CSS Images & Transitions

CISC-2350-R01 | Fall 2017 | Week 10-2

Ruta Kruliauskaite

Today's Agenda

- Attendance

- Review: CSS grids

- Homework Show & Tell (grids)

- Styling images

- CSS Transitions

- Homework Assignment

Grids

Composition in any visual art is the placement or arrangement of visual elements — how they are organized on a page.

guardian.co.uk

News   Sport   Comment   Culture   Business   Money   Life & style   Travel   Environment   TV   Blogs   Data   Mobile   Offers   Jobs

Culture ▸ Art and design   Books   Film   Music   Stage   Cuts blog   Kids' books   TV & radio   Games   Blogs   Podcasts   Video

culture

🔊 Webfeed

Editors' picks

Cannes 2011

Film Weekly from Cannes

Podcast: Jason Solomons meets Cannes festival president Gilles Jacob, Palme d'Or juror Mahamat-Saleh Haroun, electro-duo Air and our own Joe Cornish

📷 Post your comment

📹 Video: Space milkshake anyone?

○ Tilda Swinton: "It's bloody having a child"

○ Diary: Will I ever leave Hotel Cannes-ifornia?

○ **Full coverage: Cannes 2011**

Music

Come to our new band show!

Rumer, Wretch 32 and Summer Camp share the bill at New band of the day night for Guardian and Observer readers

○ Alexis Petridis reviews Kate Bush's new album

○ Rihanna and the rise of raunch pop

○ News: Fleetwood Mac to reunite in 2012

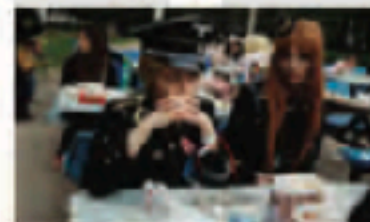○ Alexis Petridis interviews Donovan

---

Books

Sci-fi is no place for dabblers

'Write about what you love, rather than what you have contempt for', says Iain M Banks

💬 165 comments

Photography

Ng Han Guan

In pictures: Ng Han Guan is a Singaporean photographer based in Beijing. In this set of photographs he captures various facets of daily life

Stage

Pinter in a cell

Video: Can you make five plays by Harold Pinter into a piece of immersive theatre? Andrew Dickson found out

TV and radio

Last night's TV

Review: Inside the Human Body did more to make death less scary than any theologian

💬 21 comments

○ Your next box set:

