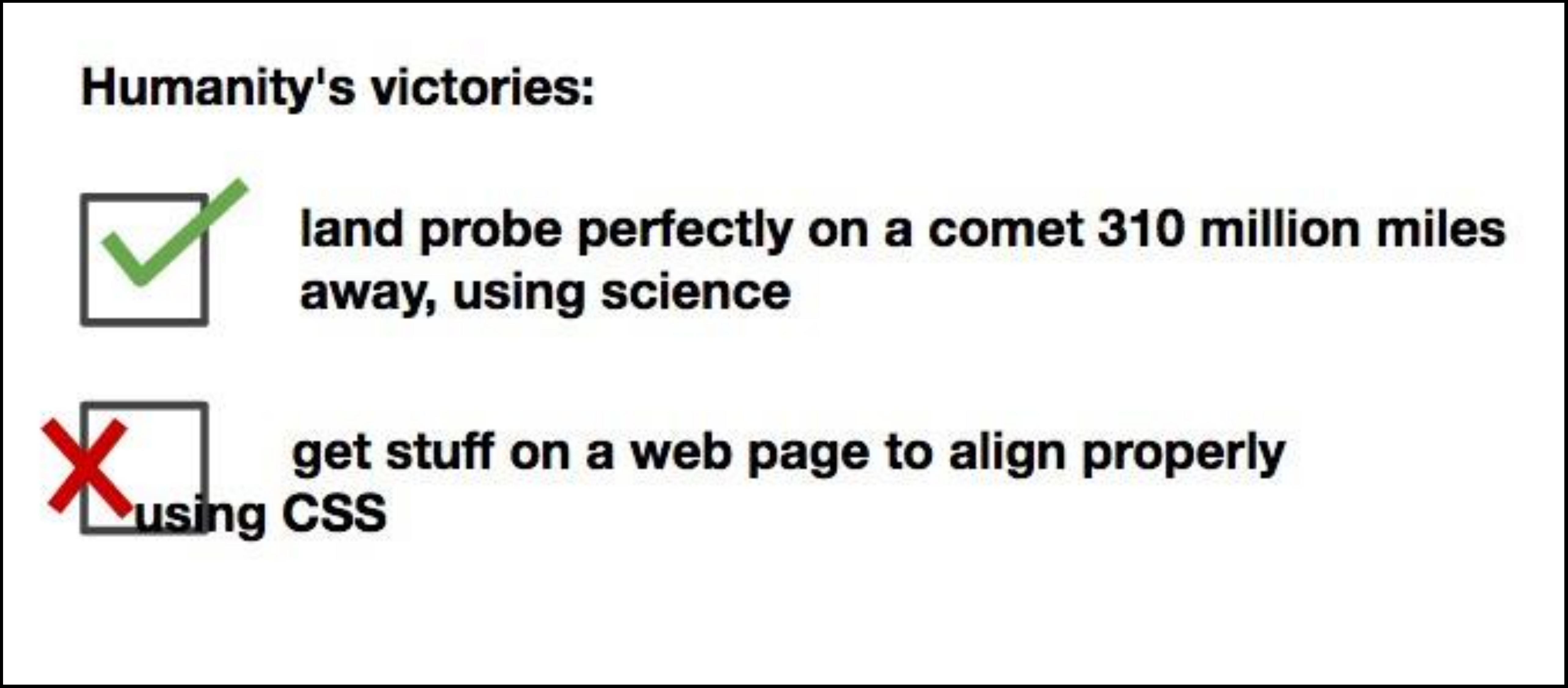
JEN KRAMER
HARVARD UNIVERSITY EXTENSION SCHOOL

CSS4 GRID: TRUE LAYOUT FINALLY ARRIVES

www.github.com/jen4web/css4grid



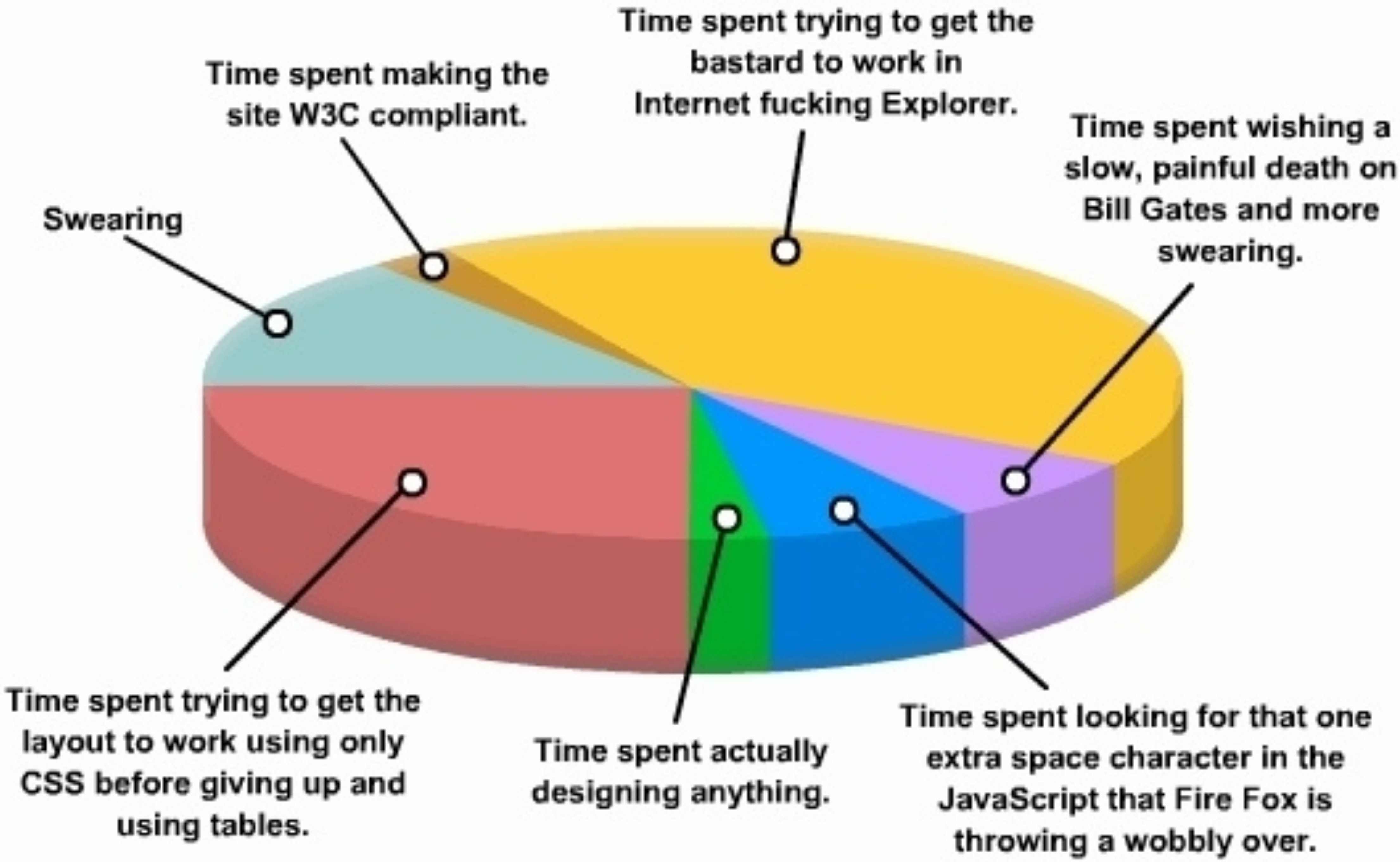
Humanity's victories:

