**Highlights**

* jQuery Page 1
* Node.js and NPM Page 4
* CSS Preprocessors Page 5
  + Less
  + Sass
* Bower Page 8

**jQuery**

***jQuery*** is a lightweight JavaScript library. Bootstrap JS components are built upon jQuery

* Features available with jQuery that allows you to:
  + ***HTML/DOM manipulation***
  + ***CSS manipulation*** – add or remove CSS classes dynamically
  + ***HTML events*** – where some code need to executed after some event
  + ***Effects and animations***
  + ***Ajax***
* jQuery Syntax Example:
  + $(selector).action()
    - ***$***: define/access jQuery
    - ***(selectore)***: HTML element
      * ***Using an element:*** “p”, “button” etc
      * ***Using an ID:*** Ex: “#myCarousel”
      * ***Using a class:*** Ex: “.btn”, “.btn.button-default”
      * ***Attribute:*** Ex:
        + “[href]”
        + “[data-toggle= “tooltip”]”

it says that any element that has href attribute

* + - * ***Current Element:*** $(this)
    - ***action()***: action to be performed on the element
    - ***Ex***: $(“p”).hide(), $(“#mycarousel”).carousel(“pause”)
* Bootstap jQuery Example:

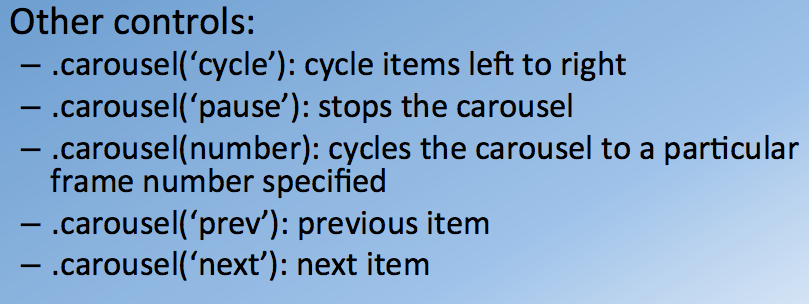
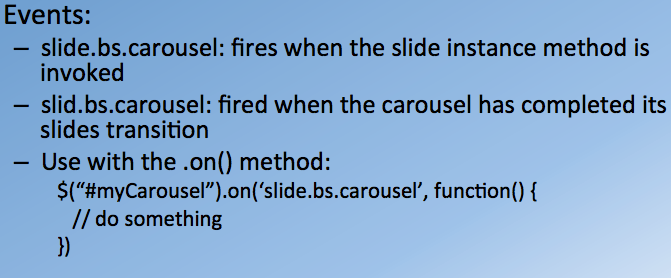
*<script>*

*$(document).ready(function() {*

*$(‘[data-toggle=“tooltip”]’)*

*});*

*</script>*

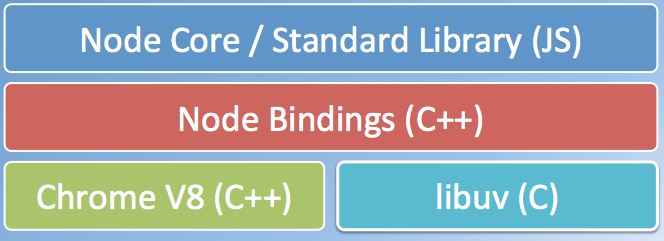
* jQuery Events
  + User’s interaction on webpage causes DOM events.
  + jQuery has event methods for DOM events. Ex: ready(), click(), mousedonw(), on(), etc.
  + Events:
    - ***Mouse:***
      * *Click*
      * *Double click*
      * *Mouse enter*
      * *Mouse leave*
    - ***Keyboard***
      * *Keypress*
      * *Keydown*
      * *Keyup*
    - ***Form***
      * *Submit*
      * *Change*
      * *Focus*
      * *Blur*
    - ***Document, window***
      * *Load*
      * Resize
      * Scroll
      * Unload
* Bootstrap ***Carousel*** example to see with jQuery
  + Data Attributes can be used with ***data*** attributes (data-\*):
    - data-slide=“prev|next”
    - data-slide-to=”0|1|2|…”
    - data-ride=“carousel”
    - data-interval=5000
  + JS based control
    - $(“.carousel”).carousel();
    - $(“.carousel”).carousel({interval: 2000});
    - Other controls controls:
      * 
  + Events
    - ***slide.bs.carousel:***  fires when the slide instance method is invoked.
    - ***slid.bs.carousel:*** fired when the carousel has completed its slid transition
    - 

**Node.js and NPM**

Historically we needed to use a completely different language of server side like ruby, python, Java, ASP.net, PHP, and so on. But with node, you can write server-side implementation in JavaScript. With a single language (JavaScript) you can write both front and back-end.

***Node.js*** is JavaScript runtime build on Chrome V8 JavaScript engine. The Chrome V8 JavaScript Engine was built originally to a part of the Chrome browser and to support the JavaScript that runs within your browser. Once you separate the engine out of the browser and let it run on your machine, you're able to support the running of JavaScript programs on your machine.