Live webchat
An agent talks

Children's books
Mini heroes

On this site

Art and design

Art

Architecture

Design

Photography

Your photographs

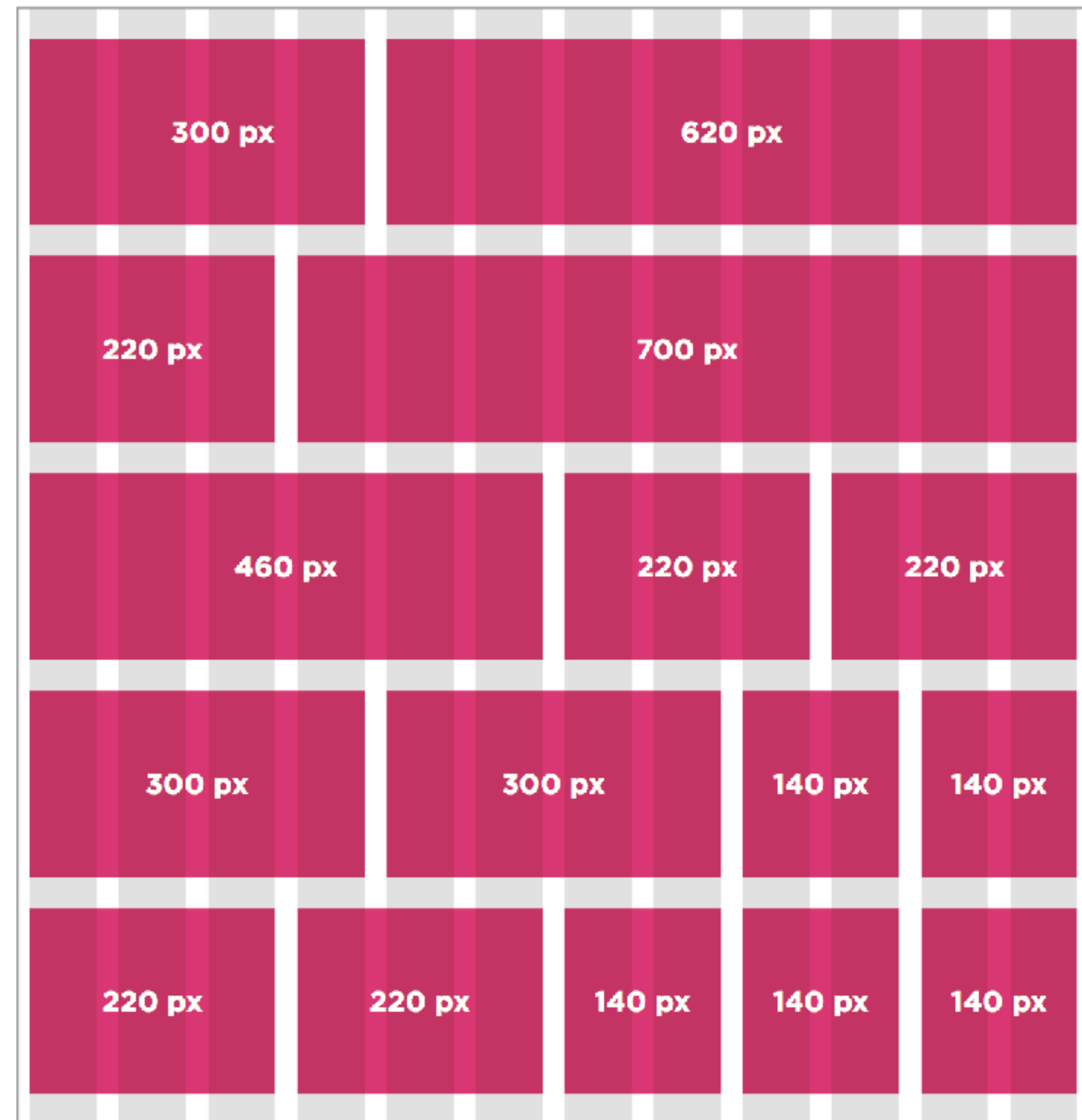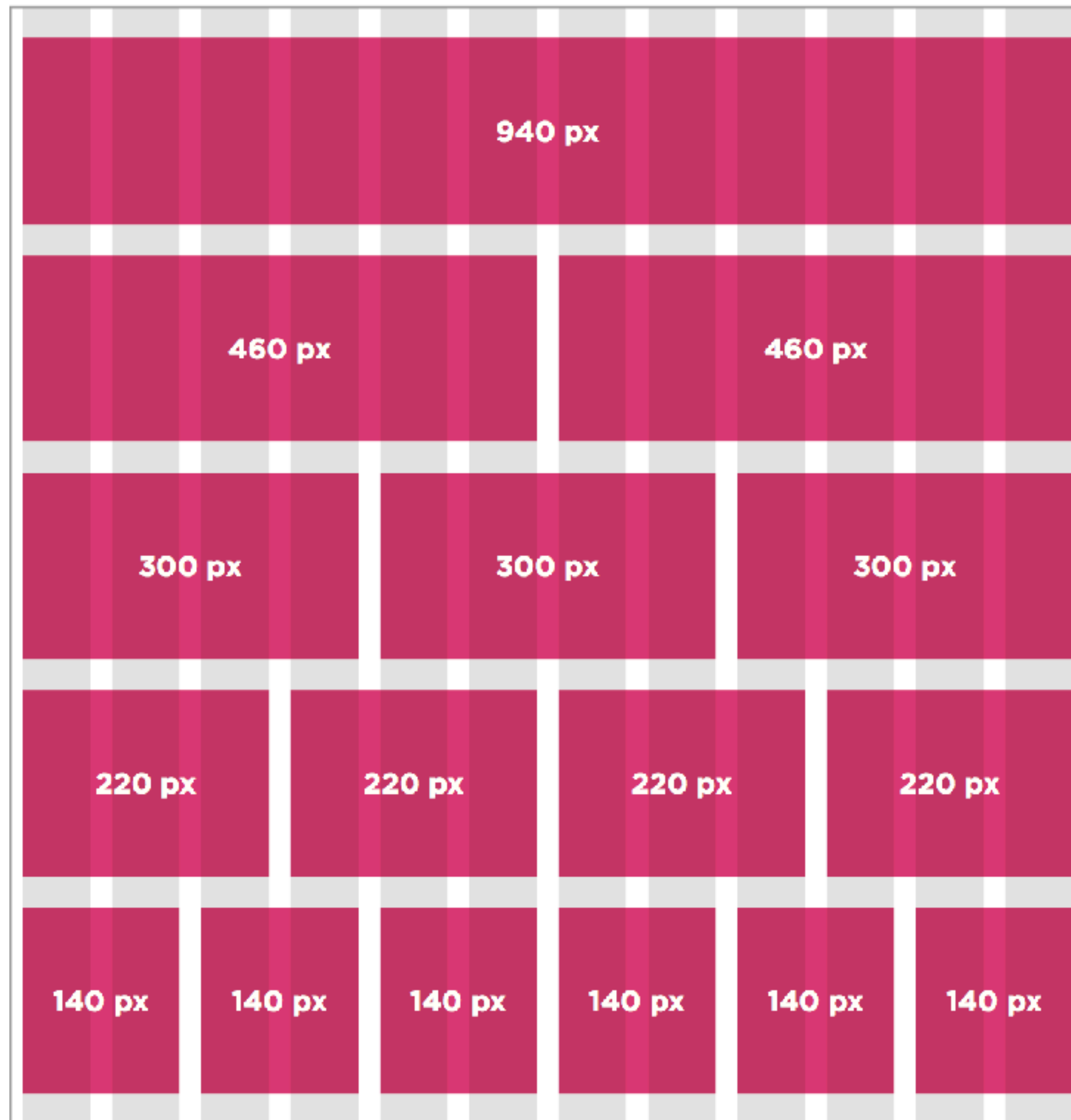Jonathan Jones on art