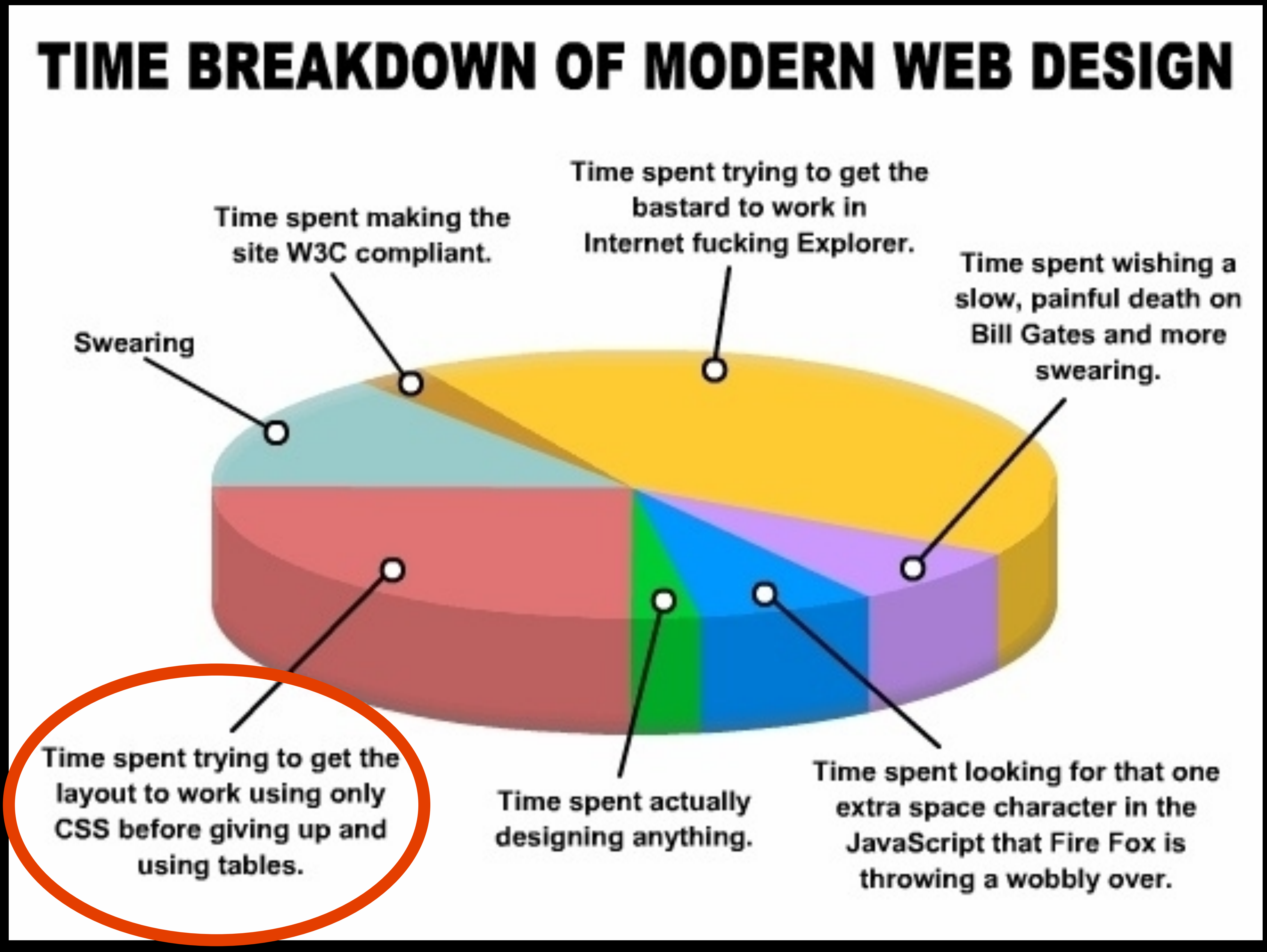
☑ land probe perfectly on a comet 310 million miles away, using science

☒ get stuff on a web page to align properly using CSS

EXIT
EXIT
A
B
C
1
D
E
F
2
3
4

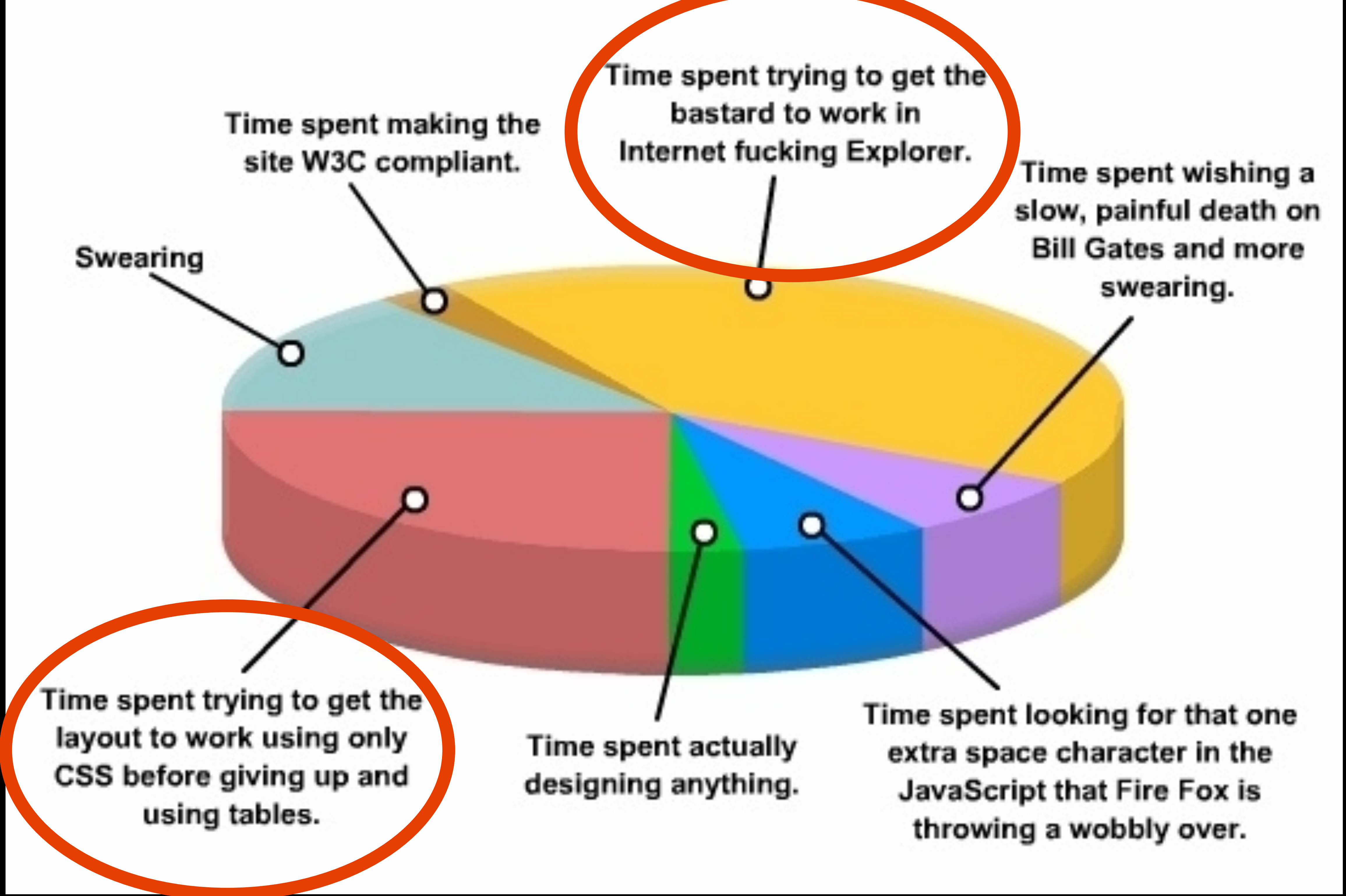
TIME BREAKDOWN OF MODERN WEB DESIGN





TIME BREAKDOWN OF MODERN WEB DESIGN
Swearing
Time spent making the site W3C compliant.
Time spent trying to get the bastard to work in Internet fucking Explorer.
Time spent wishing a slow, painful death on Bill Gates and more swearing.
Time spent trying to get the layout to work using only CSS before giving up and using tables.
Time spent actually designing anything.
Time spent looking for that one extra space character in the JavaScript that Fire Fox is throwing a wobbly over.

