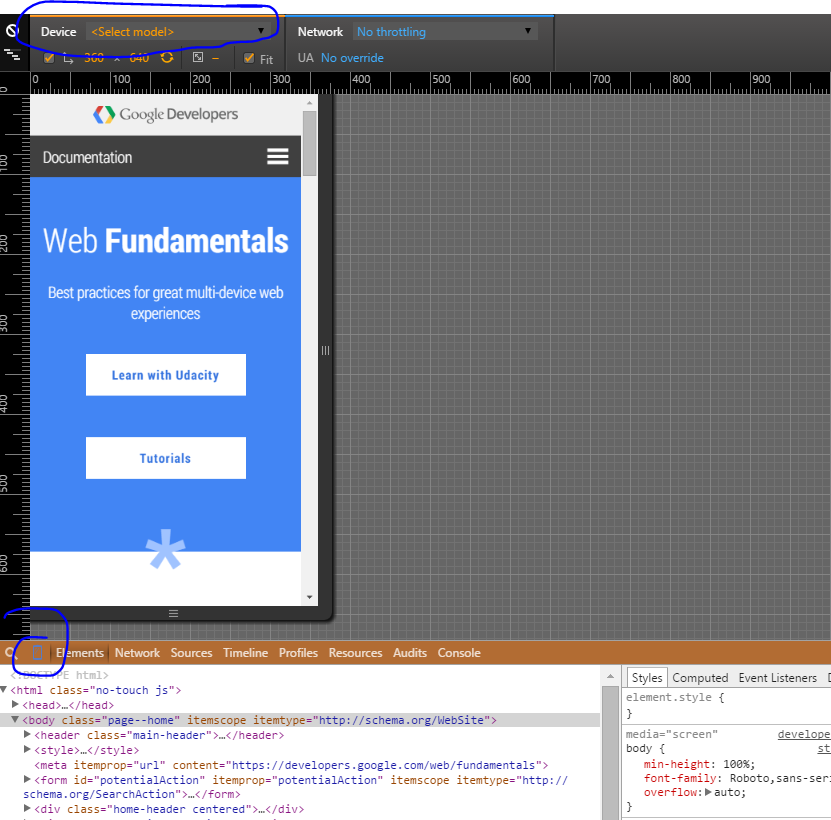
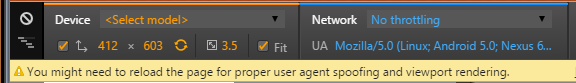
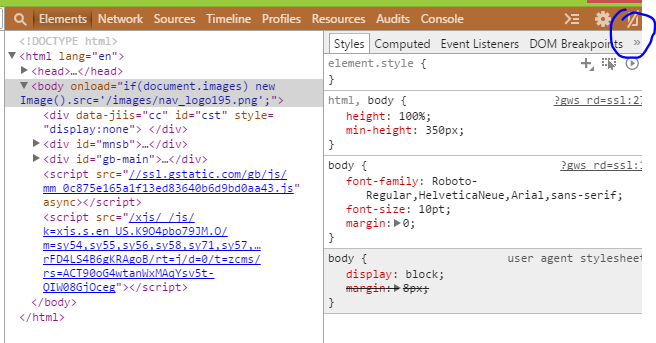
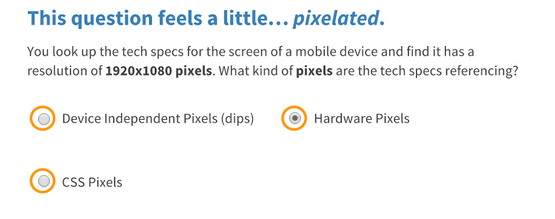
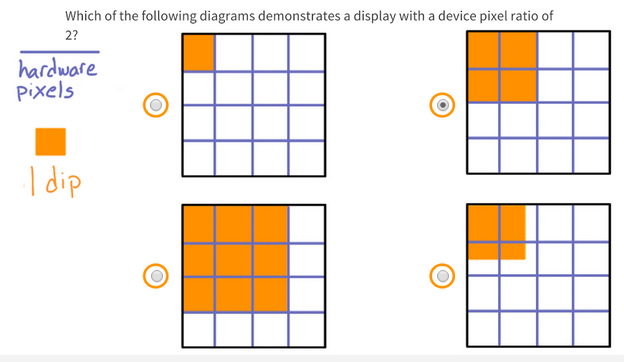
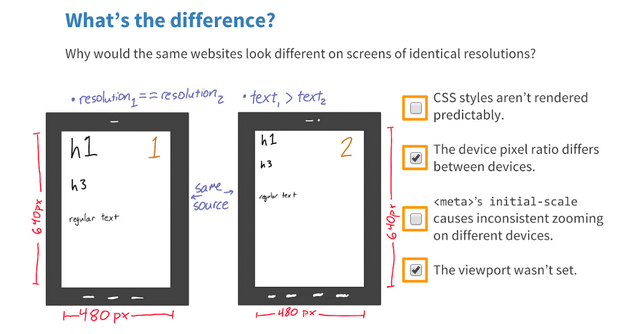
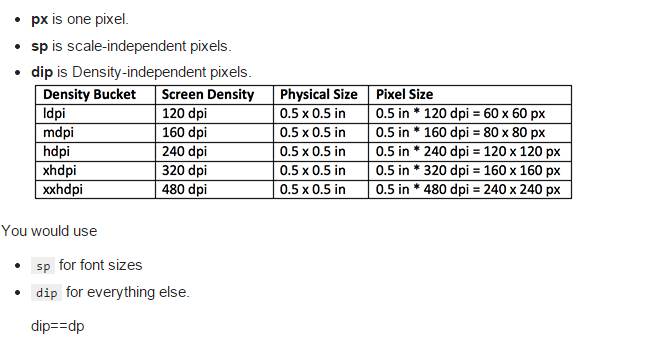