Books

News

Reviews

Books genres

Books blog

Film

Grids

- Creates a continuity between different pages which may use different designs
- Helps users predict where to find information on various pages
- Makes it easier to add new content to the site in a consistent way
- Helps people collaborate on the design of a site in a consistent way

960 pixel wide 12 column grid

940 px

460 px   **460 px**

300 px   **300 px**   **300 px**

220 px   **220 px**   **220 px**   **220 px**

140 px   **140 px**   **140 px**   **140 px**   **140 px**   **140 px**

300 px   **620 px**

220 px   **700 px**

460 px   **220 px**   **220 px**

300 px   **300 px**   **140 px**   **140 px**

220 px   **220 px**   **140 px**   **140 px**   **140 px**

We can build grids from scratch but there are also a lot of CSS frameworks that we can use.

CSS Frameworks

- Frameworks are essentially another stylesheet we add to our code (like our style.css)
- They aim to make common tasks in CSS easier
- It's a bunch of classes that style content for us and indicate to the HTML how wide each section should be

960gs classes

container_12 -> Acts as a container for the whole page

grid_12 ->Create a block element that's 12 columns wide

grid_4 -> Create a block element that's 4 columns wide

...And every number between 1 & 12

Questions?

Homework show & tell

http://bit.ly/2xzLHvt

Some things to keep in mind about CSS styling...

CSS Styling

- You can adjust almost anything to any element in your HTML, so describing them all is going to take forever
- It's really going to start being case-by-case basis for everything you want to do.
  - Google what you want to do! (i.e. "solid border buttons css")
  - Check http://www.w3schools.com/
  - Check Mozilla documentation
- And for more advanced things (i.e. "How can I put an overlay over an image css"), google it, and you'll see many threads from places like Stack Exchange and Stack Overflow.
- Notice I'm always adding 'css' in my google searches, to specify it's for that and not photoshop or another coding language

Images

Controlling the size and alignment of images affect the presentation of your page.

