



Empowering Providers with
Innovative Healthcare Solutions

Angular Workshop

By: Kewal Zanzmeria
kewalz@dosepack.com
z.kewal@gmail.com

Why Angular?



- Angular 2 Is Easier
- TypeScript
- Familiarity
- Performance and Mobile
- Project Architecture and Maintenance

TypeScript



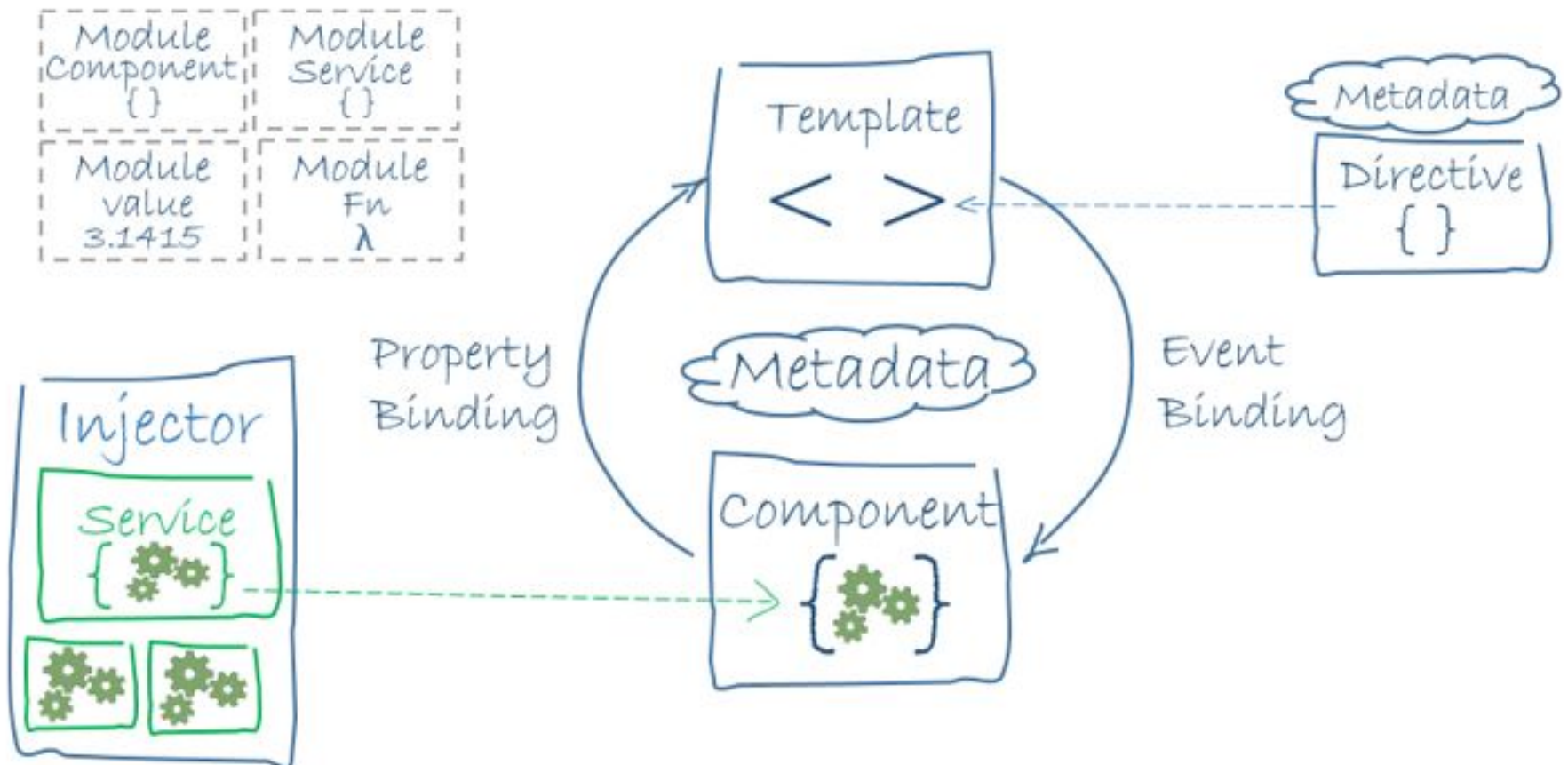
- const & let
- Multiline strings
- Modules
- Arrow functions
- Destructuring / Rest / Spread
- Classes
- ES7 -> Decorators

Angular Architecture



- Modules
- Components
- Templates
- Metadata
- Data binding
- Directives
- Services
- Dependency injection