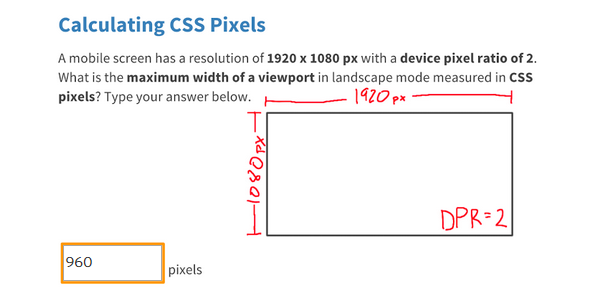
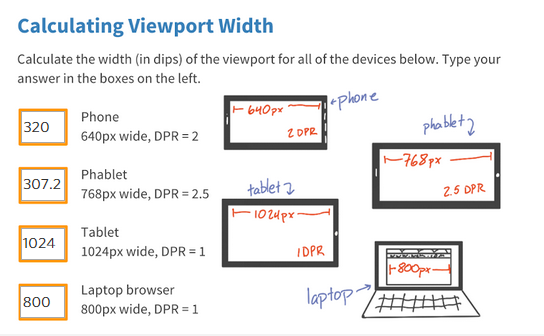
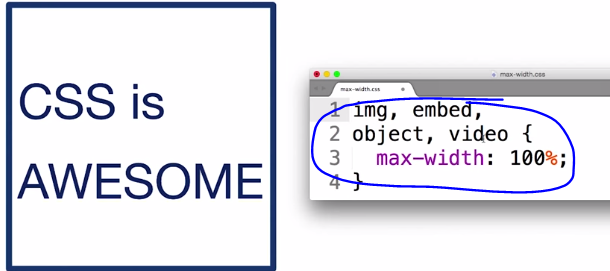
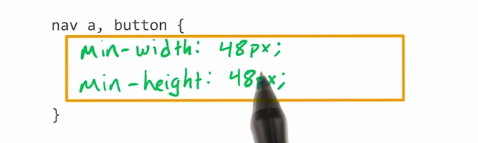
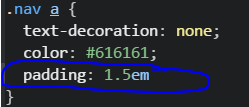
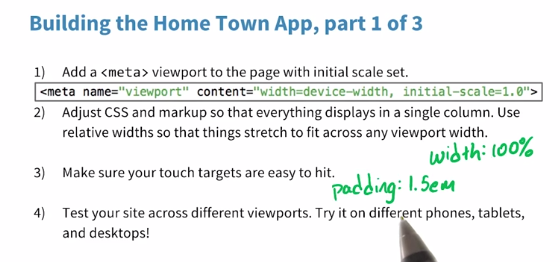
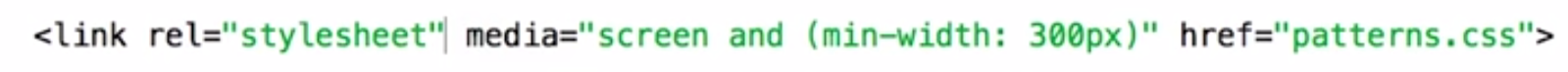