CSS Images

- Specifying the size and alignment of an image we're using
- Adding background images to boxes

1. Image sizes

Changing image size

- Use *width* and *height* properties to control the size of an image
- Specifying image sizes will help your page to load more smoothly
  - telling the browser how much space to leave for an image will allow for the rest of the page to be rendered without waiting for the image to download
- For consistency you should use same sized images across of your pages:
  - Small portrait: 220 x 360
  - Small landscape: 330 x 210
  - Feature photo: 620 x 400

Changing image size

- A convention for sizing images, since most websites use only a couple of image sizes, is to have a couple of classes dedicated to each size of image. Remember the images still need to be in the right aspect ratio.

```
img.small {
    width: 80px;
     height: 60px;
 }
img.medium {
    width: 160px;
    height: 120px;
}
img.large{
    width: 240px;
    height: 180px;
    }
```

**Remember the "." signifies a class, which can be used on many different elements

1. Image align

Aligning images

- align vs. float
- You can use *float* in two ways:
  1. The *float* property is added to the class that was created to represent the size of the image (e.g. small)
  2. New classes are created with names such as `align-left` or `align-right` to align the images to the left or right of the page. These class names are used in addition to classes that indicate the size of the image.

```
img.align-left{
 float: left;
 margin-right: 10px;
}
img.align-right{
 float: right;
 margin-left: 10px;
}
```

HTML:
```
<p><img src="images/pic.jpg" alt="Picture"
class="align-right medium"/></p>
```

Centering images

- Remember images are **inline**, so they don't default to a new line like p or h1 or others (they float together with text)
- To center an image by itself on the page, we need to make it a block element instead of an inline element, and then use margin: auto to center it.

```
img.center{
 display: block;
 margin: 0px auto;
}
```

HTML:
```
<p><img src="images/pic.jpg" alt="Picture"
class="center medium"/></p>
```

3. Background images

background-image

- Allows you to place an image behind any HTML element
- Could be entire page or a specific HTML element (<div>, <p>, etc.)
- By default, a background image will repeat to fill the entire box.
- Background images are often the last thing on the page to load (which can make a website seem slow to load).

```
p{
 background-image: url("images/pic.jpg");
}
```

Example 1
Example 2

background-repeat

- Can have four values:
  - *repeat:* the background image is repeated both horizontally and vertically (the default way it is shown if the background-repeat isn't used)
  - *repeat-x:* the image is repeated horizontally only
  - *repeat-y:* the image is repeated vertically only
  - *no-repeat:* the image is shown only once

```
p{
background-image: url("images/pic.jpg");
background-repeat: repeat-x;
}
```

Example

background-attachment

- Specifies whether a background image should stay in one position or move as the user scrolls up and down the page. Can have two values:
  - *fixed:* the background image stays in the same position on the page
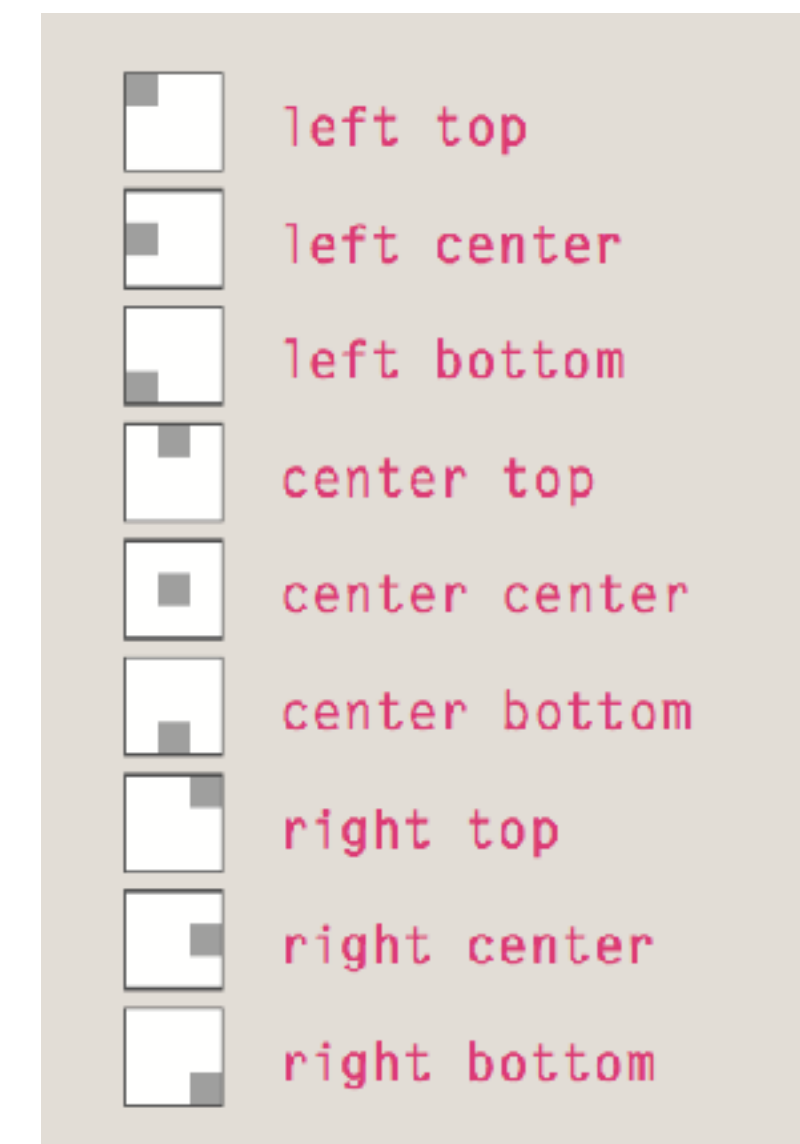 - *scroll:* the background image moves up and down as the user scrolls up and down the page

```
p{
 background-image: url("images/pic.jpg");
 background-repeat: repeat-x;
 background-attachment: scroll;
}
```

