



# RWD

## Class 7

@rwdkent

# Overview

Fonts & Media Queries

Becoming Responsive

Mini Exam Study Guide

Hands-On Activity

Layout and Flexible Grids

Boxes & Layout Assignment



# Fonts + Media Queries Review



# Hands-On Exercise

<http://codepen.io/challahan/pen/ZLgPoR>



# Becoming Responsive

Responsive Process (Chapter 5, Responsive Web Design Book)



How can design and  
development become more  
collaborative in a responsive  
design project?



# Mini Exam Review

Mini Exam Review



# HTML Layout, Flexible Grids and Media Queries





# Layouts in HTML/CSS (Flexbox and Floats)



The boxes on screen can  
be positioned next to one  
another in two main ways:  
**floats and flexbox**



A 3rd way, **CSS Grid**, is now available and is great for multidimensional (think row+column) layouts



# Floats

By using the “float: value” property in CSS, you can force elements to float over to the left or the right beside another element. Think of a floated image within text.



Lorem  
Ipsum

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nullam sodales pretium ipsum. Etiam ut enim augue. Etiam mi tortor, pulvinar at dictum faucibus, mollis eget nunc.

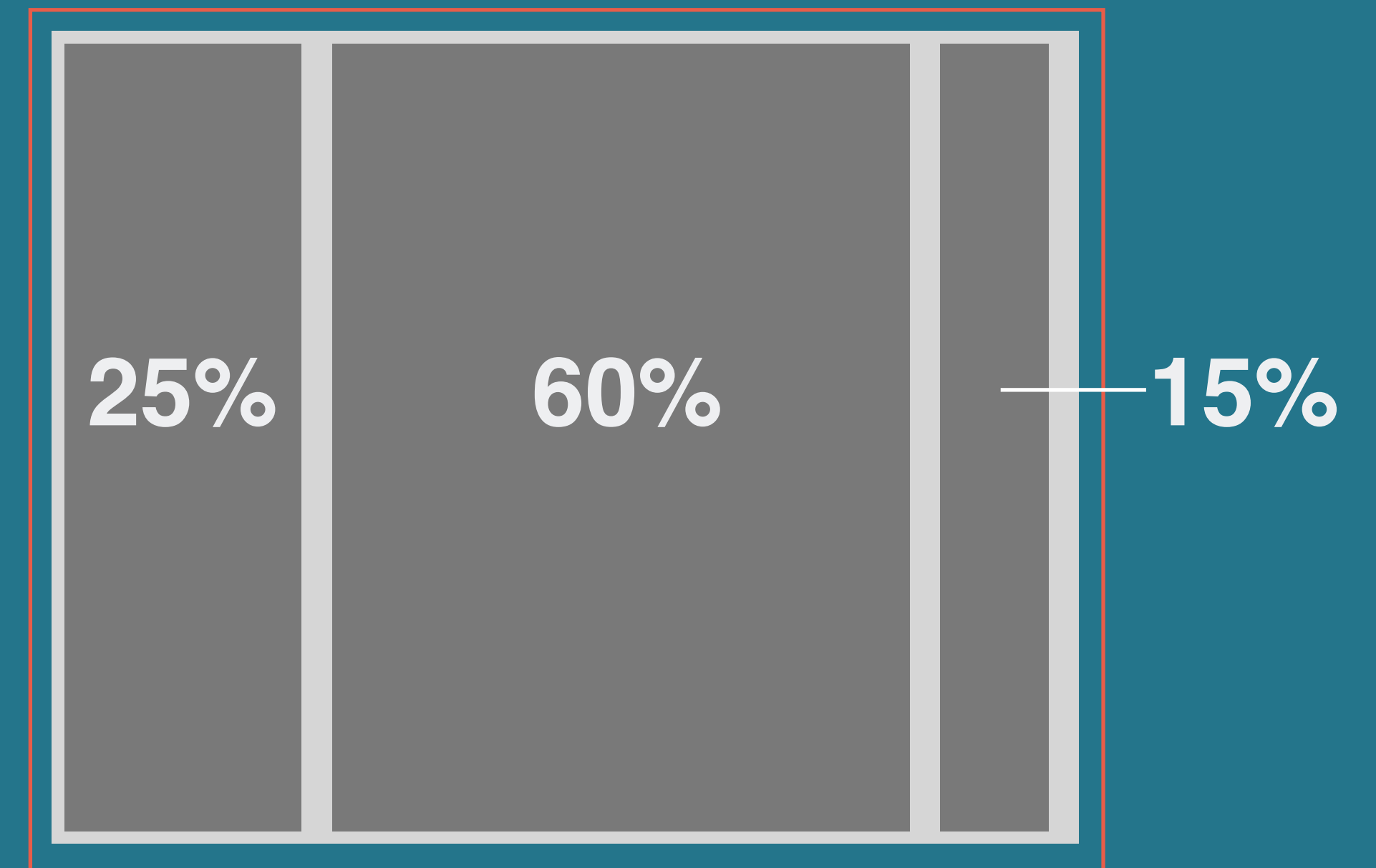
Morbi justo velit, rutrum vel placerat in, adipiscing vitae sapien. Vivamus sit amet lacus enim, vel scelerisque justo. Nunc dignissim vehicula urna ac porta. Suspendisse potenti.

