February 24th, 2018 –

Angular\_js1 [codecademy]

A Javascript web framework aimed to make web apps simple to build and easy to maintain.

# First app with angular\_js

## Static Sites

### app.js

#### Module

##### Contains the different components of an AngularJS app

### index.html

#### Directive

##### Tells AngularJS where the module will live

##### ng-app

###### used to define the application scope

##### ng-controller

###### Directive that defines the controller scope

Properties attached to $scope from controller now available in the div

##### Generalization of directives

###### Bind behavior to HTML elements, AngularJS walks through each HTML element looking for directives, then triggers the behavior it finds

i.e. attach a scope or loop through an array

#### Expressions

##### Used to display values on the page

###### i.e. $scope.title (from controller)

### maincontroller.js

#### Controller

##### Manages the apps data

###### i.e. store a string (like title) and attach it to $scope

### Typical work flow

#### Create module, and use ng-app in the view to define the application scope

#### Create a controller and use ng-controller in the view to define the controller scope

#### Add data to the $scope in the controller so it can be displayed in the view

### Filters

#### Purpose

##### Format the data in the view

#### How it works

##### AngularJS get the value, sends it into the filter, the pipe -- | -- takes the output on the left and “pipes” it to the right, and the filter outputs a formatted result

##### Filters help to separate content in the controller from presentation in the view

##### [Built in filters](https://docs.angularjs.org/api/ng/filter)

### Quick Review

#### Module 🡪 contains different components of an AngularJS

#### Controller 🡪 manages the app’s data

#### Expression 🡪 displays values on the page

#### Filter 🡪 formats the value of an expression

### ng-repeats

#### Use array in the controller to store objects

#### In the view add the directive, ng-repeat=“product in products” to loop through array and displace each element

## Interactive app

### App.js

### index.html

#### ng-click

##### Notice the function doesn’t interact with the view at all; just updates the controller, and any change in controller then shows up in view.

### maincontroller.js

## Generalizations

### User visits AngularJS app

### The view presents the app’s data through the use of expressions, filters, and directives.

### User clicks an element in the view, if the element has a directive it runs the function

### The function updates the state of the data

### The view automatically changes and displays the updated data.

#### Note: The page doesn’t need to reload at any point!

# Directives

## Used app.directive to create a new directive named ‘appInfo’ has 3 parts

### Restrict: specifies how the directive will be used in the review

#### E means it will be used as a new HTML element

### Scope: specifies that we’ll pass info into this directive through an attribute. The = tells the directive to look for an attribute with same name

### templateURL specifies the HTML to use to display data in scope

## Usefulness of Directives

### Readability – expressive HTML allows you to understand app’s behavior just by reading HTML

### Reusability – don’t need to reuse code

## Built-In and Custom Directives

### Used to simplify and display static content

### Used to interact with user (via clicks for example)

## Generalizations

### Directives are a powerful way to create self-contained interactive components.

### AngularJS treats interactivity as a native component of HTML

#### Unlike jQuery which adds interactivity as a layer on top of HTML

# Services

## Purpose: When hardcoding data into a controller becomes to challenging (hundreds of entries, or changing data for example) it becomes better to read data from a live server by creating service

## 

# Routing

# putting it all together