Example

background-position

- When an image is not being repeated, you can use the *background-position* property to specify where in the browser window the background image should be placed.
- This property usually has a pair of values. The first represents the horizontal position and the second represents the vertical.

```
p{
 background-image: url("images/pic.jpg");
 background-repeat: no-repeat;
 background-position: center top;
}
```

left top

left center

left bottom

center top

center center

center bottom

right top

right center

right bottom

Example

background-position

- If you only specify one value, the second value will default to center.
- You can also use a pair of pixels or percentages. These represent the distance from the top left corner of the browser window (or containing box). The top left corner is equal to 0% 0%.

```
p{
 background-image: url("images/pic.jpg");
 background-repeat: no-repeat;
 background-position: 50% 50%;
}
```

background

- Acts like a shorthand for all of the other background properties and also the background-color property.
- The properties must be specified in the following order, but you can miss any value if you do not want to specify it:
    1. background-color
    2. background-image
    3. background-repeat
    4. background-attachment
    5. background-position

```
p{
 background: #efefef url("images/pic.jpg") no-repeat top right;
}
```

background

- CSS3 will also support the use of multiple background images by repeating the *background* shorthand:

```
p{
 background:
  url("images/pic.jpg")
  no-repeat top left,
  url("images/pic2.jpg"
  no-repeat bottom left,
  url("images/pic3.jpg")
  no-repeat centre top;
}
```

4. Button manipulation

hover state buttons / links

- Button/link states are a pretty common thing
- Links default to a blue underlined text, and the mouse goes to the pointer when we hover, but we can specify in CSS if we want something different
- These are the states, where button link is followed by ":":
    - :hover
    - :visited
    - :active

```
a{
 color: red;
 text-decoration: none;   <--**This is stopping it from the default underline on a link
}

a:hover {
 color: blue;
 text-decoration: underline;
}
```

Example

Questions?

CSS images and links resources

images
https://www.w3schools.com/css/css3_images.asp
https://www.godaddy.com/garage/6-simple-css-tricks-for-images/
https://css-tricks.com/responsive-images-css/

background-image
https://www.w3schools.com/CSSref/pr_background-image.asp
https://css-tricks.com/almanac/properties/b/background-image/

links
https://www.w3schools.com/css/css_link.asp

Last note on layout...