Praesent nisi dolor, ornare quis sodales id, lacinia ut mauris.

```
img {  
  float: left;  
  width: 50px;  
}
```

# Floats

By using this property, we can create complex float-based column layouts, each column at various sizes.



# Floats

```
.section1 {  
  float: left;  
  width: 25%;  
}
```

```
.section2 {  
  float: left;  
  width: 50%;  
}
```

```
.section3 {  
  float: left;  
  width: 15%;  
}
```



Problem: Floated columns  
don't know when to end  
("clear") and have no defined  
height so we must use a  
clearfix.



# Pros of Float Layouts

Tried and true - used on most websites today

Works in old browsers





# Cons of Float Layouts

Requires hacks for simple layout functionality (clearfix, equal height columns, etc.)

Much more code to accomplish than flexbox.



# So What's Flexbox?



Allows you to change the size of boxes within a container to create advanced row or column layouts.



# Pros of Flexbox

Allows for advanced flexible layouts

Can change direction/order

Vertical and horizontal centering built in

Allows for equal height columns

Doesn't need "clearfix"



# Cons of Flexbox

Incompatible with old browsers (requires fallback using floats - IE11 and up)

Newer standard, so sites developed years ago may not use it.



# How Does Flexbox Work?



Start by telling a “container” box element to display: flex. This will make all of its children display as flexible items.



Section 1

Section 2

Section 3







```
main {  
  display: flex;  
}
```



[http://codepen.io/challahan/  
professor/YGgNxR/](http://codepen.io/challahan/pen/YGgNxR/)



# When Can I Use Flexbox? TODAY!



# Can I use flexbox

? ⚙ Settings


1 result found

## Flexible Box Layout Module - CR

Method of positioning elements in horizontal or vertical stacks.  
Support includes the support for the all properties prefixed with `flex` as well as `display: flex`, `display: inline-flex`, `align-content`, `align-items`, `align-self`, `justify-content` and `order`.

U.S.A.	85.34%	+	11.92%	=	97.26%
unprefixed:	83.82%	+	9.31%	=	93.14%
Global	82.33%	+	14.72%	=	97.05%
unprefixed:	81.46%	+	4.44%	=	85.91%

Current aligned Usage relative Date relative Show all

IE	Edge <sup>*</sup>	Firefox	Chrome	Safari	Opera	iOS Safari <sup>*</sup>	Opera Mini <sup>*</sup>	Android Browser <sup>*</sup>	Chrome for Android
			49						
			51			8.4 			
		47	52			9.2			
8	13	48	53	9.1		9.3			
<sup>4</sup> 11	14	49	54	10	40	10	all	53	53
		50	55	TP	41				
		51	56		42				
		52	57						