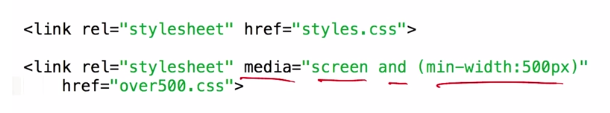
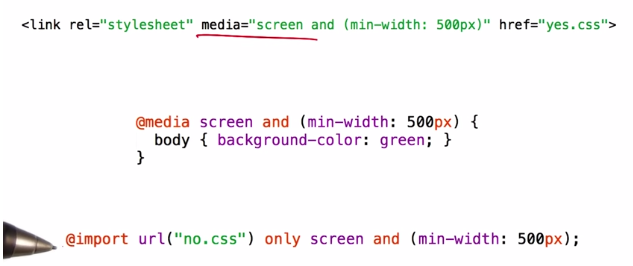
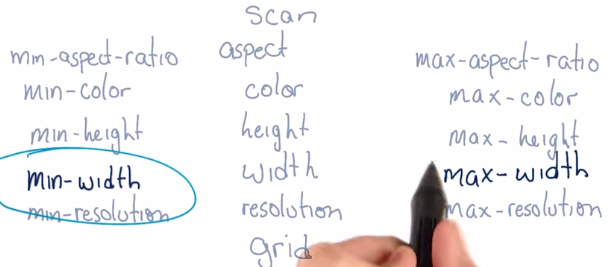
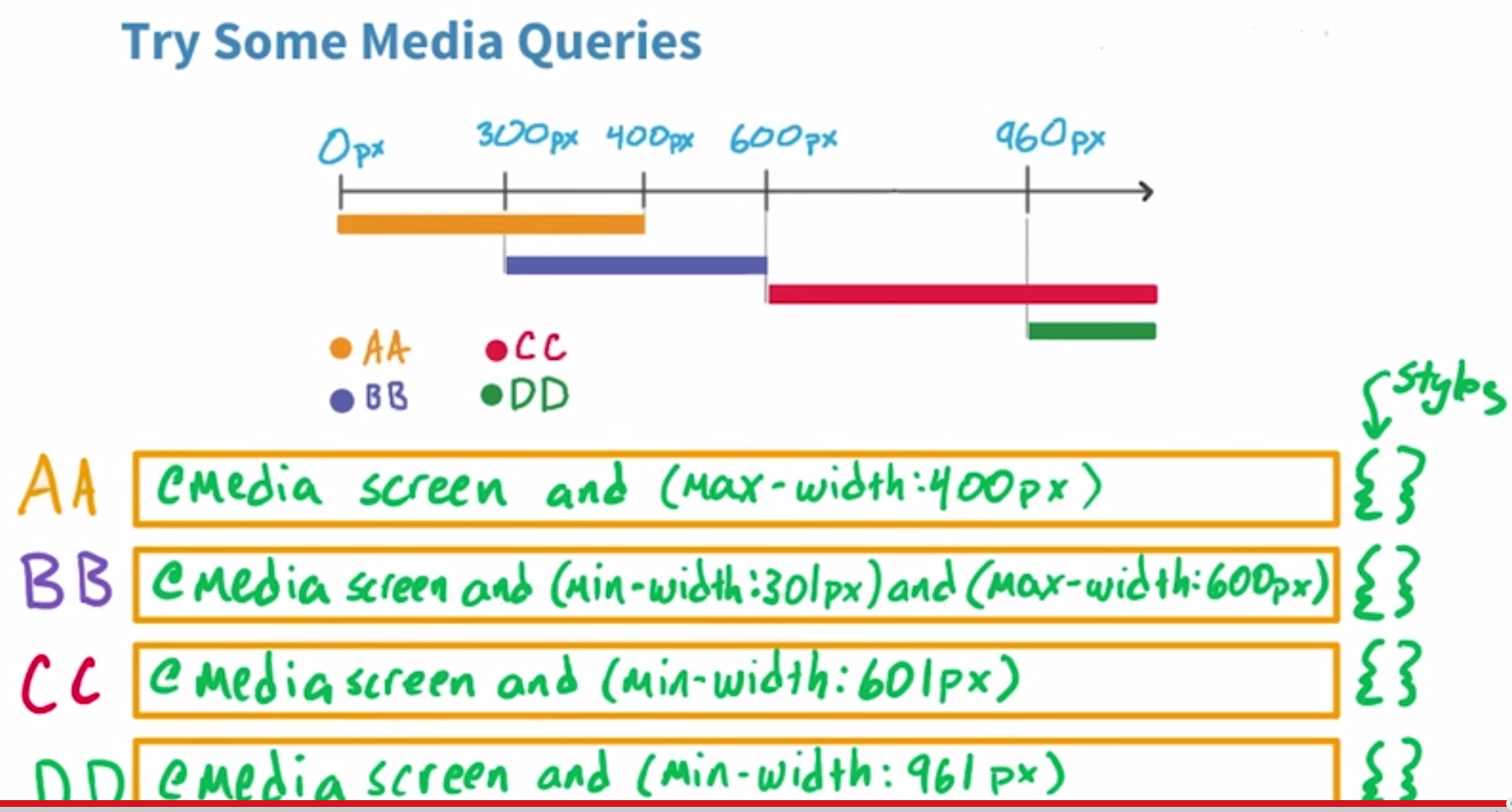
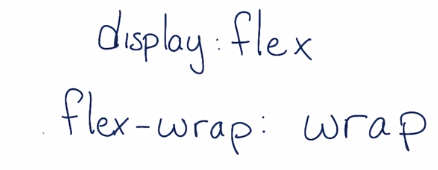
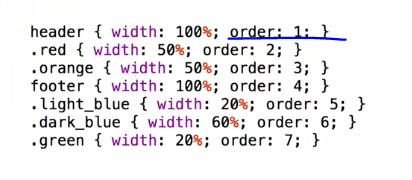
# Why Responsive