Historically the way developers split up elements

- Utilizing div elements with ID and classes
- A lot of ID and class elements to keep track of

Newer approach

- HTML5 has more element names so developers don't have as many div elements and it's easier for them to navigate
- It's also more effective for screen readers that can sort importance of page content
- Feel free to use either, but new element names are more made for accessibility


```
<body>
    <div id="page">
        <header>

            <nav>
        </header>

        <div id="content">                    <aside>
            <article>

            </article>

            <article>

            </article>
        </div>

        <footer>
    </div>
</body>
```
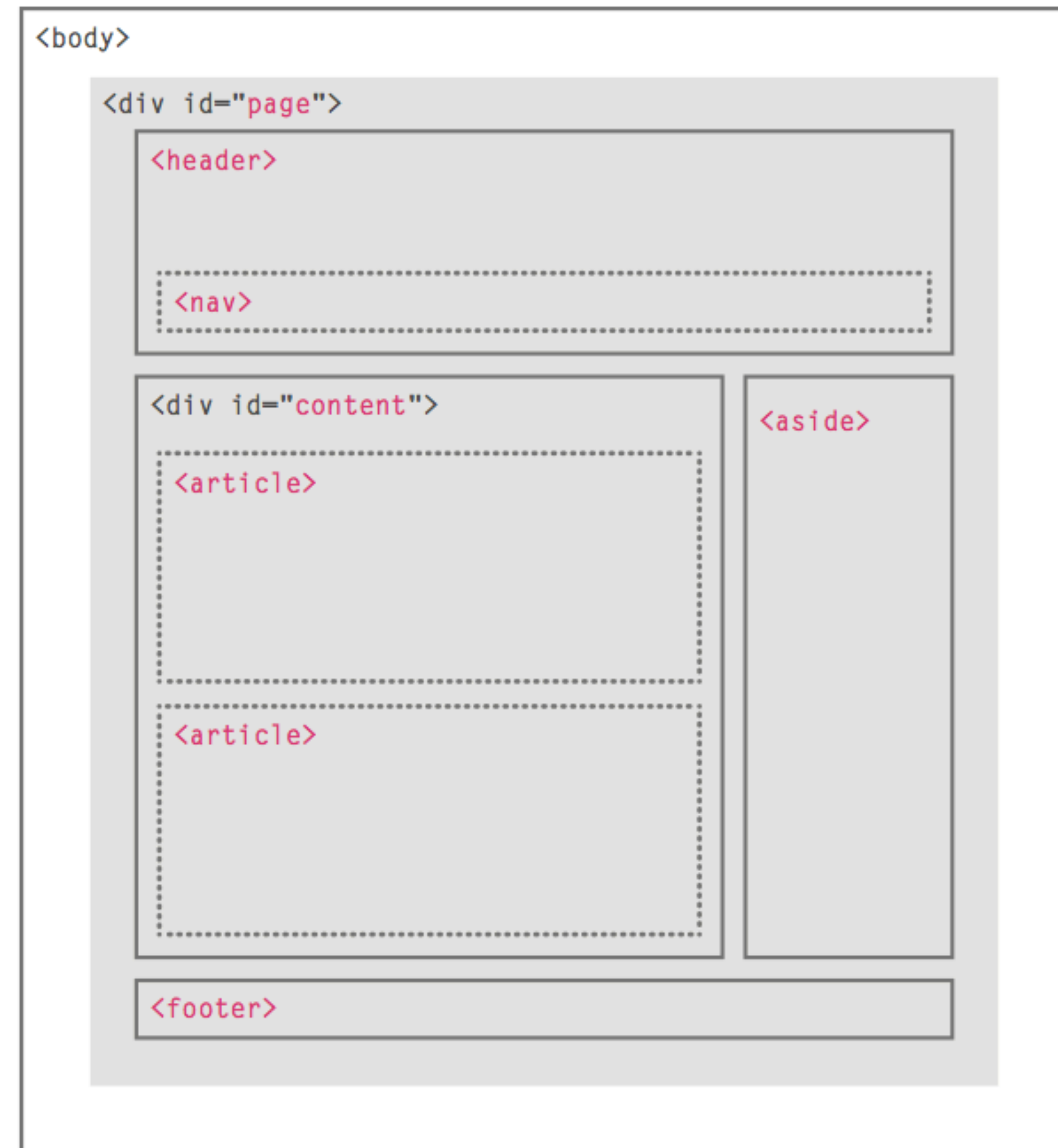
CSS Transitions

CSS Transitions & Animations

- CSS is getting better at making transitions and animations (something that used to be reserved for JavaScript)
- We can't get too advanced, but we can add an element of style to our pages
- We can make elements fade in/out, animate position and more