TIME BREAKDOWN OF MODERN WEB DESIGN

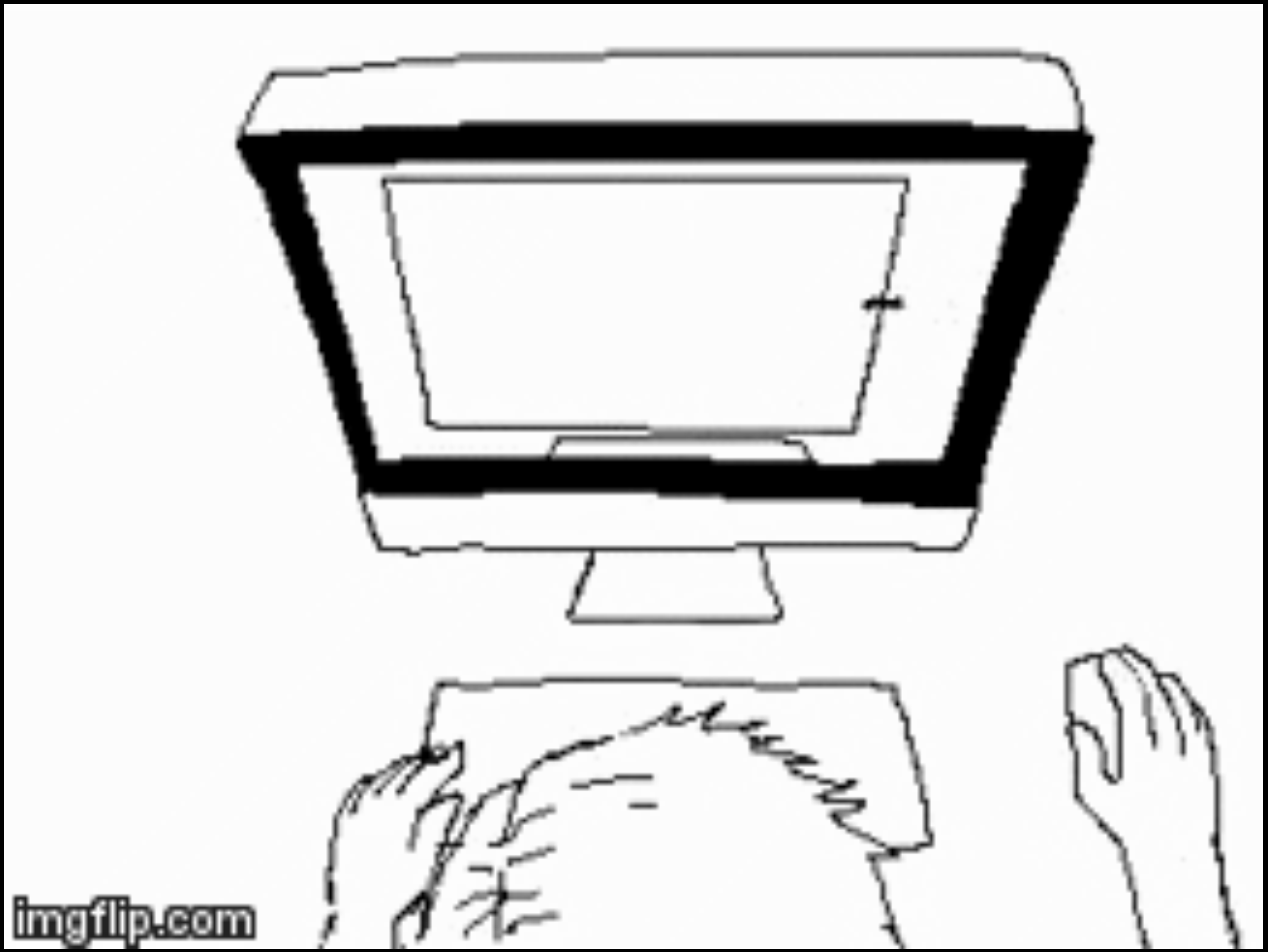


PART 1

RESPONSIVE DESIGN

RESPONSIVE DESIGN

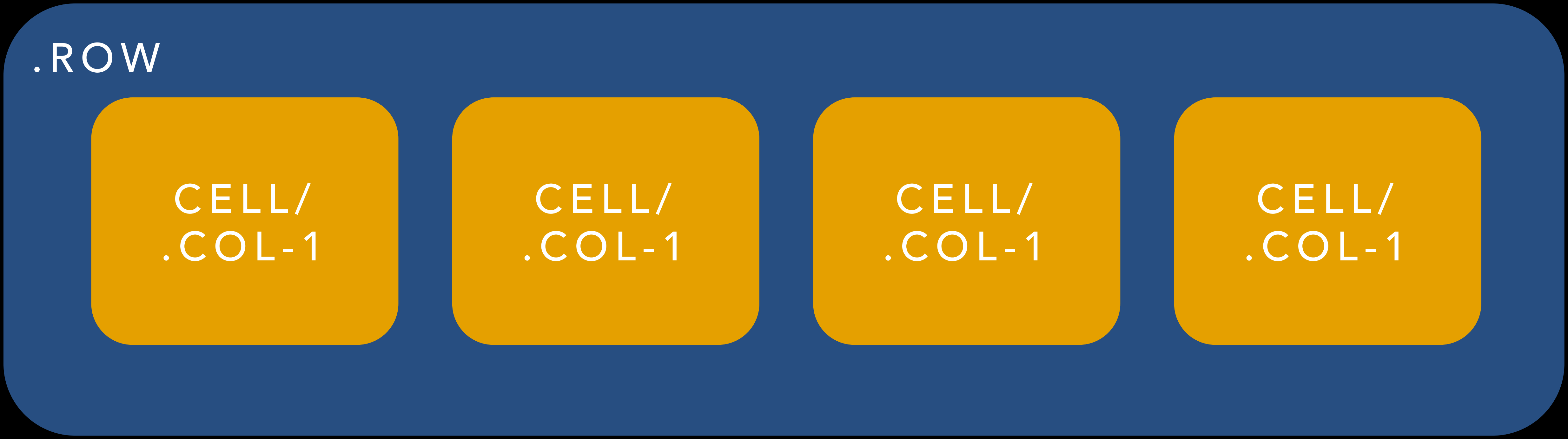
- Defined by three characteristics
    - Flexible grid-based layout
    - Media queries (CSS3)
    - Images that resize
- www.alistapart.com/articles/responsive-web-design/



imgflip.com

FLOATS

- A hack from the start, right after table-based layout!
- Features rows and cells.
- Rows clear the floats on the cells.
- Source ordering determines display, though some (minor) rearrangement is possible.
- Major disadvantage: equal column heights



.ROW

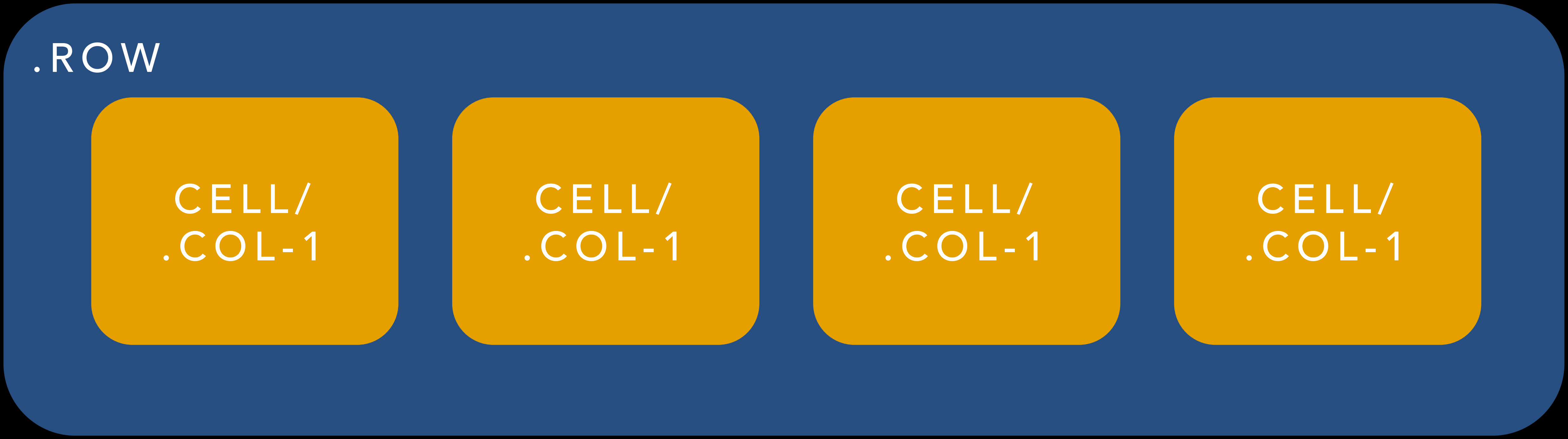
CELL/ .COL-1

CELL/ .COL-1

CELL/ .COL-1

CELL/ .COL-1

```
<div class=“row”>
     <div class=“col-1”></div>
     <div class=“col-1”></div>
     <div class=“col-1”></div>
     <div class=“col-1”></div>
</div>
```



