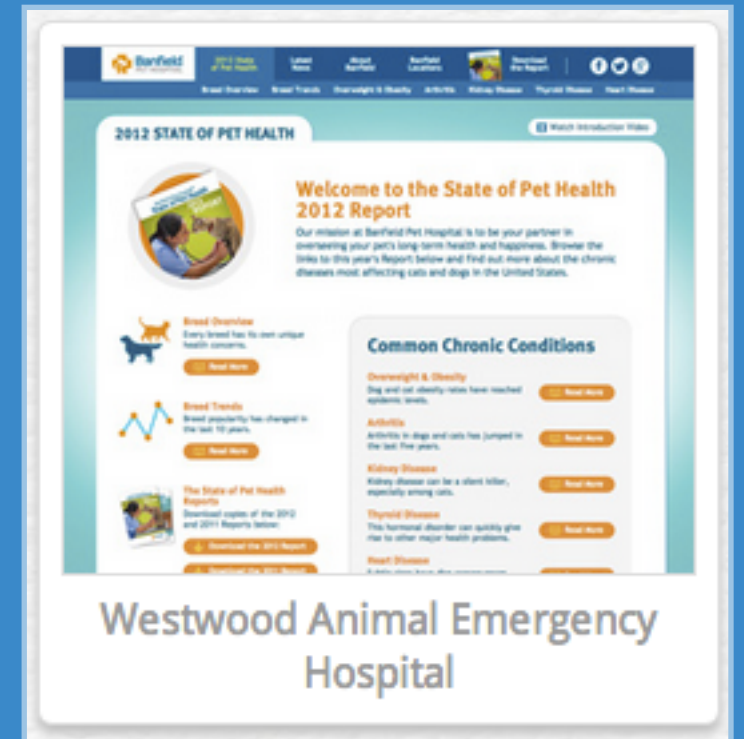



```
.site-title {  
  min-height: 30px;  
}
```



```
.site-title {  
  min-height: 30px;  
}
```

css-tricks.com/magic-numbers-in-css/



Component Dashboards

Documentation,
Examples, Styleguide

0. Overview

1. Buttons

2. Forms

3. Source Code

4. Text Styling

5. Listings

6. Boxed Groups

7. Icons

8. Navigation

9. Behavior

10. Discussion

11. Colors

12. Animations

13. Select Menu

2.1

forms.scss

Typical form groups — `<dl.form>` with a `<dt>` containing the label and `<dd>` containing the form elements.

- **.required** - This group is required to submit the form.

Example Text

Example Label

Example Text *

Example Label *

.required

```
<dl class="form">
  <dt><label>Example Text</label></dt>
  <dd><input type="text" class="textfield" value="Example Value"></dd>
</dl>
```

Component List

Example Text

Example Label

Example Text *

Example Label *

.required

```
<dl class="form">
  <dt><label>Example Text</label></dt>
  <dd><input type="text" class="textfield" value="Example Value"></dd>
</dl>
```

0. Overview

1. Buttons

2. Forms

3. Source Code

4. Text Styling

5. Listings

6. Boxed Groups

7. Icons

8. Navigation

9. Behavior

10. Discussion

11. Colors

12. Animations

13. Select Menu

2.1

forms.scss

Typical form groups — `<dl.form>` with a `<dt>` containing the label and `<dd>` containing the form elements.

- **.required** - This group is required to submit the form.

Example Text

Ex

Exa

Cl

Example Text *

Example Value

Example Label *

Choose an option

```
<dl class="form">
  <dt><label>Example Text</label></dt>
  <dd><input type="text" class="textfield" value="Example Value"></dd>
</dl>
```

Documentation

0. Overview

1. Buttons

2. Forms

3. Source Code

4. Text Styling

5. Listings

6. Boxed Groups

7. Icons

8. Navigation

9. Behavior

10. Discussion

11. Colors

12. Animations

13. Select Menu

2.1

forms.scss

Typical form groups — `<dl>`

containing the form elements

- **.required** - This group is

Examples

Example Text

Example Label

Example Text *

Example Label *

.required

```
<dl class="form">
  <dt><label>Example Text</label></dt>
  <dd><input type="text" class="textfield" value="Example Value"></dd>
</dl>
```

0. Overview

1. Buttons

2. Forms

3. Source Code

4. Text Styling

5. Listings

6. Boxed Groups

7. Icons

8. Navigation

9. Behavior

10. Discussion

11. Colors

12. Animations

13. Select Menu

2.1

forms.scss

Typical form groups — `<dl.form>` with a `<dt>` containing the label and `<dd>` containing the form elements.