Angular Architecture

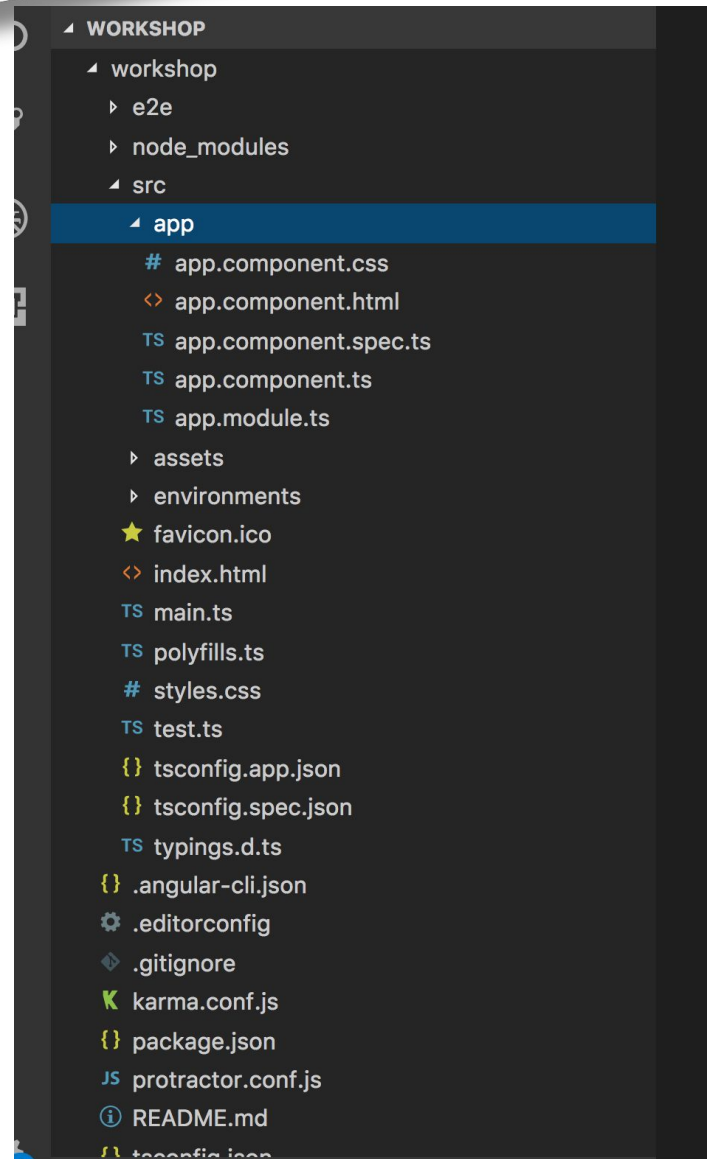


Angular CLI



-

Project Directory Tree



Setup & Bootstrapping App



- Index.html
- PlatformBrowserDynamic

Templates



Data Binding

- Interpolation {}
- Property binding []
- Event binding ()
- ngModel [()]

Templates



Pipes

- `{{ person.name | uppercase }}`
- `{{ person.dateOfBirth | date: 'shortDate' }}`

Templates



Template Reference Variables

- `<input type="text" [value]="person.name" #personName>`
- `Hello {{ personName.value }}!`

Templates



Inputs & Outputs

- `@Input() student: string;`
- `@Output() selectEvent : EventEmitter = new EventEmitter();`

Lifecycle Hooks



- `ngOnChanges`
- `ngOnInit`
- `ngAfterContentInit`
- `ngOnDestroy`

Directives



- structural directives: *ngFor and *ngIf
- Attribute directives: ngModel and ngClass

- @Injectable()
- Dependency injection is a coding pattern in which a class receives its dependencies from external sources rather than creating them itself.
- Injector: An injector maintains a container of service instances that it has previously created.
- Providers: A provider is something (i.e., a “recipe”) that can create or return a service, typically the service class itself.

Routing and navigation



- RouterModule
- Routes
- RouterOutlet
- Lazy Load
- Guard

Forms



- Reactive forms
- *ReactiveFormsModule*
- FormControl & FormGroup
- *FormBuilder*
- Template (formControlName directive)
- novalidate
- setValue & patchValue
- valueChanges
- Validators

Http and communication with server



- HttpClient
- Import HttpClientModule from `@angular/common/http`
- Get, post
- Type Checking responses
- Error Handling
- responseType (non json data)

Debugging application -



- Augury
- Console.log
- Chrome Debugger

Developer Tools & References



- IDE - VSCode (<https://code.visualstudio.com>)
- Language - Typescript (<https://www.typescriptlang.org>)
- Framework - Angular (<https://angular.io>)
- Online IDE - Stack Blitz (<https://stackblitz.com>)
- Command Line Tool - Angular CLI (<https://cli.angular.io>)
- Reusable Components - Angular Material (<https://material.angular.io>)
- Linter - TSLint (<https://palantir.github.io/tslint>)
- Debug & Inspection Tool - Augury (<https://augury.angular.io>)

Thank You.



Empowering Providers with
Innovative Healthcare Solutions