Paul Beresuita





2 weeks













http://todomvc.com/



Why Angular?

<demo> <helloworld> </demo>



Best Practices

Project Structure

File Type

```
▼ 🗀 app
  ▼ 🗀 css
        📴 app.css
        bootstrap.css
        bootstrap.css.map
        bootstrap.min.css
        bootstrap-theme.css
        bootstrap-theme.css.map
        bootstrap-theme.min.css
  ▼ 🛅 html
        view1.html
        li view2.html
  ▼ 🗀 js
        app.config.js
        app.module.js
        idirective 1.js
        directive2.js
        service1.js
        service2.js
        uiew1Controller.js
        uiew2Controller.js
▶ ☐ libs
  index.html
```

File Type - Specific

```
▼ 🗀 app
  ▼ 🗀 css
        app.css
        bootstrap.css
        bootstrap.css.map
        bootstrap.min.css
        bootstrap-theme.css
        bootstrap-theme.css.map
        bootstrap-theme.min.css
  ▼ html
        view1.html
        li view2.html
  ▼ 🗀 js
     ▼ □ controllers
           ■ view1Controller.js
           view2Controller.js
     ▼ ☐ directives
           ig directive 1.js
           directive2.js
     service1.js
           service2.js
        app.config.js
        app.module.js
▶ ☐ libs
  index.html
```

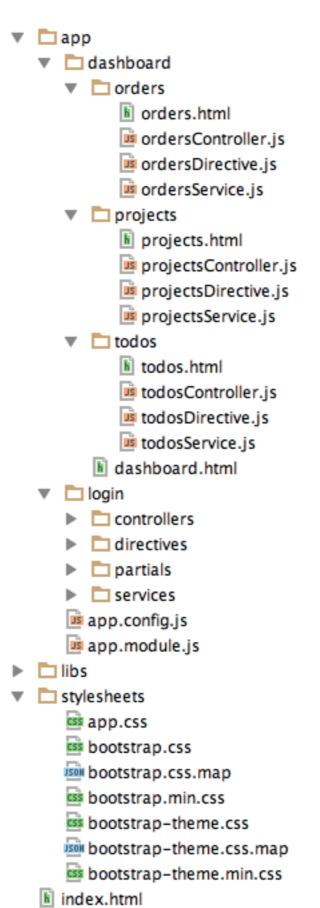
File Type - Feature

```
▼ 🗀 app
  ▼ □ dashboard
        li dashboard.html
        dashboardController.js
        dashboardDirective.js
        dashboardService.js
  ▼ 🗀 login
        login.html
        loginController.js
        loginDirective.js
        loginService.js
     app.config.js
     app.module.js
▶ ☐ libs
app.css
     bootstrap.css
     bootstrap.css.map
     bootstrap.min.css
     bootstrap-theme.css
     bootstrap-theme.css.map
     bootstrap-theme.min.css
  index.html
```

File Type - Feature & Type

```
▼ □ app
  ▼ □ dashboard
     ▼ □ controllers
           is dashboardController.js
     ▼ ☐ directives
           is dashboardDirective.js
     ▼ □ partials
          dashboard.html
     dashboardService.js
  ▼ login
     ▼ □ controllers
          is loginController.js
     ▼ ☐ directives
           is loginDirective.js
     ▼ □ partials
          login.html
     loginService.js
     app.config.js
     app.module.js
▶ ☐ libs
stylesheets
  index.html
```

File Type - Feature & Sub Features



Minification

<demo> <minification> </demo>



```
(function() {
     'use strict';
    angular.module('app').config(function($stateProvider, $urlRouterProvider) {
         $urlRouterProvider.otherwise("/main");
         $stateProvider
              .state('login', {
                  url: "/login",
                  templateUrl: "app/login/login.html",
                  controller: 'LoginController'
              })
              .state('main', {
                  url: "/main",
templateUrl: "app/main/main.html",
controller: 'MainController'
              })
    });
})();
```

Controllers

Use controllerAs Syntax

View

```
<!-- avoid -->
<div ng-controller="Customer">
    {{ name }}
</div>
```

```
<!-- recommended -->
<div ng-controller="Customer as customer">
    {{ customer.name }}
</div>
```

```
/* avoid */
function Customer($scope) {
    $scope.name = {};
    $scope.sendMessage = function() { };
}
```

Controller

```
/* recommended - but see next section */
function Customer() {
   this.name = {};
   this.sendMessage = function() { };
}
```

```
/* avoid */
function Customer() {
   this.name = {};
   this.sendMessage = function() { };
}
```

```
/* recommended */
function Customer() {
   var vm = this;
   vm.name = {};
   vm.sendMessage = function() { };
}
```

Bindable members on top

```
/* avoid */
function Sessions() {
    var vm = this;

    vm.gotoSession = function() {
        /* ... */
    };
    vm.refresh = function() {
        /* ... */
    };
    vm.search = function() {
        /* ... */
    };
    vm.sessions = [];
    vm.title = 'Sessions';
```

```
/* recommended */
function Sessions() {
   var vm = this;
   vm.gotoSession = gotoSession;
   vm.refresh = refresh;
   vm.search = search;
   vm.sessions = [];
   vm.title = 'Sessions';
   function gotoSession() {
    /* */
   }
   function refresh() {
     /* */
   function search() {
    /* */
   }
```

Don't pollute the controller

<demo> <pollutedController> </demo>

Directives

Use a prefix

1 file per directive

Manipulate the DOM in a directive



Angular 2.0 is written in TypeScript

Angular 2.0 has no controllers

No 2-way data binding (sort of)

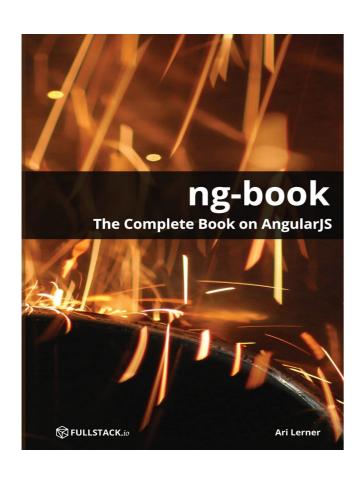
Resources





Angular Best Practices

- https://www.airpair.com/angularjs/posts/preparing-for-the-future-of-angularjs
- http://blog.mgechev.com/2015/04/06/angular2-first-impressions/
- http://angularjs.blogspot.com/2015/03/forms-in-angular-2.html (angular 2.0 forms)
- https://github.com/johnpapa/angular-styleguide#controllers



mg-newsletter



egghead.io