.ROW

CELL/ .COL-1

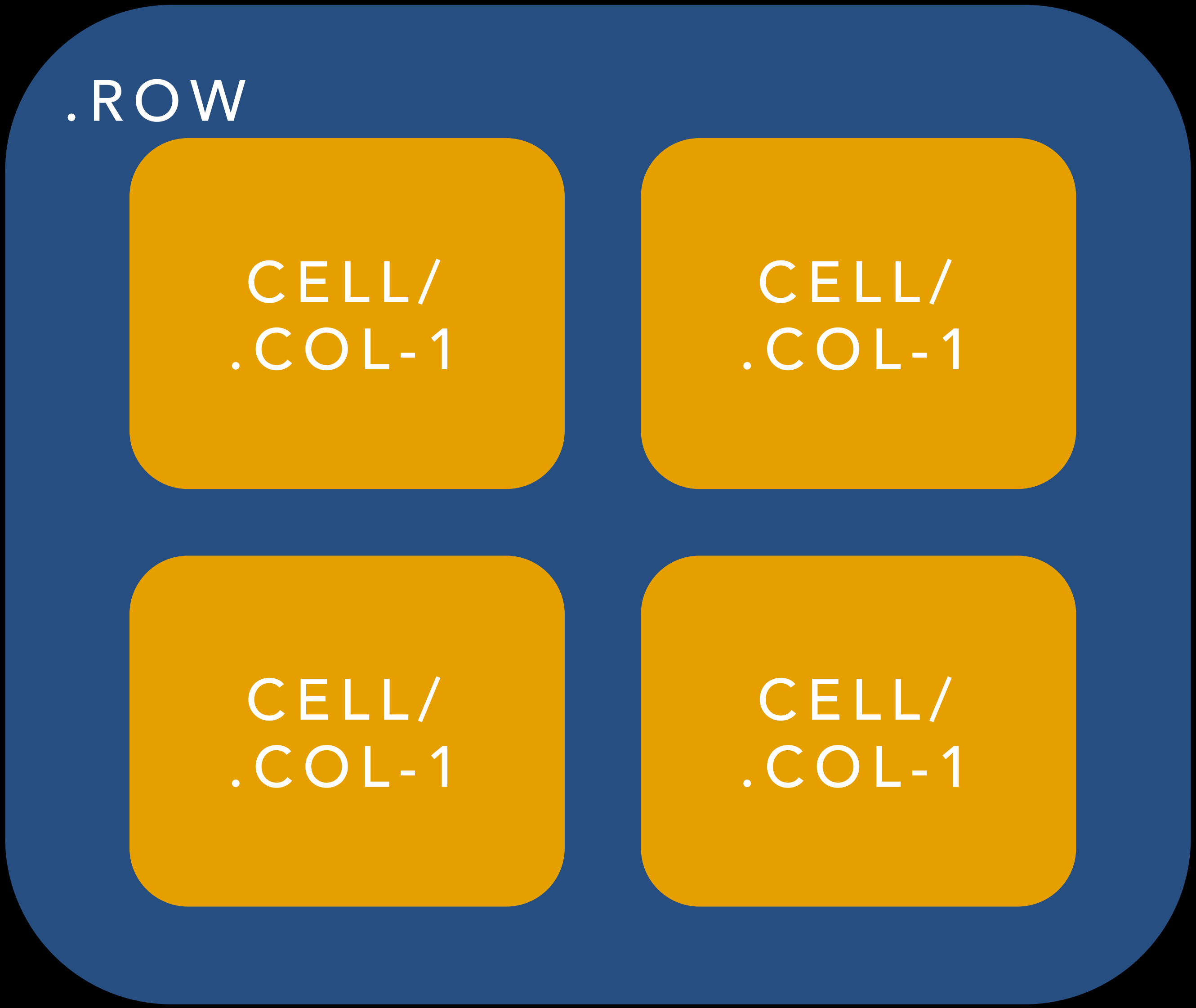
CELL/ .COL-1

CELL/ .COL-1

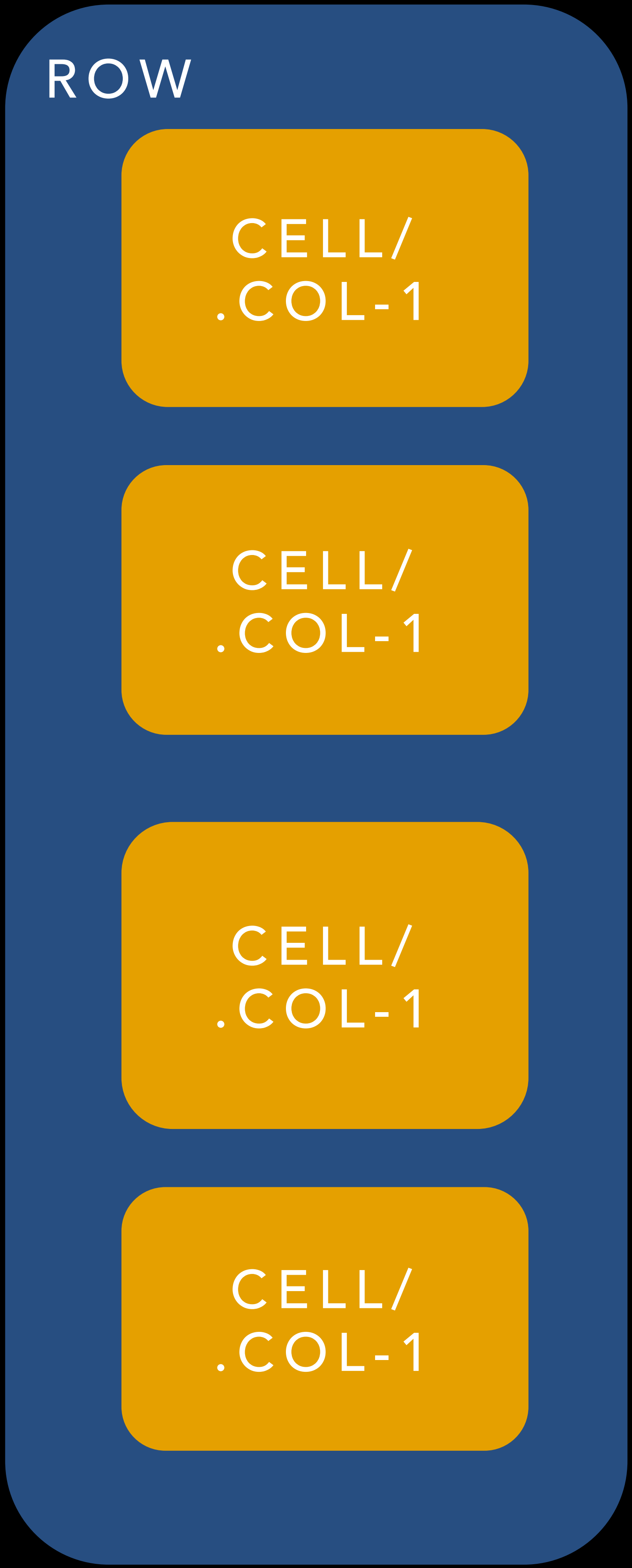
CELL/ .COL-1

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

```
.col-1 {
   float: left;
   margin-left: 4%;
   width: 20%;
}
```

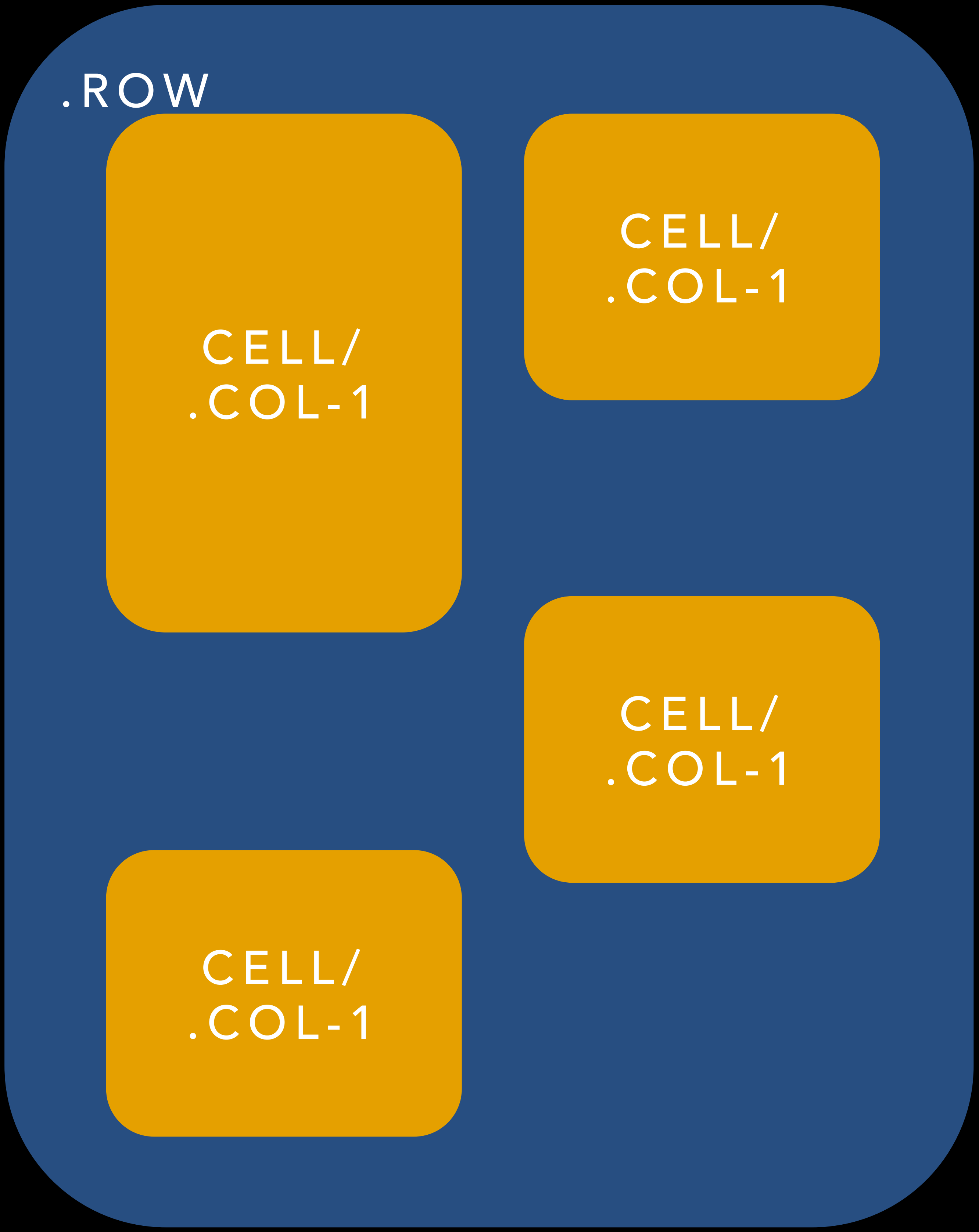


```
@media only screen and (min-width: 480px)
                   and (max-width: 767px) {
    .col-1 {
       width: 44%;
    }
}
```



```
@media only screen and
       (max-width: 479px) {

    .col-1 {
        width: 98%;
        margin: 1%;
        float: none;
    }
}
```