CSS Transitions

- Allows you to change property values smoothly over a given duration
- To create a transition effect, you must specify two things:
    - the CSS property you want to add an effect to
    - the duration of the effect
- If the duration part is not specified, the transition will have no effect, because the default value is 0.

```
#transition {
    width: 100px;
    height: 100px;
    background: red;
    -webkit-transition: width 2s, height 4s; /* For Safari 3.1 to 6.0 */
    -moz-transition: width 2s, height 4s;
    transition: width 2s, height: 4s;
}
#transition:hover {
    width: 300px;
    height: 300px;
}
```

Example

Transition timing

- You can change how the speed of transitions happens:
- Use transition-timing-function selector to make that happen:
  - ease: slow start, then fast, then end slowly (default)
  - linear: the same speed from start to end
  - ease-in: a slow start
  - ease-out: a slow end
  - ease-in-out: a slow start and end

```
{transition-timing-function: linear;}

{transition-timing-function: ease;}

{transition-timing-function: ease-in;}

{transition-timing-function: ease-out;}

{transition-timing-function: ease-in-out;}
```

Example

Transition delay

- transition-delay: specifies a delay (in sec) for the transition effect

```
#transition {
    width: 100px;
    height: 100px;
    background: red;
     -webkit-transition: width 2s, height 4s; /* For Safari 3.1 to
 6.0 */
    -moz-transition: width 2s, height 4s;
    transition: width 2s, height: 4s;
    transition-delay: 2s;
}
#transition:hover {
    width: 300px;
    height: 300px;
}
```

Example

Transition transformation

- Specifies how the position of the element should change (e.g. rotate)

```
#transition {
    width: 100px;
    height: 100px;
    background: red;
      -webkit-transition: width 2s, height 4s; /* For Safari 3.1 to 6.0 */
    -moz-transition: width 2s, height 4s;
    transition: width 2s, height: 4s;
    transition-delay: 2s;
    transition-transform: 2s;
}
#transition:hover {
    width: 300px;
    height: 300px;
    transform: rotate(180deg);
}
```

Example

Fade In and Out

- To use transition effects to fade something in or out, we can play with opacity.

```css
#transition {
    opacity: 0.0;
    background: red;
    -webkit-transition: opacity 0.5s ease-in-out; /* For
 Safari 3.1 to 6.0 */
    -moz-transition: opacity 0.5s ease-in-out;
    transition: opacity 0.5s ease-in-out;}
#transition:hover {
    opacity: 1.0;
}
```

Example

Image hover example

Scrolling div example

Scrolling div 2 example

CSS transitions resources

transitions

https://www.w3schools.com/css/css3_transitions.asp

https://www.w3schools.com/cssref/css3_pr_transition.asp

https://www.w3schools.com/cssref/css3_pr_transition-delay.asp

https://www.w3schools.com/cssref/css3_pr_transition-duration.asp

https://www.w3schools.com/cssref/css3_pr_transition-property.asp

https://www.w3schools.com/cssref/css3_pr_transition-timing-function.asp

transform

https://www.w3schools.com/cssref/css3_pr_transform.asp

https://www.w3schools.com/css/css3_2dtransforms.asp

https://www.w3schools.com/css/css3_3dtransforms.asp

filter

https://www.w3schools.com/cssref/css3_pr_filter.asp

Homework assignment

Homework

By Sunday, November 5, 6pm

1. **Web review presentations: Michaela, Jake**
   1. Post the website you will present to #general channel on Slack by Sunday 6pm
2. **Read and review the following CSS documentation on images and transitions:**
   - All links given in class slides above
3. **Choose a CSS trick on background-image or transition that we haven't spoken about in class and make a single HTML page that includes:**
   1. CSS trick from this page: https://css-tricks.com/almanac/properties/
   2. Explanation on what the trick is about (paragraph or two)
   3. Demo of the trick / effect itself (working example)
   4. Use other CSS properties, so it looks pretty
   5. Upload your code to a new folder on Github and post two links on #general channel on Slack:
      - Link to your project's repository (folder)
      - Working link to the Github page (yourusername.github.io/pathtofolder)