Responsive Web Design with Flexbox and CSS Grid

Day 2: CSS Grid

Jen Kramer @jen4web

Why CSS Grid?

- Built into CSS specification (now a recommendation).
- No “row” markup required.
  - Mind the parents and children.
- Grid is designed to work in 2 dimensions.
- ~~“Use Flexbox for UI elements, but use Grid for major layout.”~~
- **Use Flexbox and Grid where they make the most sense.**

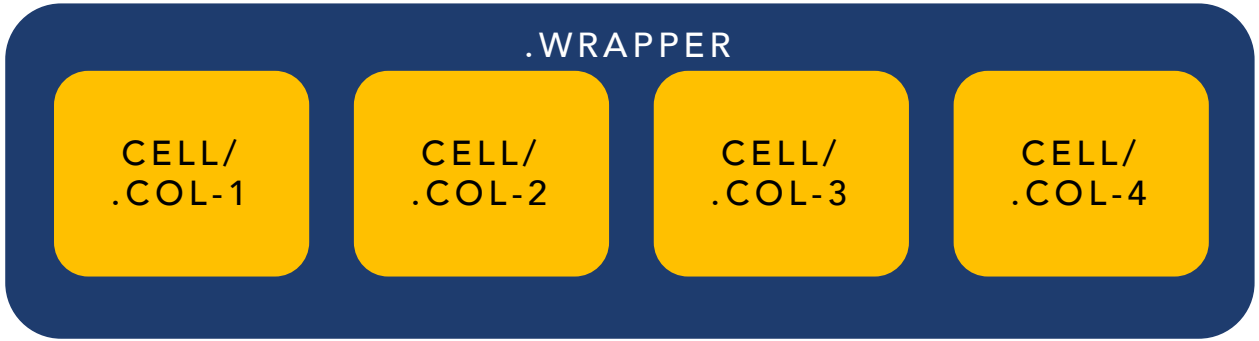
Figure 1 *Exemplary Flex Layout Example*

Figure 2 *Exemplary Grid Layout Example*

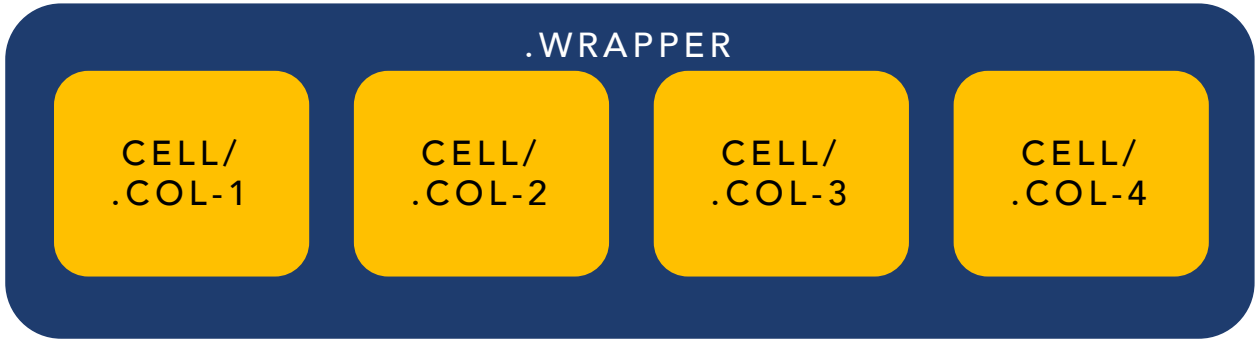
https://drafts.csswg.org/css-grid/

Browser support

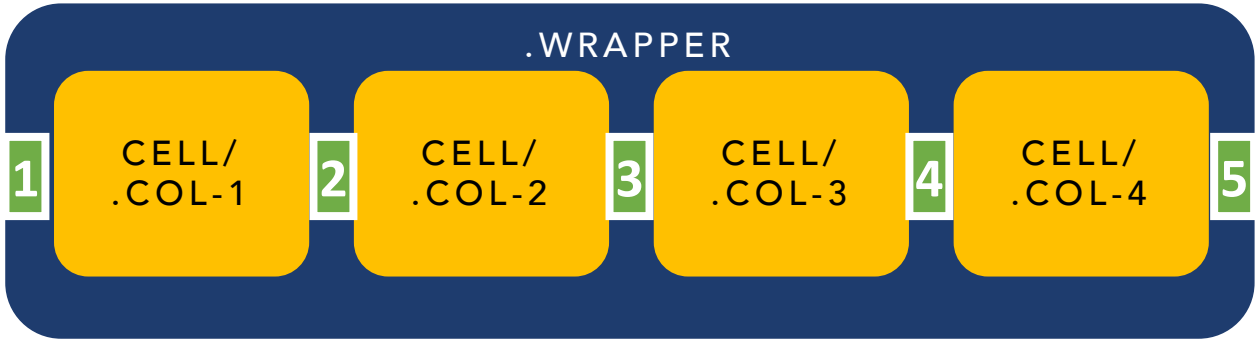
- **CSS Grid** (level 1) supported by all major modern browsers.
- **Subgrid** (level 2) supported only by Firefox.
- Partial support/no support with the usual suspects.
- http://caniuse.com/#search=grid



