Angular Course

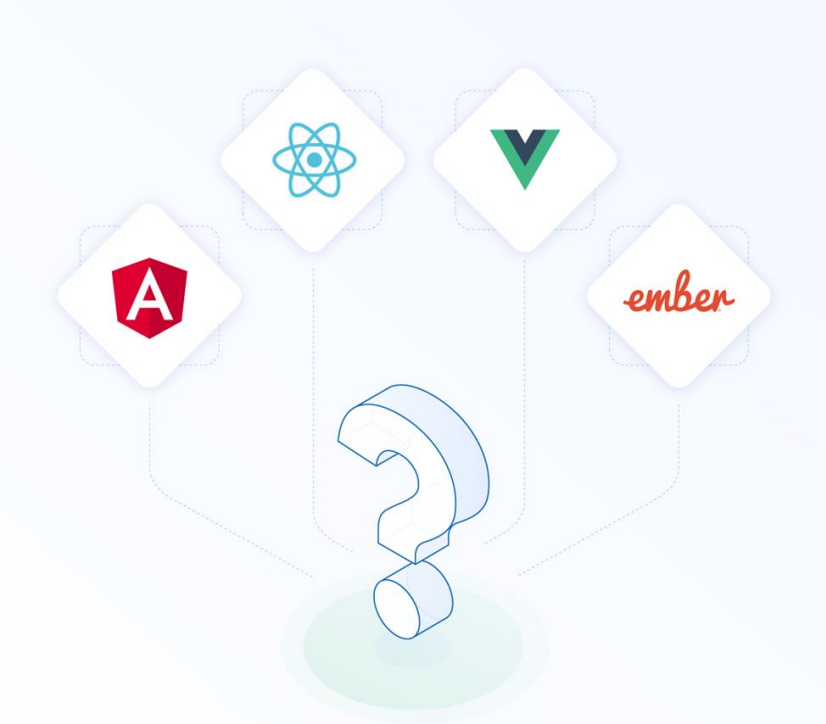
In this course you will get a hands-on of the major features of Angular

David Oliveira
Senior Consultant
23th January, 2018

Why Angular?

HITACHI
Inspire the Next

- Availability of Learning Resources
- Popularity
- Core Features
- Usability
- Ease of Integration (with Other Libraries)
- Outputs: Web, Mobile, Desktop



[https://thehappymindset.com/choose-best-front-end-framework/] toptal

Angular Basics

HITACHI
Inspire the Next

- Components
- Forms
- Directives
- Pipes
- Services
- Routing
- Observables

Angular Non-Basics

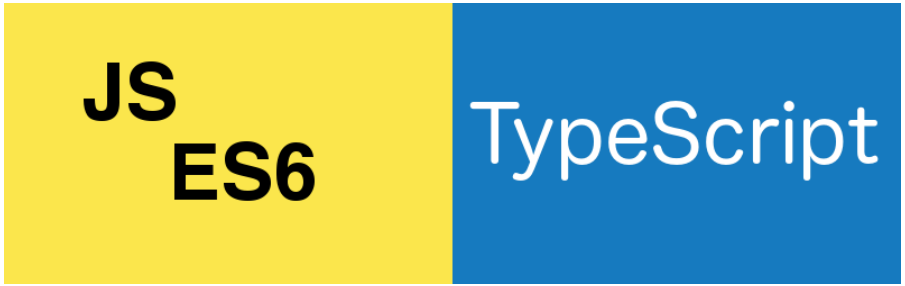
HITACHI
Inspire the Next

- App_Initializer
- Redux in Angular
- Unit Testing
- The mechanics of DOM updates in Angular (View encapsulation)
- Analyze an Angular App’s Bundle Size
- Etc.