There can be layout problems with floats.

This can be resolved with JavaScript, with a column equalizer script.

```
/* rearranging the columns */
[class*="col-"] {
    position: relative;
}
.col-push-1 {
    left: 26%;
}
.col-pull-3 {
    left: -74%;
}
```

NEXT UP

FLEXBOX

FLEXBOX

- The first layout elements – but not designed to lay out whole web pages
- Features flex-containers (row) and flex-items (cells). Both are required to work.
- Excels at vertical centering and equal heights
- Very easy to reorder boxes
- Major disadvantages:
  - Wasn’t designed to be locked down for layouts! Works in 1 dimension only.
  - Browser support and syntax is challenging.

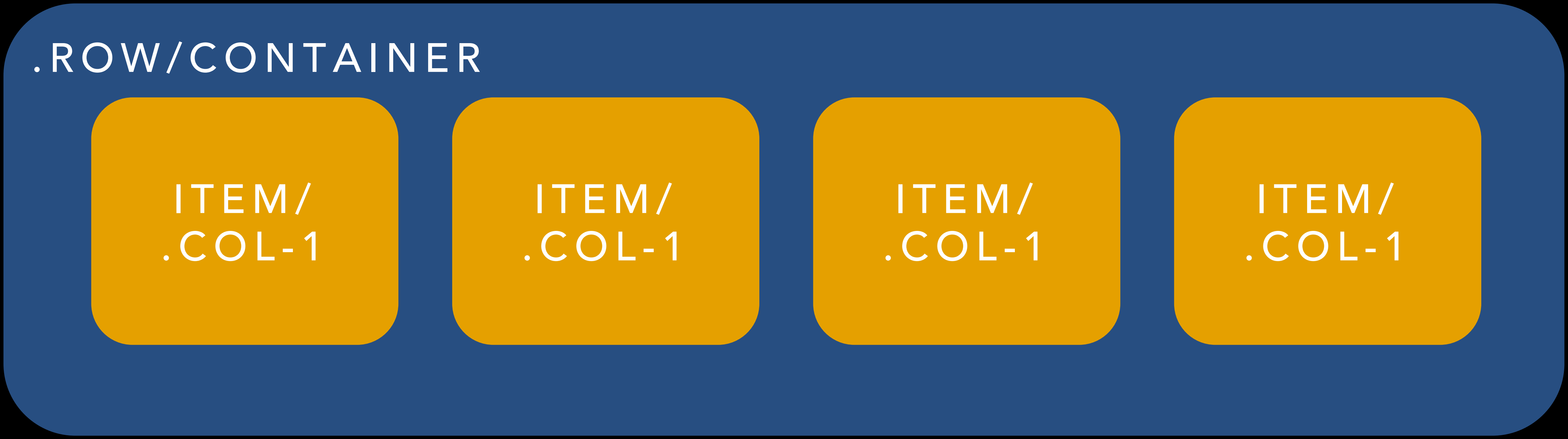
THREE VERSIONS OF FLEXBOX

- 2009: `display: box;`
- 2011: `display: flexbox;` ("tweener" syntax)
- 2016: `display: flex;`
- Prefixing may still be required depending on browser support desired

```
.flex-container {
    display: -ms-flexbox;
    display: -webkit-flex;
    display: flex;
}
```

CURRENT SUPPORT

- Internet Explorer
  - <= IE 9: not supported
  - IE 10 supports “tweener” syntax (ms prefix)
  - IE 11, Edge: Full support (though buggy in IE 11)
- Safari 7.1/8, iOS Safari 7/8 require webkit prefix
- Others support current syntax (including Opera!)
- http://caniuse.com/#feat=flexbox


.ROW/CONTAINER
ITEM/
.COL-1
ITEM/
.COL-1
ITEM/
.COL-1
ITEM/
.COL-1

```
<div class=“row”>
    <div class=“col-1”></div>
    <div class=“col-1”></div>
    <div class=“col-1”></div>
    <div class=“col-1”></div>
</div>
```

```
.row {
  display: flex;
  flex: 0 1 auto;
  flex-flow: row wrap;
  justify-content: center;
  margin: 1%;
}
```

Change flex-flow to other values to change direction of rows – row reverse, column reverse, no wrap

```
.col-1 {
    flex: 0 0 24%; /* desktop */
    flex: 0 0 48%; /* tablet */
    flex: 0 0 98%; /* phone */
}
```

To change widths on .col-1, change the flex-basis property. This is more flexible than width.

```
/* rearranging the columns */

.col-push-1 {
  order: 2;
}
.col-pull-3 {
  order: 1;
}
```

FINALLY!

CSS4 GRID

WHY CSS4 GRID?

- Built into CSS4 specification (in development).
- No “row” markup required.
- Grid is designed to work in 2 dimensions.
- Use Flexbox for UI elements, but use Grid for major layout.