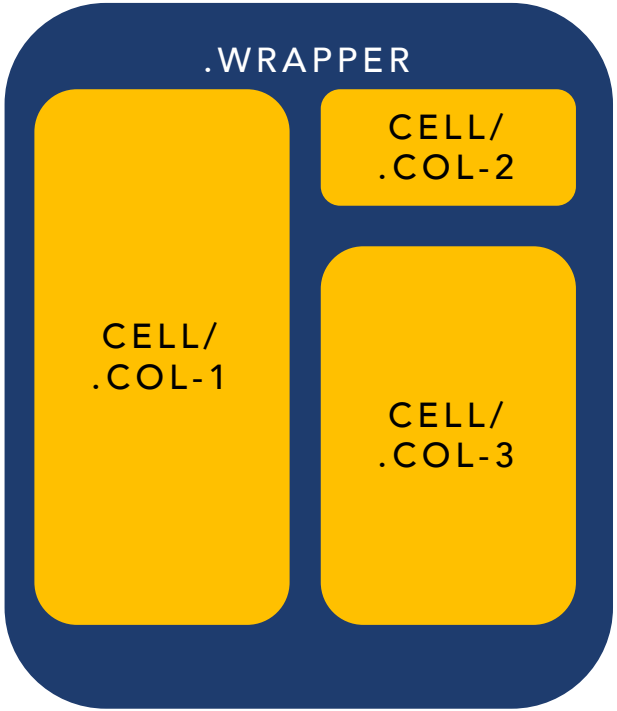

```
<div class=“wrapper”>
  <div class=“col-1”></div>
  <div class=“col-2”></div>
  <div class=“col-3”></div>
  <div class=“col-4”></div>
</div>
```




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



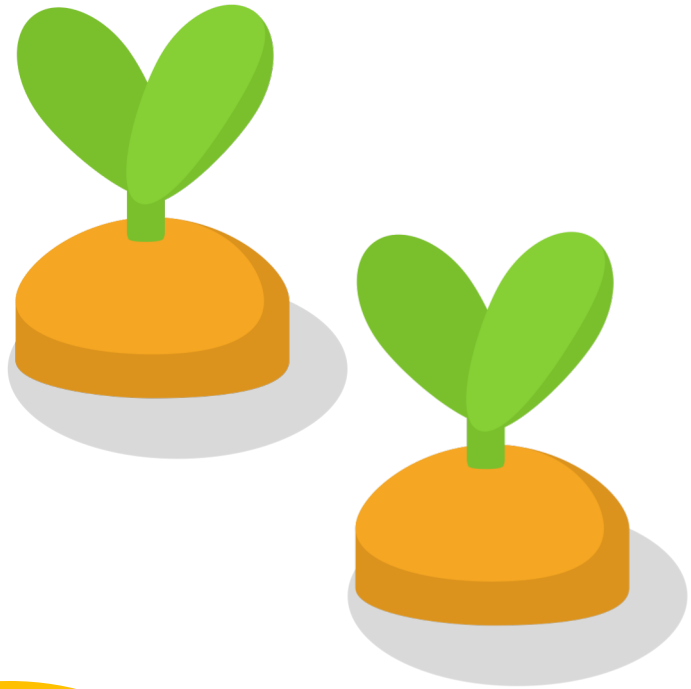

```
.col-1 {
    grid-column: 1 / 2;
}
.col-2 {
    grid-column: 2 / 3;
}
.col-3 {
    grid-column: 3 / 4;
}
```




```
.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;
}
```

Additional Syntax

- Span notations
- Named lines (instead of numbers)
- Named grid template areas (header, footer, etc)
- Tons of shorthand vs. longhand for properties
- **This is one of the most confusing parts of Grid**