- **.required** - This group is required to submit the form.

Example Text

Example Label

.required

Example Text *

Example Label *

Code Sample

```
<dl class="form">
  <dt><label>Example Text</label></dt>
  <dd><input type="text" class="textfield" value="Example Value"></dd>
</dl>
```


December 5, 2011

Knyle Style Sheets

So I've been writing CSS for somewhere around 13 years now. Some might think I've learned the right way to write CSS in that time — but if you ask me all I've learned is the most efficient way to drive someone insane.

CSS is complicated. It's not object oriented. It's not hierarchical. It's a specificity based cascade applied to a dynamic hierarchical data structure that few people truly comprehend. Trying to impart this knowledge on someone is a very difficult task with extremely minimal rewards. I used to think that imparting this knowledge was the path toward writing maintainable CSS within a team.

Maintainability comes from shared understanding

It's hard to define maintainability. In my eyes it has to do with creating an *shared understanding*. Anyone who has owned an aircooled Volkswagen knows how to adjust valves on any other aircooled Volkswagen. This is because of one of the best technical books ever written: [How to Keep Your Volkswagen Alive](#).

Everyone I know who owned a Bug, Bus, Ghia or Thing owns a copy of this book and ~~understands how to work on their car~~. This book is in large part responsible for that shared

```
// A button suitable for giving stars to someone.
//
// :hover          - Subtle hover highlight.
// .stars-given    - A highlight indicating you've already
//                  given a star.
// .stars-given:hover - Subtle hover highlight on top of stars-
//                  given styling.
// .disabled       - Dims the button to indicate it cannot
//                  be used.
//
//
// Styleguide 1.1.
a.button.star{
    ...
    &.star-given{
        ...
    }
    &.disabled{
        ...
    }
}
```

```
// A button suitable for giving stars to someone.
//
// :hover          - Subtle hover highlight.
// .stars-given    - A highlight indicating you've already
//                  given a star.
// .stars-given:hover - Subtle hover highlight on top of stars-
//                  given styling.
// .disabled       - Dims the button to indicate it cannot
//                  be used.
//
```

```
// Styleguide 1.1.
a.button.star{
```

```
  ...
  &.star-given{
    ...
  }
  &.disabled{
    ...
  }
}
```



Structure

```
// A button suitable for giving stars to someone.
//
// :hover          - Subtle hover highlight.
// .stars-given    - A highlight indicating you've already
//                  given a star.
// .stars-given:hover - Subtle hover highlight on top of stars-
//                  given styling.
// .disabled       - Dims the button to indicate it cannot
//                  be used.
//
//
// Styleguide 1.1.
a.button.star{
  ...
  &.star-given{
    ...
  }
  &.disabled{
    ...
  }
}
```



Description

```
// A button suitable for giving stars to someone.
//
// :hover          - Subtle hover highlight.
// .stars-given    - A highlight indicating you've already
//                  given a star.
// .stars-given:hover - Subtle hover highlight on top of stars-
//                  given styling.
// .disabled        - Dims the button to indicate it cannot
//                  be used.
```

```
// Styleguide 1.1.
a.button.star{
```

```
  ...
  &.star-given{
    ...
  }
  &.disabled{
    ...
  }
}
```



Variations

Your standard form button.