Node environment uses an event driven, non-blocking, I/O model which makes it very lightweight and efficient to handle primarly data driven websites



* Node.js Use Cases
  + There are a lot of UI utilities written in JavaScript for web development
    - Bower
    - Grunt
    - Gulp, etc.
  + Server-side Development
    - Web Server, Business, logic, Database access

***NPM*** – Node package manager manages ecosystem of node modules / packages.

* There are very useful libraries that can be used with node. These in the packages in the form of node modules
* A package contains:
  + JS files
    - Implemented libraries
  + package.json (manifest)

**CSS Preprocessors**

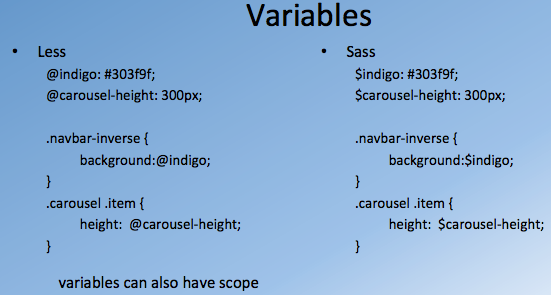
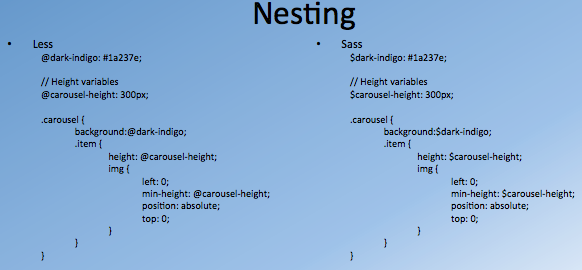
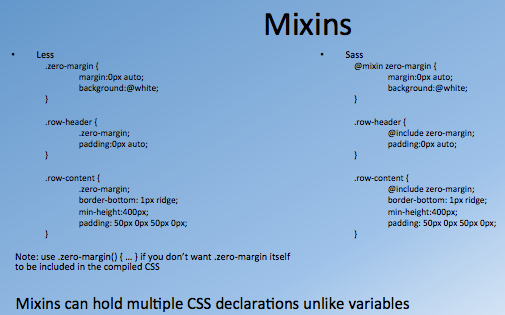
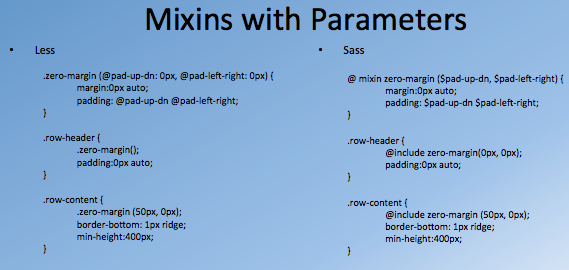
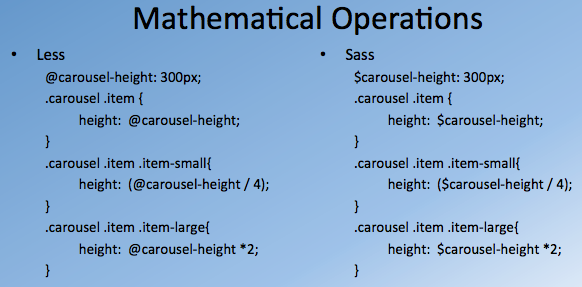
**CSS**

* great for defining styles and repeated applying them to various HTML elements.
* CSS is limited when it comes to becoming a typical programming language.
* It doesn’t support anything like ***variables, nesting, expression, and functions***
  + Without the availability of such features maintainability of CSS suffers

**CSS Preprocessors**

* Popular ones
  + Less
  + Sass (Syntactically Awesome Style Sheets)
  + Stylus
* All Preprocessors are compiled into CSS syntax before use in a webpage
  + Expectation is that written code will automatically transformed into equivalent CSS code
* Bootstrap was originally available in Less, now it available in Sass as well

**Typical CSS preprocessors features**

* Variables
  + 
  + ***Variables*** are defined in:
    - ***Sass as $***
    - ***Less as @***
  + Variables also have scopes
    - Availability of that variable is within that scope
* Nesting selectors
  + 
  + Nesting is when you have class nested in a class
  + .carusel then .carusel.item instead you can have carousel parent class then subclass going in that parent class
* Mixins
  + Sometimes you group of classes that share similar properties.
  + With standard CSS you have to repeat the properties.
  + With Sass/Less you can create a mix with group of properties as shown below with
  + 
  + You can define mixin:
    - In Less as ***.mixin\_name() { … }***
    - In Sass as ***@mixin mixin\_name { … }***
  + 
* Functions and Expressions
  + 
  + These can be used to modify the variables
* Other features
  + Function:
    - Math Operators
    - List Operators
    - String Operators
    - Color operations
      * You can take color value and darken it. You can modify hover color if needed.
    - Color blending
  + Imports
    - Have flexibility to import existing Less/Sass classes list into other Less/Sass classes respectfully

**Bower**

* Without bower there would be a lot of manual work to fetch libraries.
* There are other tools for automation in web development
  + ***Bower:*** Package Manager for the web
  + ***Grunt and Gulp:*** Task automation
  + ***Yo:*** Web app scaffolding
  + ***Yeoman:***  Workflow designed around using Yo, Bower and Grunt

***Bower:***

* Building a web app, we need to fetch a lot packages that are helpful in building our app
* Helps manage dependencies, such as:
  + Frameworks
  + Libraries
    - Bootstrap, Font Awesome, jQuery, etc.
  + Assets
  + Utilities
* Makes it really easy to fetch and update dependencies
* Installing Bower
  + Use node: ***sudo npm install –g bower***
    - –g is used for global install
  + Create bower.json file
    - It keeps tracks all the components that being used in the current project
    - Use command line ***bower init***
  + Use ***bower install library\_name –S***
    - ***library\_name*** can be replaced with bootstrap, font-awesome, etc
    - ***–S*** updates the bower.json file