<https://css-tricks.com/snippets/css/a-guide-to-flexbox/>



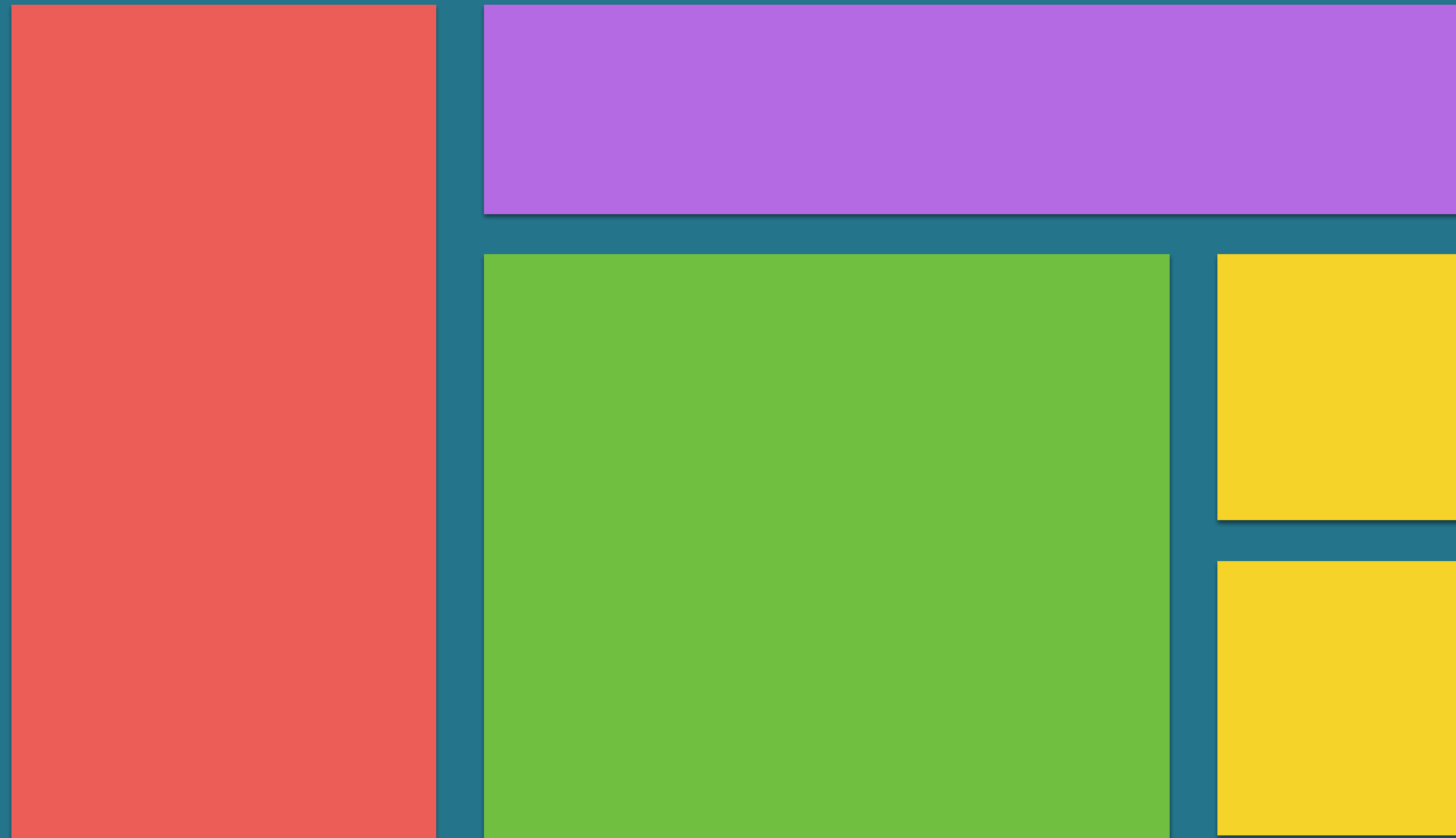
# Single Dimension



# CSS Grid



# Multi-Dimension





**Better support for 2 dimensional  
grids**

**Deals with things like gutters and  
uneven rows better.**



# Boxes & Layout Assignment



# For Next Week

DUE: Boxes and Layout

MINI EXAM: In class, 30 min.

READ: Bringing Design and Research Closer Together article

READ: CSS Grid Article