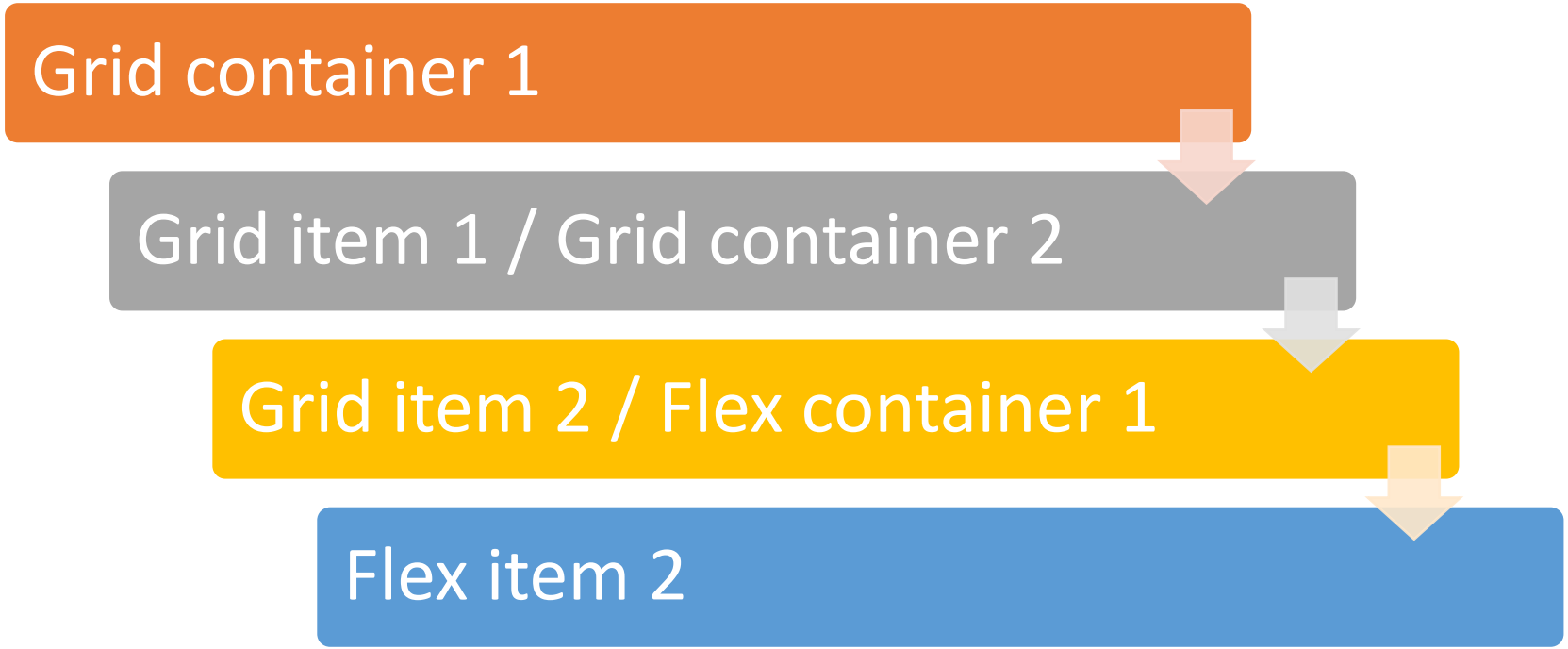


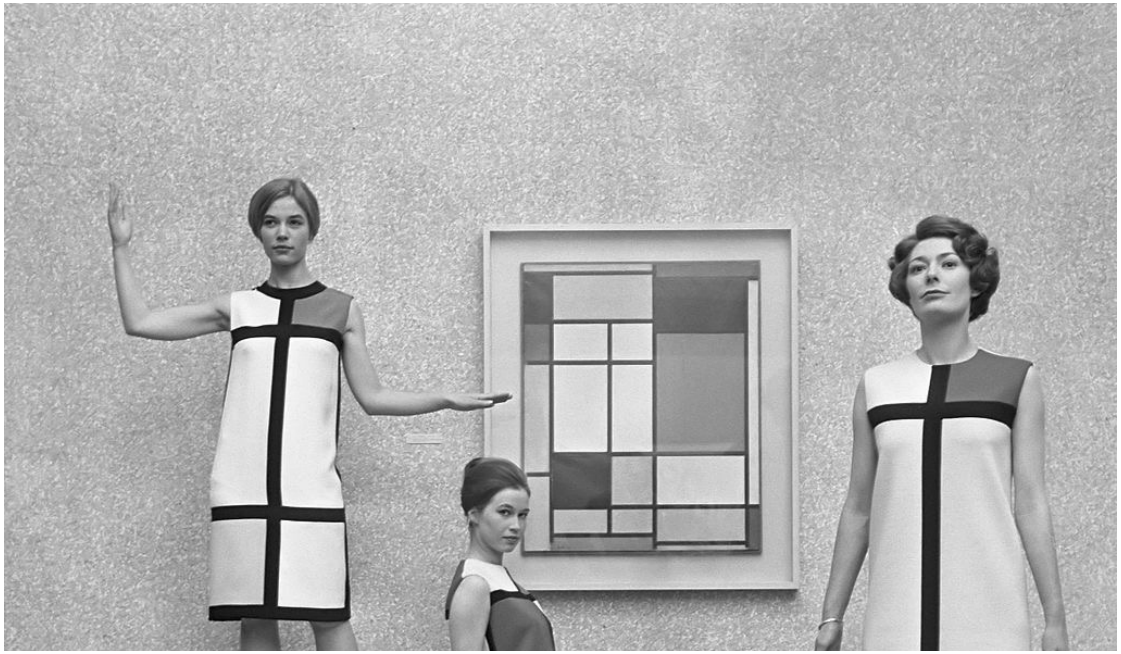
- Game to practice Grid syntax:
  - CSS Grid Garden: https://cssgridgarden.com

Tracking parents (containers) and children (items)

A single element may be

- A grid item AND a grid container
- A flex item AND a flex container
- A flex item AND a grid container
- A grid item AND a flex container


Grid container 1
Grid item 1 / Grid container 2
Grid item 2 / Flex container 1
Flex item 2



Code example

- 9-mondrian-painting