- **:hover** - Highlights when hovering.
- **:disabled** - Dims the button when disabled.
- **.primary** - Indicates button is the primary action.
- **.smaller** - A smaller button

Example Button

Example Button

:hover

Example Button

:disabled

Example Button

.primary

Example Button

.smaller

```
<button class="$modifier_class">Example Button</button>
```

Add Code

Specificity

One ID to rule them all


```
<div id="content">  
  <h1>One does not simply:</h1>  
  <ul>  
    <li>Walk into Mordor</li>  
    <li>Override an ID</li>  
  </ul>  
</div>
```

```
<div id="content">  
  <h1>One does not simply:</h1>  
  <ul>  
    <li>Walk into Mordor</li>  
    <li>Override an ID</li>  
  </ul>  
</div>
```

```
<div id="content">  
  <h1>One does not simply:</h1>  
  <ul>  
    <li>Walk into Mordor</li>  
    <li>Override an ID</li>  
  </ul>  
</div>
```

```
#content ul {  
    margin-left: 10px;  
}  
#content ul li {  
    list-style: square;  
}
```

```
<div id="content">
```

```
...
```

```
<aside id="related">
```

```
<h2>Other overused memes:</h2>
```

```
<ul>
```

```
<li>What if I told you...</li>
```

```
<li>I don't always...</li>
```

```
<li>Keep calm...</li>
```

```
</ul>
```

```
</aside>
```

```
...
```

```
#content ul {  
    margin-left: 10px;  
}  
#content ul li {  
    list-style: square;  
}  
#content #related ul {  
    margin-left: 0;  
}  
#content #related ul li {  
    list-style: none;  
    background: url(rage-face.png);  
}
```

```
#content ul {  
    margin-left: 10px;  
}  
#content ul li {  
    list-style: square;  
}  
#content #related ul {  
    margin-left: 0;  
}  
#content #related ul li {  
    list-style: none;  
    background: url(rage-face.png);  
}
```

Override

A dark blue rectangular box with a white border and the word "Override" in white text. Two orange arrows originate from the bottom-left corner of the box. One arrow points to the "margin-left: 0;" line in the "#content #related ul {" rule. The other arrow points to the "list-style: none;" line in the "#content #related ul li {" rule.

```
#content ul {  
    margin-left: 10px;  
}  
#content ul li {  
    list-style: square;  
}  
#content #related ul {  
    margin-left: 0;  
}  
#content #related ul li {  
    list-style: none;  
    background: url(rage-face.png);  
}
```

IDs



The diagram consists of a dark blue rectangular box with a white border containing the text 'IDs'. Two orange arrows originate from the bottom-left corner of this box. One arrow points diagonally down and to the left, ending at the '#related' part of the selector '#content #related ul' in the CSS code. The second arrow points diagonally down and to the left, ending at the '#related' part of the selector '#content #related ul li' in the CSS code.


```
<div id="content">
```

```
...
```

```
<aside id="related">
```

```
<div class="warning">
```

```
<h2>This post contains:</h2>
```

```
<ul>
```

```
<li>Vulgarity</li>
```