1. Motivation behind responsive design:
   1. Make websites to work for all devices
2. Devices emulator for web developers:
   1. 
3. 
   1. UA: user agent
4. Use dev tools on mobile:
   1. [Download and install Chrome Canary](http://www.google.com/intl/en/chrome/browser/canary.html)
   2. Enable debugging mode in your mobile phone
   3. Go to canary and type chrome://inspect
   4. You should see the device
   5. You can inspect webpages running mobile devices through your desktop
   6. You can also show the mobile view on your desktop:
      1. 

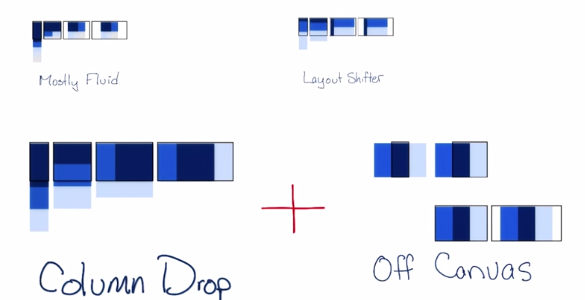
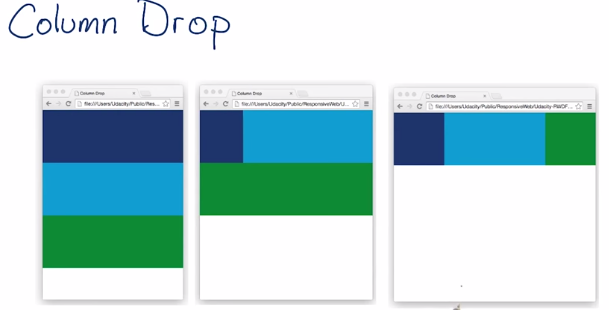
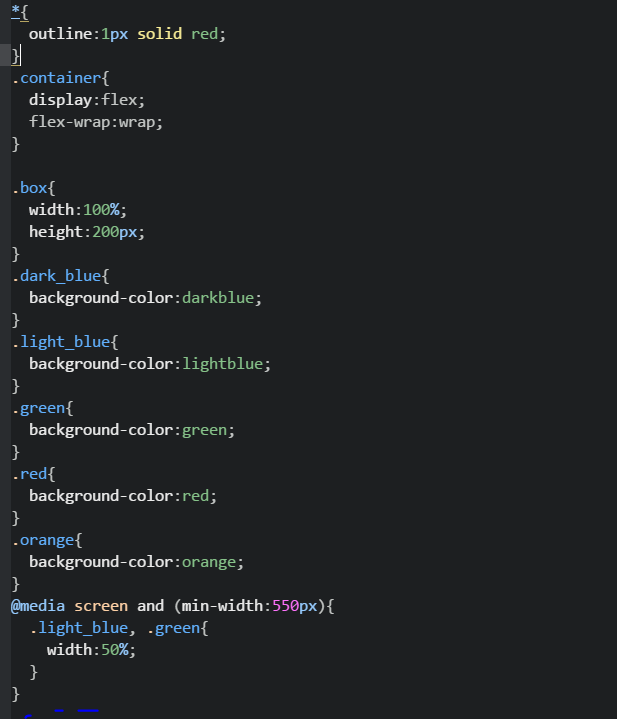
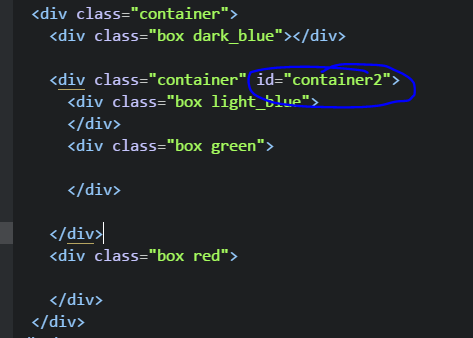
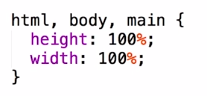
# Defining the Viewport