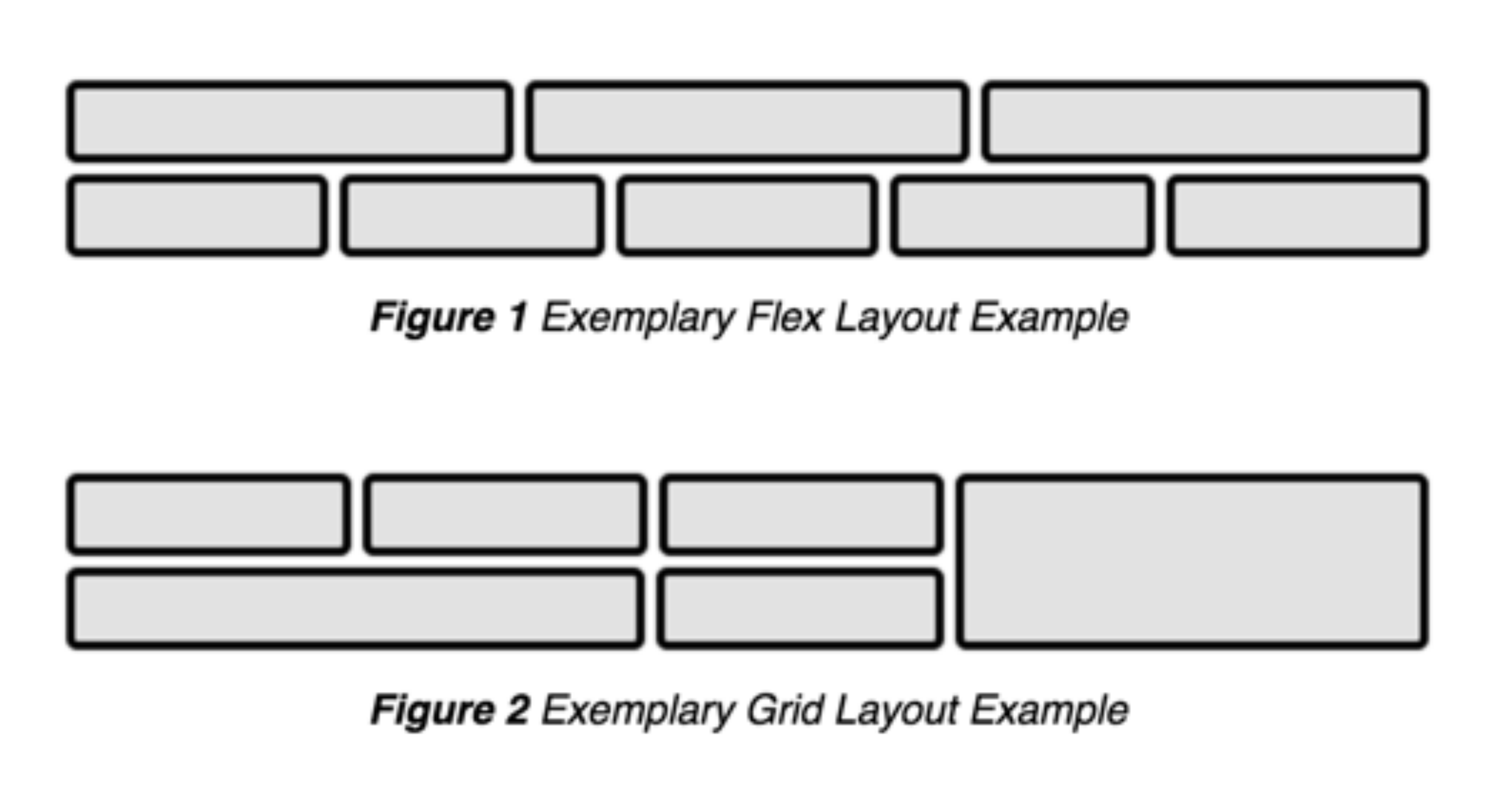


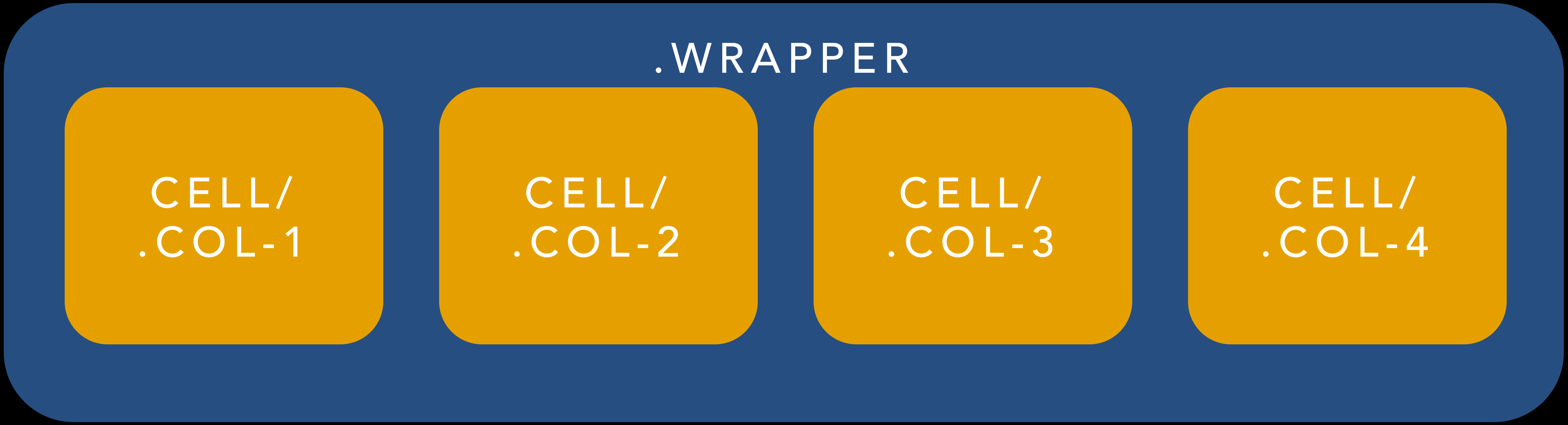
Figure 1 *Exemplary Flex Layout Example*

Figure 2 *Exemplary Grid Layout Example*

https://drafts.csswg.org/css-grid/

ISSUES, WE'VE HAD A FEW

- Almost no browser support without flags: http://igalia.github.io/css-grid-layout/enable.html
- Spec isn't fully implemented in some browsers, and/or there are bugs