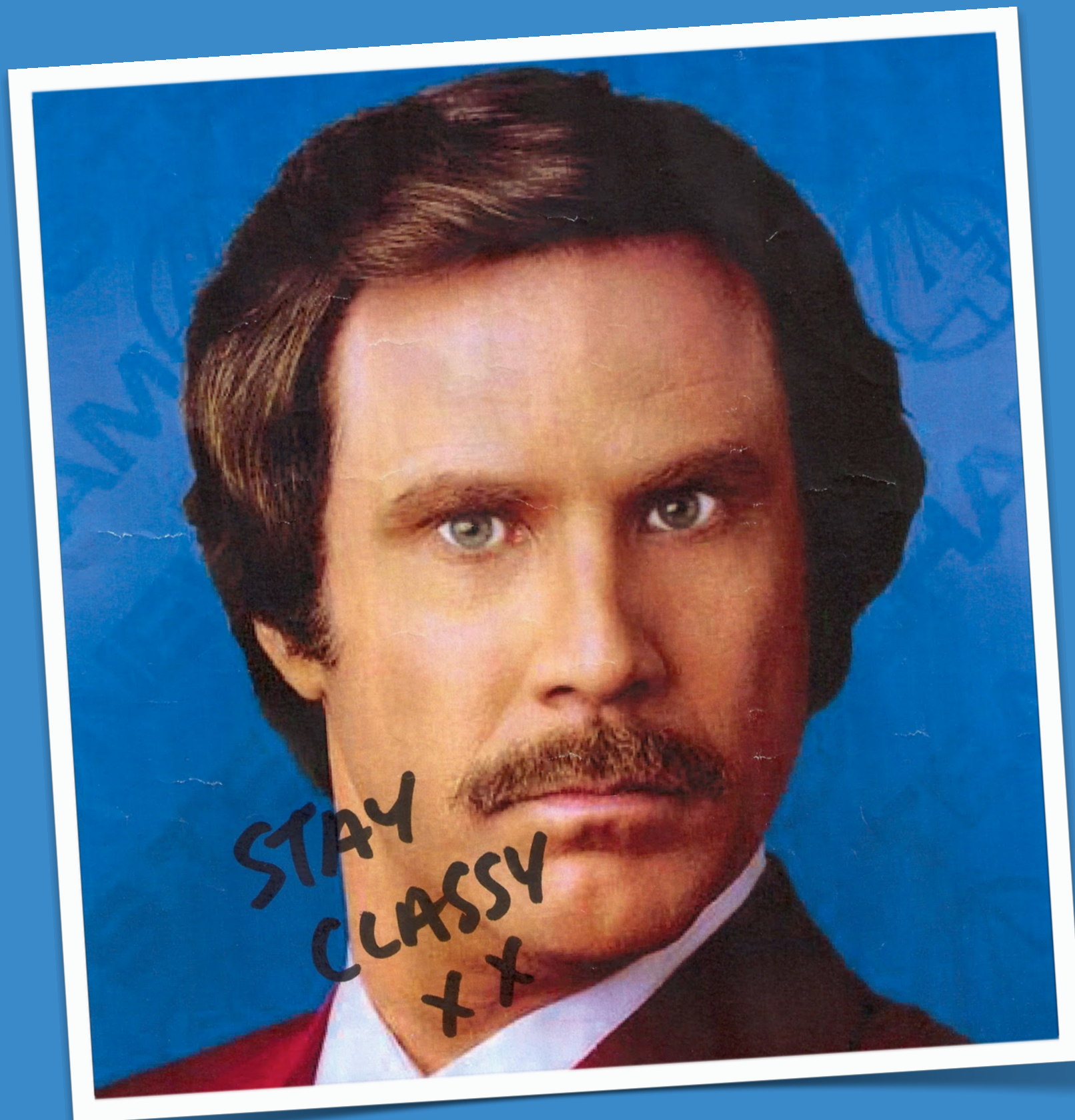
```
<li>Grumpy Cats</li>
```

```
</ul>
```

```
</div>
```

```
...
```

```
#content #related ul {  
    margin-left: 0;  
}  
#content #related ul li {  
    list-style: none;  
    background: url(rage-face.png);  
}  
#content #related .warning ul {  
    margin-left: 10px;  
}  
#content #related .warning ul li {  
    background: none;  
    list-style: disc;  
}
```



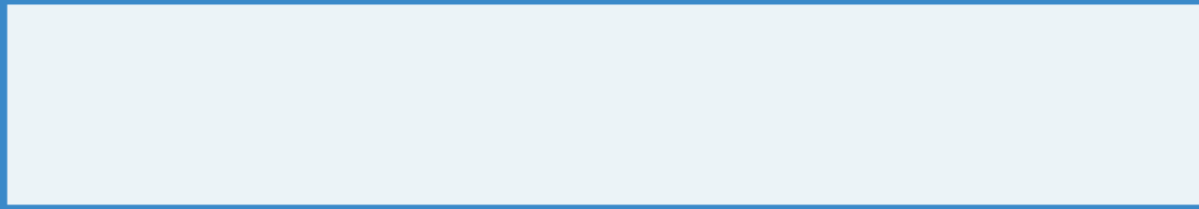
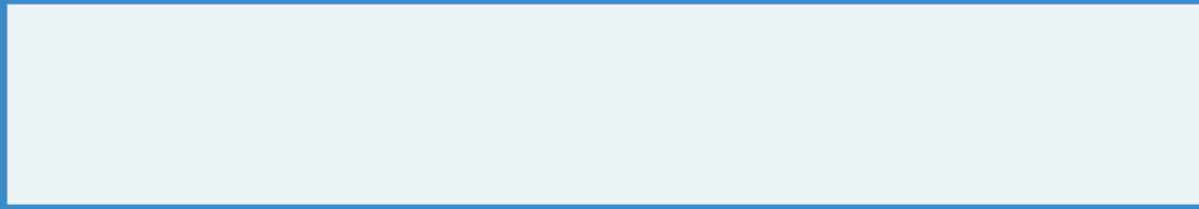
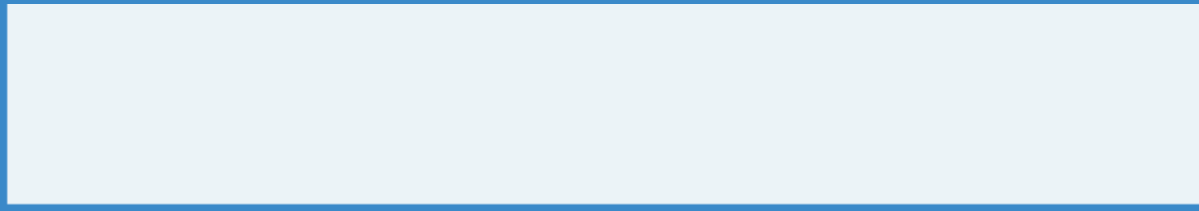
STAY
CLASSY
XX

```
<div class="content">  
  <h1>One does not simply:</h1>  
  <ul class="list">  
    <li class="item">...</li>  
    <li class="item">...</li>  
  </ul>  
</div>
```

```
<div class="content">  
  <h1>One does not simply:</h1>  
  <ul class="list">  
    <li class="item">...</li>  
    <li class="item">...</li>  
  </ul>  
</div>
```