TypeScript

HITACHI
Inspire the Next

```
function doSomething(someData)
{
    // do something
}
```



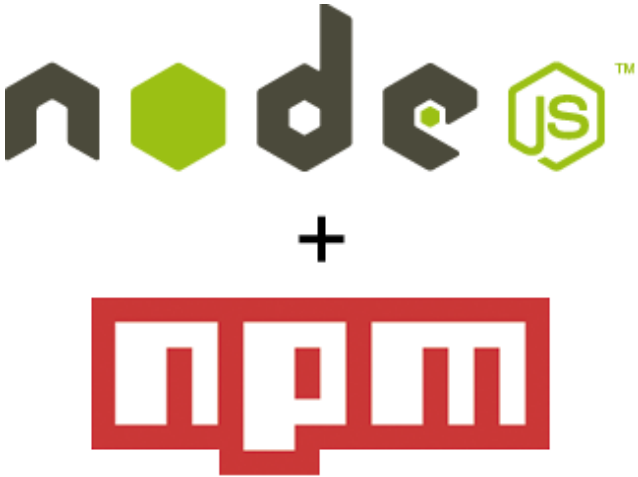
Type and Properties of someData?

Node JS + NPM

HITACHI
Inspire the Next

Node JS

Node.js® is a JavaScript runtime built on Chrome's V8 JavaScript engine. Node.js uses an event-driven, non-blocking I/O model that makes it lightweight and efficient.

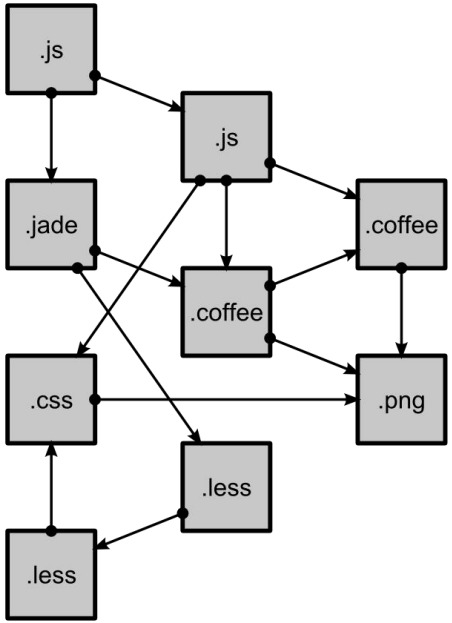


NPM

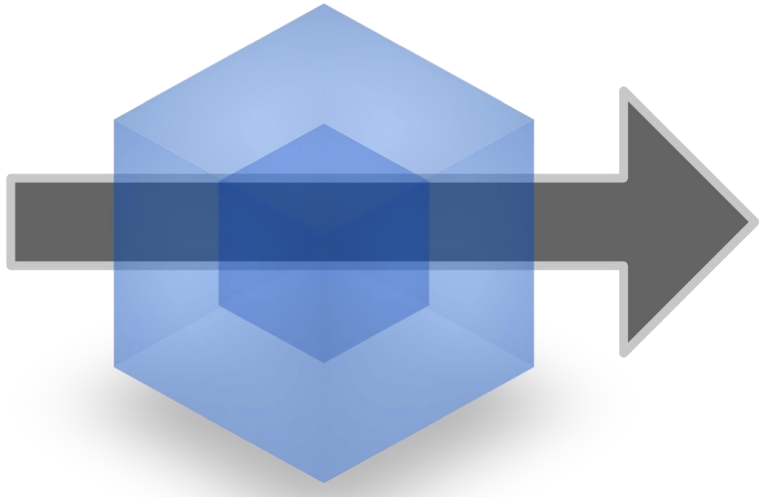
Node.js' package ecosystem, npm, is the largest ecosystem of open source libraries in the world.

Webpack

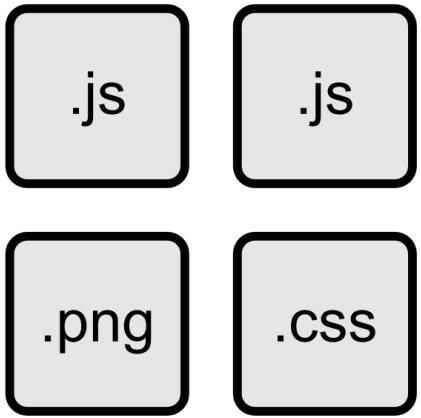
HITACHI
Inspire the Next



modules
with dependencies



webpack
MODULE BUNDLER



static
assets

Angular CLI

HITACHI
Inspire the Next

The Angular CLI is a tool to initialize, develop, scaffold and maintain Angular applications