.WRAPPER

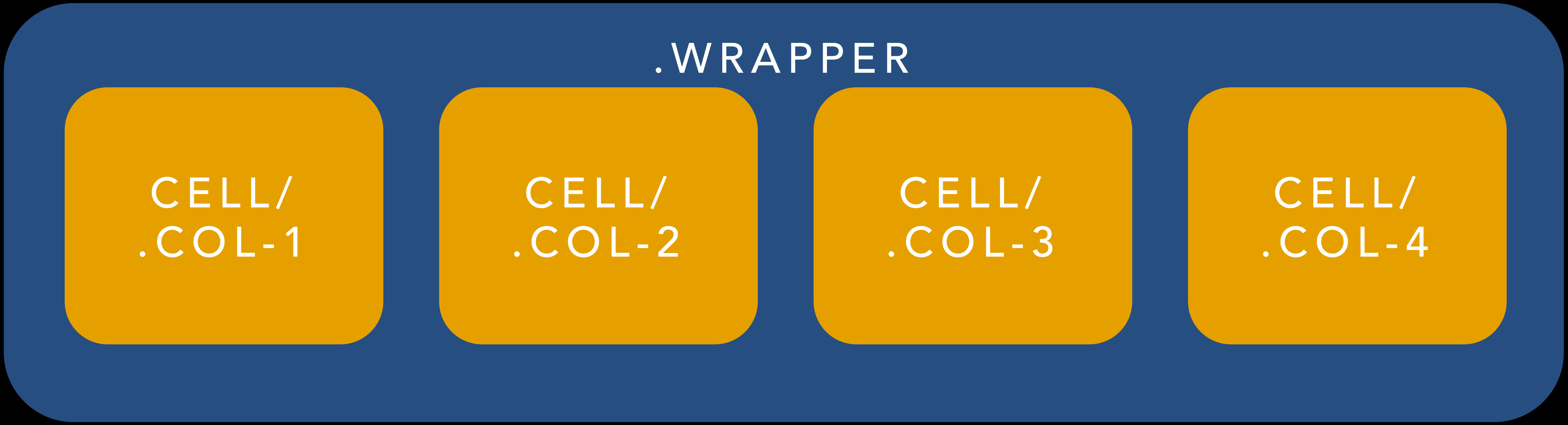
CELL/ .COL-1

CELL/ .COL-2

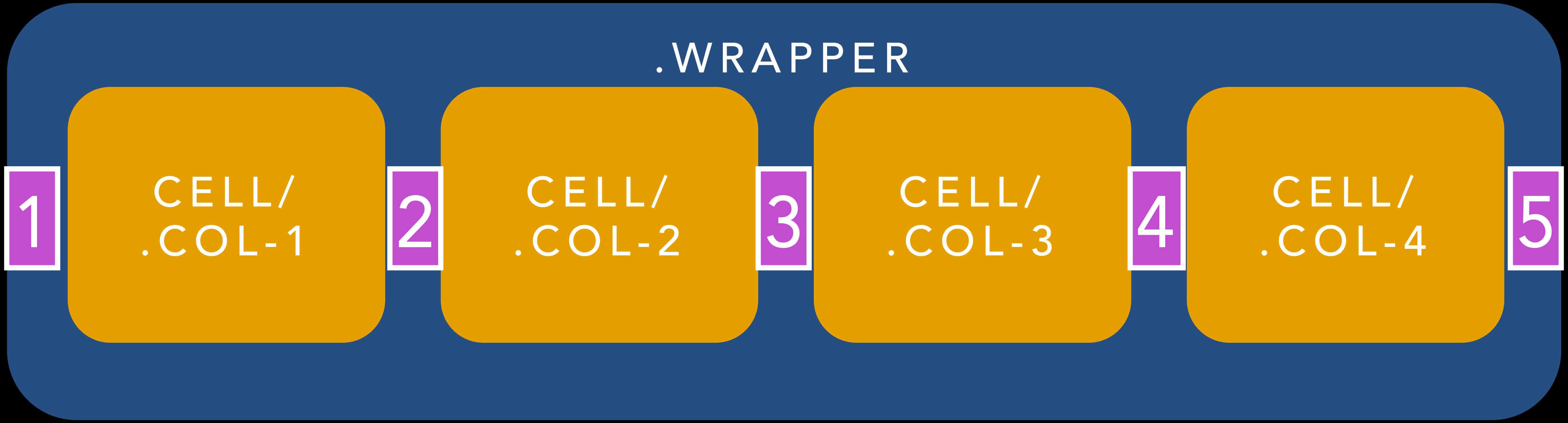
CELL/ .COL-3

CELL/ .COL-4

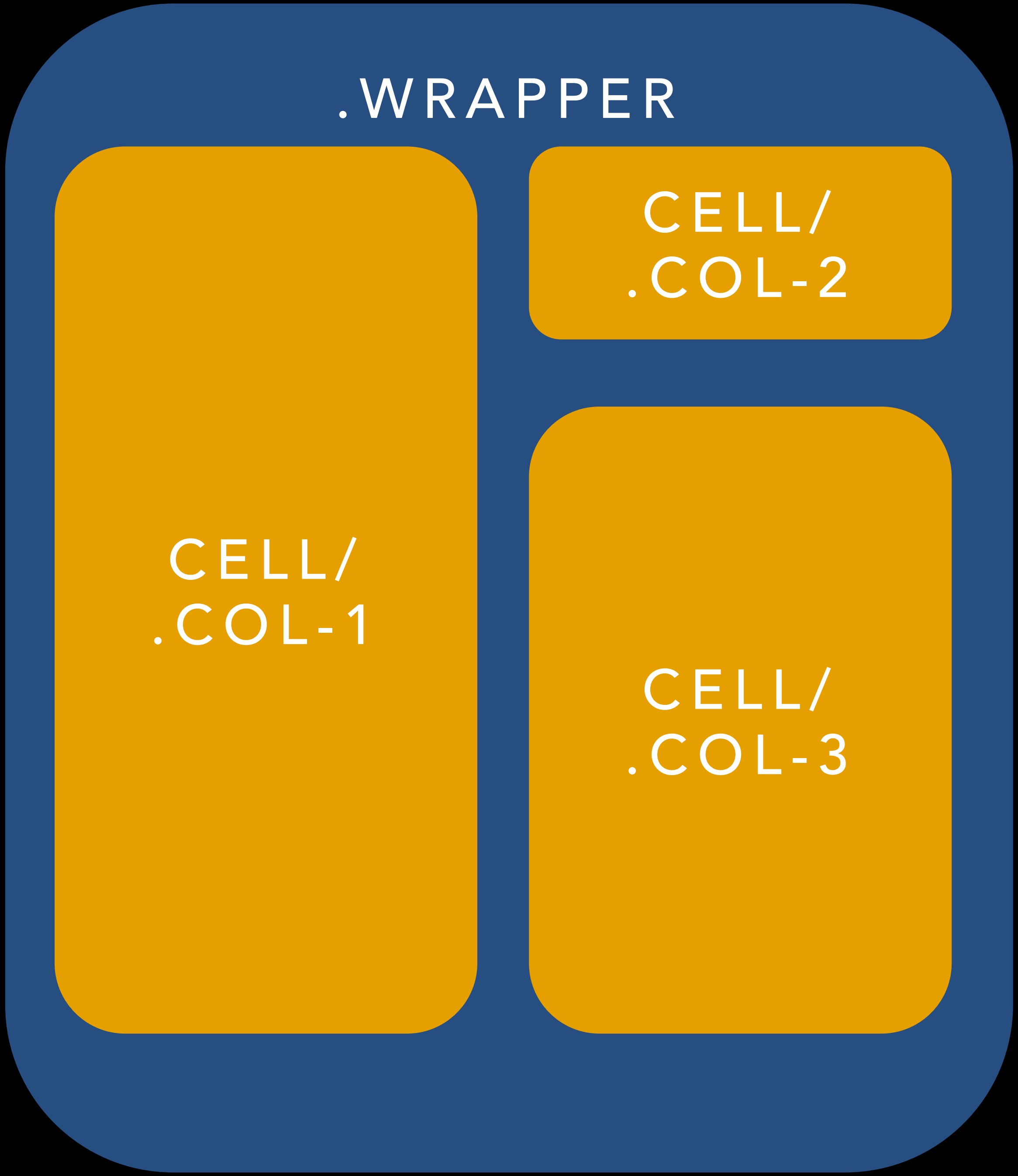
```
<div class=“col-1”></div>
<div class=“col-2”></div>
<div class=“col-3”></div>
<div class=“col-4”></div>
```



```
.wrapper {
  display: grid;
  grid-gap: 10px;
}
```



```
.col-1 {
  grid-column: 1 / 2;
}
.col-2 {
  grid-column: 2 / 3;
}
.col-3 {
  grid-column: 3 / 4;
}
```

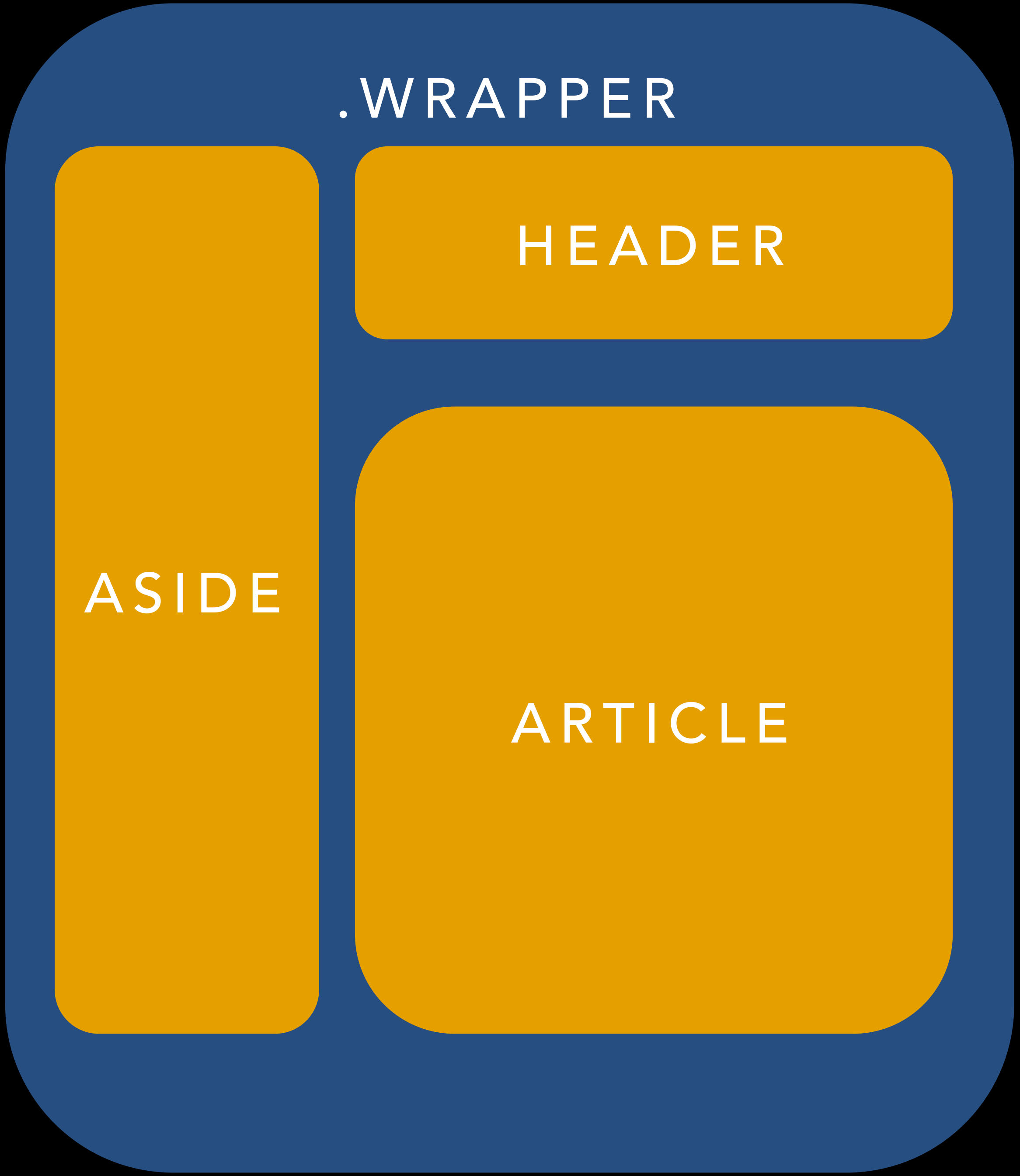


```
.col-1 {
  grid-column: 1 / 2;
  grid-row: 1 / 3;
}
.col-2 {
  grid-column: 2 / 3;
  grid-row: 1 / 2;
}
.col-3 {
  grid-column: 2 / 3;
  grid-row: 2 / 3;
}
```