What?!



```
.content .list {  
    margin-left: 10px;  
}  
  
.content .list .item {  
    list-style: square;  
}
```

```
<div class="content">
  ...
  <aside class="related">
    <h2>Other overused memes:</h2>
    <ul class="rage-list">
      <li class="item">...</li>
      <li class="item">...</li>
      <li class="item">...</li>
    </ul>
  </aside>
  ...
```



```
.content .list {...}
.content .list .item {...}
.rage-list .item {
  background: url(rage-face.png);
}
```

```
.content .list {...}  
.content .list .item {...}  
.rage-list .item {  
  background: url(rage-face.png);  
}
```

Look Ma!
No Overrides

```
<div class="content">
```

```
...
```

```
<aside class="related">
```

```
<div class="warning">
```

```
<h2>This post contains:</h2>
```

```
<ul class="list">
```

```
<li class="item">...</li>
```

```
<li class="item">...</li>
```

```
</ul>
```

```
</div>
```

```
...
```

```
<div class="content">
...
<aside class="related">
  <div class="warning">
    <h2>This post contains:</h2>
    <ul class="list">
      <li class="item">...</li>
      <li class="item">...</li>
    </ul>
  </div>
...

```

```
.content .list {...}  
.content .list .item {...}  
.range-list .item {...}  
.warning .list .item {  
  list-style: disc;  
}
```

```
#content ul {
    margin-left: 10px;
}
#content ul li {
    list-style: square;
}
#content #related ul {
    margin-left: 0;
}
#content #related ul li {
    list-style: none;
    background: url(rage-face.png);
}
#content #related .warning ul {
    margin-left: 10px;
}
#content #related .warning ul li {
    background: none;
    list-style: disc;
}
```

```
.content .list {  
    margin-left: 10px;  
}  
.content .list .item {  
    list-style: square;  
}  
.rage-list .item {  
    background: url(rage-face.png);  
}  
.warning .list .item {  
    list-style: disc;  
}
```

```
.content .list {  
  margin-left: 10px;  
}  
.content .  
  list-style-type: none;  
}  
rage-list .item {  
  background: url(rage-face.png);  
}  
.warning .list .item {  
  list-style: disc;  
}
```

Shorter Selectors


```
.content .list {  
  margin-left: 10px;  
}  
.content .list .item  
  list-style: square  
}  
.rage-list .item {  
  background: url(rage-face.png);  
}  
.warning .list .item {  
  list-style: disc;  
}
```

Less Code

```
.content .list {  
  margin-left: 10px;  
}  
.content .li  
  list-style  
}  
rage-list .item {  
  background: url(rage-face.png);  
}  
.warning .list .item {  
  list-style: disc;  
}
```

Less Duplication

```
.content .list {  
  margin-left: 10px;  
}  
.content .list .item  
  list-style: square;  
}  
.rage-list .item {  
  background: url(rage-face.png);  
}  
.warning .list .item {  
  list-style: disc;  
}
```

More Re-use

Write to
Re-write

Fix Bugs

LATEST: 10.17

UPDATE

CHANGES IN VERSION 10.17:
THE CPU NO LONGER OVERHEATS
WHEN YOU HOLD DOWN SPACEBAR.