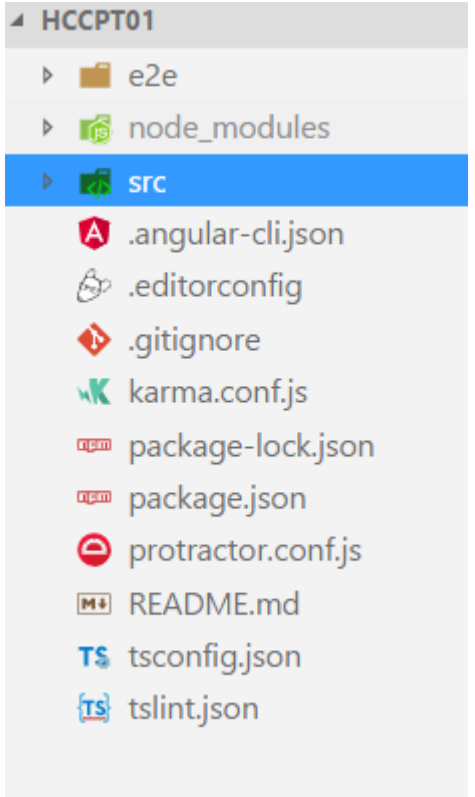
npm install -g @angular/cli

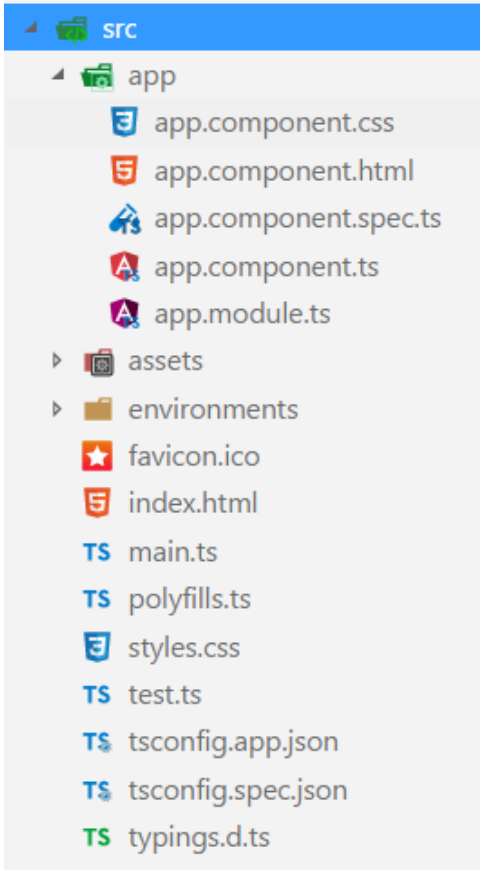
ng new hccpt
cd hccpt
ng serve

Navigate to http://localhost:4200/.
The app will automatically reload if you change any of the source files.