ALTERNATE SYNTAX

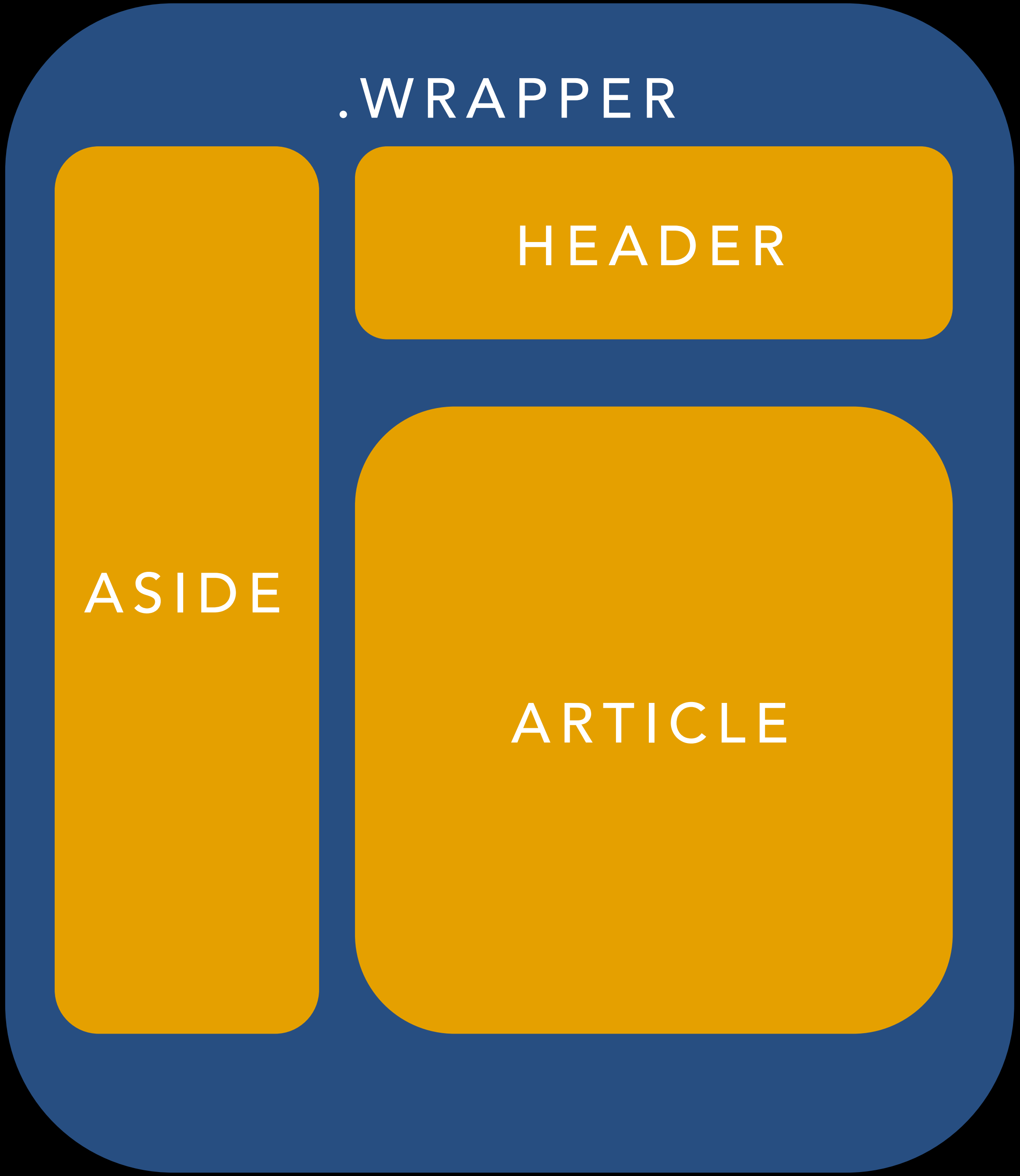
- Named grid template areas (header, footer, etc): http://gridbyexample.com/examples/#example11

REORDERING



```
<div class="wrapper">
   <header></header>
   <article></article>
   <aside></aside>
</div>
```

REORDERING



Show code in GitHub!

reordering.html

reordering.css

https://github.com/jen4web/css4grid

CODE EXAMPLE

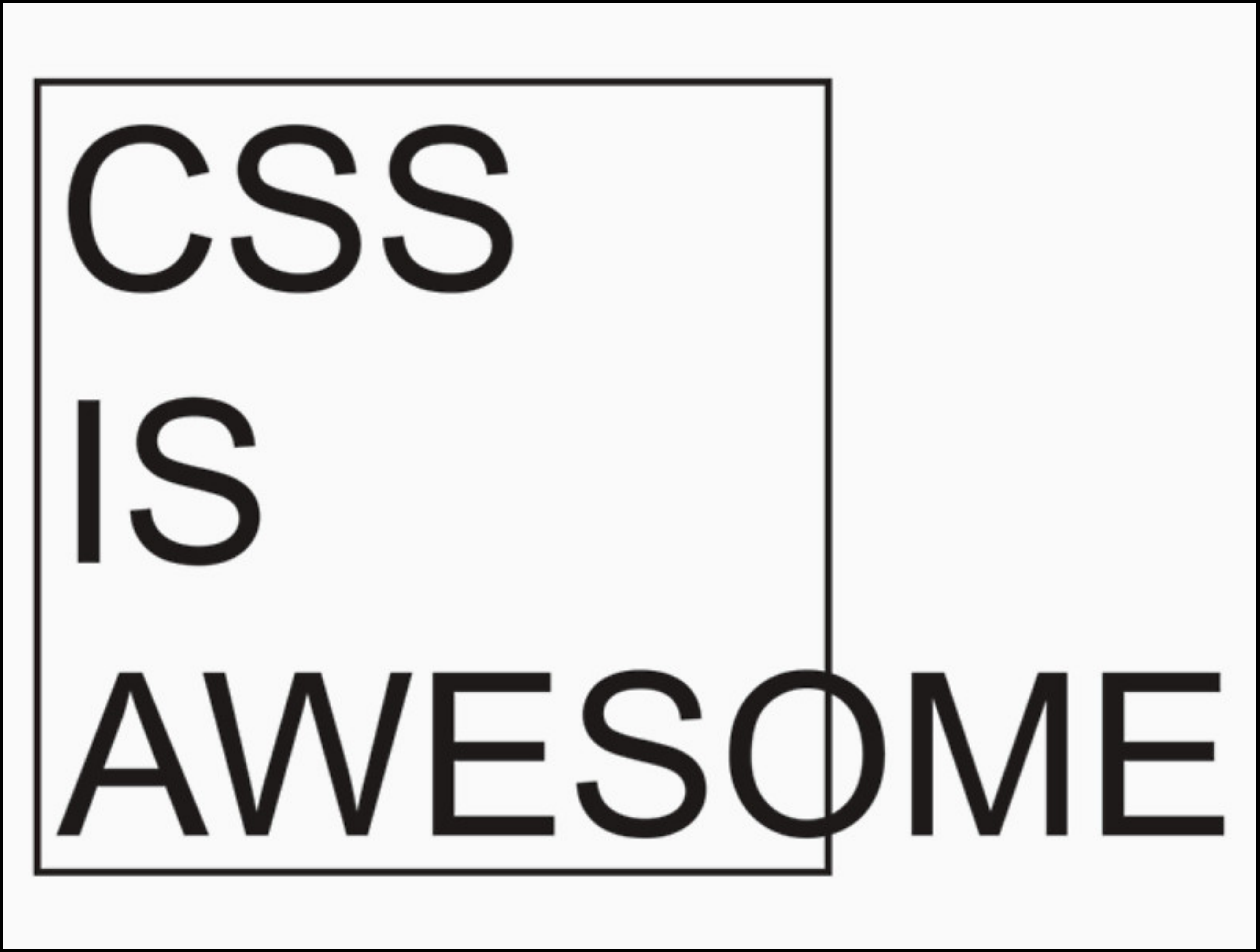
www.github.com/jen4web/css4grid

CURRENT RECOMMENDATIONS

- Need browser support? Can’t go wrong with floats.
- If you are OK with a few prefixes and learning something new, Flexbox may be a reasonable interim step.
- Keep waiting for CSS4 Grid! Coming to a browser near you Real Soon!

RECOMMENDED RESOURCES

- CSS Tricks: https://css-tricks.com/snippets/css/complete-guide-grid/
- Grid by Example: http://gridbyexample.com/
- W3C Spec: https://www.w3.org/TR/css-grid-1/
- W3C Editor’s Draft, updated Sept 8: https://drafts.csswg.org/css-grid/



CSS
IS
AWESOME

QUESTIONS?

Jen Kramer

Arlington, MA, USA

Phone: 802-257-2657

jen@jenkramer.org

www.jenkramer.org

Twitter: @jen4web

Facebook: facebook.com/webdesignjen

Code available at
www.github.com/jen4web/css4grid

Slides available at
www.slideshare.net/jen4web