COMMENTS:

LONGTIMEUSER4 WRITES:

THIS UPDATE BROKE MY WORKFLOW!
MY CONTROL KEY IS HARD TO REACH,
SO I HOLD SPACEBAR INSTEAD, AND I
CONFIGURED EMACS TO INTERPRET A
RAPID TEMPERATURE RISE AS "CONTROL".

ADMIN WRITES:

THAT'S HORRIFYING.

LONGTIMEUSER4 WRITES:

LOOK, MY SETUP WORKS FOR ME.
JUST ADD AN OPTION TO REENABLE
SPACEBAR HEATING.

EVERY CHANGE BREAKS SOMEONE'S WORKFLOW.

<http://xkcd.com/1172/>

Fixing Bugs

Fixing Bugs

It's not the ^{bug}~~fall~~ that
kills you

Fixing Bugs

It's the ~~sudden stop~~

^
backwards compatibility

Fear of Regression

Fear of Regression
is a friction

Fear of Regression
slows progress

Fear of Regression
stifles innovation

Fear of Regression
is what stops you

No Fear

Version your code

Don't Push

Let teams pull

But I only have one
CSS file!

One file per component

But I only have one
CSS file!

“Man, I hate
this solution...
but it works”

“Man,
this solution
is messed up;
what does it
even do?”

```
div {  
  background/**/: blue;  
}
```

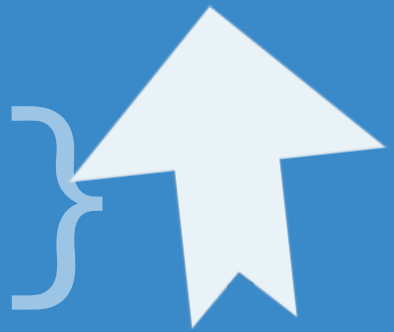
```
div {  
  background/**/: blue;  
}
```



Hack or fat-finger?

```
div {  
  #color: blue;  
}
```

```
div {  
  #color: blue;
```



Huh?

CSSDoc

```
div {  
  /**  
   * @workaround IE color bug  
   * @affected IE6, IE7, IE8  
   * @css-for IE6, IE7  
   */  
  #color: blue;  
}
```

```
div {  
  /**  
   * @workaround IE color bug  
   * @affected IE6, IE7, IE8  
   * @css-for IE6, IE7  
   */  
  #color: blue;  
}
```

```
div {  
  /**  
   * @workaround IE color bug  
   * @affected IE6, IE7, IE8  
   * @css-for IE6, IE7  
   */  
  #color: blue;  
}
```

```
div {  
  /**  
   * @workaround IE color bug  
   * @affected IE6, IE7, IE8  
   * @css-for IE6, IE7  
   */  
  #color: blue;  
}
```

```
div {  
  /**  
   * @workaround IE color bug  
   * @affected IE6, IE7, IE8  
   * @css-for IE6, IE7  
   */  
  #color: blue;  
}
```

```
div {  
  /**  
   * @workaround IE color bug  
   * @affected IE6, IE7, IE8  
   * @css-for IE6, IE7  
   */  
  #color: blue;  
}
```

”

”


```
div {  
    color: blue;  
    height : 12px; }  
p{ color: #339; }
```

```
div {  
  color: #0F0;  
  height: 12px;  
}  
  
p {  
  color: #339;  
}
```

Use Design Patterns

You already know
them

Leave out
the details
that don't
matter

Build a
pattern
library

Don't start
an arms
race with
selectors

Version your
CSS and
live free
again

Document
your
intentions

Treat style
with respect

SMA CSS

OO CSS

Credits

Font Awesome by Dave Gandy

<http://fontawesome.github.com/Font-Awesome>

Inconsolata Font

<http://www.levien.com/type/myfonts/inconsolata.html>

Kudos to Michael P. Gilbert

<http://www.mpgilbert.com/>

Thanks

@klamping