1. Viewport defines the area of screen the browser can render contents to.
   1. Pixel density
   2. Dip:
      1. Device independent pixel, relate pixel to distance,
2. No meta viewport:
   1. It assumes that the browser render the webpage on 980dip
3. Css pixels = dip
4. Pixels-related questions:
   1. 
   2. Device ratio pixel = pixel/dip
      1. 
   3. 
   4. 
   5. 
      1. Css pixels = dip
   6. 
5. Viewport are defined by dip
6. Add meta viewport:
   1. 
      1. Device-wdith unit is dip
      2. Initial-scale = 1🡺 1 css pixel = 1 dip
7. Relative position should be used
   1. 
   2. Good practice to make sure the content will flow the container
8. Buttons:
   1. 48px x 48px
   2. 
9. Start small for design
   1. Starting from small 🡺 prioritize content
   2. Performance
10. 
    1. 1.5em means that the padding is the 1.5 size of the font
11. Build a webpage for mobile:
    1. 

# Building up

1. A responsive site change for different devices
2. Selective CSS
   1. Media queries
   2. 
   3. 
      1. Just stick with screen and print
3. Different ways to use media queries:
   1. 
      1. Avoid import
   2. Link vs @media:
      1. Link: has more css file and more requests
      2. @media has bigger files
      3. 
4. Next step media queries:
   1. 
5. Add media queries inside the html file:
   1. 
6. Breakpoints:
   1. Where do I put the breakpoints
      1. Look at the contents to find breakpoints
   2. Content-centric
7. Pixels:
   1. 
8. Breakpoint and media queries give us a lot of control:
   1. Patterns: gird-fluid system:
9. Flexbox is one of the most important tools:
   1. 
   2. Order add the order attribute to flex box to help the flex box:
      1. 

# Responsive Web Deisgn Fundamentals

1. Common patterns:
   1. 
   2. Using flex box + media queries
2. Column drop:
   1. 
      1. Three break points
      2. After the last breakpoint, the margins are applied
      3. 
3. Mostly fluid:
   1. 
   2. It is important to specify both height and weigth
   3. Code mostly fluid layout:
      1. 
      2. 
4. Layout shifter:
   1. Use order attribute to reorder the
      1. Order: -1 will appear first
   2. Change an individual instance properties of an class:
      1. 
      2. 
5. Off canvas:
   1. Place less important information off sceen
   2. 
   3. 
6. HTML semantics elements:
   1. 

# Optimizations

1. Images:
   1. Let the browser chose the best images to fit different resolutions
2. Responsive image – new course:
3. Tables
   1. Hidden columns
   2. No more columns
   3. Contained tables
      1. Easiest one
4. Line length is important to building websites
   1. Font-size:16px
   2. Line-height; 1.2em
      1. For text heavy site:
         1. 18px
         2. 1.25em
5. Major breakpoints + minor breakpoints
6. Major breakpoints: change layout
7. Minor breakpoints:
   1. Font size
   2. Padding
   3. Margin
   4. Etc
8. Breakpoints + media queries