Angular Solution File Structure

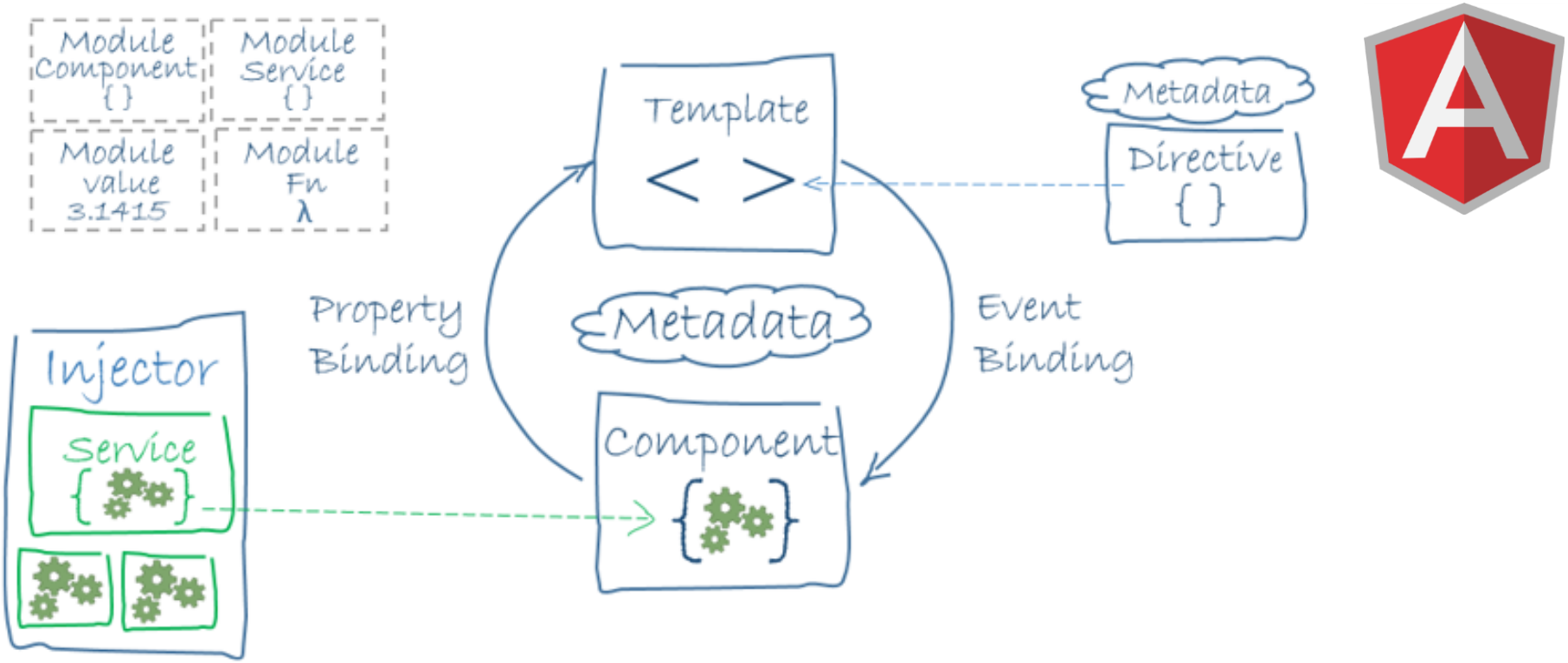
HITACHI
Inspire the Next





Angular Architecture Overview

HITACHI
Inspire the Next



Angular Course

In this course you will get a hands-on of the major features of Angular

David Oliveira
Senior Consultant
30th January, 2018

Our App to build with Angular

HITACHI
Inspire the Next

We will build an application to manage a Condominium Administration with the following features:

- Add/Edit Contacts (Owners and Suppliers)
- Add/Edit Payments
- Dashboard with Charts about Payments

Create Application

HITACHI
Inspire the Next

https://github.com/davidoliveira/AngularCourse

ng new KondominioApp --routing --style=scss

cd KondominioApp

npm install bootstrap@next --save

npm install --save @ng-bootstrap/ng-bootstrap

ng serve

https://github.com/angular/angular-cli/wiki/generate

https://github.com/angular/angular-cli/wiki/stories

Angular Modules

HITACHI
Inspire the Next

```
@NgModule({
  declarations: [
    AppComponent
  ],
  imports: [
    BrowserModule,
    AppRoutingModule,
    NgbModule.forRoot()
  ],
  providers: [],
  bootstrap: [AppComponent]
})
export class AppModule { }
```

https://angular.io/guide/ngmodules

https://angular.io/guide/ngmodule-vs-jsmodule

declarations—this application's lone component.

imports—import BrowserModule to have browser specific services such as DOM rendering, sanitization, and location.

providers—the service providers.

bootstrap—the *root* component that Angular creates and inserts into the index.html host web page.

A declarable can only belong to one module, so only declare it in one @NgModule. When you need it elsewhere, import the module that has the declarable you need in it.

Angular Modules

HITACHI
Inspire the Next

ng g component Login

ng g module Contacts --routing

- ng g component Index
- ng g component Detail

ng g module Payments --routing

- ng g component Index
- ng g component Detail

Angular Routing

HITACHI
Inspire the Next

https://angular.io/guide/router

https://angular.io/guide/lazy-loading-ngmodules

```
const routes: Routes = [
  { path: 'login', component: LoginComponent },
  {
    path: 'contacts',
    loadChildren: 'app/contacts/contacts.module#ContactsModule'
  },
  {
    path: 'payments',
    loadChildren: 'app/payments/payments.module#PaymentsModule'
  }
];

@NgModule({
  imports: [RouterModule.forRoot(routes)],
  exports: [RouterModule]
})
export class AppRoutingModule { }
```

Questions and Discussion

HITACHI

Inspire the Next