AngularTS

Building apps with AngularJS & TypeScript

TypeScript: Whats 'n' Whys

Integrating with JavaScript libraries

AngularJS: Whats 'n' Whys

Reference Project

In Practice

Dos 'n' Don'ts

TypeScript: Whats 'n' Whys

Integrating with JavaScript libraries

AngularJS: Whats 'n' Whys

Reference Project

In Practice

Dos 'n' Don'ts

TypeScript: Whats 'n' Whys

Whats

- High level language that compiles to JavaScript
- Object Oriented (or not)
- Strongly typed (or not)
- MS backed
- Open-source
- Focuses on tightly integrate existing JS libs
- Closures, Generics, Annotations and Interfaces
- String interpolation
- ECMA6 now, ECMA7 soon

TypeScript: Whats 'n' Whys

Whys

- We can go full-typed!
- All JS is valid TS
- We get to use newer ES features before browsers actually support it
- Compile-time exists!
- Way to bypass what stinks in JS
- We get to keep using our favorite JS libs
- Used and endorsed by Angular 2
- Arrow functions
- Porting from JavaScript is easy

TypeScript: Whats 'n' Whys -> Types

```
var isDone: boolean = false;
var lines: number = 42;
var name: string = "Anders";
var list: number[] = [1, 2, 3];
var list: Array<number> = [1, 2, 3];
enum Color {Red, Green, Blue}
function bigHorribleAlert(): void {
```

TypeScript: Whats 'n' Whys -> Functions

```
var f1 = function(i: number): number { return i * i; }
var f2 = function(i: number) { return i * i; }
var f3 = (i: number): number => { return i * i; }
var f4 = (i: number) => { return i * i; }
var f5 = (i: number) => i * i;
```

TypeScript: Whats 'n' Whys -> Interfaces

```
interface Person {
  age?: number;
var p: Person = { name: "Bobby", move: () => {} };
var validPerson: Person = { name: "Bobby", age: 42, move: () => {} };
var invalidPerson: Person = { name: "Bobby", age: true };
```

TypeScript: Whats 'n' Whys -> Classes

```
class Point {
  x: number;
  constructor(x: number, public y: number = 0) {
  static origin = new Point(0, 0);
```

TypeScript: Whats 'n' Whys

- Inheritance
- Generics
- String interpolation
- Modules

TypeScript: Whats 'n' Whys

Integrating with JavaScript libraries

AngularJS: Whats 'n' Whys

Reference Project

In Practice

Dos 'n' Don'ts

Integrating with JavaScript libraries

- Declaration files (.d.ts) are used to describe libraries
- Typings: TypeScript Definition Manager
 - Has a command line utility that works much like NPM or Bower
 - npm install typings --global
 - typings install angular --global --save
 - Use /// <reference path="/typings/index.d.ts"> to use the definitions

TypeScript: Whats 'n' Whys

Integrating with JavaScript libraries

AngularJS: Whats 'n' Whys

Reference Project

In Practice

Dos 'n' Don'ts

AngularJS: Whats 'n' Whys

Whats

- MV* Framework for the web.
- Data binding
- Dependency Injection
- Promises
- Directives
- Reusable components
- Localization
- Form validation

AngularJS: Whats 'n' Whys

Whys

- Single Page Web Applications are hard to implement; Angular makes it easy
- JS is hell; Angular helps keeping it in check
- Abstracts data-to-presentation conversion
- Enables componentization
- Facilitates separation between presentation and business logic

But HOW!?

Services & Factories

Controllers

Routing

Directives

Dependency Injection

Data Binding

TypeScript: Whats 'n' Whys

Integrating with JavaScript libraries

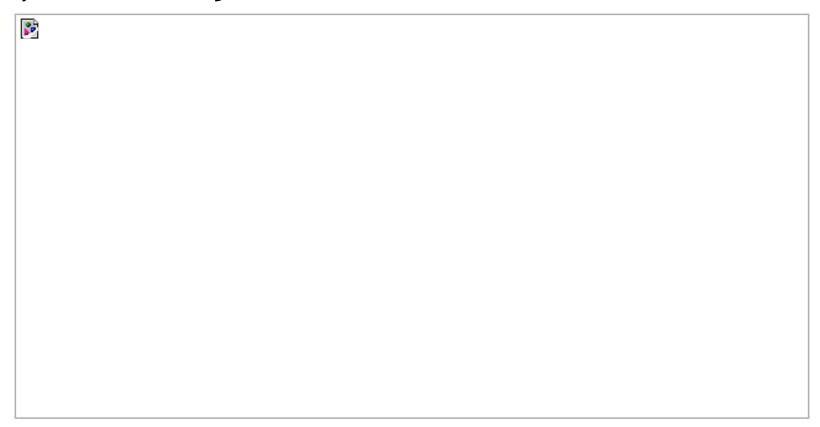
AngularJS: Whats 'n' Whys

Reference Project

In Practice

Dos 'n' Don'ts

Reference Project



TypeScript: Whats 'n' Whys

Integrating with JavaScript libraries

AngularJS: Whats 'n' Whys

Reference Project

In Practice

Dos 'n' Don'ts

In Practice

TypeScript: Whats 'n' Whys

Integrating with JavaScript libraries

AngularJS: Whats 'n' Whys

Reference Project

In Practice

Dos 'n' Don'ts

Dos 'n' Don'ts

Do

- Use 'controllerAs' syntax
- Use \$inject
- Use Angular services more broadly
- Componentize your application
- Use Angular's own services (\$timeout, \$interval, \$location)
- Use promises instead of callbacks

Dos 'n' Don'ts

Don't

- Overuse events
- Overuse \$watch
- Use \$rootScope as a global repository
- Manipulate DOM in controllers
- Use ng-init

TypeScript: Whats 'n' Whys

Integrating with JavaScript libraries

AngularJS: Whats 'n' Whys

Reference Project

In Practice

Dos 'n' Don'ts

AngularTS

Building apps with AngularJS & TypeScript

Bonus: Helpful Links

https://learnxinyminutes.com/docs/typescript/

https://www.typescriptlang.org/docs/handbook/writing-declaration-files.html